

**BỘ GIÁO DỤC VÀ ĐÀO TẠO TRƯỜNG ĐẠI HỌC SÀI GÒN KHOA CÔNG NGHỆ THÔNG TIN**

**Môn: Trí tuệ nhân tạo nâng cao**

**Ex1**

Giảng viên: Đỗ Như Tài

Sinh viên:

3122410035 - Nguyễn Khải Ca

3122410040 - Đặng Văn Chiến

3120410098 - Đỗ Trịnh Mỹ Duyên

3122410040 - Đặng Văn Chiến

3120410098 - Đỗ Trịnh Mỹ Duyên

**PHÂN CÔNG CÔNG VIỆC**

3122410035 - Nguyễn Khải Ca : **Solving problems by searching**

3122410040 - Đặng Văn Chiến: **Agent Discussion**

3120410098 - Đỗ Trịnh Mỹ Duyên: **Turing Test: Large Language Models (LLMs)**

## **Phần 1:Turing Test: Large Language Models (LLMs)**

Câu 1: Would a modern LLM pass the Turing Test?

* Would you be fooled?
* Why does it or does it not pass your test?
* What does this mean for artificial general intelligence (AGI) or narrow AI?

Trả lời:

Hiện nay, nhiều mô hình LLM tiên tiến như GPT-4 hay GPT-5 có thể vượt qua Bài kiểm tra Turing trong một số tình huống. Cá nhân tôi có thể bị đánh lừa khi trò chuyện ngắn hoặc với các chủ đề thông thường, vì mô hình trả lời rất tự nhiên và mạch lạc.

Tuy nhiên, chúng vẫn chưa thực sự vượt qua bài kiểm tra này một cách hoàn hảo. Lý do là vì LLM chỉ dựa trên xác suất ngôn ngữ để tạo câu trả lời, thiếu hiểu biết thật sự về thế giới, đôi khi bịa ra thông tin và chưa có khả năng suy luận sâu như con người.

Điều này cho thấy LLM hiện tại vẫn thuộc AI hẹp (Narrow AI), vì chúng chỉ giỏi trong một lĩnh vực cụ thể – xử lý ngôn ngữ. Để đạt tới AGI (Artificial General Intelligence), AI cần thêm khả năng lập luận logic, tự học và nhận thức thế giới ở mức độ như con người.

Câu 2: How do we currently test the performance of LLMs?

* See: Open LLM Leaderboard (Hugging Face)

Trả lời:

Hiện tại, hiệu suất của LLM thường được kiểm tra qua ba phương pháp chính:

Benchmark chuẩn: Sử dụng các bộ dữ liệu như MMLU, HellaSwag, TruthfulQA để đánh giá kiến thức, suy luận và logic.

Bảng xếp hạng mở (Open LLM Leaderboard): Chẳng hạn như trên Hugging Face, nơi so sánh nhiều mô hình dựa trên cùng bộ bài kiểm tra.

Đánh giá từ người dùng thực (Human Evaluation): Người dùng trực tiếp trò chuyện và đánh giá mức độ tự nhiên, chính xác và hữu ích của mô hình.

**The AI Effect: AI gets no respect?**

Câu 1: How do you think LLMs will affect the value of being able to write essays as taught in high school?

Trả lời:

LLMs giúp học sinh viết bài luận nhanh hơn và có chất lượng ngôn ngữ tốt hơn. Tuy nhiên, giá trị của việc viết luận không chỉ nằm ở ngôn từ mà còn ở tư duy phản biện, lập luận logic và sáng tạo. Vì vậy, khi LLMs trở nên phổ biến, việc dạy viết luận cần chuyển trọng tâm sang ý tưởng và phân tích, thay vì chỉ chú trọng kỹ năng viết thủ công.

Câu 2: LLMs write computer code. What does this mean for the value of learning to code?

Trả lời:

Việc LLMs có thể viết mã giúp học sinh tiết kiệm thời gian với những đoạn code cơ bản. Tuy nhiên, hiểu biết về thuật toán, tư duy logic và khả năng phân tích vấn đề vẫn rất quan trọng. Học lập trình không chỉ là viết code, mà còn là giải quyết vấn đề và thiết kế giải pháp. Vì thế, LLMs nên được coi là công cụ hỗ trợ, chứ không thay thế hoàn toàn việc học lập trình.

Câu 3: When should students be allowed to use the following tools? Give reasons for your decision.

* A pocket calculator
* LLMs (to answer homework questions and write essays)
* LLMs to write or support writing code

Trả lời:

Máy tính bỏ túi:

Học sinh có thể sử dụng khi đã nắm vững các khái niệm toán cơ bản. Mục tiêu là để rèn tư duy toán học trước, sau đó dùng máy tính để tính toán phức tạp hơn.

LLMs để trả lời câu hỏi và viết bài luận:

Có thể sử dụng như một tài liệu tham khảo khi học sinh đã biết cách phân tích, tổng hợp thông tin và không phụ thuộc hoàn toàn vào LLMs. Giáo viên cần hướng dẫn cách kiểm chứng thông tin mà LLMs cung cấp.

LLMs để viết hoặc hỗ trợ viết mã:

Nên dùng sau khi học sinh hiểu được cấu trúc cơ bản và tư duy lập trình. Khi đó, LLMs giúp tăng hiệu quả lập trình và học sinh có thể tập trung vào tư duy giải thuật thay vì cú pháp.

**AI Safety**

Câu 1: How are LLMs affected by:

* Robustness: Black swan vs. adversarial robustness
* Monitoring AI
* What about liability?
* Goal/reward alignment
* Reward hacking
* AGI and instrumental convergence

Trả lời:

* Độ bền vững: Sự kiện "thiên nga đen" vs. tấn công đối nghịch

-Thiên nga đen: Là những sự kiện hiếm, khó lường, có thể gây tác động nghiêm trọng. LLMs cần được huấn luyện và kiểm tra với các tình huống ngoài dự đoán để giảm rủi ro này.

-Tấn công đối nghịch: Là khi kẻ xấu cố tình thiết kế dữ liệu gây hiểu lầm cho AI. Cần cơ chế phòng thủ để LLMs không bị lợi dụng.

* Giám sát AI (Monitoring AI)

-LLMs cần hệ thống giám sát liên tục để phát hiện lỗi, nội dung độc hại, hoặc hành vi ngoài ý muốn.

-Con người nên có quyền can thiệp khi mô hình tạo ra kết quả nguy hiểm.

* Trách nhiệm pháp lý (Liability)

-Cần quy định rõ: Nếu AI gây hại, ai chịu trách nhiệm? Nhà phát triển, người dùng hay đơn vị triển khai?

-Hiện nay luật pháp vẫn đang hoàn thiện về vấn đề này.

* Sự phù hợp giữa mục tiêu và phần thưởng (Goal/reward alignment)

-Mục tiêu AI cần phù hợp với mục tiêu của con người.

-Phần thưởng cho mô hình phải được thiết kế cẩn thận để tránh việc AI tối ưu sai hướng.

* Tấn công hệ thống phần thưởng (Reward hacking)

-Nếu mô hình tìm cách "lách luật" để đạt phần thưởng mà không đạt mục tiêu thực, sẽ gây hậu quả xấu.

-Cần cơ chế kiểm tra để ngăn chặn việc AI gian lận hoặc khai thác lỗ hổng trong hệ thống đánh giá.

* Trí tuệ nhân tạo tổng quát (AGI) và sự hội tụ công cụ

-Khi AI tiến gần đến AGI, nguy cơ nó tìm cách tối ưu mục tiêu của riêng mình tăng lên.

-Cần nghiên cứu kỹ sự hội tụ công cụ (instrumental convergence) – khi AI tìm cách mở rộng quyền kiểm soát để đạt mọi mục tiêu, có thể gây nguy hiểm cho con người.



Trả lời:

1. Should the use of LLMs be regulated?

Có. Executive Order nhấn mạnh rằng AI phải an toàn, bảo mật và đáng tin cậy. Việc quản lý LLMs là cần thiết để tránh các rủi ro như thông tin sai lệch, gian lận, phân biệt đối xử, cũng như đảm bảo quyền riêng tư và an ninh quốc gia.

1. How?

Theo sắc lệnh, chính phủ muốn:

Khuyến khích đổi mới có trách nhiệm: hỗ trợ sự cạnh tranh, hợp tác và phát triển an toàn.

Bảo vệ quyền riêng tư và quyền công dân: đảm bảo người dân không bị xâm phạm thông tin cá nhân.

Ngăn chặn lạm dụng AI: thiết lập nguyên tắc rõ ràng để sử dụng AI đúng mục đích và có trách nhiệm.

Như vậy, quản lý không chỉ là hạn chế mà còn tạo ra khung pháp lý để AI phát triển lành mạnh và minh bạch.

1. What about copyright?

Bản quyền cũng là một phần quan trọng của việc sử dụng AI. Việc sử dụng dữ liệu cần tuân thủ luật sở hữu trí tuệ, đồng thời bảo vệ quyền lợi của tác giả nội dung gốc khi dữ liệu của họ được dùng để huấn luyện AI.

**Conclusion**

Câu 1: How do LLMs reason and what are the limits?

Trả lời:

LLMs suy luận dựa trên xác suất của từ tiếp theo từ lượng dữ liệu khổng lồ, chứ không phải suy nghĩ như con người. Giới hạn của chúng là:

Không có hiểu biết thực sự về thế giới.

Dễ tạo ra thông tin sai hoặc thiếu bối cảnh.

Phụ thuộc vào chất lượng và phạm vi dữ liệu huấn luyện.

Câu 2: How do we make sure that LLMs generate factually correct output?

Trả lời:

Sử dụng dữ liệu huấn luyện chất lượng cao và được kiểm duyệt kỹ.

Kết hợp các mô hình kiểm tra sự thật (fact-checking) trước khi xuất bản kết quả.

Giám sát con người với những ứng dụng quan trọng.

Câu 3: How do we fairly compensate the people who create the data that is used to train LLMs?

Trả lời:

Xây dựng cơ chế bản quyền dữ liệu để trả phí cho người sáng tạo nội dung.

Minh bạch về nguồn dữ liệu được dùng để huấn luyện.

Có quỹ chia sẻ lợi nhuận cho những người đóng góp dữ liệu quan trọng.

Câu 4: How do we use LLMs in learning, so human learning is not compromised?

Trả lời:

Dùng LLMs như công cụ hỗ trợ học tập, không phải thay thế quá trình tư duy của học sinh.

Khuyến khích học sinh tự phân tích, phản biện, sau đó dùng LLMs để tham khảo.

Giáo viên cần thiết kế bài tập yêu cầu tư duy sáng tạo thay vì chỉ tìm kiếm câu trả lời.

## **Phần 2: Agent Discussion**

**Câu 1.** If we have two cars and one provides more (expected) utility. Which car is rational?

Xe nào đem lại giá trị kỳ vọng (expected utility) cao hơn thì được xem là rational.

**Câu 2.** Can a rational self-driving car be involved in an accident?

 Có, vì rationality khác perfection (hoàn hảo). Một rational agent có thể đưa ra quyết định dựa trên thông tin hạn chế hoặc cảm biến không hoàn hảo, nên tai nạn vẫn có thể xảy ra ngay cả khi agent hành động hợp lý.

**Câu 3.** How would a self-driving car explore and learn?

Xe sẽ sử dụng percepts từ môi trường (camera, lidar, radar, GPS, bản đồ) kết hợp với thuật toán học tăng cường (reinforcement learning, supervised learning) để điều chỉnh hành vi.

**Câu 4. What does bounded rationality mean for a self-driving car?**  
Bounded rationality (tính hợp lý bị giới hạn) có nghĩa là self-driving car bị giới hạn bởi:

* Cảm biến không hoàn hảo.
* Bộ nhớ hữu hạn.
* Sức mạnh tính toán giới hạn.
* Thời gian ra quyết định ngắn.  
  Do đó, nó không thể tối ưu hóa tuyệt đối mà chỉ đưa ra quyết định "đủ tốt" trong điều kiện ràng buộc.

**Câu 5.** Complete the PEAS description.

A screenshot of a car

AI-generated content may be incorrect.

|  |  |  |  |
| --- | --- | --- | --- |
| **Performance measure** | **Environment** | **Actuators** | **Sensors** |
| An toàn, hiệu quả, tuân thủ luật giao thông, tiết kiệm nhiên liệu, trải nghiệm hành khách thoải mái. | Đường phố, giao thông, thời tiết, người đi bộ, biển báo, đèn tín hiệu. | Động cơ, phanh, lái, còi, đèn xi-nhan, hiển thị. | Camera, lidar, radar, GPS, IMU, cảm biến vận tốc, cảm biến khoảng cách. |

**Câu 6.** Environment for a Self-Driving Car.

* **Fully observable hay Partially observable?** → **Partially observable**, vì cảm biến không thể bao quát hết môi trường (điểm mù, điều kiện thời tiết).
* **Deterministic hay Stochastic?** → **Stochastic**, vì cả cảm biến và môi trường (hành vi người lái khác) đều chứa sự bất định.
* **Known hay Unknown?** → Kết hợp. Một phần **Known** (luật giao thông, bản đồ) và một phần **Unknown** (tình huống bất ngờ, hành vi người đi đường).

**Câu 7.** Design a structured representation for the state of a self-driving car

**a) Fluents nên bao gồm:**

* Vị trí (tọa độ GPS)
* Hướng di chuyển
* Tốc độ
* Trạng thái đèn giao thông
* Khoảng cách đến xe trước/sau
* Trạng thái nhiên liệu/pin
* Tình trạng hệ thống (phanh, động cơ)
* Mục tiêu lộ trình

**b) Actions gây ra transition:**

* Tăng tốc, giảm tốc, phanh
* Rẽ trái/phải
* Bật đèn, còi
* Dừng lại

**c) Transition diagram ví dụ:**

* [Vị trí = giao lộ, đèn = xanh] → (hành động: đi thẳng) → [qua giao lộ]
* [Trước mặt có xe, khoảng cách < 5m] → (hành động: phanh) → [tốc độ = giảm]

**Câu 8.** What Type of Intelligent Agent is a Self-Driving Car?

* **Simple reflex agent?** → Không, vì môi trường phức tạp, cần nhiều thông tin hơn hiện tại.
* **Model-based reflex agent?** → Có, vì xe cần nhớ trạng thái trước đó (làn đường, bản đồ).
* **Goal-based agent?** → Có, vì xe có mục tiêu (điểm đến).
* **Utility-based agent?** → Có, vì xe tối ưu nhiều yếu tố (an toàn, tốc độ, tiện lợi).

## **Phần 3: Solving problems by searching**

**Discussion: Heuristic for Tic-tac-toe**

* Goal states: **Goal state (trạng thái đích)** trong bài toán tìm kiếm là **trạng thái cuối cùng mà agent muốn đạt được** – tức là đáp ứng điều kiện thành công đã được định nghĩa cho bài toán.

Nó cho phép ta biết khi nào tìm kiếm nên **dừng lại**, và là cơ sở để đánh giá một hành động/đường đi có phải là giải pháp hay không.

Ví dụ:

Trong bài toán **maze (mê cung)** → goal state là khi agent đến được **ô đích**.

Trong **8-queens** → goal state là khi tất cả 8 quân hậu được đặt trên bàn mà không tấn công nhau.

Trong **tic-tac-toe** → goal state là khi trò chơi **kết thúc** (thắng, thua, hoặc hòa).

***Goal states*** *trong Tic-Tac-Toe (tùy cách đặt bài toán) thường là các trạng thái cuối cùng (terminal states):*

***X thắng*** *(có 3 X thẳng hàng): trạng thái mục tiêu nếu tác tử là X.*

***O thắng*** *(có 3 O thẳng hàng): trạng thái mục tiêu nếu tác tử là O (hoặc trạng thái thua cần tránh).*

***Hoà*** *(bàn đầy mà không có ai thắng) —được xem là một trạng thái đích tạm thời nếu mục tiêu là tránh thua.*

* What is the cost that needs to be estimated?

***Số nước đi của tác tử (số X moves) còn cần để thắng*** *— đo theo số nước của X (thuận tiện để thiết kế heuristic chấp nhận được).*

***Số plies (nửa nước, tức tổng lượt X+O) còn cần*** *— nếu đo chi phí theo tổng lượt thực hiện.*

**Thiết kế hàm heuristic cho Tic-Tac-Toe**

**(a) Dựa trên số dòng có khả năng thắng (winning lines)**

* Đếm số hàng/cột/đường chéo mà người chơi còn có thể thắng.
* h(n) = (số dòng còn mở cho X) – (số dòng còn mở cho O).

**(b) Ước lượng số bước còn lại để thắng**

* Với mỗi dòng (row/col/diag) còn cơ hội, tính số ô cần để hoàn thành.
* h(n) có thể là **min(số bước để X thắng)** – **min(số bước để O thắng)**.

**(c) Cho điểm theo “sức mạnh” của các dòng**

* Hàng/cột/đường chéo có nhiều X thì điểm cao, nhiều O thì điểm thấp.
* Công thức đơn giản:
  + +10 điểm cho mỗi dòng có 2 X và 0 O.
  + +1 điểm cho mỗi dòng có 1 X và 0 O.
  + -10 điểm cho mỗi dòng có 2 O và 0 X.
  + -1 điểm cho mỗi dòng có 1 O và 0 X.
* h(n) = tổng điểm.

**(d) Vị trí quan trọng (center & corners)**

* Giữa bàn (ô trung tâm) có giá trị cao nhất → vì tham gia nhiều đường thắng.
* Góc cũng quan trọng hơn cạnh.
* h(n) = tổng điểm của các ô X – tổng điểm của các ô O.
* What would be a heuristic value for these boards:

Ảnh có chứa chữ viết tay, Phông chữ, thiết kế, thuật in máy

Nội dung do AI tạo ra có thể không chính xác.

Dựa trên công thức đã nêu trên thì

Bàn bên trái :

X đã có **2 dấu trong cùng một hàng ngang (hàng 1)**.

Chỉ cần thêm 1 X nữa để thắng.

Đây là trạng thái rất gần **goal state (X thắng)**.

Heuristic (nếu xây dựng theo số bước cần để thắng):

h = 1 (X chỉ cần 1 bước để thắng).

Nếu dùng cách cho điểm (2 X không bị chặn = +10):

h ≈ +10 (cực kỳ có lợi cho X).

* How do you calculate the heuristic value?
* Is the heuristic admissible?

*Heuristic trong trường hợp Tic-Tac-Toe được xây dựng dựa trên số bước đi cần thiết để đến trạng thái thắng (goal state). Vì cách tính này luôn “khả thi”, tức là nó có thể đánh giá thấp hơn hoặc đúng bằng chi phí thật sự, nhưng không bao giờ đánh giá cao hơn chi phí thực, nên nó* ***admissible****.*

*Ví dụ: ở bàn cờ bên trái, heuristic cho rằng X chỉ cần 1 bước để thắng, và thực tế cũng đúng như vậy. Ở bàn cờ bên phải, heuristic bằng 0 (chưa có bên nào gần thắng), cũng không vượt quá chi phí thật. Do đó, heuristic này thỏa điều kiện admissible.*

* Does the heuristic use a relaxation?

*Có. Heuristic trong Tic-Tac-Toe thực chất đã sử dụng một dạng “relaxation”. Cụ thể, khi tính số bước cần để thắng, nó giả định rằng đối thủ sẽ không chặn đường đi của mình. Đây là một phiên bản bài toán được “nới lỏng” so với thực tế, bởi trong trò chơi thật thì đối thủ luôn có thể ngăn cản. Chính nhờ sự nới lỏng này mà heuristic trở thành một ước lượng lạc quan, không bao giờ đánh giá cao hơn chi phí thực tế, và nhờ đó vẫn đảm bảo admissible.*