

# Lecture 21: Lý luận phức tạp

## [Giới thiệu](#)

### [What is Reasoning?](#)

[Types of Reasoning \(examples: Huang and Chang 2023\)](#)

### [Pre-LLM Reasoning Methods](#)

[Computational Semantics](#)

[Memory Networks \(Sukhbaatar et al. 2015\)](#)

[Solving Word Problems with Symbolic Reasoning](#)

### [Chain of Thought and Variants](#)

[Review: Chain-of-thought Prompting \(Wei et al. 2022\)](#)

[Review: Zero-shot Chain of Thought \(Kojima et al. 2022\)](#)

[Self Ask \(Press et al. 2022\)](#)

[Chain of Thought with Retrieval \(He et al. 2023\)](#)

[Multilingual Chain of Thought Reasoning \(Shi et al. 2022\)](#)

[Complex-based Prompting \(Fu et al. 2022\)](#)

### [Systematics Studies of Reasoning in LLMs](#)

[Reasoning is an “Emergent” Ability \(Wei et al. 2022\)](#)

[Reliability of Explanations \(Ye and Durrett 2022\)](#)

### [Training for Chain of Thought](#)

[ORCA: Training Small Models for Reasoning \(Mukherjee et al. 2024\)](#)

[Chain of Thought Reward Models \(Lightman et al. 2023\)](#)

### [Abductive Reasoning: Learning Patterns from Data](#)

[Inference to Explanations](#)

[Rule Induction with LLMs](#)

[Learning Differences between Text Collections \(Zhong et al. 2023\)](#)

### [Resources](#)

## Giới thiệu

Trong bài viết này, tôi muốn đề cập đến vấn đề lý luận phức tạp trong các mô hình ngôn ngữ. Chúng ta đã thảo luận một chút về lý luận trong các mô hình ngôn ngữ trước đây, nhưng hôm nay tôi sẽ nói về những khía cạnh mà chúng ta chưa đề cập.

## What is Reasoning?

Lý luận là quá trình sử dụng bằng chứng và logic để đưa ra kết luận và đánh giá. Trong bối cảnh các mô hình ngôn ngữ, khái niệm này có phần phức tạp hơn. Từ góc độ triết học, có hai loại lý luận chính: lý luận hình thức và lý luận không hình thức.

Lý luận hình thức chủ yếu dựa trên các giá trị chân lý nghiêm ngặt. Điều này có nghĩa là bạn có thể khẳng định một điều là đúng hoặc không đúng một cách rõ ràng. Trong thực tế, lý luận hình thức rất hiếm gặp, ngoại trừ trong các lĩnh vực như toán học hay khoa học máy tính.

Ngược lại, lý luận không hình thức dựa trên kinh nghiệm và trực giác. Trước đây, việc nghiên cứu lý luận không hình thức khá khó khăn, nhưng với sự phát triển của các mô hình ngôn ngữ lớn, điều này đã trở nên khả thi hơn. Đây được coi là một trong những bước đột phá lớn trong những năm gần đây.

## Types of Reasoning (examples: Huang and Chang 2023)

Trong nghiên cứu về lý luận trong các mô hình ngôn ngữ lớn, có ba loại lý luận chính mà chúng ta cần chú ý:

1. Lý luận suy diễn (Deductive Reasoning): Đây là quá trình sử dụng logic để đi từ các giả thuyết đến kết luận. Ví dụ, nếu chúng ta biết rằng "tất cả các loài động vật có vú đều có thận" và "tất cả cá voi đều là động vật có vú", thì chúng ta có thể kết luận rằng "tất cả cá voi đều có thận".

2. Lý luận quy nạp (Inductive Reasoning): Loại lý luận này dựa trên quan sát để đưa ra một kết luận tổng quát. Ví dụ, khi thấy một sinh vật có cánh, chúng ta có thể dự đoán rằng đó là một con chim. Tuy nhiên, điều này không phải lúc nào cũng đúng, vì không phải tất cả sinh vật có cánh đều là chim.

3. Lý luận suy diễn giải thích (Abductive Reasoning): Đây là quá trình từ một quan sát để dự đoán giải thích có khả năng xảy ra nhất. Chẳng hạn, nếu một chiếc xe không khởi động và có một vũng chất lỏng dưới động cơ, chúng ta có thể suy luận rằng xe có thể bị rò rỉ ở bộ tản nhiệt.

Ngoài ba loại lý luận này, còn nhiều loại khác như lý luận tương tự (analogical reasoning), nhưng hôm nay chúng ta chỉ tập trung vào ba loại chính. Cần lưu ý rằng các loại lý luận này có thể không hoàn toàn rõ ràng và có thể có sự chồng chéo giữa các định nghĩa khác nhau.

## Pre-LLM Reasoning Methods

### Computational Semantics

Trước khi đi vào các phương pháp lý luận hình thức, tôi muốn giới thiệu một số phương pháp lý luận sơ bộ. Phương pháp đầu tiên liên quan đến lý luận hình thức trong ngữ nghĩa tính toán, một lĩnh vực đã tồn tại từ lâu. Phương pháp này sử dụng lý luận suy diễn, bắt đầu từ các tiền đề và đi đến các kết luận cuối cùng.

Trong lý luận hình thức, chúng ta sử dụng các ký hiệu như "tất cả" và "tồn tại". Ví dụ, câu "tất cả X đều chết" có thể suy ra rằng Mia và Zed đã chết. Một ví dụ khác là "tất cả X là động vật có vú" có thể dẫn đến việc "X có thận".

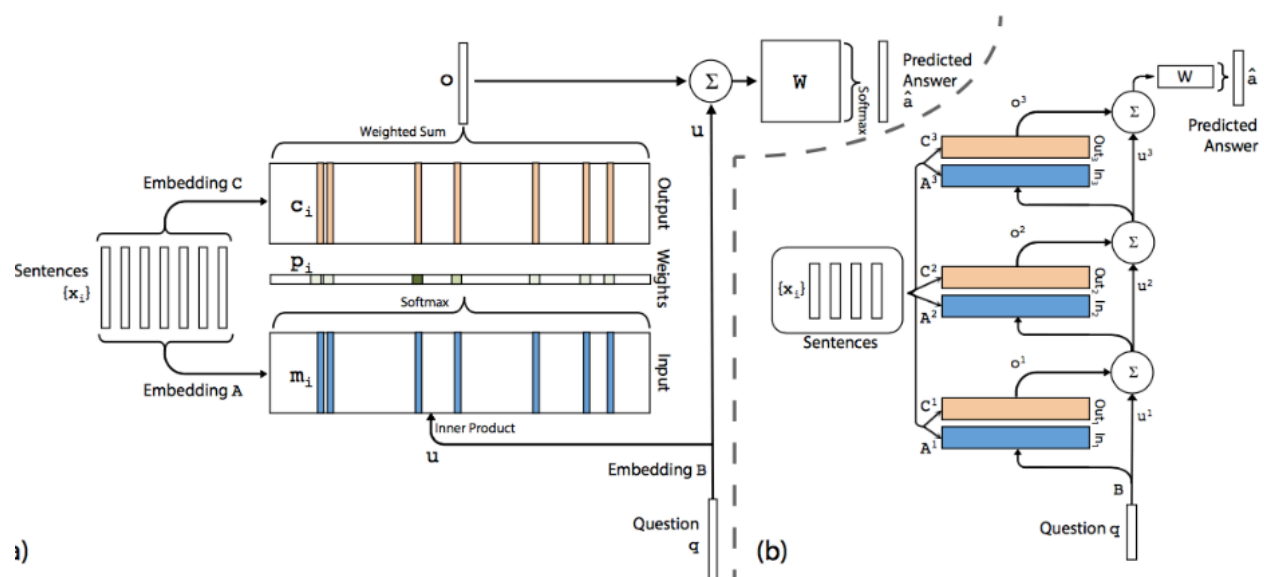
Tài liệu tham khảo yêu thích của tôi về chủ đề này là cuốn sách của Blackburn và Buz, nơi cung cấp nhiều ví dụ và giải thích chi tiết về cách thực hiện các suy diễn. Mặc dù các mạng nơ-ron có thể thực hiện một số dạng lý luận thông qua phương pháp Chain of Thought, nhưng điều này chỉ là một sự gần đúng và không hiệu quả cho các câu hỏi phức tạp.

Chẳng hạn, trong Prolog, bạn có thể hỏi một cơ sở tri thức liệu tất cả những người làm việc tại CMU có phải là giáo sư và có bằng tiến sĩ hay không. Trong khi đó, một mô hình ngôn ngữ, dù có quyền truy cập vào hồ sơ của mọi người, cũng không thể đảm bảo trả lời chính xác câu hỏi này, đặc biệt là khi có nhiều bước suy diễn.

Một số vấn đề với lý luận hình thức là nó chỉ xử lý các tuyên bố đúng hoặc sai một cách nghiêm ngặt. Khi gặp các vấn đề phức tạp hơn, việc chứng minh trở nên tốn kém về mặt tính toán. Gần đây, có những thuật toán tìm kiếm trong không gian chứng minh sử dụng mô hình nơ-ron để tăng tốc độ tìm kiếm bằng cách chọn các giả thuyết hứa hẹn nhất. Ví dụ, Sean Wellik tại CMU đang nghiên cứu về chứng minh định lý toán học, kết hợp giữa phương pháp cổ điển và hiện đại.

## Memory Networks (Sukhbaatar et al. 2015)

Mạng nhớ hoạt động dựa trên khả năng ghi và đọc từ bộ nhớ. Cụ thể, khi có một truy vấn (query), mô hình sẽ lấy embedding của truy vấn đó, sau đó tính toán tích vô hướng và áp dụng hàm softmax để tạo ra một trọng số cho các embedding. Quá trình này tương tự như cơ chế attention, cho phép mô hình tổng hợp thông tin từ một cơ sở dữ liệu lớn. Ngoài ra, mạng nhớ còn có khả năng cập nhật bộ nhớ thông qua các thao tác ghi, cho phép nó không chỉ đọc mà còn ghi thông tin mới vào bộ nhớ.



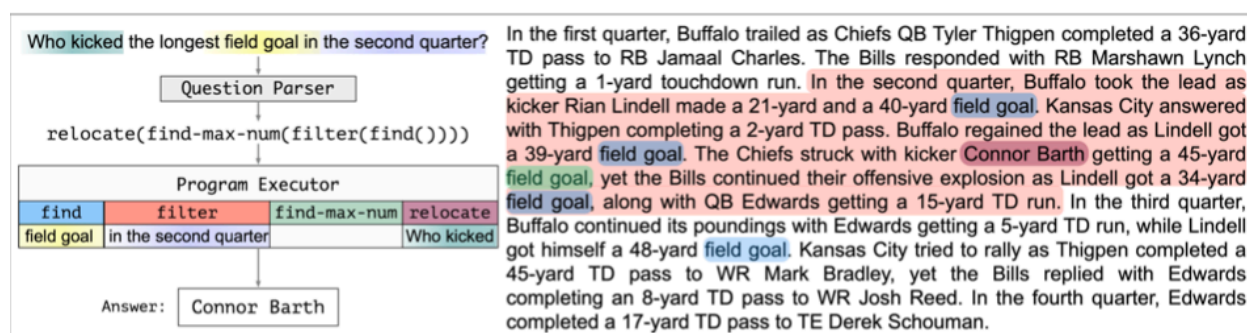
Một trong những vấn đề lớn hiện nay với các mô hình ngôn ngữ lớn là chúng không thể liên tục cập nhật bộ nhớ của mình. Mặc dù có thể thêm văn bản vào bộ nhớ, nhưng việc này có những giới hạn nhất định, vì văn bản không phải lúc nào cũng là cách tốt nhất để mã hóa tất cả những

gì đã được học trong quá khứ. Do đó, việc tích hợp các kiến trúc như mạng nhớ vào các mô hình ngôn ngữ có thể là một hướng nghiên cứu thú vị trong tương lai.

## Solving Word Problems with Symbolic Reasoning

Một khía cạnh thú vị mà chúng ta chưa bàn nhiều là khả năng giải quyết các câu hỏi thông qua lập luận biểu tượng. Mặc dù chúng ta đã đề cập đến một số khía cạnh trong việc tạo mã, nhưng lập luận biểu tượng đã tồn tại từ lâu và có thể mang lại nhiều lợi ích.

Cách thức hoạt động của lập luận biểu tượng là bạn có thể sử dụng các phép toán như tìm kiếm, lọc và xác định giá trị lớn nhất từ một tập hợp dữ liệu. Những phép toán này cho phép chúng ta thao tác một cách rõ ràng với dữ liệu, chẳng hạn như tìm ra số lớn nhất trong một tập hợp hoặc lọc ra những thông tin phù hợp từ một khối văn bản lớn.



Điều này trở nên quan trọng vì một số nhiệm vụ mà mạng nơ-ron hiện tại gặp khó khăn, như việc tìm kiếm số lớn nhất trong một tập dữ liệu lớn hoặc xác định các yếu tố phù hợp trong một tập hợp thông tin. Hiện tại, lập luận biểu tượng chưa được áp dụng rộng rãi trong các mô hình ngôn ngữ lớn, chủ yếu vì các nhà nghiên cứu đang tập trung vào các kỹ thuật prompting để thực hiện các phép lập luận này.

Tuy nhiên, tôi tin rằng đây là một lĩnh vực đáng để xem xét và tìm kiếm cách tích hợp với các mô hình hiện tại. Những vấn đề mà tôi muốn nhấn mạnh trong phần này là những thách thức mà các mô hình hiện tại vẫn chưa giải quyết tốt, bao gồm việc thực hiện nhiều bước trên các tập hợp đầu vào, đọc và ghi nhớ thông tin trong thời gian dài, cũng như lọc thông tin từ các đoạn văn bản lớn để tìm ra thông tin liên quan.

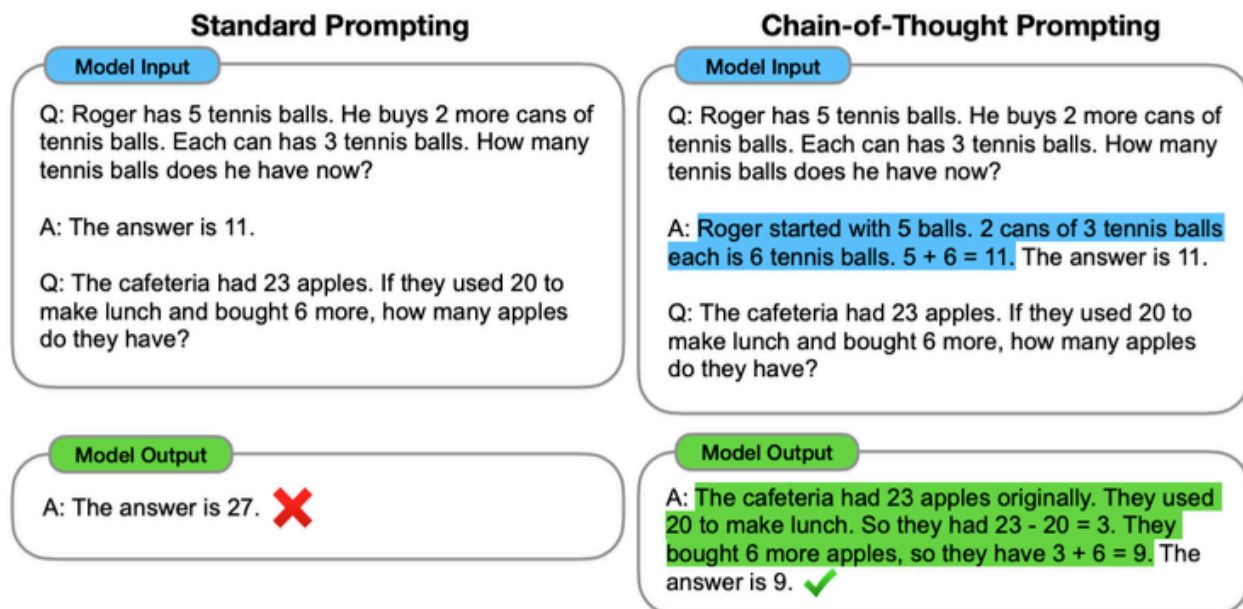
## Chain of Thought and Variants

### Review: Chain-of-thought Prompting (Wei et al. 2022)

Chúng ta nhắc lại Chain-of-thought là một phương pháp khác biệt so với cách prompting thông thường, nơi mà chúng ta chỉ có một câu hỏi và câu trả lời.

Cụ thể, trong Chain-of-thought, chúng ta bắt đầu với một câu hỏi, sau đó là quá trình suy diễn để đi đến câu trả lời. Ví dụ, giả sử Roger có năm quả bóng và hai cái thùng, mỗi thùng chứa ba

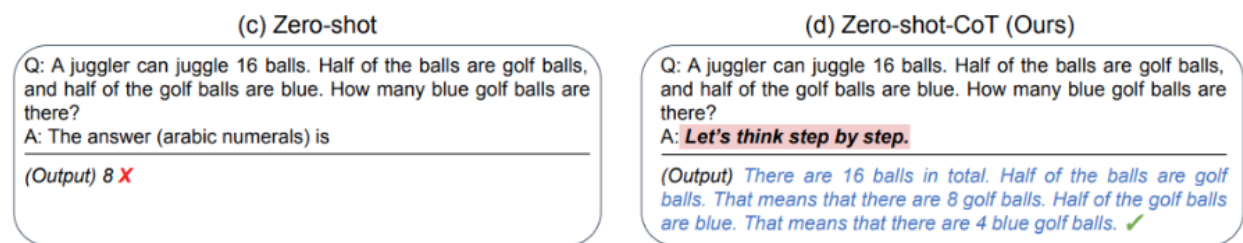
quả bóng tennis. Để tính tổng số quả bóng tennis, ta có thể thực hiện phép tính: 5 (quả bóng) + 6 (quả bóng tennis) = 11. Như vậy, câu trả lời là 11.



Bằng cách thêm quá trình suy diễn này vào gợi ý, chúng ta có thể khuyến khích mô hình thực hiện các bước suy diễn tương tự trong quá trình kiểm tra. Điều này đã chứng minh là cải thiện đáng kể hiệu suất của một số nhiệm vụ, đặc biệt là những nhiệm vụ mà chúng ta không thể dự đoán ngay câu trả lời một cách trực tiếp.

## Review: Zero-shot Chain of Thought (Kojima et al. 2022)

Trong bài viết trước, tôi đã đề cập đến khái niệm "zero shot Chain of Thought reasoning". Đây là một phương pháp mà chúng ta chỉ cần cung cấp cho mô hình một gợi ý như "let's think step by step". Khi đó, mô hình sẽ có khả năng thực hiện quá trình suy luận theo chuỗi một cách hiệu quả.



## Self Ask (Press et al. 2022)

Trong phần này, chúng ta sẽ khám phá một số phương pháp nâng cao mà các nhà nghiên cứu đang sử dụng để cải thiện khả năng suy luận của các mô hình ngôn ngữ lớn.

Một trong những phương pháp đáng chú ý là "self-ask". Một trong những vấn đề hiện tại với các mô hình ngôn ngữ lớn là chúng không giỏi trong việc đặt câu hỏi tiếp theo. Thực tế, chúng không được huấn luyện để làm điều này. Ví dụ, khi bạn tương tác với ChatGPT, tôi chưa bao

giờ thấy nó đặt câu hỏi tiếp theo. Điều này không phải vì các mô hình ngôn ngữ không có khả năng, mà có thể OpenAI cho rằng việc này sẽ tạo ra trải nghiệm người dùng không tốt.

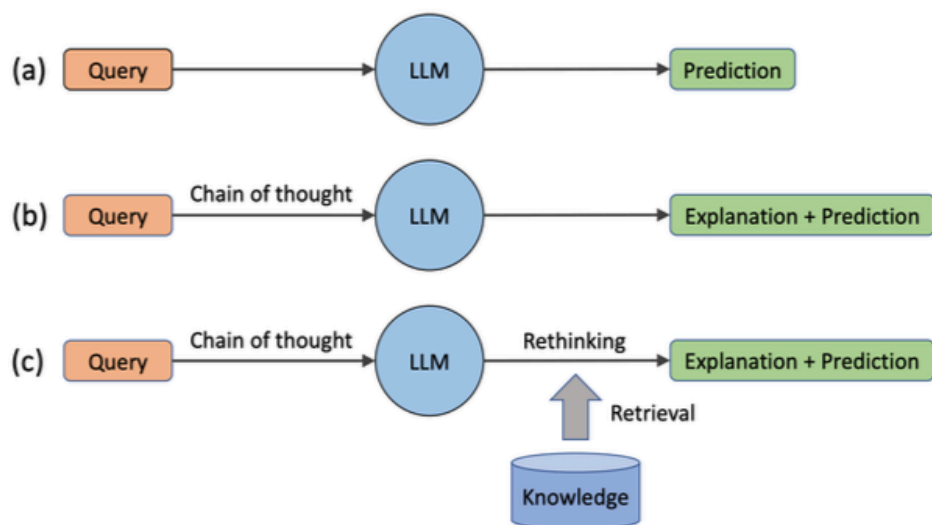
Phương pháp "self-ask" hoạt động bằng cách yêu cầu mô hình tự đặt câu hỏi xem có cần câu hỏi tiếp theo hay không. Ví dụ, nếu câu hỏi là "Ai sống lâu hơn, Theodore Haecker hay Harry Von Watkins?", mô hình sẽ tự hỏi liệu có cần câu hỏi tiếp theo không. Nếu câu trả lời là "Có", nó sẽ hỏi "Theodore Haecker bao nhiêu tuổi khi ông qua đời?" và cung cấp câu trả lời trung gian. Cuối cùng, mô hình sẽ đưa ra câu trả lời chính xác.

Phương pháp này chỉ là một biến thể của Chain of Thought, không sử dụng thông tin bên ngoài mà chỉ nhằm khai thác thông tin từ mô hình một cách trực tiếp. Tuy nhiên, nghiên cứu cho thấy phương pháp này rất hữu ích. Ngoài ra, còn có những phương pháp khác có khả năng tìm kiếm thông tin cụ thể để trả lời các câu hỏi, mạnh mẽ hơn so với những gì chúng ta đã thảo luận ở đây.

### Chain of Thought with Retrieval (He et al. 2023)

Ý tưởng chính ở đây là thay vì chỉ thực hiện Chain of Thought đơn thuần, phương pháp này sẽ truy xuất các câu liên quan khi thực hiện Chain of Thought.

Cụ thể, nếu mô hình không biết một người đã bao nhiêu tuổi khi qua đời, nó sẽ không thể trả lời câu hỏi đó. Để khắc phục vấn đề này, các nhà nghiên cứu đã áp dụng phương pháp truy xuất dựa trên BM25 từ Wikipedia cho mỗi câu trả lời trong Chain of Thought. Họ sẽ lấy khoảng 10 tài liệu liên quan và sử dụng những tài liệu này để hướng dẫn mô hình tiếp tục phát triển Chain of Thought của mình.



### Multilingual Chain of Thought Reasoning (Shi et al. 2022)

Chúng ta thường phải đưa ra quyết định thiết kế: liệu có nên trả lời câu hỏi bằng ngôn ngữ mà câu hỏi được đặt ra hay không? Ví dụ, nếu tôi đặt câu hỏi bằng tiếng Nhật, liệu mô hình có nên

thực hiện toàn bộ quá trình suy nghĩ bằng tiếng Nhật và trả lời bằng tiếng Nhật, hay nên chuyển sang tiếng Anh, ngôn ngữ mà mô hình đã được huấn luyện nhiều hơn?

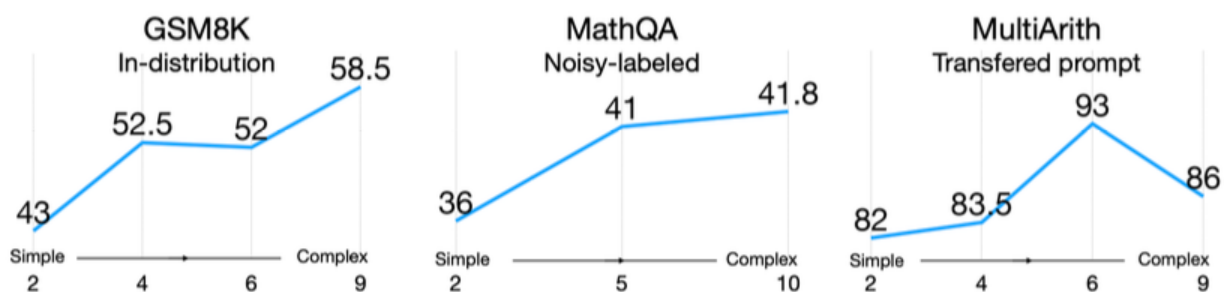
Nghiên cứu cho thấy rằng việc thực hiện Chain of Thought bằng tiếng Anh thường mang lại kết quả tốt hơn. Mặc dù có thể có sự khác biệt tùy thuộc vào ngôn ngữ, nhưng hầu hết các ngôn ngữ được thử nghiệm đều cho thấy rằng việc sử dụng tiếng Anh giúp cải thiện khả năng suy luận của mô hình. Cụ thể, có thể đạt được mức tăng trung bình khoảng 7 điểm trong kết quả.

Khi thử nghiệm với tiếng Nhật, tôi nhận thấy rằng mô hình dường như hoạt động thông minh hơn khi sử dụng tiếng Anh. Điều này cho thấy rằng khả năng trí tuệ không nên bị ràng buộc bởi ngôn ngữ mà câu hỏi được đặt ra. Cuối cùng, việc đánh giá kết quả có thể bị ảnh hưởng bởi cách dịch ngược lại ngôn ngữ gốc, nhưng điều này không làm giảm giá trị của những phát hiện này.

### Complex-based Prompting (Fu et al. 2022)

Trong nghiên cứu gần đây, một điều thú vị đã được phát hiện liên quan đến mối quan hệ giữa chất lượng giải thích và độ chính xác của dự đoán. Cụ thể, nếu giải thích của bạn sai, dự đoán cuối cùng cũng có xu hướng sai theo. Điều này có nghĩa là những sai sót trong các bước trung gian của giải thích có thể làm sai lệch kết quả cuối cùng.

Một trong những cách mà các nhà nghiên cứu đã tìm ra để cải thiện chất lượng giải thích là thông qua việc kéo dài độ dài của các giải thích. Họ nhận thấy rằng nếu giải thích có nhiều bước lý luận hơn, độ chính xác của dự đoán cũng tăng lên. Một nghiên cứu đã chỉ ra rằng, khi so sánh giữa một "simple reasoning chain" và một "more complex reasoning chain", độ chính xác có thể tăng khoảng 15% cho cùng một vấn đề. Điều đáng chú ý là các Chain of Thought này là tự nhiên, không phải là kết quả của việc huấn luyện mô hình để tạo ra các Chain of Thought dài hơn.



Để cải thiện độ chính xác, các nhà nghiên cứu đã áp dụng phương pháp lấy mẫu nhiều reasoning path và thực hiện "self consistency" trên các reasoning path dài hơn. "Self consistency" ở đây có nghĩa là thực hiện bỏ phiếu theo đa số cho các câu trả lời từ nhiều reasoning path khác nhau. Họ đã loại bỏ những path có chất lượng thấp hơn, từ đó nâng cao độ chính xác tổng thể.



# Systematics Studies of Reasoning in LLMs

## Reasoning is an “Emergent” Ability (Wei et al. 2022)

Trong nghiên cứu về khả năng suy luận của các mô hình ngôn ngữ lớn, một trong những kết quả quan trọng là khả năng "Chain of Thought". Khả năng này được coi là một "emergent ability", có nghĩa là nó tăng lên đáng kể khi kích thước mô hình đạt đến một mức nhất định.

Nhiều người đã bối rối về việc tại sao một mô hình lớn lại có thể cải thiện hiệu suất một cách đột ngột. Thực tế, điều này không phải là phép màu mà là một hiện tượng có thể giải thích. Một nghiên cứu năm 2023 đã làm rõ vấn đề này, cho thấy rằng khả năng xuất hiện chủ yếu phụ thuộc vào cách đo lường độ chính xác của mô hình.

Khi mô hình ngày càng lớn, độ chính xác trong việc dự đoán "reasonable next token" cũng tăng lên. Ví dụ, nếu một mô hình có độ chính xác dự đoán token là 75%, thì khả năng để dự đoán đúng một chuỗi các token sẽ tăng lên rất nhiều. Nếu có khoảng 5-8 bước có thể xảy ra sai sót trong quá trình suy luận, việc dự đoán đúng từng bước sẽ trở nên quan trọng hơn.

Khi chúng ta exponentiate (lũy thừa) các độ chính xác này lên 5, kết quả cho thấy rằng nếu độ chính xác của token tiếp theo thấp, khả năng dự đoán đúng chuỗi sẽ gần như bằng không cho đến khi độ chính xác đạt khoảng 75%. Điều này cho thấy rằng khả năng xuất hiện không chỉ liên quan đến việc dự đoán một token mà còn đến cách mà xác suất của token đúng tăng lên.

Để thực hiện các thí nghiệm thú vị về suy luận trên các mô hình nhỏ hơn, không chỉ nên đo lường độ chính xác mà còn nên xem xét "log likelihood" (xác suất log) của các chuỗi suy luận. Điều này sẽ giúp thấy được sự cải thiện một cách rõ ràng hơn.

## Reliability of Explanations (Ye and Durrett 2022)

Có hai cách tiếp cận để giải quyết vấn đề: một là đưa ra giải thích trước và sau đó dự đoán câu trả lời, hai là dự đoán câu trả lời trước và sau đó mới đưa ra giải thích. Theo kinh nghiệm, nếu bạn sử dụng một mô hình mạnh mẽ, như các mô hình tiên tiến hiện nay, thì việc đưa ra giải thích trước và sau đó mới dự đoán câu trả lời sẽ mang lại kết quả tốt hơn.

Nguyên nhân là do phương pháp "Chain of Thought" giúp mô hình phân tích câu hỏi thành những phần đơn giản hơn, từ đó dễ dàng hơn trong việc đưa ra câu trả lời, đặc biệt là trong các bài toán liên quan đến lý thuyết toán học. Ví dụ, mô hình text-davinci-002, được coi là tiên tiến vào thời điểm viết bài này, đã cho thấy sự cải thiện khoảng 5 điểm phần trăm khi sử dụng phương pháp giải thích trước và dự đoán sau.



		Acc	Fac	Con	Acc=Fac	Acc=Con
<i>reliability of explanations generated by InstructGPT</i>						
InstructGPT	SYNTH (E-P)	58.4	72.8	64.8	66.5	68.8
	SYNTH (P-E)	54.8	51.6	95.2	<b>89.6</b>	57.2
	ADVHP (E-P)	62.0	79.6	91.2	<b>80.0</b>	68.4
	ADVHP (P-E)	54.0	69.2	82.0	<b>77.6</b>	67.2
	E-SNLI (P-E)	62.0	—	98.8	—	62.0
<i>reliability of explanations generated by other LLMs on SYNTH</i>						
OPT (175B)	SYNTH (E-P)	30.0	77.2	47.2	45.6	58.8
	SYNTH (P-E)	39.6	64.0	81.2	<b>69.2</b>	49.6
GPT-3	SYNTH (E-P)	46.8	59.2	64.8	<b>66.8</b>	61.2
	SYNTH (P-E)	52.4	52.4	83.2	<b>78.4</b>	58.0
text-davinci-002	SYNTH (E-P)	86.0	91.6	85.2	<b>91.2</b>	84.8
	SYNTH (P-E)	81.6	83.2	96.4	<b>95.8</b>	82.8

Tuy nhiên, đối với các mô hình yếu hơn, như GPT-3 không được huấn luyện cho phương pháp Chain of Thought hoặc mô hình OPT, kết quả không giống như vậy. Hiện nay, hầu hết các mô hình đều cho thấy rằng việc đưa ra giải thích trước và sau đó mới dự đoán sẽ mang lại hiệu quả tốt hơn.

Trong bài báo này, các tác giả đã không chỉ đo lường độ chính xác của câu trả lời thông qua "chain of thoughts" mà còn xem xét tính xác thực của lời giải thích. Họ đánh giá xem lời giải thích có phù hợp với quá trình suy diễn hay không, cũng như tính nhất quán giữa câu trả lời và lời giải thích để xác định xem chúng có khớp với nhau hay không.

Nghiên cứu này sử dụng một số tập dữ liệu tổng hợp, cho phép đo lường các bước suy luận bằng toán học. Kết quả cho thấy, khi câu trả lời và lời giải thích nhất quán, đặc biệt là đối với các mô hình mạnh hơn, thì độ chính xác của dự đoán cũng cao hơn. Cụ thể, nếu lời giải thích có tính xác thực cao, điều này sẽ dẫn đến độ chính xác cao hơn của dự đoán thực tế.

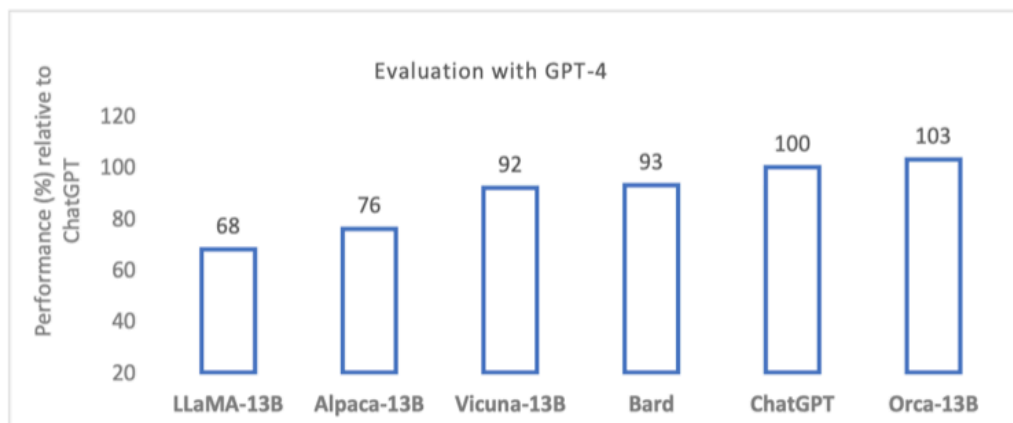
## Training for Chain of Thought

### ORCA: Training Small Models for Reasoning (Mukherjee et al. 2024)

Hiện nay, có hai phương pháp chính đào tạo cho Chain of Thought mà mọi người thường sử dụng. Phương pháp đầu tiên là tạo ra một lượng lớn dữ liệu tổng hợp đại diện cho các Chain of Thought và sử dụng chúng để đào tạo các mô hình. Đây là phiên bản nghiên cứu nổi tiếng nhất.

Cụ thể, họ đã tạo ra một tập dữ liệu Chain of Thought lớn và đa dạng từ GPT-3.5 và GPT-4, bao gồm 5 triệu hướng dẫn phức tạp. Trong đó, khoảng 1 triệu hướng dẫn được tạo ra từ GPT-4 và 4 triệu từ GPT-3.5, do việc tạo ra các chuỗi dài từ GPT-4 tốn kém. Kết quả là họ đạt được độ chính xác cao trong các nhiệm vụ liên quan đến Chain of Thought so với các tập dữ

liệu khác như Alpaca, vốn nhỏ hơn và không có nhiều chuỗi suy nghĩ, hay Vicuna, cũng tương tự như vậy.



### Chain of Thought Reward Models (Lightman et al. 2023)

Trong một nghiên cứu thú vị, các tác giả đã phát triển một phương pháp tự động đánh giá chất lượng của các “chains of thought”. Họ thu thập phản hồi từ con người cho từng bước trong quá trình suy diễn. Cụ thể, họ hỏi người dùng xem từng bước có tốt hay không; nếu có, họ sẽ nhận được một biểu tượng mặt cười, còn nếu không, sẽ là mặt buồn. Dữ liệu này được sử dụng để huấn luyện một mô hình phần thưởng (reward model) nhằm dự đoán chất lượng của từng bước trong “chains of thought”.

Mô hình này có khả năng xác định các bước sai và từ đó ảnh hưởng đến kết quả cuối cùng. Điều đặc biệt là mô hình không cần câu trả lời đúng để huấn luyện, cho phép nó được áp dụng cho nhiều câu hỏi khác nhau. Lý do cho sự hiệu quả này là “chains of thought” giúp dễ dàng hơn trong việc tạo ra từng bước và đánh giá tính chính xác của từng bước so với việc đánh giá câu trả lời tổng thể.

Một điểm quan trọng là nếu một bước thất bại, điều đó có thể dẫn đến việc các bước sau cũng không chính xác. Tuy nhiên, theo thông tin từ nghiên cứu, các bước được đánh giá độc lập, và việc thất bại ở một bước không nhất thiết có nghĩa là bước đó sai, có thể nó chỉ không được sử dụng.

Sau khi huấn luyện mô hình thưởng, các tác giả đã áp dụng mô hình này để huấn luyện các mô hình ngôn ngữ lớn theo phương pháp “chain of thought”. Quá trình này diễn ra như sau: đầu tiên, LLMs sẽ sinh ra một chuỗi các bước suy diễn cho một câu hỏi cụ thể. Sau đó, mô hình phần thưởng sẽ được sử dụng để đánh giá chất lượng của từng bước trong chuỗi. Những bước được đánh giá cao sẽ được “tăng cường” (upweight) trong quá trình huấn luyện, giúp mô hình học hỏi từ những phản hồi tích cực và cải thiện khả năng suy diễn của mình.

# Abductive Reasoning: Learning Patterns from Data

## Inference to Explanations

Trong phần này, chúng ta sẽ cùng tìm hiểu về lý thuyết lập luận suy diễn, hay còn gọi là suy diễn từ dữ liệu. Ý tưởng cơ bản là liệu chúng ta có thể tìm ra quy tắc nào đó giải thích cho một mẫu dữ liệu nhất định hay không.

Ví dụ, nếu chúng ta có một số thí nghiệm với các hình khối như hình trụ và hình vuông đặt trên một khối màu hồng, chúng ta nhận thấy rằng khi đặt cả hình trụ và hình khối lập phương lên khối hồng, sẽ phát ra tiếng động, nhưng nếu chỉ đặt hình trụ, thì không có tiếng động nào. Từ những quan sát này, chúng ta muốn khám phá các quy tắc tiềm ẩn giải thích cho những gì chúng ta đã thấy.

Có một số lý do tại sao việc tìm ra những quy tắc này lại quan trọng. Đầu tiên, chúng ta có thể muốn tạo ra những giải thích dễ hiểu cho con người. Khi có thể chỉ ra rằng một mẫu quy tắc nào đó tồn tại trong dữ liệu, chúng ta có thể giúp con người phân tích và hiểu rõ hơn về dữ liệu đó.

Gần đây, có một sự quan tâm lớn đến việc sử dụng các mô hình ngôn ngữ lớn trong nghiên cứu khoa học, nhằm đưa ra những giải thích hợp lý cho dữ liệu. Nếu chúng ta có thể làm được điều này, nó sẽ rất thú vị.

Thứ hai, các mô hình ngôn ngữ hiện tại không thực sự tốt trong việc duy trì tính nhất quán khi thực hiện các nhiệm vụ khó khăn trên một số lượng lớn ví dụ. Nếu chúng ta có thể xem xét tất cả dữ liệu cùng một lúc, suy diễn ra các quy tắc tổng quát từ chúng, và sau đó áp dụng những quy tắc này để dự đoán các ví dụ mới, chúng ta có thể nâng cao độ chính xác tổng thể.

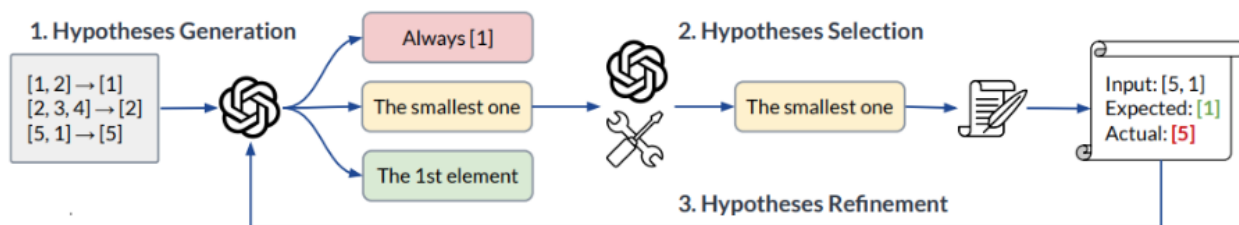
Điều này cũng tương tự như cách mà con người học hỏi. Chúng ta không chỉ đơn thuần ghi nhớ từng ví dụ; thay vào đó, chúng ta thường trừu tượng hóa các quy tắc tổng quát từ một vài ví dụ để có thể áp dụng cho các trường hợp mới.

Cuối cùng, khái niệm này cũng liên quan đến việc suy diễn chương trình từ các ví dụ đầu vào và đầu ra. Tuy nhiên, suy diễn từ dữ liệu có thể tổng quát hơn, không chỉ giới hạn ở việc tạo ra chương trình mà còn có thể là ngữ pháp, giải thích, hoặc bất kỳ điều gì khác tương tự.

## Rule Induction with LLMs

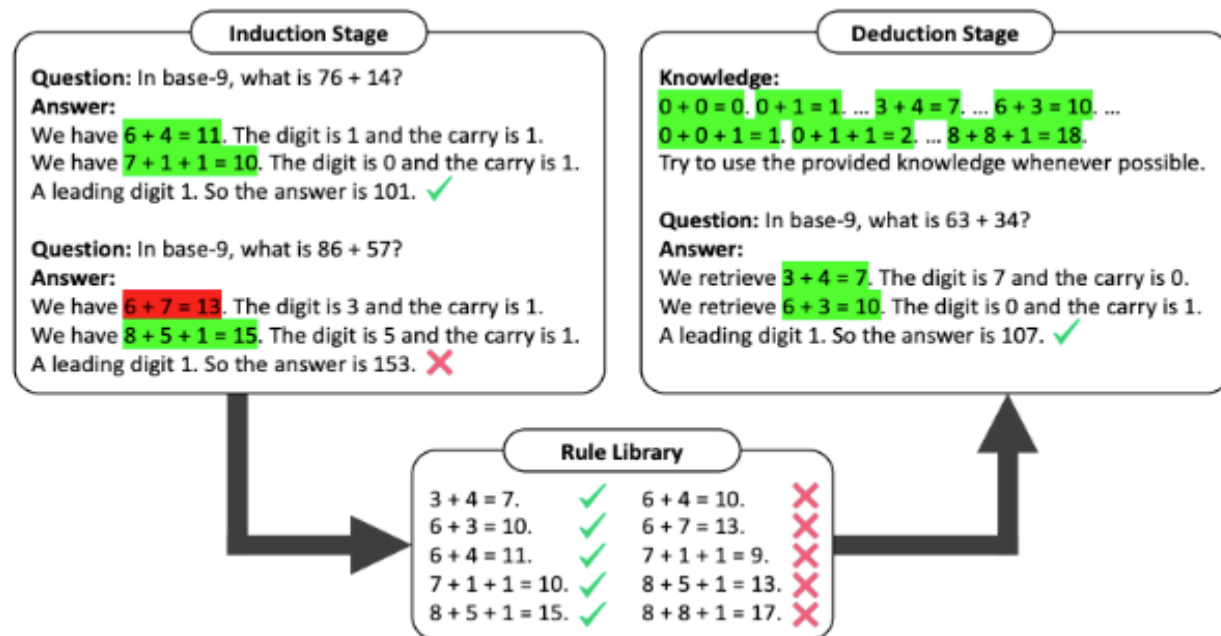
Gần đây, có một số nghiên cứu thú vị về suy luận quy tắc với các mô hình ngôn ngữ lớn. Một trong những bước đầu tiên của nghiên cứu này là tạo ra giả thuyết. Quá trình này bắt đầu bằng việc lấy các ví dụ đầu vào và đầu ra, từ đó dự đoán các quy tắc như "câu trả lời luôn là một", "chọn số nhỏ nhất", hoặc "chọn phần tử đầu tiên". Sau đó, các quy tắc này được đánh giá bằng cách sử dụng một mô hình ngôn ngữ khác hoặc một bộ đánh giá biểu tượng.

Nếu là chương trình, có thể sử dụng bộ đánh giá biểu tượng, nếu là mô hình ngôn ngữ, có thể yêu cầu mô hình chọn câu trả lời. Kết quả đầu ra được so sánh với kết quả mong đợi để xác minh tính đúng đắn của quy tắc. Sau đó, quá trình tinh chỉnh giả thuyết diễn ra, có thể bao gồm việc phản hồi về những điểm sai và dần dần cải thiện giả thuyết.



Một biến thể khác của ý tưởng này sử dụng phương pháp luận khác, nhưng cả hai đều có giá trị. Phương pháp này sử dụng các giả thuyết trong suy luận theo “chains of thought” và giữ lại những giả thuyết cho kết quả đúng.

Ví dụ, trong một bài toán cộng số trong hệ cơ số 9, mô hình ngôn ngữ đã đưa ra câu trả lời đúng và từ đó trích xuất các quy tắc như “ $6 + 4 = 11$ ” và “ $7 + 1 + 1 = 10$ ”. Các quy tắc này được thêm vào thư viện quy tắc. Nếu câu trả lời sai, quy tắc sẽ không được thêm vào hoặc được thêm vào như một ví dụ tiêu cực.



Trong bước suy luận suy diễn cuối cùng, các quy tắc này được sử dụng để cải thiện độ chính xác. Nghiên cứu gần đây cho thấy mô hình ngôn ngữ mạnh như GPT-4 có thể tự xác minh giả thuyết của mình, loại bỏ nhu cầu về bộ đánh giá biểu tượng.

Mặc dù các ứng dụng hiện tại chủ yếu là các ví dụ đơn giản, nhưng tiềm năng của việc trích xuất các mẫu tổng quát hơn để cải thiện hệ thống dựa trên mô hình ngôn ngữ là rất lớn. Đây là một hướng nghiên cứu đầy hứa hẹn.

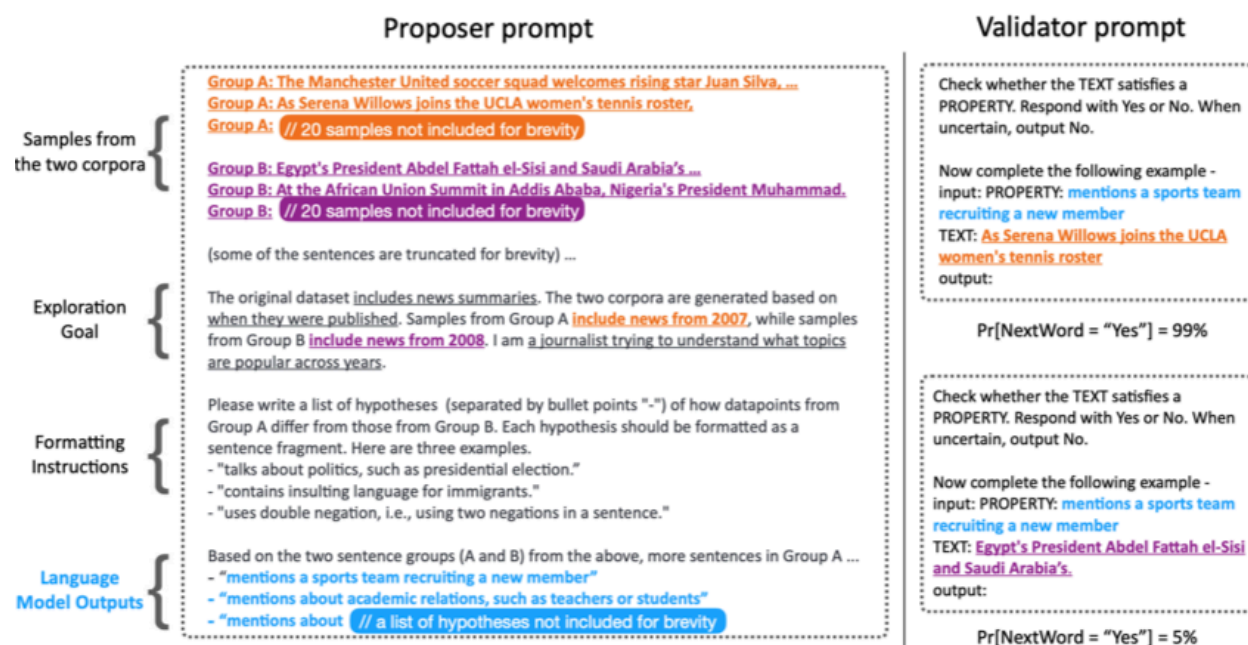
Học công cụ có thể được coi là một phần của quá trình này. Trong học công cụ, các chức năng được học không nhất thiết phải là giải thích tốt nhất cho dữ liệu, nhưng ít nhất là các quy tắc hữu ích để giải quyết nhiệm vụ. Phương pháp này có thể tương tự như bước xác minh, nơi các công cụ không hữu ích bị loại bỏ để tạo ra một tập hợp quy tắc ngắn gọn hơn.

Liệu chúng ta có thể sử dụng công cụ để giải quyết các vấn đề suy luận khái niệm phức tạp hơn không? Đây là một câu hỏi mở và cần được nghiên cứu thêm.

## Learning Differences between Text Collections (Zhong et al. 2023)

Một nghiên cứu thú vị mà tôi muốn giới thiệu liên quan đến việc học sự khác biệt giữa các tập hợp văn bản. Ý tưởng chính là tự động trích xuất sự khác biệt giữa các văn bản từ những người dùng thuốc thực sự và những người dùng giả dược. Đây là một phần quan trọng trong các thử nghiệm y tế, nơi các bác sĩ cố gắng tìm ra sự khác biệt nhất quán giữa hai nhóm này.

Phương pháp được sử dụng trong nghiên cứu này bao gồm việc phân loại các văn bản thành hai nhóm dựa trên thời gian xuất bản. Ví dụ, nhóm A có thể bao gồm các tin tức từ năm 2007, trong khi nhóm B là từ năm 2008. Sau đó, mô hình ngôn ngữ được sử dụng để đưa ra các giả thuyết về sự khác biệt giữa hai nhóm này, chẳng hạn như nhóm A có thể đề cập đến các đội thể thao nhiều hơn.



Tuy nhiên, có một số hạn chế khi sử dụng mô hình ngôn ngữ. Thứ nhất, mô hình có thể tạo ra thông tin không chính xác. Thứ hai, kích thước ngữ cảnh mà mô hình có thể xử lý là khá nhỏ.

Để khắc phục điều này, nghiên cứu đã mở rộng tập dữ liệu và sử dụng các giả thuyết như một bộ phân loại để kiểm tra trên các ví dụ từ hai tập hợp văn bản.

Sau khi có được các kết quả phân loại, bước tiếp theo là thực hiện kiểm định ý nghĩa thống kê để xác định sự khác biệt thực sự giữa hai tập hợp dữ liệu. Các giả thuyết được sắp xếp theo giá trị ý nghĩa, và những giả thuyết có giá trị P thấp nhất được coi là có khả năng là sự khác biệt thực sự.

Một thử nghiệm áp dụng phương pháp này để tìm sự khác biệt giữa các câu mà mô hình ngôn ngữ phù hợp tốt với tín hiệu não của con người và những câu không phù hợp. Kết quả cho thấy mô hình ngôn ngữ ít phù hợp hơn với các câu có ngôn ngữ ẩn dụ hoặc liên quan đến quan hệ giữa các cá nhân.

Tóm lại, việc sử dụng mô hình ngôn ngữ để trả lời các câu hỏi nghiên cứu phức tạp là một hướng đi thú vị, giúp giảm bớt công việc chú thích dữ liệu thủ công. Tuy nhiên, cần thận trọng và kết hợp với các phương pháp khác để đảm bảo độ tin cậy của kết quả.

## Resources

1. <https://phontron.com/class/anlp2024/lectures/#complex-reasoning-april-02>
2. [Memory Networks \(Sukhbaatar et al. 2015\)](#)
3. [Chain-of-Thought Prompting Elicits Reasoning in Large Language Models \(Wei et al. 2022\)](#)
4. [Let's think Step by Step \(Kojima et al. 2022\)](#)
5. [Self Ask \(Press et al. 2022\)](#)
6. [Chain of Thought with Retrieval \(He et al. 2023\)](#)
7. [Multilingual Chain of Thought \(Shi et al. 2022\)](#)
8. [Complexity-based Prompting \(Fu et al. 2022\)](#)
9. [Chain-of-Thought Prompting Elicits Reasoning in Large Language Models \(Wei et al. 2022\)](#)
10. [Reliability of Explanations \(Ye and Durrett 2022\)](#)
11. [ORCA \(Mukherjee et al. 2023\)](#)
12. [Let's Verify Step-by-step \(Lightman et al. 2023\)](#)
13. [Rule Inference with LLMs \(Qiu et al. 2023\)](#)
14. [Goal-driven Discovery of Distributional Differences \(Zhong et al. 2023\)](#)
15. [LLMs can learn rules \(Zhu et al. 2023\)](#)