

Bản Tin AI Hằng Ngày

Cập nhật công nghệ AI mới nhất

"Success is not final, failure is not fatal: it is the courage to continue that counts."

↪ Thành công không phải là điểm kết, thất bại không phải là kết thúc — điều quan trọng là can đảm tiếp tục.

— Winston Churchill

Hành trình mới là điều quan trọng — đừng tự mãn khi thành công, đừng bỏ cuộc khi thất bại.

TIN TỨC NỔI BẬT

1

9 Giải pháp Lập trình AI thay thế cho Phát triển Terminal: Hướng dẫn Hoàn chỉnh 2026

9 AI Coding Alternatives for Terminal Development: Complete 2026 Guide

Pasquale Pillitteri

[Đọc bài viết →](#)

Bài viết "9 Giải pháp Lập trình AI thay thế cho Phát triển Terminal: Hướng dẫn Hoàn chỉnh 2026" của Pasquale Pillitteri cung cấp cái nhìn tổng quan về các giải pháp lập trình được hỗ trợ bởi AI cho phát triển terminal. Những giải pháp thay thế này nhằm mục đích tự động hóa quá trình lập trình, tăng hiệu suất và năng suất. Hướng dẫn này nhấn mạnh chín công cụ lập trình AI có thể được sử dụng trong phát triển terminal, bao gồm hoàn thành mã được hỗ trợ bởi AI, gỡ lỗi và kiểm thử. Những công cụ này sử dụng các thuật toán học máy để phân tích mã và cung cấp đề xuất, giảm nhu cầu lập trình thủ công. Bài viết không chỉ định tên của chín giải pháp lập trình AI, nhưng nó ngụ ý rằng chúng cung cấp một loạt các tính năng, từ hoàn thành mã cơ bản đến các khả năng gỡ lỗi và kiểm thử tiên tiến. Hướng dẫn này dành cho các nhà phát triển muốn tận dụng AI trong quy trình phát triển terminal của họ, với mục tiêu cải thiện tốc độ và độ chính xác của mã. Bằng cách sử dụng các công cụ được hỗ trợ bởi AI này, các nhà phát triển có thể tập trung vào các nhiệm vụ cấp cao hơn, chẳng hạn như thiết kế và giải quyết vấn đề, trong khi để lại các nhiệm vụ lập trình tầm thường cho AI.

2

Claude Code so với GitHub Copilot 2026: SWE-bench, Giá cả [Đã Kiểm tra]

Claude Code vs GitHub Copilot 2026: SWE-bench, Pricing [Tested]

tech-insider.org

[Đọc bài viết →](#)

Trong một so sánh gần đây, Claude Code và GitHub Copilot đã được thử nghiệm trong một điểm chuẩn Kỹ thuật Phần mềm (SWE). Kết quả cho thấy cả hai công cụ mã hóa được hỗ trợ bởi AI đều hoạt động tốt, nhưng với một số điểm khác biệt đáng chú ý. Claude Code, được phát triển bởi Anthropic, đã thể hiện một lợi thế nhỏ trong một số lĩnh vực, chẳng hạn như chất lượng mã và khả năng bảo trì. Tuy nhiên, GitHub Copilot, được phát triển bởi GitHub, đã vượt trội trong việc hoàn thành mã và tích hợp với nền tảng. Khi nói đến giá cả, cả hai công cụ đều cung cấp các tùy chọn cạnh tranh. GitHub Copilot có sẵn dưới dạng dịch vụ dựa trên đăng ký, với một tầng miễn phí cho các dự án mã nguồn mở và một kế hoạch trả phí bắt đầu từ \$10 mỗi người dùng mỗi tháng. Claude Code, mặt khác, cung cấp một kế hoạch miễn phí với các tính năng hạn chế, cũng như một kế hoạch trả phí với các khả năng tiên tiến hơn bắt đầu từ \$15 mỗi người dùng mỗi tháng. Sự lựa chọn giữa hai công cụ cuối cùng phụ thuộc vào nhu cầu và ngân sách cụ thể của một cá nhân hoặc tổ chức. Việc thử nghiệm và đánh giá thêm có thể là cần thiết để xác định công cụ nào là lựa chọn tốt nhất.

3

Xây dựng Lưu lượng Công việc Sử dụng Cortex AI và Máy chủ MCP Quản lý

Build Agent-Powered Workflows Using Cortex AI and Managed MCP Servers

Snowflake

[Đọc bài viết →](#)

Snowflake đã giới thiệu một tích hợp mới cho phép người dùng xây dựng các quy trình làm việc được hỗ trợ bởi các tác nhân sử dụng Cortex AI và máy chủ Managed MCP. Tích hợp này cho phép các nhà phát triển tận dụng sức mạnh của Cortex AI để tự động hóa và tối ưu hóa các quy trình làm việc phức tạp, đồng thời cũng được hưởng lợi từ khả năng mở rộng và độ tin cậy của máy chủ Managed MCP của Snowflake. Với tích hợp này, người dùng có thể tạo các quy trình làm việc tùy chỉnh sử dụng các khả năng học máy của Cortex AI để phân tích dữ liệu, đưa ra dự đoán và tự động hóa các nhiệm vụ. Các quy trình làm việc có thể được xây dựng và triển khai bằng giao diện trực quan của Snowflake và có thể được quản lý và giám sát dễ dàng thông qua nền tảng. Sự kết hợp giữa Cortex AI và máy chủ Managed MCP

cung cấp một nền tảng mạnh mẽ để xây dựng và triển khai các quy trình làm việc có khả năng mở rộng và độ tin cậy cao. Tích hợp này được thiết kế để giúp các nhà phát triển và nhà khoa học dữ liệu tăng tốc quy trình làm việc của họ, cải thiện năng suất và đưa ra quyết định dựa trên dữ liệu với sự dễ dàng và chính xác cao hơn. Bằng cách tận dụng điểm mạnh của cả hai công nghệ, người dùng có thể mở khóa những thông tin mới và thúc đẩy kết quả kinh doanh.

4

Lỗ hổng bảo mật cho phép kẻ tấn công kiểm soát toàn bộ Mac đang bị khai thác tích cực

Vulnerability giving attackers full control of Macs is under active exploitation

Ars Technica

[Đọc bài viết →](#)

Một lỗ hổng bảo mật nghiêm trọng trên macOS, được theo dõi với mã CVE-2026-65400, đang bị các hacker khai thác tích cực. Lỗ hổng này cho phép kẻ tấn công thực thi mã độc và giành quyền kiểm soát toàn bộ Mac mà không cần mật khẩu. Vấn đề này bắt nguồn từ một lỗi trong tính năng chia sẻ màn hình của macOS, cho phép các hacker từ xa xem màn hình và kiểm soát bàn phím và chuột. Các quan chức Hà Lan đã cảnh báo rằng lỗ hổng này đang bị khai thác khi cổng 5900 được lộ trên Internet, và Apple đã phát hành một bản vá cho macOS Tahoe, Sequoia và Sonoma. Để bảo vệ chống lại lỗ hổng này, người dùng được khuyến nghị nên giữ cổng 5900 đóng, thậm chí khi sử dụng chia sẻ màn hình, và cài đặt bản cập nhật bảo mật mới nhất. Ngoài ra, người dùng nên tắt chia sẻ màn hình khi không sử dụng và chỉ bật nó khi cần thiết. Hiện tại, không có dấu hiệu cho thấy lỗ hổng này đang được sử dụng để cài đặt malware ngoài các miner Monero.

5

Tải xuống: Quy tắc mới của Flock, Tương lai của sao chép và Tế bào Trẻ em

The Download: Flock's new rules, cloning's future, and children's cells

MIT Tech Review

[Đọc bài viết →](#)

Tin tức công nghệ nổi bật bao gồm một thay đổi lớn của gã khổng lồ công nghệ cảnh sát Flock, đang hạn chế truy cập vào cơ sở dữ liệu đọc biển số xe của họ để ngăn chặn lạm dụng. Công ty sẽ yêu cầu các sĩ quan nhập số trường hợp hình sự trước khi tìm kiếm cơ sở dữ liệu và mở rộng kiểm toán tự động đối với các tìm kiếm đáng ngờ. Tuy nhiên, biện pháp này có thể không hoàn hảo. Các nhà khoa học đã đạt được

một bước đột phá trong công nghệ nhân bản, thành công trong việc biến phôi chuột đực thành cái bằng cách sử dụng phương pháp dựa trên CRISPR. Phát triển này có thể có những ý nghĩa quan trọng đối với nỗ lực bảo tồn và có thể dẫn đến việc tạo ra gia súc và thậm chí là "bản sao người không có não" đã được chỉnh sửa gen. Trong tin tức khác, các nhà nghiên cứu đang làm việc để tạo ra một cơ sở dữ liệu toàn diện về mô khỏe mạnh của nhi để hiểu rõ hơn về cách các tế bào của trẻ phát triển và phản ứng với thuốc. Điều này có thể tiết lộ cách các bệnh xuất hiện ở người lớn bắt đầu sớm hơn trong cuộc sống. Các máy bay không người lái của Ukraine đã đánh bại lực lượng Mỹ trong một cuộc tập trận quân sự, lộ ra những điểm yếu trước các cuộc tấn công bằng máy bay không người lái. Ngoài ra, Apple đã phát triển mô hình AI riêng của mình cho Trung Quốc với sự hỗ trợ của Alibaba, và các nhà lập pháp đang xem xét việc quản lý việc sử dụng chatbot như những người bạn đồng hành.

6

Tại sao các Agent Quản lý là Điều lớn tiếp theo trong Xây dựng Agent

Why managed agents are the next big thing in agent building

LangChain Blog

[Đọc bài viết →](#)

Sự phát triển của "các tác nhân được quản lý" là bước tiến quan trọng tiếp theo trong việc xây dựng tác nhân, theo những tiến bộ gần đây trong lĩnh vực này. Khái niệm này xuất hiện sau khi ra mắt các khung AI và ứng dụng đầu tiên, chẳng hạn như LangChain, ChatGPT và AutoGPT, những ứng dụng này đã tận dụng Large Language Models (LLMs) vào năm 2022 và 2023. Khi các model này được cải thiện, các ứng dụng AI phức tạp hơn đã được xây dựng, nhưng chúng vẫn chưa giống với các tác nhân mà chúng ta biết ngày nay. Tuy nhiên, khi LLMs có khả năng chạy trong vòng lặp và gọi các công cụ, một nền tảng vững chắc cho các tác nhân đã được thiết lập. Khái niệm về Agent Harnesses, chẳng hạn như Claude Code, Pi và Deep Agents, đã xuất hiện khi các developer tìm ra các công cụ và môi trường phù hợp để thêm vào vòng lặp này. Việc ra mắt Deep Agents gần một năm trước là một trong những nỗ lực đầu tiên để đưa các mẫu này vào một harness có thể sử dụng chung. Kể từ đó, những hiểu biết đã định hình hướng tiếp theo, bao gồm việc phát triển các nguyên thủy chung cho việc chạy agent harnesses ở quy mô lớn và các tiêu chuẩn cho việc điều khiển hành vi của tác nhân. Các trải nghiệm của managed agent, nơi agent harness được chạy trên cơ sở hạ tầng được quản lý, đã trở thành điều lớn tiếp theo trong việc xây dựng tác nhân. Những trải

nghiệm này giúp các builder dễ dàng tạo ra các tác nhân sản xuất bằng cách đóng gói harness với cơ sở hạ tầng, giảm thiểu sự phức tạp và thách thức liên quan đến việc chạy tác nhân trong sản xuất.

7

Một khung đánh giá đã tìm thấy điều mà đánh giá định tính không thể: Các mô hình AI tự tin nhất khi sai

An eval harness found what qualitative review couldn't: AI models are most confident when wrong

VentureBeat

[Đọc bài viết →](#)

Trong việc phát triển các công cụ được hỗ trợ bởi mô hình ngôn ngữ lớn (LLM), một bước quan trọng thường bị bỏ qua do tính chất nhàm chán của nó: xác minh độ chính xác của đầu ra của mô hình. Khoảng cách giữa "nghe có vẻ đúng" và "đúng có thể kiểm chứng" là nơi mà hầu hết các công cụ doanh nghiệp được hỗ trợ bởi LLM thất bại im lặng. Một đánh giá định tính, nơi một mẫu đầu ra được xem xét bởi người có kiến thức về lĩnh vực, có thể bắt được các lỗi rõ ràng nhưng bỏ qua các vấn đề tinh vi hơn. Một khung đánh giá, đánh giá đầu ra của mô hình so với sự thật được gắn nhãn, là một phương pháp hiệu quả hơn. Khung này hoạt động bằng cách tạo một tập dữ liệu tổng hợp với các câu trả lời chính xác đã biết, đánh giá đầu ra của mô hình so với tập dữ liệu này và tính điểm độ chính xác của nó. Kết quả cho thấy mô hình tự tin nhất trong các câu trả lời sai của nó, đặc biệt là trong các kịch bản có tín hiệu chống chéo. Phát hiện này sẽ không thể nhìn thấy nếu không có khung đánh giá. Đối với các đội triển khai công cụ được hỗ trợ bởi LLM, điều quan trọng là phải đo lường độ chính xác so với các câu trả lời chính xác đã biết, chứ không chỉ xem xét tính lưu loát và tính nhất quán. Xây dựng một tập dữ liệu sự thật cơ bản tổng hợp là phần khó khăn nhất nhưng quan trọng nhất của quá trình này, vì nó buộc các đội phải định nghĩa "đúng" có nghĩa là gì cho trường hợp sử dụng cụ thể của họ.

8

Người tiên phong Xử lý Tín hiệu Số Bede Liu qua đời ở tuổi 91

Digital Signal Processing Pioneer Bede Liu Dies At 91

IEEE Spectrum

[Đọc bài viết →](#)

Bede Liu, một nhà tiên phong nổi tiếng trong lĩnh vực xử lý tín hiệu số, đã qua đời vào ngày 7 tháng 5 tại tuổi 91. Ông được coi rộng rãi là một trong những người sáng lập của xử lý tín hiệu số hiện đại, một lĩnh vực

áp dụng các thuật toán toán học để phân tích, sửa đổi và truyền tín hiệu. Liu đã giảng dạy kỹ thuật điện tại Đại học Princeton trong hơn 50 năm và giữ chức chủ tịch bộ phận kỹ thuật điện và máy tính của trường đại học từ năm 1994 đến 1997. Nghiên cứu của ông đã đóng vai trò quan trọng trong quá trình chuyển đổi từ xử lý tín hiệu tương tự sang số của âm thanh, hình ảnh và video. Công việc của ông đã đặt nền tảng cho các hệ thống truyền thông hiện đại, hệ thống đa phương tiện và thiết bị điện tử tiêu dùng, với nghiên cứu của ông về bộ xử lý tín hiệu số tiết kiệm năng lượng đã góp phần phát triển các công nghệ được sử dụng bởi hàng tỷ người, bao gồm điện thoại di động, video trực tuyến và truyền thông internet. Liu đã nhận được Huy chương Xử lý Tín hiệu Jack S. Kilby của IEEE năm 2018 cho những đóng góp liên tục của ông trong lĩnh vực xử lý tín hiệu số. Ông đã viết cuốn sách giáo khoa có ảnh hưởng "Xử lý Tín hiệu Số: Lý thuyết, Thiết kế và Thực hiện" và xuất bản 250 bài báo kỹ thuật, với công việc của ông vẫn được tham khảo rộng rãi ngày nay.

TIPS & TRICKS CHO DEV

Tối ưu hóa AI observability

Vấn đề: Việc giám sát hiệu suất của mô hình AI trở nên phức tạp.

Cách làm: Sử dụng LangSmith để theo dõi và phân tích kết quả, ví dụ: `langsmith observe --model my_model`.

Đánh giá: Hiệu quả cao, nên dùng khi cần phân tích sâu.

Kiểm soát chi phí AI

Vấn đề: Chi phí vận hành mô hình AI tăng cao.

Cách làm: Áp dụng Arize Phoenix để theo dõi và kiểm soát chi phí, ví dụ: `arize phoenix --cost-control`.

Đánh giá: Giảm chi phí, nên dùng khi cần tối ưu hóa.

Tăng cường tracing

Vấn đề: Khó khăn trong việc theo dõi và phân tích dữ liệu đầu vào.

Cách làm: Sử dụng Langfuse để tăng cường tracing, ví dụ: `langfuse trace --input data.csv`.

Đánh giá: Cải thiện hiệu suất, nên dùng khi cần phân tích dữ liệu.

BÀI HỌC AI HÔM NAY CHO DEV

1. Tối ưu chi phí & hiệu năng LLM

Dev cần biết về tối ưu chi phí và hiệu năng của Large Language Model (LLM) để đảm bảo ứng dụng của mình hoạt động hiệu quả và tiết kiệm chi phí. Việc tối ưu

hóa LLM giúp giảm thiểu thời gian và tài nguyên cần thiết cho việc đào tạo và triển khai mô hình.

2. Giải thích ngắn gọn — tại sao dev cần biết: Việc tối ưu hóa LLM giúp giảm thiểu chi phí và tăng tốc độ xử lý dữ liệu, từ đó giúp ứng dụng hoạt động mượt mà hơn.

3. Ví dụ thực tế hoặc code snippet minh họa: Ví dụ, có thể sử dụng kỹ thuật fine-tuning để tối ưu hóa LLM cho một nhiệm vụ cụ thể, giúp giảm thiểu số lượng tham số và tăng tốc độ xử lý.

4. Tip hoặc bước tiếp theo: Để bắt đầu tối ưu hóa LLM, hãy bắt đầu bằng cách phân tích nhu cầu và mục tiêu của ứng dụng, sau đó áp dụng các kỹ thuật như fine-tuning, pruning, và quantization để giảm thiểu chi phí và tăng hiệu năng.

Luôn đi đầu trong thế giới AI! · Stay ahead in AI!

Nguồn: Google News · Groq AI