# 1. Fundamentos de Aprendizado de Máquina e Inteligência Artificial

## 1.1 Inteligência Artificial (IA)

IA é um campo amplo que abrange o desenvolvimento de sistemas inteligentes capazes de executar tarefas que normalmente exigem inteligência humana, como percepção, raciocínio, aprendizado, resolução de problemas e tomada de decisões. IA serve como um termo abrangente para diversas técnicas e abordagens, incluindo aprendizado de máquina, aprendizado profundo e IA generativa, entre outras.

## 1.2 Machine Learning (ML)

ML é um tipo de IA para compreender e desenvolver métodos que possibilitam o aprendizado por máquinas. Esses métodos usam dados para melhorar o desempenho do computador em um conjunto de tarefas.

## 1.3 Deep Learning (DL)

O aprendizado profundo utiliza o conceito de neurônios e sinapses, semelhante à estrutura do nosso cérebro. Um exemplo de aplicativo de aprendizado profundo é o **Amazon Rekognition**, que pode analisar milhões de imagens, vídeos em streaming e armazenados em segundos.

## 1.4 IA Generativa

A IA Generativa é um subconjunto do aprendizado profundo, pois pode adaptar modelos construídos com esse recurso, mas sem retreinamento ou ajuste fino.

Os sistemas de IA Generativa são capazes de gerar novos dados com base nos padrões e estruturas aprendidos com os dados de treinamento.

# 2. Fundamentos do Machine Learning

ML é um subconjunto da IA que se concentra no desenvolvimento de algoritmos e modelos estatísticos para que sistemas computacionais possam aprender com dados e fazer previsões ou tomar decisões sem serem explicitamente programados. Modelos de ML aprendem padrões e relacionamentos a partir de dados, em vez de depender de regras codificadas para obter instruções. Esses modelos são treinados em grandes conjuntos de dados, e sua precisão e desempenho melhoram com o tempo, à medida que processam mais dados.

A construção de um modelo de aprendizado de máquina envolve coleta e preparação de dados, seleção de um algoritmo apropriado, treinamento do modelo nos dados preparados e avaliação de seu desempenho por meio de testes e iterações.



## 2.1 Training data

O processo de aprendizado de máquina começa com a coleta e o processamento de dados de treinamento. Dados ruins costumam ser chamados de garbage in, garbage out, e, portanto, um modelo de ML é tão bom quanto os dados usados ​​para treiná-lo. Embora a preparação e o processamento de dados sejam, às vezes, um processo rotineiro, é sem dúvida a etapa mais crítica para fazer com que todo o modelo funcione conforme o esperado ou para prejudicar seu desempenho.

### 2.1.1 Dados rotulados

Dados rotulados são um conjunto de dados em que cada instância ou exemplo é acompanhado por um rótulo ou variável de destino que representa a saída ou classificação desejada. Esses rótulos são normalmente fornecidos por especialistas humanos ou obtidos por meio de um processo confiável.

Exemplo: Em uma tarefa de classificação de imagens, os dados rotulados consistiriam em imagens juntamente com seus rótulos de classe correspondentes (por exemplo, gato, cachorro, carro).

### 2.1.2 Dados Não Rotulados

Dados não rotulados são conjuntos de dados em que as instâncias ou exemplos não possuem rótulos ou variáveis-alvo associados. Os dados consistem apenas em recursos de entrada, sem nenhuma saída ou classificação correspondente.

Exemplo: Uma coleção de imagens sem rótulos ou anotações.

### 2.1.3 Dados estruturados

Dados estruturados referem-se a dados organizados e formatados de maneira predefinida, normalmente na forma de tabelas ou bancos de dados com linhas e colunas. Esse tipo de dado é adequado para algoritmos tradicionais de aprendizado de máquina que exigem recursos e rótulos bem definidos. Os seguintes são os tipos de dados estruturados:

**Dados tabulares**: incluem dados armazenados em planilhas, bancos de dados ou arquivos CSV, com linhas representando instâncias e colunas representando recursos ou atributos.

**Dados de séries temporais**: esse tipo de dado consiste em sequências de valores medidos em pontos sucessivos no tempo, como preços de ações, leituras de sensores ou dados meteorológicos.

### 2.1.4 Dados não estruturados

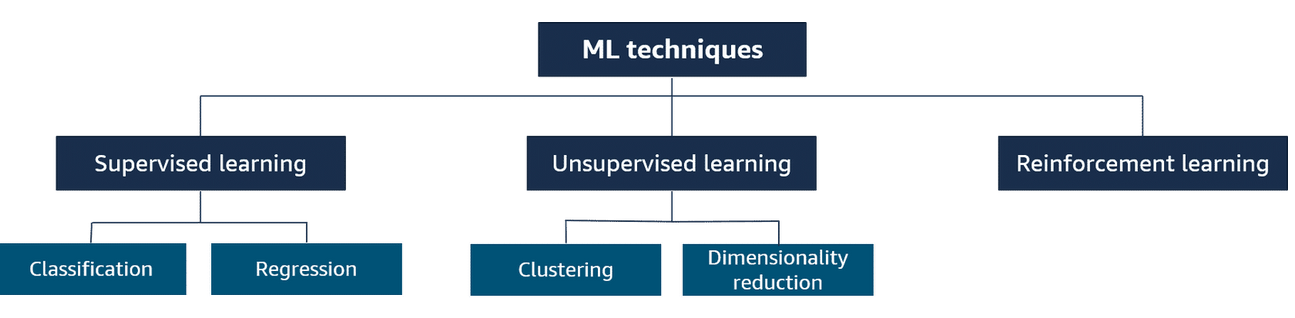
Dados não estruturados são dados que não possuem uma estrutura ou formato predefinido, como texto, imagens, áudio e vídeo. Esse tipo de dado requer técnicas de aprendizado de máquina mais avançadas para extrair padrões e insights significativos.

**Dados de texto**: incluem documentos, artigos, postagens em mídias sociais e outros dados textuais.

**Dados de imagem**: incluem imagens digitais, fotografias e quadros de vídeo.

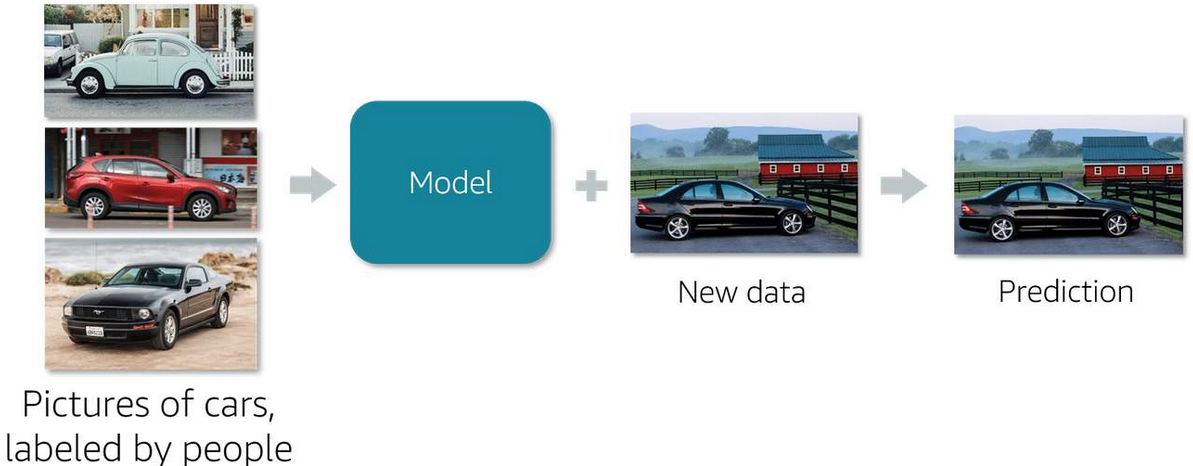
## 2.2 Processos do Machine Learning

Os dados de treinamento compilados são inseridos em algoritmos de aprendizado de máquina. O processo de aprendizado de ML é tradicionalmente dividido em três grandes categorias: aprendizado supervisionado, aprendizado não supervisionado e aprendizado por reforço.



2.2.1 Aprendizado supervisionado

Os algoritmos são treinados com base em dados rotulados. O objetivo é aprender uma função de mapeamento que possa prever a saída para novos dados de entrada não identificados. Chama-se aprendizado supervisionado porque é necessário um supervisor. O supervisor é rotulado como dados de treinamento. Como qualquer aluno, um algoritmo supervisionado precisa aprender por meio de exemplos. Essencialmente, esse tipo de algoritmo usa dados de treinamento para ajudar a determinar os padrões e as relações entre as entradas e saídas. Por exemplo, imagens de carros rotulados por pessoas como carros são fornecidas ao modelo. Então, quando o modelo recebe uma nova imagem de um carro não rotulado, o modelo pode prever que se trata de um carro.



Tipos de ML supervisionado

**Classificação** é uma técnica de aprendizado supervisionado usada para atribuir rótulos ou categorias a novas instâncias de dados não identificadas com base em um modelo treinado. O modelo é treinado em um conjunto de dados rotulado, onde cada instância já está atribuída a uma classe ou categoria conhecida. O objetivo da classificação é aprender padrões a partir dos dados de treinamento e usá-los para prever a classe ou categoria para novas instâncias de dados não rotuladas.

Os casos de uso incluem:

Detecção de fraudes; Classificação de imagens; Retenção de clientes; diagnóstico

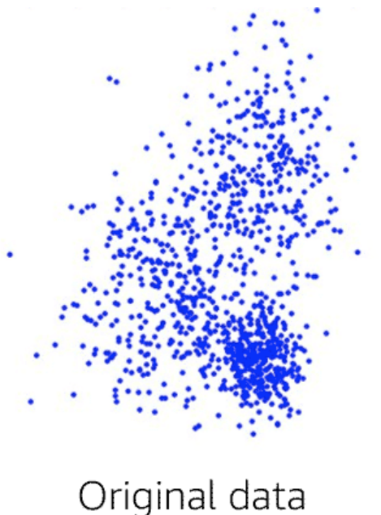
**Regressão** é uma técnica de aprendizado supervisionado usada para prever valores contínuos ou numéricos com base em uma ou mais variáveis de entrada. Ela é usada para modelar a relação entre uma variável dependente (o valor a ser previsto) e uma ou mais variáveis independentes (as características ou entradas usadas para a previsão).

Os casos de uso incluem:

Previsão de popularidade de publicidade; Previsão do tempo; Previsão de mercado; estimativa da expectativa de vida; Previsão do crescimento populacional

2.2.2 Aprendizado não supervisionado

Refere-se a algoritmos que aprendem com dados não rotulados. O objetivo é descobrir padrões, estruturas ou relacionamentos inerentes aos dados de entrada. Nesse tipo de aprendizado, a máquina precisa descobrir e criar os rótulos por si só. Esses modelos usam os dados apresentados para detectar propriedades emergentes de todo o conjunto de dados e, em seguida, construir padrões.



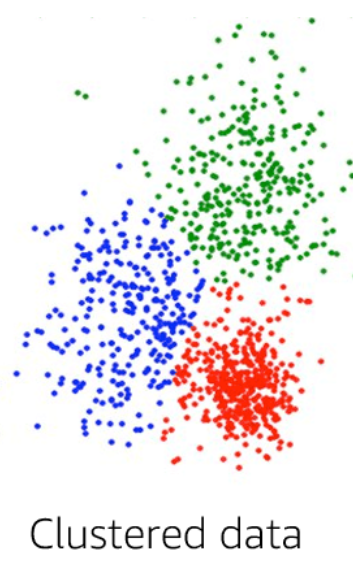
Tipos de ML não supervisionado

**Agrupamento**. Esse tipo de algoritmo agrupa dados em diferentes clusters com base em características semelhantes ou distâncias entre os pontos de dados para melhor compreender os atributos de um cluster específico.

Por exemplo, ao analisar os hábitos de compra dos clientes, um algoritmo não supervisionado pode identificar uma empresa como grande ou pequena.

Os casos de uso incluem:

Segmentação de clientes; Marketing direcionado; Sistemas recomendados



**Redução de dimensionalidade** é uma técnica de aprendizado não supervisionado usada para reduzir o número de características ou dimensões em um conjunto de dados, preservando as informações ou padrões mais importantes.

Os casos de uso incluem:

Visualização de big data; Compressão significativa; descoberta de estrutura; Elicitação de características

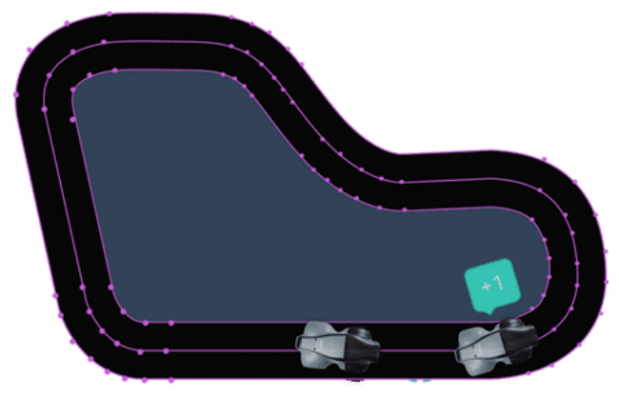
2.2.3 No aprendizado por reforço

A máquina recebe apenas uma pontuação de desempenho como orientação, e no aprendizado semi-supervisionado, apenas uma parte dos dados de treinamento é rotulada. O feedback é fornecido na forma de recompensas ou penalidades por suas ações, e a máquina aprende com esse feedback para aprimorar sua tomada de decisão ao longo do tempo.

No aprendizado por reforço, um agente aprende continuamente por tentativa e erro à medida que interage em um ambiente. O aprendizado por reforço é amplamente útil quando a recompensa de um resultado desejado é conhecida, mas o caminho para alcançá-lo não é — e esse caminho requer muita tentativa e erro para ser descoberto.

Por exemplo, no simulador AWS DeepRacer, o agente é o carro virtual e o ambiente é uma pista de corrida virtual. As ações são entradas de aceleração e direção para o carro. O objetivo é completar a pista o mais rápido possível e sem se desviar da pista.

O carro precisa aprender o comportamento de direção desejado para atingir o objetivo de completar a pista. Para aprender isso, recompensas são usadas para incentivar o modelo a aprender o comportamento de direção desejado.



## 2.3 Inferência

Após o treinamento do modelo, é hora de iniciar o processo de utilização das informações que o modelo aprendeu para fazer previsões ou tomar decisões. Isso se chama inferência.

Existem dois tipos principais de inferência em aprendizado de máquina: inferência em lote e inferência em tempo real.

**A inferência em lote** ocorre quando o computador coleta uma grande quantidade de dados, como imagens ou texto, e os analisa de uma só vez para fornecer um conjunto de resultados. Esse tipo de inferência é **frequentemente usado para tarefas como análise de dados**, em que a velocidade do processo de tomada de decisão não é tão crucial quanto a precisão dos resultados.

**A inferência em tempo real** ocorre quando o computador precisa tomar decisões rapidamente, em resposta a novas informações conforme elas chegam. Isso é importante para aplicações em que a tomada de decisão imediata é crucial, como em chatbots ou carros autônomos. **O computador precisa processar os dados recebidos e tomar uma decisão quase instantaneamente, sem perder tempo analisando um grande conjunto de dados.**

## 2.4 Quando IA e ML são soluções adequadas

A IA é uma boa escolha para os seguintes casos de uso:

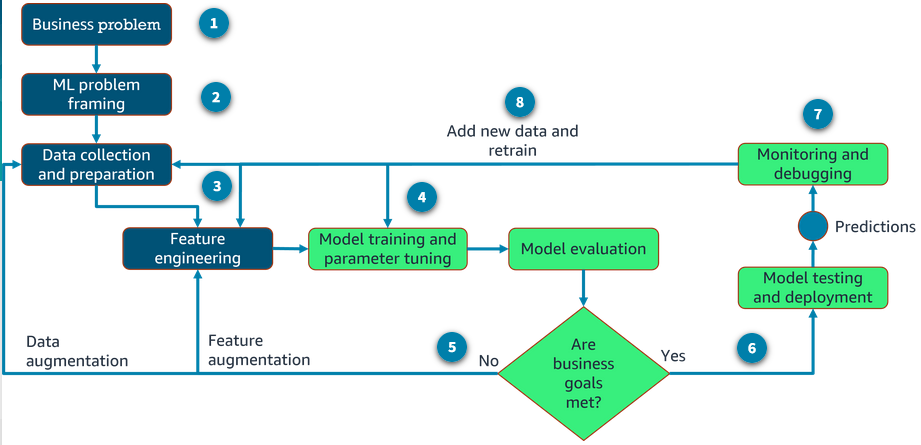
**Codificar as regras é desafiador:** Muitas tarefas humanas não podem ser resolvidas adequadamente usando soluções simples e baseadas em regras. Veja a filtragem de spam, por exemplo. Determinar se um e-mail recebido é legítimo ou spam é uma tarefa complexa que nem sempre pode ser resolvida de forma eficaz por meio de um conjunto de regras predefinidas. Há muitas variáveis em jogo. Quando as regras dependem de muitos fatores, se sobrepõem ou precisam ser ajustadas com precisão, torna-se difícil para os humanos codificá-las com precisão. O ML pode ser usado para resolver esse tipo de problema de forma eficaz.

**A escala do projeto é desafiadora:** No exemplo da filtragem de spam, um humano pode ser capaz de analisar algumas centenas de e-mails e decidir se são spam ou não. No entanto, escalar essa tarefa para examinar milhões de e-mails seria tedioso e ineficiente. Soluções de ML são adequadas para problemas de grande escala como este.

## 2.5 Abordagem alternativa para IA e ML

Por exemplo, você não precisa de ML se puder determinar um valor alvo usando regras simples, cálculos ou etapas predeterminadas. Você pode programar as etapas sem precisar de nenhum aprendizado orientado por dados.

## 2.6 Ciclo de Vida de Desenvolvimento de Aprendizado de Máquina



### 2.6.1 Definir metas de negócios

O ML começa com um objetivo de negócios. As partes interessadas do negócio definem o valor, o orçamento e os critérios de sucesso.

### 2.6.2 Enquadramento do problema de ML

A formulação do problema envolve articular o problema de negócios e convertê-lo em um problema de aprendizado de máquina.

O cientista de dados, os engenheiros de dados e os arquitetos de ML trabalham com os especialistas no assunto (SMEs) da linha de negócios para determinar se é apropriado usar ML para resolver o problema de negócios. Nesta fase, as equipes podem trabalhar na descoberta. Elas determinarão se possuem os dados, as habilidades e outros recursos adequados para entregar a solução de negócios com sucesso.

### 2.6.3 Processamento de dados

Após a formulação do problema, a próxima fase é a preparação e o pré-processamento dos dados.

Para treinar um modelo de ML preciso, os desenvolvedores usam o processamento de dados para convertê-los em um formato utilizável.

As etapas de processamento de dados incluem coleta e integração de dados, pré-processamento e visualização de dados e engenharia de recursos.

A coleta e a integração de dados garantem que os dados brutos estejam em um local centralizado e acessível. O pré-processamento e a visualização de dados envolvem a transformação dos dados brutos em um formato compreensível. A engenharia de recursos é o processo de criação, transformação, extração e seleção de variáveis dos dados.

### 2.6.4 Desenvolvimento do modelo

O desenvolvimento do modelo consiste no treinamento, ajuste e avaliação do modelo.

É um processo iterativo que pode ser executado diversas vezes ao longo deste fluxo de trabalho.

Inicialmente, após o treinamento, o modelo não produzirá os resultados esperados. Portanto, os desenvolvedores realizarão engenharia de recursos adicional e ajustarão os hiper parâmetros do modelo antes do retreinamento.

### 2.6.5 Retreinamento

Se o modelo não atender às metas de negócios, é necessário analisar novamente os dados e os recursos para identificar maneiras de aprimorá-lo.

### 2.6.6 Implantação

Se os resultados forem satisfatórios, o modelo é implantado em produção. O modelo agora está pronto para fazer previsões e inferências.

### 2.6.7 Monitoramento

O sistema de monitoramento do modelo garante que o modelo mantenha o nível desejado de desempenho por meio de detecção e mitigação precoces. O monitoramento também ajuda a depurar problemas e a entender o comportamento do modelo.

### 2.6.8 Iterações

O ciclo de vida do aprendizado de máquina é um processo iterativo. O modelo é continuamente aprimorado e refinado à medida que novos dados se tornam disponíveis ou os requisitos mudam. Essa natureza iterativa ajuda a garantir que o modelo permaneça preciso e relevante ao longo do tempo.

## 2.7 Desenvolvendo Soluções de ML com o Amazon SageMaker

### 2.7.1 Coletando, analisando e preparando seus dados

O Amazon SageMaker Data Wrangler é uma ferramenta de baixo código e sem código (LCNC). Ele fornece uma solução completa para importar, preparar, transformar, caracterizar e analisar dados usando uma interface web. Os clientes podem adicionar seus próprios scripts e transformações em Python para personalizar os fluxos de trabalho.

Para usuários mais avançados e preparação de dados em escala, o Amazon SageMaker Studio Classic vem com integração de sessões interativas do Amazon EMR e do AWS Glue para lidar com a preparação de dados em larga escala e fluxos de trabalho de aprendizado de máquina em seu notebook do SageMaker Studio Classic.

Por fim, usando a API de Processamento do SageMaker, os clientes podem executar scripts e notebooks para processar, transformar e analisar conjuntos de dados. Eles também podem usar várias estruturas de ML, como scikit-learn, MXNet ou PyTorch, enquanto se beneficiam de ambientes de aprendizado de máquina totalmente gerenciados.

Ao final desta etapa, os clientes geralmente obtêm recursos para definir o modelo e os dados para o treinamento deste modelo.

### 2.7.2 Gerenciando Recursos

Opções do SageMaker Feature Store, incluindo ingestão de dados, armazenamento e disponibilização.

O Amazon SageMaker Feature Store auxilia cientistas de dados, engenheiros de machine learning e clínicos gerais a criar, compartilhar e gerenciar recursos para desenvolvimento de ML.

Os recursos armazenados no armazenamento podem ser recuperados e enriquecidos antes de serem disponibilizados aos modelos de ML para inferências.

### 2.7.3 Treinamento e avaliação de modelos

O SageMaker fornece um recurso de tarefa de treinamento para treinar e implantar modelos usando algoritmos integrados ou algoritmos personalizados.

O SageMaker inicia as instâncias de computação de ML e usa o código de treinamento e o conjunto de dados de treinamento para treinar o modelo. Ele salva os artefatos do modelo resultantes em um bucket S3, que pode ser usado posteriormente para inferência.

Os clientes que buscam uma opção LCNC podem usar o Amazon SageMaker Canvas. Com o SageMaker Canvas, eles podem usar aprendizado de máquina para gerar previsões sem a necessidade de escrever nenhum código.

O Amazon SageMaker JumpStart fornece modelos pré-treinados, de código aberto e Docker, que os clientes podem usar para uma ampla gama de tipos de problemas.

Modelos pré-treinados exigem o mínimo de esforço e são modelos prontos para implantação ou para ajuste fino e implantação usando o SageMaker JumpStart.

Modelos integrados disponíveis no SageMaker exigem mais esforço e escalabilidade se o conjunto de dados for grande e recursos significativos forem necessários para treinar e implantar o modelo.

Se não houver uma solução integrada que funcione, tente desenvolver uma que utilize imagens pré-criadas para frameworks de aprendizado de máquina e aprendizado profundo para frameworks suportados, como scikit-learn, TensorFlow, PyTorch, MXNet ou Chainer.

Você pode criar sua própria imagem Docker personalizada, configurada para instalar os pacotes ou softwares necessários.

Algoritmos integrados do SageMaker

* Aprendizado supervisionado
* Aprendizado não supervisionado
* Processamento de imagens
* Análise de texto

### 2.7.4 Avaliação do modelo

Os clientes podem usar o Amazon SageMaker Experiments para experimentar múltiplas combinações de dados, algoritmos e parâmetros, enquanto observam o impacto de alterações incrementais na precisão do modelo.

O ajuste de hiper parâmetros é uma maneira de encontrar a melhor versão dos seus modelos. O Ajuste Automático de Modelos do Amazon SageMaker faz isso executando vários trabalhos com diferentes hiper parâmetros em combinação e medindo cada um deles por uma métrica escolhida por você.

### 2.7.5 Implantação

Com o SageMaker, os clientes podem implantar seus modelos de ML para fazer previsões, também conhecido como inferência. O SageMaker oferece uma ampla seleção de infraestrutura de ML e opções de implantação de modelos para ajudar a atender a todas as suas necessidades de inferência de ML.

### 2.7.6 Monitoramento

Com o Amazon SageMaker Model Monitor, os clientes podem observar a qualidade dos modelos do SageMaker ML em produção. Eles podem configurar o monitoramento contínuo ou o monitoramento conforme o cronograma. O SageMaker Model Monitor ajuda a manter a qualidade do modelo detectando violações dos limites definidos pelo usuário para qualidade de dados, qualidade do modelo, desvio de viés e desvio de atribuição de recursos.

## 2.8 Avaliação de Desempenho de Modelos de Aprendizado de Máquina

### 2.8.1 Conjuntos de dados de avaliação do modelo

A avaliação ocorre após o treinamento do modelo. Os dados utilizados são particionados em três partes: conjunto de treinamento, conjunto de validação e conjunto de teste. O conjunto de treinamento é usado para treinar o modelo. Os conjuntos de validação e teste são os que você usará para avaliar o desempenho do modelo treinado.

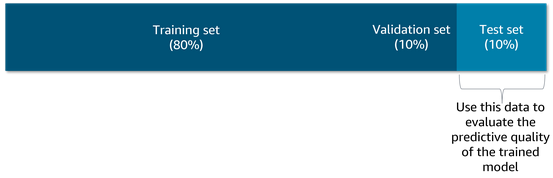
Conjunto de Validação

Para começar a avaliar como o modelo responde em um ambiente sem treinamento, comece analisando os dados que foram separados como conjunto de validação. Você precisa garantir que o modelo generalize para dados que ainda não viu. O modelo ainda precisa ser aprimorado antes de determinar que está pronto para produção.



Conjunto de teste

Depois de aprimorar o modelo usando os dados de validação, você estará pronto para testá-lo uma última vez para garantir que sua qualidade preditiva atenda aos seus padrões.

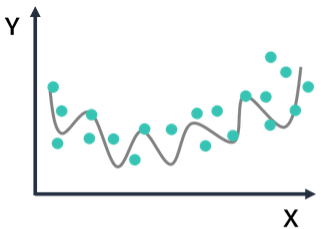


### 2.8.2 Ajuste do Modelo

O ajuste do modelo é importante para entender a causa raiz da baixa precisão do modelo. Essa compreensão o guiará para tomar medidas corretivas. Você pode determinar se um modelo preditivo está subajustando ou superajustando os dados de treinamento observando o erro de predição nos dados de treinamento e nos dados de avaliação.

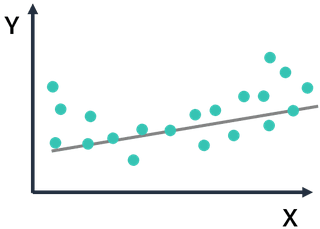
Superajuste

Superajuste ocorre quando o modelo tem um bom desempenho nos dados de treinamento, mas não nos dados de avaliação. Isso ocorre porque o modelo memorizou os dados que viu e não consegue generalizar para exemplos não vistos.



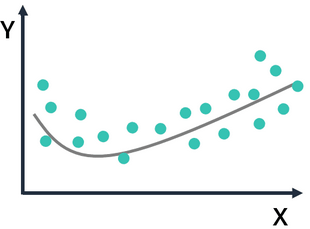
Subajuste

Subajuste ocorre quando o modelo tem um desempenho ruim nos dados de treinamento. Isso ocorre porque o modelo não consegue capturar a relação entre os exemplos de entrada (geralmente chamados de X) e os valores-alvo (geralmente chamados de Y). Se o seu modelo estiver subajustando e apresentando um desempenho ruim nos dados de treinamento, pode ser que ele seja muito simples (os recursos de entrada não sejam expressivos o suficiente) para descrever bem o alvo.



Balanceado

O modelo é balanceado quando não está superajustado ou subajustado aos dados de treinamento.



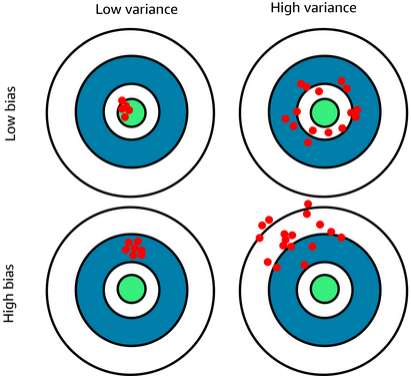
### 2.8.3 Viés e variância

Ao avaliar modelos, tanto o viés quanto a variância contribuem para os erros que o modelo comete em dados não observados, o que afeta sua generalização.

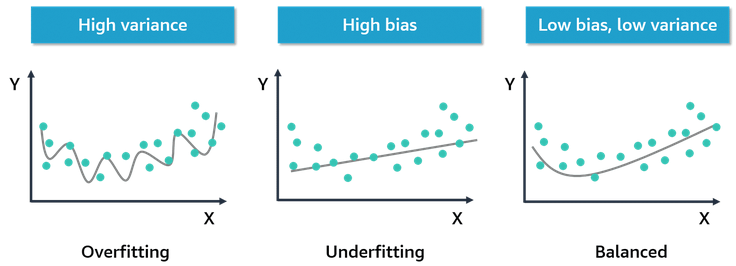
Um alvo é uma boa analogia porque, em termos gerais, o centro do alvo é para onde você mira seus dardos. O centro do alvo, nessa situação, é o rótulo ou alvo — ele prevê o valor do seu modelo — e cada ponto é um resultado que seu modelo produziu durante o treinamento.

Em ML, o algoritmo ideal tem baixo viés e pode modelar com precisão a relação real. O algoritmo ideal também tem baixa variabilidade, produzindo previsões consistentes em diferentes conjuntos de dados.

Pense no viés como a lacuna entre o valor previsto e o valor real, enquanto a variância descreve a dispersão dos seus valores previstos.



Modelos balanceados apresentam baixo viés e baixa variância



### 2.8.4 Problemas de classificação e regressão

A forma como você avalia um modelo de aprendizado de máquina depende do tipo de problema de ML com o qual está trabalhando

### 2.8.5 Métricas de classificação

Acurácia

Precisão

Recall

F1

AUC-ROC

### 2.8.6 Métricas de problemas de classificação

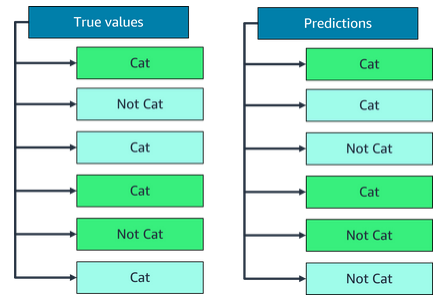
Exemplo de classificação

A seguir, apresentamos um problema de classificação binária em que um modelo de reconhecimento de imagem rotula os dados como "cat" ou "not cat".

Etapa 1: Envie as observações retidas, cujos valores-alvo você conhece, para o modelo.

Etapa 2: Compare as previsões retornadas pelo modelo com o valor alvo conhecido.

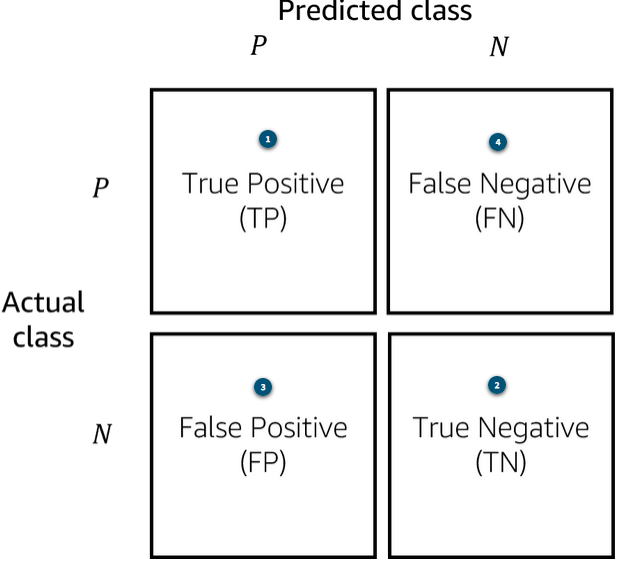
Etapa final: Calcule uma métrica de resumo que mostre a correspondência entre os valores previstos e verdadeiros.



### 2.8.7 Matriz de confusão

Uma matriz de confusão pode ajudar a classificar por que e como um modelo erra. Ela é o bloco de construção para executar esses tipos de avaliações de modelo para problemas de classificação. Revise o gráfico a seguir, que é uma matriz de confusão para o exemplo de reconhecimento de imagem. A matriz fornece uma comparação de alto nível de como as classes previstas corresponderam às classes reais.

Após a aplicação do modelo aos dados de teste, cada uma das quatro caixas na matriz incluirá um número agregado de ocorrências únicas de verdadeiros positivos, falsos positivos, falsos negativos e verdadeiros negativos.



1 Verdadeiro positivo (TP)

Se o rótulo ou classe real for "gato", que é identificado como "P" para positivo na matriz de confusão, e o rótulo ou classe previsto também for "gato", então você tem um resultado verdadeiro positivo. Este é um bom resultado para o seu modelo.

2 Verdadeiro Negativo (TN)

Da mesma forma, se você tiver um rótulo real de "não gato", identificado como "N" para negativo na matriz de confusão, e o rótulo ou classe previsto também for "não gato", então você tem um verdadeiro negativo. Este também é um bom resultado para o seu modelo. Em ambos os casos, seu modelo previu o resultado correto ao usar os dados de teste.

3 Falso Positivo (FP)

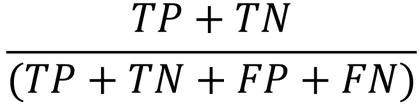
Isso é menos que o ideal e ocorre quando a classe real é negativa, então "não gato", mas a classe prevista é positiva, então "gato". Isso é chamado de falso positivo porque a previsão é positiva, mas incorreta.

4 Falso Negativo (FN)

Isso também é menos que o ideal. Um falso negativo ocorre quando a classe real é positiva, então "gato", mas a classe prevista é negativa, então "não gato".

### 2.8.8 Acurácia

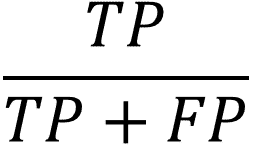
Para calcular a precisão do modelo, também conhecida como sua pontuação, some as previsões corretas e divida esse número pelo número total de previsões.



Embora a precisão seja uma métrica amplamente utilizada para problemas de classificação, ela apresenta limitações. Essa métrica é menos eficaz quando há muitos casos verdadeiros negativos em seu conjunto de dados. É por isso que duas outras métricas são frequentemente usadas nessas situações: precisão e recall.

### 2.8.9 Precisão

A precisão remove as previsões negativas do cenário. A precisão é a proporção de previsões positivas que estão realmente corretas. Você pode calculá-la dividindo a contagem de verdadeiros positivos pelo número total de positivos.



Quando o custo de falsos positivos é alto em sua situação empresarial específica, a precisão pode ser uma boa métrica. Pense em um modelo de classificação que identifique e-mails como spam ou não. Nesse caso, você não quer que seu modelo rotule um e-mail legítimo como spam e impeça seus usuários de vê-lo.

### 2.8.10 Recall

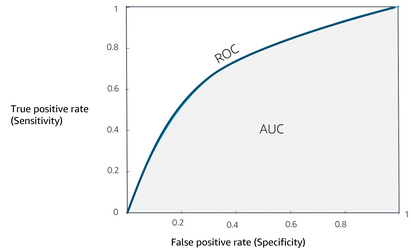
Além da precisão, há também o recall (ou sensibilidade). No recall, você está observando a proporção de conjuntos corretos que são identificados como positivos. O recall é calculado dividindo a contagem de verdadeiros positivos pela soma dos verdadeiros positivos e falsos negativos. Observando essa proporção, você tem uma ideia da eficiência do algoritmo na detecção de, por exemplo, gatos.



Pense em um modelo que precisa prever se um paciente tem uma doença terminal ou não. Nesse caso, usar a precisão como métrica de avaliação não leva em conta os falsos negativos no modelo. É extremamente importante e vital para o sucesso do modelo que ele não gere resultados falsos negativos. Um falso negativo seria não identificar um paciente como portador de uma doença terminal quando, na verdade, ele tem uma doença terminal. Nessa situação, a recordação é uma métrica mais adequada.

### 2.8.11 AUC-ROC

A área sob a curva do operador receptor (AUC-ROC) é outra métrica de avaliação. A ROC é uma curva de probabilidade e a AUC representa o grau ou medida de separabilidade.



Em geral, a AUC-ROC pode mostrar como é a curva para verdadeiro positivo em comparação com falso positivo em vários limiares. Isso significa que, ao calcular a curva AUC-ROC, você plota múltiplas matrizes de confusão em diferentes limiares e as compara entre si para descobrir o limiar necessário para o seu caso de uso comercial.

### 2.8.12 Métricas de regressão

Erro quadrático médio

R²

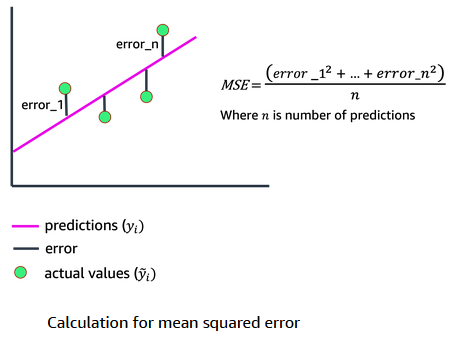
### 2.8.13 Métricas de problemas de regressão

No caso de um problema de regressão, existem outras métricas comuns que você pode usar para avaliar seu modelo, incluindo o erro quadrático médio e o R². O erro quadrático médio é muito comumente usado.

### 2.8.14 Erro quadrático médio

O objetivo geral do erro quadrático médio (EQM) é o mesmo das métricas de classificação. Você determina a previsão a partir do modelo e compara a diferença entre a previsão e o resultado real.

Mais especificamente, você pega a diferença entre a previsão e o valor real, eleva essa diferença ao quadrado e, em seguida, soma todas as diferenças ao quadrado para todas as observações.



Quanto menor o EQM, melhor a precisão preditiva do modelo.

### 2.8.15 R²

O R² é outra métrica comumente usada em problemas de regressão linear. O R² explica a fração da variância considerada pelo modelo. É como uma porcentagem, representando um número de 0 a 1. Quando o R² está próximo de 1, geralmente indica que grande parte da variância nos dados pode ser explicada pelo próprio modelo.

O MSE se concentra no erro quadrático médio das previsões do modelo para fornecer uma medida do desempenho do modelo. O R² fornece uma medida da qualidade do ajuste do modelo aos dados. Ambos são importantes, mas oferecem perspectivas diferentes.

2.8.16 Erros que eu mais cometo

**1. Viés e Variância – O Dilema do Modelo**

🔷 O que é *viés*?

O viés mede o quanto o seu modelo está errando de forma sistemática.

Um modelo com alto viés está simplificando demais o problema.

Isso leva a underfitting: ele não consegue aprender os padrões dos dados de treino nem de teste.

Exemplo:

Imagine tentar prever o preço de imóveis só com a quantidade de quartos. Isso tem alto viés: você está ignorando localização, metragem, idade, etc.

🔶 O que é *variância*?

A variância mede o quanto o seu modelo muda se os dados de treino mudarem.

Alta variância indica que o modelo aprendeu os dados de treino muito bem (inclusive os ruídos!), mas não generaliza bem.

Isso leva a overfitting.

Exemplo:

Você treina um modelo com muitos parâmetros (ex: uma rede profunda) e ele acerta tudo no treino, mas vai mal no teste.

⚖️ Resumo rápido

| Característica | Alto Viés | Alta Variância |
| --- | --- | --- |
| Tipo de erro | Simplificação demais | Sensível a ruído |
| Resultado | Underfitting | Overfitting |
| Solução | Aumentar complexidade | Regularizar, simplificar |

💡 Como isso aparece na prova?

"O modelo tem baixo erro no treino, mas alto erro no teste" → Overfitting → alta variância

"O modelo tem alto erro tanto no treino quanto no teste" → Underfitting → alto viés

Cuidado: muitos erram isso porque confundem “variância” com “erro” — mas é sobre como o modelo reage a mudanças nos dados.

**🎯 2. Precisão e Recall – Métricas de Classificação**

Essas duas são muito confundidas, mas são essenciais para entender como um modelo de classificação se comporta.

🔵 O que é *precisão* (precision)?

Das previsões positivas que o modelo fez, quantas estavam corretas?

Precisa˜o=VPVP+FPPrecisa˜o=VP+FPVP​

VP: Verdadeiros positivos

FP: Falsos positivos

Exemplo:

Você cria um modelo para detectar fraude. Ele marcou 100 casos como fraude, mas só 70 eram mesmo. A precisão é 70%.

🔴 O que é *recall*?

Dos casos realmente positivos, quantos o modelo encontrou?

Recall=VPVP+FNRecall=VP+FNVP​

FN: Falsos negativos

Exemplo:

Havia 100 fraudes no total, mas o modelo só identificou 70. O recall é 70%.

⚖️ Diferença prática:

| Cenário | Alta precisão | Alto recall |
| --- | --- | --- |
| Pega só os casos “certeza” | ✅ | ❌ |
| Pega tudo, até com risco | ❌ | ✅ |
| Aplicação ideal | Diagnóstico médico (⚠️ recall), email spam (🎯 precisão) |  |

💡 Dicas para a prova:

A pergunta menciona falsos positivos? → pense em precisão

A pergunta menciona falsos negativos? → pense em recall

Alta precisão, baixo recall → modelo só classifica como positivo quando tem certeza (mas perde muitos positivos!)

Alto recall, baixa precisão → pega todos os positivos, mas com mais falsos positivos

📌 Seus erros mais frequentes

| Tema | Erro típico | Dica prática |
| --- | --- | --- |
| Viés/variância | Confunde overfitting com underfitting | Lembre: over = complexidade, under = simples |
| Precisão/recall | Associa recall com falsos positivos | Falsos positivos → precisão |
| Métricas | Confunde precisão com acurácia | Acurácia é geral, precisão é só para positivos |
| Ajustes de modelo | Marca regularização como solução para underfitting | Regularização trata overfitting |

## 2.9 Implantação do Modelo

A implantação do modelo é a integração do modelo e seus recursos em um ambiente de produção para que ele possa ser usado para criar previsões.

### 2.9.1 API auto hospedada

Em uma abordagem de API auto hospedada, você implanta e hospeda seus modelos de ML em sua própria infraestrutura, seja localmente ou na nuvem (usando máquinas virtuais ou contêineres). Essa abordagem envolve a configuração e o gerenciamento da infraestrutura necessária, como servidores web, balanceadores de carga e bancos de dados, para servir seus modelos de ML como APIs.

### 2.9.2 API Gerenciada

Os serviços de API gerenciada são serviços baseados em nuvem que fornecem um ambiente totalmente gerenciado para implantar e hospedar seus modelos de ML como APIs. O SageMaker é um exemplo. Esses serviços abstraem o gerenciamento da infraestrutura subjacente para que você possa se concentrar na construção e implantação de seus modelos.

As vantagens das APIs auto hospedadas incluem maior controle sobre a infraestrutura, potencial economia de custos (dependendo do uso) e a capacidade de personalizar o ambiente de implantação. No entanto, essa abordagem exige mais sobrecarga operacional e responsabilidade pelo gerenciamento e manutenção da infraestrutura.

A escolha entre um serviço de API gerenciado ou uma API auto hospedada para implantação de ML depende de fatores como os requisitos específicos do seu caso de uso, o nível de controle e personalização necessários, os recursos e a expertise disponíveis e as considerações de custo.

### 2.9.3 Implantação de Modelos com o SageMaker

Implantação com um clique ou uma única chamada de API; Escalonamento automático; Serviços de hospedagem de modelos; Endpoints HTTPS que podem hospedar vários modelos

Tempo real

* A inferência em tempo real é ideal para cargas de trabalho de inferência com requisitos de tempo real, interatividade e baixa latência.

Transformação em lote

* Use a transformação em lote quando precisar obter inferências de grandes conjuntos de dados e não precisar de um endpoint persistente. Você também pode usá-la quando precisar pré-processar conjuntos de dados para remover ruído ou viés que interferem no treinamento ou na inferência do seu conjunto de dados.

Assíncrono

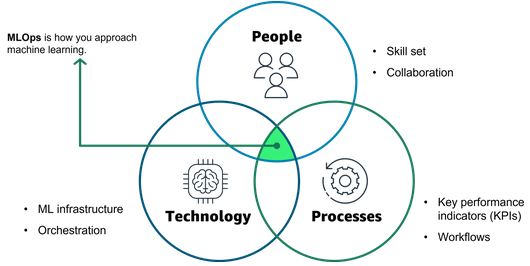
* A inferência assíncrona do SageMaker é um recurso do SageMaker que enfileira solicitações recebidas e as processa de forma assíncrona. Essa opção é ideal para solicitações com grandes tamanhos de payload (até 1 GB), longos tempos de processamento (até uma hora) e requisitos de latência quase em tempo real.

Sem servidor

* A inferência sem servidor sob demanda é ideal para cargas de trabalho que têm períodos ociosos entre picos de tráfego e podem tolerar inicializações a frio. É uma opção de inferência desenvolvida especificamente para implantar e escalar modelos de ML sem configurar ou gerenciar nenhuma infraestrutura subjacente.

## 2.10 Conceitos Fundamentais de MLOps

MLOps combina pessoas, tecnologia e processos para fornecer soluções colaborativas de ML. MLOps refere-se à prática de operacionalizar e otimizar o ciclo de vida de aprendizado de máquina de ponta a ponta, desde o desenvolvimento e implantação do modelo até o monitoramento e a manutenção. Ajuda a garantir que os modelos não sejam apenas desenvolvidos, mas também implantados, monitorados e retreinados de forma sistemática e repetida. É uma extensão dos princípios e práticas de DevOps para o domínio específico de sistemas de aprendizado de máquina.



Assim como o DevOps, o MLOps se baseia em uma abordagem colaborativa e otimizada para o ciclo de vida de desenvolvimento do aprendizado de máquina. É a interseção de pessoas, processos e tecnologia que otimiza as atividades de ponta a ponta necessárias para desenvolver, construir e operar cargas de trabalho de aprendizado de máquina.

### 2.10.1 Usando MLOps

O MLOps considera os aspectos únicos de projetos de inteligência artificial e aprendizado de máquina (IA/ML) em gerenciamento de projetos, integração e entrega contínuas (CI/CD) e garantia de qualidade. Com ele, você pode melhorar o tempo de entrega, reduzir defeitos e tornar a ciência de dados mais produtiva.

### 2.10.2 Objetivos do MLOps

Um objetivo do MLOps é colocar as cargas de trabalho de ML em produção e mantê-las em operação. Para atingir esse objetivo, o MLOps adota muitos princípios e práticas de DevOps para o desenvolvimento, treinamento, implantação, monitoramento e retreinamento de modelos de aprendizado de máquina. O objetivo é usar o MLOps para:

Aumentar o ritmo do ciclo de vida de desenvolvimento do modelo por meio da automação.

Aprimorar as métricas de qualidade por meio de testes e monitoramento.

Promover uma cultura de colaboração entre cientistas de dados, engenheiros de dados, engenheiros de software e operações de TI.

Forneça transparência, explicabilidade, audibilidade e segurança dos modelos usando a governança de modelos.

### 2.10.3 Benefícios do MLOps

Produtividade

* Ao fornecer ambientes de autoatendimento com acesso a conjuntos de dados selecionados, engenheiros e cientistas de dados podem se mover mais rapidamente e perder menos tempo com dados ausentes ou inválidos.

Confiabilidade

* Ao incorporar práticas de CI/CD, os desenvolvedores podem implantar rapidamente com maior qualidade e consistência.

Repetibilidade

* Ao automatizar todas as etapas do ciclo de vida de desenvolvimento de aprendizado de máquina, você pode garantir um processo repetível, incluindo como o modelo é treinado, avaliado, versionado e implantado.

Auditoria

* Ao versionar todas as entradas e saídas, desde experimentos de ciência de dados até dados de origem e modelos treinados, você pode demonstrar exatamente como o modelo foi construído e onde foi implantado.

Qualidade de Dados e Modelos

* Com o MLOps, você pode aplicar políticas que protegem contra vieses no modelo e rastreiam alterações nas propriedades estatísticas dos dados e na qualidade do modelo ao longo do tempo.

### 2.10.4 Princípios-chave do MLOps

Controle de versão

* Para reprodutibilidade, os fluxos de trabalho de aprendizado de máquina devem rastrear alterações em ativos como dados, código e modelos. Eles podem ser revertidos para versões anteriores quando necessário. De modo geral, o controle de versão e a revisão de código proporcionam aprendizado de máquina reprodutível e confiável.

Automação

* Para repetibilidade, consistência e escalabilidade, você pode automatizar as várias etapas do pipeline de aprendizado de máquina. Isso inclui as etapas de ingestão de dados, pré-processamento, treinamento do modelo e validação e implantação. Testes automatizados ajudam a descobrir problemas antecipadamente para correções rápidas de erros e aprendizados.

CI/CD

* + A integração contínua estende a validação e o teste de código para dados e modelos no pipeline.
  + A entrega contínua implanta automaticamente o modelo recém-treinado ou o serviço de previsão de modelos.
  + O treinamento contínuo retreina automaticamente os modelos de ML para reimplantação.
  + O monitoramento contínuo utiliza o monitoramento de dados e o monitoramento de modelos de métricas relacionadas aos negócios.

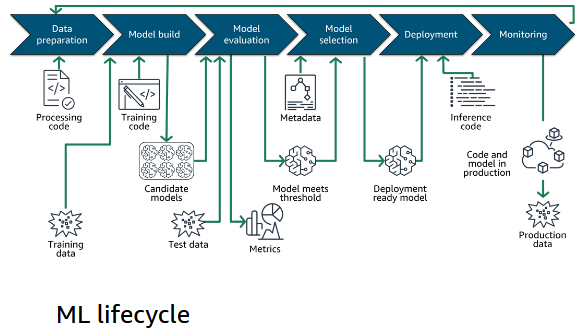
Governança de Modelos

* Uma boa governança de sistemas de aprendizado de máquina exige uma colaboração próxima entre cientistas de dados, engenheiros e stakeholders do negócio. Documentação clara, canais de comunicação eficazes e mecanismos de feedback ajudam a alinhar todos e a aprimorar os modelos ao longo do tempo. Também é crucial proteger dados confidenciais, proteger o acesso e cumprir as regras de conformidade. Um processo estruturado para revisar, validar e aprovar modelos antes da implantação verifica a imparcialidade, a parcialidade e a ética. A governança gerencia todos os aspectos dos sistemas para garantir a eficiência.

### 2.10.5 Ciclo de vida de ML e MLOps

Gerenciar código, dados e modelos ao longo do ciclo de vida de ML requer os seguintes pontos de contato:

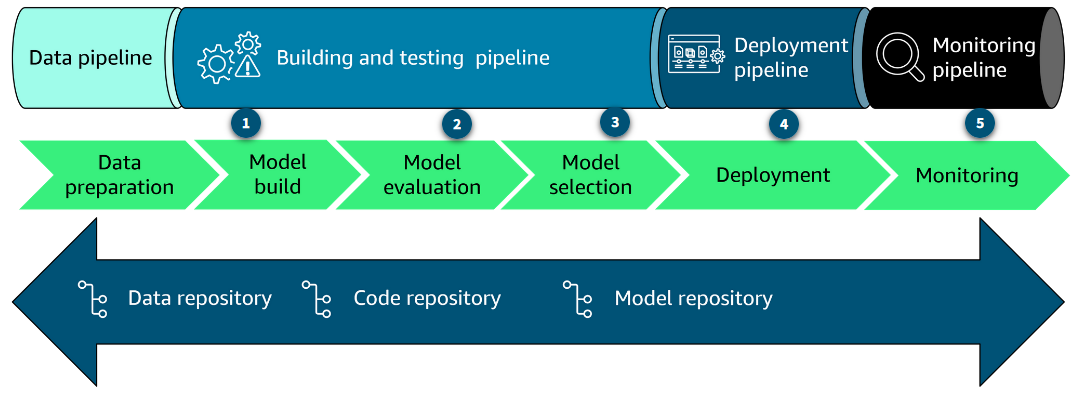
* Processamento de código na preparação de dados
* Dados de treinamento e código de treinamento na construção de modelos
* Modelos candidatos, dados de teste e validação na avaliação de modelos
* Metadados durante a seleção do modelo
* Modelos prontos para implantação e código de inferência durante a implantação
* Código de produção, modelos e dados para monitoramento



Com o MLOps, você operacionaliza os processos de desenvolvimento, implantação, monitoramento e governança de modelos de ML.

### 2.10.6 Implementando MLOps

Um ciclo de vida de ML em produção normalmente contém pipelines separados de treinamento e implantação.



1 Construção do modelo

* O pipeline de construção do modelo cria novos modelos na inicialização, por exemplo, quando novos dados ficam disponíveis.

2 Avaliação do modelo

* Quando o pipeline de construção do modelo for concluído, você poderá implementar medidas de controle de qualidade na etapa de registro do modelo. A etapa de controle de qualidade pode ser manual (com intervenção humana) ou automatizada. Se um modelo atender às métricas de desempenho básicas, ele poderá ser registrado em um registro de modelos.

3 Aprovação do modelo

* Você pode usar o registro para aprovar ou rejeitar versões do modelo. A ação de aprovação do modelo pode atuar como um início para iniciar o pipeline de implantação.

4 Implantação do modelo

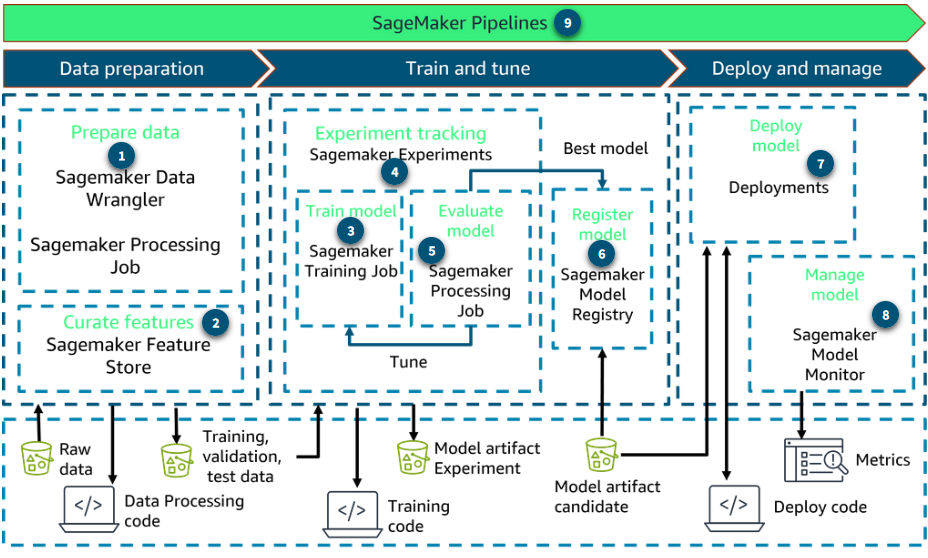
* O pipeline de implantação é mais semelhante aos sistemas tradicionais de CI/CD. Este pipeline inclui etapas como as seguintes:
  + Código-fonte
  + Construção
  + Implantação no ambiente de preparação
  + Teste
  + Promoção para o ambiente de produção

5 Modelo em produção

* Assim que o modelo estiver em produção, você deverá obter feedback do sistema em produção. Para soluções de ML, monitore a infraestrutura de hospedagem, a qualidade dos dados e o desempenho do modelo.

### 2.10.7 Serviços da AWS para MLOps

No diagrama a seguir, você pode ver quais serviços da AWS podem ser usados para implementar um pipeline de MLOps.



1 Preparar dados

* O SageMaker Data Wrangler é uma ferramenta LCNC que fornece uma solução completa para importar, preparar, transformar, caracterizar e analisar dados usando uma interface web.
* Usando a API de Processamento do SageMaker, cientistas de dados podem executar scripts e notebooks para processar, transformar e analisar conjuntos de dados de diversas estruturas de ML, como Scikit-learn, MXNet ou PyTorch, enquanto se beneficiam de ambientes de aprendizado de máquina totalmente gerenciados.

2 Armazenar recursos

* O SageMaker Feature Store ajuda cientistas de dados, engenheiros de aprendizado de máquina e clínicos gerais a criar, compartilhar e gerenciar recursos para desenvolvimento de ML.

3 Treinar

* O SageMaker fornece um recurso de tarefa de treinamento para treinar modelos usando algoritmos integrados ou algoritmos personalizados.

4 Experimentos

* Use os Experimentos do SageMaker para experimentar com múltiplas combinações de dados, algoritmos e parâmetros, enquanto observa o impacto de alterações incrementais na precisão do modelo.

5 Tarefa de processamento

* O Processamento do SageMaker refere-se aos recursos para executar tarefas de pré-processamento e pós-processamento de dados, engenharia de recursos e avaliação de modelos na infraestrutura totalmente gerenciada do SageMaker.

6 Registro

* Com o Registro de Modelos do SageMaker, você pode catalogar modelos, gerenciar versões de modelos, gerenciar o status de aprovação de um modelo ou implantar modelos em produção.

7 Implantações

* Com o SageMaker, você pode implantar seus modelos de ML para fazer previsões, também conhecido como inferência. O SageMaker oferece uma ampla seleção de opções de infraestrutura e implantação de modelos de ML para ajudar a atender a todas as suas necessidades de inferência de ML.

8 Monitorar modelo

* Com o SageMaker Model Monitor, você pode monitorar a qualidade dos modelos de ML do SageMaker em produção.

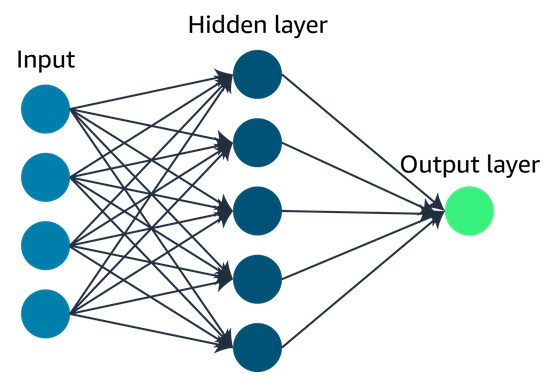
9 Pipelines

* Você pode usar os Pipelines de Construção de Modelos do Amazon SageMaker para criar fluxos de trabalho completos que gerenciam e implantam tarefas do SageMaker.

# 3. Fundamentos do Deep Learning

## 3.1 Redes Neurais

As redes neurais possuem diversas unidades minúsculas chamadas nós, que estão conectadas entre si. Esses nós são organizados em camadas. As camadas incluem uma camada de entrada, uma ou mais camadas ocultas e uma camada de saída.



Quando mostramos a uma rede neural muitos exemplos, como dados sobre clientes que compraram determinados produtos ou usaram determinados serviços, ela descobre como identificar padrões ajustando as conexões entre seus nós. É como se os nós estivessem se comunicando e, lentamente, descobrindo os padrões que separam os diferentes tipos de clientes.

Quando uma rede neural aprende a reconhecer esses padrões a partir dos exemplos, ela pode então analisar dados de clientes completamente novos que nunca viu antes e ainda fazer previsões sobre o que eles podem comprar ou como podem se comportar.

Alguns ramos da IA ​​onde o aprendizado profundo é usado para aprimorar os resultados:

**A visão computacional** é um campo da inteligência artificial que possibilita aos computadores interpretar e compreender imagens e vídeos digitais. O aprendizado profundo revolucionou a visão computacional ao fornecer técnicas poderosas para tarefas como classificação de imagens, detecção de objetos e segmentação de imagens.

**O processamento de linguagem natural (NLP)** é um ramo da inteligência artificial que lida com a interação entre computadores e linguagens humanas. O aprendizado profundo fez avanços significativos no NLP, possibilitando tarefas como classificação de texto, análise de sentimentos, tradução automática e geração de linguagem.

# 4. Fundamentos da IA Generativa

A IA generativa é um subconjunto do aprendizado profundo. Ela pode adaptar modelos construídos com aprendizado profundo sem a necessidade de retreinamento ou ajuste fino. A IA generativa é capaz de gerar novos dados com base nos padrões e estruturas aprendidos com os dados de treinamento. A IA generativa pode criar novos conteúdos, incluindo conversas, histórias, imagens, vídeos, músicas e código.

## 4.1 Capacidades da IA Generativa

### 4.1.1 Adaptabilidade

Modelos de IA Generativa podem se adaptar a diversas tarefas e domínios, aprendendo com dados e gerando conteúdo adaptado a contextos ou requisitos específicos. Como a IA Generativa é flexível, ela pode ser usada para uma ampla gama de aplicações em diferentes setores.

### 4.1.2 Responsividade

Modelos de IA Generativa podem gerar conteúdo em tempo real, o que resulta em tempos de resposta rápidos e interações dinâmicas. Isso é particularmente útil para chatbots, assistentes virtuais e outros aplicativos interativos que exigem respostas imediatas.

### 4.1.3 Simplicidade

A IA Generativa pode simplificar tarefas complexas, automatizando processos de criação de conteúdo. Por exemplo, modelos de linguagem de IA podem gerar texto semelhante ao humano, o que reduz o tempo e o esforço necessários para a geração de conteúdo.

4.1.4 Criatividade e Exploração

Modelos de IA Generativa podem gerar novas ideias, designs ou soluções, combinando e recombinando elementos de maneiras únicas. Isso pode fomentar a criatividade e a exploração de novas possibilidades.

### 4.1.5 Eficiência de Dados

Alguns modelos de IA Generativa podem aprender com quantidades relativamente pequenas de dados e gerar novas amostras consistentes com os dados de treinamento. Isso pode ser útil quando os dados são escassos ou difíceis de obter.

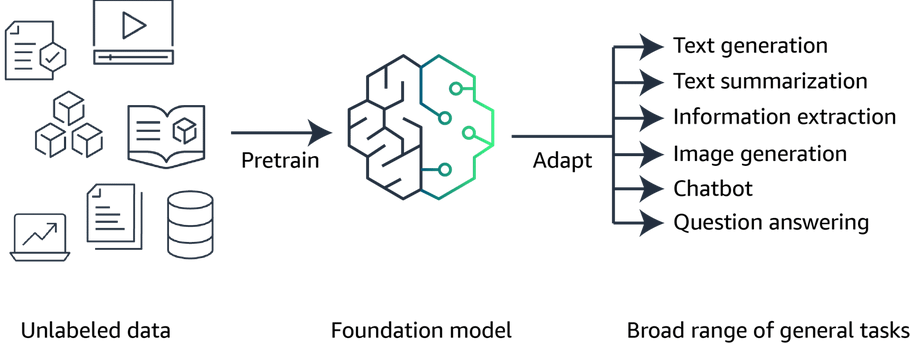
4.1.6 Personalização

A IA generativa pode criar conteúdo personalizado, adaptado às preferências ou características individuais, o que aprimora a experiência e o engajamento do usuário.

### 4.1.7 Escalabilidade

Quando treinados, os modelos de IA generativa podem gerar grandes quantidades de conteúdo rapidamente. Isso os torna adequados para tarefas que exigem a produção de conteúdo em escala.

## 4.2 Foundation Models



A IA generativa é alimentada por modelos pré-treinados em dados em escala de internet, e esses modelos são chamados de modelos de base (FM). Com os FM, em vez de coletar dados rotulados para cada modelo e treinar vários modelos como no ML tradicional, você pode adaptar um único FM para executar múltiplas tarefas. Essas tarefas incluem geração de texto, sumarização de texto, extração de informações, geração de imagens, interações com chatbots e resposta a perguntas. Os FM também podem servir como ponto de partida para o desenvolvimento de modelos mais especializados.

### 4.2.1 FM ciclo de Vida – Seleção de Dados

Dados não rotulados podem ser usados ​​em larga escala para pré-treinamento, pois são muito mais fáceis de obter em comparação com dados rotulados. Dados não rotulados incluem dados brutos, como imagens, arquivos de texto ou vídeos, sem rótulos informativos significativos para fornecer contexto. FMs exigem treinamento em conjuntos de dados massivos de diversas fontes.

### 4.2.2 FM ciclo de Vida – Pré-Treino

FMs são normalmente pré-treinados por meio de aprendizado autos-supervisionado. Com o aprendizado autos-supervisionado, exemplos rotulados não são necessários. O aprendizado autos-supervisionado utiliza a estrutura dos dados para gerar rótulos automaticamente.

Durante o estágio inicial de pré-treinamento, o algoritmo do FM pode aprender o significado, o contexto e a relação das palavras nos conjuntos de dados. Por exemplo, o modelo pode aprender se "beber" significa "bebida", o substantivo, ou "engolir o líquido", o verbo.

Após o pré-treinamento inicial, o modelo pode ser pré-treinado novamente com dados adicionais. Isso é conhecido como pré-treinamento contínuo. O objetivo é expandir a base de conhecimento do modelo e melhorar sua capacidade de compreensão e generalização em diferentes domínios ou tarefas.

### 4.2.3 FM ciclo de Vida – Otimização

Modelos de linguagem pré-treinados podem ser otimizados por meio de técnicas como engenharia de prompts, geração aumentada de recuperação (RAG) e ajuste fino em dados específicos da tarefa. Esses métodos variam em complexidade e custo.

### 4.2.4 FM ciclo de Vida – Avaliação

Independentemente de você ajustar um modelo ou usar um modelo pré-treinado pronto para uso, o próximo passo lógico é avaliá-lo. O desempenho de um FM pode ser medido usando métricas e benchmarks apropriados. Avaliar o desempenho do modelo e sua capacidade de atender às necessidades do negócio é importante.

### 4.2.5 FM ciclo de Vida – Implantação

Quando o FM atende aos critérios de desempenho desejados, ele pode ser implantado no ambiente de produção de destino. A implantação pode envolver a integração do modelo em aplicativos, APIs ou outros sistemas de software.

### 4.2.6 FM ciclo de Vida – Feedback e melhoria contínua

Após a implantação, o desempenho do modelo é monitorado continuamente, e o feedback é coletado de usuários, especialistas da área ou outras partes interessadas. Esse feedback, juntamente com os dados de monitoramento do modelo, é usado para identificar áreas de melhoria, detectar possíveis vieses ou desvios e informar futuras iterações do modelo. O ciclo de feedback permite o aprimoramento contínuo do modelo base por meio de ajustes finos, pré-treinamento contínuo ou retreinamento, conforme necessário.

## 4.3 Amazon Bedrock

O Amazon Bedrock oferece acesso a uma seleção de FMs de alto desempenho de empresas líderes em IA, como AI21 Labs, Anthropic, Cohere, Meta, Mistral AI, Stability AI e Amazon.

Com esses FMs como base, você pode otimizar ainda mais seus resultados com engenharia rápida, ajustes finos ou RAG.

## 4.4 Tipos de FM

### 4.4.1 Grandes modelos de linguagem (LLM)

Grandes modelos de linguagem (LLMs) podem ser baseados em uma variedade de arquiteturas, mas a arquitetura mais comum nos modelos de última geração atuais é a arquitetura de transformador. LLMs baseados em transformadores são modelos poderosos que podem compreender e gerar textos com aparência humana. Eles são treinados com grandes quantidades de dados de texto da internet, livros e outras fontes, e aprendem padrões e relações entre palavras e frases.

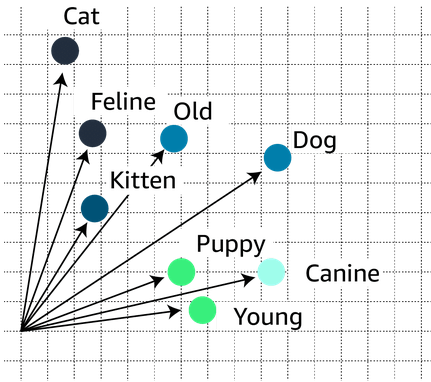
Os LLMs usam esses **tokens, embeddings e vetores** para compreender e gerar texto. Os modelos conseguem capturar relações complexas na linguagem, permitindo que eles gerem textos coerentes e contextualmente apropriados, respondam a perguntas, resumirem informações e até mesmo se envolvam em escrita criativa.

**Tokens são as unidades básicas de texto que o modelo processa**. Os tokens podem ser palavras, frases ou caracteres individuais, como um ponto final. Os tokens também fornecem padronização dos dados de entrada, o que facilita o processamento pelo modelo.

Por exemplo, a frase "Um cachorrinho está para um cachorro assim como um gatinho está para um gato." pode ser dividida nos seguintes tokens: "Um" "filhote de cachorro" "é" "para" "cachorro" "como" "um" "gatinho" "é" "para" "gato."

**Embeddings são representações numéricas de tokens**, onde cada token recebe um vetor (uma lista de números) que captura seu significado e relacionamentos com outros tokens. Esses vetores são aprendidos durante o processo de treinamento e permitem que o modelo compreenda o contexto e as nuances da linguagem.

Por exemplo, o vetor de embedding para o token "gato" pode ser próximo aos vetores para "felino" e "gatinho" no espaço de embedding, indicando que eles são semanticamente relacionados. Dessa forma, o modelo pode entender que "gato" é semelhante a "felino" e "gatinho" sem ser explicitamente programado com esses relacionamentos.

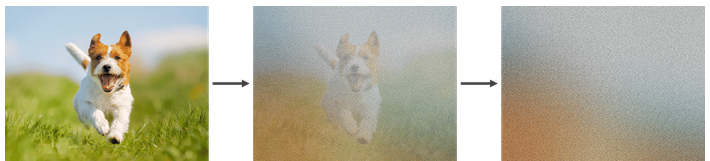


### 4.4.2 Modelos de Difusão

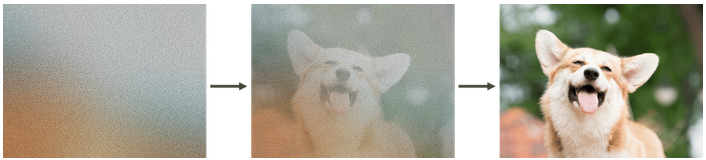
Difusão é um sistema de arquitetura de aprendizado profundo que começa com ruído puro ou dados aleatórios. Os modelos adicionam gradualmente informações cada vez mais significativas a esse ruído até obterem uma saída clara e coerente, como uma imagem ou um texto. Os modelos de difusão aprendem por meio de um processo de duas etapas: difusão direta e difusão reversa.

Embora algumas das aplicações mais conhecidas e impressionantes dos modelos de difusão tenham sido os modelos de texto para imagem, os modelos de difusão podem ser aplicados a uma variedade de tarefas além da simples geração de imagens.

Usando a **difusão direta**, o sistema introduz gradualmente uma pequena quantidade de ruído em uma imagem de entrada até que somente o ruído permaneça.



Na **difusão reversa**, a imagem ruidosa é gradualmente introduzida para redução de ruído até que uma nova imagem seja gerada.



### 4.4.3 Modelos multimodais

Em vez de depender apenas de um único tipo de entrada ou saída, como texto ou imagens, os modelos multimodais podem processar e gerar múltiplos modos de dados simultaneamente. Por exemplo, um modelo multimodal pode receber uma imagem e algum texto como entrada e, em seguida, gerar uma nova imagem e uma legenda descrevendo-a como saída.

Esses tipos de modelos aprendem como diferentes modalidades, como imagens e texto, estão conectadas e podem influenciar umas às outras. Modelos multimodais podem ser usados ​​para automatizar a legendagem de vídeos, criar gráficos a partir de instruções de texto, responder a perguntas de forma mais inteligente combinando texto e informações visuais e até mesmo traduzir conteúdo, mantendo os recursos visuais relevantes.

## 4.5 Outros modelos generativos

### 4.5.1 Redes adversárias generativas (GAN)

GANs são um tipo de modelo generativo que envolve duas redes neurais competindo entre si em uma estrutura de jogo de soma zero. As duas redes são geradoras e discriminadoras.

**Geradora**: Esta rede gera novos dados sintéticos (por exemplo, imagens, texto ou áudio) tomando ruído aleatório como entrada e transformando-o em dados que se assemelham à distribuição dos dados de treinamento.

**Discriminadora:** Esta rede recebe dados reais do conjunto de treinamento e dados sintéticos gerados pelo gerador como entrada. Seu objetivo é distinguir entre os dados reais e os gerados.

Durante o treinamento, o gerador tenta gerar dados que possam enganar o discriminador, fazendo-o pensar que são reais, enquanto o discriminador tenta classificar corretamente os dados reais e os gerados. Esse processo adversarial continua até que o gerador produza dados indistinguíveis dos dados reais.

### 4.5.2 Auto codificadores variacionais (VAE)

VAEs são um tipo de modelo generativo que combina ideias de auto codificadores (um tipo de rede neural) e inferência variacional (uma técnica da estatística bayesiana). Em uma VAE, o modelo consiste em duas partes:

**Codificador:** Esta rede neural recebe os dados de entrada (por exemplo, uma imagem) e os mapeia para um espaço latente de menor dimensão, que captura as características essenciais dos dados.

**Decodificador:** Esta rede neural recebe a representação latente do codificador e gera uma reconstrução dos dados de entrada originais.

O aspecto fundamental das VAEs é que o espaço latente é incentivado a seguir uma distribuição de probabilidade específica (geralmente uma distribuição gaussiana), o que permite a geração de novos dados por meio de amostragem desse espaço latente e passagem das amostras pelo decodificador.

## 4.6 Otimizando saídas do modelo

Um FM pode ser otimizado de diversas maneiras. Essas técnicas variam em complexidade e custo, sendo a Prompt Engineering a opção mais rápida e de menor custo.

### 4.6.1 Prompt Engineering

Prompts atuam como instruções para modelos de base. A engenharia de prompts concentra-se no desenvolvimento, design e otimização de prompts para aprimorar a saída dos FMs de acordo com suas necessidades. Ela oferece uma maneira de orientar o comportamento do modelo para os resultados que você deseja alcançar.

O formato de um prompt depende da tarefa que você está atribuindo a um modelo. Ao explorar exemplos de engenharia de prompts, você revisará prompts que contêm alguns ou todos os seguintes elementos:

**Instruções**: Esta é uma tarefa para o FM realizar. Ela fornece uma descrição da tarefa ou instrução sobre como o modelo deve executar.

**Contexto**: Esta é uma informação externa para orientar o modelo.

**Dados de entrada**: Esta é a entrada para a qual você deseja uma resposta.

**Indicador de saída:** Este é o tipo ou formato de saída.

**Exemplo de prompt**

Você é um jornalista experiente que se destaca em condensar artigos longos em resumos concisos. Resuma o texto a seguir em 2 a 3 frases.

Texto: [Texto de artigo longo vai aqui]

### 4.6.2 Afinação

O ajuste fino do modelo base do FM pode melhorar o desempenho. O **ajuste fino** é um processo de **aprendizado supervisionado** que envolve a utilização de um **modelo pré-treinado** e a adição de conjuntos de dados específicos e menores. A adição desses conjuntos de dados mais específicos modifica os pesos dos dados para melhor alinhamento com a tarefa.

Há duas maneiras de ajustar um modelo:

**O ajuste fino de instruções** utiliza exemplos de como o modelo deve responder a uma instrução específica. O ajuste por prompt é um tipo de ajuste fino de instruções.

**O aprendizado por reforço a partir de feedback humano** (Reinforcement learning from human feedback - RLHF) fornece dados de feedback humano, resultando em um modelo mais alinhado às preferências humanas.

Considere este caso de uso para ajuste fino. Se você estiver trabalhando em uma tarefa que exige conhecimento do setor, pode utilizar um modelo pré-treinado e ajustá-lo com dados do setor. Se a tarefa envolver pesquisa médica, por exemplo, o modelo pré-treinado pode ser ajustado com artigos de periódicos médicos para obter resultados mais contextualizados.

### 4.6.3 Geração aumentada de recuperação

A geração aumentada de recuperação (RAG) é uma técnica que fornece dados relevantes para o domínio como contexto para produzir respostas com base nesses dados. Essa técnica é semelhante ao ajuste fino. No entanto, em vez de precisar ajustar um FM com um pequeno conjunto de exemplos rotulados, a RAG recupera um pequeno conjunto de documentos relevantes e os utiliza para fornecer contexto para responder à solicitação do usuário. A RAG não altera os pesos do modelo base, enquanto o ajuste fino altera os pesos do modelo.

## 4.7 Desafios da IA Generativa

### 4.7.1 Violações regulatórias

**Risco**

Modelos de IA Generativa treinados com dados sensíveis podem gerar inadvertidamente uma saída que viola regulamentações, como a exposição de informações de identificação pessoal (PII).

**Mitigação**

Para minimizar o risco de violações de privacidade, implemente técnicas rigorosas de anonimização de dados e preservação da privacidade durante o treinamento do modelo. Para garantir a conformidade com as regulamentações de privacidade, realize auditorias e avaliações completas dos dados usados para treinar o modelo.

### 4.7.2 Riscos sociais

**Risco**

A possibilidade de conteúdo indesejado que possa refletir negativamente em sua organização é um risco social.

**Mitigação**

Teste e avalie todos os modelos antes de implantá-los em produção.

### 4.7.3 Preocupações com segurança e privacidade de dados

**Risco**

As informações compartilhadas com seu modelo podem incluir informações pessoais e podem violar as leis de privacidade.

**Mitigação**

Para proteger dados sensíveis, implemente medidas de segurança cibernética, como criptografia e firewalls.

### 4.7.4 Toxicidade

**Risco**

Modelos de IA generativa podem gerar conteúdo inflamatório, ofensivo ou inapropriado.

**Mitigação**

Cure os dados de treinamento identificando essas frases com antecedência e removendo-as dos dados de treinamento. Isso evita que sejam geradas como saída.

Use modelos de guarda-corpo. Esses modelos detectarão e filtrarão conteúdo indesejado.

### 4.7.5 Alucinações

**Risco**

O modelo gera respostas imprecisas que não são consistentes com os dados de treinamento. Essas respostas são chamadas de alucinações.

**Mitigação**

Ensine aos usuários que tudo deve ser verificado. Não se pode confiar em modelos de base (FM) para verificar se suas próprias histórias são baseadas na realidade e em fatos. As alucinações podem ser ainda mais mitigadas verificando se o conteúdo é verificado com fontes independentes. Além disso, o conteúdo gerado pode ser marcado como não verificado para alertar o usuário de que a verificação será necessária.

### 4.7.6 Interpretabilidade

**Risco**

Os usuários podem interpretar mal a saída do modelo, o que pode levar a conclusões ou decisões incorretas.

**Mitigação**

Utilize conhecimento específico do domínio para o desenvolvimento e desempenho do modelo, fornecendo informações essenciais para as entradas do modelo de dados.

### 4.7.7 Não determinismo

**Risco**

O modelo pode gerar saídas diferentes para a mesma entrada, o que pode causar problemas em aplicações onde a confiabilidade é fundamental.

**Mitigação**

Realize testes no modelo para identificar quaisquer fontes de não determinismo. Execute o modelo várias vezes e compare a saída para garantir a consistência.

## 4.8 Fatores a Considerar ao Selecionar um Modelo de IA Generativa

Primeiro, é essencial definir a tarefa ou aplicação específica que você deseja que o modelo execute, como geração de texto, criação de imagens ou geração de código. Os modelos são otimizados para diferentes tarefas, portanto, escolher a correta é crucial para alcançar os resultados desejados.

Alguns dos principais fatores a considerar ao selecionar um modelo de IA generativa apropriado incluem:

### 4.8.1 Tipos de modelo

Existem muitos tipos de modelos, cada modelo tem suas próprias capacidades e desafios.

**Laboratórios AI21**

**Modelos Jurassic-2**

Geração de texto; Sumarização; Paráfrase; Bate-papo; Extração de informações

Serviços financeiros – resumir documentos extensos

Varejo – gerar descrições de produtos

**Amazon**

**Amazon Titan**

Sumarização de texto; Classificação; Perguntas e respostas abertas; Extração de informações; Embeddings; Busca

Publicidade – criar imagens com qualidade de estúdio

Atendimento ao cliente – gerar resumos abstratos em tempo real

**Antrópico**

**Claude**

Geração de conteúdo; Tradução de texto; Resposta a perguntas; Sumarização de texto; Explicação e geração de código

Desenvolvedor – geração e depuração de código

Jurídico – analisar documentos jurídicos e responder a perguntas

**IA de Estabilidade**

**Difusão Estável**

Gerar imagens fotorrealistas a partir de entrada de texto

Melhorar a qualidade das imagens geradas

Jogos e metaverso – criar personagens, cenas e mundos

Publicidade e marketing – criar campanhas publicitárias e recursos de marketing

**Cohere**

**Comando**

Geração de texto; Extração de informações; Perguntas e respostas; Sumarização

Atendimento ao cliente – dar suporte a chatbots

Varejo – fornecer descrições de produtos

Saúde – resumir as principais ideias de um texto longo

**Meta**

**Lhama**

Resposta a perguntas; Chat; Resumo; Paráfrase; Análise de sentimentos; Geração de texto

Suporte ao cliente – chatbots

### 4.8.2 Requisitos de desempenho

Os requisitos de desempenho são outro fator a ser considerado ao selecionar um modelo de IA generativa. Esses requisitos incluem precisão, confiabilidade da saída, entre outros. Avalie o desempenho geral do modelo para avaliar sua adequação a uma tarefa específica. Você também deve testar o modelo em diferentes conjuntos de dados para garantir a confiabilidade. Por fim, monitore seu desempenho ao longo do tempo para garantir que ele permaneça consistente.

### 4.8.3 Restrições

Recursos computacionais (por exemplo, potência de GPU, potência de CPU ou memória disponíveis)

Disponibilidade de dados (por exemplo, tamanho e qualidade dos dados de treinamento)

Requisitos de implantação (por exemplo, local ou na nuvem)

Alguns modelos podem ter demandas maiores de recursos ou exigir configurações específicas de hardware, o que pode impactar seu caso de uso.

### 4.8.4 Capacidades

Outro fator a ser considerado são as capacidades do modelo. A IA generativa abrange uma ampla gama de capacidades. Ela pode executar diferentes tarefas com diferentes graus de qualidade de saída e níveis de controle ou personalização. Por exemplo, alguns modelos podem ser melhores na geração de texto, enquanto outros podem se destacar na geração de imagens ou na execução de tarefas multimodais, como a geração de texto para imagem. Portanto, é importante entender as capacidades específicas necessárias para sua aplicação antes de selecionar um modelo de IA generativa.

### 4.8.5 Conformidade

A conformidade é outro fator. Modelos de IA generativa podem apresentar preocupações morais, incluindo vieses, problemas de privacidade e potencial uso indevido. Ao avaliar um modelo específico, considere sua conformidade e implicações morais, particularmente em áreas sensíveis como saúde, finanças e aplicações jurídicas. Deve-se considerar fatores como justiça, transparência ou rastreabilidade, responsabilização, alucinação e toxicidade. Além disso, o modelo deve estar em conformidade com as diretrizes regulatórias relevantes.

### 4.8.6 Custo

Outro fator importante é o custo. Os modelos de IA generativa podem variar em termos de custo. Considere a relação entre o tamanho e a velocidade do modelo. Modelos maiores geralmente são mais precisos, mas são caros e oferecem poucas opções de implantação. Por outro lado, modelos menores são mais baratos e rápidos, além de oferecerem mais alternativas de implantação.

Ao usar IA generativa para criação de conteúdo, você pode reduzir os custos de mão de obra e aumentar a eficiência, especialmente para tarefas repetitivas que exigem esforço humano significativo.

Lembre-se de avaliar todas as despesas relacionadas à implantação, manutenção, hardware, software e outros custos associados.

### 4.8.7 Sustentabilidade

Refere-se à capacidade dos sistemas de IA de serem desenvolvidos e implantados de forma social, ambiental e economicamente sustentável a longo prazo.

Agência responsável em IA responsável refere-se à capacidade de um sistema de IA de fazer bons julgamentos e agir de maneira socialmente responsável. A seguir, são apresentados os principais aspectos da agência moral para IA.

**Alinhamento de valores**

Alinhamento de valores é a capacidade de compreender, avaliar e tomar decisões com base em princípios morais, em vez da pura maximização da utilidade. Isso requer alinhamento de valores entre os objetivos e valores do sistema de IA e os valores humanos responsáveis.

**Habilidades de raciocínio responsável**

Habilidades de raciocínio responsável é a capacidade de pensar logicamente em dilemas morais e ponderar diversas considerações responsáveis ao tomar decisões. A IA precisa de capacidades de lógica e raciocínio para aplicar princípios responsáveis a novas situações.

O sistema de IA deve ter a capacidade de se envolver em raciocínio responsável e compreender conceitos, princípios e estruturas morais. Deve ser capaz de aplicá-los em contexto a situações específicas.

**Nível apropriado de autonomia**

O sistema de IA deve ter o nível apropriado de autonomia, com limites e mecanismos claros para supervisão e intervenção humana, particularmente em domínios de alto risco ou sensíveis.

**Transparência e responsabilização**

O sistema de IA deve ser transparente sobre seu processo de tomada de decisão. Deve permitir supervisão e responsabilização externas para garantir que suas ações sejam justificadas de forma responsável.

De modo geral, uma agência responsável exige que a IA tenha inteligência sofisticada, comparável à cognição humana, para aplicar adequadamente o raciocínio ético no mundo real. Isso continua sendo um imenso desafio para a IA atual.

### 4.8.8 Considerações ambientais para a seleção de um modelo

**Consumo de energia**

Otimizar a eficiência energética em sistemas de IA, usar fontes de energia renováveis sempre que possível e considerar a pegada de carbono geral das operações de IA.

**Utilização de recursos**

Uma IA responsável deve ter como objetivo maximizar a eficiência dos recursos, promover a reutilização e a reciclabilidade do hardware e minimizar o desperdício eletrônico.

**Avaliação de impacto ambiental**

Avaliações de impacto ambiental devem ser realizadas e estratégias de mitigação devem ser implementadas, se necessário.

**Considerações econômicas para a seleção de um modelo**

As considerações econômicas na IA responsável incluem os potenciais benefícios e custos das tecnologias de IA e o impacto sobre os empregos e a economia. Por exemplo, a IA pode automatizar certas tarefas e melhorar a eficiência, mas também pode levar à perda de empregos e à desigualdade. Além disso, há preocupações com a concentração de poder e dados nas mãos de poucas empresas, o que poderia levar a monopólios e a uma maior desigualdade.

## 4.9 Métricas de Negócios para IA Generativa

Ao quantificar o desempenho, a eficácia e o retorno sobre o investimento (ROI) das aplicações de IA por meio de métricas de negócios relevantes, as organizações podem obter insights valiosos sobre o valor entregue. Elas também podem identificar áreas de melhoria e tomar decisões informadas para otimizar a alocação de recursos e a estratégia.

### 4.9.1 Satisfação do usuário

A satisfação do usuário coleta o feedback do usuário para avaliar sua satisfação com o conteúdo ou as recomendações geradas pela IA.

Caso de uso: Mensurando e melhorando a satisfação do usuário para um site de e-commerce

Uma empresa de e-commerce deseja monitorar e aprimorar a satisfação geral do usuário com seu site para aumentar a fidelidade do cliente, a repetição de compras e o boca a boca positivo.

### 4.9.2 Receita média por usuário

A receita média por usuário (ARPU) calcula a receita média gerada por usuário ou cliente atribuída à aplicação de IA generativa.

Caso de uso: Analisando e otimizando a geração de receita por usuário.

As equipes de marketing e produto de uma empresa de e-commerce desejam entender a eficácia da monetização de sua base de usuários e identificar oportunidades de melhoria.

### 4.9.3 Desempenho entre domínios

O desempenho entre domínios mede a capacidade do modelo de IA generativa de apresentar um desempenho eficaz em diferentes domínios ou setores.

Caso de uso: Monitorando e otimizando uma plataforma de e-commerce multidomínio

A AnyCompany opera uma grande plataforma de e-commerce com múltiplos domínios, atendendo a diferentes categorias de produtos e regiões geográficas. Ela utiliza a métrica de desempenho entre domínios para monitorar o desempenho geral de sua plataforma de e-commerce em todos os domínios.

### 4.9.4 Taxa de conversão

A taxa de conversão monitora a taxa de conversão para gerar conteúdo ou recomendar os resultados desejados, como compras, cadastros ou métricas de engajamento.

Caso de uso: Otimizando um site de e-commerce para taxas de conversão mais altas

Um gerente de marketing de uma loja de roupas online é responsável por analisar e melhorar o desempenho do site em termos de conversão de visitantes em clientes pagantes. Para isso, eles monitoram de perto a métrica da taxa de conversão, que mede a porcentagem de visitantes do site que concluem uma ação desejada, como uma compra.

### 4.9.5 Eficiência

A métrica de eficiência avalia a eficiência do modelo de IA generativa em termos de utilização de recursos, tempo de computação e escalabilidade.

Caso de uso: Melhoria da eficiência da linha de produção

Exemplo: A Corp Manufacturing Company opera uma linha de produção para a montagem de dispositivos eletrônicos. A empresa visa otimizar a eficiência de sua linha de produção para reduzir custos e aumentar a produtividade.

## 4.10 Preparação Responsável para Conjuntos de Dados

### 4.10.1 Balanceamento de Conjuntos de Dados

Conjuntos de dados balanceados são importantes para a criação de modelos de IA responsáveis que não discriminem injustamente ou apresentem vieses indesejados.

Conjuntos de dados balanceados devem representar todos os grupos de pessoas ou tópicos de dados. Isso significa que o conjunto de dados deve conter um número adequado de exemplos ou instâncias de cada grupo para garantir que o modelo não seja enviesado em favor de nenhum grupo ou fator específico. O conceito de conjuntos de dados balanceados é particularmente importante em aplicações como contratação, empréstimo ou justiça criminal, onde justiça e equidade são essenciais.

Para obter conjuntos de dados balanceados, os dados coletados precisam ser inclusivos e diversos, e os dados também precisam ser curados para otimizá-los para treinamento.

A inclusão e a diversidade na coleta de dados garantem que os processos de coleta de dados sejam justos e imparciais. A coleta de dados deve refletir com precisão as diversas perspectivas e experiências necessárias para o caso de uso do sistema de IA. Isso inclui uma gama diversificada de fontes, pontos de vista e dados demográficos. Ao fazer isso, o sistema de IA pode trabalhar para garantir que as decisões sejam imparciais em seu desempenho.

A inclusão e a diversidade na coleta de dados são preocupações primárias para dados com foco em pessoas. Isso ocorre porque alienar grupos de pessoas nos dados de treinamento pode levar a danos sociais e repercussões legais. No entanto, a inclusão e a diversidade na coleta de dados devem ser o foco principal, independentemente do tópico. Por exemplo, a coleta de dados sobre pessoas, pesquisa científica, geografia, clima, produtos e outros tópicos deve ser realizada com foco na diversidade de cada tópico.

Ao promover a inclusão e a diversidade na IA, as organizações podem promover a justiça, a transparência e a responsabilização em seus sistemas de IA e contribuir para o desenvolvimento responsável da tecnologia de IA.

### 4.10.2 Curadoria de dados

A curadoria de conjuntos de dados é o processo de rotular, organizar e pré-processar os dados para que possam funcionar com precisão no modelo. A curadoria pode ajudar a garantir que os dados sejam representativos do problema em questão e livres de vieses ou outros problemas que possam afetar a precisão do modelo de IA. A curadoria ajuda a garantir que os modelos de IA sejam treinados e avaliados com base em dados confiáveis e de alta qualidade, relevantes para a tarefa que se destinam a executar.

As principais etapas da curadoria de dados incluem o pré-processamento, o aumento de dados e a auditoria regular.



**Pré-processamento de dados**

Pré-processe os dados para garantir que sejam precisos, completos e imparciais. Técnicas como limpeza de dados, normalização e seleção de recursos podem ajudar a eliminar vieses no conjunto de dados.

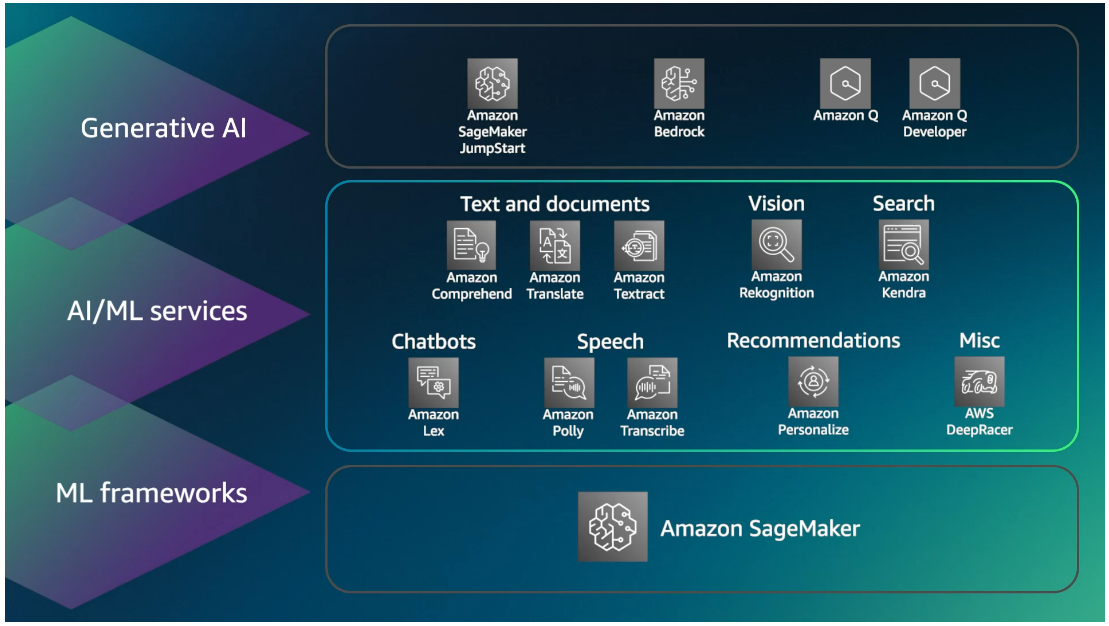
**Aumento de dados**

Use técnicas de aumento de dados para gerar novas instâncias de grupos sub-representados. Isso pode ajudar a equilibrar o conjunto de dados e evitar vieses em direção a grupos mais representados.

**Auditoria regular**

Audite regularmente o conjunto de dados para garantir que ele permaneça equilibrado e justo. Verifique se há vieses e tome medidas corretivas, se necessário.

# 5. AWS Infraestrutura e tecnologias



## 5.1 ML frameworks

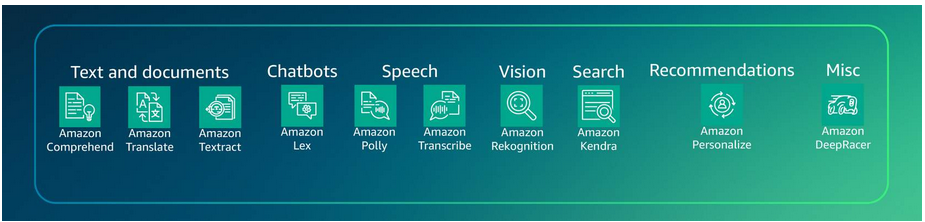


A camada de frameworks de ML desempenha um papel crucial no desenvolvimento e na implantação de modelos de machine learning. No centro dessa camada está o Amazon SageMaker. O SageMaker oferece as ferramentas certas para criar, treinar e executar LLMs e outros FMs de forma eficiente e econômica.

### 5.1.1 Amazon SageMaker

Com o SageMaker, você pode criar, treinar e implantar modelos de ML para qualquer caso de uso com infraestrutura, ferramentas e fluxos de trabalho totalmente gerenciados. O SageMaker elimina o trabalho pesado de cada etapa do processo de ML para facilitar o desenvolvimento de modelos de alta qualidade. O SageMaker fornece todos os componentes usados ​​para ML em um único conjunto de ferramentas, para que os modelos cheguem à produção mais rapidamente, com muito menos esforço e a um custo menor.

## 5.2 AI/ML Serviços



A AWS fornece uma camada robusta de serviços de IA/ML oferecendo soluções prontas para uso, como o Amazon Comprehend para tarefas de processamento de linguagem natural e o Amazon Kendra para pesquisa inteligente em dados organizacionais. Essa camada inclui uma ampla gama de serviços que fornecem aos desenvolvedores recursos de IA/ML sem a necessidade de gerenciamento extensivo de infraestrutura ou conhecimento especializado.

### 5.2.1 Amazon Comprehend

O Amazon Comprehend utiliza ML e processamento de linguagem natural (NLP) para ajudar você a descobrir insights e relacionamentos em seus dados não estruturados. Este serviço executa as seguintes funções:

Identifica o idioma do texto

Extrai frases-chave, lugares, pessoas, marcas ou eventos

Compreende o quão positivo ou negativo o texto é

Analisa o texto usando tokenização e classes gramaticais

E organiza automaticamente uma coleção de arquivos de texto por tópico

### 5.2.2 Amazon Translate

O Amazon Translate é um serviço de tradução automática neural que oferece traduções rápidas, de alta qualidade e acessíveis. A tradução automática neural é uma forma de automação de tradução de idiomas que utiliza modelos de aprendizado profundo para oferecer uma tradução mais precisa e natural do que os algoritmos tradicionais de tradução estatística e baseada em regras. Com o Amazon Translate, você pode localizar conteúdo, como sites e aplicativos, para seus diversos usuários, traduzir grandes volumes de texto para análise e implementar com eficiência a comunicação entre usuários em diferentes idiomas.

### 5.2.3 Amazon Textract

O Amazon Textract é um serviço que extrai automaticamente texto e dados de documentos digitalizados. Ele vai além do reconhecimento óptico de caracteres (OCR), identificando também o conteúdo de campos em formulários e informações armazenadas em tabelas.

### 5.2.4 Amazon Lex

O Amazon Lex é um serviço de IA totalmente gerenciado para projetar, construir, testar e implantar interfaces conversacionais em qualquer aplicativo usando voz e texto. O Amazon Lex oferece funcionalidades avançadas de aprendizado profundo, como reconhecimento automático de fala (ASR) para conversão de fala em texto e compreensão de linguagem natural (NLU) para reconhecimento da intenção do texto. Isso permite que você crie aplicativos com experiências de usuário altamente envolventes e interações conversacionais realistas, além de criar novas categorias de produtos. Com o Amazon Lex, as mesmas tecnologias de aprendizado profundo que impulsionam a Amazon Alexa agora estão disponíveis para qualquer desenvolvedor. Você pode criar com eficiência bots conversacionais sofisticados em linguagem natural e sistemas de resposta de voz interativa (IVR) habilitados para voz.

### 5.2.5 Amazon Polly

O Amazon Polly é um serviço que transforma texto em fala realista. O Amazon Polly permite criar aplicativos que falam, permitindo a criação de categorias totalmente novas de produtos com recursos de fala. O Amazon Polly é um serviço de IA que utiliza tecnologias avançadas de aprendizado profundo para sintetizar a fala, simulando a voz humana. O Amazon Polly inclui uma ampla seleção de vozes realistas, distribuídas em dezenas de idiomas, permitindo que você selecione a voz ideal e crie aplicativos com recursos de fala que funcionam em diversos países.

### 5.2.6 Amazon Transcribe

O Amazon Transcribe é um serviço de reconhecimento automático de fala (ASR) para conversão automática de fala em texto. O serviço pode transcrever arquivos de áudio armazenados em formatos comuns, como WAV e MP3, com carimbos de data e hora para cada palavra, para que você possa localizar rapidamente o áudio na fonte original pesquisando o texto. Você também pode enviar um fluxo de áudio ao vivo para o Amazon Transcribe e receber um fluxo de transcrições em tempo real. O Amazon Transcribe foi projetado para lidar com uma ampla gama de características acústicas e de fala, incluindo variações de volume, tom e velocidade de fala. Os clientes podem usar o Amazon Transcribe para uma variedade de aplicações comerciais, incluindo as seguintes:

Transcrição de chamadas de atendimento ao cliente baseadas em voz

Geração de legendas em conteúdo de áudio e vídeo

Condução de análise de conteúdo (baseada em texto) em conteúdo de áudio e vídeo

### 5.2.7 Amazon Rekognition

O Amazon Rekognition facilita a adição de análises de imagens e vídeos aos seus aplicativos. Ele utiliza tecnologia comprovada, altamente escalável e de aprendizado profundo, que não requer experiência em ML para ser usada. Com o Amazon Rekognition, você pode identificar objetos, pessoas, textos, cenas e atividades em imagens e vídeos, e até mesmo detectar conteúdo inapropriado. O Amazon Rekognition também oferece recursos de análise facial e busca facial altamente precisos. Você pode usá-lo para detectar, analisar e comparar rostos em uma ampla variedade de casos de uso para verificação de usuários, contagem de pessoas e segurança pública.

### 5.2.8 Amazon Kendra

O Amazon Kendra é um serviço de busca inteligente com tecnologia de aprendizado de máquina. O Amazon Kendra reinventa a busca corporativa para seus sites e aplicativos. Seus funcionários e clientes podem encontrar facilmente o conteúdo que procuram, mesmo quando ele está espalhado por vários locais e repositórios de conteúdo dentro da sua organização.

### 5.2.9 Amazon Personalize

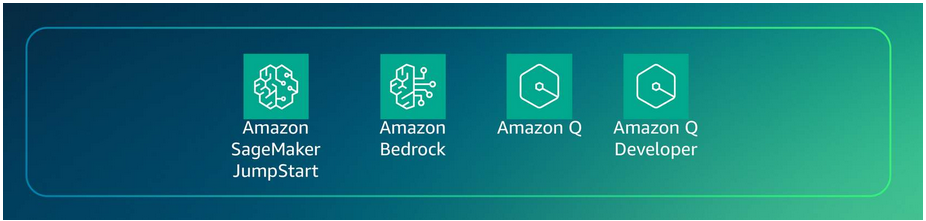
O Amazon Personalize é um serviço de ML que os desenvolvedores podem usar para criar recomendações individualizadas para clientes que utilizam seus aplicativos.

Com o Amazon Personalize, você fornece um fluxo de atividades do seu aplicativo (visualizações de página, cadastros, compras e assim por diante). Você também fornece um inventário dos itens que deseja recomendar, como artigos, produtos, vídeos ou músicas. Você pode optar por fornecer ao Amazon Personalize informações demográficas adicionais dos seus usuários, como idade ou localização geográfica. O Amazon Personalize processa e examina os dados, identifica o que é significativo, seleciona os algoritmos corretos e treina e otimiza um modelo de personalização personalizado para os seus dados.

### 5.2.10 AWS DeepRacer

O AWS DeepRacer é um carro de corrida em escala 1/18 que oferece uma maneira interessante e divertida de começar a usar o aprendizado por reforço (RL). RL é uma técnica avançada de ML que adota uma abordagem de treinamento de modelos muito diferente de outros métodos de ML. Seu superpoder é aprender comportamentos muito complexos sem a necessidade de dados de treinamento rotulados, além de poder tomar decisões de curto prazo enquanto otimiza para um objetivo de longo prazo.

## 5.3 IA Generativa



### 5.3.1 Amazon SageMaker Jumpstart

O SageMaker JumpStart ajuda você a começar a usar ML rapidamente. Para facilitar o início, o SageMaker JumpStart oferece um conjunto de soluções para os casos de uso mais comuns, que podem ser facilmente implantadas. As soluções são totalmente personalizáveis ​​e demonstram o uso de modelos e arquiteturas de referência do AWS CloudFormation para que você possa acelerar sua jornada em ML. O SageMaker JumpStart também oferece suporte à implantação e ao ajuste fino de mais de 150 modelos populares de código aberto, como processamento de linguagem natural, detecção de objetos e modelos de classificação de imagens, com apenas um clique.

### 5.3.2 Amazon Bedrock

O Amazon Bedrock é um serviço totalmente gerenciado que disponibiliza FMs da Amazon e de startups líderes em IA por meio de uma API. Com a experiência sem servidor do Amazon Bedrock, você pode começar rapidamente, experimentar FMs, personalizá-los de forma privada com seus próprios dados e integrar e implantar FMs perfeitamente em seus aplicativos da AWS.

### 5.3.3 Amazon Q

O Amazon Q pode ajudar você a obter respostas rápidas e relevantes para perguntas urgentes, resolver problemas, gerar conteúdo e tomar medidas usando os dados e a expertise encontrados nos repositórios de informações, códigos e sistemas corporativos da sua empresa. Ao conversar com o Amazon Q, você recebe informações e conselhos imediatos e relevantes para ajudar a otimizar tarefas, acelerar a tomada de decisões e estimular a criatividade e a inovação.

### 5.3.4 Amazon Q Developer

Projetado para melhorar a produtividade dos desenvolvedores, o Amazon Q Developer oferece recomendações de código baseadas em ML para acelerar o desenvolvimento de aplicativos em C#, Java, JavaScript, Python e TypeScript. O serviço se integra a diversos ambientes de desenvolvimento integrado (IDEs) e ajuda os desenvolvedores a escrever código mais rapidamente, gerando funções inteiras e blocos lógicos de código — geralmente com mais de 10 a 15 linhas de código.

## 5.4 Serviços e recursos da AWS para proteger sistemas de IA

### 5.4.1 Introdução à defesa em profundidade

Existem quatro serviços de segurança fundamentais da AWS recomendados para qualquer carga de trabalho, qualquer cliente e qualquer setor. Você pode obter benefícios da implementação desses serviços de segurança fundamentais profundamente integrados, como os seguintes:

* Central de Segurança
* AWS KMS
* Guard Duty
* AWS Shield Avançado

Cada serviço fornece proteção em um dos principais domínios de segurança de resposta a incidentes, proteção de dados, detecção de ameaças e proteção de rede e aplicativos.

### 5.4.2 Serviços de segurança da AWS

A AWS fornece vários serviços e recursos para proteger sistemas de IA. Os serviços a seguir são usados para gerenciar identidades de usuários e acesso a recursos, identificar e proteger dados confidenciais e proteger seus sistemas e aplicativos de IA.

**Identifique dados confidenciais antes de treinar modelos**

O Amazon Macie usa ML para automatizar a descoberta de dados confidenciais em grande escala.

**Gerencie identidades e acesso a serviços e recursos da AWS**

Com o AWS Identity and Access Management (IAM), você pode especificar quem ou o que pode acessar serviços e recursos na AWS.

**Limite o acesso aos seus dados, modelos e saídas**

Aplique uma política de privilégios mínimos a dados, modelos e aplicativos de treinamento usando o AWS IAM Identity Center e o IAM Access Analyzer. Aqui estão alguns outros serviços que você pode usar para limitar o acesso:

* Explore outros recursos de confiança zero para adicionar controles de acesso refinados com o AWS Verified Access e o Amazon Verified Permissions.
* Use o AWS Verified Access para eliminar ainda mais os custos, a complexidade e os problemas de desempenho relacionados a redes privadas virtuais (VPNs).

Você pode usar o Amazon SageMaker Role Manager para criar e gerenciar funções do IAM baseadas em persona para necessidades comuns de ML.

O Amazon SageMaker Role Manager fornece três personas de função pré-configuradas e permissões predefinidas para atividades comuns de ML. Essas personas de função são as seguintes:

* Persona de cientista de dados
* Pessoa MLOps
* Persona de computação do SageMaker

**Proteja os dados contra exfiltração (roubo de dados) e manipulação**

Para obter controles sólidos sobre a entrada e saída de dados de sistemas de IA, você pode definir políticas rígidas do AWS Network Firewall e da Amazon VPC. Isso impedirá a movimentação de dados para dentro e para fora de suas VPCs e redes.

**Proteja cargas de trabalho de IA com detecção inteligente de ameaças**

Além do Amazon GuardDuty, o Amazon Inspector e o Amazon Detective podem ajudar na detecção inteligente de ameaças. Esses serviços ajudam a identificar atividades suspeitas, como exfiltração de credenciais da AWS (roubo) e uso suspeito de APIs de usuários, incluindo APIs do Amazon Bedrock e do Amazon SageMaker.

**Automatize a resposta a incidentes e a conformidade**

Ao automatizar tarefas de segurança na AWS, você pode ficar mais seguro reduzindo erros de configuração humana. Isso dá à sua equipe mais tempo para se concentrar em outros trabalhos críticos para o seu negócio. A automação de tarefas torna mais conveniente para sua equipe de segurança trabalhar em estreita colaboração com as equipes de desenvolvedores e operações para criar e implantar código com mais rapidez e segurança.

Por exemplo, ao empregar tecnologias como ML, você pode descobrir, classificar e proteger dados confidenciais de forma automática e contínua na AWS.

Você também pode automatizar as verificações de segurança de infraestrutura e aplicativos para aplicar continuamente seus controles de segurança e conformidade e ajudar a garantir confidencialidade, integridade e disponibilidade em todos os momentos.

Você pode automatizar a resposta a incidentes e a conformidade com os serviços da AWS que você aprendeu anteriormente, como os seguintes:

* Hub de segurança da AWS
* Configuração da AWS
* Gerenciador de auditoria da AWS
* Artefato da AWS

**Defenda seus aplicativos e dados da Web de IA generativa**

Além do AWS Shield Advanced e do AWS Firewall Manager, que você aprendeu anteriormente, você também pode usar o AWS WAF para proteger seus aplicativos e dados da web.

O AWS WAF ajuda você a se proteger contra explorações comuns da web e bots que podem afetar a disponibilidade, comprometer a segurança ou consumir recursos excessivos. Com o AWS WAF, você pode fazer o seguinte:

* Filtre o tráfego da Web.
* Evite fraudes de controle de conta.
* Use o AWS WAF Bot Control para controlar o tráfego de bots generalizado (como scrapers, scanners, crawlers). O tráfego de bot generalizado pode consumir recursos em excesso, distorcer métricas, causar tempo de inatividade ou executar outras atividades indesejadas.

# 6. Vantagens e benefícios das soluções de IA AWS

## 6.1 Desenvolvimento e implantação acelerados

O Amazon Q Developer pode gerar código em tempo real. A Amazon lançou um desafio de produtividade durante a prévia do Amazon Q Developer. Os participantes que utilizaram o serviço tiveram 27% mais chances de concluir tarefas com sucesso, em média 57% mais rápido do que aqueles que não utilizaram o Amazon Q Developer.

O SageMaker lida com tarefas como pré-processamento de dados, treinamento de modelos e implantação. Assim, os desenvolvedores podem se concentrar na lógica da aplicação e na experiência do usuário.

O Amazon Bedrock fornece acesso a modelos e APIs pré-treinados. Assim, os desenvolvedores podem integrar rapidamente recursos de IA em suas aplicações sem a necessidade de treinamento extensivo ou hardware especializado. Isso acelera o processo de desenvolvimento e permite ciclos de iteração mais rápidos, reduzindo o tempo de lançamento no mercado de aplicações com tecnologia de IA.

## 6.2 Escalabilidade e otimização de custos

Com modelos de precificação por uso, as empresas pagam apenas pelos recursos que consomem. Isso reduz os custos iniciais e facilita a utilização eficiente dos recursos.

A infraestrutura global e os recursos de computação distribuída da AWS permitem que os aplicativos sejam escalonados perfeitamente entre regiões e lidem com grandes conjuntos de dados ou tráfego de alto volume.

## 6.3 Flexibilidade e acesso a modelos

A AWS atualiza e expande continuamente seus serviços de IA, fornecendo acesso aos avanços mais recentes em modelos, técnicas e algoritmos de aprendizado de máquina.

A Amazon Bedrock oferece uma seleção de FMs de alto desempenho de empresas líderes em IA, como AI21 Labs, Anthropic, Cohere, Meta, Mistral AI, Stability AI e AWS, por meio de uma única API.

## 6.4 Integração com ferramentas e serviços da AWS

Serviços como Amazon Comprehend e Amazon Rekognition oferecem recursos de IA prontos para uso que podem ser facilmente incorporados aos aplicativos.

Os serviços de IA da AWS integram-se perfeitamente a outros serviços da AWS, permitindo que os desenvolvedores criem soluções completas que utilizam múltiplos serviços de nuvem.

O ecossistema da AWS oferece uma ampla gama de ferramentas, SDKs e APIs, permitindo que os desenvolvedores incorporem recursos de IA aos seus aplicativos existentes ou criem aplicativos totalmente novos baseados em IA.

# 7. Custos

Ao trabalhar com serviços de IA e ML na AWS, é essencial entender as diversas considerações de custo envolvidas. Essas compensações podem impactar fatores como capacidade de resposta, disponibilidade, redundância, desempenho, cobertura regional, modelos de precificação, taxa de transferência e a capacidade de usar modelos personalizados.

## 7.1 Capacidade de resposta e disponibilidade

Os serviços de IA generativa da AWS são projetados para serem altamente responsivos e disponíveis. No entanto, níveis mais altos de responsividade e disponibilidade geralmente têm um custo maior. Por exemplo, serviços com menor latência e maior disponibilidade (por exemplo, implantação multirregional) normalmente terão preços mais altos em comparação com alternativas com garantias de desempenho e disponibilidade menores.

## 7.2 Redundância e Cobertura Regional

Para garantir redundância e alta disponibilidade, os serviços de IA generativa da AWS podem ser implantados em várias Zonas de Disponibilidade ou até mesmo em várias Regiões da AWS. Essa redundância tem um custo adicional, pois os recursos precisam ser provisionados e os dados replicados em vários locais.

## 7.3 Desempenho

A AWS oferece diferentes opções de computação (por exemplo, CPU, GPU e aceleradores de hardware personalizados) para serviços de IA generativa. Opções de alto desempenho, como instâncias de GPU, geralmente têm um custo mais alto, mas podem fornecer melhorias significativas de desempenho para determinadas cargas de trabalho.

## 7.4 Preços baseados em tokens

Muitos serviços de IA generativa da AWS, como o Amazon Q Developer e o Amazon Bedrock, usam um modelo de preços baseado em tokens. Isso significa que você paga pelo número de tokens (uma unidade de texto ou código) gerados ou processados ​​pelo serviço. Quanto mais tokens você gerar ou processar, maior será o custo.

## 7.5 Taxa de transferência provisionada

Alguns serviços de IA generativa da AWS, como Amazon Polly e Amazon Transcribe, permitem provisionar uma quantidade específica de taxa de transferência (por exemplo, capacidade de processamento de áudio ou texto) com antecedência. Níveis mais altos de taxa de transferência provisionada normalmente têm um custo mais alto, mas podem garantir um desempenho previsível para cargas de trabalho com tempo limitado.

## 7.6 Modelos personalizados

A AWS fornece modelos pré-treinados para diversas tarefas de IA generativa, mas você também pode trazer seus próprios modelos personalizados ou ajustar os modelos existentes. O treinamento e a implantação de modelos personalizados podem incorrer em custos adicionais, dependendo da complexidade do modelo, dos dados de treinamento e dos recursos computacionais necessários.

# 8. Exemplos de casos de uso do mundo real

## 8.1 Mídia e entretenimento

**Geração de conteúdo**: a IA pode criar roteiros, diálogos ou até mesmo histórias completas para filmes, programas de TV e jogos.

**Realidade virtual:** a IA pode criar ambientes virtuais imersivos e interativos para jogos ou simulações.

**Nova geração:** a IA pode gerar artigos ou resumos com base em dados brutos ou eventos.

## 8.2 Varejo

**Resumos de avaliações de produtos:** a IA pode gerar resumos de avaliações de produtos para que os consumidores encontrem informações relevantes rapidamente.

**Otimização de preços:** a IA pode modelar diferentes cenários de preços para determinar estratégias de precificação ideais que maximizem os lucros.

**Provas virtuais:** a IA pode gerar modelos virtuais de clientes para provas virtuais, o que pode melhorar a experiência de compra online.

**Otimização do layout da loja:** a IA pode gerar os layouts de loja mais eficientes para melhorar a experiência de compra do cliente e impulsionar as vendas.

## 8.3 Assistência médica

**AWS HealthScribe:** Este serviço da AWS capacita fornecedores de software para a área da saúde a criar aplicativos clínicos que geram notas clínicas automaticamente, analisando conversas entre pacientes e médicos.

**Medicina personalizada:** Ao gerar planos de tratamento com base na composição genética específica do paciente, no estilo de vida e na progressão da doença, a IA pode contribuir para um atendimento mais eficaz e personalizado.

**Aprimorar a imagem médica:** A IA pode aprimorar, reconstruir ou até mesmo gerar imagens médicas, como raios-X, ressonâncias magnéticas ou tomografias computadorizadas, que podem auxiliar em um diagnóstico mais preciso.

## 8.4 Ciências da vida

**Descoberta de fármacos:** a IA pode gerar novas estruturas moleculares potenciais para fármacos, acelerar o processo de descoberta de fármacos e reduzir custos.

**Previsão de dobramento de proteínas:** a IA pode prever as estruturas 3D de proteínas com base em sua sequência de aminoácidos, o que é crucial para a compreensão de doenças e o desenvolvimento de novas terapias.

**Biologia sintética:** a IA pode gerar projetos para sistemas biológicos sintéticos, como organismos projetados ou circuitos biológicos.

## 8.5 Serviços financeiros

**Mecanismos de detecção de fraudes:** a IA pode ajudar a criar conjuntos de dados sintéticos para aprimorar sistemas de IA e ML, simulando diversos padrões de lavagem de dinheiro.

**Gestão de portfólios:** a IA pode simular diversos cenários de mercado e auxiliar na criação e gestão de portfólios de investimento robustos.

**Cobrança de dívidas:** a IA pode gerar as estratégias de comunicação e negociação mais eficazes para cobrança de dívidas, aumentando a taxa de sucesso.

## 8.6 Fabricação

**Manutenção preditiva:** Ao analisar dados históricos de produção, a IA pode prever cronogramas de manutenção que proporcionarão os resultados mais eficientes das máquinas e reduzirão os tempos de inatividade.

**Otimização de processos:** A IA pode gerar os processos de produção mais eficientes modelando diferentes cenários e otimizando variáveis como custo, tempo, uso de recursos e assim por diante.

**Design de produto:** A IA pode ser usada para criar novos designs de produtos com base em parâmetros e restrições definidos. Ela pode gerar múltiplas opções de design e otimizar fatores como custo, materiais, desempenho e assim por diante.

**Ciência dos materiais:** A IA pode ajudar a gerar novas composições de materiais com as propriedades desejadas.

# 9 Exemplos de Aplicações de IA

## 9.1 Visão Computacional

Visão computacional é um campo da inteligência artificial que permite aos computadores interpretar e compreender imagens e vídeos digitais. O aprendizado profundo revolucionou a visão computacional, fornecendo técnicas poderosas para tarefas como classificação de imagens, detecção de objetos e segmentação de imagens.

## 9.2 Processamento de Linguagem Natural

PLN é um ramo da inteligência artificial que lida com a interação entre computadores e linguagens humanas. O aprendizado profundo fez avanços significativos no PLN. Ele pode executar tarefas como classificação de texto, análise de sentimentos, tradução automática e geração de linguagem.

## 9.3 Processamento Inteligente de Documentos

O PDI é um aplicativo que extrai e classifica informações de dados não estruturados, gera resumos e fornece insights acionáveis.

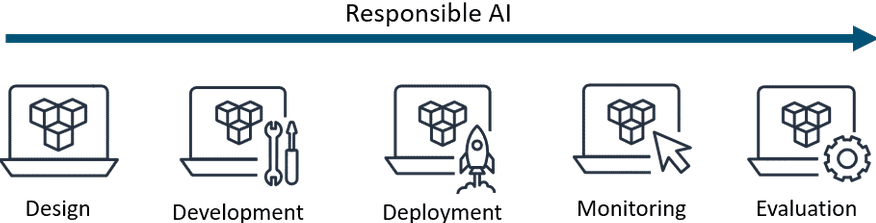
## 9.4 Detecção de Fraudes

Detecção de Fraudes refere-se ao processo de identificar e prevenir atividades fraudulentas ou comportamentos não autorizados com um sistema, processo ou transação.

# 10 IA responsável

IA responsável refere-se a práticas e princípios que garantem que os sistemas de IA sejam transparentes e confiáveis, ao mesmo tempo em que mitigam potenciais riscos e resultados negativos. Esses padrões responsáveis devem ser considerados durante todo o ciclo de vida de uma aplicação de IA. Isso inclui as fases iniciais de projeto, desenvolvimento, implantação, monitoramento e avaliação.

Para operar a IA de forma responsável, as empresas devem garantir proativamente o seguinte sobre seu sistema:



• Seja totalmente transparente e responsável, com mecanismos de monitoramento e supervisão implementados.

• Seja gerenciado por uma equipe de liderança responsável pelas estratégias de IA responsável.

• Seja desenvolvido por equipes com experiência em princípios e práticas de IA responsável.

• É construído seguindo as diretrizes de IA responsável.

## 10.1 Precisão dos modelos

O principal problema que os desenvolvedores enfrentam em aplicações de IA é a precisão. Tanto as aplicações de IA tradicionais quanto as generativas são alimentadas por modelos treinados em conjuntos de dados. Esses modelos podem fazer previsões ou gerar conteúdo com base apenas nos dados com os quais foram treinados. Se não forem treinados corretamente, você obterá resultados imprecisos. Portanto, é importante abordar o viés e a variância em seu modelo.

### 10.1.1 Viés

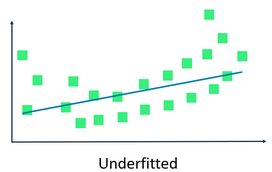
Viés em um modelo significa que ele não possui características importantes dos conjuntos de dados. Isso significa que os dados são muito básicos. O viés é medido pela diferença entre as previsões esperadas do modelo e os valores reais que estamos tentando prever. Se a diferença for pequena, o modelo tem viés baixo. Se a diferença for grande, o modelo tem viés alto. Quando um modelo tem viés alto, ele está subajustado. **Subajustado** significa que o modelo não está capturando diferenças suficientes nas características dos dados e, portanto, apresenta baixo desempenho nos dados de treinamento.

### 10.1.2 Variância

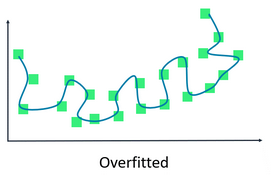
A variância se refere à sensibilidade do modelo a flutuações ou ruído nos dados de treinamento. O problema é que o modelo pode considerar o ruído nos dados como importante na saída. Quando a variância é alta, o modelo se torna tão familiarizado com os dados de treinamento que pode fazer previsões com alta precisão. Isso ocorre porque ele está capturando todas as características dos dados. No entanto, quando novos dados são introduzidos no modelo, sua precisão cai. Isso ocorre porque os novos dados podem ter características diferentes com as quais o modelo não foi treinado. Isso introduz o problema de sobreajuste. O **sobreajuste** ocorre quando o modelo tem um bom desempenho nos dados de treinamento, mas não nos dados de avaliação. Isso ocorre porque o modelo está memorizando os dados que viu e não consegue generalizar para exemplos não vistos.

### 10.1.3 Tradeoff viés-variância

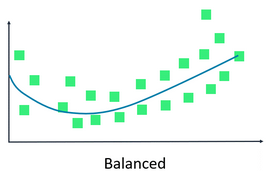
O tradeoff viés-variância ocorre quando você otimiza seu modelo com o equilíbrio certo entre viés e variância. Isso significa que você precisa otimizar seu modelo para que ele não fique subajustado ou superajustado. O objetivo é obter um modelo treinado com o menor tradeoff de viés e variância para um determinado conjunto de dados.



No exemplo subajustado, o viés é alto e a variância é baixa. Aqui, a regressão é uma linha reta. Isso nos mostra que o modelo está subajustando os dados porque não está capturando todas as características dos dados.



No exemplo superajustado, o viés é baixo e a variância é alta. Aqui, a curva de regressão se ajusta perfeitamente aos dados. Isso significa que ele está capturando ruído e, essencialmente, memorizando os dados. Ele não terá um bom desempenho com novos dados.



No exemplo balanceado, o viés é baixo e a regressão é baixa. Aqui, a regressão é uma curva. É isso que você quer. Captura características suficientes dos dados, sem capturar ruído.

Para ajudar a superar erros de viés e variância, você pode usar o seguinte:

**Validação Cruzada**

A validação cruzada é uma técnica para avaliar modelos de ML treinando vários modelos de ML em subconjuntos dos dados de entrada disponíveis e avaliando-os no subconjunto complementar dos dados. A validação cruzada deve ser usada para detectar overfitting.

**Aumentar a Amostra**

Adicione mais amostras de dados para aumentar o escopo de aprendizado do modelo.

**Regularização**

Use regularização. A regularização é um método que penaliza valores de peso extremos para ajudar a evitar que modelos lineares overfitting exemplos de dados de treinamento.

**Modelos mais simples**

Use arquiteturas de modelo mais simples para ajudar com overfitting. Se o modelo estiver subajustado, ele pode ser muito simples.

**Redução de dimensão**

Aplique redução de dimensão. A redução de dimensão é um algoritmo de aprendizado de máquina não supervisionado que tenta reduzir a dimensionalidade (número de características) em um conjunto de dados, mantendo o máximo de informações possível.

**Encerrar treinamento antecipadamente**

Encerre o treinamento antecipadamente para que o modelo não memorize os dados.

## 10.2 Dimensões centrais da IA responsável

### 10.2.1 Justiça

A justiça é crucial para o desenvolvimento de sistemas de IA responsáveis. Com justiça, os sistemas de IA promovem a inclusão, previnem a discriminação, defendem valores e normas legais responsáveis e constroem confiança com a sociedade.

### 10.2.2 Explicabilidade

Explicabilidade refere-se à capacidade de um modelo de IA de explicar ou justificar claramente seus mecanismos e decisões internos, de modo que sejam compreensíveis para os humanos.

Os humanos devem entender como os modelos estão tomando decisões e abordar quaisquer questões de viés, confiança ou justiça.

### 10.2.3 Privacidade e segurança

Privacidade e segurança na IA responsável referem-se a dados protegidos contra roubo e exposição. Mais especificamente, isso significa que, em termos de privacidade, os indivíduos controlam quando e se seus dados podem ser usados. Em termos de segurança, verifica-se se nenhum sistema ou usuário não autorizado terá acesso aos dados do indivíduo.

### 10.2.4 Transparência

A transparência comunica informações sobre um sistema de IA para que as partes interessadas possam fazer escolhas informadas sobre o uso do sistema. Algumas dessas informações incluem processos de desenvolvimento, capacidades e limitações do sistema.

Ela fornece a indivíduos, organizações e partes interessadas acesso para avaliar a imparcialidade, a robustez e a explicabilidade dos sistemas de IA. Eles podem identificar e mitigar potenciais vieses, reforçar padrões responsáveis e promover a confiança na tecnologia.

### 10.2.5 Veracidade e robustez

Veracidade e robustez em IA referem-se aos mecanismos para garantir que um sistema de IA opere de forma confiável, mesmo em situações inesperadas, incertezas e erros.

O objetivo da veracidade e robustez na IA responsável é desenvolver modelos de IA resilientes a mudanças nos parâmetros de entrada, distribuição de dados e circunstâncias externas.

Isso significa que o modelo de IA deve manter a confiabilidade, a precisão e a segurança em ambientes incertos.

### 10.2.6 Governança

Governança é um conjunto de processos usados para definir, implementar e aplicar práticas de IA responsáveis dentro de uma organização.

A governança aborda vários problemas de responsabilidade, legais ou sociais que a IA generativa pode gerar.

Por exemplo, políticas de governança podem ajudar a proteger os direitos dos indivíduos à propriedade intelectual. Também podem ser usadas para garantir a conformidade com leis e regulamentos. A governança é um componente vital da IA responsável para uma organização que busca incorporar as melhores práticas responsáveis.

### 10.2.7 Segurança

A segurança na IA responsável refere-se ao desenvolvimento de algoritmos, modelos e sistemas de forma que sejam responsáveis, seguros e benéficos para os indivíduos e para a sociedade como um todo.

Isso significa que os sistemas de IA devem ser cuidadosamente projetados e testados para evitar causar danos não intencionais aos seres humanos ou ao meio ambiente. Aspectos como viés, uso indevido e impactos descontrolados precisam ser considerados proativamente.

### 10.2.8 Controlabilidade

Controlabilidade em IA responsável refere-se à capacidade de monitorar e orientar o comportamento de um sistema de IA para alinhá-lo aos valores e intenções humanas. Envolve o desenvolvimento de arquiteturas controláveis, para que quaisquer problemas não intencionais possam ser gerenciados e resolvidos.

## 10.3 Benefícios comerciais da IA responsável

### 10.3.1 Aumento da confiança e da reputação

Os clientes têm maior probabilidade de interagir com aplicações de IA se acreditarem que o sistema é justo e seguro. Isso melhora sua reputação e o valor da marca.

### 10.3.2 Conformidade regulatória

À medida que as regulamentações de IA surgem, as empresas com estruturas éticas robustas de IA estão melhores posicionadas para cumprir as diretrizes sobre privacidade de dados, justiça, responsabilidade e transparência.

### 10.3.3 Mitigação de riscos

Práticas de IA responsável ajudam a mitigar riscos como viés, violações de privacidade, brechas de segurança e impactos negativos não intencionais na sociedade. Isso reduz responsabilidades legais e custos financeiros.

### 10.3.4 Vantagem competitiva

Empresas que priorizam a IA responsável podem se diferenciar dos concorrentes e obter uma vantagem competitiva, especialmente à medida que a conscientização do consumidor sobre a ética da IA aumenta.

### 10.3.5 Melhoria na tomada de decisões

Sistemas de IA desenvolvidos com justiça, responsabilidade e transparência em mente são mais confiáveis e menos propensos a produzir resultados tendenciosos ou falhos, o que leva a melhores decisões baseadas em dados.

### 10.3.6 Melhoria nos produtos e negócios

A IA responsável incentiva uma abordagem diversificada e inclusiva para o desenvolvimento da IA. Por se basear em perspectivas e experiências variadas, ele pode gerar soluções mais criativas e inovadoras.

## 10.4 Serviços e Ferramentas da Amazon para IA Responsável

A AWS oferece serviços como o Amazon SageMaker e o Amazon Bedrock, que possuem ferramentas integradas para ajudar você com IA responsável. Essas ferramentas abrangem tópicos como avaliação de modelos básicos, salvaguardas para IA generativa, detecção de vieses, explicações sobre predição de modelos, monitoramento e revisões humanas e aprimoramento da governança.

## 10.5 Análise das ferramentas de serviço da Amazon para IA responsável

### 10.5.1 Avaliação do modelo de base

Você deve sempre avaliar um FM para determinar se ele é adequado ao seu caso de uso específico. Para ajudar você a fazer isso, a Amazon oferece avaliação de modelos no Amazon Bedrock e no Amazon SageMaker Clarify.

**Avaliação de modelos no Amazon Bedrock**

Com a avaliação de modelos no Amazon Bedrock, você pode avaliar, comparar e selecionar o melhor modelo de base para o seu caso de uso em apenas alguns cliques. O Amazon Bedrock oferece opções de avaliação automática e humana.

A avaliação automática oferece métricas predefinidas, como precisão, robustez e toxicidade.

A avaliação humana oferece métricas subjetivas ou personalizadas, como simpatia, estilo e alinhamento com a voz da marca. Para avaliação humana, você pode usar seus funcionários internos ou uma equipe gerenciada pela AWS como revisores.

**SageMaker Clarify**

O SageMaker Clarify oferece suporte à avaliação de FM. Você pode avaliar automaticamente os FMs para seu caso de uso de IA generativa com métricas como precisão, robustez e toxicidade para apoiar sua iniciativa de IA responsável.

Para critérios ou conteúdo diferenciado que exija julgamento humano sofisticado, você pode optar por usar sua própria força de trabalho ou uma força de trabalho gerenciada fornecida pela AWS para revisar as respostas do modelo.

### 10.5.2 Proteções para IA generativa

Com o **Amazon Bedrock Guardrails,** você pode implementar proteções para seus aplicativos de IA generativa com base em seus casos de uso e políticas de IA responsável. As Guardrails ajudam a controlar a interação entre usuários e FMs, filtrando conteúdo indesejado e prejudicial, eliminando informações de identificação pessoal (PII) e aprimorando a segurança e a privacidade do conteúdo em aplicações de IA generativa. Você pode criar várias guardrails com diferentes configurações, adaptadas a casos de uso específicos. Além disso, você pode monitorar e analisar continuamente as entradas do usuário e as respostas do FM que podem violar as políticas definidas pelo cliente nas guardrails.

**10.5.2.1 Nível consistente de segurança da IA**

O Amazon Bedrock Guardrails avalia as entradas do usuário e as respostas do FM com base em políticas específicas do caso de uso e fornece uma camada adicional de salvaguardas, independentemente do FM subjacente. O Amazon Bedrock Guardrails pode ser aplicado em FMs, incluindo Anthropic Claude, Meta Llama 2, Cohere Command, AI21 Labs Jurassic, Amazon Titan Text e modelos ajustados. Os clientes podem criar vários guardrails, cada um configurado com uma combinação diferente de controles, e usá-los em diferentes aplicativos e casos de uso. O Amazon **Bedrock Guardrails** também pode ser integrado ao **Amazon Bedrock Agents** para criar aplicativos de IA generativa alinhados às suas políticas de IA responsáveis.

**10.5.2.2 Bloqueie tópicos indesejados**

Ao usar uma descrição curta e em linguagem natural, o Amazon Bedrock Guardrails permite definir um conjunto de tópicos a serem evitados no contexto do seu aplicativo. O Amazon Bedrock Guardrails detecta e bloqueia entradas de usuários e respostas de FM que se enquadram em tópicos restritos. Por exemplo, um assistente bancário pode ser projetado para evitar tópicos relacionados a conselhos de investimento.

**10.5.2.3 Filtrar conteúdo prejudicial**

O Amazon Bedrock Guardrails fornece filtros de conteúdo com limites configuráveis para filtrar conteúdo prejudicial nas categorias de ódio, insultos, conteúdo sexual e violência. A maioria dos FMs já oferece proteções integradas para impedir a geração de respostas prejudiciais. Além dessas proteções, o Amazon Bedrock Guardrails permite configurar limites nas diferentes categorias para filtrar interações prejudiciais. O Amazon Bedrock Guardrails avalia automaticamente as consultas dos usuários e as respostas de FM para detectar e ajudar a prevenir conteúdo que se enquadra em categorias restritas. Por exemplo, um site de comércio eletrônico pode projetar seu assistente online para evitar o uso de linguagem imprópria, como discurso de ódio ou insultos.

**10.5.2.4 Redigir PII para proteger a privacidade do usuário**

O Amazon Bedrock Guardrails ajuda a detectar PII em entradas de usuários e respostas de FM. Com base no caso de uso, você pode rejeitar seletivamente entradas que contenham PII ou redigir PII em respostas de FM. Por exemplo, você pode redigir informações pessoais dos usuários enquanto gera resumos de transcrições de conversas de clientes e agentes em um call center.

### 10.5.3 Detecção de viés

O **SageMaker Clarify** ajuda a identificar possíveis vieses em modelos e conjuntos de dados de aprendizado de máquina sem a necessidade de codificação extensiva. Você especifica recursos de entrada, como gênero ou idade, e o SageMaker Clarify executa uma tarefa de análise para detectar possíveis vieses nesses recursos. O SageMaker Clarify então fornece um relatório visual com uma descrição das métricas e medições de viés potencial para que você possa identificar as etapas para remediar o viés.

Você pode usar o **Amazon SageMaker Data Wrangler** para balancear seus dados em caso de desequilíbrios. O SageMaker Data Wrangler oferece três operadores de balanceamento: **subamostragem aleatória, superamostragem aleatória e a Técnica de Sobreamostragem de Minorias Sintéticas (SMOTE) para rebalancear dados em seus conjuntos de dados desbalanceados.**

### 10.5.4 Explicação da predição do modelo

O SageMaker Clarify é integrado ao Amazon SageMaker Experiments para fornecer pontuações detalhando quais recursos mais contribuíram para a predição do seu modelo em uma entrada específica para modelos tabulares, de processamento de linguagem natural (PLN) e de visão computacional. Para conjuntos de dados tabulares, o SageMaker Clarify também pode gerar um gráfico de importância de recursos agregados que fornece insights sobre o processo geral de previsão do modelo. Esses detalhes podem ajudar a determinar se uma entrada específica do modelo tem mais influência do que o esperado no comportamento geral do modelo.

O SageMaker Experiments é um recurso do SageMaker que você pode usar para criar, gerenciar, analisar e comparar seus experimentos de aprendizado de máquina.

### 10.5.5 Monitoramento e revisões humanas

**Amazon SageMaker Model Monitor**

O Amazon SageMaker Model Monitor monitora a qualidade dos modelos de aprendizado de máquina do SageMaker em produção. Você pode configurar o monitoramento contínuo com um endpoint em tempo real (ou uma tarefa de transformação em lote executada regularmente) ou o monitoramento programado para tarefas de transformação em lote assíncronas. Com o SageMaker Model Monitor, você pode definir alertas que o notificam quando há desvios na qualidade do modelo. Com a detecção precoce e proativa desses desvios, você pode tomar ações corretivas.

**Amazon A2I**

Amazon Augmented AI (Amazon A2I) é um serviço que ajuda a criar os fluxos de trabalho necessários para a revisão humana de previsões de ML. O Amazon A2I oferece revisão humana a todos os desenvolvedores e elimina o trabalho pesado e indiferenciado associado à criação de sistemas de revisão humana ou ao gerenciamento de um grande número de revisores humanos.

### 10.5.6 Aprimoramento da governança

O SageMaker fornece ferramentas de governança específicas para ajudar você a implementar a IA de forma responsável. Essas ferramentas oferecem maior controle e visibilidade sobre seus modelos de IA. Você pode capturar e compartilhar informações do modelo e se manter informado sobre o comportamento do modelo, como vieses, tudo em um só lugar.

As ferramentas de governança incluem o seguinte:

• Amazon SageMaker Role Manager: com o SageMaker Role Manager, os administradores podem definir permissões mínimas em minutos.

• Amazon SageMaker Model Cards: com o SageMaker Model Cards, você pode capturar, recuperar e compartilhar informações essenciais do modelo, como usos pretendidos, classificações de risco e detalhes de treinamento, desde a concepção até a implantação.

• Painel de Modelos do Amazon SageMaker: Com o Painel de Modelos do SageMaker, você pode manter sua equipe informada sobre o comportamento do modelo em produção, tudo em um só lugar.

### 10.5.7 Proporcionando transparência

Os Cartões de Serviço de IA da AWS são um recurso para ajudar você a entender melhor os serviços de IA da AWS. Os Cartões de Serviço de IA são uma forma de documentação de IA responsável que fornece um único local para encontrar informações sobre os casos de uso e limitações pretendidos, as opções de design de IA responsável e as práticas recomendadas de implantação e otimização de desempenho para os serviços de IA da AWS.

Cada Cartão de Serviço de IA contém quatro seções que abrangem o seguinte:

• Conceitos básicos para ajudar os clientes a entenderem melhor o serviço ou os recursos do serviço

• Casos de uso e limitações pretendidos

• Considerações sobre o design de IA responsável

• Orientações sobre implantação e otimização de desempenho

## 10.6 Transparência e Explicabilidade

A transparência ajuda a entender **COMO** um modelo toma decisões. A transparência também facilita a auditoria do sistema.

A explicabilidade ajuda a entender **POR QUE** o modelo tomou a decisão que tomou. Ela fornece insights sobre as limitações de um modelo.

### 10.6.1 Modelos transparentes e explicáveis comparados a modelos de caixa-preta

Modelos que carecem de transparência e explicabilidade são frequentemente chamados de modelos de caixa-preta. Esses modelos usam algoritmos complexos e inúmeras camadas de redes neurais para fazer previsões, mas não fornecem insights sobre seu funcionamento interno.

Modelos transparentes e explicáveis têm várias vantagens sobre modelos de caixa-preta.

**Maior confiança**

**Mais fácil de depurar e otimizar para melhorias**

**Melhor compreensão dos dados e do processo de tomada de decisão do modelo**

### 10.6.2 Soluções para modelos transparentes e explicáveis

Não existe uma solução padrão para a criação de modelos transparentes e explicáveis. Dependendo do caso de uso do modelo, você pode usar técnicas diferentes.

**Estruturas de explicabilidade**

Existem várias estruturas de explicabilidade disponíveis, como SHapley Value Added (SHAP), Layout-Independent Matrix Factorization (LIME) e Counterfactual Explanations, que podem ajudar a resumir e interpretar as decisões tomadas por sistemas de IA.

**Documentação transparente**

Mantenha documentação clara e abrangente da arquitetura do sistema de IA.

**Monitoramento e auditoria**

Os sistemas de IA devem ser monitorados e auditados para garantir que estejam funcionando conforme o esperado e não apresentem viés ou comportamento discriminatório.

**Supervisão e envolvimento humano**

Incorpore a supervisão e o envolvimento humanos em processos críticos de tomada de decisão.

**Explicações contrafactuais**

Forneça explicações contrafactuais que mostrem como a saída mudaria se certos recursos de entrada fossem diferentes.

**Explicações da interface do usuário**

Projete interfaces de usuário que forneçam explicações claras e compreensíveis das saídas, da lógica e das limitações do sistema de IA aos usuários finais, para que eles possam tomar decisões informadas.

### 10.6.3 Riscos de modelos transparentes e explicáveis

• Aumentar a complexidade do desenvolvimento e da manutenção do modelo pode aumentar os custos.

• Criar vulnerabilidades no modelo, nos dados e nos algoritmos pode ser explorado por pessoas mal-intencionadas.

• Apresentar expectativas irreais de que o modelo seja perfeitamente transparente e explicável. Em algumas situações, isso pode não ser possível ou mesmo pretendido.

• Fornecer informações em excesso pode gerar preocupações com privacidade e segurança. Isso também pode comprometer a vantagem competitiva do modelo.

## 10.7 Ferramentas da AWS para transparência

Para ajudar na transparência, a Amazon oferece os **Cartões de Serviço de IA da AWS e os Cartões de Modelo do Amazon SageMaker**. A diferença entre eles é que, com os Cartões de Serviço de IA, a Amazon fornece documentação transparente sobre os serviços da Amazon que ajudam você a construir seus serviços de IA. Com os Cartões de Modelo do SageMaker, você pode catalogar e fornecer documentação sobre modelos que você cria ou desenvolve.

## 10.8 Ferramentas da AWS para explicabilidade

### 10.8.1 SageMaker Clarify

O SageMaker Clarify é integrado ao SageMaker Experiments para fornecer pontuações detalhando quais recursos contribuíram mais para a previsão do seu modelo em uma entrada específica para modelos tabulares, de PLN e de visão computacional. Para conjuntos de dados tabulares, o SageMaker Clarify também pode gerar um gráfico de importância de recursos agregados, que fornece insights sobre o processo geral de previsão do modelo. Esses detalhes podem ajudar a determinar se uma entrada específica do modelo tem mais influência do que o esperado no comportamento geral do modelo.

### 10.8.2 SageMaker Autopilot

O Amazon SageMaker Autopilot usa ferramentas fornecidas pelo SageMaker Clarify para ajudar a fornecer insights sobre como os modelos de ML fazem previsões. Essas ferramentas podem ajudar engenheiros de ML, gerentes de produto e outras partes interessadas internas a entender as características do modelo. Para confiar e interpretar as decisões tomadas com base nas previsões do modelo, tanto consumidores quanto reguladores confiam na transparência do aprendizado de máquina.

A funcionalidade explicativa do SageMaker Autopilot determina a contribuição de recursos ou entradas individuais para a saída do modelo e fornece insights sobre a relevância de diferentes recursos. Você pode usá-lo para entender por que um modelo fez uma previsão após o treinamento ou usá-lo para fornecer uma explicação por instância durante a inferência.

## 10.9 Compensações (trade-offs) de interpretabilidade

A interpretabilidade é uma característica da transparência do modelo. Interpretabilidade é o grau em que um ser humano consegue entender a causa de uma decisão. Isso pode soar muito parecido com explicabilidade, mas há uma diferença.

**Interpretabilidade**

Interpretabilidade é o acesso a um sistema para que um ser humano possa interpretar a saída do modelo com base nos pesos e características. Por exemplo, se uma empresa deseja alta transparência do modelo e quer entender exatamente por que e como o modelo está gerando previsões, ela precisa observar a mecânica interna do método de IA/ML.

Exemplo: Um economista pode querer construir um modelo de regressão multivariada para prever uma taxa de inflação. Ele pode visualizar os parâmetros estimados das variáveis do modelo para medir a saída esperada, considerando diferentes exemplos de dados. Nesse caso, a transparência total é garantida e o economista pode responder exatamente o porquê e como do comportamento do modelo.

**Explicabilidade**

Explicabilidade é como pegar um modelo de ML e explicar o comportamento em termos humanos. Por meio de métodos agnósticos de modelo (por exemplo, gráficos de dependência parcial, gráficos de dependência SHAP ou modelos substitutos), é possível descobrir o significado entre as atribuições de dados de entrada e as saídas do modelo. Com essa compreensão, é possível explicar a natureza e o comportamento do modelo de IA/ML.

Exemplo: Um veículo de comunicação utiliza uma rede neural para atribuir categorias a diferentes artigos. O veículo não consegue interpretar o modelo em profundidade. No entanto, pode utilizar uma abordagem agnóstica de modelo para avaliar os dados do artigo de entrada em comparação com as previsões do modelo. Com essa abordagem, descobre-se que o modelo está atribuindo a categoria esportes a artigos de negócios que mencionam organizações esportivas. Embora o veículo de comunicação não tenha utilizado a interpretabilidade do modelo, ainda assim foi possível derivar uma resposta explicável para revelar o comportamento do modelo.

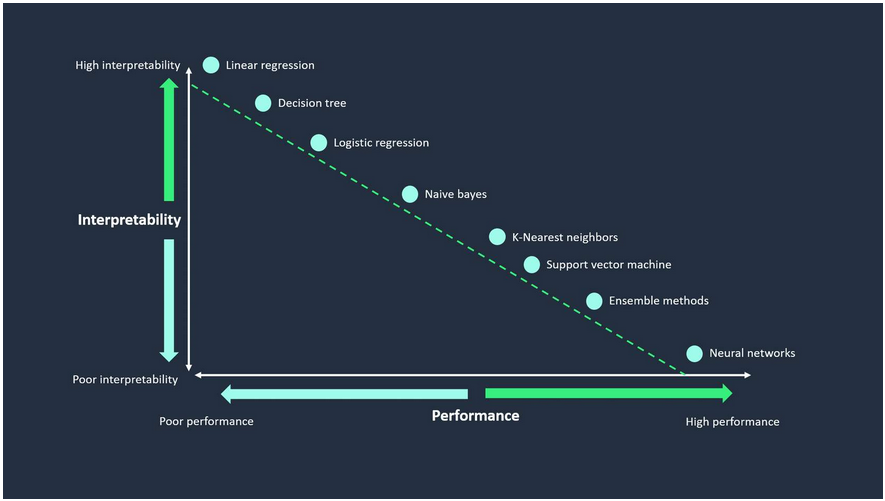


Diagrama que mostra como a interpretabilidade de um modelo pode afetar o desempenho.

Se uma empresa deseja alta transparência do modelo e deseja entender exatamente por que e como o modelo está gerando previsões, precisa de um modelo que ofereça **interpretabilidade**. No entanto, uma alta interpretabilidade normalmente tem um custo em termos de desempenho, como visto no diagrama.

Se uma empresa deseja atingir alto desempenho, mas ainda deseja ter uma compreensão geral do comportamento do modelo, a **explicabilidade** do modelo passa a desempenhar um papel mais importante.

Ao iniciar um novo projeto de IA/ML, você precisa considerar se a interpretabilidade é necessária. A explicabilidade do modelo pode ser usada em qualquer caso de uso de IA/ML, mas se for necessária transparência detalhada, a seleção do método de IA/ML se torna limitada.

## 10.10 Compensações (trade-offs) entre segurança e transparência

### 10.10.1 Segurança do modelo

A segurança do modelo é a capacidade de um sistema de IA de evitar causar danos em suas interações com o mundo. Isso inclui evitar danos sociais, como viés em algoritmos de tomada de decisão, e evitar exposições a vulnerabilidades de privacidade e segurança. A segurança do modelo é importante para garantir que os sistemas de IA sejam usados de maneiras que beneficiem a sociedade e não causem danos a indivíduos ou grupos.

### 10.10.2 Compensações entre segurança e transparência do modelo

Com a segurança do modelo se concentrando na proteção de informações e a transparência do modelo se concentrando na exposição de informações, você pode entender que há um equilíbrio delicado necessário entre elas.

**Precisão**

Modelos complexos, como grandes redes neurais, tendem a ser mais precisos, mas menos interpretáveis, do que modelos lineares mais simples, que são mais transparentes.

**Privacidade**

Técnicas de preservação da privacidade, como privacidade diferencial, podem melhorar a segurança, mas dificultar a inspeção dos modelos. Isso pode torná-los menos transparentes.

**Segurança**

Restringir ou filtrar as saídas do modelo por motivos de segurança pode reduzir a transparência do raciocínio original do modelo.

Modelos de trem com isolamento térmico altamente seguros (modelos treinados em redes privadas e sem acesso a dados externos) podem ser menos abertos à auditoria externa.

### 10.10.3 Controlabilidade do modelo

Um modelo controlável é aquele em que você pode influenciar as previsões e o comportamento do modelo alterando aspectos dos dados de treinamento. Uma maior controlabilidade proporciona mais transparência ao modelo e permite a correção de vieses e saídas indesejados.

A controlabilidade do modelo é medida pelo nível de controle que você tem sobre o modelo, alterando os dados de entrada. Modelos mais controláveis são mais fáceis de direcionar para os comportamentos desejados. Isso é importante para a imparcialidade, pois você precisa ser capaz de entender e controlar o viés no modelo. A controlabilidade de um modelo também é importante para a transparência e a depuração do modelo.

A controlabilidade depende da arquitetura do modelo. Modelos lineares tendem a ser mais controláveis do que modelos neurais complexos. Você pode testar a controlabilidade avaliando se a manipulação dos dados, como adicionar ou remover exemplos, causa as mudanças esperadas nas saídas e previsões do modelo. A controlabilidade pode ser aprimorada por meio de técnicas de aumento de dados e pela adição de restrições ao processo de treinamento do modelo.

## 10.11 Princípios do Design Centrado no Ser Humano para IA Explicável

### 10.11.1 Design para tomada de decisão amplificada

O princípio do design para tomada de decisão amplificada apoia os tomadores de decisão em situações de alto risco. Este princípio busca maximizar os benefícios do uso da tecnologia, minimizando potenciais riscos e erros, especialmente aqueles que podem ocorrer quando humanos tomam decisões sob estresse ou em ambientes de alta pressão. Isso pode levar a melhores resultados para indivíduos, organizações e a sociedade como um todo.

Principais aspectos do design para tomada de decisão amplificada

**Clareza**

O design para clareza garante que as informações sejam apresentadas de uma forma fácil de entender e interpretar, sem introduzir vieses ou mal-entendidos.

**Simplicidade**

Projetar para a simplicidade minimiza a quantidade de informações que precisam ser processadas pelo usuário, ao mesmo tempo em que fornece todas as informações necessárias para a tomada de decisão.

**Usabilidade**

Projetar para a usabilidade significa projetar uma tecnologia que seja fácil de usar e navegar, independentemente do nível de conhecimento ou habilidades técnicas do usuário.

**Reflexividade**

Projetar para a reflexividade significa projetar uma tecnologia que leve os usuários a refletir sobre seu processo de tomada de decisão e os incentive a assumir a responsabilidade por suas escolhas.

**Responsabilidade**

Projetar para a responsabilidade atribui consequências às decisões tomadas usando tecnologia amplificada, para que os usuários sejam responsabilizados por suas ações.

### 10.11.2 Design para tomada de decisão imparcial

O princípio e as práticas do design para tomada de decisão imparcial visam garantir que o design de processos, sistemas e ferramentas de tomada de decisão esteja livre de vieses que possam influenciar os resultados. Isso pode ter impactos significativos nos resultados da tomada de decisão e ajudar a promover a justiça e o uso eficiente dos recursos.

O design para tomada de decisão imparcial envolve:

• Identificar e avaliar potenciais vieses.

•Desenvolver processos e ferramentas de tomada de decisão que sejam transparentes e justos.

• Capacitar os tomadores de decisão para reconhecer e mitigar vieses.

Aspectos-chave da concepção para uma tomada de decisão imparcial

**Transparência**

**Igualdade**

**Treinamento**

### 10.11.3 Design para aprendizagem humana e de IA

Design para aprendizagem humana e de IA é um processo que visa criar ambientes e ferramentas de aprendizagem que sejam benéficos e eficazes tanto para humanos quanto para a IA. Ele abrange uma gama de estratégias e abordagens que levam em consideração os pontos fortes e as limitações únicas de cada aluno, bem como os objetivos e propósitos da experiência de aprendizagem.

Principais aspectos do design para aprendizagem humana e de IA

**Aprendizagem cognitiva**

Na aprendizagem de IA, isso envolve a criação de ambientes de aprendizagem onde os sistemas de IA aprendem com instrutores e especialistas humanos e adquirem experiência e conhecimento por meio de cenários simulados ou do mundo real.

**Personalização**

Ao usar análise de dados e algoritmos de ML, os desenvolvedores podem criar recomendações e algoritmos de aprendizagem personalizados que se adaptam ao estilo de aprendizagem e às necessidades únicas de cada aluno.

**Design centrado no usuário**

O design centrado no usuário envolve a criação de ambientes e ferramentas de aprendizagem intuitivos e acessíveis a uma ampla gama de alunos, incluindo aqueles com deficiências ou barreiras linguísticas. Ao priorizar a experiência do usuário e a usabilidade, os designers podem garantir que os ambientes de aprendizagem sejam eficazes e envolventes para todos os usuários.

### 10.11.4 Aprendizagem por reforço a partir do feedback humano

Aprendizagem por reforço a partir do feedback humano (APR) é uma técnica de ML que utiliza feedback humano para otimizar modelos de ML e promover a autoaprendizagem com mais eficiência. As técnicas de aprendizagem por reforço (APR) treinam o software para tomar decisões que maximizam as recompensas, o que torna seus resultados mais precisos. A APR incorpora o feedback humano na função de recompensas, para que o modelo de ML possa executar tarefas alinhadas aos objetivos, desejos e necessidades humanas. A APR é usada tanto em aplicações de IA tradicional quanto em IA generativa.

Alguns dos benefícios do RLHF incluem o seguinte:

• Melhora o desempenho da IA

• Fornece parâmetros de treinamento complexos

• Aumenta a satisfação do usuário

### 10.11.5 Amazon SageMaker Ground Truth

O SageMaker Ground Truth oferece o conjunto mais abrangente de recursos com interação humana para incorporar feedback humano em todo o ciclo de vida do ML, a fim de melhorar a precisão e a relevância do modelo. O SageMaker Ground Truth inclui um anotador de dados para recursos de RLHF. Você pode fornecer feedback e orientação diretos sobre a saída gerada por um modelo. Os dados, chamados de dados de comparação e classificação, são efetivamente um modelo de recompensa ou função de recompensa que é então usado para treinar o modelo. Você pode usar dados de comparação e classificação para personalizar um modelo existente para o seu caso de uso ou para ajustar um modelo criado do zero.

# 11. Ciclo de vida do aplicativo de IA Generativa

## 11.1 Defina um caso de uso

Na primeira etapa, são identificados os requisitos para incorporar recursos de IA generativa em um aplicativo. Isso pode envolver a análise das funcionalidades do aplicativo, das necessidades do usuário e das metas de negócios para determinar onde a IA generativa pode agregar valor.

Esta fase é a base que define o caminho para todo o projeto, fazendo o seguinte:

•  Definindo o problema a ser resolvido

•  Reunindo requisitos relevantes

•  Alinhando as expectativas das partes interessadas

Acertar esse estágio é imperativo, porque informa todas as etapas subsequentes e, em última análise, determina o sucesso ou o fracasso do aplicativo de IA generativa. Durante essa fase crucial, as equipes devem analisar cuidadosamente o espaço do problema, consultar especialistas no assunto e traduzir as necessidades de negócios em especificações técnicas que possam orientar o processo de desenvolvimento.

### 11.1.1 Casos de uso de negócios

Um caso de uso de negócios é uma narrativa estruturada que descreve como um sistema ou processo deve se comportar da perspectiva de um ator ou parte interessada. Ajuda a comunicar os requisitos funcionais de um sistema ou processo.

Partes de um caso de uso

Nome do caso de uso

Breve descrição

Atores

Pré-condições

Fluxo básico (cenário principal de sucesso)

Fluxos alternativos (extensões)

Pós-condições

Regras de negócios

Requisitos não funcionais

Suposições

Notas ou informações adicionais

### 11.1.2 Abordando casos de uso de negócios com IA generativa

Métricas-chave:

Economia de custos

Economia de tempo

Melhoria da qualidade

Satisfação do cliente

Ganhos de produtividade

**ABORDAGENS**

Automação de processos

Tomada de decisão aumentada

Personalização e customização

Geração de conteúdo criativo

Análise exploratória e inovação

## 11.2 Selecione um modelo de fundação

Com base nos requisitos identificados, um modelo de IA generativa apropriado é selecionado a partir de modelos pré-treinados existentes ou desenvolvido do zero. Essa decisão depende de fatores como a disponibilidade de modelos pré-treinados adequados, a complexidade do caso de uso e a disponibilidade de dados específicos do domínio para treinamento.

### 11.2.1 Critérios de seleção de modelo pré-treinados

Os modelos pré-treinados oferecem uma vantagem valiosa ao encapsular o conhecimento destilado de grandes quantidades de dados. Esses modelos podem ser ajustados em dados específicos da tarefa, potencialmente levando a uma convergência mais rápida e melhor generalização. No entanto, os modelos pré-treinados podem conter vieses indesejáveis ou não capturar totalmente as nuances do domínio de destino.

Os critérios de seleção para escolher um modelo pré-treinado dependem dos requisitos do caso de uso de negócios.

Custar

Modelos pré-treinados podem ser caros, especialmente para modelos maiores e mais complexos. O custo pode incluir taxas de licenciamento, recursos computacionais para inferência e possíveis custos de personalização ou ajuste fino. É essencial avaliar as restrições orçamentárias e pesar o custo em relação aos benefícios esperados.

Modalidade

Os modelos generativos de IA podem ser projetados para diferentes modalidades, como geração de texto, geração de imagem, geração de áudio ou geração multimodal (combinando várias modalidades). A escolha da modalidade depende do formato de saída desejado e da aplicação de destino.

Latência

Alguns aplicativos exigem geração em tempo real ou de baixa latência, e outros podem tolerar tempos de processamento mais longos. A velocidade de inferência do modelo e os recursos computacionais disponíveis devem ser avaliados para garantir uma latência aceitável para o caso de uso de destino.

Suporte multilíngue

Se o aplicativo requer a geração de conteúdo em vários idiomas, é crucial selecionar um modelo que suporte os idiomas desejados ou possa ser adaptado a novos idiomas por meio de técnicas como o aprendizado por transferência.

Tamanho do modelo

Modelos maiores geralmente têm requisitos computacionais mais altos e podem consumir mais recursos durante a inferência. No entanto, eles geralmente têm um desempenho melhor em tarefas complexas. O tamanho do modelo deve ser balanceado em relação aos recursos computacionais disponíveis e aos requisitos de desempenho.

Complexidade do modelo

Modelos mais complexos, como aqueles baseados em arquiteturas de transformadores ou modelos de linguagem grandes, podem lidar com tarefas mais avançadas, mas podem ser mais desafiadores de implantar e otimizar. Modelos mais simples podem ser preferidos para ambientes com recursos limitados ou casos de uso mais simples.

Personalização

Alguns modelos pré-treinados oferecem a capacidade de ajustá-los ou adaptá-los a domínios ou tarefas específicas. Essa personalização pode melhorar o desempenho, mas pode exigir recursos computacionais adicionais e dados rotulados.

Comprimento de entrada/saída

Os modelos generativos podem ter limitações nos comprimentos máximos de sequência de entrada ou saída que podem manipular. As aplicações que exigem geração de formato longo ou processamento de dados de entrada extensos devem considerar modelos capazes de lidar com os comprimentos de entrada/saída desejados.

Considerações de responsabilidade

É importante avaliar as implicações responsáveis do uso de modelos generativos de IA pré-treinados, como possíveis vieses, riscos de desinformação ou uso indevido. Os modelos devem ser examinados por suas fontes de dados de treinamento e possíveis impactos sociais.

Implantação e integração

A facilidade de implantação, a compatibilidade com a infraestrutura existente e a disponibilidade de ferramentas ou bibliotecas para integrar o modelo ao aplicativo de destino devem ser consideradas.

## 11.3 Melhore o desempenho

### 11.3.1 Engenharia rápida

A engenharia rápida é a maneira mais rápida de aproveitar o poder dos grandes modelos de linguagem (LLMs). Ao interagir com um LLM por meio de prompts (uma série de perguntas, declarações ou instruções), você pode ajustar o comportamento de saída do LLM com base no contexto específico da saída que deseja alcançar.

A engenharia de prompt refere-se ao processo de elaborar cuidadosamente os prompts de entrada ou instruções fornecidas ao modelo para gerar as saídas ou comportamentos desejados. O fraseado, a estrutura e o conteúdo do prompt podem influenciar significativamente a qualidade, a relevância e as características das saídas geradas. A engenharia de prompt visa otimizar os prompts para direcionar a geração do modelo na direção desejada, usando os recursos do modelo enquanto mitiga possíveis vieses ou saídas indesejáveis.

Alguns aspectos-chave da engenharia rápida incluem o seguinte:

•  **Projetar:** Elaboração de prompts claros, inequívocos e ricos em contexto que comuniquem efetivamente a tarefa ou saída desejada ao modelo

•  **Aumento:** Incorporar informações adicionais ou restrições nos prompts, como exemplos, demonstrações ou instruções específicas da tarefa, para orientar o processo de geração do modelo

•  **Afinação:** Refinar e ajustar iterativamente os prompts com base nas saídas e no desempenho do modelo, geralmente por meio de avaliação humana ou métricas automatizadas

•  **Conjunto:** Combinar vários prompts ou estratégias de geração para melhorar a qualidade geral e a robustez das saídas

•  **Mineração:** Explorar e identificar prompts eficazes por meio de técnicas como pesquisa de prompt, geração de prompt ou recuperação de prompt de grandes bibliotecas de prompt  .

### 11.3.2 Técnicas de prompt

Técnicas de engenharia rápida são estratégias usadas para orientar modelos generativos de IA. Algumas técnicas comuns de engenharia de prompt incluem o seguinte:

Aviso de disparo zero

Solicitação de poucos disparos

Solicitação da cadeia de pensamento (CoT)

Auto consistência

Árvore de pensamentos (ToT)

Geração aumentada de recuperação (RAG)

Raciocínio Automático e Uso de Ferramentas (ART)

Solicitação do ReAct

RAG

RAG é uma técnica de processamento de linguagem natural (NLP) que combina os recursos de sistemas de recuperação e modelos de linguagem generativa para produzir saídas de texto informativas e de alta qualidade.

O RAG incorpora dois componentes principais.

**Sistema de recuperação**

Esse componente recupera informações relevantes de um grande corpus de dados de texto, como uma base de dados de conhecimento, páginas da Web ou outras fontes textuais. O sistema de recuperação usa técnicas como recuperação de informações, indexação esparsa ou recuperação densa para identificar as passagens ou documentos mais relevantes para uma determinada consulta de entrada ou contexto.

**Modelo de linguagem generativa**

Esse componente é um grande modelo de linguagem pré-treinado, como GPT-3, BART ou T5, que pode gerar texto em linguagem natural. O modelo de linguagem usa a consulta de entrada ou o contexto, juntamente com as informações relevantes recuperadas. E a partir disso, gera uma saída de texto coerente e fluente que combina o conhecimento recuperado com seus próprios recursos de compreensão e geração de linguagem.

O RAG tem vários aplicativos de negócios, incluindo os seguintes:

Construindo sistemas inteligentes de resposta a perguntas

Expandir e enriquecer as bases de conhecimento existentes

Gerando conteúdo de alta qualidade

### 11.3.3 Ajuste fino

O ajuste fino é outra maneira de melhorar ainda mais o desempenho de um modelo de fundação. O ajuste fino refere-se ao processo de pegar um modelo de linguagem pré-treinado e treiná-lo em uma tarefa específica ou conjunto de dados específico de domínio. O ajuste fino permite que o modelo adapte seu conhecimento e recursos para melhor atender aos requisitos do caso de uso de negócios. Embora os FMs sejam pré-treinados por meio de aprendizado auto supervisionado e tenham capacidade inerente de entender informações, o ajuste fino do modelo base de FM pode melhorar o desempenho.

Há duas maneiras de ajustar um modelo:

1 - O ajuste fino de instruções usa exemplos de como o modelo deve responder a uma instrução específica. O ajuste imediato é um tipo de ajuste fino de instruções.

2 - O aprendizado por reforço a partir de feedback humano (RLHF) fornece dados de feedback humano, resultando em um modelo mais alinhado com as preferências humanas.

### 11.3.4 Criando um modelo de fundação do zero

No contexto do ciclo de vida do aplicativo de IA generativa, a criação de um modelo do zero envolve o treinamento de uma arquitetura de modelo completamente nova em um conjunto de dados personalizado, sem usar modelos ou pesos pré-existentes. Essa abordagem geralmente é realizada quando não há modelos pré-treinados adequados disponíveis para a tarefa ou domínio específico, ou quando os requisitos de precisão, desempenho ou personalização são excepcionalmente altos.

O processo de criação de um modelo do zero começa com a definição da arquitetura do modelo, que envolve a seleção da arquitetura de rede neural apropriada, camadas e hiper parâmetros com base no problema em questão. Esta etapa requer uma compreensão profunda dos conceitos e técnicas de aprendizado de máquina, bem como experiência de domínio para garantir que o modelo seja projetado para capturar os padrões e recursos relevantes nos dados.

Em seguida, um conjunto de dados grande e diversificado deve ser coletado, limpo e pré-processado para servir como dados de treinamento para o modelo. Esse conjunto de dados deve ser representativo dos dados do mundo real que o modelo encontrará e deve abranger uma ampla gama de cenários e variações.

Depois que o conjunto de dados é preparado, o modelo é inicializado com pesos aleatórios e treinado usando vários algoritmos de otimização em um processo iterativo. Durante o treinamento, os pesos do modelo são ajustados com base nos dados de entrada e nas saídas de destino correspondentes, com o objetivo de minimizar a função de perda e melhorar o desempenho do modelo nos dados de treinamento.

Criar um modelo do zero permite a personalização completa e a adaptação ao problema específico. Mas tem um custo significativo em termos de recursos computacionais, tempo e experiência necessários. Muitas vezes, é uma abordagem mais adequada para pesquisas ou aplicações altamente especializadas, onde os modelos pré-treinados existentes são inadequados ou indisponíveis.

Modelos pré-treinados, como modelos de linguagem grandes ou modelos de visão computacional, oferecem uma solução econômica. Eles já passaram por um treinamento extensivo em grandes quantidades de dados, portanto, exigem menos recursos computacionais e tempo para ajuste fino ou aprendizado de transferência. No entanto, esses modelos pré-treinados nem sempre atingem o nível desejado de precisão ou desempenho para tarefas ou domínios específicos.

Buscar uma abordagem mais personalizada, como treinar um modelo do zero ou ajustar fortemente um modelo pré-treinado, pode gerar maior precisão e melhor desempenho adaptado ao caso de uso específico. No entanto, essa personalização tem um custo mais alto em termos de recursos computacionais, aquisição de dados e conhecimento especializado necessário para treinamento e otimização.

Ao decidir sobre a técnica de personalização apropriada para seu aplicativo de IA generativa, as organizações devem avaliar cuidadosamente a relação custo-precisão. Eles devem equilibrar suas restrições orçamentárias, requisitos de desempenho e a disponibilidade de dados específicos de domínio de alta qualidade.

### 11.3.5 Tarefas automatizadas de várias etapas com agentes

Agentes são componentes ou entidades de software projetados para executar ações ou tarefas específicas de forma autônoma ou semiautônoma, com base em regras ou algoritmos predefinidos.

No caso do Amazon Bedrock, os agentes são usados para gerenciar e executar várias tarefas de várias etapas relacionadas ao provisionamento de infraestrutura, implantação de aplicativos e atividades operacionais.

Aqui estão alguns exemplos de tarefas que os agentes podem realizar:

•  **Coordenação de tarefas:** Os agentes coordenam a conclusão das subtarefas na ordem correta e garantem que as dependências e os pré-requisitos sejam atendidos antes de prosseguir para a próxima etapa. Eles gerenciam o fluxo de informações e dados entre diferentes subtarefas e garantem que a tarefa geral progrida sem problemas.

•  **Relatórios e registro:** Os agentes podem fornecer logs e relatórios detalhados sobre o progresso e o status de tarefas de várias etapas, incluindo métricas, dados de desempenho e informações de diagnóstico. Isso ajuda na solução de problemas, auditoria e análise do processo geral.

•  **Escalabilidade e simultaneidade:** Os agentes podem ser projetados para lidar com várias instâncias de tarefas de várias etapas simultaneamente. Isso permite a implementação paralela e melhora a taxa de transferência e a escalabilidade gerais.

•  **Integração e comunicação:** Os agentes geralmente devem se integrar a outros sistemas, serviços ou componentes para trocar dados, iniciar ações ou receber notificações. Eles podem se comunicar por meio de APIs, filas de mensagens ou outros canais de comunicação.

No caso do Amazon Bedrock, os agentes podem ser responsáveis por tarefas como provisionamento e configuração de recursos de nuvem (por exemplo, instâncias do EC2, balanceadores de carga ou bancos de dados). Eles também podem implantar aplicativos ou serviços em vários ambientes, automatizar tarefas operacionais, como backups ou dimensionamento, e monitorar a integridade geral e o desempenho da infraestrutura.

## 11.4 Avalie os resultados

### 11.4.1 Avaliação humana:

A avaliação humana envolve fazer com que os humanos interajam com o modelo de fundação e avaliem seu desempenho com base em critérios específicos. Isso pode envolver tarefas como conversas abertas, respostas a perguntas, geração de texto ou outros casos de uso específicos. Os avaliadores humanos podem fornecer feedback qualitativo sobre fatores como coerência, relevância, factualidade e qualidade geral dos resultados do modelo. Embora a avaliação humana seja frequentemente considerada o padrão-ouro, ela pode ser demorada e cara, especialmente para avaliações em larga escala.

### 11.4.2 Conjuntos de dados de referência:

Os conjuntos de dados de referência são coleções selecionadas de dados projetadas especificamente para avaliar o desempenho de modelos de linguagem ou outros sistemas de IA. Esses conjuntos de dados geralmente consistem em exemplos ou tarefas cuidadosamente selecionados que cobrem uma ampla gama de tópicos, complexidades e fenômenos linguísticos. Os modelos são avaliados executando-os nesses conjuntos de dados de referência e medindo seu desempenho usando métricas ou tarefas predefinidas.

Alguns conjuntos de dados de referência populares para tarefas de processamento de linguagem natural incluem o seguinte:

O benchmark General Language Understanding Evaluation (GLUE) é uma coleção de conjuntos de dados para avaliar tarefas de compreensão de linguagem, como classificação de texto, resposta a perguntas e inferência de linguagem natural.

O SuperGLUE é uma extensão do GLUE com tarefas mais desafiadoras e foco na compreensão da linguagem composicional.

O Stanford Question Answering Dataset (SQuAD) é um conjunto de dados para avaliar os recursos de resposta a perguntas.

O Workshop sobre Tradução Automática (WMT) é uma série de conjuntos de dados e tarefas para avaliar sistemas de tradução automática.

Esses conjuntos de dados de referência fornecem uma maneira padronizada de comparar o desempenho de diferentes modelos de fundação e acompanhar o progresso ao longo do tempo.

### 11.4.3 Métricas automatizadas:

Essas métricas normalmente medem aspectos específicos das saídas do modelo, como o seguinte:

Perplexidade (uma medida de quão bem o modelo prevê o próximo token)

Pontuação BLEU (para avaliar a tradução automática)

Pontuação F1 (para avaliar tarefas de classificação ou reconhecimento de entidade)

As métricas automatizadas podem ser úteis para iterações rápidas e ajustes finos durante o desenvolvimento do modelo, mas muitas vezes não conseguem capturar as nuances e complexidades da linguagem humana e podem não se alinhar perfeitamente com os julgamentos humanos.

### 11.4.4 Métricas relevantes

Recall-Oriented Understudy for Gisting Evaluation (ROUGE) é um conjunto de métricas usadas para avaliar sistemas de resumo automático e tradução automática. Ele mede a qualidade de um resumo ou tradução gerado, comparando-o com um ou mais resumos ou traduções de referência.

Bilingual Evaluation Understudy (BLEU) é uma métrica usada para avaliar a qualidade do texto gerado por máquina, particularmente no contexto da tradução automática. Ele mede a semelhança entre um texto gerado e uma ou mais traduções de referência, considerando precisão e brevidade.

BERTScore é uma métrica que avalia a semelhança semântica entre um texto gerado e um ou mais textos de referência. Ele usa modelos pré-treinados de Representações de Codificador Bidirecional de Transformadores (BERT) para calcular incorporações contextualizadas para os textos de entrada e, em seguida, calcula a similaridade de cosseno entre eles.

## 11.5 Implantação

Após o teste bem-sucedido, o aplicativo com recursos integrados de IA generativa é implantado no ambiente de produção. Mecanismos de monitoramento são estabelecidos para rastrear o desempenho, o uso e possíveis problemas ou vieses associados às saídas do modelo de IA generativa.

**Custar:** Pague pelos recursos que você usa sem taxas mínimas.

**Regiões:** A implantação do modelo é limitada a determinadas regiões da AWS.

**Quotas:** Certifique-se de ter os recursos de serviço adequados para sua conta da AWS.

**Segurança:**

Se o seu modelo for implantado na infraestrutura da AWS, a responsabilidade de segurança será compartilhada entre a empresa e a AWS.

# 12 Noções básicas sobre prompts

O uso de estratégias rápidas eficazes pode oferecer os seguintes benefícios:

•  Aprimore os recursos do modelo e reforce suas medidas de segurança.

•  Equipe o modelo com conhecimento específico do domínio e ferramentas externas sem modificar seus parâmetros ou passar por ajustes finos.

•  Interaja com modelos de linguagem para compreender totalmente seu potencial.

•  Obtenha saídas de maior qualidade fornecendo entradas de maior qualidade.

## 12.1 Elementos de um prompt

* Instruções: Essa é uma tarefa para o modelo de linguagem grande. Ele fornece uma descrição da tarefa ou instrução sobre como o modelo deve ser executado.
* Contexto: Esta é uma informação externa para orientar o modelo.
* Dados de entrada: Esta é a entrada para a qual você deseja uma resposta.
* Indicador de saída: Este é o tipo ou formato de saída.

## 12.2 Inspiração negativa

Às vezes, é mais fácil orientar um modelo em direção a uma saída desejada incluindo o que você não deseja incluir na saída. A solicitação negativa é usada para orientar o modelo para longe da produção de certos tipos de conteúdo ou da exibição de comportamentos específicos. Envolve fornecer ao modelo exemplos ou instruções sobre o que ele não deve gerar ou fazer.

## 12.3 Parâmetros de inferência

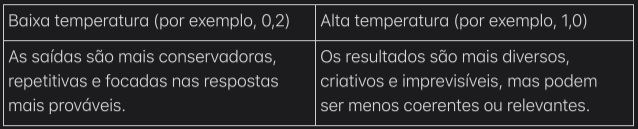
Ao interagir com FMs, muitas vezes você pode configurar parâmetros de inferência para limitar ou influenciar a resposta do modelo. Os parâmetros disponíveis variam de acordo com o modelo que você está usando. Os parâmetros de inferência se encaixam em uma variedade de categorias, sendo as mais comuns *aleatoriedades, diversidade* e *comprimento*.

### 12.3.1 Aleatoriedade e diversidade

Esta é a categoria mais comum de parâmetro de inferência. Os parâmetros de aleatoriedade e diversidade influenciam a variação nas respostas geradas, limitando os resultados a resultados mais prováveis ou alterando a forma da distribuição de probabilidade dos resultados. Três dos parâmetros mais comuns são temperatura, top k e top p

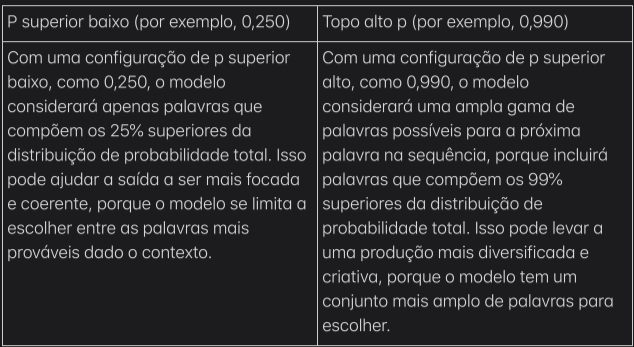
**Temperatura**

Este parâmetro controla a aleatoriedade ou criatividade da saída do modelo. Uma temperatura mais alta torna a saída mais diversificada e imprevisível, e uma temperatura mais baixa a torna mais focada e previsível. A temperatura é definida entre 0 e 1. A seguir estão exemplos de diferentes configurações de temperatura.



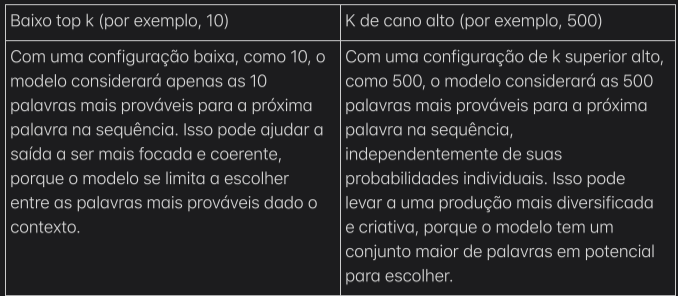
**Topo p**

Top p é uma configuração que controla a diversidade do texto, limitando o número de palavras que o modelo pode escolher com base em suas probabilidades. O topo p também é definido em uma escala de 0 a 1. A seguir estão exemplos de diferentes configurações de p superior.



**Topo k**

O top k limita o número de palavras ao top k palavras mais prováveis, independentemente de suas probabilidades percentuais. Por exemplo, se o top k for definido como 50, o modelo considerará apenas as 50 palavras mais prováveis para a próxima palavra na sequência, mesmo que essas 50 palavras representem apenas uma pequena parte da distribuição de probabilidade total.



### 12.3.2 Comprimento

A categoria de parâmetro de inferência de comprimento refere-se às configurações que controlam o comprimento máximo da saída gerada e especificam as sequências de parada que sinalizam o fim do processo de geração.

**Comprimento máximo**

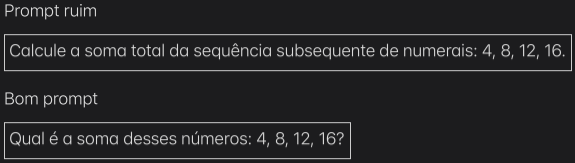
A configuração de comprimento máximo determina o número máximo de tokens que o modelo pode gerar durante o processo de inferência. Esse parâmetro ajuda a evitar que o modelo gere saída excessiva ou infinita, o que pode levar ao esgotamento de recursos ou a um comportamento indesejável. O valor apropriado para essa configuração depende da tarefa específica e do comprimento de saída desejado. Por exemplo, em tarefas de geração de linguagem natural, como resumo ou tradução de texto, o comprimento máximo pode ser definido com base no comprimento típico do texto de destino. Em tarefas de geração aberta, como escrita criativa ou sistemas de diálogo, um comprimento máximo mais alto pode ser desejável para permitir saídas mais estendidas.

**Sequências de parada**

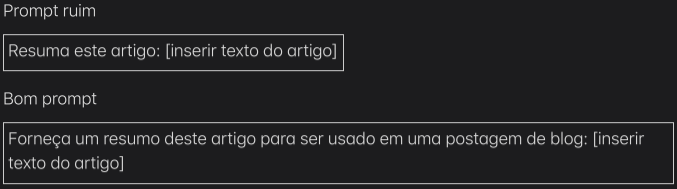
As sequências de parada são tokens especiais ou sequências de tokens que sinalizam ao modelo para parar de gerar mais saídas. Quando o modelo encontrar uma sequência de parada durante o processo de inferência, ele encerrará a geração independentemente da configuração de comprimento máximo. As sequências de parada são particularmente úteis em tarefas em que o comprimento de saída desejado é variável ou difícil de prever com antecedência. Por exemplo, em sistemas de inteligência artificial (IA) conversacional, a sequência de parada pode ser um token de fim de conversa ou uma frase específica que indica o fim da resposta.   As sequências de parada podem ser predefinidas ou geradas dinamicamente com base na entrada ou na própria saída gerada. Em alguns casos, várias sequências de parada podem ser especificadas, permitindo que o modelo pare a geração ao encontrar qualquer uma das sequências definidas.

### 12.3.3-Práticas recomendadas para solicitação

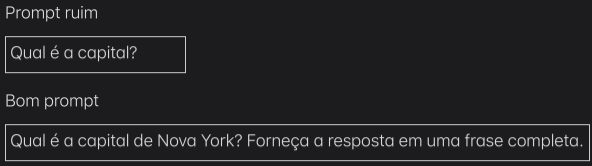
**Seja claro e conciso.**



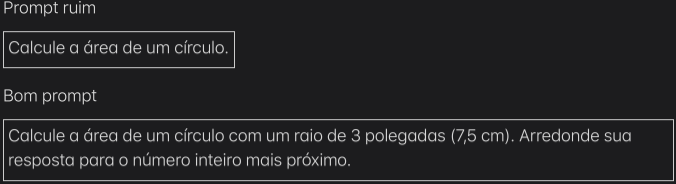
**Inclua contexto, se necessário.**



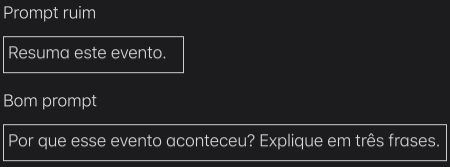
**Use diretivas para o tipo de resposta apropriado.**



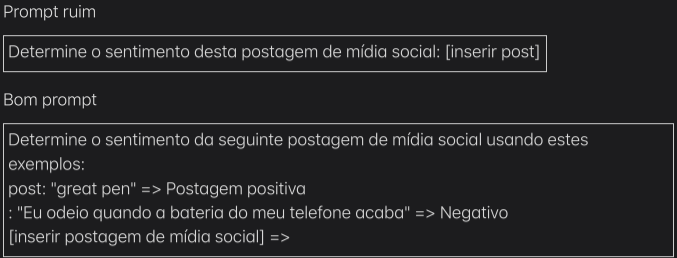
**Considere a saída no prompt.**



**Inicie os prompts com um interrogatório.**



**Forneça um exemplo de resposta.**



**Divida tarefas complexas.**

Os modelos de fundação podem ficar confusos quando solicitados a executar tarefas complexas. Divida tarefas complexas usando as seguintes técnicas:

Divida a tarefa em várias subtarefas. Se você não conseguir obter resultados confiáveis, tente dividir a tarefa em vários prompts.

Pergunte ao modelo se ele entendeu sua instrução. Forneça esclarecimentos com base na resposta do modelo.

Se você não sabe como dividir a tarefa em subtarefas, peça ao modelo para pensar passo a passo. Por exemplo, você pode solicitar que o modelo divida a tarefa em subtarefas, aborde o problema sistematicamente ou raciocine sobre o problema uma etapa de cada vez.

**Experimente e seja criativo.**

Tente prompts diferentes para otimizar as respostas do modelo. Determine quais prompts alcançam resultados efetivos e quais prompts obtêm resultados imprecisos. Ajuste suas instruções de acordo. Prompts novos e instigantes podem levar a resultados inovadores.

**Use modelos de prompt.**

Os modelos de prompt são estruturas ou formatos predefinidos que podem ser usados para fornecer entradas consistentes aos FMs. Eles ajudam a garantir que os prompts sejam formulados de uma maneira que seja facilmente compreendida pelo modelo e possam levar a resultados mais confiáveis e de maior qualidade. Os modelos de prompt geralmente incluem instruções, contexto, exemplos e espaços reservados para informações relevantes para a tarefa em questão.

Os modelos de prompt podem ajudar a simplificar o processo de interação com modelos, facilitando a integração deles em vários aplicativos e fluxos de trabalho.

## 12.4 Técnicas de engenharia rápida

### 12.4.1 Aviso de disparo zero

A solicitação zero-shot é uma técnica em que um usuário apresenta uma tarefa a um modelo generativo sem fornecer exemplos ou treinamento explícito para essa tarefa específica. Nessa abordagem, o usuário conta com o conhecimento geral e os recursos do modelo para entender e executar a tarefa sem qualquer exposição prévia, ou *fotos*, de tarefas semelhantes. Notavelmente, os FMs modernos demonstraram um desempenho impressionante de tiro zero, lidando efetivamente com tarefas para as quais não foram explicitamente treinados.

Para otimizar a solicitação de disparo zero, considere as seguintes dicas:

•  Quanto maior e mais capaz o FM, maior a probabilidade de obter resultados efetivos de prompts de disparo zero.

•  O ajuste de instruções, um processo de ajuste fino de modelos para melhor alinhamento com as preferências humanas, pode aprimorar os recursos de aprendizado zero-shot. Uma abordagem para dimensionar o ajuste de instruções é por meio do aprendizado por reforço a partir de feedback humano (RLHF), onde o modelo é treinado iterativamente com base em avaliações humanas de suas saídas.

### 12.4.2 Solicitação de poucos disparos

A solicitação de poucos disparos é uma técnica que envolve fornecer um modelo de linguagem com exemplos contextuais para orientar sua compreensão e saída esperada para uma tarefa específica. Nessa abordagem, você complementa o prompt com entradas de amostra e suas saídas desejadas correspondentes, dando ao modelo *algumas fotos* ou demonstrações para condicioná-lo para a tarefa solicitada. Embora a solicitação de poucos disparos forneça um modelo com vários exemplos, você também pode usar o prompt de disparo único ou único fornecendo apenas um exemplo.

Ao empregar uma técnica de solicitação de poucos tiros, considere as seguintes dicas:

•  Certifique-se de selecionar exemplos que representem a tarefa que você deseja que o modelo execute e abranjam uma gama diversificada de entradas e saídas. Além disso, procure usar exemplos claros e concisos que demonstrem com precisão o comportamento desejado.

•  Experimente o número de exemplos. O número ideal de exemplos a serem incluídos em um prompt de poucos disparos pode variar dependendo da tarefa, do modelo e da complexidade dos próprios exemplos. Geralmente, fornecer mais exemplos pode ajudar o modelo a entender melhor a tarefa. Mas muitos exemplos podem introduzir ruído ou confusão.

### 12.4.3 Sugestão de cadeia de pensamento

A solicitação da cadeia de pensamento (CoT) é uma técnica que divide tarefas de raciocínio intrincadas em etapas intermediárias menores. Essa abordagem pode ser empregada usando técnicas de solicitação de tiro zero ou de poucos tiros. Os prompts do CoT são adaptados a tipos de problemas específicos. Para iniciar o processo de raciocínio da cadeia de pensamento em um modelo de aprendizado de máquina, você pode usar a frase "Pense passo a passo". Recomenda-se usar a solicitação de CoT quando a tarefa exigir várias etapas ou uma série de raciocínios lógicos.

## 12.5 Usos indevidos e riscos imediatos

### 12.5.1 Envenenamento, sequestro e injeção imediata

**Envenenamento**

Envenenamento refere-se à introdução intencional de dados mal-intencionados ou tendenciosos no conjunto de dados de treinamento de um modelo. Isso pode fazer com que o modelo produza resultados tendenciosos, ofensivos ou prejudiciais, intencionalmente ou não.

**Sequestro e injeção imediata**

O sequestro e a injeção de prompt referem-se à técnica de influenciar as saídas de modelos generativos incorporando instruções específicas nos próprios prompts.

Por exemplo, um ator mal-intencionado pode criar prompts para um modelo de geração de texto que contenha conteúdo prejudicial, antiético ou tendencioso. Ao fazer isso, o modelo pode gerar saídas de texto prejudiciais, antiéticas ou tendenciosas semelhantes. Esses textos gerados podem ser armados pelo ator para criar notícias falsas, propagar desinformação ou produzir outros tipos de conteúdo malicioso em escala.

O objetivo é sequestrar o comportamento do modelo e fazer com que ele produza saídas que se alinhem com as intenções do invasor, como gerar desinformação ou executar código malicioso.

### 12.5.2 Exposição e vazamento imediato

**Exposição**

A exposição refere-se ao risco de expor informações sensíveis ou confidenciais a um modelo generativo durante o treinamento ou inferência. Um FM pode revelar inadvertidamente esses dados confidenciais de seu corpus de treinamento, levando a possíveis vazamentos de dados ou violações de privacidade.

**Vazamento imediato**

O vazamento de prompt refere-se à divulgação ou vazamento não intencional dos prompts ou entradas (independentemente de serem dados protegidos ou não) usados em um modelo. O vazamento imediato não expõe necessariamente dados protegidos. Mas pode expor outros dados usados pelo modelo, o que pode revelar informações de como o modelo funciona e isso pode ser usado contra ele.

### 12.5.3 Jailbreak

Jailbreak refere-se à prática de modificar ou contornar as restrições e medidas de segurança implementadas em um modelo generativo ou assistente de IA para obter acesso ou funcionalidade não autorizados.

Quando um modelo de IA é desenvolvido, ele normalmente é treinado com certas restrições éticas e de segurança para evitar o uso indevido ou saídas prejudiciais. Essas restrições podem incluir a filtragem de conteúdo explícito ou ofensivo, a restrição do acesso a informações confidenciais ou a limitação da capacidade de realizar determinadas ações ou comandos.

As tentativas de jailbreak envolvem a criação de prompts cuidadosamente construídos ou sequências de entrada que visam contornar ou explorar vulnerabilidades nos mecanismos ou restrições de filtragem do sistema de IA. O objetivo é "sair" das limitações do modelo pretendido.

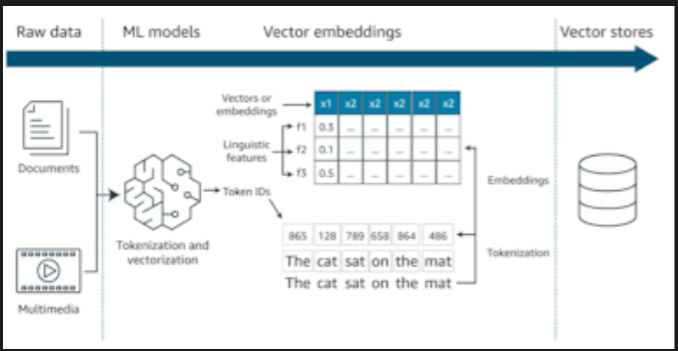
# 13 Geração Aumentada por Recuperação (RAG)

## 13.1 Conjuntos de dados corporativos

Embora os LLMs possam gerar texto, imagem, áudio e muito mais semelhantes aos humanos a partir de prompts, esse recurso pode não atender às necessidades específicas das empresas. Aplicativos corporativos personalizados exigem que esses modelos processem dados relevantes de conjuntos de dados corporativos.   As empresas coletam grandes quantidades de dados internos, incluindo documentos, apresentações, manuais do usuário, relatórios e resumos de transações, todos desconhecidos para o FM. Quando esses modelos ingerem e usam fontes de dados corporativos, eles adquirem conhecimento específico do domínio, permitindo que produzam resultados personalizados e altamente relevantes que atendam às necessidades da empresa.   Para fornecer os dados corporativos relevantes como contexto adicional ao modelo de linguagem, juntamente com o prompt, essa adição ajuda o modelo a fornecer resultados mais precisos. Determinar o contexto correto envolve a pesquisa de conjuntos de dados corporativos usando o texto de prompt. As incorporações vetoriais desempenham um papel crucial nesse processo.

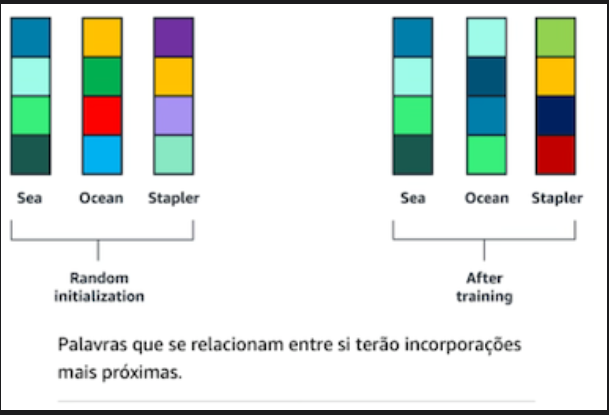
## 13.2 Incorporações vetoriais

A incorporação é o processo pelo qual texto, imagens e áudio recebem representação numérica em um espaço vetorial. A inserção geralmente é executada por um modelo de aprendizado de máquina (ML). O diagrama a seguir fornece mais detalhes sobre a inserção.



Os conjuntos de dados corporativos, como documentos, imagens e áudio, são passados para modelos de ML como tokens e são vetorizados. Esses vetores em um espaço n-dimensional, juntamente com os metadados sobre eles, são armazenados em bancos de dados vetoriais criados especificamente para recuperação mais rápida.

Aqui está um exemplo de duas palavras: mar e oceano. Eles são inicializados aleatoriamente e suas primeiras incorporações são diversas. À medida que o treinamento avança, suas incorporações se tornam mais semelhantes porque geralmente aparecem próximas umas das outras e em contextos semelhantes. Para a finalidade deste exemplo, as incorporações próximas são representadas por cores na mesma paleta. Portanto, a palavra mar e oceano usam cores semelhantes. No entanto, o grampeador tem uma incorporação completamente diferente, por isso usa um conjunto separado de cores.



## 13.3 Armazenando vetores

A função principal dos bancos de dados vetoriais é armazenar de forma compacta bilhões de vetores de alta dimensão que representam palavras e entidades. Os bancos de dados vetoriais fornecem pesquisas de similaridade ultrarrápidas nesses bilhões de vetores em tempo real.

Os algoritmos mais comuns usados para realizar a pesquisa de similaridade são k-vizinhos mais próximos (k-NN) ou similaridade de cosseno.

A Amazon Web Services (AWS) oferece as seguintes opções viáveis de banco de dados vetorial:

Amazon OpenSearch Service (provisionado)

Amazon OpenSearch sem servidor

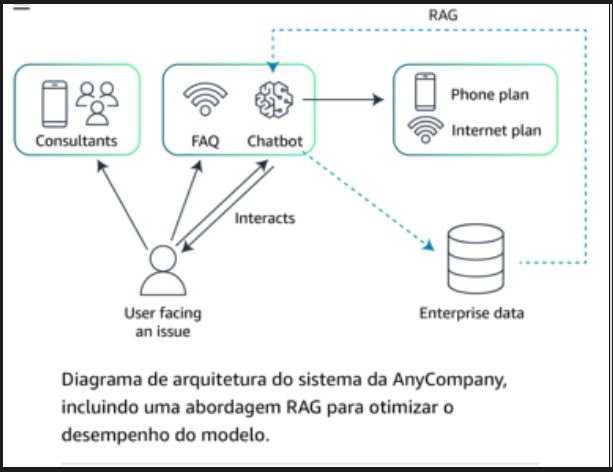
pgvector no Amazon Relational Database Service (Amazon RDS) para PostgreSQL

pgvector na edição compatível com PostgreSQL do Amazon Aurora

Amazon Kendra

## 13.4 RAG no contexto do caso de negócios da AnyCompany

A seguir está uma versão atualizada do diagrama de arquitetura da AnyCompany, incluindo o sistema RAG. O chatbot agora é capaz de consultar um banco de dados contendo dados corporativos e usá-lo para fornecer respostas mais precisas e contextuais aos usuários.



# 14 Agentes

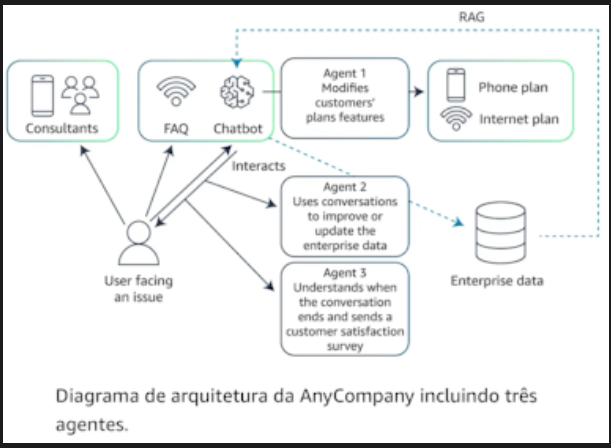
## 14.1 Principais funções dos agentes

Os agentes podem desempenhar diferentes funções em um aplicativo de IA generativa, como as seguintes:

**Operações intermediárias:** Os agentes podem atuar como intermediários, facilitando a comunicação entre o modelo de IA generativa e vários sistemas de back-end. O modelo de IA generativa lida com a compreensão da linguagem e a geração de respostas. Os vários sistemas de back-end incluem itens como bancos de dados, plataformas de CRM ou ferramentas de gerenciamento de serviços.

**Inicialização de ações:** os agentes podem ser usados para executar uma ampla variedade de tarefas. Essas tarefas podem incluir ajustar as configurações do serviço, processar transações, recuperar documentos e muito mais. Essas ações são baseadas nas necessidades específicas dos usuários compreendidas pelo modelo de IA generativa.

**Integração de feedback:** os agentes também podem contribuir para o processo de aprendizado do sistema de IA coletando dados sobre os resultados de suas ações. Esse feedback ajuda a refinar o modelo de IA, aumentando sua precisão e eficácia em interações futuras.



# 15 Ajuste fino

O ajuste fino é fundamental porque ajuda a fazer o seguinte:

**Aumentar a especificidade**: Adapte as respostas ou previsões do modelo às nuances de um domínio ou tarefa específica que não foram adequadamente abordadas no treinamento inicial.

**Melhorar a precisão**: Melhore o desempenho do modelo em tarefas especializadas treinando em dados específicos do domínio, reduzindo assim os erros que ocorrem devido à natureza genérica do treinamento básico.

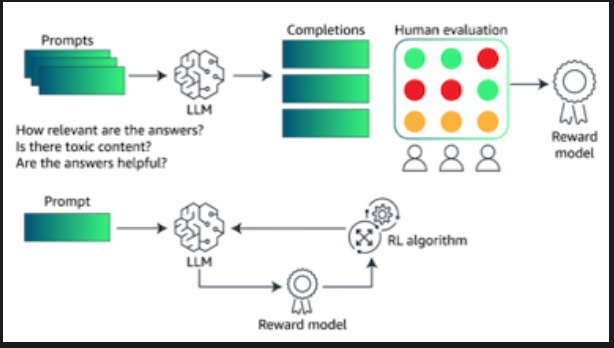
**Reduza os vieses**: Aborde e mitigue quaisquer vieses inerentes aos dados de treinamento inicial, tornando o modelo mais justo e apropriado para diferentes aplicações.

**Aumente a eficiência**: simplifique as operações do modelo em contextos específicos, reduzindo potencialmente os requisitos computacionais e acelerando os tempos de resposta.

## 15.1 As diferentes abordagens de ajuste fino

**Noajuste de estrutura:** essa abordagem envolve treinar novamente o modelo em um novo conjunto de dados que consiste em prompts seguidos pelas saídas desejadas. Isso é estruturado de forma que o modelo aprenda a seguir melhor instruções específicas. Esse método é particularmente útil para melhorar a capacidade do modelo de entender e executar comandos do usuário com precisão, tornando-o altamente eficaz para aplicativos interativos, como assistentes virtuais e chatbots.

**Aprendizado por reforço a partir de feedback humano (RLHF):** Essa abordagem é uma técnica de ajuste fino em que o modelo é inicialmente treinado usando aprendizado supervisionado para prever respostas semelhantes às humanas. Em seguida, ele é refinado por meio de um processo de aprendizado por reforço, onde um modelo de recompensa construído a partir do feedback humano orienta o modelo para gerar resultados mais preferíveis. Este método é eficaz no alinhamento dos resultados do modelo com valores e preferências humanas, aumentando assim sua utilidade prática em aplicações sensíveis.



RLHF refere-se à melhoria do modelo aprendendo com o feedback, como classificações, preferências, demonstrações, utilidade ou toxicidade, fornecidos por humanos. O RLHF é usado durante a fase de pré-treinamento do modelo, mas também pode ser usado para ajustar o modelo.

**Adaptando modelos para domínios específicos:** Essa abordagem envolve o ajuste fino do modelo em um corpus de texto ou dados específicos de um determinado setor ou setor. Um exemplo disso seriam documentos legais para uma IA legal ou registros médicos para uma IA de saúde. Essa especificidade permite que o modelo funcione com maior grau de relevância e precisão em tarefas específicas de domínio, fornecendo respostas mais úteis e sensíveis ao contexto.

**Transferência de aprendizagem:** Essa abordagem é um método em que um modelo desenvolvido para uma tarefa é reutilizado como ponto de partida para um modelo em uma segunda tarefa. Para modelos fundamentais, isso geralmente significa pegar um modelo que foi treinado em um conjunto de dados vasto e geral e, em seguida, ajustá-lo em um conjunto de dados menor e específico. Este método é altamente eficiente no uso de recursos e conhecimentos aprendidos da fase de treinamento geral e aplicá-los a um escopo mais restrito com menos treinamento adicional necessário.

**Pré-treinamento contínuo:** Essa abordagem envolve estender a fase de treinamento de um modelo pré-treinado, alimentando-o continuamente com dados novos e emergentes. Essa abordagem é usada para manter o modelo atualizado com as informações, vocabulário, tendências ou descobertas de pesquisa mais recentes, garantindo que seus resultados permaneçam relevantes e precisos ao longo do tempo.

## 15.2 Preparando os dados para a etapa de ajuste fino

Durante a fase inicial de treinamento, um modelo fundamental é treinado em um conjunto de dados vasto e diversificado. Esse conjunto de dados normalmente abrange uma ampla gama de tópicos para desenvolver uma ampla compreensão e recursos gerais. Os objetivos durante esta fase são os seguintes:

**Cobertura extensiva**: garantir que o conjunto de dados cubra um amplo espectro de conhecimento para dar ao modelo uma compreensão básica robusta

**Diversidade**: Incluindo tipos variados de dados de várias fontes para equipar o modelo com a capacidade de lidar com uma ampla gama de tarefas

**Generalização**: Concentrar-se na construção de um modelo que possa generalizar em diferentes tarefas e domínios sem adaptação específica

A preparação de dados para esta fase envolve a coleta do máximo de dados possível. Os dados geralmente são de fontes disponíveis publicamente, conjuntos de dados selecionados e, às vezes, dados proprietários, dependendo do uso pretendido do modelo. Os dados precisam de limpeza completa e possivelmente anonimização para garantir a privacidade e a conformidade com os regulamentos.

## 15.3 Preparação de dados para ajuste fino

O ajuste fino, por outro lado, é um processo mais direcionado em que um modelo pré-treinado é adaptado para ter um bom desempenho em uma tarefa específica ou em um domínio específico. A preparação de dados para ajuste fino é diferente do treinamento inicial devido aos seguintes motivos:

**Especificidade**: O conjunto de dados para ajuste fino é muito mais focado, contendo exemplos que são diretamente relevantes para as tarefas ou problemas específicos que o modelo precisa resolver.

**Alta relevância**: os dados devem ser altamente relevantes para os resultados desejados. Os exemplos incluem documentos legais para uma IA jurídica ou interações de atendimento ao cliente para uma IA de suporte ao cliente.

**Qualidade sobre quantidade**: embora o treinamento inicial exija grandes quantidades de dados, o ajuste fino geralmente pode alcançar melhorias significativas com conjuntos de dados muito menores, mas bem selecionados.

## 15.4 Principais etapas no ajuste fino da preparação de dados

A lista a seguir percorre as principais etapas no ajuste fino da preparação de dados:

**Curadoria de dados**: Embora seja uma continuação, isso envolve um processo de seleção mais rigoroso para garantir que cada dado seja altamente relevante. Essa etapa também garante que os dados contribuam para o aprendizado do modelo no contexto específico.

**Rotulagem**: No ajuste fino, a precisão e a relevância dos rótulos são primordiais. Eles orientam os ajustes do modelo para se especializar no domínio de destino.

**Governança e conformidade**: considerando que o ajuste fino geralmente usa dados mais especializados, é fundamental garantir a governança de dados e a conformidade com os regulamentos específicos do setor.

**Verificação de representatividade e viés**: é essencial garantir que o conjunto de dados de ajuste fino não introduza ou perpetue vieses que possam distorcer o desempenho do modelo de maneiras indesejáveis.

**Integração de feedback**: Para métodos como RLHF, incorporar feedback de usuários ou especialistas diretamente no processo de treinamento é crucial. Isso é mais sutil e interativo do que a fase inicial de treinamento.

## 15.5 Avaliação do modelo

Três métricas comumente usadas para esse propósito são Recall-Oriented Understudy for Gisting Evaluation (ROUGE), Bilingual Evaluation Understudy(BLEU) e BERTScore.

**VERMELHO**

ROUGE é um conjunto de métricas usadas para avaliar o resumo automático de textos, além da qualidade da tradução automática em PNL. A ideia principal por trás do ROUGE é contar o número de unidades sobrepostas. Isso inclui palavras, N-gramas ou fragmentos de frases entre a saída gerada por computador e um conjunto de textos de referência (criados por humanos).

Veja a seguir duas maneiras de usar a métrica ROUGE:

**ROUGE-N:** Essa métrica mede a sobreposição de n-gramas entre o texto gerado e o texto de referência. Por exemplo, ROUGE-1 refere-se à sobreposição de unigramas, ROUGE-2 refere-se a bigramas e assim por diante. Essa métrica avalia principalmente a fluência do texto e até que ponto ele inclui ideias-chave da referência.

**ROUGE-L:** Essa métrica usa a subsequência comum mais longa entre o texto gerado e os textos de referência. É particularmente bom para avaliar a coerência e a ordem da narrativa nos resultados.

ROUGE é amplamente utilizado porque não é complexo. É interpretável e se correlaciona razoavelmente bem com o julgamento humano, especialmente ao avaliar o aspecto de recordação dos resumos. As avaliações avaliam o quanto das informações importantes nos textos de origem são capturadas pelos resumos gerados.

**AZUL**

BLEU é uma métrica usada para avaliar a qualidade do texto que foi traduzido automaticamente de um idioma natural para outro. A qualidade é calculada comparando o texto gerado por máquina com uma ou mais traduções humanas de alta qualidade. O BLEU mede a precisão dos N-gramas no texto gerado por máquina que aparece nos textos de referência e aplica uma penalidade por traduções excessivamente curtas (penalidade de brevidade).

Ao contrário do ROUGE, que se concentra no recall, o BLEU é fundamentalmente uma métrica de precisão. Ele verifica quantas palavras ou frases na tradução automática aparecem nas traduções de referência. O BLEU avalia a qualidade no nível da frase, normalmente usando uma combinação de unigramas, bigramas, trigramas e quadrigramas. Uma penalidade de brevidade desencoraja traduções excessivamente concisas que podem influenciar a pontuação de precisão.

O BLEU é popular no campo da tradução automática por sua facilidade de uso e eficácia em larga escala. No entanto, apresenta limitações na avaliação da fluência e gramaticalidade da saída.

**O BERTScore**

O BERTScore usa as incorporações contextuais pré-treinadas de modelos como o BERT para avaliar a qualidade das tarefas de geração de texto. O BERTScore calcula a semelhança de cosseno entre as incorporações contextuais de palavras no candidato e nos textos de referência. Isso é diferente das métricas tradicionais que dependem de correspondências exatas de N-gramas ou palavras.

Como o BERTScore avalia a semelhança semântica em vez de confiar em correspondências lexicais exatas, ele é capaz de capturar o significado de uma maneira mais sutil. O BERTScore é menos propenso a algumas das armadilhas do BLEU e do ROUGE. Um exemplo disso é sua sensibilidade a paráfrases menores ou uso de sinônimos que não afetam o significado geral transmitido pelo texto.

O BERTScore é cada vez mais usado ao lado de métricas tradicionais como BLEU e ROUGE para uma avaliação mais abrangente dos modelos de geração de linguagem. Isso é especialmente verdadeiro nos casos em que é importante capturar o significado semântico mais profundo do texto.

# 16 Compliance para Sistemas de IA

**Complexidade e opacidade**

Os sistemas de IA, especialmente grandes modelos de linguagem (LLMs) e IA generativa, podem ser altamente complexos com processos de tomada de decisão opacos. Isso torna difícil auditar e entender como eles chegam aos resultados, o que é crucial para a conformidade.

**Dinamismo e adaptabilidade**

Os sistemas de IA geralmente são dinâmicos e podem se adaptar e mudar com o tempo, mesmo após a implantação. Isso dificulta a aplicação de padrões, estruturas e mandatos estáticos.

**Capacidades emergentes**

À medida que os sistemas de IA se tornam mais avançados, eles podem exibir recursos inesperados ou emergentes que não foram previstos durante o processo regulatório. Isso requer monitoramento contínuo e adaptabilidade.

**Riscos únicos**

A IA apresenta novos riscos, como viés algorítmico, violações de privacidade, desinformação e automação alimentada por IA que desloca trabalhadores humanos. Os requisitos tradicionais podem não os abordar adequadamente.

**Responsabilidade do algoritmo**

A responsabilidade do algoritmo refere-se à ideia de que os algoritmos, especialmente aqueles usados em sistemas de IA, devem ser transparentes, explicáveis e sujeitos a medidas de supervisão e responsabilidade.

## 16.1 Cargas de trabalho regulamentadas

***Regulado*** é um termo comum usado para indicar que uma carga de trabalho pode precisar de consideração especial, devido a alguma forma de conformidade que deve ser alcançada.

Esse termo geralmente se refere a clientes que trabalham em setores com altos graus de requisitos de conformidade regulatória ou altas demandas industriais.

Alguns exemplos de indústrias são os seguintes:

* Serviços financeiros
* Saúde
* Aeroespaço

Você está operando em um contexto regulamentado quando deve cumprir estruturas regulatórias como HIPAA, GDPR, PCI DSS e outras.

**Perguntas a serem feitas**

* Você precisa auditar essa carga de trabalho?
* Você precisa arquivar esses dados por um período de tempo?
* As previsões criadas pelo meu modelo constituirão um registro ou outro item de dados especial?

Algum dos sistemas dos quais você obtém os dados contém classificações de dados restritas pela governança da sua organização, mas não por uma estrutura regulatória? Por exemplo, endereços de clientes.

# 17 Governança de Dados com IA

## 17.1 Qualidade e integridade dos dados

Para garantir a qualidade e integridade de seus dados, siga estas etapas:

* Estabeleça padrões e processos de qualidade de dados para garantir a precisão, integridade e consistência dos dados usados para modelos de IA e IA generativa.
* Implemente técnicas de validação e limpeza de dados para identificar e resolver anomalias e inconsistências de dados.
* Mantenha a linhagem e a proveniência dos dados para entender a origem, a transformação e o uso dos dados.

## 17.2 Proteção de dados e privacidade

Para garantir a proteção e privacidade dos dados, implemente as seguintes etapas:

* Desenvolver e aplicar políticas de privacidade de dados que protejam informações confidenciais ou pessoais.
* Implemente controles de acesso, criptografia e outras medidas de segurança para proteger os dados contra acesso não autorizado ou uso indevido.
* Estabeleça procedimentos de resposta a violações de dados e gerenciamento de incidentes para mitigar o impacto de quaisquer incidentes de segurança de dados.

## 17.3 Gerenciamento do ciclo de vida dos dados

* Classifique e catalogue ativos de dados com base em sua sensibilidade, valor e criticidade para a organização.
* Implemente políticas de retenção e descarte de dados para garantir o armazenamento, o arquivamento e a exclusão apropriados de dados.
* Desenvolva estratégias de backup e recuperação de dados para garantir a continuidade dos negócios e a resiliência dos dados.

## 17.4 IA responsável

* Estabelecer estruturas e diretrizes responsáveis para o desenvolvimento e implantação de modelos de IA e IA generativa, abordando questões como preconceito, justiça, transparência e responsabilidade.
* Implemente processos para monitorar e auditar modelos de IA e IA generativa em busca de possíveis vieses, problemas de justiça e consequências não intencionais.
* Educar e treinar equipes de desenvolvimento de IA em práticas responsáveis de IA.

## 17.5 Estruturas e funções de governança

* Estabelecer um conselho ou comitê de governança de dados para supervisionar o desenvolvimento e a implementação de políticas e práticas de governança de dados.
* Defina funções e responsabilidades claras para administradores de dados, proprietários de dados e guardiões de dados para garantir o gerenciamento de dados responsável.
* Fornecer treinamento e suporte a profissionais de inteligência artificial e aprendizado de máquina (IA/ML) e usuários de dados sobre as melhores práticas de governança de dados.

## 17.6 Compartilhamento de dados e colaboração

* Desenvolver acordos e protocolos de compartilhamento de dados para facilitar a troca segura e controlada de dados entre as fronteiras organizacionais.
* Implemente técnicas de virtualização ou federação de dados para permitir o acesso a fontes de dados distribuídas sem comprometer a propriedade ou o controle dos dados.
* Promova uma cultura de tomada de decisão orientada por dados e governança colaborativa de dados em toda a organização.

## 17.7 Conceitos em governança de dados

**Ciclos de vida dos dados**

Os ciclos de vida dos dados referem-se ao gerenciamento de dados durante toda a sua vida útil, desde a criação até o eventual descarte ou arquivamento. No contexto de cargas de trabalho de IA, o ciclo de vida dos dados abrange os seguintes estágios no ciclo de vida dos dados usados para treinar e implantar modelos de IA:

* Coleção
* Processamento
* Armazenamento
* Consumo
* Descarte ou arquivamento
* Registro de dados

O registro de dados envolve o registro sistemático de dados relacionados ao processamento de uma carga de trabalho de IA. Isso pode incluir o seguinte:

* Rastreando entradas
* Saídas de rastreamento
* Métricas de desempenho do modelo
* Eventos do sistema

O registro de dados eficaz é necessário para depurar, monitorar e entender o comportamento dos sistemas de IA.

**Residência de dados**

A residência de dados refere-se ao local físico onde os dados são armazenados e processados. No contexto de cargas de trabalho de IA, as considerações de residência de dados podem incluir o seguinte:

* Conformidade com os regulamentos de privacidade de dados
* Requisitos de soberania de dados
* Proximidade dos dados com os recursos de computação usados para treinamento e inferência

**Monitoramento de dados**

O monitoramento de dados envolve a observação e análise contínuas de dados usados em cargas de trabalho de IA. Isso pode incluir o seguinte:

* Monitorando a qualidade dos dados
* Identificação de anomalias (Uma anomalia é um ponto de dados inesperado que se desvia significativamente da norma.)
* Rastreamento de desvio de dados (O desvio de dados é observado quando a distribuição dos dados de entrada muda ao longo do tempo.)

O monitoramento também ajuda a garantir que os dados usados para treinamento e inferência permaneçam relevantes e representativos.

**Análise de dados**

Os métodos de análise de dados são usados para entender as características, padrões e relacionamentos dentro dos dados usados para cargas de trabalho de IA.

Esses métodos ajudam a obter insights sobre os dados. Eles incluem o seguinte:

* Análise estatística
* Visualização de dados
* Análise exploratória de dados (EDA): EDA é uma tarefa para descobrir padrões, entender relacionamentos, validar suposições e identificar anomalias nos dados.

**Retenção de dados**

As políticas de retenção de dados definem por quanto tempo os dados devem ser mantidos para cargas de trabalho de IA. Isso pode ser influenciado por fatores como os seguintes:

* Requisitos regulamentares
* Atualizando dados históricos para retreinamento de modelo
* Custo do armazenamento de dados

## 17.8 Estratégias de governança

**Políticas**

* As políticas devem abordar áreas como gerenciamento de dados, treinamento de modelos, validação de resultados, segurança e supervisão humana.
* As políticas também devem abranger aspectos como propriedade intelectual, mitigação de preconceitos e proteção da privacidade.
* Certifique-se de que essas políticas sejam revisadas e atualizadas regularmente para acompanhar a evolução da tecnologia e dos requisitos regulatórios.

**Cadência de revisão**

* O processo de revisão pode envolver uma combinação de revisões técnicas, legais e responsáveis de IA em diferentes estágios do ciclo de vida de desenvolvimento e implantação.
* Estabeleça um cronograma claro para essas revisões, como mensal, trimestral ou semestral, dependendo da complexidade e do perfil de risco das soluções.
* Certifique-se de que o processo de revisão inclua diversas perspectivas das partes interessadas, incluindo especialistas no assunto, equipes jurídicas e de conformidade e usuários finais.

**Estratégias de revisão**

* As revisões técnicas devem se concentrar no desempenho do modelo, na qualidade dos dados e na robustez dos algoritmos subjacentes.
* As revisões não técnicas devem avaliar o alinhamento das soluções com as políticas organizacionais, princípios de IA responsável e requisitos regulatórios.
* Incorpore procedimentos de teste e validação para validar os resultados das soluções de IA generativa antes da implantação.
* Estabeleça estruturas claras de tomada de decisão para determinar quando e como intervir ou modificar as soluções com base nos resultados da revisão.

**Padrões de transparência**

* Inclua a publicação de informações sobre os modelos de IA, seus dados de treinamento e as principais decisões tomadas durante o processo de desenvolvimento.
* Forneça documentação clara e acessível sobre os recursos, limitações e casos de uso pretendidos das soluções de IA generativa.
* Estabeleça canais para que as partes interessadas, incluindo usuários finais, forneçam feedback e levantem preocupações sobre as soluções.

**Requisitos de treinamento da equipe**

* Forneça treinamento abrangente sobre mitigação de vieses e práticas responsáveis de IA.
* Incentive a colaboração multifuncional e o compartilhamento de conhecimento para promover uma cultura de desenvolvimento responsável de IA.
* Considere a implementação de programas contínuos de treinamento e certificação para manter os membros da equipe atualizados com os últimos avanços e mudanças regulatórias.

## 17.9 Monitorando um sistema de IA

**Métricas de desempenho**

* Precisão do modelo: A proporção de previsões corretas feitas pelo modelo
* Precisão: A proporção de previsões positivas verdadeiras em relação ao número total de previsões positivas feitas pelo modelo
* Lembrar: A proporção de previsões positivas verdadeiras em relação ao número total de instâncias positivas reais nos dados
* Pontuação F1: A média harmônica de precisão e recall, que fornece uma medida equilibrada do desempenho do modelo
* Latência: O tempo gasto pelo modelo para fazer uma previsão, que é uma medida importante do desempenho prático de um modelo.

**Monitoramento de infraestrutura**

* Recursos de computação (por exemplo, CPU, memória, GPU)
* Desempenho da rede
* Armazenamento
* Logs do sistema.

**Monitoramento de viés e justiça**

Avalie regularmente o sistema de IA quanto a possíveis vieses e resultados injustos.

**Monitoramento de conformidade e IA responsável**

Certifique-se de que as operações e saídas do sistema de IA estejam em conformidade com os regulamentos relevantes, padrões do setor e diretrizes responsáveis. Monitore possíveis violações ou problemas que possam levantar preocupações de conformidade ou IA responsável.

# 18 Considerações de segurança

## 18.1 Detecção de ameaças

* Identifique e monitore possíveis ameaças à segurança, como agentes mal-intencionados que tentam explorar vulnerabilidades em sistemas de IA ou usam IA generativa para fins maliciosos. Veja a seguir alguns exemplos:
  + Gerando conteúdo falso
  + Manipulação de dados
  + Automatizando ataques
* Você pode ajudar na detecção de ameaças desenvolvendo e implantando sistemas de detecção de ameaças com tecnologia de IA. Você pode analisar o tráfego de rede, o comportamento do usuário e outras fontes de dados para detectar e responder a possíveis ameaças.

## 18.2 Gerenciamento de vulnerabilidades

* Identifique e resolva vulnerabilidades em sistemas de IA e IA generativa, incluindo bugs de software, pontos fracos do modelo e possíveis vetores de ataque (por exemplo, malware, vírus e anexos de e-mail).
* Realize regularmente avaliações de segurança, testes de penetração (tentativa de encontrar e explorar vulnerabilidades) e revisões de código para descobrir e resolver vulnerabilidades.
* Implemente processos robustos de gerenciamento e atualização de patches para garantir que os sistemas de IA sejam mantidos atualizados e seguros.

## 18.3 Proteção de infraestrutura

* Proteja a infraestrutura subjacente que dá suporte a IA e sistemas de IA generativa, como os seguintes:
  + Plataformas de computação em nuvem
  + Dispositivos de borda
  + Armazenamentos de dados
* Implemente controles de acesso fortes, segmentação de rede, criptografia e outras medidas de segurança para proteger a infraestrutura contra acesso não autorizado e ataques.
* Certifique-se de que a infraestrutura de IA seja resiliente e possa resistir a falhas, ataques ou outras interrupções.

## 18.4 Injeção imediata

* Empregue técnicas, como filtragem de prompt, limpeza e validação, para garantir que os prompts de entrada sejam seguros e não contenham conteúdo mal-intencionado.
* Desenvolva modelos robustos e procedimentos de treinamento resistentes a ataques de injeção imediatos.

## 18.5 Criptografia de dados

* Implemente mecanismos de criptografia fortes para proteger os dados em repouso e em trânsito. Dados em repouso referem-se a dados armazenados em servidores, bancos de dados ou dispositivos locais. Os dados em trânsito referem-se aos dados transmitidos durante a comunicação entre diferentes componentes do sistema de IA.
* Certifique-se de que as chaves de criptografia sejam gerenciadas adequadamente e protegidas contra acesso não autorizado.

## 18.6 O OWASP Top 10 para LLMs

O Open Web Application Security Project (OWASP) top 10 é a lista padrão do setor das 10 principais vulnerabilidades que podem afetar um sistema LLM de IA generativa.

1 **Injeção imediata:** Entradas de usuário mal-intencionadas que podem manipular o comportamento de um modelo de linguagem

2 **Tratamento de saída inseguro:** Falha em higienizar ou validar adequadamente as saídas do modelo, levando a vulnerabilidades de segurança

3 **Envenenamento de dados de treinamento:** Introduzir dados mal-intencionados no conjunto de treinamento de um modelo, fazendo com que ele aprenda comportamentos prejudiciais

4 **Modelo de negação de serviço:** Técnicas que exploram vulnerabilidades na arquitetura de um modelo para interromper sua disponibilidade

5 **Vulnerabilidades da cadeia de suprimentos:** Pontos fracos no software, hardware ou serviços usados para criar ou implantar um modelo

6 **Divulgação de informações confidenciais:** Vazamento de dados confidenciais por meio de saídas de modelo ou outros canais não intencionais

7 **Design de plug-in inseguro:** Falhas no projeto ou implementação de componentes opcionais do modelo que podem ser exploradas

8 **Agência excessiva:** Conceder a um modelo muita autonomia ou capacidade, levando a ações não intencionais e potencialmente prejudiciais

9 **Excesso de confiança:** Dependência excessiva dos recursos de um modelo, levando ao excesso de confiança e falha em auditar adequadamente seus resultados

10 **Roubo de modelo:** Acesso não autorizado ou cópia dos parâmetros ou arquitetura de um modelo, permitindo sua reutilização ou uso indevido

# 19 Entendendo os dados e a linhagem do modelo

## 19.1 Citando fontes e documentando origens

A citação da fonte e a documentação das origens dos dados são tarefas essenciais que contribuem para proteger seus sistemas de IA. Essas tarefas ajudam a garantir a transparência, rastreabilidade e responsabilidade dos dados e informações usados no sistema de IA. Isso é importante para manter a integridade e a confiabilidade do sistema. Essas tarefas envolvem fornecer informações sobre as fontes dos dados usados para treinar o modelo de IA generativa e a proveniência dos dados.

A citação da fonte na IA generativa refere-se ao ato de atribuir e reconhecer adequadamente as fontes dos dados usados para treinar o modelo.

É necessário identificar as fontes das quais os dados de treinamento foram coletados, como as seguintes:

* Datasets
* Bancos
* Outras fontes

Além disso, é necessário identificar quaisquer licenças, termos de uso ou permissões relevantes associadas aos dados.

A citação precisa da fonte ajuda os usuários e as partes interessadas a entender as origens das informações usadas para gerar o conteúdo produzido por IA. Isso os prepara para avaliar a confiabilidade da saída.

Documentar as origens dos dados no contexto da IA generativa envolve fornecer informações detalhadas sobre a proveniência ou o local de origem dos dados usados para treinar o modelo.

Isso inclui o seguinte:

* Detalhes sobre o processo de coleta de dados
* Os métodos usados para selecionar e limpar os dados
* Qualquer pré-processamento ou transformações aplicadas aos dados

Documentar as origens dos dados é importante para entender os possíveis vieses, limitações ou problemas de qualidade que podem estar presentes nos dados de treinamento. Isso pode afetar o desempenho e a confiabilidade do modelo de IA generativa.

## 19.2 Ferramentas e técnicas

### 19.2.1 Linhagem de dados

A linhagem de dados é uma técnica usada para rastrear o histórico de dados, incluindo sua origem, transformação e movimentação por diferentes sistemas.

No contexto da IA generativa, a linhagem de dados pode ser usada para documentar a jornada dos dados de treinamento, desde suas fontes iniciais até o modelo final.

Essas informações podem ser usadas para fornecer citações detalhadas de fontes e documentação de origem de dados para transparência e reprodutibilidade.

### 19.2.2 Catalogação

A catalogação envolve a organização sistemática e a documentação dos conjuntos de dados, modelos e outros recursos usados no desenvolvimento de um sistema de IA generativa.

Um catálogo bem mantido pode servir como um repositório abrangente de informações sobre os componentes do sistema de IA. Além disso, essas informações podem incluir fontes, licenças e metadados associados aos dados de treinamento.

A catalogação facilita o gerenciamento e a comunicação eficazes das origens dos dados e citações de fontes para usuários e partes interessadas.

### 19.2.3 Cartões de modelo

Os cartões de modelo são um formato padronizado para documentar os principais detalhes sobre um modelo de ML, incluindo seu uso pretendido, características de desempenho e possíveis limitações.

No contexto da IA generativa, os cartões de modelo podem ser usados para fornecer citações de fontes e documentação de origem de dados. Isso ajuda os usuários a entender a proveniência (linhagem) dos dados usados para treinar o modelo.

Os cartões de modelo podem incluir detalhes sobre os conjuntos de dados usados, suas fontes, licenças e quaisquer vieses conhecidos ou problemas de qualidade nos dados de treinamento.

**Cartões de modelo do Amazon SageMaker**

Você pode usar o Amazon SageMaker Model Cards para documentar detalhes críticos sobre seus modelos de ML em um único local para simplificar a governança e a geração de relatórios.

Os cartões de modelo podem catalogar detalhes, como o uso pretendido e a classificação de risco de um modelo, detalhes e métricas de treinamento, resultados de avaliação e observações. Ele também cataloga chamadas adicionais, como considerações, recomendações e informações personalizadas. Ao criar cartões de modelo, você pode fazer o seguinte:

* Forneça orientação sobre como um modelo deve ser usado.
* Apoiar as atividades de auditoria com descrições detalhadas do treinamento e desempenho do modelo.
* Comunique como um modelo se destina a dar suporte às metas de negócios.

# 20 Práticas recomendadas para engenharia de dados segura

## 20.1- Revisão do uso de dados na IA generativa

**Dados do usuário**

Os dados do usuário representam as entradas ou requisitos específicos fornecidos pelos clientes ou usuários finais. Esses dados são usados para gerar ou personalizar a saída do modelo de IA generativa.

Para todos os escopos de aplicativos, o cliente controla seus dados.

**Ajustando dados**

Esses dados são usados para adaptar ou ajustar o modelo de IA generativa pré-treinado às necessidades ou preferências específicas dos clientes ou do domínio do aplicativo.

Os dados de ajuste fino geralmente são um subconjunto dos dados de treinamento ou dados adicionais coletados do domínio do aplicativo.

**Dados de treinamento**

Os dados de treinamento são o conjunto de dados abrangente usado para treinar o modelo inicial de IA generativa pré-treinada.

Os dados de treinamento geralmente são uma coleção grande e diversificada de dados, como texto, imagens ou áudio, dependendo do aplicativo específico.

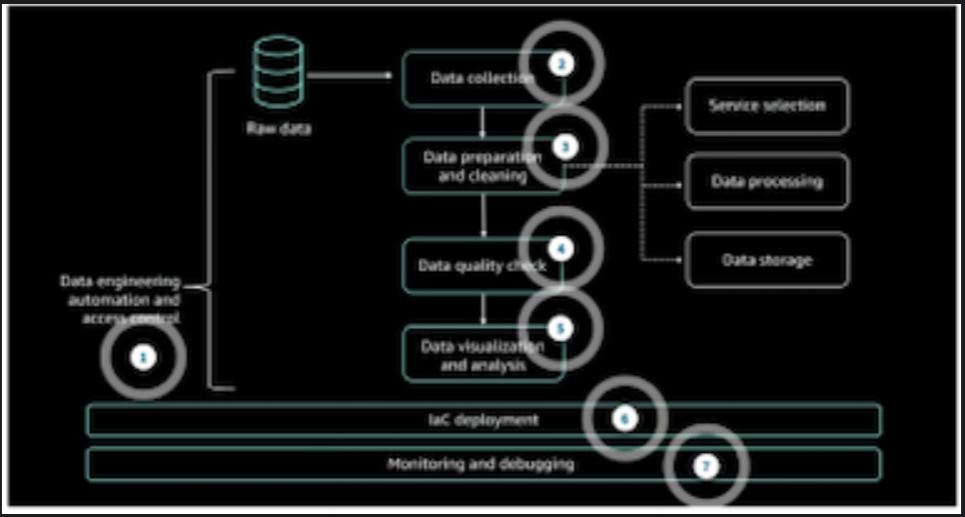
Os dados de treinamento são usados para construir o conhecimento e os recursos fundamentais do modelo de IA generativa.

## 20.2 Fluxos de dados em um aplicativo de IA generativa



## 20.3 O que é o ciclo de vida da engenharia de dados?

É um processo iterativo em que os dados são coletados, preparados e analisados. Esses dados são usados para treinar, avaliar e melhorar continuamente os modelos de IA ou IA generativa. Esse ciclo de vida garante que os dados subjacentes sejam de alta qualidade, representativos e otimizados para o caso de uso específico de IA ou IA generativa. Em última análise, esse processo contribui para o sucesso e o desempenho dos sistemas de IA ou IA generativa.



## 20.4 Engenharia de dados segura

As práticas seguras de engenharia de dados são essenciais para garantir a segurança e a confiabilidade da IA e dos sistemas de IA generativa.

### 20.4.1 Avaliando a qualidade dos dados

* Defina métricas e benchmarks claros de qualidade de dados, como os seguintes:
  + **Completude:** Os dados de treinamento abrangem uma gama diversificada e abrangente de cenários, sem lacunas ou vieses significativos.
  + **Exatidão:** Os dados de entrada usados para treinar modelos de IA são precisos, atualizados e representativos dos cenários do mundo real aos quais o modelo será aplicado.
  + **Pontualidade:** Também chamado de moeda, mede a idade dos dados em um armazenamento de dados.
  + **Consistência:** Mantenha a coerência e a consistência lógica dos dados em todo o processo de desenvolvimento e implantação de IA.
* Implemente verificações e testes de validação de dados em vários estágios do pipeline de dados.
* Execute perfis e monitoramento regulares de dados para identificar problemas de qualidade de dados.
* Estabeleça um ciclo de feedback para resolver problemas de qualidade de dados e melhorar continuamente.
* Mantenha a linhagem de dados e metadados detalhados para entender a origem e a transformação dos dados.

### 20.4.2 Implementação de tecnologias que aprimoram a privacidade

* Implemente mascaramento de dados, ofuscação de dados ou mecanismos de privacidade diferenciais para reduzir o risco de violações de dados.
* Use criptografia, tokenização ou computação segura de várias partes para proteger os dados durante o processamento e armazenamento.

### 20.4.3 Controle de acesso a dados

* Estabeleça uma estrutura abrangente de governança de dados com políticas e procedimentos claros para acesso, uso e compartilhamento de dados.
* Implemente controles de acesso baseados em funções e permissões refinadas para restringir o acesso a dados confidenciais.
* Use mecanismos de autenticação e autorização, como logon único, autenticação multifator (MFA) ou soluções de gerenciamento de identidade e acesso (IAM).
* Monitore e registre todas as atividades de acesso a dados para detectar e investigar qualquer acesso não autorizado ou anomalias.
* Revise e atualize regularmente os direitos de acesso com base no princípio do privilégio mínimo.

### 20.4.4 Integridade dos dados

* Implemente a validação de dados e verificações de integridade em vários estágios do pipeline de dados, como validação de esquema, verificações de integridade referencial e validações de regras de negócios.
* Mantenha uma estratégia robusta de backup e recuperação de dados para garantir que os dados possam ser restaurados em caso de erros, falhas do sistema ou desastres naturais.
* Empregue princípios de gerenciamento de transações e atomicidade para garantir a consistência e a confiabilidade dos dados durante o processamento e a transformação de dados.
* Mantenha uma linhagem de dados detalhada e trilhas de auditoria para rastrear a origem, as transformações e as alterações feitas nos dados.
* Monitore e teste regularmente os controles de integridade de dados para garantir sua eficácia e fazer os ajustes necessários.