BIS - MT · 13/06/2026

Bản Tin AI Hằng Ngày

Cập nhật công nghệ AI mới nhất

“To be yourself in a world that is constantly trying to make you something else is the greatest accomplishment.”

↪ Là chính mình trong một thế giới liên tục cố tạo ra bạn thành người khác là thành tựu vĩ đại nhất.

— Ralph Waldo Emerson

Dũng cảm sống theo giá trị và con người thật của mình, không bị cuốn theo kỳ vọng xã hội, là hình thức tự do và thành công cao nhất.

TIN TỨC NỔI BẬT

1

Ai cũng muốn thử sức với vibe coding — và Google cũng không ngoại lệ với Stitch, dự án tiếp nối Jules

Everyone's looking to get in on vibe coding — and Google is no different with Stitch, its follow-up to Jules

VentureBeat Đọc bài viết →

> Google đang tham gia xu hướng "vibe coding" với việc phát hành Stitch, dự án tiếp nối Jules. Vibe coding là việc sử dụng AI để tạo ra code có tính thẩm mỹ cao và dễ đọc, thường ưu tiên phong cách hơn là chức năng. Việc Google tham gia vào lĩnh vực này cho thấy sự quan tâm ngày càng tăng đối với phương pháp lập trình này. Stitch được thiết kế để hoạt động cùng với Jules, một dự án trước đây của Google nhằm tạo ra một codebase dễ đọc và dễ bảo trì hơn. Chi tiết về Stitch vẫn chưa được tiết lộ đầy đủ, nhưng việc phát hành nó cho thấy Google cam kết khám phá các khả năng của vibe coding. Khi ngành công nghệ tiếp tục phát triển, các công ty ngày càng tìm kiếm những cách để làm cho việc lập trình hiệu quả và thú vị hơn. Sự phát triển của các công cụ như Stitch và Jules phản ánh xu hướng này, và sẽ rất thú vị để xem các dự án này định hình tương lai của các phương pháp lập trình như thế nào.

2

Claude Code đấu Cursor 2026: Đạt 80.8% SWE-bench, 1M Context [Đã kiểm nghiệm]

Claude Code vs Cursor 2026: 80.8% SWE-bench, 1M Context [Tested]

tech-insider.org Đọc bài viết →

Trong một bài kiểm tra benchmark gần đây, Claude Code và Cursor 2026 đã được đưa ra so sánh để xác định khả năng hoạt động của chúng. Kết quả cho thấy Claude Code đạt được số điểm ấn tượng 80.8% trên SWE-bench, một benchmark được sử dụng rộng rãi để đánh giá các language model. Ngoài ra, Claude Code còn thể hiện khả năng xử lý các context phức tạp bằng cách xử lý thành công 1 triệu input context. Mức hiệu suất này cho thấy Claude Code đã có những bước tiến đáng kể trong quá trình phát triển, có khả năng định vị nó là một đối thủ hàng đầu trong lĩnh vực language model. Kết quả kiểm tra cung cấp những hiểu biết có giá trị về khả năng của Claude Code và Cursor 2026, mang đến cái nhìn thoáng qua về trạng thái hiện tại của công nghệ language model. Cần có thêm các thử nghiệm và đánh giá để hiểu đầy đủ ý nghĩa của những kết quả này và xác định các ứng dụng tiềm năng của các model này.

3

Giới thiệu Open Agent Specification (Agent Spec): Định dạng thống nhất cho AI Agent

Introducing the Open Agent Specification (Agent Spec): A Unified Representation for AI Agents

Oracle Blogs Đọc bài viết →

Oracle đã giới thiệu Open Agent Specification (Agent Spec), một định dạng thống nhất cho các AI agent. Agent Spec nhằm mục đích cung cấp một framework chung để các developer tạo, tích hợp và tương tác với các AI agent trên nhiều nền tảng và ứng dụng khác nhau. Specification này định nghĩa một cấu trúc tiêu chuẩn cho các AI agent, cho phép giao tiếp và cộng tác liền mạch giữa các agent và hệ thống khác nhau. Agent Spec được thiết kế linh hoạt và có thể mở rộng, cho phép các developer dễ dàng thêm các tính năng và khả năng mới khi cần. Nó cũng cung cấp một cách rõ ràng và nhất quán để mô tả hành vi, mục tiêu và sở thích của các AI agent, giúp dễ dàng hiểu và làm việc với chúng hơn. Bằng cách áp dụng Agent Spec, các developer có thể tạo ra các AI agent có khả năng tương tác và tái sử dụng cao hơn, giảm độ phức tạp và chi phí liên quan đến việc tích hợp nhiều agent và hệ thống. Specification này là mã nguồn mở và được cộng đồng phát triển, đảm bảo rằng nó vẫn linh hoạt và thích ứng với các nhu cầu phát triển AI đang thay đổi.

4 Các khóa học mới của OpenAI Academy cho kỷ nguyên làm việc tiếp theo

New OpenAI Academy courses for the next era of work

OpenAI Blog Đọc bài viết →

OpenAI đã ra mắt một loạt các khóa học mới của Academy, được thiết kế để trang bị cho các cá nhân những kỹ năng thực tế cần thiết cho kỷ nguyên làm việc tiếp theo. Các khóa học tập trung vào ba lĩnh vực chính: xây dựng kỹ năng AI thực tế, tạo ra các workflow có thể lặp lại và áp dụng agent vào công việc hàng ngày. Bằng cách tham gia các khóa học này, các cá nhân có thể tích lũy kinh nghiệm thực hành và phát triển chuyên môn cần thiết để tích hợp AI một cách hiệu quả vào cuộc sống nghề nghiệp của họ. Các khóa học của Academy là một nguồn tài nguyên quý giá cho bất kỳ ai muốn đi trước trong một môi trường làm việc đang phát triển nhanh chóng, nơi AI ngày càng đóng vai trò quan trọng. Với các khóa học này, OpenAI đặt mục tiêu trao quyền cho mọi người khai thác tiềm năng của AI và thúc đẩy đổi mới trong các lĩnh vực tương ứng của họ. Các khóa học là một phần trong nỗ lực không ngừng của OpenAI nhằm làm cho AI dễ tiếp cận và mang lại lợi ích hơn cho tất cả mọi người.

5 Nhân viên Meta cực kỳ không hài lòng với kế hoạch Hackathon AI toàn công ty của Zuckerberg

Meta Employees Absolutely Hate Zuckerberg's Plan for a Companywide AI Hackathon

Wired Đọc bài viết →

Các nhân viên Meta đang bày tỏ sự thất vọng và không hài lòng về thông báo của CEO Mark Zuckerberg về một cuộc thi hackathon AI toàn công ty, dự kiến diễn ra từ ngày 14 đến 16 tháng 7. Nhiều nhân viên cảm thấy họ có quá nhiều việc phải làm sau các đợt sa thải hàng loạt gần đây, khiến họ có ít thời gian để tham gia sự kiện. Những người khác thì nản lòng do tinh thần làm việc thấp và niềm tin vào ban quản lý suy giảm. Một số nhân viên đã chỉ ra rằng hackathon sẽ không được tính vào đánh giá hiệu suất, càng làm tăng thêm sự thất vọng. Sự kiện này được cho là nhằm xây dựng tình đồng nghiệp và thúc đẩy đổi mới, nhưng các nhân viên tỏ ra hoài nghi, viện dẫn những lo ngại về văn hóa và các ưu tiên của công ty. Hackathon là một trong số các sáng kiến mà Zuckerberg đã đưa ra để giải quyết những chỉ trích nội bộ và tái tạo năng lượng cho lực lượng lao động, bao gồm tăng ngân sách cho các buổi team offsite và loại bỏ hot desking ở một số văn phòng.

> Tuy nhiên, các nhân viên đang đặt câu hỏi về thời điểm và tính khả thi của hackathon, với khối lượng công việc hiện tại và sự bất ổn nội bộ.

6

Hình dạng, Đối xứng và Cấu trúc: Vai trò thay đổi của Toán học trong nghiên cứu Machine Learning

Shape, Symmetries, and Structure: The Changing Role of Mathematics in Machine Learning Research

The Gradient Đọc bài viết →

> Trong những năm gần đây, nghiên cứu machine

7

Cách chúng tôi giúp GitHub Copilot CLI chọn lọc hơn khi giao việc

How we made GitHub Copilot CLI more selective about delegation

GitHub Blog Đọc bài viết →

> GitHub đã thực hiện các cải tiến đối với Công cụ dòng lệnh Copilot của mình, một công cụ sử dụng trí tuệ nhân tạo (AI) để hỗ trợ các nhà phát triển (developer). Công ty nhằm mục đích làm cho công cụ này trở nên chọn lọc hơn về thời điểm nó ủy quyền nhiệm vụ cho các đại lý phụ (subagent). Ban đầu, Công cụ dòng lệnh Copilot thường ủy quyền nhiệm vụ một cách không cần thiết, dẫn đến sự gia tăng về khối lượng công việc phối hợp, cuộc gọi công cụ và thời gian chờ. Điều này có thể cản trở năng suất và tạo ra ma sát. Để giải quyết vấn đề này, GitHub đã giới thiệu "ủy quyền đại lý phụ thông minh hơn", giúp đại lý chính quyết định khi nào ủy quyền nhiệm vụ cho các đại lý phụ. Tính năng này hiện đã được triển khai cho tất cả lưu lượng sản xuất, cho phép các nhà phát triển cập nhật Công cụ dòng lệnh Copilot của GitHub để tận dụng cải tiến. Bằng cách chọn lọc hơn về ủy quyền, Công cụ dòng lệnh Copilot cập nhật có thể cung cấp trải nghiệm hiệu quả và năng suất hơn cho các nhà phát triển.

8

Kimi K2.7-Code giảm 30% token suy nghĩ — nhưng người dùng thực tế nói các benchmark không đáng tin cậy

Kimi K2.7-Code cuts thinking tokens 30% — but practitioners say the benchmarks don't check out

VentureBeat Đọc bài viết →

> Moonshot AI đã phát hành bản cập nhật mã nguồn mở cho dòng mô hình mã hóa K2, được gọi là Kimi K2.7-Code. Mô hình mới này tuyên bố có khả năng suy luận tinh gọn và tăng hiệu suất hàng chục phần trăm, bao gồm giảm 30% việc sử dụng token suy nghĩ so với người tiền nhiệm K2.6. Lợi ích hiệu suất này dự kiến sẽ ảnh hưởng trực tiếp đến chi phí suy luận cho các đội chạy luồng công việc đại lý. Tuy nhiên, các chuyên gia đã đặt ra câu hỏi về độ chính xác của điểm số chuẩn của Moonshot AI, vốn là độc quyền và có thể không phản ánh hiệu suất trong thế giới thực. Các chuẩn mực độc lập, chẳng hạn như DeepSWE, chưa được gửi và một số nhà nghiên cứu đã báo cáo kết quả hỗn hợp khi thử nghiệm K2.7-Code với các mô hình khác. Mặc dù vậy, Moonshot AI tuyên bố rằng K2.7-Code tạo ra sự khái quát hóa đáng tin cậy hơn trên các ngôn ngữ lập trình và loại nhiệm vụ khác nhau. Mô hình này có thể được triển khai thông qua vLLM hoặc SGLang và được phát hành theo giấy phép MIT sửa đổi, với trọng số có sẵn trên HuggingFace.

TIPS & TRICKS CHO DEV

Chain-of-Thought Prompting

Vấn đề: Model không hiểu được ngữ cảnh và logic của câu hỏi.

Cách làm: Sử dụng kỹ thuật chain-of-thought prompting, ví dụ: "Tôi muốn tìm hiểu về các loại động vật hoang dã, hãy giúp tôi tìm hiểu về các loại động vật hoang dã ở châu Phi, sau đó là châu Á".

Đánh giá: Hiệu quả trong việc giúp model hiểu được ngữ cảnh và logic của câu hỏi.

Few-Shot Prompting

Vấn đề: Model không có đủ dữ liệu để đào tạo.

Cách làm: Sử dụng kỹ thuật few-shot prompting, ví dụ: "Hãy giúp tôi viết một đoạn văn về chủ đề du lịch, với các từ khóa 'du lịch', 'nghỉ dưỡng', 'khách sạn'".

Đánh giá: Hiệu quả trong việc giúp model học hỏi từ ít dữ liệu.

System Prompt Design

Vấn đề: Model không hiểu được yêu cầu của người dùng.

Cách làm: Thiết kế hệ thống prompt phù hợp, ví dụ: "Hãy giúp tôi tìm hiểu về các loại thực phẩm□□, với các tiêu chí 'giá cả', 'chất lượng', 'sự tiện lợi'".

Đánh giá: Hiệu quả trong việc giúp model hiểu được yêu cầu của người dùng.

BÀI HỌC AI HÔM NAY CHO DEV

1. **Tối ưu chi phí & hiệu năng LLM**

2. Để tối ưu hóa chi phí và hiệu năng của mô hình ngôn ngữ lớn (LLM), các nhà phát triển cần biết cách tinh chỉnh và tối ưu hóa mô hình. Điều này giúp giảm thiểu chi phí tính toán và tăng tốc độ xử lý, đồng thời vẫn duy trì hiệu suất của mô hình.

3. Ví dụ, sử dụng kỹ thuật fine-tuning và LoRA (Low-Rank Adaptation) có thể giúp giảm kích thước mô hình và tăng tốc độ xử lý. Ví dụ code: `model = LLM.from_pretrained('base_model'); model.fine_tune(dataset, epochs=5)`

4. Tip: Sử dụng thư viện như Hugging Face Transformers để tối ưu hóa và tinh chỉnh mô hình LLM, và thử nghiệm với các kỹ thuật khác nhau để tìm ra phương pháp tối ưu cho ứng dụng của bạn.