

Створення додатків на базі LLM

Мета

Надати основи щодо того, що таке LLM, на що вони здатні та як створювати додатки на основі LLM.

Важливе питання

Хто коли-небудь користувався ChatGPT/ Bing AI/ Google Bard або подібною моделлю?

План

- Вступ: Розуміння LLM
- Цикл проєкту з використанням генеративного ШІ
- Вибір правильного LLM для вашого проєкту
- Основи Langchain для побудови LLM-додатків
- Інженерія запитів (Prompt Engineering): Максимізація потенціалу LLM

Що таке LLM?

Велика мовна модель (LLM) — це тип моделі машинного навчання, призначеної для розуміння та генерації тексту, схожого на людський, на основі шаблонів, які вона вивчила з величезних обсягів текстових даних.

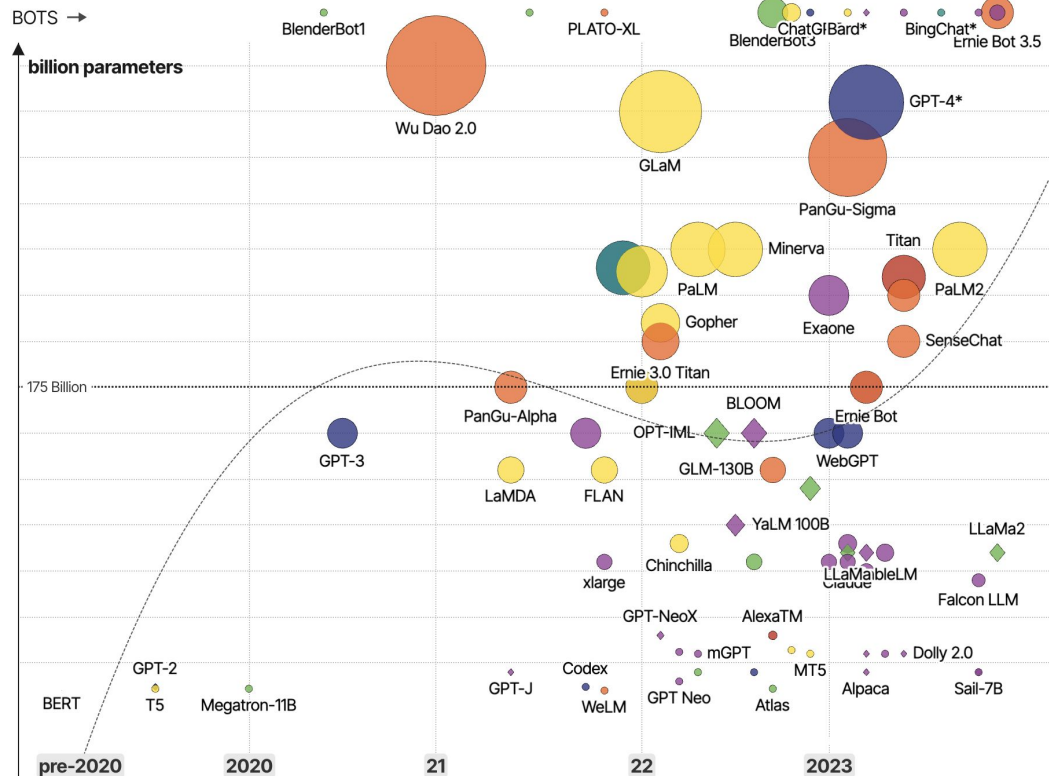
Чому ці моделі є “великими”?

Цей ярлик “великий” стосується кількості параметрів, які має мовна модель. Деякі з найуспішніших великих мовних моделей мають сотні мільярдів параметрів.

The Rise and Rise of A.I. Large Language Models (LLMs) & their associated bots like ChatGPT

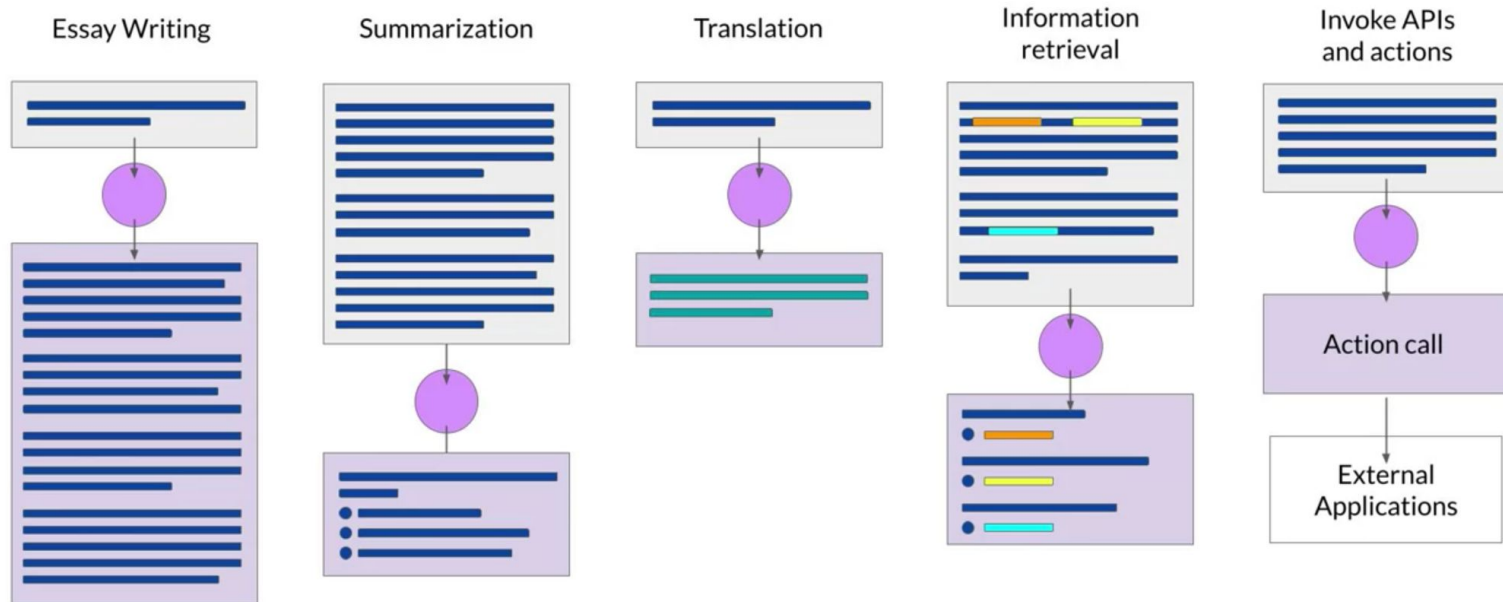
size = no. of parameters ◇ open-access

● Amazon-owned ● Chinese ● Google ● Meta / Facebook ● Microsoft ● OpenAI ● Other



Великі мовні моделі добре справляються з багатьма завданнями

Ось основні типи завдань.



Завдання, з якими LLM класно справляються 1/2

Розуміння природної мови (NLU): LLM відмінно справляються з розумінням контексту, настрою та нюансів у тексті, що робить їх цінними для завдань, таких як аналіз настроїв, класифікація тексту тощо.

Генерація природної мови (NLG): LLM можуть генерувати текст, подібний до людського. Цю можливість використовують у чат-ботах, створенні контенту, генерації історій тощо.

Відповіді на запитання: LLM можуть використовуватися для розробки систем, які надають конкретні відповіді на запити користувачів, часто використовуються в пошукових системах або спеціалізованих Q&A додатках.

Переклад: Хоча моделі, такі як трансформер Google*, були спочатку розроблені з акцентом на переклад, LLM також можуть бути точно налаштовані для завдань машинного перекладу.

Summarization: LLM можуть бути навчені надання коротких резюме довших текстів, корисних для стислого викладу новин, ведення нотаток тощо.

*Vaswani et al. "Attention is all you need", 2017

Завдання, з якими LLM класно справляються 2/2

Генерація та допомога в програмуванні: Деякі LLM навчені на наборах даних коду і можуть допомагати розробникам, пропонуючи код, виявляючи помилки або навіть генеруючи код на основі описів природною мовою.

Отримання знань: LLM можуть використовуватися для отримання конкретних частин інформації з даного набору даних або надання пояснень з широкого спектра тем на основі даних їх навчання.

Тонке налаштування для конкретних завдань: Хоча LLM навчені на великих і різноманітних наборах даних, їх можна далі точно налаштувати (finetuning) на конкретні набори даних, щоб досягти успіху в певних завданнях, таких як медична діагностика з текстових даних, аналіз юридичних документів тощо.

Завдання, з якими LLM класно справляються

Розуміння природної мови (NLU): LLM відмінно справляються з розумінням контексту, настрою та нюансів у тексті, що робить їх цінними для завдань, таких як аналіз настроїв, класифікація тексту тощо.

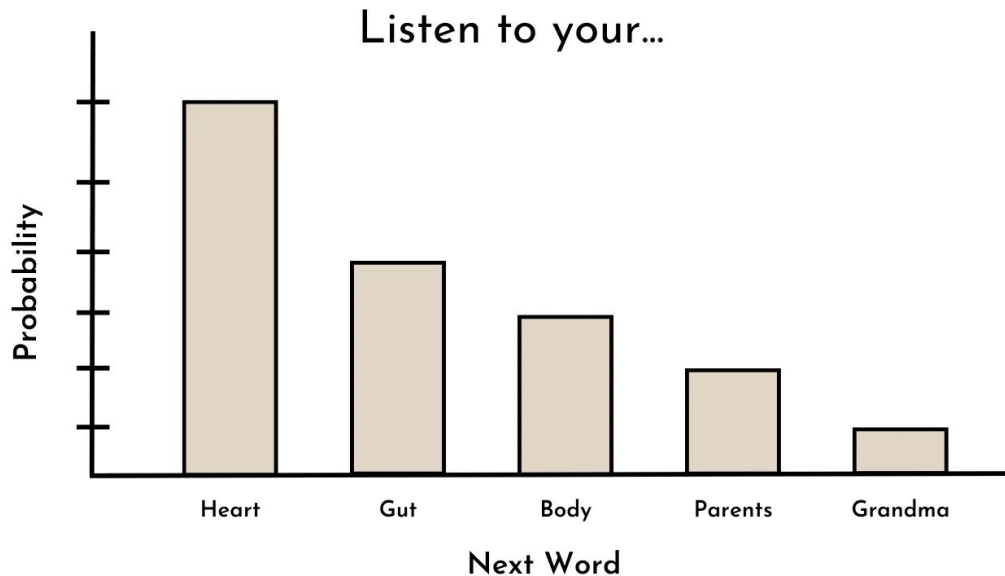
Генерація природної мови (NLG): LLM можуть генерувати текст, подібний до людського. Цю можливість використовують у чат-ботах, створенні контенту, генерації історій тощо.

Відповіді на запитання: LLM можуть використовуватися для розробки систем, які надають конкретні відповіді на запити користувачів, часто використовуються в пошукових системах або спеціалізованих Q&A додатках.

Переклад: Хоча моделі, такі як трансформер Google*, були спочатку розроблені з акцентом на переклад, LLM також можуть бути точно налаштовані для завдань машинного перекладу.

На якій задачі навчені LLM

Основним завданням для навчання більшості найсучасніших LLM є прогнозування слів. Іншими словами, враховуючи послідовність слів, яка ймовірнісний розподіл наступного слова?

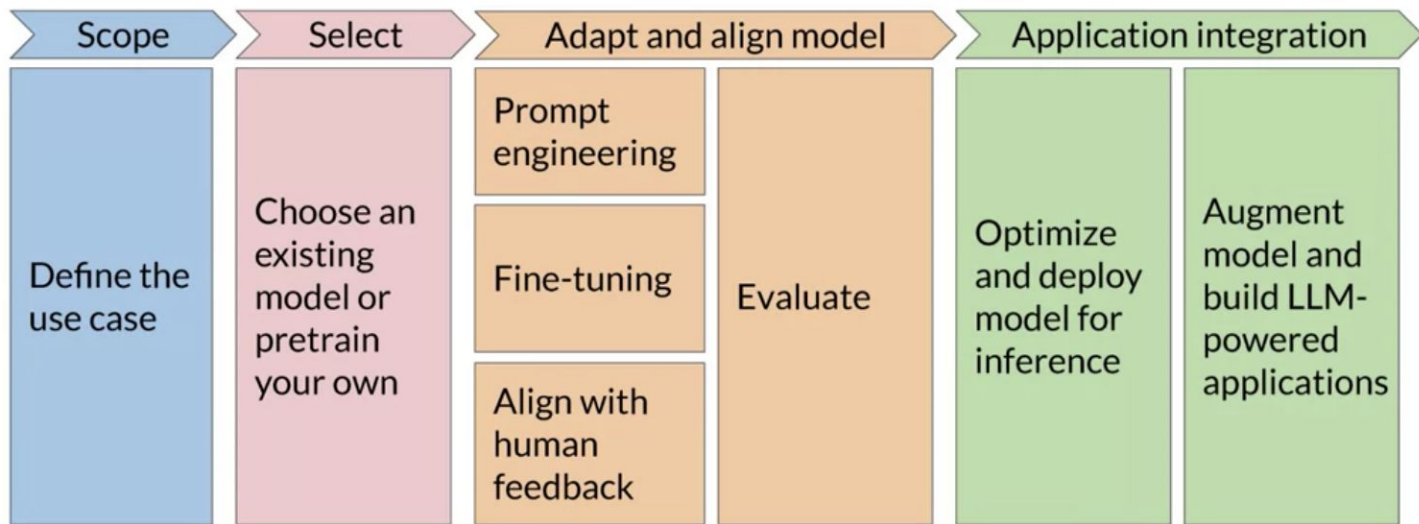


Ціль навчання LLMs

Основним завданням машинного навчання для таких моделей, як GPT (Генеративний попередньо навчений трансформер), є автогресивне мовне моделювання. У цьому завданні модель передбачає наступне слово (або токен) у послідовності на основі попередніх слів.

Як розробити застосунок на основі LLM?

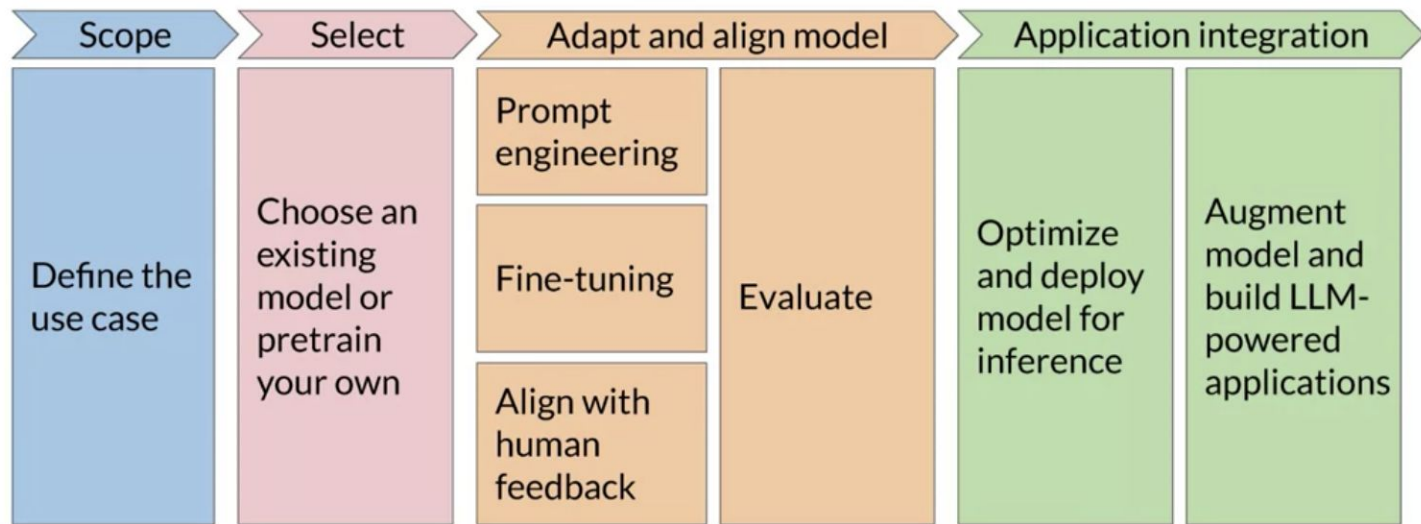
Розглянемо життєвий цикл проєкту з використанням генеративного ШІ в загальному.



Джерело: [Generative AI with Large Language Models](#)

Як розробити застосунок на основі LLM?

Розглянемо життєвий цикл проєкту з використанням генеративного ШІ в загальному.



Ми вже говорили про варіанти використання, а також сьогодні поговоримо про вибір правильної моделі та інженерії підказок.

Вибір правильної моделі під задачу

Перед тим, як зануритися в безліч доступних великих мовних моделей, важливо зрозуміти ваш конкретний випадок використання. Задайте собі такі питання:

- Яка головна мета використання великої мовної моделі?
- Які саме завдання ви хочете, щоб модель виконувала?
(наприклад, генерування тексту, аналіз настроїв, переклад)
- Який очікуваний обсяг вашого використання мовної моделі?
(наприклад, низький трафік або продакшн впровадження)
- Чи є у вас які-небудь обмеження на доступні обчислювальні ресурси?


Уточнивши свої вимоги та очікування, ми можемо звузити вибір і вибрати модель, яка найкраще відповідає нашим потребам.

Ключові міркування щодо вибору моделі

- **Розмір моделі та можливості:** Різні мовні моделі мають різний розмір і можливості. Розгляньте компроміси між розміром моделі, можливостями та потребами в ресурсах на основі вашого випадку використання.
- **Дані для попереднього навчання та межа знань:** Мовні моделі навчаються на величезних обсягах текстових даних. Однак межа знань моделі є важливим фактором для розгляду. Моделі, треновані на даних до певної дати, можуть не містити інформації з недавніх джерел. Якщо вашому додатку потрібна актуальна інформація, важливо вибрати модель з новішою межею знань.
- **Можливості finetuning:** Деякі великі мовні моделі дозволяють finetuning, де ви можете навчити модель на своїй специфічній області або специфічному наборі даних. Якщо finetuning важливий для вашого випадку використання, переконайтеся, що модель, яку ви обрали, підтримує тонке налаштування і пропонує необхідну вам гнучкість.
- **Продуктивність і затримка (latency):** Оцінюйте показники продуктивності моделі, такі як точність, precision і recall, на відповідних показниках (benchmarks) або завданнях. Крім того, розгляньте час відповіді (response time) або затримку моделі, особливо якщо вам потрібні програми в реальному часі або з низькою затримкою.
- **Доступність та вартість:** Різні мовні моделі мають різні міркування щодо доступності та вартості. Деякі моделі є відкритими і доступними безкоштовно, в той час як інші можуть вимагати ліцензій або мати пов'язані з ними витрати на використання. Розгляньте свій бюджет і наявність ресурсів, щоб вибрати модель, яка відповідатиме вашим фінансовим можливостям.

Де знайти моделі з відкритим кодом

<https://huggingface.co/models>

 **Hugging Face**

Models Datasets Spaces Docs Solutions Pricing Log In Sign Up

Tasks Libraries Datasets Languages Licenses Other

Multimodal


- Feature Extraction
- Text-to-Image
- Image-to-Text
- Text-to-Video
- Visual Question Answering
- Document Question Answering
- Graph Machine Learning


Computer Vision


- Depth Estimation
- Image Classification
- Object Detection
- Image Segmentation
- Image-to-Image
- Unconditional Image Generation
- Video Classification
- Zero-Shot Image Classification


Models 327,097


new Full-text search Sort: Trending


 **tiiuae/falcon-180B**
Text Generation • Updated 5 days ago • 30.2k • 487


 **tiiuae/falcon-180B-chat**
Text Generation • Updated 5 days ago • 4.85k • 254


 **lillyasviel/sd_control_collection**
Updated 2 days ago • 263


 **stabilityai/stable-diffusion-xl-base-1.0**
Text-to-Image • Updated 7 days ago • 1.05M • 2.54k


 **meta-llama/Llama-2-7b**
Text Generation • Updated Jul 19 • 2.4k

 **baichuan-inc/Baichuan2-13B-Chat**
Text Generation • Updated 3 days ago • 170k • 107

 **WizardLM/WizardCoder-Python-34B-V1.0**
Text Generation • Updated 2 days ago • 31k • 544

 **meta-llama/Llama-2-7b-chat-hf**
Text Generation • Updated Aug 9 • 474k • 1.1k

 **Phind/Phind-CodeLlama-34B-v2**
Text Generation • Updated 14 days ago • 6.11k • 226

 **PY007/TinyLlama-1.1B-step-50K-105b**
Text Generation • Updated 2 days ago • 4.64k • 72

**Давайте подивимося, як
працювати з цими моделями
за допомогою популярного
фреймворку!**

LangChain



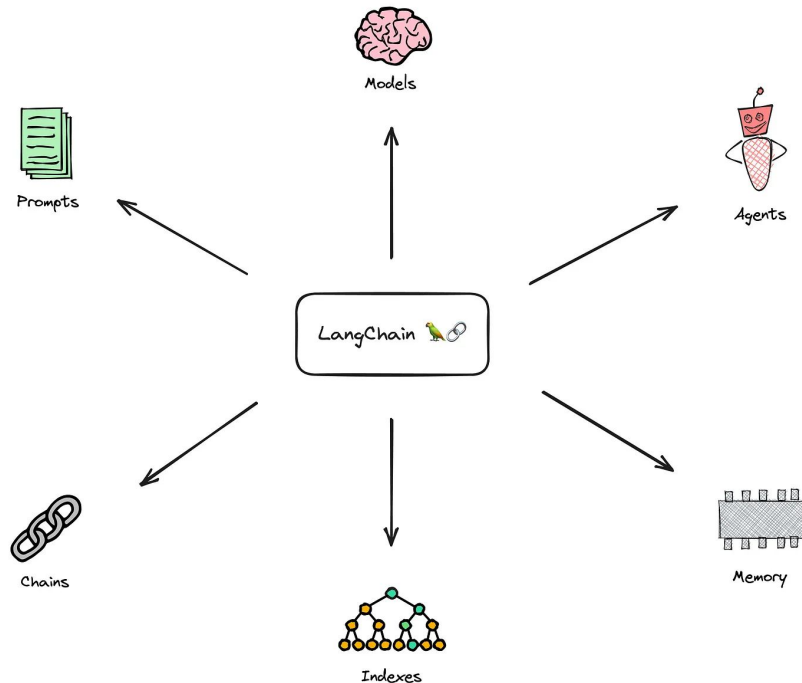
LangChain

LangChain – це фреймворк з відкритим кодом, розроблений для полегшення розробки додатків, що використовують великі мовні моделі (LLMs).

Основна ідея бібліотеки полягає в тому, що ми можемо утворювати ланцюги (chains) з різних компонент, щоб розширити функціонал LLMів. LangChain складається з кількох компонент - модулів.

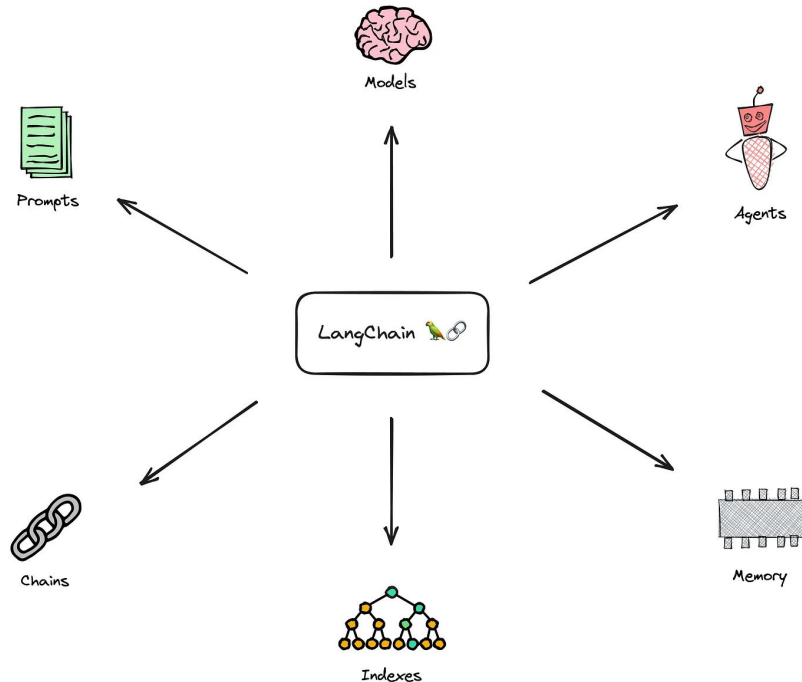
Модулі в Langchain

- **Запити/Prompts:** Цей модуль дозволяє нам створювати динамічні промпти, використовуючи шаблони. Він може адаптуватися до різних типів LLM в залежності від розміру контекстного вікна та вхідних змінних, які використовуються як контекст, таких як історія розмов, результати пошуку, попередні відповіді та багато інших.
- **Моделі/Models:** Цей модуль надає абстрактний рівень для підключення до більшості доступних сторонніх API LLM. Він має API-з'єднання з ~40 публічними LLM, моделями чатів та вбудованими моделями.
- **Пам'ять/Memory:** Це дає LLM доступ до історії розмов.



Модулі в Langchain

- **Індекси/Indexes:** Індекси належать до способів структурування документів, щоб LLM могли найкраще з ними взаємодіяти. Цей модуль містить утиліти для роботи з документами та інтеграцію з різними векторними базами даних.
- **Агенти/Agents:** Деякі додатки вимагають не лише попередньо визначеного ланцюга викликів до LLM або інших інструментів, але й потенційно невідомого ланцюга, який залежить від введення даних користувачем. У таких типах ланцюгів є агент з доступом до набору інструментів. В залежності від вводу користувача, агент може вирішити, який — якщо є — інструмент викликати.
- **Ланцюги/Chains:** Використання LLM в ізоляції підходить для деяких простих додатків, але багато більш складних вимагають “ланцюгування” LLM, або один з одним, або з іншими експертами (моделями). LangChain надає стандартизований інтерфейс для ланцюгів, а також кілька загальних реалізацій ланцюгів для зручності використання.



Додаткові матеріали

[Open Youtube playlist about Langchain](#)

[Langchain tutorials](#)

[LangChain documentation](#)

Хороший курс для старту: [Generative AI with Large Language Models](#)