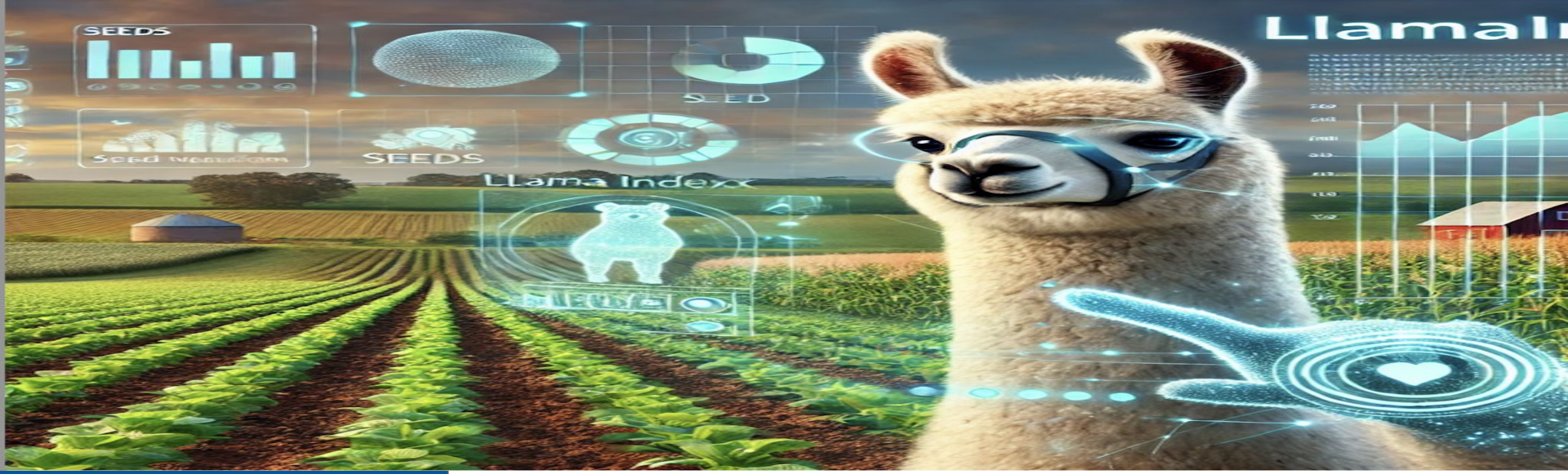


ChatBot
Aula 02

Prof. Me Daniel Vieira

SENAI

Agenda

1- LLM - Large Language Model
2 - Benefícios da LLM
3- LLM de Destaque
4 - Como utilizar a LLM Open Source
5 - Lhama Index
6- Exemplo

LLM - Long Large Model

A inteligência artificial está redefinindo os limites do possível e seu uso vai além do que vemos em assistentes virtuais ou sistemas automatizados.
No centro dessa revolução estão os **Modelos de Linguagem de Grande Porte (LLMs), como GPT e Copilot**.

Mas o que muita gente não sabe é que as LLMs abertas oferecem um novo nível de flexibilidade, transparência e controle para desenvolvedores e empresas.

LLM - Long Large Model

ChatGPT, Copilot e Gemini são exemplos de produtos prontos que utilizam LLMs por trás dos panos para fornecer funcionalidades avançadas de linguagem.

Muitos usuários e usuárias interagem com esses modelos sem nem perceber, por meio de assistentes “ocultos” que trabalham automatizando tarefas, sugerindo palavras, otimizando processos de busca e aprimorando a experiência digital de forma quase imperceptível.

LLM - Long Large Model

O sucesso desses modelos não é apenas fruto de avanço técnico, mas principalmente de quantias imensas de investimento por parte de BigTechs como Google e Microsoft.

Embora o poder financeiro e computacional tenha impulsionado grandes avanços e a popularidade da IA, a dominância de apenas algumas empresas nesse mercado tão amplo levanta algumas questões importantes

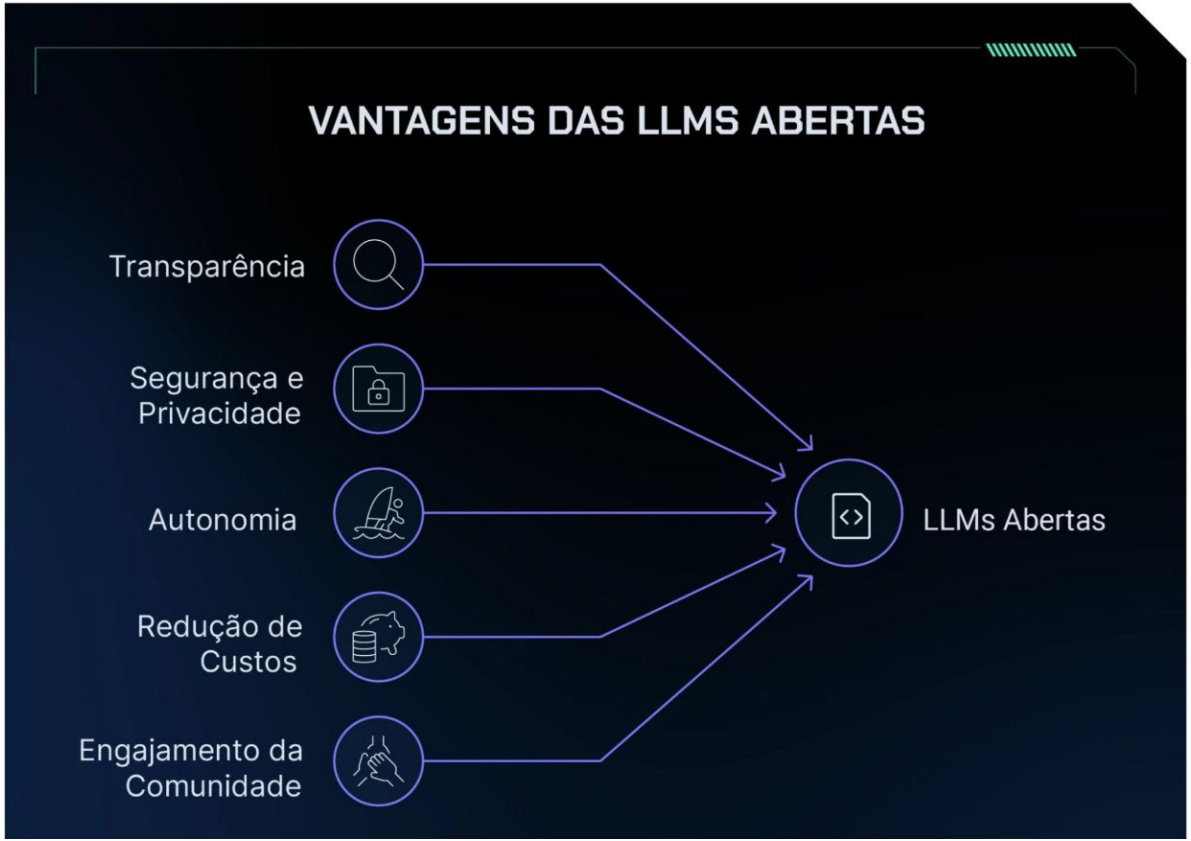
LLM - Long Large Model

- Como podemos **confiar** em decisões automatizadas de modelos proprietários **se não temos acesso ao seu funcionamento interno**?
- Quais são os **impactos de vieses que não podem ser corrigidos** por quem utiliza a tecnologia?
- Como **garantir a segurança dos dados** pessoais e empresariais ao utilizar essas LLMs?
- **Quem fica de fora** dessa tecnologia considerando os **custos** necessários para sua utilização?

LLM - Long Large Model

- Como podemos **confiar** em decisões automatizadas de modelos proprietários **se não temos acesso ao seu funcionamento interno**?
- Quais são os **impactos de vieses que não podem ser corrigidos** por quem utiliza a tecnologia?
- Como **garantir a segurança dos dados** pessoais e empresariais ao utilizar essas LLMs?
- **Quem fica de fora** dessa tecnologia considerando os **custos** necessários para sua utilização?

Benefícios das LLM Open Source



Benefícios das LLM Open Source

Uma das principais vantagens dos modelos abertos é a **transparência** total sobre seu funcionamento.

O código-fonte e os dados utilizados para o treinamento ficam disponíveis, permitindo que a comunidade possa examiná-los.

Isso minimiza a possibilidade de vieses ocultos ou práticas que possam ser implementadas sem que as pessoas usuárias saibam.

Além disso, a transparência fomenta a confiança e possibilita melhorias contínuas, já que qualquer pessoa pode identificar problemas e propor soluções.

Benefícios das LLM Open Source

Mais segurança e privacidade

Compartilhar dados com empresas terceiras pode gerar preocupações quanto ao vazamento de informações sensíveis.

Mesmo com acordos em relação à segurança dos dados manipulados pelas LLMs, não se sabe exatamente como eles estão sendo tratados.

LLMs abertas podem ser executadas localmente ou em um servidor de preferência, permitindo maior controle sobre o ambiente em que os dados são tratados.

Isso reduz risco e traz muito mais segurança, principalmente em setores críticos.

Benefícios das LLM Open Source

Autonomia

Com LLMs abertas, empresas e desenvolvedores têm liberdade para adaptar os modelos às suas necessidades específicas, seja para desenvolver funcionalidades específicas ou integrar a tecnologia em fluxos de trabalhos próprios, sem depender e/ou se preocupar com atualizações imprevisíveis

Benefícios das LLM Open Source

Redução de custos

Comparados à modelos proprietários, LLMs abertas são significativamente mais baratas. Ainda é necessário investir em infraestrutura e execução - porém, não existem taxas de licenciamento e o custo por token tende a ser muito menor.

Benefícios das LLM Open Source

Comunidade engajada

A força das LLMs abertas está na colaboração de uma comunidade ativa e diversa, composta por pesquisadores, desenvolvedores e entusiastas ao redor do mundo. Essa comunidade está sempre identificando e corrigindo falhas, compartilhando avanços e acelerando o desenvolvimento.

Além disso, um ambiente colaborativo faz muita diferença na construção de aprendizado coletivo sólido, proporcionando que a área evolua com mais rapidez e qualidade.

LLM de Destaque

Bloom

O BLOOM é uma iniciativa colaborativa da Hugging Face e do projeto BigScience, e se destaca por ser um dos poucos modelos verdadeiramente multilíngues.

Ele foi treinado com a participação de centenas de pesquisadores ao redor do mundo, com foco em inclusão e diversidade linguística.

LLM de Destaque

Mistral

A Mistral é uma organização composta por um time de desenvolvedores e cientistas de alto nível fortemente comprometidos com a transparência e acessibilidade tecnológica, que aceleram muito o jogo das LLMs abertas. A empresa lançou diversos modelos com diferentes propósitos: focado em geração de texto, matemática, geração de código, etc.

LLM de Destaque

GPT-Neo e GPT-J

São alternativas abertas aos modelos GPT proprietários desenvolvidos pela EleutherAI que podem lidar com qualquer tarefa de processamento de linguagem natural.

Mesmo com uma quantidade menor de parâmetros do que os modelos GPT mais avançados, apresentam um ótimo desempenho e são amplamente utilizados em projetos.

LLM de Destaque

Falcon

Desenvolvido pelo Instituto de Inovação em IA dos Emirados Árabes, o Falcon é um modelo eficiente que compete diretamente com alternativas proprietárias.

Ele é conhecido por ser leve e rápido, com alta performance mesmo em dispositivos menos potentes. Isso o torna ideal para empresas que buscam reduzir custos computacionais sem sacrificar a qualidade.

LLM de Destaque

LLaMA

O LLaMA é uma família de modelos de linguagem desenvolvida pela Meta, projetada para ser mais leve e eficiente do que muitos modelos do mercado.

Uma das grandes vantagens do LLaMA é sua capacidade de fornecer alto desempenho em tarefas de processamento de linguagem natural, ao mesmo tempo em que exige menos recursos computacionais em comparação a modelos como o GPT-3.

LLM de Destaque

LLaMA

O LLaMA é uma família de modelos de linguagem desenvolvida pela Meta, projetada para ser mais leve e eficiente do que muitos modelos do mercado.

Uma das grandes vantagens do LLaMA é sua capacidade de fornecer alto desempenho em tarefas de processamento de linguagem natural, ao mesmo tempo em que exige menos recursos computacionais em comparação a modelos como o GPT-3.

LLM de Destaque

Gemma

Gemma é uma família de LLMs aberta do Google que tem como base a mesma tecnologia utilizada para criar os modelos Gemini.

A proposta do Gemma é fornecer modelos flexíveis que se adequam a diversas tarefas de processamento de linguagem natural, desde geração de texto até análise de dados complexos.

Como utilizar uma LLM open source ?

1 - Escolha do Modelo Adequado: Cada LLM aberta tem suas especialidades. Modelos como Falcon e LLaMA são versáteis para tarefas gerais de NLP, enquanto outros, como Mathstral e Codestral, são mais adequados para áreas específicas como matemática e geração de código. Avaliar a documentação oficial pode ajudar a selecionar a opção que melhor atende às necessidades.

Como utilizar uma LLM open source ?

2 - **Preparação do Ambiente**: Antes de executar uma LLM, é essencial configurar um ambiente compatível. É possível rodar esses modelos localmente, utilizando hardware potente com GPUs ou TPUs, ou em serviços de nuvem como o Hugging Face Spaces. Também é importante instalar frameworks como PyTorch ou TensorFlow, dependendo do modelo escolhido.

Como utilizar uma LLM open source ?

3 - **Ajustes**: Um dos maiores benefícios das LLMs abertas é a possibilidade de ajuste fino (*fine-tuning*). Ao treinar o modelo com dados específicos da sua aplicação,é possível melhorar sua performance para tarefas personalizadas, como um chatbot especializado ou uma ferramenta de recomendação. Ferramentas como Hugging Face Transformers facilitam o processo, oferecendo pipelines prontas para treinamento e inferência.

Como utilizar uma LLM open source ?

Além do processo de *fine-tuning*, há também a possibilidade de ajustar o modelo com a técnica RAG (*Retrieval-Augmented Generation), em situações em que a base de dados está sempre mudando e crescendo, ou para economizar recursos computacionais evitando o* fine-tuning* completo.

Como utilizar uma LLM open source ?

Com o RAG, o modelo não precisa ser treinado novamente com todos os dados. Em vez disso, ele consulta uma base de conhecimento externa em tempo real, combinando recuperação de informações com geração de texto.

Como utilizar uma LLM open source ?

4 - **Teste e Validação**: Antes de colocar o modelo em produção, é essencial testar sua performance com dados de validação. Isso garante que o modelo esteja livre de erros e apresente um desempenho consistente. Para aplicações sensíveis, como saúde ou finanças, é importante verificar se o modelo atende aos requisitos éticos e regulatórios.

Como utilizar uma LLM open source ?

5- **Integração e Produção**: Após validar o modelo, ele pode ser integrado ao sistema. Em ambientes de produção, é recomendado monitorar continuamente o desempenho e o consumo de recursos para evitar gargalos e garantir a escalabilidade.

Como utilizar uma LLM open source ?

6 - Atualizações e Manutenção: Manter uma LLM atualizada é fundamental para preservar sua eficiência e segurança. Modelos abertos costumam receber contribuições e melhorias da comunidade, então é útil acompanhar as movimentações nos fóruns e repositórios! Além de reavaliar periodicamente a necessidade de ajustes para refletir mudanças nas demandas da aplicação.

Lhama Index

O **LlamaIndex** é um framework que facilita a integração de modelos de linguagem de grande porte (LLMs) com dados externos, permitindo a construção de soluções personalizadas de IA.

Lhama Index

Ele organiza e estrutura o acesso a informações, potencializando a precisão das respostas geradas pelas LLMs..

Se as LLMs são a "mente" que cria respostas, o LlamaIndex seria como um "mapa", ajudando o modelo a encontrar o caminho correto dentro de um grande volume de informações.

Ele indexa dados de diversas fontes – como documentos, bancos de dados ou APIs – permitindo que o modelo de linguagem acesse essas informações de forma rápida e organizada.

Lhama Index

Ele organiza e estrutura o acesso a informações, potencializando a precisão das respostas geradas pelas LLMs..

Se as LLMs são a "mente" que cria respostas, o LlamaIndex seria como um "mapa", ajudando o modelo a encontrar o caminho correto dentro de um grande volume de informações.

Ele indexa dados de diversas fontes – como documentos, bancos de dados ou APIs – permitindo que o modelo de linguagem acesse essas informações de forma rápida e organizada.

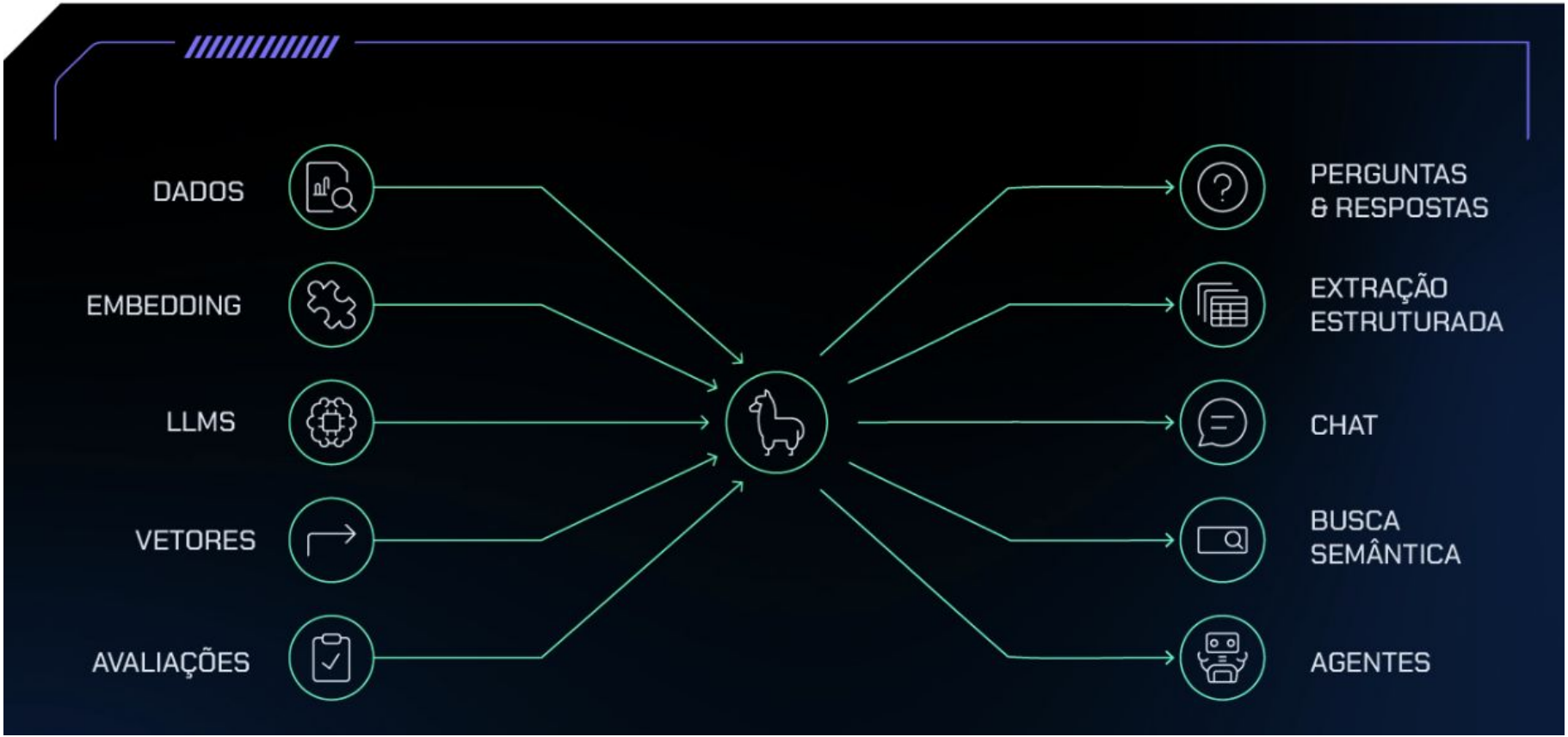
Lhama Index

Esse framework faz parte de um ecossistema mais amplo, composto também pelo LlamaCloud e pelo LlamaParse, que facilitam todas as etapas de implementação de uma solução de IA.

O LlamaCloud é uma plataforma que oferece infraestrutura escalável e automação em nuvem, enquanto o LlamaParse cuida da ingestão e pré-processamento de dados.

Juntos, esses componentes formam um sistema robusto, desde a ingestão até a entrega de respostas geradas pela IA.

Lhama Index


DADOS
EMBEDDING
LLMS
VETORES
AVALIAÇÕES
PERGUNTAS & RESPOSTAS
EXTRAÇÃO ESTRUTURADA
CHAT
BUSCA SEMÂNTICA
AGENTES

Aplicações Lhama Index

A ingestão é o momento de conectar e preparar as fontes de dados que serão consultadas.

O LlamaIndex, em conjunto com o LlamaParse, é capaz de lidar com dados em diversos formatos, como arquivos PDF, e-mails, entradas de APIs e até dados oriundos de bancos de dados SQL e NoSQL.

O LlamaParse é responsável por extrair, pré-processar e padronizar esses dados, garantindo que estejam prontos para serem indexados, enquanto o LlamaIndex fica com a tarefa de recuperar e organizar as informações.

Indexação

A indexação é o ponto chave do LlamaIndex. É nesse momento que os dados são organizados para garantir consultas velozes e eficazes - na etapa de indexação, o conteúdo é organizado de forma que não precisa ser processado toda vez que uma consulta é feita.

Indexação

Isso melhora o desempenho e garante uma resposta rápida e precisa.

Tipos de Indexação:

Indexação Hierárquica: Organiza os dados em estruturas que facilitam a recuperação de informações relacionadas.
Indexação de Texto Livre: Indexa grandes volumes de texto, permitindo consultas rápidas baseadas em palavras-chave ou contextos.
Indexação Incremental: Permite a atualização contínua de índices sem a necessidade de uma reindexação completa, ideal para sistemas dinâmicos com informações em constante mudança.

Retrieval-Augmented Generation (RAG)

O método RAG combina a geração de texto por modelos de linguagem com consultas em tempo real a uma base de dados externa.

Retrieval-Augmented Generation (RAG)

O método RAG combina a geração de texto por modelos de linguagem com consultas em tempo real a uma base de dados externa.

O LlamaIndex pode ser aplicado nesse contexto e criar um sistema que busque dados relevantes e, ao mesmo tempo, gere respostas baseadas nesses dados.

Por exemplo, no desenvolvimento de um assistente pessoal corporativo, o LlamaIndex pode ser utilizado para buscar informações em documentações internas da empresa e gerar respostas em linguagem natural.

Assim, é possível que um assistente responda perguntas com base em dados atualizados e específicos do contexto da empresa. Isso garante que as respostas serão precisas, além de economizar recursos computacionais (em comparação com fine-tuning).

Fine-Tuning

Embora o método de RAG reduza a necessidade de *fine-tuning*, há casos em que ajustes específicos no modelo são desejados.

O ajuste fino realiza alterações diretamente nos parâmetros do modelo, com base em novos conjuntos de dados.

Esse processo é custoso, mas traz melhorias significativas no desempenho de um modelo em tarefas específicas.

Agentes

Fluxo de Trabalho com Agentes:

- **Definição de Tarefas**: Um agente recebe uma tarefa ou uma série de ações a serem executadas.
- **Consulta ao Índice**: O agente consulta o LlamaIndex para acessar informações relevantes.
- **Tomada de Decisão**: Com base nos dados recuperados, o agente decide qual ação tomar ou qual resposta gerar.

Agentes

- **Integração com LlamaCloud**
- Todos os processos descritos acima podem estar integrados ao LlamaCloud, uma plataforma de infraestrutura que permite escalabilidade automática e gestão de recursos para os projetos com LlamaIndex.

Integrando Lhama Index

Experimentando com LlamaIndex no Colab

```
!pip install llama_index.llms.groq
```

Integrando Lhama Index

Então, importamos o módulo Groq, que é uma plataforma especializada na execução eficiente de modelos de linguagem de grande porte (LLMs).

O Groq otimiza o uso de recursos de hardware, permitindo que modelos de IA funcionem com alta performance, especialmente em tarefas que demandam processamento intensivo, como consultas de linguagem natural.

Este módulo nos permite conectar o LlamaIndex à infraestrutura da Groq, utilizando a chave de API para autenticação e acesso ao modelo específico escolhido para nossas consultas.

Integrando Lhama Index

```
from llama_index.llms.groq import Groq
```

Agora, precisamos de uma chave de API que permita utilizar o Llama. Você pode criar uma chave através do site do Groq. Crie sua conta, crie a API Key e copie.

No Colab, para adicionar uma chave de API, clique no ícone de chave que fica no menu lateral esquerdo. Então, dê um nome (aqui utilizei `GROQ_API`) e, no valor, cole sua chave. É necessário permitir acesso ao notebook.

Para acessar a chave, podemos utilizar o seguinte código:

```
from google.colab import userdata
GROQ_API = userdata.get('GROQ_API')
```

Integrando Lhama Index

Agora, é necessário definir qual LLM vamos utilizar.

```
llm = Groq(model='llama3-70b-8192', api_key=GROQ_API)
response = llm.complete('Qual é a
substância que dá o aroma do alecrim?')
response
paragrafos = response.text.split("\n\n")
for paragrafo in paragrafos:
    print(paragrafo)
    print()
```

Agentes

Fluxo de Trabalho com Agentes:

- **Definição de Tarefas**: Um agente recebe uma tarefa ou uma série de ações a serem executadas.
- **Consulta ao Índice**: O agente consulta o LlamaIndex para acessar informações relevantes.
- **Tomada de Decisão**: Com base nos dados recuperados, o agente decide qual ação tomar ou qual resposta gerar.

Referências

https://console.groq.com/playground

https://www.alura.com.br/artigos/llamaindex

https://github.com/alura-cursos/llamaIndex_pandas_query/blob/main/Notebooks/Aula_1.ipynb

https://www.alura.com.br/artigos/o-que-e-rag

Obrigado!

Prof. Me Daniel Vieira

Email: danielvieira2006@gmail.com

Linkedin: Daniel Vieira

Instagram: Prof daniel.vieira95