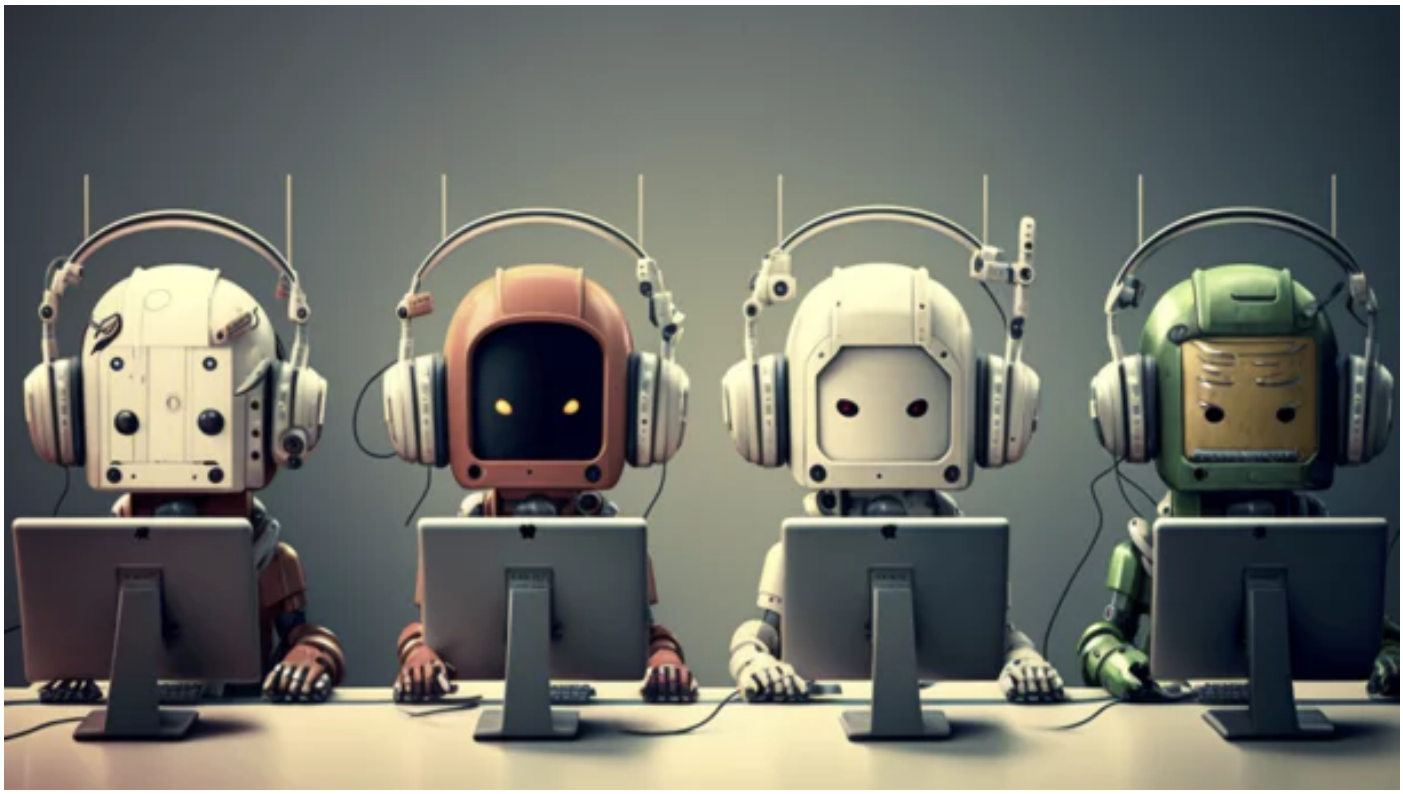
heise+

AutoGPT: KI-Agenten beginnen, auf GPT-4-Basis autonom in der Welt zu handeln

Das Verschmelzen großer Sprachmodelle mit interaktiven Agenten führt zu Interaktionsmustern, die glaubwürdige Simulationen menschlichen Verhaltens ermöglichen.



KI Midjourney | Bearbeitung: c't

14.04.2023 18:57 Uhr | **Developer**

Von Silke Hahn

Auf den ersten Blick schaut alles niedlich aus: eine Kleinstadt als Computersimulation. In interaktiven Puppenhäusern im Stile von The Sims interagieren 25 KI-Agenten in natürlicher Sprache untereinander in einer abgeschirmten Umgebung (Sandbox). Ein Agent möchte eine Valentinstagsparty ausrichten, und zwei Tage lang verbreiten seine Kumpels die E[illegible]ngen. Alle koordinieren sich so, dass sie zur rechten Zeit bei der Party erscheinen.

Allerdings bewegen keine Menschen die Figuren, diese bewegen sich selbst und treffen unaufgefordert Entscheidungen. Fast möchte man sagen, "autonom". Denn eine Agentenstruktur erweitert hier die Fähigkeiten großer Sprachmodelle, eine vollständige Aufzeichnung sämtlicher Erfahrungen des handelnden "Agenten" zu speichern. All dies geschieht in natürlicher Sprache, wie auch wir Menschen sie zur Verständigung und zur Interaktion mit unserer Umwelt einsetzen.

Interaktives Experiment mit KI-Agenten

Das Setting entspringt keinem Kindergeburtstag, sondern einem wissenschaftlichen Experiment zu den Fähigkeiten moderner Künstlicher Intelligenz (KI). Der darüber verfasste Forschungsbericht ("Generative Agents: Interactive Simulacra of Human Behavior") beantwortet teils die Frage, inwieweit heutige KI-Systeme schon in der Lage sind, menschliches Verhalten nachzuahmen. Die Agenten gehen Tätigkeiten nach, die an menschliches Tun erinnern: Sie wachen auf, bereiten Frühstück zu, begeben sich zur Arbeit. Malen, schreiben, bilden sich "Meinungen", nehmen einander wahr und stoßen Gespräche an.

Screenshot aus der Demo: "Generative Agents: Interactive Simulacra of Human Behavior"
(Bild: arXiv Demo)

Die Agenten beobachten, planen und "reflektieren" das Erfahrene ohne menschliches Einwirken. So entspricht es ihrer Architektur, die laut ihren Schöpfern (einem Team von Forschern aus Stanford und von Google) zu glaubwürdigem, menschenähnlichem Verhalten führen soll. Das beobachtbare Vorgehen in neutralen Begriffen zu beschreiben, ist nicht ganz einfach, da potenziell Projektion der Beobachter aus eigenen Erfahrungen und Erinnerungen mit im Spiel ist.

Handlungsfähig: Agenten sitzen auf großem Sprachmodell

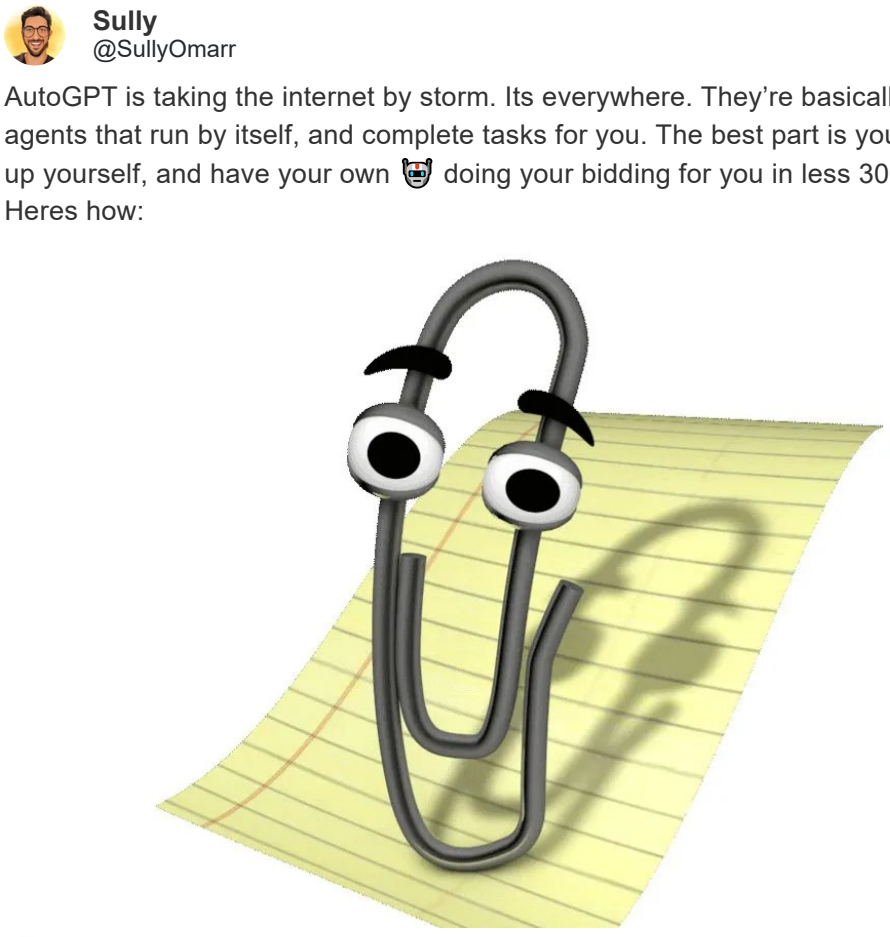
Dafür greifen sie auf ein großes Sprachmodell zu. Sie zeigen, wie einfach es offenbar ist, ein Large Language Model (LLM) durch einen relativ simplen Handlungsloop und Prompt dazu zu befähigen, Aufgaben in der echten Welt auszuführen durch die Softwareschicht eines Agenten. Die Agenten sind aus der Box, und im Internet häufen sich Berichte von Personen, die über KI-Agenten im Stil von AutoGPT aus dem Internet Pizza bestellen lassen und kleine Aufgaben an sie delegieren. Das E[illegible]idende ist, dass sie nicht nur untereinander agieren können wie in dem abgeschirmten Experiment, sondern pr[illegible]ell auch mit Menschen.

Hinter "AutoGPT" verbirgt sich kein neues User-Interface wie ChatGPT, sondern das Konzept des Selbstpromptens großer Sprachmodelle zum Automatisieren zahlreicher Aufgaben mit Text und Code. Auto kommt hier von autonomer Steuerung: Self-Prompting und Auto-Prompting sind gemeint. Nach einer initialen Aufforderung durch einen Prompt (Aufforderung in natürlicher Sprache) beginnt das große Sprachmodell, weitere Prompts zu entwickeln und auszuführen, die erneut zu weiteren Anweisungen führen können, die das Programm sich selbst erteilt.

Real-World-Schnittstellen für halbautonome KI-Agenten

Wirkmächtig ist der Ansatz dadurch, dass er sich mit Werkzeugen in der echten Welt verbinden lässt. Diese Werkzeuge befähigen Agenten und ihre Programme etwa dazu, das Internet zu durchsuchen, Code nicht nur zu schreiben, sondern auch zu testen, anderen Agenten, Programmen oder auch Menschen Anweisungen zu erteilen oder Anweisungen zu erhalten. Unzählige Anwendungsmöglichkeiten sind denkbar, bis hin zur Steuerung von Robotern, die auch körperlich in der Welt agieren. Einzelne API-Aufrufe an das Sprachmodell werden dazu in Schleifen verknüpft (Agent Loop), wodurch der Eindruck entsteht, dass die so beschaffenen Agenten eigenständig wahrnehmen, denken und handeln können, wie der Entwickler Andrej Karpathy auf Twitter schreibt (der on-and-off bei OpenAI mitwirkte, derzeit wieder mitwirkt).

Besonders populär ist zurzeit das AutoGPT-Verzeichnis auf GitHub, in dem experimenteller Code bereitsteht, "um GPT-4 voll autonom zu machen", wie es in der Repositorybeschreibung heißt. Das Repository enthält eine Demo, ausführliche Installationshinweise, Informationen zur Nutzung, zum Sprachmodus, die Konfiguration für die API-Schlüssel bei Google, Hinweise zum Einrichten des erforderlichen Speichers, einen "GPT-3.5-Only-Modus", Tipps zur Bildsynthese und Hinweise zu den Beschränkungen sowie zum Ausführen von Tests. Mit den geteilten Daten ist es Agenten möglich, das Internet zu durchsuchen, um Informationen zu sammeln, Lang- und Kurzzeiterinnerungen lassen sich verwalten, GPT-4-Instanzen für die Texterzeugung einrichten, Zugang zu beliebten Webseiten und Plattformen herstellen und Daten mit GPT-3.5 sowohl speichern als auch zusammenfassen.



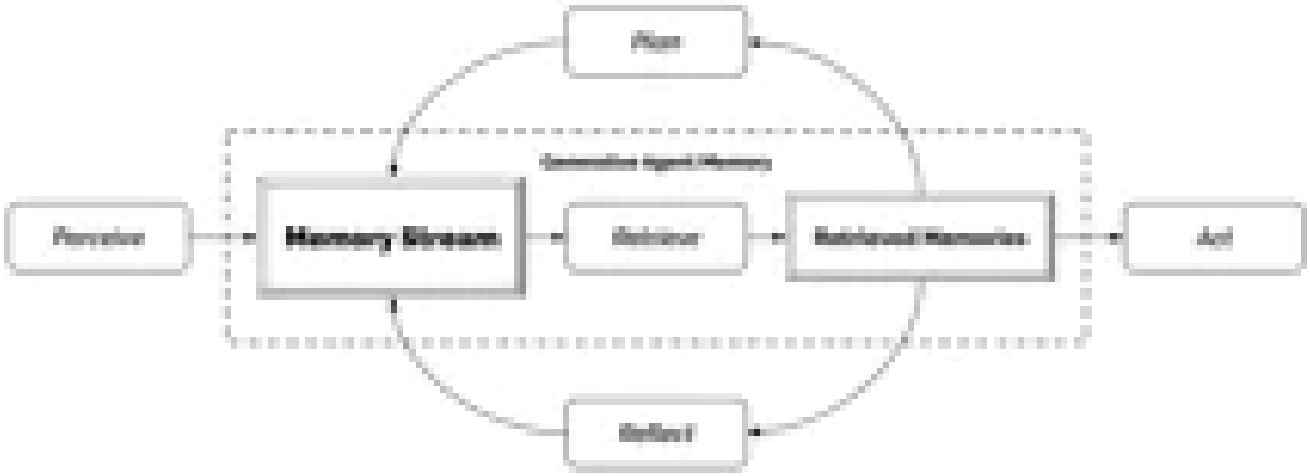
Pizza bestellen, Unfug bauen – rekursives Selbstdebuggen

Wer das Experiment auf eigene Faust angehen will, benötigt Visual Studio Code und einen "devcontainer" (die Datei .devcontainer soll so konfiguriert sein, dass Entwickler direkt loslegen können – mehr dazu im Repository). Unabdingbar ist auch ein API-Zugang zu OpenAI, hierüber können Kosten entstehen. Wahlweise können auch API-Zugänge zu Pinecone und ElevenLabs eingesetzt werden (Pinecone, falls das als Speicherlösung gewählt wird, und ElevenLabs für eine mögliche Sprachausgabe). Der Fantasie von Developern ist vermutlich keine Grenze gesetzt, da auch andere API-Zugänge als die im Projekt beschriebenen sich verknüpfen lassen.

Für die Installation und genaue Umsetzung mögen Interessierte bei GitHub vorbeischauen. Eine Schritt-für-Schritt-Anleitung findet sich auch in mehreren Threads auf Twitter, unter anderem mit einer Linksammlung. Im Internet finden sich zahllose Beispiele von Testläufen, etwa dem sprachgesteuerten Bestellen von Pizza über einen selbst installierten Agenten mit Internetzugang via API. Bereits seit Anfang April beherrscht AutoGPT den Umgang mit Code, kann seinen eigenen Code schreiben (mittels GPT-4) und ihn über Python-Skripte ausführen lassen. Das ermöglicht rekursives Debuggen, Entwickeln und eine Art Selbstoptimierung. Zu AutoGPT besteht auch ein Discord-Kanal.

Großes Sprachmodell im Gerüst: Architektur der Agenten

Die Architektur eines solchen Agenten ist einfach gestrickt, das Erinnerungsvermögen und Speichern von Erfahrungen ist dabei grundlegend: Eine Beobachtung wird einem Erinnerungsstrom zugeführt, aus dem der Agent auf einzelne Erinnerungen zugreifen kann. Neu hinzukommende Beobachtungen werden kontinuierlich aufgenommen, während der Agent aus dem Pool ihm zugänglicher Erinnerungen neue Wahrnehmungen analysiert ("reflektiert") und künftige Handlungen plant. Das Gerüst aus Erinnerung, eigenständigem Zugriff, Aufruf von Gelerntem und dem darüber stattfindendem "Nachdenken" und "Planen" heißt in der Forschung "Scaffolded LLM" (zu Deutsch etwa "in ein Gerüst gepacktes große Sprachmodell"). Es gilt als ein Schritt hin zu Computern, die in natürlicher Sprache arbeiten.



Generativer Agent: Schema eines Scaffolded-LLM-Programms
(Bild: Generative Agents Paper (Stanford/ Google))

Die Idee zu dieser Architektur ist nicht ganz neu, sie stammt von dem ungarisch-amerikanischen Mathematiker John von Neumann, der Beiträge zur Logik, Quantenmechanik und Spieltheorie leistete. Von Neumann gilt als einer der Gründerväter der Informatik. Das Von-Neumann-Modell (auch Princeton-Architektur) ist eine Computer-Architektur, die von Neumann und weitere Forscher 1945 erstmals beschrieben. Sie beschreibt das Design eines elektronischen digitalen Computers mit fünf Komponenten:

- einer Prozessoreinheit mit arithmetischer Logik und Prozessorregistern
- einer Kontrolleinheit mit einem Anweisungsregister und einem Programmzähler
- Speicher (Memory) zum Aufbewahren von Daten und Anweisungen
- externem Massenspeicher

- Input- und Output-Mechanismen

Skizze der nach John von Neumann benannten Architektur
(Bild: Wikipedia)

Die Fähigkeit, Anweisungen als Daten zu behandeln, ist es, was Assembler, Compiler, Linker, Loader und andere automatisierte Programmwerkzeuge erst ermöglicht. Das Schaltungskonzept war seinerzeit revolutionär, da es rasche Änderungen an Programmen ohne Änderung an der Hardware (etwa fest verschaltete Programme oder Lochkarten) erlaubte. Konrad Zuse hatte bereits 1936 ähnliche Ideen ausgearbeitet und in Patenten festgehalten, diese zwischen 1938 und 1941 mechanisch umgesetzt und mit dem Zuse Z3 den ersten funktionsfähigen Digitalrechner der Welt gebaut. Zuse und von Neumann entwickelten die Ideen wohl unabhängig voneinander. Mit der nach von Neumann benannten Architektur lassen sich fast alle Komponenten einer Turingmaschine umsetzen (abgesehen von dem unbegrenzten Speicher, der Alan Turing vorschwebte, da dieser technisch nicht umsetzbar ist).

Scaffolded LLM ähnelt der Von-Neumann-Architektur

Die Architektur eines "in ein Gerüst gekleideten großen Sprachmodells" (Scaffolded LLM) ähnelt der Von-Neumann-Architektur stark. Im Kern haben wir es mit einem LLM zu tun, das Anweisungen (Prompts) empfängt und in natürlicher Sprache ausführt. Eine Reihe von Musteranweisungen führt die Aufgaben näher aus und bestimmt die Daten, auf denen das KI-Programm ausführbar ist. Ein Speicher hält größeren Kontext vor als in ein LLM eingegeben werden könnte, in ihn kann die zentrale Recheneinheit schreiben und aus ihm auslesen.

Der Verfasser eines Blogbeitrags zum Thema spricht von einer "konvergenten Evolution" der Von-Neumann-Architektur, die für sich genommen wenig überraschend sei – handele es sich doch um eine natürliche Abstraktion zum Gestalten von Rechenmaschinen. Der Unterschied besteht darin, dass dieser "Computer" nicht auf Bits, sondern auf Textbasis läuft. Dieser Computer in natürlicher Sprache (NL für Natural Language) ist in der Theorie universell und praktisch, da Menschen vorzugsweise in natürlicher Sprache interagieren. Für zahlreiche Aufgaben sei es einfacher, sie in wenigen Sätzen zu beschreiben, als sie präzise in Programmcode zu übersetzen.

N[illegible]ichsprachige Computer und digitale Rechenmaschinen

Beren Millidge leitet die Forschungsabteilung des KI-Start-ups Conjecture und ist Postdoc-Forscher im Fach Computational Neuroscience in Oxford. Millidge hat die Analogie ausführlich dargelegt, was hier nur in Kürze wiedergegeben werden kann (ausführlich in seinem Blogpost): Ihm zufolge seien die großen Sprachmodelle (LLM) das Äquivalent zur CPU eines herkömmlichen Computers. Statt Bits in Registern seien die Grundeinheiten der Rechenvorgänge hier Token in Kontextfenstern und eine solche Natural Language Processing Unit (NLPU) habe als "natürlichen" Typ Strings (wie die CPU die Bits). Prompt (Anweisung) und Kontext entsprächen dem RAM, also einfach zugänglicher Speicher zum raschen Ausführen durch CPU beziehungsweise LLM.

Der Speicher (Memory) hingegen sei hier die Vektordatenbank, und die Heuristik (etwa als Vektorsuche über Embeddings) zum Auffinden bestimmter "Erinnerungen" und Speicherorte soll Millidge zufolge dem Memory Controller in digitalen Computern entsprechen. Die Interaktion mit der Außenwelt findet in digitalen Rechnern über Treiber, Hardware und Softwaremodule statt, die es der CPU ermöglichen, externe Hardware wie Drucker, Maus, Bildschirm zu steuern. Scaffolded LLMs haben stattdessen Plug-ins und ähnliche Mechanismen. Das große Sprachmodell im Inneren werde durch einen "einrüstenden" (scaffolding) Code umhüllt. Dieser Programmcode sorgt für das Einbetten von Protokollen zum Verknüpfen einzelner LLM-Aufrufe. Dadurch ließen sich reaktive Handlungsschleifen (agent loops) oder rekursive Programme, etwa zum Zusammenfassen von Büchern und Texten, einbauen. Die Protokolle sind nach Millidge die eigentlichen Programme, die auf diesem Natürlichen-Sprach-Computer laufen.

Natürliche Abstraktion: Es geht um mehr als um die Agenten

Der Computer-Neurowissenschaftler führt den Vergleich weiter aus, insbesondere auf grundlegende Unterschiede zu digitalen Computern geht er umfassend ein. Wichtig ist eine Begriffsklärung: Millidge vermeidet den Begriff "agentisierter" Sprachmodelle (agentized LLMs) durch das breiter angelegte Konzept "eingerüsteter" bzw. eingehüllter Sprachmodelle (scaffolded LLMs). Zurzeit seien Agenten ein Hype-Thema, allerdings gehe es um mehr als um rein agentische Anwendungen. KI-Agenten sind natürliche Abstraktionen, die für bestimmte Arten von Aufgaben geeignet sind. Es gebe jedoch andere mögliche Programme in natürlicher Sprache, lautet der Einwand.

Interessant sind auch die Schlüsse, die Millidge zur wirtschaftlichen Position von Foundation-Model-Anbietern wie OpenAI zieht. Anbieter großer Sprachmodelle (LLM, auch Basismodell) besetzen die gleiche Nische wie große Chiphersteller im Zeitalter digitaler Computer, etwa wie Intel. Das Geschäftsmodell sei vergleichbar: Das Training der Grundmodelle gehe mit enormen Kosten einher, vergleichbar mit den Kosten zum Errichten einer neuen Fabrik. Sie verkaufen eine kommodifizierte Ware als Massenprodukt (hier vergleicht Millidge API-Aufrufe mit Prozessoren) bei einer großen Gewinnspanne, zugleich aber auch beträchtlichen Kosten (beim KI-System durch die Inferenz im Betrieb).

Prognose: Konsolidierung wie in der Halbleiterindustrie?

Davon ausgehend wagt der Forscher eine Analogie für die Entwicklung des Wirtschaftszweigs, mit Blick auf die Halbleiterindustrie: "Wir sollten eine Konsolidierung zu wenigen Hauptanbietern erwarten, die als Oligopolisten jeweils hohe Fixkosten haben und sich in relativ starkem Wettbewerb zueinander befinden werden." Millidge geht nicht davon aus, dass diese Oligopole jemals "Geld drucken" werden in einer Weise, wie es bei Software as a Service (SAAS) oder Software-basierten Unternehmen der Fall ist und war. Rechenvorgänge in natürlicher Sprache (NLOPs für Natural Language Operations) unterscheiden sich zudem von den Gleitkommazahlberechnungen (Floating Point Operations, kurz FLOPs) durch unterschiedliche "intrinsische Schwierigkeiten".

Kleinere Sprachmodelle können für bestimmte Aufgaben besonders geeignet sein, andere Aufgaben erfordern eher große LLMs auf dem neuesten Stand der Technik. Daher sei unwahrscheinlich, dass es jemals ein einziges, einheitliches Zentralsystem wie eine uniforme CPU (im Beispiel: LLM) geben werde. Eine Vielzahl heterogener Aufrufe unterschiedlicher Arten von Modellen sei wahrscheinlicher, in unterschiedlichen Größenordnungen und mit verschieden gearteten Spezialisierungen.

Ausblick: Agenten-Demo und Robotersteuerung

Wer den Agenten aus dem Stanford-Experiment bei ihrem Tun und Treiben zuschauen mag, kann sie in einer Demo durch ihren Tag begleiten. Jeder Agent und jede Agentin ist darin anklickbar, sodass ein komplexeres Handlungsgeschehen des ausgewählten Agenten in seinem Umfeld samt Interaktion mit weiteren Agenten beobachtbar wird. Zu jedem Agenten gibt es in Echtzeit Status-Updates in Textform, die beschreiben, was er gerade tut, wo er sich aufhält und mit wem er (oder sie) sich worüber unterhält.

Lesen Sie auch

Vorbild "Die Sims": ChatGPT treibt in virtueller Stadt autonome Agenten an

MIT Technology Review

Prinzipiell denkbar wäre ein Agieren und Interagieren von Robotern in der physischen Welt in ähnlicher Weise, angeschlossen an große Sprachmodelle zum Aufrufen und Beziehen von Informationen und Verarbeiten von Eindrücken aus der über Sensoren wahrgenommenen Umgebung. Das allerdings ist noch Zukunftsmusik. Beim aktuellen Tempo der KI-Entwicklung könnten solche Umsetzungen jedoch nicht allzu lange auf sich warten lassen. Unter anderem hat die KI-Forschungsabteilung von Microsoft, dem Hauptpartner von OpenAI, gerade einen Forschungsbericht zum Umsetzen von Anweisungen in natürlicher Sprache in ausführbare Roboterhandlungen mittels ChatGPT vorgelegt (Fallstudie zur ChatGPT-betriebenen Robotersteuerung in verschiedenen Umgebungen).

Zu dem Projekt liegen Materialien in einem GitHub-Repository bereit: Das Forschungsteam teilt die Prompts, die in der Mensch-Roboter-Kommunikation zum Einsatz kamen. Laut Projektbeschreibung lassen sie sich ohne Weiteres anpassen und in bestehende Roboter- und visuelle Erkennungssysteme integrieren.

(sih)

Kommentare lesen (55)

Zur Startseite

MEHR ZUM THEMA

CHATGPT KÜNSTLICHE INTELLIGENZ MACHINE LEARNING

Fo[illegible]i heise online: Machine Learning

TEILE DIESEN BEITRAG

Kurzlink: https://heise.de/-8963809

Das Beste aus heise+ »

Tests

Test BMW 230e: Plug-in-Hybrid mit reichlich Leistung und geringem Verbrauch

Die zweite Auflage des 2er-Vans weitet als Plug-in-Hybrid die elektrischen Fähigkeiten deutlich aus. Er legt bei Leistung und Reichweite spürbar zu. Ein Test.

heise+

Tests

Fünf smarte Citybikes von Ampler bis VanMoof im Test

Sie bringen einen entspannt ans Ziel, eignen sich für den Einkauf, führen Statistiken über gefahrene Strecken und wehren sich gegen Diebe. Fünf E-Bikes im Test.

heise+

Ratgeber

Microsofts Upgrade-Skandal: Das Support-Ende für Windows 10 und seine Auswirkung

Damit Windows 11 auf einem PC läuft, muss dieser ungewöhnlich hohe Anforderungen erfüllen: Wir erklären, was genau damit gemeint ist und welche Folgen das hat.

heise+

Ratgeber

Nextcloud Hub 4: Die neuen KI-Funktionen bietet so keine Konkurrenz

Nextcloud hat in der neuen Version an vielen Stellen nachgebessert. Wir zeigen, wie gut das Open-Source-Paket gegen Microsoft, Google und Slack bestehen kann.

heise+

Ratgeber

Besserer Sound für Spiele: Sechs externe Gaming-Soundkarten im Test

Breite Soundkulisse, knackige Bässe und akkurate Ortung: Ein DAC verspricht hörbare Klangverbesserungen gegenüber Onboard-Sound. Wir testen beliebte Modelle.

heise+

Ratgeber

E[illegible]bauto mit Solaranlage laden: Wie es optimal klappt

Die Solaranlage auf dem Dach kann das Elektroauto nebenbei mit überschüssigem Strom versorgen – wenn Sie bei der Planung ein paar Details beachten.

heise+

Hintergrund

Hausvernetzung: Das Automationsprotokoll KNX in Theorie und Praxis

Wir schildern, wie das KNX-Protokoll funktioniert, wie Sie ohne Neubau an eine KNX-Anlage kommen und wie das Zusammenspiel unterschiedlichster Produkte klappt.

heise+

Tests

Opel Astra GSe im Test: Plug-in-Hybrid in flotter Verpackung

Opel verpackt den stärkeren von zwei Plug-in-Hybriden im Astra in einem Sportler-Dress. Kombiniert der GSe rasante Fahrleistungen mit einem geringen Verbrauch?

heise+

nach oben

Alle Angebote

IT News

Ne cker

heise Developer

heise Netze

heise Open Source

heise Security

Online-Magazine

heise+

Telepolis

heise Autos

TechStage

tipps+tricks

Services

Stellenmarkt heise Jobs

Weiterbildung

heise Download

Preisvergleich

Whitepaper/Webcasts

Netzwerk-Tools

Spielen bei Heise

Loseblattwerke

iMonitor

Heise Medien

heise Shop

Abo

Veranstaltungen

Arbeiten bei Heise

Mediadaten

Presse

Newsletter heise-Bot Push

Datenschutz

Cookies & Tracking

Impressum

Kontakt

Barriere melden

Mediadaten

Verträge kündigen

4141313