## Módulo 04 - Otimizando o desempenho do LLM

LangChain simplifica cada estágio do ciclo de vida do aplicativo LLM, incluindo desenvolvimento, produção e implantação. Fornece componentes para torná-lo mais eficiente para executar as tarefas comuns de gerenciamento de contexto ou o sequenciamento de etapas ao interagir com um LLM.

* LangChain é open source e está disponível atualmente em três linguagens de programação: Python, TypeScript e JavaScript.
* O LangChain inclui um componente retriever para buscar documentos relevantes para combinar com modelos de linguagem.

O LangChain inclui um componente retriever para buscar documentos relevantes para combinar com modelos de linguagem. Quando um usuário envia uma consulta, o retriever pesquisa no índice de documentos para encontrar os documentos mais relevantes. Ele então envia os resultados para o aplicativo para processamento posterior.

O prompt é uma instrução de texto única fornecida ao LLM como entrada para obter uma resposta.

Você pode usar índices para estruturar documentos e otimizar como os LLMs interagem com eles.

Ao criar aplicativos RAG na AWS, você pode usar o Amazon Kendra para indexar e consultar várias fontes de dados. O Amazon Kendra é um serviço totalmente gerenciado que fornece recursos de pesquisa semântica para classificação de ponta de documentos e passagens.

Armazenando e recuperando dados com memória

A memória LangChain fornece o mecanismo para armazenar e resumir (se necessário) elementos de conversação anteriores que são incluídos no contexto em invocações subsequentes.

* Maneiras de implementar memória conversacional para um chatbot:

ConversationBufferMemory: O ConversationBufferMemory é o tipo mais comum de memória no LangChain. Ele inclui conversas passadas que aconteceram entre o usuário e o LLM.

ConversationChain: O ConversationBufferMemory é construído sobre o ConversationChain, que é projetado para gerenciar conversas.

Usando cadeias para sequenciar componentes

As cadeias são úteis ao processar mais dados do que podem ser contidos no contexto.

Cadeias referem-se a sequências de chamadas - seja para um LLM, uma ferramenta ou uma etapa de pré-processamento de dados. [Linguagem de Expressão LangChain](https://python.langchain.com/v0.1/docs/expression_language/)

[(abre em uma nova aba)](https://python.langchain.com/v0.1/docs/expression_language/)

(LCEL) é a principal maneira de interagir com cadeias, e é ótima para construir cadeias. Mas há dois tipos de cadeias prontas para uso que o LangChain suporta:

* Cadeia construída usando LCEL.
* Cadeias legadas construídas por subclasses de uma classe legada 'Chain', ou seja, LLMChain etc.