

What is LangChain? AI App Development Framework Explained

Home » Blog » What is LangChain? AI App Development Framework Explained



What is LangChain?

What is LangChain?

LangChain is a framework written on TypeScript for developing applications powered by large language models (LLMs), which can be written in Python and JavaScript. As of October 2023, LangChain can be used in Node.js (ESM and

[Salesforce](#) ▾[Mobile Apps](#)[Web Apps](#)[Health](#)

Harrison Chase, based out of San Francisco, launched LangChain back in October 2022 as an open-source project. The initial version of LangChain was also launched the same year. The basic concept of LangChain is that it provides a set of tools and abstractions that make it easier to connect LLMs to other data sources, interact with their environment, and build complex applications. Simply put, LangChain can help you build AI-powered applications that not only provide responses through LLMs such as Open AI's GPT-3 or GPT-4 but also let you connect an external database to provide data to the user.

In April 2023, LangChain **announced** a seed investment of \$10 Million from Benchmark, and just a week later they announced that they raised another \$20 Million in funding at a valuation of roughly \$200 Million from **Sequoia Capital**.

What you'll learn in this guide:

What you'll learn in this guide:

- Important Features of LangChain?
- What are the Components of LangChain?
- What are the Advantages of LangChain?
- What are the Applications of LangChain?
- What is the Procedure to Utilize LangChain in Applications?
- Conclusion
- FAQ

Important features of LangChain

Utilization of LangChain

→ Model Interaction

Model Interaction, also called model I/O, is a module that lets LangChain interact with any language model. It is also capable of performing tasks such as managing inputs to the model and extracting information from its outputs.

→ Customization

LangChain is highly customizable, allowing developers to develop applications to their specific needs. For example, developers can customize the prompts that are fed to LLMs, as well as the way that responses are processed and generated. This means that your application can respond in a particular tone or even track specific information.

→ Memory

LangChain enables LLMs to retain information across interactions by supporting both, short-term and long-term memory. This means you could feed a piece of information into your AI application and it could retain that information for a long time so that you don't have to feed the information again and again. This is particularly useful for building **chatbots** that can provide context-aware responses.

Salesforce ▾

Mobile Apps

Web Apps

Health



To build a powerful AI-powered application, you need strong building blocks and components are just that. Essentially they are modular building blocks that are ready to be used to build powerful applications. There are a total of 7 components in LangChain, they are:

→ Schema

Schema in LangChain is a set of rules that define the structure and format of the data that can be used with the platform. It is used to ensure that all data is consistent and can be easily parsed and processed by LangChain's tools and services. Basically Schema in LangChain is a piece of unstructured data.

There are four main types of schema in LangChain:

[Salesforce](#) ▾[Mobile Apps](#)[Web Apps](#)[Health](#)

3. **Examples:** This represents a labeled example, which consists of an input and an output. Examples are used to train and evaluate language models.
4. **Document:** This represents a piece of unstructured data, such as a web page or a PDF file. Documents can contain text, images, code, and other types of data.

→ Models

Models are the components that perform the actual work of processing and generating text. They can be used for a variety of tasks, such as:

1. Generating text, such as poems, code, scripts, musical pieces, emails, letters, etc.
2. Translating languages
3. Answering questions in an informative way
4. Summarizing text
5. Classifying text

There are 4 models in LangChain:

1. **LLMs:** LLMs are the most powerful and versatile type of model. They can be used for a wide range of tasks, but they can also be the most computationally expensive to train and use.
2. **Chat Models:** Chat models are specifically designed for having conversations. They are often backed by LLMs, but they are tuned to be more responsive and engaging.
3. **Text Embedding Models:** Text embedding models take text as input and return a list of floats. This list of floats can then be used to perform tasks such as machine translation and text similarity.

Custom Model: Custom models can be trained on your own data to perform specific tasks. For example, you could train a custom model to answer questions about your company's products or services.

Prompts are instructions or input provided by a user to guide the model's response, helping it understand the context and generate relevant and coherent language-based output, such as answering questions, completing sentences, or engaging in a conversation.

Prompts can be simple, such as a single question or instruction, or they can be more complex, including multiple examples or constraints. The quality of the prompt has a significant impact on the quality of the output, so it is important to carefully craft prompts.

→ Indexes

Indexes in LangChain are a way to organize and structure documents so that they can be easily retrieved and used by language models. Indexes are typically based on the content of the documents, but they can also be based on other factors, such as the date the document was created or the source of the document.

LangChain provides a variety of different index implementations, including:

1. **Vector indexes:** These indexes store a vector representation of each document. Vector representations are created by embedding the text of the document into a high-dimensional space. Vector indexes can be used to efficiently search for documents that are similar to a given query.
2. **Metadata indexes:** These indexes store metadata about the documents, such as the title, author, and date. Metadata indexes can be used to quickly find documents that match specific criteria.
3. **Full-text indexes:** These indexes store the full text of each document. Full-text indexes can be used to search for documents that contain specific keywords or phrases.

[Salesforce](#) ▾[Mobile Apps](#)[Web Apps](#)[Health](#)

memory in LangChain refers to the ability of a LangChain application to store and retrieve information from previous interactions with a user. This allows the application to maintain context over a conversation, answer follow-up questions accurately, and provide a more human-like interaction.

There are two main types of memory in LangChain:

1. **Short-term memory:** This stores information about the most recent interactions with the user. Short-term memory is typically used to answer follow-up questions or to provide context for the current interaction.
2. **Long-term memory:** This stores information about the user's preferences and history. Long-term memory can be used to generate more personalized responses and to improve the accuracy of the application over time.

LangChain provides a number of different memory implementations, including:

1. **In-memory memory:** This is the simplest type of memory, and it stores all of the data in memory. In-memory memory is fast, but it can be expensive to use for large applications.
2. **Database memory:** This type of memory stores the data in a database. Database memory is more scalable than in-memory memory, but it can be slower.
3. **Cloud memory:** This type of memory stores the data in a cloud storage service. Cloud memory is scalable and cost-effective, but it can be slower than database memory.

→ Chains

Multiple components combined together form chains. Chains make it easy to implement complex applications by making them more modular and simple to

Salesforce ▾

Mobile Apps

Web Apps

Health

→ Agents

Agents allow LLMs to interact with their environment. Agents can plan sequences of actions to achieve the user's goal and can execute those sequences, including calling other LLMs, APIs, and databases. Agents are also responsible for reasoning about the user's input and deciding which actions to take and which not to take.

Advantages of using LangChain

Advantages of using LangChain



→ Simplified development

[Salesforce](#) ▾[Mobile Apps](#)[Web Apps](#)[Health](#)[11 levels.](#)

→ Increased flexibility

LangChain allows developers to connect LLMs to a wide range of other data sources and services. This gives developers more flexibility in how they design and build their applications. For example, a developer could use LangChain to build a chatbot that can access information from a private database or perform actions in the real world.

→ Improved performance

LangChain is optimized for performance, so developers can build applications that are responsive and scalable. This is important for applications that need to handle a large number of users or requests.

→ Open source

LangChain is an open-source project, which means that it is free to use and modify. This makes LangChain a good choice for developers who want to have more control over their applications.

→ Community support

LangChain has launched fairly recently, but despite that, it has managed to gain a large and active **community** of users and developers. This means that there is a

Applications of Langchain

Applications of Langchain

Applications of LangChain

Innovations using LangChain are limitless, but some of the main use cases of LangChain are:

→ Chatbots

LangChain can be used to build **chatbots** that can answer questions, generate text, and perform other tasks in a natural language way. Chatbots built with LangChain can be used in a variety of industries, such as customer service, education, and healthcare.

→ Document summarization

LangChain can be used to build applications that can summarize documents in a concise and informative way. This can be useful for a variety of tasks, such as generating meeting summaries or summarizing research papers.

→ Code generation

LangChain can be used to build applications that can **generate code** in a variety of programming languages. This can help developers to be more productive and to write code that is more reliable and efficient.

Salesforce ▾

Mobile Apps

Web Apps

Health

Translate text from one

language to another. This can be useful for a variety of tasks, such as translating documents, websites, and software applications.

→ Creative writing

LangChain can be used to build applications that can generate creative text formats, such as poems, code, scripts, musical pieces, emails, letters, etc. It can also be used to generate **images from text.** This can be used for a variety of tasks, such as generating marketing copy or writing creative content for social media.

→ Customer service

LangChain can be used to build customer service applications that can answer questions, resolve issues, and provide support to customers in a natural language way.

→ Education

LangChain can be used to build educational applications that can provide personalized instruction, generate practice problems, and assess student learning.

→ Healthcare

LangChain can be used to build healthcare applications that can provide medical information, answer patient questions, and schedule appointments.

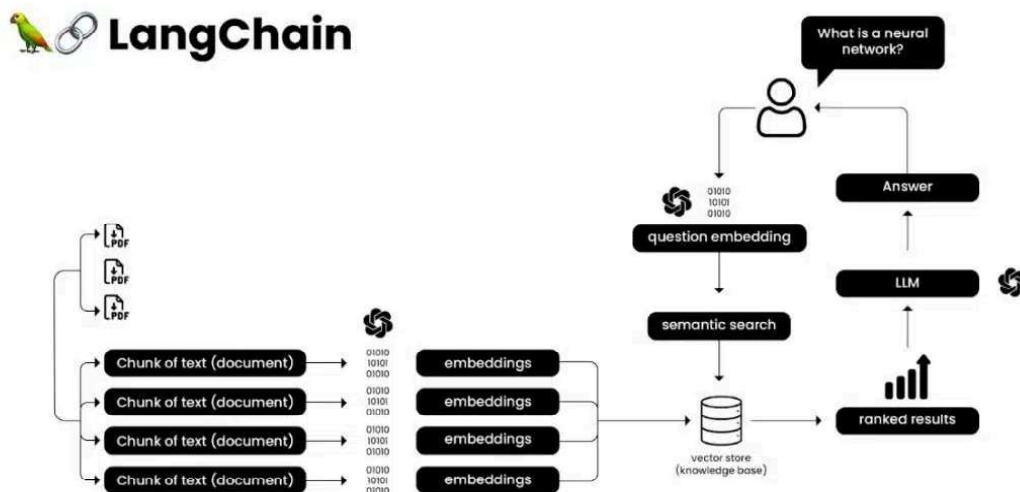
→ Research

→ Entertainment

LangChain can be used to build entertainment applications that can generate stories, games, and other creative content.

Procedure to Utilize LangChain in Applications

Procedure to Utilize LangChain in Applications



→ Install LangChain

LangChain is available as a Python package. You can install it using the following command:

[Salesforce](#) ▾[Mobile Apps](#)[Web Apps](#)[Health](#)

LangChain supports a variety of LLMs, including GPT-3, Hugging Face, and Jurassic-1 Jumbo. Choose the LLM that is best suited for your needs.

→ Create a chain

As told earlier, a chain in LangChain is a sequence of components that are connected together to perform a specific task. For example, a chain could be used to build a chatbot that can answer questions and generate text.

→ Add components to your chain

LangChain provides a variety of pre-built components, such as LLMs, data sources, and processing functions. If you become familiar with LangChain, creating your own custom components is also a possibility.

→ Connect the components in your chain

Once you have added components to your chain, you need to connect them together. This can be done by using the Chain and Agent classes.

→ Run your chain

Once you have connected the components in your chain, you can run it to perform the desired task.

Conclusion

[Salesforce](#) ▾[Mobile Apps](#)[Web Apps](#)[Health](#)

LangChain is a powerful new tool for natural language processing (NLP). It allows users to connect different NLP tasks together, creating complex and sophisticated workflows. LangChain is also highly scalable, making it possible to process large amounts of text data quickly and efficiently. LangChain has the potential to revolutionize the way we interact with computers. It can be used to create new and innovative applications in a wide range of fields.

Making a full-fledged AI-powered app with LangChain requires a team of developers who are familiar with AI programming and are proficient in other programming languages. If you are looking to get an App Developed on LangChain you can consider getting a quote from Deligence Technologies. We are a team of experienced professionals who specialize in app development and provide **LangChain Services** as well.

[Salesforce](#) ▾[Mobile Apps](#)[Web Apps](#)[Health](#)

① **Why is it called LangChain?**

① **What is LangChain used for?**

① **What is the difference between LangChain and LLM?**

① **Why is LangChain so popular?**

① **Which companies use LangChain?**

● **What problem does LangChain solve?**

LangChain mainly solves the problem of not getting personalized responses via LLMs.

Share This Article



About the Author: [Sanjay Kumar](#)



Sanjay Kumar, CEO and Co-Founder at Deligence Technologies. 19+ years of IT Experience, MCA, MBA (Finance), Spiritual Seeker.

Related Posts

[Salesforce](#) ▾[Mobile Apps](#)[Web Apps](#)[Health](#)

Type Script Benefits

March 26th, 2024 | 0 Comments



Mean S

Decem

Leave A Comment

Comment...

Name (required)

Email (required)

Website

☐ Save my name, email, and website in this browser for the next time I comment.

POST COMMENT

CERTIFICATE

Salesforce ▾

Mobile Apps

Web Apps

Health

© Copyright 2014-2025 | Deligence Technologies Inc. | All rights reserved.