1. Giới thiệu chủ đề

Hôm nay, chúng ta sẽ đi sâu vào kiến trúc Transformer và ứng dụng của nó trong các mô hình ngôn ngữ lớn (LLM). Nội dung này có giá trị đối với cả người mới bắt đầu lẫn những ai muốn tối ưu hóa ứng dụng AI.

Các khái niệm quan trọng sẽ đề cập

- Copilot & Agent: Hệ thống AI hỗ trợ người dùng trong nhiều tác vụ khác nhau.
- Token & Context Window: Cách AI xử lý ngôn ngữ và ghi nhớ thông tin trong cuộc hội thoại.
- Tham số & Trọng số: Các yếu tố quyết định hiệu suất của LLM.
- Chi phí API: Ảnh hưởng của giá thành đến việc triển khai AI.

2. Kết quả bình chọn "Trận chiến AI"

Cuộc bầu chọn giữa GPT-4, Claude-3 Opus và Gemini đã mang đến những kết quả bất ngờ. Claude-3 Opus (Blake) giành chiến thắng, cho thấy sự ưa chuộng của người dùng đối với khả năng phản hồi của mô hình này.

=> Gợi ý: Bạn có thể tự thực hiện bài kiểm tra tương tự để có cái nhìn khách quan hơn. Ngoài ra, hãy thử trò chơi "**Outsmart**" trên trang web cá nhân để so sánh AI trong các tình huống thực tế.

3. Lịch sử phát triển mô hình Transformer

- 2017: Google công bố bài báo "Attention Is All You Need", giới thiệu kiến trúc Transformer.
- 2018: GPT-1 ra đời, cùng với BERT từ Google.
- 2019: GPT-2 xuất hiện.
- 2020: GPT-3 ra đời.
- 2022: ChatGPT (GPT-3.5 + RLHF) tạo bước đột phá lớn.
- 2023: GPT-4 ra mắt.
- 2024: GPT-4o và các phiên bản mới tiếp tục phát triển.

4. Thế giới phản ứng thế nào về AI?

Giai đoạn đầu

Hứng thú: AI thể hiện khả năng trả lời thông minh, chính xác.

Hoài nghi: Một số chuyên gia gọi AI là "vẹt xác suất" (stochastic parrot) – mô hình dự đoán hơn là hiểu thực sự.

Những xu hướng đáng chú ý:

- AI không "hiểu" như con người, nhưng quy mô lớn giúp tạo ra trí thông minh xuất hiện (Emergent Intelligence).
- Trước đây: Prompt Engineer từng là nghề hot, nhưng giờ giảm nhu cầu do AI ngày càng dễ sử dụng.
- Custom GPTs trên GPT Store từng phổ biến nhưng hiện nay đang bão hòa.
- Co-Pilot AI như Microsoft Copilot, GitHub Copilot đang được tích hợp rộng rãi vào công cụ làm việc.

- Xu hướng tương lai: Agentic AI – AI có thể tự lập kế hoạch, chia nhỏ nhiệm vụ, phối hợp với AI khác, và duy trì trí nhớ lâu dài.

5. Tham số (Parameters) và Trọng số (Weights) trong LLM

Tham số = Trọng số (hầu hết trường hợp)

Trọng số quyết định cách AI dự đoán từ tiếp theo dựa trên dữ liệu huấn luyện.

Mô hình	**Số tham số**
Machine Learning truyền thống	20 - 200
GPT-1 (2018)	117 triệu
GPT-2 (2019)	1,5 tỷ
GPT-3 (2020)	175 tỷ
GPT-4 (2023)	1,76 nghìn tỷ
Mô hình tiên tiến nhất	Khoảng 10 nghìn tỷ (chưa công bố)

Dòng LLaMA & Gemma

- Gemma & LLaMA 3.2: 2 tỷ tham số (tối ưu, nhẹ).
- LLaMA 3.1 có nhiều phiên bản:
  - 8 tỷ tham số (trung bình).
  - 70 tỷ tham số (mạnh hơn).
  - 405 tỷ tham số (mạnh nhất trong mã nguồn mở).

6. Tokens – Đơn vị nhỏ nhất trong LLM

AI có thể học theo nhiều cách:

1. Học theo ký tự: Kích thước từ vựng nhỏ nhưng khó ghép ký tự thành từ.
2. Học theo từ: Hiểu nghĩa nhanh nhưng từ vựng quá lớn.
3. Phương pháp subword (GPT hiện tại) – cân bằng giữa hai cách trên.

Lợi ích của Subword Tokenization

- Nhận diện tên riêng tốt hơn.
- Xử lý từ có nhiều biến thể hậu tố.
- Giữ ý nghĩa từ trong trường hợp từ hiếm.

Nguyên tắc Tokenization

- Từ phổ biến được giữ nguyên.
- Từ hiếm bị chia nhỏ nhưng vẫn giữ ý nghĩa.

Ước tính số token

- 1 token ≈ 4 ký tự tiếng Anh (~3-4 ký tự với tiếng Việt).
- 1.000 tokens ~ 750 từ tiếng Anh (~600-700 từ tiếng Việt).
- Tác phẩm của Shakespeare ≈ 1,2 triệu tokens.

7. Context Window – Giới hạn trí nhớ của AI

Context Window xác định số token tối đa mà mô hình có thể xử lý trong một lần.

Bao gồm prompt gốc, cuộc hội thoại trước đó, prompt mới nhất và đầu ra của AI.

Các mô hình hàng đầu:

- Gemini 1.5 Flash – 1 triệu tokens (cao nhất).
- Claude 3.5 – 200.000 tokens.
- GPT-4 Turbo – 128.000 tokens.

=> Ứng dụng thực tế: Nếu muốn hỏi về toàn bộ tác phẩm Shakespeare, tất cả nội dung phải nằm trong **Context Window.**

8. Chi phí API & Cách tối ưu hóa

Chi phí API phụ thuộc vào số token đầu vào & đầu ra.

Mô hình	Chi phí (Input / Output per million tokens)
Claude 3.5 Sonnet	$3 / $15
GPT-4 Mini	$0.15 / $0.60

Cách tiết kiệm chi phí

- Giới hạn số token đầu ra.
- Theo dõi bảng giá khi xây dựng hệ thống lớn.
- Dùng Llama nếu không muốn tốn phí API.

9. Tổng kết Ngày thứ tư

Học được gì hôm nay?

- Kiến trúc Transformer & lịch sử phát triển.
- Tokenization và Context Window trong AI.
- Chi phí API và cách tối ưu hóa.
- So sánh mô hình AI và kết quả Trận chiến AI.

Ứng dụng trong thực tế

- Gọi API OpenAI và LLaMA để tóm tắt nội dung.
- Hiểu vì sao LLM gặp khó khăn với bài toán đếm chữ cái.
- Biết cách kiểm soát chi phí API khi sử dụng AI.

Hướng đi tiếp theo

Nâng cao hiểu biết về Agentic AI & khả năng tự lập kế hoạch của AI. Tìm hiểu về cách LLM xử lý dữ liệu và tối ưu hóa Context Window.