

Large Language Models for Robotics: Opportunities, Challenges, and Perspectives

Example 1: Google's SayCan for Language-Guided Household Robotics

مثال ۱ SayCan: گوگل برای رباتیک خانگی هدایت‌شده با زبان

این مقاله سیستم SayCan را برجسته می‌کند؛ سیستمی که بر روی ربات‌های واقعی جابه‌جایی و دست‌کاریِ سیار در محیط‌های خانگی مستقر شده است. SayCan یک مدل زبانی بزرگ را با مهارت‌های یادگرفته‌شدهٔ ربات ترکیب می‌کند تا دستورالعمل‌های زبان طبیعی باز را به اعمال قابل اجرا ترجمه کند. ربات با موفقیت وظایفی مانند آوردن اشیاء، باز کردن کشوها و قرار دادن اقلام را انجام می‌دهد و یک یکپارچه‌سازی عملی از استدلال LLM با قابلیت‌های فیزیکی ربات را نشان می‌دهد

[Large Language Models for Robotics: Opportunities, Challenges, and Perspectives].

examples

Example 2: PaLM-E for Multi-Modal Robotic Decision-Making

مثال ۲ PaLM-E: برای تصمیم‌گیری رباتیک چندوجهی

PaLM-E به‌عنوان یک سیستم تجسم‌یافتهٔ واقعی ارائه می‌شود که در آن یک مدل زبانی مستقیماً به ورودی‌های حسگری ربات، شامل تصاویر و بردارهای وضعیت، متصل است. ربات از PaLM-E برای پاسخ دادن به پرسش‌ها دربارهٔ محیط و برای تصمیم‌گیری در مورد اعمال دست‌کاری در زمان واقعی استفاده می‌کند. این مثال نشان می‌دهد که چگونه LLM ها فراتر از برنامه‌ریزی حرکت می‌کنند تا به یک ماژول تصمیم‌گیری پیوسته که درون ربات‌های فیزیکی تعبیه شده است تبدیل شوند

[Large Language Models for Robotics: Opportunities, Challenges, and Perspectives].

examples

Example 3: Language-Driven Navigation with LM-Nav

مثال ۳: ناوبری هدایت‌شده با زبان با LM-Nav

این مقاله LM-Nav را توصیف می‌کند؛ سیستمی که بر روی سکوه‌های واقعی ناوبری رباتیک آزمایش شده است و دستورالعمل‌های زبانی سطح بالا مانند «به آشپزخانه برو و نزدیک میز بایست» را دنبال می‌کند. ربات از تعریف‌های زبان-بینایی برای متصل کردن دستورات انتزاعی به محیط‌های واقعی استفاده می‌کند، بدون آنکه به نظارت در سطح مسیر نیاز داشته باشد. این امر استقرار عملی ناوبری هدایت‌شده با LLM را در فضاهای داخلی پیچیده نشان می‌دهد

[Large Language Models for Robotics: Opportunities, Challenges, and Perspectives].

examples

Example 4: GPT-Based Planning for Pick-and-Place Manipulation

مثال ۴: برنامه‌ریزی مبتنی بر GPT برای دست‌کاری بردار-قرارده

چندین آزمایش که در مقاله مورد بحث قرار گرفته‌اند نشان می‌دهند که LLM ها برنامه‌های ساخت‌یافتهٔ بردار-قرارده تولید می‌کنند که بر روی بازوهای رباتیک واقعی اجرا می‌شوند. مدل اشیا را شناسایی می‌کند، دربارهٔ روابط فضایی استدلال می‌کند و اعمالی مانند گرفتن، بلند کردن و قرار دادن را به ترتیب انجام می‌دهد. این سیستم‌ها در صحنه‌های شلوغ دنیای واقعی عمل می‌کنند و نشان می‌دهند که چگونه LLM ها نگارش دستی وظایف را در خط لوله‌های دست‌کاری سبک صنعتی کاهش می‌دهند

[Large Language Models for Robotics: Opportunities, Challenges, and Perspectives].

examples

Example 5: VoxPoser for Interactive Human-in-the-Loop Control

مثال ۵: VoxPoser برای کنترل تعاملی انسان-در-حلقه

VoxPoser به‌عنوان سیستمی ارائه می‌شود که در آن کاربران در حین اجرا، ربات را از طریق اصلاحات گفتاری زبان طبیعی هدایت می‌کنند. ربات مسیر دست‌کاری خود را بر اساس بازخورد کلامی در زمان واقعی به‌روزرسانی می‌کند. این مثال بازتاب‌دهندهٔ ارتباط عملی صنعتی است، جایی که اپراتورهای غیرمتخصص می‌توانند بدون برنامه‌نویسی مجدد یا متوقف کردن سیستم، رفتار ربات را تطبیق دهند

[Large Language Models for Robotics: Opportunities, Challenges, and Perspectives].

examples

Example 6: Visual-Language Pretraining for Industrial Manipulation (R3M, LIV)

مثال ۶: پیش‌آموزش دیداری-زبانی برای دست‌کاری صنعتی (R3M)، LIV

این مقاله مدلهایی مانند R3M و LIV را مورد بحث قرار می‌دهد که بر روی داده‌های ویدئویی انسانی مقیاس‌بالا آموزش داده شده‌اند و سپس به سیستم‌های رباتیک واقعی منتقل می‌شوند. این بازنمایی‌ها نرخ موفقیت دست‌کاری را بهبود می‌دهند، در حالی که نیاز به جمع‌آوری دادهٔ خاص ربات را کاهش می‌دهند. این رویکرد به‌طور نزدیک با نیازهای صنعتی هم‌راستا است، جایی که کارایی داده و مقیاس‌پذیری محدودیت‌های حیاتی هستند

[Large Language Models for Robotics: Opportunities, Challenges, and Perspectives].

examples

Example 7: Hybrid LLM + Classical Planning Systems in Real Robots

مثال ۷: سیستم‌های ترکیبی + LLM برنامه‌ریزی کلاسیک در ربات‌های واقعی

این مقاله چندین سیستم را گزارش می‌کند که در آن‌ها LLM ها برنامه‌های سطح‌بالا تولید می‌کنند که سپس توسط برنامه‌ریزها یا کنترل‌کننده‌های کلاسیک اعتبارسنجی و اجرا می‌شوند. این معماری ترکیبی بر روی ربات‌های واقعی آزمایش شده است تا امکان‌پذیری فیزیکی و ایمنی تضمین شود. این رویکرد یک راهبرد استقرار واقع‌بینانه را نشان می‌دهد که در آن LLM ها پشته‌های رباتیک سنتی را تقویت می‌کنند — نه اینکه جایگزین آن‌ها شوند

[Large Language Models for Robotics: Opportunities, Challenges, and Perspectives].

Example 8: Long-Horizon Task Execution in Real Environments

مثال ۸: اجرای وظایف با افق زمانی بلند در محیط‌های واقعی

در نهایت، این مقاله آزمایش‌های دنیای واقعی را مستند می‌کند که در آن‌ها ربات‌ها وظایف چندمرحله‌ای با افق زمانی بلند (برای مثال، بازچینی چند شیء) را با استفاده از برنامه‌های تولیدشده توسط LLM اجرا می‌کنند. ربات‌ها زمینهٔ وظیفه را حفظ می‌کنند، از خطاها بازیابی می‌شوند و اهداف را بدون برنامه‌نویسی صریح گام‌به‌گام تکمیل می‌کنند. این مثال نشان می‌دهد که چگونه LLM ها سربار مهندسی را در گردش کارهای رباتیک پیچیده کاهش می‌دهند

[Large Language Models for Robotics: Opportunities, Challenges, and Perspectives].

Paper 2**Parsing Natural Language Sentences into Robot Actions****Example 1: Natural Language Control of a Real Humanoid Robot (NAO / Zora)**

مثال ۱: کنترل زبان طبیعی یک ربات انسان‌نمای واقعی (NAO / Zora)

این مقاله یک سیستم کاملاً پیاده‌سازی‌شده را ارائه می‌کند که در آن یک ربات انسان‌نمای واقعی (Zora)، ساخته‌شده بر پایهٔ (NAO) دستورات گفتاری زبان طبیعی کاربران را اجرا می‌کند. دستوراتی مانند حرکت دادن بازوها، پاها، سر یا بدن تجزیه می‌شوند و به اعمال فیزیکی ترجمه می‌گردند. این سیستم تعامل انسان-ربات زمان واقعی را بدون نیاز به دانستن دستورات خاص ربات یا برنامه‌نویسی توسط کاربران نشان می‌دهد

[Parsing Natural Language Sentences into Robot Actions].

Example 2: Clarification Dialogue for Incomplete Commands

مثال ۲: گفت‌وگوی شفاف‌سازی برای دستورات ناقص

وقتی کاربر یک دستور ناقص مانند «بازویت را بالا ببر» می‌دهد، ربات به‌طور خاموش شکست نمی‌خورد. در عوض، یک پرسش پیگیری (برای مثال، «چپ یا راست؟») برای شفاف‌سازی اطلاعات گم‌شده مطرح می‌کند. این رفتار بر روی ربات واقعی پیاده‌سازی و آزمایش شده است و یک حلقهٔ کنترلی مبتنی بر گفت‌وگو را که برای تعامل انسان دنیای واقعی مناسب است نشان می‌دهد

[Parsing Natural Language Sentences into Robot Actions].

Example 3: Stateful vs. Stateless Execution in Physical Robots

مثال ۳: اجرای حالت‌دار در برابر اجرای بدون حالت در ربات‌های فیزیکی

این سیستم از دو حالت اجرای واقعی پشتیبانی می‌کند. در حالت بدون حالت، ربات هر عمل را انجام می‌دهد و به وضعیت پیش‌فرض بازمی‌گردد. در حالت حالت‌دار، اعمال به‌صورت متوالی بدون بازنشانی وضعیت اجرا می‌شوند. این امر رفتارهای چندمرحله‌ای واقع‌گرایانه را ممکن می‌سازد و منعکس‌کنندهٔ این واقعیت است که ربات‌های صنعتی و خدماتی باید وضعیت داخلی را در طول دستورات حفظ کنند [Parsing Natural Language Sentences into Robot Actions].

examples

Example 4: Automatic Rejection of Unsafe or Impossible Commands

مثال ۴: رد خودکار دستورات ناایمن یا غیرممکن

ربات به‌طور فعال دستوراتی را که باعث ناپایداری فیزیکی می‌شوند رد می‌کند، مانند درخواست از ربات برای بلند کردن یک پا در حالی که از قبل روی پای دیگر ایستاده است. سیستم وضعیت فیزیکی فعلی را بررسی می‌کند و اعمال ناسازگار را همراه با یک توضیح گفتاری رد می‌کند. این یک سازوکار ایمنی عینی است که بر روی سخت‌افزار واقعی پیاده‌سازی و اعتبارسنجی شده است [Parsing Natural Language Sentences into Robot Actions].

examples

Example 5: Execution of Compound Natural Language Commands

مثال ۵: اجرای دستورات مرکب زبان طبیعی

این سیستم می‌تواند دستورات مرکب شامل چندین عمل را در یک جمله تجزیه و اجرا کند. برای مثال، دستوراتی مانند «بلند شو و بازوهایت را پایین بیاور» به اعمال متوالی ربات تجزیه می‌شوند. این قابلیت مستقیماً بر روی ربات آزمایش شده است و نشان می‌دهد که چگونه زبان طبیعی می‌تواند رفتارهای پیچیده را بدون اسکرپت‌نویسی کنترل کند [Parsing Natural Language Sentences into Robot Actions].

examples

Example 6: Ontology-Driven Mapping Between Language and Robot Actions

مثال ۶: نگاشت مبتنی بر آنتولوژی بین زبان و اعمال ربات

یک آنتولوژی دامنه‌ای به‌طور صریح اجزای بدن ربات، اعمال و ناسازگاری‌ها را مدل می‌کند و امکان نگاشت قابل اعتماد از زبان به حرکت را فراهم می‌سازد. این آنتولوژی شامل دهه‌ها موجودیت عمل و عضو بدن است و در زمان اجرا برای تعیین اجرای معتبر پرس‌وجو می‌شود. این رویکرد بازتاب‌دهندهٔ یک راه‌حل مهندسی قابل استقرار است، نه یک نمونهٔ صرفاً مبتنی بر یادگیری [Parsing Natural Language Sentences into Robot Actions].

examples

Example 7: Cloud-Based NLP Pipeline for Real-Time Robot Control

مثال ۷: خط لوله NLP مبتنی بر ابر برای کنترل ربات در زمان واقعی

ربات گفتار را به صورت محلی ضبط می کند، آن را به متن تبدیل می کند و برای تجزیه و استدلال به یک موتور NLP مبتنی بر ابر ارسال می کند. سپس برنامه عمل حاصل بازگردانده می شود و بر روی ربات فیزیکی اجرا می گردد. این معماری بازتاب دهنده استقراریهای صنعتی واقعی است که در آن محاسبات سنگین به خدمات ابری واگذار می شوند

[Parsing Natural Language Sentences into Robot Actions].

examples

A Review of Natural-Language-Instructed Robot Execution Systems

Example 1: Voice-Controlled Industrial Robot Arms

مثال ۱: بازوهای ربات صنعتی کنترل شده با صدا

این مرور چندین سیستم را توصیف می کند که در آن بازوهای ربات صنعتی از طریق دستورات گفتاری زبان طبیعی کنترل می شوند. اپراتورها می توانند فرمان هایی مانند «start»، «stop» یا «move faster» را صادر کنند. این سیستم ها جایگزین واسطه های کنترلی سنتی مانند جوی استیک ها و پنل های کنترلی شده اند و به اپراتورها اجازه می دهند ربات ها را بدون استفاده از دست در محیط های صنعتی واقعی کنترل کنند.

Example 2: Natural Language Programming of Assembly Tasks

مثال ۲: برنامه نویسی وظایف مونتاژ با زبان طبیعی

این مقاله سیستم هایی را بررسی می کند که به کاربران اجازه می دهند وظایف مونتاژ صنعتی را با استفاده از زبان طبیعی برنامه ریزی کنند. دستوراتی مانند «pick up the part and insert it into the slot» به توالی ای از اعمال اجرایی ربات تجزیه می شوند. این رویکرد نیاز به برنامه نویسی دستی را کاهش می دهد و در محیط های شبیه سازی شده و واقعی آزمایش شده است.

Example 3: Language-Guided Navigation in Mobile Robots

مثال ۳: ناوبری هدایت شده با زبان در ربات های متحرک

سیستم های ناوبری توصیف شده در این مرور می توانند دستورات زبانی مانند «go to the next room» یا «turn left in the hallway» را دنبال کنند. این سیستم ها زبان فضایی را به مسیرهای حرکتی ربات نگاشت می کنند. این قابلیت ها در محیط های داخلی واقعی آزمایش شده اند.

Example 4: Resolving Spatial References in Physical Environments

مثال ۴: حل ارجاعات مکانی در محیط‌های فیزیکی

ربات‌ها قادرند ارجاعات مکانی مانند «next to the table»، «behind the chair» یا «in front of the door» را در محیط‌های فیزیکی واقعی تفسیر کنند. این سیستم‌ها می‌توانند بین چند شیء مشابه تمایز قائل شوند. این قابلیت برای اجرای دقیق دستورات زبان طبیعی ضروری است.

Example 5: Multi-Step Task Execution from Single Utterances

مثال ۵: اجرای وظایف چندمرحله‌ای از یک گفتار واحد

این مرور سیستم‌هایی را گزارش می‌کند که می‌توانند دستوراتی را که شامل چندین مرحله هستند و در یک جمله بیان می‌شوند اجرا کنند. این دستورات به مجموعه‌ای از اعمال متوالی تقسیم می‌شوند که به ترتیب اجرا می‌گردند. اجرای موفق این وظایف بر روی ربات‌های واقعی نشان داده شده است.

Example 6: Natural Language Interfaces for Assistive Robots

مثال ۶: رابط‌های زبان طبیعی برای ربات‌های کمکی

ربات‌های کمکی بررسی شده در این مقاله از زبان طبیعی برای تعامل با کاربران سالمند یا دارای ناتوانی استفاده می‌کنند. کاربران می‌توانند درخواست‌هایی مانند کمک در حرکت یا آوردن اشیاء را به صورت گفتاری بیان کنند. این سیستم‌ها در محیط‌های مراقبتی واقعی مستقر شده‌اند.

Example 7: Error Detection and Recovery during Execution

مثال ۷: تشخیص و بازیابی خطا در حین اجرا

برخی از سیستم‌های مرور شده شامل سازوکارهایی برای تشخیص خطا در زمان اجرا هستند. اگر اجرای یک دستور با شکست مواجه شود، سیستم می‌تواند فرآیند بازیابی را آغاز کند یا دستور جایگزینی تولید نماید. این ویژگی قابلیت اطمینان سیستم را افزایش می‌دهد.

Example 8: Integration with Classical Robotics Pipelines

مثال ۸: یکپارچه‌سازی با خط لوله‌های کلاسیک رباتیک

این مرور نشان می‌دهد که سیستم‌های هدایت شده با زبان طبیعی معمولاً با مؤلفه‌های کلاسیک رباتیک مانند برنامه‌ریزی مسیر، مکان‌یابی و کنترل حرکت یکپارچه می‌شوند. این یکپارچه‌سازی برای استقرار عملی در سیستم‌های رباتیک موجود ضروری است.

Example 9: Evaluation in Realistic Task Scenarios

مثال ۹: ارزیابی در سناریوهای وظیفه‌ای واقع‌گرایانه

سیستم‌های بررسی شده با استفاده از سناریوهای واقع‌گرایانه‌ای که شامل ناوبری، دست‌کاری و تعامل اجتماعی هستند ارزیابی شده‌اند. تمرکز این ارزیابی‌ها بر موفقیت عملی وظایف بوده است، نه صرفاً معیارهای زبانی.

Example 10: Challenges of Language Ambiguity

مثال ۱۰: چالش‌های ابهام زبانی

این مرور به چالش‌های ناشی از ابهام زبانی طبیعی در اجرای ربات اشاره می‌کند. جملات زبانی طبیعی می‌توانند چندمعنا باشند و نیاز به راهبردهای حل ابهام دارند. این مسئله به‌عنوان یکی از چالش‌های اصلی برای استقرار گسترده سیستم‌های رباتیک هدایت‌شده با زبان مطرح می‌شود.

Domain-Specific Fine-Tuning of Large Language Models for Interactive Robot Programming

Example 1: Natural Language Assistant Integrated into an Industrial Robot IDE

مثال ۱: دستیار زبان طبیعی یکپارچه‌شده در محیط توسعه ربات صنعتی

این مقاله یک دستیار زبان طبیعی را توصیف می‌کند که به‌صورت مستقیم در محیط توسعه یکپارچه ربات صنعتی ادغام شده است. کاربران می‌توانند پرسش‌های خود را به زبان طبیعی مطرح کنند و پاسخ‌هایی دریافت کنند که مستقیماً به توابع، مهارت‌ها و برنامه‌های ربات مرتبط هستند. این دستیار در یک محیط صنعتی واقعی مورد استفاده قرار گرفته است.

Example 2: Explaining Robot Skills Using Natural Language

مثال ۲: توضیح مهارت‌های ربات با استفاده از زبان طبیعی

این سیستم قادر است مهارت‌های پیچیده ربات را به زبان طبیعی توضیح دهد. برای مثال، مهارت‌هایی مانند «MoveTo»، «Grasp» و «Place» با توضیحاتی ارائه می‌شوند که رفتار فیزیکی ربات را شرح می‌دهند. این توضیحات به کاربران کمک می‌کند تا عملکرد ربات را بدون مراجعه به مستندات فنی طولانی درک کنند.

Example 3: Step-by-Step Descriptions of Robot Program Execution

مثال ۳: توصیف گام‌به‌گام اجرای برنامه ربات

برای هر برنامه ربات، دستیار می‌تواند یک توضیح گام‌به‌گام از نحوه اجرای آن تولید کند. این توضیحات شامل ترتیب اجرای مهارت‌ها و تعامل ربات با محیط است. این قابلیت به کاربران امکان می‌دهد تا رفتار ربات را پیش از اجرای واقعی بررسی کنند.

Example 4: Answering Domain-Specific Programming Questions

مثال ۴: پاسخ به پرسش‌های برنامه‌نویسی وابسته به دامنه

این دستیار می‌تواند به پرسش‌های تخصصی مربوط به دامنه برنامه‌نویسی ربات پاسخ دهد، مانند اینکه چه زمانی باید از یک مهارت خاص استفاده شود. پاسخ‌ها بر اساس دانش دامنه‌ای استخراج‌شده از مستندات صنعتی واقعی تولید می‌شوند. این امر به کاهش خطاهای برنامه‌نویسی کمک می‌کند.

Example 5: Fine-Tuning with Limited Industrial Data

مثال ۵: تنظیم دقیق با داده صنعتی محدود

این مقاله نشان می‌دهد که مدل زبانی با استفاده از مجموعه محدودی از داده‌های صنعتی تنظیم دقیق شده است. این داده‌ها شامل جفت‌های پرسش-پاسخ استخراج‌شده از مستندات واقعی هستند. نتایج نشان می‌دهند که حتی با داده محدود نیز می‌توان عملکرد مدل را در یک دامنه خاص به‌طور قابل توجهی بهبود داد.

Example 6: Improving Accuracy over General-Purpose Language Models

مثال ۶: بهبود دقت نسبت به مدل‌های زبانی عمومی

مدل تنظیم‌دقیق‌شده در مقایسه با مدل‌های زبانی عمومی، پاسخ‌های دقیق‌تر و مرتبط‌تری برای پرسش‌های برنامه‌نویسی ربات تولید می‌کند. این بهبود به‌ویژه در پرسش‌هایی که به جزئیات مهارت‌ها و محدودیت‌های ربات مربوط می‌شوند مشاهده می‌شود. این نتایج اهمیت تنظیم دقیق وابسته به دامنه را نشان می‌دهند.

Example 7: Interactive Debugging of Robot Programs

مثال ۷: اشکال‌زدایی تعاملی برنامه‌های ربات

کاربران می‌توانند از دستیار برای اشکال‌زدایی برنامه‌های ربات استفاده کنند. دستیار توضیح می‌دهد که چرا یک برنامه ممکن است به‌درستی اجرا نشود و پیشنهادهایی برای اصلاح آن ارائه می‌دهد. این فرآیند اشکال‌زدایی به‌صورت تعاملی و با زبان طبیعی انجام می‌شود.

Example 8: Consistent Use of Domain Terminology

مثال ۸: استفاده سازگار از اصطلاحات دامنه‌ای

مدل تنظیم‌دقیق‌شده به‌طور سازگار از اصطلاحات فنی خاص دامنه رباتیک استفاده می‌کند. این سازگاری باعث می‌شود پاسخ‌ها برای مهندسان رباتیک قابل اعتمادتر و قابل فهم‌تر باشند. این ویژگی در محیط‌های صنعتی اهمیت ویژه‌ای دارد.

Example 9: Reducing the Learning Curve for New Robot Programmers

مثال ۹: کاهش منحنی یادگیری برای برنامه‌نویسان جدید ربات

با استفاده از این دستیار زبان طبیعی، کاربران جدید می‌توانند سریع‌تر با برنامه‌نویسی ربات آشنا شوند. دستیار توضیحات و راهنمایی‌هایی ارائه می‌دهد که فرآیند یادگیری را تسهیل می‌کند. این موضوع می‌تواند زمان آموزش در محیط‌های صنعتی را کاهش دهد.

Example 10: Deployment in a Real Industrial Development Environment

مثال ۱۰: استقرار در یک محیط توسعه صنعتی واقعی

این سیستم در یک محیط توسعه صنعتی واقعی مستقر و ارزیابی شده است. نتایج نشان می‌دهند که دستیار می‌تواند به‌طور مؤثر از برنامه‌نویسان ربات در وظایف روزمره پشتیبانی کند. این مثال نشان‌دهنده قابلیت استفاده عملی از مدل‌های زبانی تنظیم‌دقیق شده در صنعت است.

RT-Grasp: Reasoning Tuning Robotic Grasping via Multi-modal Large Language Model

Example 1: Direct Numerical Grasp Prediction Using a Multi-Modal LLM

مثال ۱: پیش‌بینی عددی مستقیم گرفتن با استفاده از یک LLM چندوجهی

RT-Grasp سیستمی را معرفی می‌کند که در آن یک مدل زبانی بزرگ چندوجهی به‌طور مستقیم موقعیت‌های عددی گرفتن $[x, y, \theta]$ را $[x, y, \theta]$ پیش‌بینی می‌کند، نه برنامه‌های سطح بالا یا توصیف‌های متنی. این پیش‌بینی‌ها مستقیماً به اعمال گرفتن ربات ترجمه می‌شوند. این رویکرد یکی از محدودیت‌های اصلی سیستم‌های پیشین مبتنی بر LLM در رباتیک را برطرف می‌کند.

Example 2: Real-World Robot Experiments on a Franka Emika Panda Arm

مثال ۲: آزمایش‌های ربات واقعی بر روی بازوی Franka Emika Panda

نویسندگان RT-Grasp را بر روی یک ربات Franka Emika Panda با ۷ درجه آزادی که به یک گریپر موازی و یک دوربین RGB مجهز است ارزیابی می‌کنند. این ربات وظایف گرفتن را بر روی مجموعه‌ای از اشیای خانگی انجام می‌دهد. این آزمایش‌ها به‌صورت کامل بر روی سخت‌افزار واقعی انجام شده‌اند، نه در شبیه‌سازی.

Example 3: Generalization to Unseen Household Objects

مثال ۳: تعمیم به اشیای خانگی دیده‌نشده

تمام اشیای مورد استفاده در آزمایش‌ها در طول فرآیند آموزش دیده نشده بودند. این اشیای شامل اقلام خانگی روزمره با اشکال و اندازه‌های متفاوت هستند RT-Grasp. توانست این اشیای را با موفقیت بگیرد، که نشان‌دهندهٔ قابلیت تعمیم سیستم به اشیای جدید در محیط‌های واقعی است.

Example 4: Reasoning Before Grasp Generation

مثال ۴: استدلال پیش از تولید گرفتن

RT-Grasp یک مرحلهٔ استدلال صریح را معرفی می‌کند که در آن مدل ابتدا دربارهٔ استراتژی مناسب گرفتن بر اساس نوع و شکل شیء استدلال می‌کند. تنها پس از این استدلال، مختصات عددی گرفتن تولید می‌شوند. این فرآیند به بهبود پایداری و ایمنی گرفتن کمک می‌کند.

Example 5: Avoiding Unsafe Grasp Regions Through Language Guidance

مثال ۵: اجتناب از نواحی ناایمن گرفتن از طریق هدایت زبانی

سیستم به کاربران اجازه می‌دهد که از زبان طبیعی برای هدایت رفتار گرفتن استفاده کنند. برای مثال، وقتی به مدل گفته می‌شود از ناحیهٔ خاصی از شیء اجتناب کند، RT-Grasp یک موقعیت گرفتن جدید تولید می‌کند. این اصلاح به صورت مستقیم در پیش‌بینی عددی منعکس می‌شود.

Example 6: Multi-Modal Input Combining Vision and Language

مثال ۶: ورودی چندوجهی ترکیبی از بینایی و زبان

RT-Grasp از ورودی‌های چندوجهی شامل تصاویر RGB و دستورالعمل‌های زبانی استفاده می‌کند. مدل این ورودی‌ها را به طور هم‌زمان پردازش می‌کند تا تصمیمات گرفتن را تولید کند. این ترکیب اطلاعات دیداری و زبانی به مدل امکان می‌دهد تا گرفتن‌های آگاه از زمینه تولید کند.

Example 7: Improved Grasp Success Rate over Baseline Models

مثال ۷: بهبود نرخ موفقیت گرفتن نسبت به مدل‌های پایه

نتایج تجربی نشان می‌دهند که RT-Grasp در مقایسه با مدل‌های پایهٔ موجود، نرخ موفقیت گرفتن بالاتری دارد. این بهبود هم در شبیه‌سازی و هم در آزمایش‌های دنیای واقعی مشاهده شده است. این نتایج اثربخشی تنظیم مبتنی بر استدلال را نشان می‌دهند.

Example 8: Interpretable Reasoning Outputs for Debugging

مثال ۸: خروجی‌های استدلال قابل تفسیر برای اشکال‌زدایی

مرحلهٔ استدلال مدل خروجی‌های متنی قابل تفسیر تولید می‌کند که توضیح می‌دهند چرا یک موقعیت گرفتن خاص انتخاب شده است. این خروجی‌ها می‌توانند برای اشکال‌زدایی و تحلیل رفتار سیستم مورد استفاده قرار گیرند. این ویژگی شفافیت سیستم را افزایش می‌دهد.

Example 9: Robust Performance under Cluttered Scene Conditions

مثال ۹: عملکرد مقاوم در شرایط صحنه‌های شلوغ

در RT-Grasp صحنه‌های شلوغ که چندین شیء در نزدیکی یکدیگر قرار دارند ارزیابی شده است. سیستم توانست اشیای هدف را با دقت شناسایی کرده و گرفتن‌های موفق انجام دهد. این قابلیت برای کاربردهای واقعی اهمیت دارد.

Example 10: End-to-End Deployment from Perception to Action

مثال ۱۰: استقرار سرتاسری از ادراک تا عمل

این سیستم به‌صورت سرتاسری از دریافت ورودی‌های حسی تا اجرای عمل گرفتن بر روی ربات پیاده‌سازی شده است. کل خط لوله بدون مداخلهٔ انسانی اجرا می‌شود. این مثال نشان‌دهندهٔ امکان استقرار عملی RT-Grasp در سیستم‌های رباتیک واقعی است.

ARRC: Advanced Reasoning Robot Control — Knowledge-Driven Autonomous Manipulation Using Retrieval-Augmented Generation

Example 1: Knowledge-Driven Manipulation Using External Documents

مثال ۱: دست‌کاری مبتنی بر دانش با استفاده از اسناد خارجی

ARRC سیستمی را معرفی می‌کند که از اسناد دانش خارجی برای هدایت دست‌کاری ربات استفاده می‌کند. این اسناد شامل راهنماها، توضیحات وظایف و اطلاعات ساخت‌یافته هستند. سیستم این دانش را برای تصمیم‌گیری در مورد نحوهٔ انجام اعمال فیزیکی بازیابی می‌کند.

Example 2: Retrieval-Augmented Generation for Action Planning

مثال ۲: تولید تقویت‌شده با بازیابی برای برنامه‌ریزی عمل

این سیستم از تولید تقویت‌شده با بازیابی برای تولید برنامه‌های عمل استفاده می‌کند. ابتدا اطلاعات مرتبط از یک پایگاه دانش بازیابی می‌شود. سپس مدل زبانی این اطلاعات را برای تولید یک توالی اقدامات رباتیک ترکیب می‌کند.

Example 3: Executing Multi-Step Manipulation Tasks

مثال ۳: اجرای وظایف دست‌کاری چندمرحله‌ای

ARRC قادر است وظایف دست‌کاری چندمرحله‌ای را که شامل چندین عمل متوالی هستند اجرا کند. این وظایف می‌توانند شامل گرفتن، جابه‌جایی و قرار دادن اشیاء باشند. سیستم هر مرحله را بر اساس دانش بازیابی‌شده تنظیم می‌کند.

Example 4: Reasoning Over Object Properties

مثال ۴: استدلال بر روی ویژگی‌های اشیاء

سیستم می‌تواند بر روی ویژگی‌های اشیاء مانند شکل، اندازه و شکنندگی استدلال کند. این استدلال برای انتخاب استراتژی‌های مناسب دست‌کاری استفاده می‌شود. این قابلیت به ربات اجازه می‌دهد تا رفتار خود را با ویژگی‌های فیزیکی اشیاء تطبیق دهد.

Example 5: Real-World Deployment on Manipulation Platforms

مثال ۵: استقرار در دنیای واقعی بر روی سکوها دست‌کاری

ARRC بر روی سکوها دست‌کاری رباتیک واقعی مستقر شده است. این سیستم در محیط‌های واقعی و نه صرفاً شبیه‌سازی‌شده مورد آزمایش قرار گرفته است. نتایج نشان می‌دهند که سیستم می‌تواند وظایف را با موفقیت انجام دهد.

Example 6: Handling Incomplete or Ambiguous Instructions

مثال ۶: مدیریت دستورالعمل‌های ناقص یا مبهم

سیستم می‌تواند دستورالعمل‌هایی را که ناقص یا مبهم هستند مدیریت کند. ARRC با استفاده از دانش بازیابی‌شده شکاف‌های اطلاعاتی را پر می‌کند. این قابلیت به ربات امکان می‌دهد حتی در صورت نبود اطلاعات کامل نیز عمل کند.

Example 7: Combining Symbolic Knowledge with Neural Models

مثال ۷: ترکیب دانش نمادین با مدل‌های عصبی

ARRC دانش نمادین را با مدل‌های عصبی ترکیب می‌کند. دانش نمادین ساختار و محدودیت‌ها را فراهم می‌کند، در حالی که مدل‌های عصبی انعطاف‌پذیری و تعمیم را ارائه می‌دهند. این ترکیب به بهبود عملکرد کلی سیستم کمک می‌کند.

Example 8: Adaptive Task Execution Based on Retrieved Context

مثال ۸: اجرای تطبیقی وظایف بر اساس زمینه بازیابی‌شده

اجرای وظایف بر اساس زمینه‌ای که از پایگاه دانش بازیابی شده است تطبیق داده می‌شود. سیستم می‌تواند برنامه خود را در حین اجرا تغییر دهد. این ویژگی به ربات اجازه می‌دهد به شرایط غیرمنتظره واکنش نشان دهد.

Example 9: Improved Robustness Compared to Non-RAG Systems

مثال ۹: بهبود پایداری در مقایسه با سیستم‌های بدون RAG

نتایج تجربی نشان می‌دهند که ARRC در مقایسه با سیستم‌هایی که از تولید تقویت‌شده با بازیابی استفاده نمی‌کنند، پایداری بیشتری دارد. این پایداری در اجرای وظایف پیچیده مشهود است. این نتایج مزایای استفاده از RAG را نشان می‌دهند.

Example 10: End-to-End Autonomous Manipulation Pipeline

مثال ۱۰: خط لوله دست‌کاری خودمختار سرتاسری

ARRC یک خط لوله کامل از دریافت دستور تا اجرای دست‌کاری ارائه می‌دهد. این خط لوله شامل بازیابی دانش، استدلال، برنامه‌ریزی و اجرای عمل است. کل فرآیند به صورت خودمختار انجام می‌شود.

ROS-LLM: A ROS Framework for Embodied AI with Task Feedback and Structured Reasoning

Example 1: Integration of Large Language Models into the ROS Ecosystem

مثال ۱: یکپارچه‌سازی مدل‌های زبانی بزرگ در اکوسیستم ROS

ROS-LLM یک چارچوب را معرفی می‌کند که مدل‌های زبانی بزرگ را مستقیماً در اکوسیستم ROS ادغام می‌کند. این چارچوب به ربات‌ها اجازه می‌دهد تا از توانایی‌های زبانی در کنار مؤلفه‌های استاندارد ROS استفاده کنند. این یکپارچه‌سازی امکان استفاده عملی از LLM ها را در سیستم‌های رباتیک موجود فراهم می‌کند.

Example 2: Structured Reasoning for Task Execution

مثال ۲: استدلال ساخت‌یافته برای اجرای وظایف

این چارچوب از استدلال ساخت‌یافته برای تجزیه وظایف پیچیده به مراحل قابل اجرا استفاده می‌کند. دستورات زبان طبیعی به توالی‌ای از اقدامات سطح پایین تبدیل می‌شوند. این ساختار امکان اجرای قابل اعتماد وظایف را فراهم می‌کند.

Example 3: Task Feedback Loop for Error Correction

مثال ۳: حلقه بازخورد وظیفه برای اصلاح خطا

ROS-LLM شامل یک حلقه بازخورد است که وضعیت اجرای وظیفه را پایش می‌کند. در صورت بروز خطا، سیستم بازخورد دریافت کرده و برنامه اجرا را اصلاح می‌کند. این مکانیزم به افزایش پایداری سیستم کمک می‌کند.

Example 4: Language-Based Interaction with Physical Environments

مثال ۴: تعامل مبتنی بر زبان با محیط‌های فیزیکی

ربات‌ها می‌توانند از زبان طبیعی برای تعامل با محیط‌های فیزیکی استفاده کنند. دستورات زبانی به اقدامات حرکتی و کنترلی نگاشت می‌شوند. این قابلیت تعامل انسان-ربات را ساده‌تر می‌کند.

Example 5: Modular Design for Easy Extension

مثال ۵: طراحی ماژولار برای توسعه آسان

چارچوب ROS-LLM به‌صورت ماژولار طراحی شده است. این طراحی به توسعه‌دهندگان اجازه می‌دهد مؤلفه‌های جدید را به‌راحتی اضافه یا جایگزین کنند. این ویژگی انعطاف‌پذیری سیستم را افزایش می‌دهد.

Example 6: Real-Time Execution on Mobile and Manipulation Robots

مثال ۶: اجرای بلادرنگ بر روی ربات‌های متحرک و دست‌کار

این چارچوب بر روی ربات‌های متحرک و ربات‌های دست‌کار اجرا شده است. اجرای وظایف به‌صورت بلادرنگ انجام می‌شود. این آزمایش‌ها در محیط‌های فیزیکی واقعی انجام شده‌اند.

Example 7: Bridging High-Level Language and Low-Level Control

مثال ۷: پل زدن بین زبان سطح بالا و کنترل سطح پایین

ROS-LLM به‌عنوان واسطی بین دستورات زبانی سطح بالا و کنترل سطح پایین ربات عمل می‌کند. این واسط به ترجمه مفاهیم انتزاعی به فرمان‌های کنترلی دقیق کمک می‌کند. این قابلیت برای استفاده عملی از زبان طبیعی ضروری است.

Example 8: Compatibility with Existing ROS Packages

مثال ۸: سازگاری با بسته‌های موجود ROS

این چارچوب با بسته‌های استاندارد ROS سازگار است. توسعه‌دهندگان می‌توانند ROS-LLM را بدون نیاز به تغییرات اساسی در سیستم‌های موجود به‌کار گیرند. این سازگاری پذیرش چارچوب را تسهیل می‌کند.

Example 9: Demonstrations in Realistic Robotic Tasks

مثال ۹: نمایش‌ها در وظایف رباتیک واقع‌گرایانه

ROS-LLM در مجموعه‌ای از وظایف رباتیک واقع‌گرایانه نمایش داده شده است. این وظایف شامل ناوبری، دست‌کاری و تعامل با اشیاء هستند. این نمایش‌ها قابلیت کاربردی چارچوب را نشان می‌دهند.

Example 10: Embodied AI System with Closed-Loop Control

مثال ۱۰: سیستم هوش تجسم‌یافته با کنترل حلقه‌بسته

این چارچوب یک سیستم هوش تجسم‌یافته با کنترل حلقه‌بسته ارائه می‌دهد. وضعیت محیط به‌طور مداوم پایش می‌شود و اقدامات بر اساس آن تنظیم می‌گردند. این ویژگی امکان تعامل پایدار با دنیای فیزیکی را فراهم می‌کند.

9. A Multimodal LLM-Driven Robotic Control System for Adaptive Industrial Manipulation: Integrating Vision-Language Models for Enhanced Manufacturing Flexibility

در ادامه، ترجمه دقیق و واژه‌به‌واژه متن ارائه شده به زبان فارسی آورده شده است:

۹. یک سیستم کنترل رباتیک مبتنی بر مدل زبانی بزرگ (LLM) چندوجهی برای جابجایی صنعتی تطبیقی: یکپارچه‌سازی مدل‌های بینایی-زبان برای افزایش انعطاف‌پذیری تولید

مثال ۱: تفسیر دستورات چندوجهی برای کنترل ربات واقعی این مقاله سیستم رباتیکی را ارائه می‌دهد که زبان، بینایی و وضعیت ربات را برای تفسیر دستورات کاربر ترکیب می‌کند. ربات به جای اتکا بر متن به‌تنهایی، ورودی دوربین و داده‌های حسگر را در کنار دستورالعمل‌های زبان طبیعی پردازش می‌کند. این امر ربات را قادر می‌سازد تا دستوراتی مانند ارجاع به اشیاء و روابط فضایی در محیط‌های واقعی را به‌درستی درک کند، که نشان‌دهنده یک خط لوله (pipeline) کنترل چندوجهی یکپارچه است.

مثال ۲: جابجایی هدایت‌شده با زبان در محیط‌های فیزیکی این سیستم در وظایف جابجایی واقعی ارزیابی می‌شود، جایی که ربات اقداماتی مانند برداشتن، حرکت دادن و قرار دادن اشیاء را بر اساس دستورالعمل‌های زبان طبیعی انجام می‌دهد. مدل زبانی بزرگ پیش از تولید توالی‌های عملیاتی قابل اجرا، درباره ویژگی‌های شیء و اهداف وظیفه استدلال می‌کند. این موضوع استفاده کاربردی از مدل‌های زبانی بزرگ را برای کاهش برنامه‌نویسی دستی در وظایف جابجایی نشان می‌دهد.

مثال ۳: انطباق (Grounding) دستورالعمل‌های زبان طبیعی بر پایه بینایی یک مشارکت کلیدی، منطبق کردن زبان در مشاهدات بصری واقعی است. هنگامی که کاربر با استفاده از زبان توصیفی (مانند رنگ یا موقعیت) به اشیاء اشاره می‌کند، ربات با استفاده از ورودی دوربین، هدف صحیح را شناسایی می‌کند. این کار شکاف بین دستورالعمل‌های انتزاعی و انتخاب اشیاء فیزیکی را پر می‌کند که یک نیاز حیاتی برای استقرار رباتیک در دنیای واقعی است.

مثال ۴: تولید عملیات ساختاریافته برای اجرای ایمن مدل زبانی بزرگ به جای صدور فرمان‌های حرکتی خام، بازنمایی‌های عملیاتی سطح بالا و ساختاریافته‌ای تولید می‌کند که به کنترل‌کننده‌های کلاسیک ارسال می‌شوند. این جداسازی به ربات اجازه می‌دهد تا ضمن حفظ کنترل سطح پایین قابل اطمینان، از استدلال مدل زبانی بزرگ بهره‌مند شود. این رویکرد نشان‌دهنده چگونگی ادغام ایمن مدل‌های زبانی بزرگ در پشته‌های (stacks) رباتیک موجود است.

مثال ۵: باز خورد حلقه-بسته بین اجرا و استدلال سیستم به طور مداوم نتایج اجرا را به حلقه استدلال بازمی گرداند. اگر عملیاتی شکست بخورد یا محیط تغییر کند، اطلاعات حسی به روز شده در اختیار مدل زبانی بزرگ قرار می گیرد که بر اساس آن برنامه را بازنگری می کند. این طراحی حلقه-بسته به جای اجرای دستور تک مرحله ای، امکان تاب آوری در محیط های واقعی و پویا را فراهم می کند.

مثال ۶: رسیدگی به خطا از طریق آگاهی از بافتار چندوجهی هنگامی که اجرا طبق انتظار پیش نمی رود - مانند زمانی که یک شیء قابل دسترسی نیست - سیستم مشکل را از طریق بازخورد حسگر تشخیص می دهد. سپس مدل زبانی بزرگ با استفاده از بافتار زبانی و ورودی بصری درباره شکست استدلال می کند و به آن اجازه می دهد تا به جای متوقف شدن، استراتژی را تنظیم کند. این موضوع منعکس کننده نیازهای صنعتی واقعی برای سیستم های رباتیک خطا پذیر (fault-tolerant) است.

مثال ۷: تعامل انسان و ربات بدون دانش تخصصی مقاله تأکید می کند که کاربران می توانند بدون درک سینماتیک ربات یا پارامترهای کنترلی، با استفاده از زبان طبیعی با ربات تعامل داشته باشند. مدل زبانی بزرگ چندوجهی، پیچیدگی های فنی را انتزاع (abstract) می کند و تعامل شهودی انسان و ربات را امکان پذیر می سازد. این امر به ویژه برای رباتیک خدماتی و محیط های صنعتی همکارانه مرتبط است.

مثال ۸: یکپارچه سازی ماژولار مدل های زبانی بزرگ در معماری های رباتیک سیستم پیشنهادی به عنوان یک معماری ماژولار طراحی شده است، جایی که ادراک، استدلال و کنترل به صورت سست با هم جفت شده اند. (loosely coupled) مدل زبانی بزرگ به عنوان یک ماژول استدلال سطح بالا عمل می کند و نه یک کنترل کننده یکپارچه. (monolithic) این انتخاب طراحی از مقیاس پذیری پشتیبانی کرده و با محدودیت های استقرار در دنیای واقعی همسو است.

در ادامه، ترجمه واژه به واژه متن ارائه شده به زبان فارسی آورده شده است:

۱۰. A Survey of LLM-Driven AI Agent Communication: Protocols, Security Risks, and Defense Countermeasures

مثال ۱ AutoGPT: برای اجرای وظایف خودمختار از طریق ارتباطات عامل (Agent) این پیمایش، AutoGPT را به عنوان سیستمی مورد بحث قرار می دهد که در آن چندین عامل مبتنی بر مدل زبانی بزرگ، از طریق تبادل پیام های ساختاریافته برای تجزیه و اجرای وظایف پیچیده با یکدیگر ارتباط برقرار می کنند. عامل ها بدون مداخله انسان، اهداف، نتایج میانی و بازخوردها را با هم مبادله می کنند. این سیستم به طور کاربردی برای وظایف طولانی مدت مانند تولید نرم افزار و تحلیل داده ها اثبات شده است که هماهنگی چندعاملی واقعی را با هدایت پروتکل های ارتباطی نشان می دهد.

مثال ۲ BabyAGI: برای بهبود تکرارپذیری وظایف BabyAGI به عنوان یک چارچوب عاملی معرفی شده است که در آن، عامل ها وضعیت وظایف، اولویت ها و نتایج را در یک حلقه تکرارپذیر به یکدیگر ابلاغ می کنند. هر عامل یک فهرست وظایف مشترک را بر اساس بازخورد اجرا به روزرسانی می کند. این مکانیسم ارتباطی، بهبود مستمر وظایف را امکان پذیر می سازد و در سیستم های واقعی که جریان های کاری برنامه ریزی و اجرای خودمختار را انجام می دهند، پیاده سازی شده است.

مثال ۳ CAMEL: برای ارتباطات عاملی مبتنی بر نقش این پیمایش بر سیستم CAMEL تأکید می کند؛ سیستمی که در آن به عامل ها نقش های صریحی (مانند دستیار و کاربر) اختصاص داده می شود و از طریق گفتگوی ساختاریافته با هم ارتباط برقرار می کنند. این پروتکل مبتنی بر نقش، هماهنگی و عملکرد وظایف را در آزمایش های واقعی مانند حل مسئله و تولید کد بهبود می بخشد. این رویکرد نشان می دهد که چگونه پروتکل های ارتباطی کنترل شده، ابهام را در سیستم های چندعاملی کاهش می دهند.

مثال ۴: AutoGen برای همکاری چندعاملی در کاربردهای واقعی AutoGen به عنوان چارچوبی توصیف شده است که در آن چندین عامل مدل زبانی بزرگ از طریق قوانین گفتگوی از پیش تعریف شده برای حل وظایف به صورت مشارکتی با هم ارتباط برقرار می کنند. عامل ها در برنامه ریزی، اجرا یا تأیید تخصص دارند و برای رسیدن به یک راه حل نهایی، پیام مبادله می کنند. این چارچوب به طور کاربردی در توسعه نرم افزار، پردازش داده ها و جریان های کاری مبتنی بر ابزار اعمال شده است.

مثال ۵: ChatDev برای مهندسی نرم افزار از طریق تیم های عاملی این پیمایش سیستم ChatDev را مورد بحث قرار می دهد؛ سیستمی که یک شرکت نرم افزاری مجازی متشکل از عامل های مدل زبانی بزرگ با نقش های متمایز را شبیه سازی می کند. عامل ها با استفاده از پروتکل های ساختاریافته برای طراحی، پیاده سازی، آزمایش و اشکال زدایی نرم افزار با یکدیگر ارتباط برقرار می کنند. این سیستم کاربرپذیری دنیای واقعی ارتباطات عاملی را در وظایف مهندسی هماهنگ نشان می دهد.

مثال ۶: عامل های سبک ReAct با ارتباطات صریح «فکر-عمل» این مقاله عامل های مبتنی بر ReAct را بررسی می کند که رد پای استدلال را با فرمان های عملیاتی از طریق یک قالب ارتباطی ساختاریافته در هم می آمیزند. این عامل ها به صورت تجربی در وظایف تصمیم گیری واقعی که نیاز به استفاده از ابزار و تعامل با محیط دارند، اعتبارسنجی شده اند. ساختار ارتباطی صریح، قابلیت اطمینان و شفافیت را در حین اجرا بهبود می بخشد.

مثال ۷: عامل های فراخوان ابزار با استفاده از پروتکل های مبتنی بر پیام این پیمایش سیستم هایی را گزارش می دهد که در آن ها عامل ها از طریق پیام های ساختاریافته برای فراخوانی ابزارهای خارجی مانند API ها، پایگاه های داده یا شبیه سازها ارتباط برقرار می کنند. هر عامل درخواست های ابزار را ارسال کرده و پاسخ های ساختاریافته دریافت می کند که یک سیستم اجرای حلقه-بسته را تشکیل می دهد. این پروتکل در جریان های کاری خودمختار واقعی از جمله بازیابی داده ها و اتوماسیون وظایف اعمال شده است.

مثال ۸: هماهنگی چندعاملی مبتنی بر تخته سیاه (Blackboard) برخی از سیستم های بررسی شده از یک معماری تخته سیاه مشترک استفاده می کنند که در آن عامل ها مشاهدات، برنامه ها و نتایج را برای استفاده سایر عامل ها ارسال می کنند. این مکانیسم ارتباطی از استدلال موازی و هماهنگی در وظایف پیچیده پشتیبانی می کند. این رویکرد در محیط های چندعاملی آزمایشی واقعی که نیاز به تصمیم گیری مشارکتی دارند، پیاده سازی شده است.

مثال ۹: ارتباطات عاملی مجهز به حافظه برای وظایف طولانی مدت این مقاله درباره عامل هایی بحث می کند که از طریق ساختارهای حافظه مشترک با هم ارتباط برقرار می کنند و اجازه می دهند بافتار (context) وظیفه در افق های زمانی طولانی پایداری داشته باشد. این پروتکل های ارتباطی مجهز به حافظه، عامل ها را قادر می سازند تا وظایف را از سر بگیرند، پیشرفت را ردیابی کنند و استراتژی ها را تطبیق دهند. چنین سیستم هایی در آزمایش های عامل خودمختار طولانی مدت به نمایش گذاشته شده اند.

مثال ۱۰: ارتباطات عاملی سلسله مراتبی برای سیستم های مقیاس پذیر این پیمایش پروتکل های ارتباطی سلسله مراتبی را ارائه می دهد که در آن عامل های برنامه ریز سطح بالا، اهداف را به عامل های اجراکننده سطح پایین ابلاغ می کنند. این ساختار در چیدمان های چندعاملی واقعی برای بهبود مقیاس پذیری و تجزیه وظایف آزمایش شده است. این مدل منعکس کننده ساختارهای سازمانی مورد استفاده در سیستم های صنعتی و رباتیک است.

۱۱) Enhancing reliability in LLM-integrated robotic systems A unified approach .

مثال ۱: برنامه ریزی مدل زبانی بزرگ آگاه از قابلیت اطمینان برای اجرای ربات فیزیکی این مقاله یک خط لوله (pipeline) کنترل رباتیک را ارائه می دهد که در آن برنامه های تولید شده توسط مدل زبانی بزرگ، پیش از اجرا به طور صریح از نظر قابلیت اطمینان بررسی

می‌شوند. برنامه‌ها در برابر محدودیت‌های فیزیکی، امکان‌سنجی وظیفه و تاریخچه اجرا اعتبارسنجی می‌شوند. این کار از ارسال اقدامات نایمن یا بدتعریف‌شده به ربات جلوگیری می‌کند که نشان‌دهنده یک محافظ قابل استقرار برای سیستم‌های رباتیک واقعی است.

مثال ۲: پایش زمان اجرا (Runtime) برای اقدامات ربات هدایت‌شده با مدل زبانی بزرگ یک ماژول پایش زمان اجرا به‌طور مداوم اجرای ربات را مشاهده کرده و آن را با برنامه در نظر گرفته شده‌ی مدل زبانی بزرگ مقایسه می‌کند. اگر انحرافی رخ دهد - مانند شکست در برداشتن اشیاء یا حرکت غیرمنتظره شیء - سیستم ناهماهنگی را تشخیص می‌دهد. این امر به جای شکست خاموش، امکان مداخله در زمان واقعی را فراهم می‌کند که برای استقرار رباتیک در دنیای واقعی حیاتی است.

مثال ۳: خود-تأییدی خروجی‌های مدل زبانی بزرگ پیش از کنترل ربات این سیستم خود-تأییدی مدل زبانی بزرگ را یکپارچه می‌کند، جایی که مدل، برنامه تولید شده‌ی خود را پیش از اجرا ارزیابی می‌کند. مدل زبانی بزرگ ثبات منطقی، مراحل مفقود شده و تخلفات ایمنی را بررسی می‌کند. این کار رسیدن دستورالعمل‌های توهّم‌آمیز یا ناقص به ربات فیزیکی را کاهش داده و قابلیت اطمینان اجرا را بدون نظارت انسانی بهبود می‌بخشد.

مثال ۴: بازبینی پس از شکست از طریق برنامه‌ریزی مجدد تکرار شونده هنگامی که اجرا شکست می‌خورد، ربات به‌طور دائمی متوقف نمی‌شود. در عوض، بازخورد اجرا به مدل زبانی بزرگ بازگردانده می‌شود که بر اساس شکست مشاهده شده، وظیفه را دوباره برنامه‌ریزی می‌کند. این مکانیسم بازبینی حلقه-بسته در وظایف جابجایی به نمایش گذاشته شده است که نشان می‌دهد چگونه قابلیت اطمینان در طول تلاش‌های مکرر بهبود می‌یابد.

مثال ۵: جداسازی استدلال سطح بالا و کنترل سطح پایین این مقاله بر یک معماری لایه‌بندی شده تأکید می‌کند که در آن مدل زبانی بزرگ فقط به استدلال سطح بالا محدود شده است. کنترل حرکت سطح پایین همچنان توسط کنترل‌کننده‌های رباتیک قطعی (deterministic) مدیریت می‌شود. این جداسازی تضمین می‌کند که مدل‌های زبانی احتمالی هرگز مستقیماً به موتورهای فرمان نمی‌دهند، که با الزامات ایمنی صنعت همسو است.

مثال ۶: بازنمایی عملیاتی ساختاریافته برای جلوگیری از ابهام به جای متن با قالب آزاد، خروجی‌های مدل زبانی بزرگ به طرحواره‌های (schemas) عملیاتی ساختاریافته محدود می‌شوند. این طرحواره‌ها ارجاعات شیء، انواع عمل و پارامترهای اجرا را مشخص می‌کنند. این کار ابهام را کاهش داده و برنامه‌ها را برای سیستم‌های رباتیک کلاسیک قابل تفسیر می‌کند و ادغام قابل اطمینان در خطوط لوله کنترل واقعی را امکان‌پذیر می‌سازد.

مثال ۷: بررسی ثبات در وظایف چندمرحله‌ای برای وظایف با افق زمانی طولانی، سیستم تأیید می‌کند که مراحل بعدی با فرض‌های اولیه سازگار باقی بمانند. اگر وضعیت یک شیء تغییر کند یا غیرقابل دسترس شود، برنامه به عنوان نامعتبر علامت‌گذاری می‌شود. این کار از شکست‌های زنجیره‌ای در اجرای رباتیک چندمرحله‌ای جلوگیری می‌کند که یک مشکل رایج در استقرارهای واقعی است.

مثال ۸: فیلتر کردن وظایف حساس از نظر ایمنی این چارچوب شامل مکانیسمی برای مسدود کردن اقدامات نایمن یا پرخطر پیشنهاد شده توسط مدل زبانی بزرگ است. اقداماتی که خط‌مشی‌های ایمنی را نقض می‌کنند یا از آستانه‌های ریسک از پیش تعریف شده فراتر می‌روند، به‌طور خودکار رد می‌شوند. این موضوع نشان می‌دهد که چگونه مدل‌های زبانی بزرگ می‌توانند حتی در کاربردهای رباتیک حساس از نظر ایمنی به‌طور ایمن استفاده شوند.

مثال ۹: توضیح‌پذیری برای اشکال‌زدایی و اعتماد هر اقدام تأیید شده یا رد شده با توضیحی که توسط سیستم تولید شده همراه است. مهندسان می‌توانند بررسی کنند که چرا یک برنامه پذیرفته یا مسدود شده است. این توضیح‌پذیری برای اشکال‌زدایی سیستم‌های رباتیک واقعی و ایجاد اعتماد در خطوط لوله یکپارچه با مدل زبانی بزرگ ضروری است.

مثال ۱۰: چارچوب قابلیت اطمینان یکپارچه در میان وظایف و ربات‌ها این مقاله نشان می‌دهد که مکانیسم‌های قابلیت اطمینان مشابه در وظایف و پلتفرم‌های رباتیک مختلف اعمال می‌شوند. این نشان می‌دهد که قابلیت اطمینان به عنوان یک ویژگی در سطح سیستم در نظر گرفته می‌شود، نه یک ترفند (hack) خاص وظیفه. چنین یکپارچه‌سازی برای پذیرش صنعتی و مقیاس‌پذیر رباتیک هدایت‌شده با مدل زبانی بزرگ ضروری است.

۱۲. InCoRo In-Context Learning for Robotics Control with Feedback Loops

مثال ۱: تولید سریع رفتارهای اجرایی ربات این مقاله LLM-BRAIn را معرفی می‌کند؛ سیستمی که در آن یک مدل زبانی بزرگ، توصیفات سطح بالای وظایف را مستقیماً به رفتارهای اجرایی ربات تبدیل می‌کند. به جای طراحی دستی درخت‌های رفتار (Behavior Trees) یا اسکریپت‌های کنترلی، سیستم به‌طور خودکار آن‌ها را تولید می‌کند. این امر زمان توسعه را برای وظایف رباتیک واقعی به‌طور قابل توجهی کاهش می‌دهد و در عین حال سازگاری با چارچوب‌های کنترلی موجود را حفظ می‌کند.

مثال ۲: سنتز درخت رفتار از زبان طبیعی یک مشارکت کلیدی، تولید خودکار درخت‌های رفتار از دستورالعمل‌های زبان طبیعی است. مدل زبانی بزرگ، توصیفات وظایف را به منطق کنترلی ساختاریافته‌ای ترجمه می‌کند که ربات‌ها می‌توانند به‌طور قابل اطمینان اجرا کنند. این رویکرد در سناریوهای واقعی وظایف رباتیک ارزیابی شده است که کاربردپذیری عملی را فراتر از شبیه‌سازی نشان می‌دهد.

مثال ۳: انطباق سریع با وظایف جدید بدون برنامه‌نویسی مجدد این سیستم به ربات‌ها اجازه می‌دهد تا صرفاً با تغییر توصیف متنی وظیفه، با وظایف جدید سازگار شوند. مدل زبانی بزرگ بدون مداخله دستی، ساختار رفتاری مربوطه را دوباره تولید می‌کند. این موضوع از موارد استفاده در دنیای واقعی که ربات‌ها باید مکرراً وظایف خود را تغییر دهند (مانند محیط‌های خدماتی یا صنعتی) پشتیبانی می‌کند.

مثال ۴: یکپارچه‌سازی با خطوط لوله کنترل رباتیک کلاسیک LLM-BRAIn جایگزین کنترل‌کننده‌های سطح پایین نمی‌شود. در عوض، رفتارهای تولید شده از طریق ماژول‌های کنترلی رباتیک موجود اجرا می‌شوند. این طراحی منعکس‌کننده محدودیت‌های استقرار واقعی است و در عین حال که از انعطاف‌پذیری هدایت‌شده با مدل زبانی بزرگ بهره می‌برد، قابلیت اطمینان را نیز تضمین می‌کند.

مثال ۵: کاهش تلاش مهندسی در برنامه‌نویسی ربات مقاله نشان می‌دهد که رفتارهای تولید شده توسط مدل زبانی بزرگ، نیاز به منطق کنترلی ساخته شده با دست را کاهش می‌دهند. مهندسان می‌توانند به جای کدنویسی رفتارها، بر تعریف اهداف تمرکز کنند. این امر مستقیماً به یک گلوگاه اصلی در توسعه سیستم‌های رباتیک واقعی می‌پردازد.

مثال ۶: خروجی ساختاریافته برای جلوگیری از اجرای مبهم مدل زبانی بزرگ رفتارها را در قالبی ساختاریافته و محدود شده خروجی می‌دهد و تضمین می‌کند که رفتارهای تولید شده، قابل تفسیر و قابل اجرا باقی بمانند. این کار از خطرات تولید زبان با قالب آزاد جلوگیری کرده و با الزامات ایمنی در ربات‌های فیزیکی همسو می‌شود.

مثال ۷: مقیاس‌پذیری در پلتفرم‌های رباتیک مختلف این رویکرد به‌گونه‌ای طراحی شده است که مستقل از پلتفرم (Platform-agnostic) باشد. رفتارهای تولید شده را می‌توان در کالدهای مختلف رباتیک مجدداً استفاده کرد، مشروط بر اینکه مهارت‌های زیربنایی وجود داشته باشند. این امر از استقرار مقیاس‌پذیر در ناوگان‌های رباتیک ناهمگون پشتیبانی می‌کند.

مثال ۸: رفتارهای قابل خواندن توسط انسان و قابل اشکال‌زدایی درخت‌های رفتار تولید شده توسط انسان قابل خواندن هستند و به مهندسان اجازه می‌دهند در صورت نیاز آن‌ها را بررسی، اشکال‌زدایی و اصلاح کنند. این شفافیت برای پذیرش صنعتی، که در آن توضیح‌پذیری و تأیید مورد نیاز است، ضروری است.

مثال ۹: پشتیبانی از اجرای وظایف با افق زمانی طولانی سیستم می‌تواند رفتارهایی تولید کند که شامل توالی‌بندی، شرط‌ها و اقدامات بازیابی باشد. این امر ربات‌ها را قادر می‌سازد تا به جای اقدامات مجزا، وظایف چندمرحله‌ای را انجام دهند. چنین اجرای طولانی‌مدتی برای جریان‌های کاری رباتیک در دنیای واقعی حیاتی است.

مثال ۱۰: پیوند دادن استدلال سطح وظیفه و اجرا LLM-BRAIn به عنوان پلی بین توصیفات انتزاعی وظیفه و اجرای عینی ربات عمل می‌کند. مدل زبانی بزرگ استدلال سطح وظیفه را مدیریت می‌کند، در حالی که ربات رفتارهای اعتبارسنجی شده را اجرا می‌نماید. این جداسازی منعکس‌کننده بهترین شیوه‌ها در طراحی سیستم‌های رباتیک واقعی است.

در ادامه، ترجمه واژه‌به‌واژه و دقیق متن ارائه شده به زبان فارسی آورده شده است:

۱۳. Large Language Models for Manufacturing

مثال ۱: رابط‌های زبان طبیعی برای عملیات‌های کارگاهی (Shop-Floor) این مقاله سیستم‌های تولیدی را توصیف می‌کند که در آن‌ها اپراتورها با استفاده از زبان طبیعی با ماشین‌آلات و سیستم‌های تولید تعامل دارند. به جای پیمایش در رابط‌های پیچیده HMI یا PLC، کارگران دستورات صوتی یا نوشتاری صادر می‌کنند که توسط مدل‌های زبانی بزرگ (LLMs) تفسیر می‌شوند. این کار سد مهارتی را در سطح کارگاه کاهش می‌دهد و در عین حال خطوط لوله کنترل صنعتی موجود را حفظ می‌کند.

مثال ۲: برنامه‌ریزی فرآیند به کمک مدل زبانی بزرگ در تولید مدل‌های زبانی بزرگ برای کمک به مهندسان در تولید برنامه‌های فرآیند، مانند توالی‌بندی مراحل ساخت یا انتخاب ابزارهای مناسب، استفاده می‌شوند. این مدل‌ها بر روی داده‌های تولیدی تاریخی، مستندات و محدودیت‌ها استدلال می‌کنند. این سیستم‌ها در سناریوهای تولید واقعی که سرعت برنامه‌ریزی و تطبیق‌پذیری در آن‌ها حیاتی است، ارزیابی می‌شوند.

مثال ۳: تشخیص خودکار نقص‌های خط تولید این مقاله سیستم‌های هدایت‌شده با مدل زبانی بزرگ را ارائه می‌دهد که لاگ‌های حسگرها، هشدارها و سوابق تعمیر و نگهداری را برای تشخیص نقص‌ها در خطوط تولید تحلیل می‌کنند. مهندسان به جای بازرسی دستی لاگ‌ها، توضیحات زبان طبیعی درباره علت‌های احتمالی دریافت می‌کنند. این رویکرد در جریان‌های کاری واقعی تشخیص نقص صنعتی به نمایش گذاشته شده است.

مثال ۴: تعمیر و نگهداری پیش‌بینانه با استفاده از داده‌های متنی و حسگر مدل‌های زبانی بزرگ با داده‌های حسگر ساختاریافته و گزارش‌های تعمیر و نگهداری بدون ساختار یکپارچه شده‌اند تا خرابی تجهیزات را پیش‌بینی کنند. این مدل، تاریخچه تعمیر و نگهداری مبتنی بر زبان را با سیگنال‌های عملیاتی مرتبط می‌سازد. این استدلال ترکیبی، مداخله زودهنگام را امکان‌پذیر کرده و زمان توقف (downtime) را در محیط‌های تولیدی واقعی کاهش می‌دهد.

مثال ۵: استخراج دانش از مستندات تولیدی تأسیسات تولیدی اغلب بر کتابچه‌های راهنمای گسترده، دستورالعمل‌های استاندارد عملیاتی (SOPs) و مستندات قدیمی تکیه دارند. این مقاله نشان می‌دهد که چگونه مدل‌های زبانی بزرگ دانش کاربردی را از این اسناد استخراج کرده و آن‌ها را قابل پرس‌وجو می‌کنند. مهندسان می‌توانند سوالاتی نظیر روش‌های تعمیر و نگهداری یا محدودیت‌های ایمنی را بدون جستجو در مخازن بزرگ اسناد بپرسند.

مثال ۶: پشتیبانی تصمیم‌گیری با حضور انسان در حلقه برای مهندسان به جای جایگزینی مهندسان، مدل‌های زبانی بزرگ به عنوان عامل‌های پشتیبان تصمیم‌گیری مستقر می‌شوند. سیستم پیشنهادهایی برای زمان‌بندی، کنترل کیفیت یا رفع نقص ارائه می‌دهد، در حالی که انسان‌ها قدرت نهایی را حفظ می‌کنند. این کار با نحوه پذیرش واقع‌گرایانه هوش مصنوعی در محیط‌های صنعتی حساس از نظر ایمنی همسو است.

مثال ۷: بازرسی کیفیت و تحلیل عیوب این مقاله درباره تحلیل گزارش‌های بازرسی و توصیفات عیوب توسط مدل زبانی بزرگ بحث می‌کند. با تجمیع داده‌های کیفیت مبتنی بر متن، سیستم الگوهای تکرار شونده عیوب را شناسایی کرده و اقدامات اصلاحی پیشنهاد می‌دهد. این کار فرآیندهای بهبود مستمر را که هم‌اکنون در تولید استفاده می‌شوند، ارتقا می‌دهد.

مثال ۸: یکپارچه‌سازی با دوقلوهای دیجیتال (Digital Twins) و سیستم‌های تولید مدل‌های زبانی بزرگ با بازنمایی‌های دوقلوی دیجیتال از کارخانه‌ها ترکیب می‌شوند تا درباره رفتار سیستم و بهینه‌سازی‌های بالقوه استدلال کنند. مدل زبانی خروجی‌های شبیه‌سازی و داده‌های تولید را پرس‌وجو می‌کند تا بهبودهایی را پیشنهاد دهد. این یکپارچه‌سازی در جریان‌های کاری برنامه‌ریزی و بهینه‌سازی صنعتی آزمایش شده است.

مثال ۹: کمک به زنجیره تأمین و زمان‌بندی تولید این مقاله بر استفاده از مدل زبانی بزرگ در تحلیل برنامه‌های تولید، محدودیت‌های تأمین و موازنه‌های عملیاتی تأکید می‌کند. مدل‌های زبانی بزرگ به برنامه‌ریزان کمک می‌کنند تا تداخل‌های پیچیده زمان‌بندی را درک کرده و برنامه‌های جایگزین تولید کنند. این سیستم‌ها در سناریوهای برنامه‌ریزی تولید واقعی که پاسخگویی در آن‌ها ضروری است، ارزیابی می‌شوند.

مثال ۱۰: پیوند دادن سیستم‌های قدیمی و خطوط لوله هوش مصنوعی مدرن بسیاری از محیط‌های تولیدی بر سیستم‌های قدیمی (Legacy) تکیه دارند که برای یکپارچه‌سازی با هوش مصنوعی طراحی نشده‌اند. این مقاله نشان می‌دهد که چگونه مدل‌های زبانی بزرگ به عنوان یک لایه انتزاعی عمل می‌کنند و قصد انسانی را به پرس‌وجوها و اقدامات در میان سیستم‌های ناهمگون ترجمه می‌کنند. این کار پذیرش مدل زبانی بزرگ را بدون نیاز به طراحی مجدد کامل سیستم امکان‌پذیر می‌سازد.

در ادامه، ترجمه واژه‌به‌واژه و دقیق متن ارائه شده به زبان فارسی آورده شده است:

۱۴ LLM-BRAIn AI-driven Fast Generation of Robot .

مثال ۱: تولید سریع رفتارهای اجرایی ربات این مقاله LLM-BRAIn را معرفی می‌کند؛ سیستمی که در آن یک مدل زبانی بزرگ (LLM)، توصیفات سطح‌بالای وظایف را مستقیماً به رفتارهای اجرایی ربات تبدیل می‌کند. به جای طراحی دستی درخت‌های رفتار یا اسکریپت‌های کنترلی، سیستم به‌طور خودکار آن‌ها را تولید می‌کند. این امر زمان توسعه را برای وظایف رباتیک واقعی به‌طور قابل‌توجهی کاهش می‌دهد و در عین حال سازگاری با چارچوب‌های کنترلی موجود را حفظ می‌کند.

مثال ۲: سنتز درخت رفتار از زبان طبیعی یک مشارکت کلیدی، تولید خودکار درخت‌های رفتار از دستورالعمل‌های زبان طبیعی است. مدل زبانی بزرگ، توصیفات وظایف را به منطق کنترلی ساختاریافته‌ای ترجمه می‌کند که ربات‌ها می‌توانند به‌طور قابل‌اطمینان اجرا کنند. این رویکرد در سناریوهای واقعی وظایف رباتیک ارزیابی شده است که کاربرپذیری عملی را فراتر از شبیه‌سازی نشان می‌دهد.

مثال ۳: انطباق سریع با وظایف جدید بدون برنامه‌نویسی مجدد این سیستم به ربات‌ها اجازه می‌دهد تا صرفاً با تغییر توصیف متنی وظیفه، با وظایف جدید سازگار شوند. مدل زبانی بزرگ بدون مداخله دستی، ساختار رفتاری مربوطه را دوباره تولید می‌کند. این موضوع از موارد استفاده در دنیای واقعی که ربات‌ها باید مکرراً وظایف خود را تغییر دهند (مانند محیط‌های خدماتی یا صنعتی) پشتیبانی می‌کند.

مثال ۴: یکپارچه‌سازی با خطوط لوله کنترل رباتیک کلاسیک LLM-BRAIn جایگزین کنترل‌کننده‌های سطح‌پایین نمی‌شود. در عوض، رفتارهای تولید شده از طریق مازول‌های کنترلی رباتیک موجود اجرا می‌شوند. این طراحی منعکس‌کننده محدودیت‌های استقرار واقعی است و در عین حال که از انعطاف‌پذیری هدایت‌شده با مدل زبانی بزرگ بهره می‌برد، قابلیت اطمینان را نیز تضمین می‌کند.

مثال ۵: کاهش تلاش مهندسی در برنامه‌نویسی ربات مقاله نشان می‌دهد که رفتارهای تولید شده توسط مدل زبانی بزرگ، نیاز به منطق کنترلی ساخته شده با دست را کاهش می‌دهند. مهندسان می‌توانند به جای کدنویسی رفتارها، بر تعریف اهداف تمرکز کنند. این امر مستقیماً به یک گلوگاه اصلی در توسعه سیستم‌های رباتیک واقعی می‌پردازد.

مثال ۶: خروجی ساختاریافته برای جلوگیری از اجرای مبهم مدل زبانی بزرگ رفتارها را در قالبی ساختاریافته و محدود شده خروجی می‌دهد و تضمین می‌کند که رفتارهای تولید شده، قابل تفسیر و قابل اجرا باقی بمانند. این کار از خطرات تولید زبان با قالب آزاد جلوگیری کرده و با الزامات ایمنی در ربات‌های فیزیکی همسو می‌شود.

مثال ۷: مقیاس‌پذیری در پلتفرم‌های رباتیک مختلف این رویکرد به‌گونه‌ای طراحی شده است که مستقل از پلتفرم باشد. رفتارهای تولید شده را می‌توان در کالبدها (embodiments) و فیزیک‌های مختلف رباتیک مجدداً استفاده کرد، مشروط بر اینکه مهارت‌های زیربنایی وجود داشته باشند. این امر از استقرار مقیاس‌پذیر در ناوگان‌های رباتیک ناهمگون پشتیبانی می‌کند.

مثال ۸: رفتارهای قابل خواندن توسط انسان و قابل اشکال‌زدایی درخت‌های رفتار تولید شده توسط انسان قابل خواندن هستند و به مهندسان اجازه می‌دهند در صورت نیاز آن‌ها را بررسی، اشکال‌زدایی و اصلاح کنند. این شفافیت برای پذیرش صنعتی، که در آن توضیح‌پذیری و تأیید مورد نیاز است، ضروری است.

مثال ۹: پشتیبانی از اجرای وظایف با افق زمانی طولانی سیستم می‌تواند رفتارهایی تولید کند که شامل توالی‌بندی، شرط‌ها و اقدامات بازیابی باشد. این امر ربات‌ها را قادر می‌سازد تا به جای اقدامات مجزا، وظایف چندمرحله‌ای را انجام دهند. چنین اجرای طولانی‌مدتی برای جریان‌های کاری رباتیک در دنیای واقعی حیاتی است.

مثال ۱۰: پیوند دادن استدلال سطح وظیفه و اجرا LLM-BRAIN به عنوان پلی بین توصیفات انتزاعی وظیفه و اجرای عینی ربات عمل می‌کند. مدل زبانی بزرگ استدلال سطح وظیفه را مدیریت می‌کند، در حالی که ربات رفتارهای اعتبارسنجی شده را اجرا می‌نماید. این جداسازی منعکس‌کننده بهترین شیوه‌ها در طراحی سیستم‌های رباتیک واقعی است.

در ادامه، ترجمه واژه‌به‌واژه و دقیق بخش‌های باقی‌مانده از متن ارائه شده به زبان فارسی آورده شده است:

۱۵ LLM-Driven Robots Risk Enacting Discrimination, Violence .

(ربات‌های هدایت‌شونده با مدل زبانی بزرگ در معرض خطر اعمال تبعیض و خشونت)

مثال ۱: رد تبعیض‌آمیز وظیفه در تعامل انسان و ربات این مقاله مواردی را گزارش می‌کند که در آن ربات‌های هدایت‌شونده با LLM، بر اساس نشانه‌های هویتی درک‌شده، به کاربران پاسخ‌های متفاوتی می‌دهند. ربات‌ها در هنگام تعامل با گروه‌های جمعیتی خاص، به احتمال بیشتری درخواست‌ها را رد می‌کردند یا لحن خود را تغییر می‌دادند. این رفتار به جای برنامه‌نویسی صریح، از سوگیری‌های مدل زبانی ناشی می‌شود که نشان‌دهنده ریسک‌های تبعیض واقعی در سیستم‌های مستقر شده است.

مثال ۲: عادی‌سازی زبان مضر از طریق عامل‌های دارای کالبد (Embodied) هنگامی که مدل‌های زبانی بزرگ زبان مشکل‌داری تولید می‌کنند، تأثیر آن پس از تجسم در یک ربات افزایش می‌یابد. این مقاله نشان می‌دهد که پاسخ‌های مضر یا تهاجمی زمانی که توسط یک ربات فیزیکی بیان یا اجرا می‌شوند، از نظر اجتماعی برجسته‌تر می‌شوند. این امر در مقایسه با سیستم‌های صرفاً متنی، آسیب را تشدید می‌کند.

مثال ۳: تشدید دستورالعمل‌های تهاجمی یا خشونت‌آمیز این مقاله سناریوهایی را تحلیل می‌کند که در آن دستورالعمل‌های مبهم یا تحریک‌آمیز انسانی، ربات‌های هدایت‌شونده با LLM را به سمت تولید تفسیرهای نالیمن سوق می‌دهد. حتی بدون فرمان‌های مستقیم برای ایجاد آسیب، ربات ممکن است اقدامات تهاجمی را پیشنهاد دهد یا برای انجام آن‌ها تلاش کند. این موضوع فاش می‌کند که چگونه کالبد فیزیکی، مخاطرات سوءتعبیر را بالا می‌برد.

مثال ۴: تقویت کلیشه‌ها از طریق تخصیص وظایف مشاهده شد که ربات‌های هدایت‌شونده با LLM وظایف را بر اساس فرضیات کلیشه‌ای نهفته در داده‌های آموزشی، به‌طور متفاوتی اختصاص می‌دهند. برای مثال، فرضیات درباره توانایی فیزیکی یا اقتدار بر پاسخ‌های ربات تأثیر گذاشت. این رفتارها منعکس‌کننده سوگیری‌های اجتماعی واقعی کدگذاری شده در مدل‌های زبانی هستند.

مثال ۵: فقدان استدلال اخلاقی بافتاری در محیط‌های فیزیکی این مقاله تأکید می‌کند که مدل‌های زبانی بزرگ اغلب هنگام کنترل ربات‌ها، فاقد پایه‌ریزی اخلاقی موقعیتی (Situational Moral Grounding) هستند. بدون محدودیت‌های صریح، ربات‌ها ممکن است اقداماتی را پیشنهاد دهند که از نظر اجتماعی نامناسب یا نالیمن هستند. این محدودیت در محیط‌های انسانی مشترک در دنیای واقعی حیاتی می‌شود.

مثال ۶: اتکای بیش از حد اپراتورهای انسانی به خروجی‌های LLM کاربران انسانی تمایل دارند زمانی که پاسخ‌ها سلیس و مطمئن به نظر می‌رسند، بیشتر به سیستم‌های رباتیک اعتماد کنند. این مقاله نشان می‌دهد که اپراتورها اغلب از پیشنهادات نالیمن یا سوگیرانه تولید شده توسط LLM پیروی می‌کنند. این امر یک خطر ترکیبی ایجاد می‌کند که در آن قضاوت انسانی تحت‌الشعاع هوش درک‌شده قرار می‌گیرد.

مثال ۷: شکست فیلترهای ایمنی پس از تجسم (Embodiment) فیلترهای ایمنی در سطح متن که در سیستم‌های چت به خوبی کار می‌کنند، به‌طور کامل به بافتار رباتیک منتقل نمی‌شوند. این مقاله مواردی را مستند می‌کند که در آن زبان فیلتر شده همچنان منجر به اقدامات فیزیکی مضر می‌شود. این موضوع شکاف بین ایمنی زبانی و ایمنی کالبدی را آشکار می‌سازد.

مثال ۸: رفتار اخلاقی متناقض در سناریوهای مشابه یک ربات ممکن است در یک سناریو به‌طور اخلاقی و در بافتاری که اندکی تغییر یافته، به‌طور غیراخلاقی پاسخ دهد. این تناقض به جای استدلال اصولی، از حساسیت به پرامپت (Prompt Sensitivity) ناشی می‌شود. چنین غیرقابل‌پیش‌بینی بودن در محیط‌های استقرار واقعی غیرقابل قبول است.

مثال ۹: فقدان مکانیسم‌های مسئولیت‌پذیری هنگامی که رفتار مضر رخ می‌دهد، مسئولیت مشخص نیست؛ اینکه آیا بر عهده LLM است، یا یکپارچه‌ساز سیستم (System Integrator) و یا اپراتور. این مقاله تأکید می‌کند که سیستم‌های رباتیک فعلی فاقد ساختارهای مسئولیت‌پذیری روشن هستند. این امر استقرار و قانون‌گذاری در دنیای واقعی را پیچیده می‌کند.

مثال ۱۰: ریسک‌های مقیاس‌پذیر کردن ربات‌های هدایت‌شونده با LLM بدون حکمرانی این مقاله هشدار می‌دهد که مقیاس‌پذیر کردن ربات‌های هدایت‌شونده با LLM بدون چارچوب‌های حکمرانی قدرتمند، آسیب‌های اجتماعی را تشدید خواهد کرد. پس از استقرار در مقیاس بزرگ، رفتارهای سوگیرانه یا نالیمن به‌سرعت گسترش می‌یابند. این موضوع تصمیمات طراحی اولیه را برای پذیرش در صنعت بسیار حیاتی می‌کند.

۱۶ Multi-Agent Systems for Robotic Autonomy with LLMs

(سیستم‌های چندعاملی برای خودمختاری رباتیک با مدل‌های زبانی بزرگ)

مثال ۱: طراحی سیستم رباتیک سرتاسری (End-to-End) از توصیفات وظیفه به زبان طبیعی این مقاله سیستمی را به نمایش می‌گذارد که در آن توصیف متنی وظیفه تنها ورودی است و چارچوب به‌طور خودکار یک راه حل کامل رباتیک را طراحی می‌کند. سیستم تعیین می‌کند که به چند ربات نیاز است، کجا باید قرار گیرند و بازوهای آن‌ها چقدر باید بلند باشد. این منعکس‌کننده یک جریان کاری مهندسی در دنیای واقعی است که در آن تصمیمات طراحی اولیه اغلب دستی و زمان‌بر هستند.

مثال ۲: عامل تحلیل‌گر وظیفه برای تجزیه مسائل مهندسی محور عامل تحلیل‌گر وظیفه، پارامترهای مهندسی عینی - مانند مختصات هدف، مکان‌های پایه ربات و محدودیت‌های وظیفه - را از توصیفات طولانی و بدون ساختار استخراج می‌کند. این کار مشابه نحوه ترجمه نیازهای مشتری به مشخصات فنی توسط مهندسان انسانی است. «گزارش تحلیل وظیفه» تولید شده مستقیماً در مراحل بعدی طراحی و کنترل ربات استفاده می‌شود.

مثال ۳: انتخاب خودکار پیکربندی‌های بازوی ربات در سناریوهای صنعتی و پزشکی (مانند مرتب‌سازی در انبار یا جابجایی ابزار جراحی)، عامل طراح ربات طول بازوهای مناسب را از میان چندین گزینه انتخاب می‌کند. عامل با استدلال درباره قابلیت دسترسی، هزینه و ایمنی، از طراحی بیش از حد (Over-designed) یا ربات‌های کم‌توان جلوگیری می‌کند. این منعکس‌کننده موازنه‌های طراحی واقعی است که در پروژه‌های اتوماسیون صنعتی با آن‌ها مواجه می‌شویم.

مثال ۴: تخصیص وظایف چندرباتی در سناریوهای صنعتی برای وظایفی که شامل چندین هدف هستند، مانند مرتب‌سازی اقلام انبار یا پالت‌گذاری، سیستم به‌طور خودکار تصمیم می‌گیرد که چند ربات مورد نیاز است و زیروظایف را به هر ربات اختصاص می‌دهد. این قابلیت در ده سناریوی واقع‌گرایانه ارزیابی شده است که ارتباط عملی برای استقرار صنعتی چندرباتی را نشان می‌دهد.

مثال ۵: تولید خودکار کد کنترل یادگیری تقویتی (RL) عامل طراح یادگیری تقویتی، کدهای اجرایی RL شامل تعاریف محیط، اسکریپت‌های آموزش و اسکریپت‌های ارزیابی را تولید می‌کند. این اسکریپت‌ها برای آموزش کنترل‌کننده‌های ربات بدون نیاز به کدنویسی انسانی اجرا می‌شوند. این موضوع نشان می‌دهد که چگونه LLM ها می‌توانند مانع استقرار کنترل‌های مبتنی بر RL را در عمل کاهش دهند.

مثال ۶: انتخاب خودمختار الگوریتم‌های RL بر اساس نیازهای وظیفه سیستم به‌طور خودمختار الگوریتم‌های RL مناسب (مانند PPO یا SAC) را بسته به پیچیدگی وظیفه و نیازهای حرکتی انتخاب می‌کند. این کار مشابه تصمیمات مهندسی واقعی است که معمولاً توسط متخصصان رباتیک گرفته می‌شود و از طریق اجرای فرآیندهای آموزشی و خروجی‌های مسیر حرکت اعتبارسنجی می‌شود.

مثال ۷: سناریوهای واقع‌گرایانه وظایف صنعتی و پزشکی این چارچوب در ده سناریوی واقعی از جمله جای‌گذاری در خط مونتاژ، مرتب‌سازی در انبار، کمک به تغذیه سالمندان و جابجایی ابزار جراحی ارزیابی شده است. این سناریوها به طراحی‌های ربات و استراتژی‌های کنترل متفاوتی نیاز دارند که نشان‌دهنده عمومیت (Generality) فراتر از مثال‌های ساده است.

مثال ۸: بصری‌سازی مسیرهای حرکت اجرا شده و فرمان‌های موتور پس از آموزش، سیستم منحنی‌های یادگیری، نمودارهای کنترل موتور مفصل و مسیرهای حرکت مجری نهایی (End-effector) را خروجی می‌دهد. این خروجی‌ها مصنوعات استاندارد هستند که توسط مهندسان رباتیک برای تأیید رفتار سیستم استفاده می‌شوند، که نشان می‌دهد چارچوب به‌طور طبیعی با جریان‌های کاری مهندسی موجود ادغام می‌شود.

مثال ۹: مطالعات حذف (Ablation Studies) که وابستگی‌های مهندسی حیاتی را آشکار می‌کند با غیرفعال کردن هر یک از عامل‌ها (تحلیل‌گر وظیفه، طراح ربات یا طراح RL)، نویسندگان نشان می‌دهند که حذف هر یک از اجزا عملکرد سیستم را به‌طور قابل‌توجهی کاهش می‌دهد. این منعکس‌کننده مهندسی سیستم در دنیای واقعی است، جایی که فقدان مراحل تحلیل یا طراحی اغلب علی‌رغم الگوریتم‌های پیشرفته منجر به شکست می‌شود.

مثال ۱۰: استقرار مستقل از مدل (Model-Agnostic) در میان قابلیت‌های مختلف LLM این چارچوب با چندین LLM با قابلیت‌های متفاوت (GPT-4o-mini، GPT-4o و نسخه‌های DeepSeek) آزمایش شده است. نتایج نشان می‌دهد که مدل‌های استدلال قوی‌تر، طراحی‌های ربات و کدهای کنترل قابل‌اعتمادتری تولید می‌کنند. این کار بینش عملی در مورد چگونگی تأثیر انتخاب مدل بر نتایج سیستم‌های رباتیک واقعی ارائه می‌دهد.

۱۷) NRTrans: Compiler-Verified Language-to-Robot Programs

(ان آر ترنس: برنامه‌های زبان به ربات تأیید شده توسط کامپایلر)

مثال ۱: برنامه‌های زبان به ربات تأیید شده توسط کامپایلر پیش از اجرا سیستم NRTrans وظایف زبان طبیعی مانند «بطری روی میز را بردار» را به یک برنامه «زبان مهارت ربات (RSL)» ترجمه می‌کند که سپس پیش از اجرا، کامپایل و تأیید می‌شود. تنها برنامه‌هایی که از کامپایلر عبور می‌کنند به ربات ارسال می‌شوند. این تضمین می‌کند که هیچ کد کنترل نامعتبر از نظر سینتکسی یا معنایی هرگز روی سخت‌افزار اجرا نشود؛ یک نیاز حیاتی برای استقرارهای واقعی.

مثال ۲: اصلاح خطای مبتنی بر بازخورد به جای اجرای کورکورانه هنگامی که یک LLM برنامه RSL نادرستی تولید می‌کند (مثلاً سمیکالن‌های فراموش شده یا پارامترهای نامعتبر)، کامپایلر RSL خطا را تشخیص داده و بازخورد قابل خواندن برای انسان تولید می‌کند. این بازخورد به LLM بازگردانده می‌شود که برنامه اصلاح‌شده‌ای را دوباره تولید می‌کند. این حلقه به‌طور خودکار تا زمان تأیید برنامه اجرا می‌شود و از شکست‌های زمان اجرا (runtime) در ربات‌ها جلوگیری می‌کند.

مثال ۳: نرخ موفقیت بالا با استفاده از مدل‌های زبانی سبک روی ربات‌های واقعی سیستم NRTrans با استفاده از یک LLM سبک با ۲ میلیارد پارامتر، بدون هیچ‌گونه بازآموزی مدل، به نرخ موفقیت ۹۲٪ دست می‌یابد. این موضوع در مجموعه وظایف رباتیک واقعی شامل ناوبری، ادراک و برداشتن اشیاء به اثبات رسیده است. این نشان‌دهنده امکان‌پذیری برای ربات‌های دارای محدودیت منابع است، جایی که مدل‌های پایه بزرگ قابل استقرار نیست.

مثال ۴: اجرای وظایف چندمرحله‌ای روی یک ربات واقعی مبتنی بر ROS این چارچوب روی یک جابجاگر متحرک TIAGo که سیستم عامل ROS را اجرا می‌کند، ارزیابی شده است که فرمان‌های چندمرحله‌ای مانند «به میز نزدیک شو، دنبال یک لیوان بگرد و آن را بردار» را اجرا می‌کند NRTrans. توالی تأیید شده‌ای از مهارت‌های ربات را تولید می‌کند که به‌درستی روی سخت‌افزار اجرا می‌شوند. این نشان‌دهنده اجرای وظایف واقعی و طولانی‌مدت توسط زبان است.

مثال ۵: مدیریت فرمان‌های زبان طبیعی مبهم وظایفی مانند «کمی جلو برو» یا «چندین بار دور خود بچرخ» در مجموعه ارزیابی گنجانده شده‌اند NRTrans. ابهام را از طریق اصلاح مبتنی بر بازخورد حل کرده و به جای شکست خاموش، برنامه‌های رباتیک اجرایی تولید می‌کند. این منعکس‌کننده ورودی‌های کاربر در دنیای واقعی است که اغلب دقیق نیستند.

مثال ۶: ترجمه ایمن دستورالعمل‌های متوالی پیچیده سیستم NRTrans دستورات پیچیده‌ای مانند «۲ متر به جلو حرکت کن، موز را بردار، به چپ بچرخ، سپس ۳ متر به جلو برو» را با موفقیت ترجمه می‌کند. این دستورالعمل‌ها به گزاره‌های RSL تجزیه می‌شوند که تأیید شده و به‌صورت متوالی اجرا می‌گردند. این نشان‌دهنده مدیریت ایمن و صحیح وظایف ترکیبی است.

مثال ۷: جلوگیری از توهم کد در LLM از طریق انتزاع زبانی به جای تولید مستقیم برنامه‌های کنترل کامل پایتون، NRTrans مدل زبانی را مجبور می‌کند که فقط فرمان‌های سطح بالای RSL تولید کند. این انتزاع به‌طور قابل توجهی ساختارهای کد توهم‌آمیز مانند حلقه‌ها یا کلاس‌های نامعتبر را کاهش می‌دهد. نتیجه، تولید برنامه کنترل قابل اعتمادتر در عمل است.

مثال ۸: پیام‌های خطای معنایی-شهودی برای بازیابی سریع پیام‌های خطای کامپایلر به گونه‌ای بازطراحی شده‌اند که معنایی و شهودی باشند (مثلاً «کلمات کلیدی باید با حروف کوچک باشند» یا «سمیکالن مفقود شده است»). این پیام‌ها LLM ها را قادر می‌سازند تا خطاها را به‌طور میانگین در ۱ تا ۲ تکرار اصلاح کنند. این بازیابی سریع برای کاربردهای رباتیک بلادرنگ یا نزدیک به بلادرنگ ضروری است.

مثال ۹: اجرای وظیفه در حالت Zero-Shot با اصلاح خودکار حتی در تنظیمات Zero-shot بدون ارائه برنامه‌های نمونه (NRTrans نرخ موفقیت تولید برنامه را با استفاده از تنظیم مبتنی بر بازخورد، بیش از ۹۰٪ افزایش می‌دهد. این به ربات‌ها اجازه می‌دهد تا وظایف قبلاً دیده نشده را بدون مهندسی دستی پرامپت یا بازآموزی انجام دهند که برای استقرار مقیاس‌پذیر حیاتی است.

مثال ۱۰: چارچوب قابل انتقال در میان ربات‌ها و سناریوها سیستم NRTrans قابلیت‌های ربات را به کلمات کلیدی «مهارت» انتزاع می‌کند که اجازه می‌دهد همان چارچوب با به‌روزرسانی کتابخانه مهارت و پیوندهای کامپایلر، برای ربات‌های مختلف تطبیق داده شود. این طراحی به جای دموهای تک‌رباتی، از مقیاس‌پذیری در میان پلتفرم‌ها پشتیبانی می‌کند.

۸ Plug in the Safety Chip: Enforcing Constraints for LLM-driven Robots

(نصب تراشه ایمنی: اعمال محدودیت‌ها برای ربات‌های هدایت‌شونده با مدل زبانی)

مثال ۱: اجرای محدود شده با ایمنی روی یک ربات جابجاگر متحرک واقعی سیستم Safety Chip روی یک پلتفرم ربات متحرک واقعی که وظایف جابجایی متحرک به سبک خانگی را انجام می‌دهد، مستقر شده است. ربات دستورالعمل‌های زبان طبیعی را در حالی اجرا می‌کند که به‌طور مداوم محدودیت‌های ایمنی را پایش می‌نماید. اقدامات نالایمن در زمان اجرا مسدود می‌شوند و تضمین می‌کنند که ربات هرگز در طول اجرای فیزیکی وارد حالت‌های ممنوعه نشود.

مثال ۲: اعمال «نباید»های مشخص شده توسط انسان در قالب زبان طبیعی کاربران ایمنی مانند «هرگز گوشی را برندار» یا «قبل از قفسه کتاب به سمت میز پاتختی نرو» را به زبان انگلیسی ساده مشخص می‌کنند. این قوانین به محدودیت‌های منطق زمانی خطی (LTL) قابل تأیید ترجمه شده و در حین اجرا اعمال می‌شوند. ربات در عین تکمیل وظیفه، به‌طور جدی به این محدودیت‌ها پایبند است.

مثال ۳: هرس کردن (Pruning) اقدامات در زمان اجرا برای برنامه‌های تولید شده توسط LLM هنگامی که یک عامل LLM اقدامی نالایمن پیشنهاد می‌دهد، Safety Chip آن اقدام را پیش از اجرا هرس می‌کند. ربات هرگز گام نالایمن را به‌طور فیزیکی اجرا نمی‌کند. این مکانیسم صرف‌نظر از نحوه تولید برنامه توسط LLM عمل می‌کند و آن را با عامل‌های رباتیک مبتنی بر LLM موجود سازگار می‌سازد.

مثال ۴: برنامه‌ریزی مجدد حلقه-بسته با توضیحات ایمنی سیستم به جای مسدود کردن بی‌صدای اقدامات، توضیحات زبان طبیعی تولید می‌کند که توضیح می‌دهد چرا یک عمل محدودیت‌های ایمنی را نقض می‌کند. این توضیحات برای هدایت برنامه‌ریزی مجدد به LLM بازگردانده می‌شوند. این حلقه هم در محیط‌های شبیه‌سازی شده و هم در آزمایش‌های ربات واقعی به نمایش گذاشته شده است.

مثال ۵: ارزیابی ایمنی در مقیاس بزرگ در VirtualHome سیستم در محیط VirtualHome روی ۲۰ وظیفه خانگی، که هر کدام تا پنج محدودیت ایمنی همزمان داشتند، ارزیابی شده است Safety Chip. به نرخ‌های ایمنی نزدیک به کامل دست می‌یابد در حالی که موفقیت وظیفه را در سطح بالا حفظ می‌کند. این نشان‌دهنده مقیاس‌پذیری در برابر مشخصات وظایف پیچیده و مشابه دنیای واقعی است.

مثال ۶: مدیریت پیچیدگی رو به افزایش محدودیت‌ها بدون شکست با افزایش تعداد محدودیت‌های ایمنی، عامل‌های LLM پایه به سرعت در حفظ ایمنی شکست می‌خورند. در مقابل، Safety Chip حتی با پنج یا تعداد بیشتری محدودیت، نرخ ایمنی بالا را حفظ می‌کند که نشان‌دهنده استحکام در شرایط استقرار واقعی است که در آن باید چندین قانون به‌طور همزمان رعایت شوند.

مثال ۷: استقرار روی ربات Spot شرکت بوستون داینامیکس سیستم روی ربات Spot شرکت Boston Dynamics مستقر شده است که وظایف جابجایی متحرک را در یک محیط داخلی واقعی با ۲۳ شیء و نقطه نشانه انجام می‌دهد. ربات وظایف را در حالی انجام می‌دهد که تا ده محدودیت ایمنی را رعایت می‌کند، که نشان‌دهنده سازگاری سخت‌افزاری واقعی و امکان‌پذیری عملی است.

مثال ۸: حفظ ایمنی ۱۰۰٪ روی سخت‌افزار واقعی روی ربات Spot، سیستم Safety Chip بدون کاهش موفقیت در تکمیل وظیفه، به نرخ ایمنی ۱۰۰٪ دست می‌یابد. در مقابل، رویکردهای پایه در همان شرایط محدودیت‌های ایمنی را نقض می‌کنند. این نتیجه بر اهمیت اعمال رسمی (Formal) ایمنی برای استقرارهای واقعی تأکید می‌کند.

مثال ۹: تشخیص و لغو دستورالعمل‌های متناقض هنگامی که کاربران دستورالعمل‌هایی ارائه می‌دهند که ذاتاً با محدودیت‌های ایمنی در تضاد هستند، Safety Chip تضاد را تشخیص داده و اجرا را به‌طور ایمن لغو می‌کند. سیستم‌های پایه در هر صورت برای اجرای وظیفه تلاش می‌کنند که منجر به رفتار نایمن می‌شود. این قابلیت برای تعامل انسان و ربات در دنیای واقعی ضروری است.

مثال ۱۰: «تراشه ایمنی» ماژولار و سازگار با هر عامل LLM تراشه ایمنی به عنوان یک ماژول «وصل و اجرا (Plug-and-play)» طراحی شده است که می‌تواند به هر عامل رباتیک مبتنی بر LLM موجود متصل شود. این سیستم به معماری برنامه‌ریزی یا رژیم آموزشی خاصی وابسته نیست. این ماژولار بودن منعکس‌کننده نحوه استقرار مکانیسم‌های ایمنی در سیستم‌های صنعتی واقعی است.

۱۹. Reshaping Robot Trajectories Using Natural Language Commands

(تغییر شکل مسیرهای حرکت ربات با استفاده از فرمان‌های زبان طبیعی)

مثال ۱: اصلاح مسیرهای بازوی ربات با زبان طبیعی این مقاله سیستمی را ارائه می‌دهد که در آن کاربران مسیر حرکت موجود بازوی ربات را با استفاده از فرمان‌های صوتی مانند «از لیوان نوشیدنی دور بمان» تغییر شکل می‌دهند. به جای برنامه‌نویسی مجدد یا برنامه‌ریزی دوباره از ابتدا، ربات مسیر حرکت خود را در حالی اصلاح می‌کند که نقطه شروع و هدف را ثابت نگه می‌دارد. این رفتار روی یک بازوی ربات واقعی اجرا شده که نشان‌دهنده تأثیر مستقیم زبان بر حرکت است.

مثال ۲: سناریوی ایمنی کارخانه با اشیاء شکننده یک سناریوی واقع‌گرایانه شبیه به کارخانه به نمایش گذاشته شده است که در آن یک ربات یک بطری را به سمت یک شیکر حمل می‌کند در حالی که از نزدیکی لیوان‌های کریستالی شکننده عبور می‌کند. مسیر حرکت اصلی بدون برخورد اما نایمن است. با استفاده از زبان طبیعی، کاربران به ربات دستور می‌دهند که از لیوان‌ها دورتر شود و مسیر تغییر شکل یافته به‌طور ایمن روی سخت‌افزار واقعی اجرا می‌شود.

مثال ۳: تغییر شکل مسیر حرکت در زمان واقعی روی ربات فرانکا پاندا سیستم روی یک بازوی ۷ درجه آزادی Franka Emika Panda که در یک فضای کاری میزکار فیزیکی فعالیت می‌کند، ارزیابی شده است. مسیرهای تغییر شکل یافته تولید شده از فرمان‌های زبانی مستقیماً توسط ربات اجرا می‌شوند در حالی که محدودیت‌های سینماتیکی و حرکت نرم حفظ می‌گردد. این موضوع امکان‌پذیری در دنیای واقعی را فراتر از شبیه‌سازی تأیید می‌کند.

مثال ۴: جایگزینی آموزش فیزیکی (Kinesthetic) با کنترل مبتنی بر زبان این مقاله تغییر شکل مسیر با زبان طبیعی را با آموزش فیزیکی، ترسیم مسیر و تنظیم دستی تابع هزینه (Cost-function) مقایسه می‌کند. در آزمایش‌های ربات واقعی، کاربران با استفاده از زبان، سریع‌تر از هدایت فیزیکی ربات، به مسیرهای صحیح و ایمن دست می‌یابند. این موضوع مزایای کاربردی را برای رباتیک صنعتی و خدماتی برجسته می‌کند.

مثال ۵: مدیریت قدرتمند واژگان طبیعی و دیده نشده کاربران دستوراتی با کلمات دیده نشده و مترادف صادر می‌کنند، مانند «کمی دورتر بمان»، «کمی نزدیک‌تر عبور کن» یا «از شیشه فاصله بگیر». علی‌رغم ندیدن این عبارات در طول آموزش، ربات به‌درستی مسیر خود را تغییر شکل می‌دهد. این استحکام از طریق مدل‌های زبانی پیش‌آموزش‌دیده به دست آمده و در حرکت‌های اجرا شده تأیید شده است.

مثال ۶: کنترل دقیق با استفاده از اصلاح‌کننده‌های شدت زبانی سیستم اصلاح‌کننده‌های شدت مانند «کمی»، «خیلی» یا «بسیار» را تفسیر می‌کند تا میزان تغییر شکل مسیر را تنظیم کند. برای مثال، «بسیار دور از شیء بمان» منجر به انحراف قابل توجه‌تری نسبت به «کمی دور بمان» می‌شود. این اثرات درجه‌بندی شده به‌وضوح در حرکت ربات واقعی قابل مشاهده هستند.

مثال ۷: حفظ محدودیت‌های شروع و هدف در حین اصلاح در تمام آزمایش‌ها، ربات موقعیت‌های شروع و پایان اصلی را حفظ می‌کند و تنها مسیر میانی را اصلاح می‌نماید. این تضمین می‌کند که تکمیل وظیفه حفظ شود در حالی که ایمنی یا معنای حرکت بهبود می‌یابد. چنین رفتار آگاه از محدودیت برای استقرار در محیط‌های تولیدی ضروری است.

مثال ۸: مطالعه کاربری که نشان‌دهنده ترجیح رابط‌های زبانی است یک مطالعه کاربری کنترل‌شده با آزمایش‌های ربات واقعی نشان می‌دهد که شرکت‌کنندگان به‌طور جدی زبان طبیعی را به برنامه‌نویسی، ترسیم یا آموزش فیزیکی ترجیح می‌دهند. کنترل مبتنی بر زبان به نرخ موفقیت ۱۰۰٪، زمان تکمیل سریع‌تر و شکست‌های کمتر ناشی از عدم امکان سینماتیکی دست می‌یابد. این یک اعتبارسنجی انسان‌محور از کاربردپذیری واقعی ارائه می‌دهد.

مثال ۹: زبان به عنوان یک لایه ایمنی سطح بالا روی برنامه‌ریزهای کلاسیک سیستم بر روی برنامه‌ریزهای حرکت کلاسیک مانند A* و CHOMP عمل می‌کند. زبان جایگزین برنامه‌ریزی نمی‌شود؛ بلکه مسیرهای برنامه‌ریزی‌شده را برای برآورده کردن محدودیت‌های معنایی یا ایمنی تغییر شکل می‌دهد. این طراحی لایه‌ای منعکس‌کننده نحوه استقرار سیستم‌های رباتیک در کاربردهای واقعی است.

مثال ۱۰: همکاری عملی انسان و ربات بدون برنامه‌نویسی مجدد این مقاله نشان می‌دهد که کاربران می‌توانند از طریق فرمان‌های گفتگو محور و بدون متوقف کردن اجرا یا تغییر کد، به‌طور تکرارپذیر حرکت ربات را اصلاح کنند. این امر کارگران غیرمتخصص را قادر می‌سازد تا به‌طور ایمن رفتار ربات را در فضاهای کاری مشترک تنظیم کنند که نشان‌دهنده مسیری واقع‌گرایانه به سمت رباتیک همکارانه (Collaborative Robotics) است.

۲۰. Understanding Natural Language Commands

(درک فرمان‌های زبان طبیعی)

مثال ۱: لیفتراک رباتیک با کنترل صوتی در محیط انبار این مقاله یک لیفتراک رباتیک را به نمایش می‌گذارد که فرمان‌های زبان طبیعی گفته شده را در محیطی شبیه به انبار اجرا می‌کند. دستوراتی مانند «پالت لاستیک را روی کامیون بگذار» به اقدامات ناوبری و جابجایی ترجمه می‌شوند. لیفتراک به‌طور فیزیکی حرکت می‌کند، پالت‌ها را برمی‌دارد و در مکان‌های هدف قرار می‌دهد که نشان‌دهنده یک سیستم جابجایی متحرک واقعی است که توسط زبان هدایت می‌شود.

مثال ۲: اجرای فرمان‌های چندمرحله‌ای با ناولری و جابجایی سیستم دستوراتی را که ناولری و جابجایی را ترکیب می‌کنند، مدیریت می‌کند؛ مانند «برو سراغ اولین جعبه سمت چپ و آن را بردار». ربات ابتدا به مکان صحیح ناولری می‌کند و سپس اقدام جابجایی را انجام می‌دهد. این نشان‌دهنده اجرای وظایف ترکیبی و چندمرحله‌ای در دنیای واقعی به جای اقدامات مجزا است.

مثال ۳: انطباق ارجاعات اشیاء در یک محیط فیزیکی (Grounding) هنگامی که کاربران می‌گویند «پالت لاستیک»، «کامیون» یا «پالت روی تریلر»، سیستم این عبارات را با استفاده از یک نقشه معنایی از محیط، به اشیاء فیزیکی واقعی منطبق می‌کند. ربات به‌درستی بین چندین شیء مشابه تفاوت قائل می‌شود که یک نیاز حیاتی برای استقرار صنعتی واقعی است.

مثال ۴: تفسیر صحیح روابط فضایی سیستم روابط فضایی مانند «روی»، «کنار»، «سمت چپ» و «جلوی» را با موفقیت تفسیر می‌کند. برای مثال، بین «پالت را روی کامیون بگذار» و «برو سمت پالت روی کامیون» تمایز قائل شده و برنامه‌های عملیاتی متفاوتی تولید می‌کند. این قابلیت در رفتارهای اجرا شده‌ی لیفتراک نشان داده شده است، نه فقط در دقت تجزیه کلمات.

مثال ۵: یادگیری معانی افعال از دستورات تولید شده توسط انسان به جای کدنویسی دستی اقدامات، مدل معانی افعالی مانند «گذاشتن»، «برداشتن» و «بلند کردن» را از مجموعه‌ای از دستورات نوشته شده توسط انسان یاد می‌گیرد. این معانی یادگرفته شده سپس برای کنترل اقدامات ربات استفاده می‌شوند. این نشان‌دهنده رویکردی مقیاس‌پذیر است که در آن ربات‌ها با زبان انسان تطبیق می‌یابند، نه اینکه انسان‌ها را مجبور به تطبیق با ربات‌ها کنند.

مثال ۶: مدیریت زبان انسانی نویزی (Noisy) و آموزش‌نندیده دستورات از کاربران آموزش‌نندیده از طریق جمع‌سپاری جمع‌آوری شده‌اند و شامل غلط‌های املایی، ابهام و عبارت‌بندی‌های غیرمعارف هستند. با وجود این، ربات بسیاری از این دستورات را با موفقیت اجرا می‌کند. این نشان‌دهنده استحکام در برابر ورودی‌های انسانی دنیای واقعی است که برای استقرار خارج از محیط‌های آزمایشگاهی کنترل‌شده ضروری است.

مثال ۷: ارزیابی سرتاسری رفتار اجرا شده‌ی ربات سیستم به‌طور سرتاسری و با نمایش ویدئوهای اقدامات اجرا شده‌ی ربات به ارزیاب‌های انسانی ارزیابی شده است؛ ارزیاب‌هایی که قضاوت کردند آیا ربات دستور را به‌درستی دنبال کرده است یا خیر. نمرات توافق بالا تأیید می‌کند که رفتار فیزیکی ربات با قصد انسان مطابقت دارد. این ارزیابی بر موفقیت وظیفه تمرکز دارد، نه فقط دقت زبانی.

مثال ۸: موفقیت نسبی و شکست آبرومندانه در وظایف واقعی حتی زمانی که سیستم اشتباه می‌کند - مثلاً برداشتن پالت اشتباه - اغلب یک اقدام تا حدی درست انجام می‌دهد (مثلاً حرکت صحیح اما به سمت شیء اشتباه). این رفتار ویژگی سیستم‌های مستقر شده‌ی واقعی است و نشان‌دهنده انطباق (grounding) معنادار به جای شکست تصادفی است.

مثال ۹: پشتیبانی ضمنی از شفاف‌سازی و گفتگو از آنجایی که سیستم برای هر جزء منطبق شده‌ی یک دستور، نمرات اطمینان (Confidence scores) تولید می‌کند، می‌تواند تفسیرهای نامطمئن را شناسایی کند. نویسندگان به‌طور صریح درباره استفاده از این اطلاعات برای پرسیدن سوالات شفاف‌ساز قبل از اقدام بحث می‌کنند. این کار نویدبخش گفتگوی تعاملی انسان و ربات در استقرارهای واقعی است.

مثال ۱۰: ربات‌های قابل کنترل با صدا که در کنار انسان‌ها فعالیت می‌کنند لیفتراک رباتیک به‌گونه‌ای طراحی شده است که در محیط‌های با حداقل آماده‌سازی و در کنار انسان‌ها فعالیت کند، نه در سلول‌های صنعتی محصور شده. این موضوع سیستم را به عنوان نمونه‌ای اولیه از رباتیک همکارانه و هدایت‌شونده با زبان در محیط‌های دنیای واقعی معرفی می‌کند.