

بِسْمِ اللّٰهِ الرَّحْمٰنِ الرَّحِيْمِ



دانشگاه شهید بهشتی کرمان

دانشکده فنی و مهندسی
بخش مهندسی کامپیوتر

پایان نامه تحصیلی برای دریافت درجه کارشناسی ارشد

رشته مهندسی کامپیوتر گرایش هوش مصنوعی

خودکارسازی مهندسی اعلان : تولید اعلان‌های دستوری برای مدل‌های
بزرگ زبانی جهت حل مسائل پردازش زبان طبیعی

مؤلف:

ناهید عبداللهی کرمانی

استاد راهنمای:

دکتر مهدی افتخاری

استاد مشاوره:

دکتر سحر وحدتی

۱۴۰۴ فروردین

بـنام خدا

مـموراـخـلـقـ پـوـشـ

با احـسـانـ اـنـدـهـ اـیـ جـانـ وـ اـعـتـدـ اـلـ اـخـلـقـ اـیـ بـنـامـ عـالـ مـصـرـفـ دـاـستـ وـ اوـ هـوـلـهـ نـافـرـ اـعـالـ مـاـسـتـ بـهـ مـطـورـ فـاجـمـ شـایـرـتـیـ پـوـشـ بـهـ اـسـلـ،ـ تـوـیدـ اـنـشـ جـعـیدـ وـ بـسـارـیـ نـزـکـلـیـ بـشـرـمـاـ اـنـجـیـانـ وـ اـعـتـادـ بـیـاتـ عـلـیـ اـلـخـاـهـ دـوـشـ بـلـخـادـ بـهـ کـشـرـ:

- تمام تلاش خود را برای کشف حقیقت و فهم حقیقت بکار خواهیم برد و از مرگ بود جمل و خیریت دفاعیت می‌نماییم پربرینزی کنیم.
- حقوق پژوهشگران، پژوهشگران (دانش، حیوان، نبات و آشام)، سازمان بهداشت صاحبان حق را بر ریاست می‌نشانیم و «خط آن» می‌کوشیم.
- بـاـلـیـتـ اـدـیـ وـ مـمـونـیـ آـثـرـشـوـشـ اـیـ اـنـجـ مـیـ نـسـیـمـ،ـ بـرـایـ اـخـبـرـ پـوـشـ اـسـلـ اـنـجـامـ وـ زـنـدـهـ وـ اـزـ سـرـقـتـ عـلـیـ وـ اـرـجـاعـ نـهـاـنـ اـبـتـابـ اـبـتـابـ کـنـیـمـ.
- خـسـنـ پـاـيـندـیـ بـهـ اـنـصـافـ وـ اـبـتـابـ اـزـ مـرـگـ بـهـ تـبـیـنـ تـصـبـ وـ کـیـنـهـ بـیـانـتـیـ پـوـشـ اـسـلـ،ـ رـسـانـیـ تـمـادـ اـخـدـ خـواـیـمـ کـرـدـ.
- خـسـنـ اـنـاسـ دـارـیـ،ـ اـزـ سـایـ وـ اـنـخـانـاتـ اـقـسـادـیـ،ـ اـنـسـانـ وـ فـنـ مـوـجـودـ،ـ اـسـخـادـ وـ بـهـ دـوـرـ خـواـیـمـ کـرـدـ.
- اـزـ اـنـمـاـرـ غـیرـ اـخـلـقـیـ نـتـیـجـ پـوـشـ،ـ نـهـیـ اـنـمـاـرـ مـوـازـیـ،ـ بـهـ سـانـ وـ چـنـکـدـ اـکـارـیـ پـرـبـینـیـ کـنـیـمـ.
- اـسـلـ مـرـبـدـ بـوـدـ وـ رـازـ دـارـیـ رـاهـوـ تـامـ خـواـیـتـیـ پـوـشـیـ خـوـقـرـمـ دـیـمـ.
- درـهـ بـیـانـتـیـ پـوـشـیـ بـلـانـجـ مـلـ تـبـجـدـ کـرـدـ وـ بـرـایـ تـحـقـیـقـ آـنـ مـیـ کـوشـیـمـ.
- خـوـیـشـ رـاـلـزـمـ بـهـ رـیـاضـتـ کـیـ بـیـخـدـهـیـ عـلـیـ رـشـخـوـدـ قـوـنـیـ وـ مـرـتـرـاتـ بـیـاسـتـهـیـ حـرـفـهـیـ،ـ سـانـانـ،ـ دـوـتـیـ وـ رـاهـبـرـدـهـیـ فـیـ دـرـهـ مـرـاـلـ پـوـشـ مـیـ دـانـیـمـ.
- بـیـعـتـ اـصـلـ اـخـلـقـ دـوـشـ بـرـاـهـمـیـ فـرـنـکـیـ وـ اـنـسـمـ وـ بـهـ مـطـبـرـانـگـیـ اـنـ فـرـنـکـ بـهـ تـوـجـ وـ اـثـامـ آـنـ دـجـاسـ اـنـجـ مـیـ وـزـیـمـ.



دانشگاه شیدا بهرمان

تعهدنامه

اینجانب ناهید عبداللهی کرمانی به شماره دانشجویی ۱۱۵۶۰۰۵ دانشجوی مقطع کارشناسی ارشد رشته مهندسی کامپیوتر- هوش مصنوعی دانشکده فنی مهندسی دانشگاه شهید باهنر کرمان نویسنده پایان نامه با عنوان «خودکارسازی مهندسی پرامپت : تولید متا- پرامپت (پرامپت‌های دستوری) برای مدل‌های بزرگ زبانی جهت حل مسائل پردازش زبان طبیعی» تحت راهنمایی دکتر مهدی افتخاری تأیید می‌کنم که این پایان نامه نتیجه پژوهش اینجانب می‌باشد و در عین حال که موضوع آن تکراری نیست، در صورت استفاده از منابع دیگران، نشانی دقیق و مشخصات کامل آن درج شده است. همچنین موارد زیر را نیز تعهد می‌کنم:

- ۱ - برای انتشار تمام یا قسمتی از داده‌ها یا دستاوردهای خود در مجامع و رسانه‌های علمی اعم از همایش‌ها و مجلات داخلی و خارجی به صورت مقاله، کتاب، ثبت اختراع و به صورت مکتوب یا غیرمکتوب، با کسب مجوز از دانشگاه شهید باهنر کرمان و استاد(ان) راهنما اقدام نمایم.
- ۲ - از درج اسمی افراد خارج از کمیته پایان نامه در جمع نویسنده‌گان مقاله‌های مستخرج از پایان نامه، بدون مجوز استاد(ان) راهنما اجتناب نمایم و اسمی افراد کمیته پایان نامه را در جمع نویسنده‌گان مقاله درج نمایم.
- ۳ - از درج نشانی یا وابستگی کاری (affiliation) نویسنده‌گان سازمان‌های دیگر (غیر از دانشگاه شهید باهنر کرمان) در مقاله‌های مستخرج از پایان نامه بدون تأیید استاد(دان) راهنما اجتناب نمایم.^۱
- ۴ - کلیه ضوابط و اصول اخلاقی مربوط به استفاده از موجودات زنده یا بافت‌های آنها را برای انجام پایان نامه رعایت نمایم.
- ۵ - در صورت اثبات تخلف (در هر زمان) مدرک تحصیلی صادر شده توسط دانشگاه شهید باهنر کرمان از درجه اعتبار ساقط و اینجانب هیچ‌گونه ادعایی تخواهم داشت.

کلیه حقوق مادی و معنوی این اثر (مقالات مستخرج، برنامه‌های رایانه‌ای، نرم افزارها و تجهیزات ساخته شده) مطابق با آئین نامه مالکیت فکری، متعلق به دانشگاه شهید باهنر کرمان است و بدون اخذ اجازه کتبی از دانشگاه قابل واگذاری به شخص ثالث نیست. همچنین استفاده از اطلاعات و نتایج این پایان نامه بدون ذکر مرجع مجاز نمی‌باشد. چنانچه مبادرت به عملی خلاف این تعهدنامه محرز گردد، دانشگاه شهید باهنر کرمان در هر زمان و به هر نحو مقتضی حق هرگونه اقدام قانونی را در استیفای حقوق خود دارد.

تاریخ و امضاء:
ناهید عبداللهی کرمانی
۱۴۰۴ فروردین

^۱ تنها آدرس مورد قبول برای دانشگاه به این صورت می‌باشد:

Shahid Bahonar University of Kerman, Kerman, Iran.

نام و آدرس واحدهای دانشگاه در تولیدات علمی محققان دانشگاه به تشخیص بخش و دانشکده به شرح زیر می‌باشد:
Department of Computer, Faculty of Engineering Shahid Bahonar University of Kerman, Kerman, Iran.

آدرس صحیح جهت درج در مقالات و سایر تولیدات علمی فارسی:
گروه (بخش) کامپیوتر، دانشکده فنی مهندسی، دانشگاه شهید باهنر کرمان، کرمان، ایران.

تقدیم به:

«این مجموعه را با کمال افتخار و احترام تقدیم می‌کنم به:
روح بلند بنیانگذار دانشگاه، مرحوم افضلی پور و همسر گرامیشان بانو فاخره صبا.
آنان که عاشقانه سوختند تا گرمابخش وجود ما و روشنگر راهمان باشند...
به پاس تعبیر عظیم و انسانی‌شان از کلمه ایثار.
به پاس عاطفه سرشار و گرمای امیدبخش وجودشان که در این سرددترین روزگاران بهترین پشتیبان ماست.
به پاس قلب بزرگشان، و به پاس محبت‌های بی دریغشان که هرگز فروکش نمی‌کند»

تشکر و قدردانی:

سپاس بی‌کران پروردگار یکتا را که هستی مان بخشد و در مسیر علم و دانش رهنمونمان شد و توفیق همنشینی با رهروان این راه را نصیمان ساخت، تا خوشچین خرمن معرفت باشیم.

به مصدقاق فرموده «من لَمْ يَشْكُرِ الْمُخْلوقَ لَمْ يَشْكُرِ الْخَالِقَ»، شایسته می‌دانم مراتب سپاس و قدردانی صمیمانه خود را تقدیم نمایم به استاد فرهیخته و فرزانه، جناب آقای دکتر مهدی افتخاری که با سعه‌صدر، حسن خلق و فروتنی، همواره راهنمای و پشتیبان من در مسیر انجام این پژوهش بوده‌اند و بی‌تردید این پایان‌نامه بدون هدایت‌ها و راهنمایی‌های ارزنده ایشان به ثمر نمی‌نشست.

همچنین بر خود لازم می‌دانم از استاد ارجمند، سرکار خانم دکتر سحر وحدتی که با راهنمایی‌ها و رهنمودهای علمی و دلسوزانه‌شان، نقش مؤثری در ارتقای سطح این پایان‌نامه ایفا نمودند، نهایت سپاس و امتنان را داشته باشم.

در آخر از صمیم قلب قدردان خانواده عزیزم هستم که با مهر و محبت بی‌دریغ، حمایت‌های بی‌پایان و دعاهای خیرشان، همواره در کنارم بودند و مشوق و همراه من در تمامی مراحل این مسیر پر فراز و نشیب شدند.

چکیده:

عملکرد مدل‌های زبانی بزرگ به شدت به مهندسی اعلان وابسته است. روش‌های دستی نظری برنامه‌ریزی و حل مسئله اگرچه توانسته‌اند فرآیند استدلال را بهبود بخشنده، اما غالباً در تضمین تنوع در اعلان تولیدی ناکارآمد باقی مانده و اثربخشی کلی را محدود می‌سازند. از طرفی روش مولد اعلان این محدودیت را با معرفی مکانیزم بهبود خودارجاعی برطرف نمود؛ به گونه‌ای که با بهره‌گیری از الگوریتم ژنتیک با رویکرد مسابقه دودویی، به تکامل تدریجی اعلان‌های دستوری می‌پردازد. این الگوریتم به مولد اعلان اجازه می‌دهد تا فضای اعلانات را به صورت تکرارشونده جستجو کرده و همزمان به بهینه‌سازی تنوع و عملکرد پردازد. با وجود پیشرفت‌های قابل توجه روش مولد اعلان در زمینه تولید اعلانات بهینه، چالش جدیدی به وجود آمده و آن، افزایش بار محاسباتی و پیچیدگی این روش است که استفاده عملی از آن را در بسیاری از کاربردها دشوار می‌سازد. برای رفع این مشکلات، ما روش تولید بهینه اعلان را پیشنهاد می‌کنیم که با به کارگیری یک راهبرد جستجوی محلی، دستورات را بهینه‌سازی می‌کند. این روش از مدل احتمالاتی به نام فرآیندهای نقطه‌ای دترمینانی، برای انتخاب دستورات متنوع و با کیفیت بالا استفاده می‌کند و به صورت مستقیم میان عملکرد و تنوع تعادل برقرار می‌کند، بدون آنکه به مکانیزم‌های پیچیده خودارجاعی وابسته باشد.

روش تولید بهینه اعلان را بر روی هشت مجموعه داده‌ی مرجع شامل AddSub ، SingleEq ، MultiArith ، AQuA-RAT ، CSQA ، SQA ، SVAMP ، GSM8K و AQuA-RAT مورد ارزیابی قراردادیم و به بهبود نسبی 23.4% نسبت به روش برنامه‌ریزی و حل و بهبود نسبی 54.7% نسبت به روش مولد اعلان دست پیدا کردیم.

وازگان کلیدی: مدل‌های زبانی بزرگ، مهندسی اعلان، اعلان دستوری، مدل احتمالاتی فرآیندهای نقطه‌ای دترمینانی، تنوع، کیفیت

فهرست مطالب

عنوان

صفحه

	کلیات پژوهش	۱
۱		
۲	مقدمه و بیان مسئله	۱-۱
۲	طراحی دستی اعلان	۱-۱-۱
۲	روش‌های بهینه‌سازی اعلان و خودکارسازی مهندسی اعلان	۲-۱-۱
۳	مجموعه داده‌های مربوطه	۳-۱-۱
۴	ضرورت تحقیق و اهداف	۲-۱
۵	سازماندهی پایان نامه	۳-۱
۷	پیشینه تحقیق و مفاهیم پایه	۲
۸	مقدمه	۱-۲
۸	بررسی مدل‌های زبانی بزرگ	۲-۲
۹	تاریخچه و تکامل مدل‌ها	۱-۲-۲
۱۰	معماری‌ها و کاربردهای اصلی	۲-۲-۲
۱۲	مدل بزرگ زبانی Mistral	۳-۲-۲
۱۳	مهندسی اعلان	۳-۲
۱۳	روش‌های دستی در مهندسی اعلان	۴-۲
۱۵	یادگیری درون متنی	۱-۴-۲
۱۹	روش زنجیره تفکر	۲-۴-۲
۱۹	روش استدلال بدون دیدن نمونه آموزشی	۳-۴-۲
۲۱	روش برنامه تفکر	۴-۴-۲
۲۱	روش بهینه سازی با اعلان	۵-۴-۲
۲۳	روش برنامه‌ریزی و حل	۶-۴-۲
۲۵	بهینه‌سازی خودکار اعلان‌ها	۵-۲
۲۵	روش زنجیره تفکر خودکار	۱-۵-۲
۲۶	روش مهندس اعلان خودکار	۲-۵-۲
۲۷	روش مولد اعلان	۳-۵-۲
۲۹	جمع‌بندی مباحث ارائه شده	۶-۲

۳۱	روش های پیشنهادی	۳
۳۲ مقدمه	۱-۳
۳۴ روش مولد اعلان ساده	۲-۳
۳۵ نمونه گیری	۱-۲-۳
۳۶ انتخاب	۲-۲-۳
۴۰ ارزیابی	۳-۲-۳
۴۲	نتایج و بحث	۴
۴۳ مقدمه	۱-۴
۴۳ مجموعه داده های مورد استفاده	۲-۴
۴۳ متغیرهای روش مولد اعلان ساده	۳-۴
۴۴ روش های مرجع	۴-۴
۴۵ نمایش دو بعدی اعلان ها	۵-۴
۴۵ نتایج	۶-۴
۴۶ سربار محاسباتی	۷-۴
۴۷ خروجی ها	۸-۴
۵۱	نتیجه گیری و پیشنهادات آتی	۵
۵۲ نتیجه گیری	۱-۵
۵۲ پیشنهادات آتی	۲-۵
۶۰	پیوست	
۶۲	توضیحات تکمیلی	
۶۲ مقدمه	
۶۲ اعلان های دستوری برای نمونه گیری روش مولد اعلان ساده	
۶۷ ارزیابی	
۷۰ سایر روش ها	

فهرست جداول

عنوان

صفحه

جدول ۳ - ۱ توصیف مسئله برای مجموعه داده های مختلف ۳۶

جدول ۴ - ۱ مقایسه دقیق روش مولد اعلان ساده با سایر روش های موجود ۴۳

جدول ۱ اعلان های دستوری برای سایر روش ها جهت مقایسه نتایج ۷۰

فهرست تصاویر

عنوان

صفحه

شکل ۲-۱	دیاگرام معماری ترانسفورمر	۱۱
شکل ۲-۲	مثالی از یادگیری بدون نمونه در یادگیری درون متنی	۱۶
شکل ۲-۳	مثالی از روش زنجیره تفکر برای حل یک سوال از دیتابست GSM8K	۲۰
شکل ۲-۴	مقایسه روش استدلال بدون دیدن نمونه آموزشی و روش زنجیره تفکر	۲۰
شکل ۲-۵	مقایسه روش برنامه تفکر و روش زنجیره تفکر	۲۲
شکل ۲-۶	دیاگرام روش بهینه سازی با اعلان	۲۳
شکل ۲-۷	یک مثال از روش برنامه ریزی و حل	۲۵
شکل ۲-۸	دیاگرام روش auto-CoT	۲۷
شکل ۲-۹	دیاگرام روش مولد اعلان	۲۹
شکل ۳-۱	مقایسه پیچیدگی محاسباتی روش مولد اعلان و روش مولد اعلان ساده	۳۳
شکل ۳-۲	دیاگرام روش مولد اعلان ساده	۳۴
شکل ۳-۳	دیاگرام مرحله ارزیابی روش مولد اعلان ساده	۴۱
شکل ۴-۱	نمایش دو بعدی بردار اعلان ها برای ارزیابی انتخاب در روش مولد اعلان ساده	۴۵

فهرست الگوریتم‌ها

عنوان	صفحة
الگوریتم ۱ - ۳ مولد اعلان ساده	٣٥

فصل اول:

کلیات پژوهش

۱-۱ مقدمه و بیان مسئله

در سال‌های اخیر، پیشرفت‌های چشمگیر در حوزه مدل‌های بزرگ زبانی^۱ موجب شده تا این مدل‌ها به عنوان ابزارهایی قدرتمند در پردازش زبان طبیعی^۲ و هوش مصنوعی شناخته شوند. اما عملکرد بهینه این مدل‌ها تا حد زیادی وابسته به کیفیت اعلان‌ها^۳ یا دستورات ورودی است؛ چرا که با توجه به ساختار این‌گونه مدل‌ها^۴ نحوه بیان اعلان‌ها به طور مستقیم بر تفسیر وظایف و تولید خروجی‌های مدل تأثیر می‌گذارد. از این‌رو، مهندسی اعلان^۵ به عنوان فرآیندی حیاتی برای بهبود عملکرد مدل‌ها مطرح شده است.

۱-۱-۱ طراحی دستی اعلان

روش‌های دستی طراحی اعلان مانند زنجیره تفکر^۶ [۱]، برنامه تفکر^۷ [۲] و برنامه‌ریزی و حل^۸ [۳] توانسته‌اند با تقسیم حل مسائل پیچیده به مراحل میانی، قدرت استدلال مدل‌ها را تقویت کنند؛ اما این روش‌ها به دلایل زیادی از جمله وابستگی به حدس و گمان انسانی، پوشش ناکافی فضای گسترده جستجوی اعلان‌ها و همچنین تغییرات سریع در قابلیت‌های مدل‌های زبانی، غالباً برای وظایف دامنه محور بهینه عمل نمی‌کنند. قابل ذکر است که فرآیند دستی طراحی و تنظیم اعلان‌ها زمان‌بر بوده و به علت پیچیدگی‌های موجود، نیاز به خودکارسازی این فرآیند بوجود آمد.

۱-۱-۲ روش‌های بهینه‌سازی اعلان و خودکارسازی مهندسی اعلان

جهت رفع محدودیت‌های روش‌های دستی، رویکردهای خودکارسازی مهندسی اعلان به کمک الگوریتم‌های بهینه‌سازی پیشنهاد شده‌اند. در این راستا، چندین رویکرد مطرح شده‌اند:

• الگوریتم‌های تکاملی^۹: روش‌هایی مانند مولد اعلان^{۱۰} [۱۰] از یک چارچوب تکاملی برپایه الگوریتم‌های ژنتیک استفاده می‌کنند. در این روش، اعلان‌های اولیه به صورت جمعیتی تولید می‌شوند و از طریق مراحل نمونه‌گیری، انتخاب و ارزیابی، نمونه‌های بهینه‌تر شناسایی و تقویت می‌شوند. این فرآیند که بر مبنای تغییرات تدریجی و ارزیابی مکرر استوار است، تلاش می‌کند تا بهترین نمونه‌های اعلان را

¹Large Language Models (LLMs)

²Natural Language Processing

³Prompts

⁴Transformers

⁵Prompt Engineering

⁶Chain-of-Thought (CoT)

⁷Program-of-Thoughts (PoT)

⁸Plan-and-Solve

⁹Evolutionary Algorithms

¹⁰PromptBreeder (PB)

برای وظایف مشخص پیدا کند؛ اما پیچیدگی چندلایه و وابستگی به تنظیمات دقیق، می‌تواند مانعی بر سر راه تعمیم آن در کاربردهای مختلف باشد.

- الگوریتم‌های مبتنی بر چارچوب‌های ریاضی: استفاده از روش‌هایی مانند فرآیندهای نقطه‌ای دترمینانی^۱ [۵][۶] به انتخاب زیرمجموعه‌ای از اعلان‌های متنوع و با کیفیت کمک می‌کند. این رویکرد به گونه‌ای طراحی شده که بتواند تنوع و کیفیت اعلان‌ها را به صورت همزمان بهینه کند و در نتیجه، تعداد فراخوانی‌های مدل‌های زبانی را کاهش داده و از لحاظ محاسباتی کارآمدتر باشد.
- الگوریتم‌های مبتنی بر جستجوی تصادفی و بهینه‌سازی گرادیان: برخی روش‌ها سعی در استفاده از تکنیک‌های جستجوی تصادفی و بهینه‌سازی گرادیان برای تنظیم خودکار اعلان‌ها دارند. اگرچه این روش‌ها نیز نویدبخش هستند، اما معمولاً به دلیل فضای جستجوی بسیار بزرگ و پیچیدگی‌های محاسباتی، نیاز به تنظیمات دقیق و منابع محاسباتی بالایی دارند.

۳-۱-۱ مجموعه داده‌های مربوطه

در حوزه مسائل ریاضی، مجموعه داده‌های متنوعی برای ارزیابی عملکرد اعلان‌های بهینه‌شده مورد استفاده قرار گرفته است. از جمله این مجموعه داده‌ها می‌توان به موارد زیر اشاره کرد:

- مجموعه داده‌های SingleEq [۷] و MultiArith [۸] : شامل مسائل ریاضی ساده و چند مرحله‌ای که توانایی مدل‌ها در حل مسائل محاسباتی پایه و استنتاج منطقی را مورد ارزیابی قرار می‌دهند.
- مجموعه داده‌های AddSub [۹] و SVAMP [۱۰] : مسائل جمع و تفریق و همچنین مسائل ترکیبی را شامل می‌شوند که دقت و سرعت حل مسائل را به چالش می‌کشند.
- مجموعه داده‌های SQA [۱۱] ، CSQA [۱۲] ، AQuA-RAT [۱۳] و GSMAK [۱۴] : شامل مسائل با سطوح مختلف دشواری، از جمله مسائل ریاضی پیچیده‌تر می‌باشند که علاوه بر محاسبات صحیح، نیاز به توانایی استدلال و توضیح روند حل مسئله نیز دارند.

این مجموعه داده‌ها به عنوان شاخص‌های استاندارد امکان ارزیابی دقیق و جامع عملکرد الگوریتم‌های بهینه‌سازی اعلان را فراهم می‌کنند و نقش مهمی در اثبات قابلیت تعمیم و کارایی روش‌های خودکارسازی مهندسی اعلان دارند.

^۱Determinantal Point Processes (DPPs)

۱-۲ ضرورت تحقیق و اهداف

همانطور که گفته شد، یکی از چالش‌های اساسی مهندسی اعلان بهبود و بهینه‌سازی اعلان‌ها به گونه‌ای است که توان مدل در تفسیر وظایف و تولید خروجی‌های دقیق تقویت شود. از چالش‌های موجود در اعلان‌سازی دستی میتوان به موارد زیر اشاره کرد:

- روش‌های دستی مبتنی بر تجربه و درک انسانی از مسئله طراحی شده‌اند و به تخصص و حدس انسانی وابسته‌اند. این وابستگی موجب عدم توانایی در پوشش فضای گسترده اعلان‌های ممکن شده است و اغلب به نتایج بهینه نمی‌رسند.
- فرآیند تنظیم و آزمایش دستی اعلان‌ها نیازمند تلاش‌های تکراری و زمان‌بر است که می‌تواند منابع محاسباتی و انسانی را به شدت مصرف کند.
- روش‌های دستی معمولاً برای یک دامنه یا وظیفه خاص طراحی می‌شوند و در مواجهه با تغییرات دامنه یا وظایف جدید، عملکرد مناسبی از خود نشان نمی‌دهند.

در این راستا، ضرورت تحقیق و توسعه رویکردهای خودکار در مهندسی اعلان به دلایل زیر برجسته می‌شود:

- استفاده از الگوریتم‌های بهینه‌سازی خودکار می‌تواند نیاز به تنظیم دستی و آزمون‌های متعدد را کاهش دهد و در نتیجه مصرف منابع را بهبود بخشد.
- یک سیستم خودکار قادر است به طور همزمان به دنبال اعلان‌های با کیفیت و متنوع بگردد. این امر باعث می‌شود که مدل‌های زبانی بتوانند در وظایف دامنه‌محور (به ویژه در حل مسائل ریاضی) عملکرد بهتری داشته باشند؛ چرا که از تنوع مثال‌ها و راه حل‌های ارائه شده بهره‌مند می‌شوند.

در زمینه مسائل ریاضی، که معمولاً شامل مجموعه داده‌هایی می‌شوند که در بخش قبل به تفصیل توضیح داده شدند، چالش‌های خاصی از جمله نیاز به استدلال دقیق و توان حل مسائل چند مرحله‌ای مطرح است. بهبود اعلان‌ها در این حوزه می‌تواند منجر به افزایش دقت حل مسائل ریاضی، بهبود توان استدلال و توضیح روند حل مسئله و همچنین فراهم آوردن یک چارچوب ارزیابی استاندارد برای مقایسه روش‌های مختلف مهندسی اعلان شود.

با توجه به چالش‌ها و نیازهای مطرح شده، اهداف اصلی این تحقیق بدین شرح می‌باشد:

- توسعه یک چارچوب ساده و کارآمد و معرفی رویکردنی نوین به نام تولید بهینه اعلان^۱ که از الگوریتم‌های تکاملی و مدل احتمالاتی فرآیندهای نقطه‌ای دترمینانی بهره می‌گیرد تا اعلان‌های با کیفیت و متنوع تولید کند.
- با خودکارسازی فرآیند مهندسی اعلان، تلاش می‌شود تا نیاز به تخصص و حدس انسانی کاهش یابد و نتایج به صورت سیستماتیک و قابل تکرار حاصل شود.
- رویکرد تولید بهینه اعلان دقیق و کارایی مدل‌های زبانی را در حل مسائل دامنه محور از جمله مسائل ریاضی که از نظر استدلال و محاسبات چالش برانگیز هستند، بهبود می‌بخشد.
- روش تولید بهینه اعلان با کاهش تعداد فراخوانی‌های مدل‌های زبانی در فرآیند بهینه‌سازی، از منابع محاسباتی به شکل بهینه‌تری استفاده می‌کند.
- این چارچوب به سادگی قابل تعمیم به وظایف و دامنه‌های مختلف است و در مواجهه با تغییرات ساختاری مدل‌های زبانی، عملکرد مناسبی ارائه می‌دهد.

در مجموع، این تحقیق با هدف ارائه راهکاری نوین و کارآمد در حوزه مهندسی اعلان، به دنبال ایجاد توازنی بین کیفیت، تنوع و کارایی محاسباتی است که بتواند نیازهای رو به رشد دنیای هوش مصنوعی و پردازش زبان طبیعی را برآورده سازد.

۳-۱ سازماندهی پایان نامه

در این پایان‌نامه، ساختار مطالب به گونه‌ای تدوین شده که مسیر پژوهش از مبانی نظری و معرفی مسئله تا ارائه نتایج تجربی به صورت پیوسته و منطقی دنبال شود. به عبارت دیگر، هدف از سازماندهی مطالب این است که خواننده بتواند به راحتی با مباحث پایه، چالش‌ها، روش‌های موجود، و نوآوری‌های پیشنهادی آشنا شود و در نهایت به درک جامع از دستاوردهای تحقیق دست یابد. ساختار کلی پایان‌نامه به شرح زیر است:

- فصل دوم - پیشینه تحقیق و مفاهیم پایه:
در این فصل، ابتدا به بررسی کلی مدل‌های بزرگ زبانی و اهمیت مهندسی اعلان پرداخته می‌شود. سپس، چالش‌ها و محدودیت‌های روش‌های دستی بیان شده و مسئله تحقیق به تفصیل معرفی می‌شود. هدف این فصل ایجاد زمینه نظری مناسب برای درک اهمیت بهینه‌سازی خودکار اعلان‌هاست.

¹SimplePromptBreeder (SPB)

در ادامه به بررسی جامع مطالعات پیشین در حوزه بهینه‌سازی اعلان‌ها پرداخته می‌شود. در این بخش، رویکردهای مختلف از جمله الگوریتم‌های تکاملی و سایر روش‌های خودکارسازی مهندسی اعلان مورد تحلیل قرار می‌گیرند. نقاط قوت و ضعف هر یک از این رویکردها همراه با چالش‌های موجود در هر کدام به تفصیل بررسی می‌شود.

• فصل سوم – روش پیشنهادی (تولید بهینه اعلان) :

در این فصل، رویکرد نوین پیشنهادی با عنوان تولید بهینه اعلان به صورت کامل تشریح می‌شود. ابتدا مفاهیم نظری و چارچوب ریاضی مورد استفاده، به ویژه استفاده از مدل احتمالاتی فرآیندهای نقطه‌ای دترمینانی برای تضمین تنوع و کیفیت اعلان‌ها، توضیح داده می‌شود. سپس، مراحل گام به گام الگوریتم شامل نمونه‌گیری، انتخاب و ارزیابی اعلان‌ها به تفصیل بیان شده و نوآوری‌های اصلی این روش نسبت به سایر روش‌ها برجسته می‌شود.

• فصل چهارم – نتایج و بحث:

این فصل به ارائه نتایج آزمایش‌های انجام شده بر روی چندین مجموعه داده معتبر اختصاص دارد. مجموعه داده‌های به کار رفته شامل مجموعه‌های مسائل ریاضی می‌باشد که هر یک از آن‌ها چالش‌های خاص خود در زمینه استدلال و حل مسئله را به نمایش می‌گذارند. در این فصل، عملکرد روش تولید بهینه اعلان از نظر دقیق، کارایی و صرفه‌جویی در منابع محاسباتی مورد مقایسه قرار گرفته و نتایج به دست آمده تحلیل می‌شوند.

• فصل پنجم – نتیجه‌گیری و پیشنهادات آتی:

در فصل نهایی، یافته‌های اصلی تحقیق به طور خلاصه ارائه شده و به نتیجه‌گیری کلی از دستاوردهای پژوهش پرداخته می‌شود. در این بخش، چالش‌های باقی‌مانده، محدودیت‌های تحقیق و نیز پیشنهاداتی جهت تحقیقات آتی و بهبود رویکرد ارائه می‌شود.

فصل دوم:

پیشینه تحقیق و مفاهیم پایه

مدل‌های زبانی بزرگ بر اساس داده‌های متنی گستردۀ آموزش دیده و قابلیت درک و تولید زبان طبیعی را دارند. عملکرد این مدل‌ها بر پایه پیش‌بینی کلمات بعدی در یک جمله یا به عبارتی تکمیل متن‌ون بر اساس ورودی داده‌شده است.

در سال‌های اخیر، مدل‌های زبانی بزرگ رشد چشمگیری داشته‌اند و به دنبال این رشد، ایده استفاده از این مدل‌ها در تمامی زمینه‌های پردازش زبان طبیعی مورد توجه قرار گرفته است. قابلیت یادگیری مبنی بر محتوا^۱، که باعث می‌شود مدل‌ها به ورودی‌های دریافتی دقت ویژه‌ای کنند، کیفیت متن تولید شده جدید را به متن ورودی وابسته می‌کند. فلذًا مهندسی اعلان یکی از اجزای کلیدی در بهبود عملکرد مدل‌های زبانی است. با طراحی دقیق اعلان‌ها، می‌توان ورودی‌های مدل را به گونه‌ای تنظیم کرد که پاسخ‌های تولیدی با اهداف و نیازهای خاص همخوانی بیشتری داشته باشند. این فرایند نه تنها به بهبود کیفیت و دقت خروجی‌های مدل کمک می‌کند، بلکه در کنترل و هدایت رفتار آن در مواجهه با وظایف مختلف نقش حیاتی دارد. استفاده از مهندسی اعلان زمینه‌ساز تطبیق بهتر مدل با شرایط متغیر و کاهش ابهامات در تولید جواب است.

مسئله اصلی این تحقیق، چالش‌های موجود در طراحی دستی اعلان‌هاست که به دلیل پیچیدگی و زمان بر بودن، نیازمند راهکارهایی خودکار می‌باشد. در این راستا، هدف این تحقیق ارائه روشی مبنی بر الگوریتم جستجو در فضای اعلان‌ها برای تولید خودکار اعلان‌های بهینه است. این رویکرد می‌تواند باعث بهبود دقت و کارایی مدل‌های زبانی شده و از بروز خطاهای ناشی از طراحی دستی جلوگیری کند.

۲-۲ بررسی مدل‌های زبانی بزرگ

مدل‌های زبانی بزرگ سیستم‌های پیشرفته‌ای هستند که با بهره‌گیری از تکنیک‌های یادگیری عمیق، توانایی پردازش و تولید زبان طبیعی را به سطحی بالا رسانده‌اند. این مدل‌ها با تحلیل حجم عظیمی از داده‌های متنی، قادر به درک مفاهیم، استخراج اطلاعات و تولید متنی دقیق و معنادار می‌باشند. از کاربردهای آن‌ها می‌توان به ترجمه، خلاصه‌سازی، پاسخگویی به سوالات و حتی تولید محتوا در حوزه‌های مختلف اشاره کرد. تحولات اخیر در این حوزه، مسیر توسعه و بهبود این مدل‌ها را هموار ساخته است. شناخت تاریخچه و معماری این مدل‌ها نقش مهمی در درک عملکرد و پتانسیل‌های آن‌ها دارد که در ادامه به آن می‌پردازیم.

¹In-context Learning

۱-۲-۲ تاریخچه و تکامل مدل‌ها

تکامل مدل‌های زبانی، مسیر پیچیده‌ای از رویکردهای اولیه‌ی مبتنی بر قواعد نمادین تا استفاده از شبکه‌های عصبی پیشرفته و معماری‌های نوین مانند ترنسفورمر را در بر می‌گیرد. در ادامه به تفصیل به بررسی مراحل مختلف این تکامل پرداخته می‌شود.

دوران اولیه: رویکردهای نمادین و قواعد دست‌نویس

در دهه‌های ۱۹۵۰ و ۱۹۶۰، اولین تلاش‌ها برای پردازش زبان طبیعی به وسیله‌ی روش‌های نمادین انجام شد. پژوهشگران در آن زمان سعی می‌کردند ساختارهای دستوری و قوانین زبان را به صورت صریح و دستی تعریف کنند. این رویکردها با وجود تلاش‌های ارزشمند، به دلیل محدودیت‌های محاسباتی و عدم وجود داده‌های کافی، نتوانستند به دقت و کارایی مورد انتظار دست یابند.

ورود به عصر آماری

با گذر زمان و ورود به دهه‌های ۱۹۶۰ و ۱۹۷۰، رویکردهای آماری جایگزین بخش‌هایی از روش‌های نمادین شدند. در این دوران، مدل‌های n-gram که بر مبنای احتمال وقوع یک کلمه با توجه به کلمات قبلی محاسبه می‌شدند، به عنوان اولین قدم‌های موفق در مدل‌سازی زبان مطرح شدند. اگرچه این مدل‌ها ساده بودند، اما توانستند برخی از پیچیدگی‌های اولیه‌ی پردازش زبان را کاهش دهند.

ظهور یادگیری ماشین و شبکه‌های عصبی

در دهه‌های ۱۹۸۰ و ۱۹۹۰، با پیشرفت‌های چشمگیر در فناوری‌های محاسباتی و افزایش دسترسی به داده‌های متنی، روش‌های یادگیری ماشین وارد عرصه شدند. الگوریتم‌های یادگیری ناظرت شده و غیرناظارتی به منظور تشخیص الگوهای زبانی به کار گرفته شدند. با این حال، محدودیت‌های موجود همچنان مانع از دستیابی به درک عمیق‌تر و تولید متن‌های طبیعی به سطح امروزی می‌شدند.

عصر شبکه‌های عصبی عمیق

ورود به قرن ۲۱ و بهویژه دهه ۲۰۱۰، با ظهور شبکه‌های عصبی عمیق مانند شبکه‌های عصبی بازگشتی^۱ و شبکه‌های حافظه بلندمدت^۲ همراه بود. این مدل‌ها توانستند وابستگی‌های زمانی و روابط بلندمدت موجود در متن را بهتر مدل‌سازی کنند. با این حال، چالش‌هایی همچنان در زمینه بهبود کیفیت و کارایی تولید متن وجود داشت.

انقلاب ترنسفورمر و ظهور مدل‌های بزرگ

نقشه عطف مهم در تکامل مدل‌های زبانی، معرفی معماری ترنسفورمر [۱۵] بود. این معماری با بهره‌گیری از مکانیزم توجه^۳ توانست وابستگی‌های بین کلمات را به صورت موازی و با کارایی بالا پردازش کند.

¹Recursive Neural Networks (RNNs)

²Long short-term memory (LSTM)

³Attention Mechanism

ویژگی‌های کلیدی ترنسفورمر شامل پردازش موازی داده‌ها، درک بهتر وابستگی‌های طولانی مدت در متن و افزایش سرعت و بهبود کارایی مدل‌های زبانی است.

توسعه مدل‌های پیشرفته مانند GPT و BERT

با معرفی ترنسفورمر، مدل‌های بزرگی نظیر GPT و BERT توسعه یافته‌ند:

- GPT^۱: این مدل‌ها با افزایش تعداد پارامترها (به عنوان مثال، ۳-GPT با ۱۷۵ میلیارد پارامتر) توانسته‌اند وظایفی مانند ترجمه، خلاصه‌سازی و پاسخ به سؤالات را با دقت بسیار بالا انجام دهند.
- BERT^۲: این مدل با تمرکز بر درک بهتر معنایی کلمات در متن، در وظایف مختلف پردازش زبان عملکرد قابل توجهی از خود نشان داده است.

تکنیک‌های بهبود عملکرد: تنظیم دقیق و یادگیری انتقالی

علاوه بر افزایش تعداد پارامترها و مقیاس داده‌های آموزشی، تکنیک‌هایی مانند تنظیم دقیق^۳ و یادگیری انتقالی^۴ نقش مهمی در بهبود عملکرد مدل‌های زبانی داشته‌اند. این تکنیک‌ها امکان تطبیق مدل‌های پیش‌آموزش داده شده با وظایف خاص را فراهم می‌آورند که باعث بهبود کیفیت و دقت در کاربردهای متنوع می‌شود.

پیشرفتهای حاصل از توسعه مدل‌های زبانی بزرگ، مرزهای جدیدی در تعامل انسان و ماشین ایجاد کرده است. سیستم‌های هوشمند مبتنی بر این مدل‌ها قادرند که به صورت طبیعی و انسانی با کاربران تعامل کنند، همچنین در زمینه‌های مختلفی از جمله خدمات مشتری، ترجمه ماشینی، تحلیل متون و تولید محتوا به کار گرفته شوندو وظایف پیچیده زبانی را با دقت و سرعت بالا انجام دهند.

تکامل مدل‌های زبانی از روش‌های نمادین اولیه به سوی استفاده از شبکه‌های عصبی عمیق و معماری‌های پیشرفته مانند ترنسفورمر، نشان‌دهنده یک مسیر پر فراز و نشیب اما پر از نوآوری است. این پیشرفتهای بهبود قابل توجهی در درک و تولید زبان انسانی ایجاد کرده و نقش مهمی در توسعه فناوری‌های هوش مصنوعی و تعامل انسان-ماشین داشته‌اند.

۲-۲-۲ معماری‌ها و کاربردهای اصلی

در ادامه، به تفصیل به بررسی معماری‌های کلیدی این مدل‌ها می‌پردازیم.

۱. معماری ترنسفورمر^۵: معماری ترنسفورمر پایه و اساس بسیاری از مدل‌های زبانی بزرگ است.

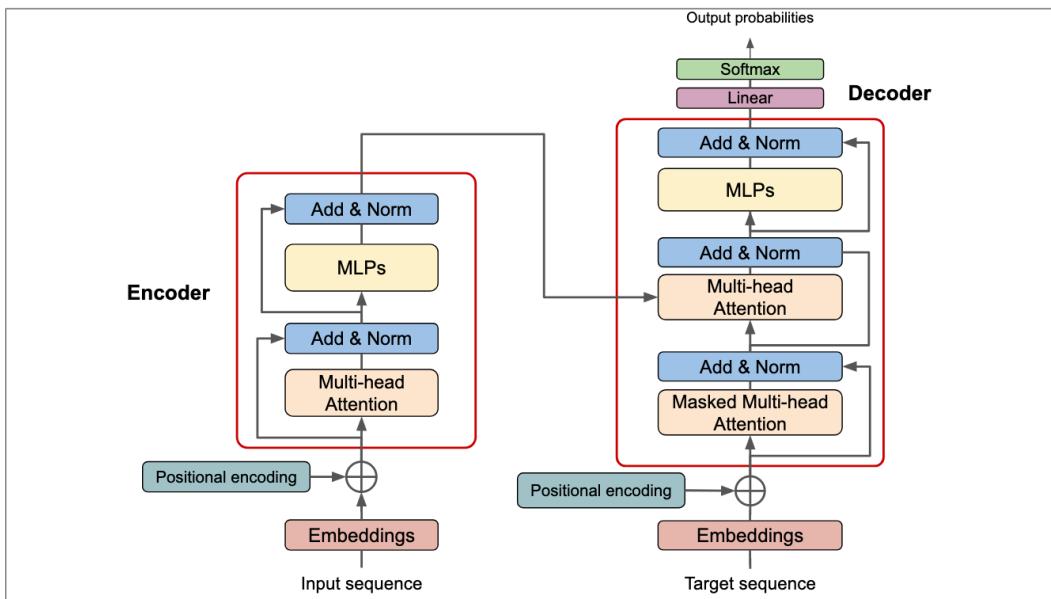
¹Generative Pre-trained Transformer

²Bidirectional Encoder Representations from Transformers

³Fine-tuning

⁴Transfer Learning

⁵Transformer



شکل ۱-۲ دیاگرام معماری ترنسفورمر

این معماری با استفاده از مکانیزم توجه^۱، امکان پردازش موازی داده‌ها و درک وابستگی‌های طولانی مدت در متن را فراهم می‌کند. برخلاف مدل‌های پیشین که بر پایه شبکه‌های عصبی بازگشتی^۲ بودند، ترنسفورمرها با حذف وابستگی‌های ترتیبی، کارایی و سرعت پردازش را بهبود بخشیدند. دیاگرام این معماری در شکل ۱-۲ نمایش داده شده است.

۲. مکانیزم توجه: مکانیزم توجه به مدل‌ها اجازه می‌دهد تا به بخش‌های مختلف ورودی با وزن‌های متفاوت نگاه کنند و وابستگی‌های معنایی را بهتر درک نمایند. این مکانیزم به ویژه در ترجمه ماشینی و خلاصه‌سازی متون کاربرد دارد.

۳. معماری‌های خودبازگشتی^۳ و خودرمزنگذار^۴:

- مدل‌های خودبازگشتی مانند سری GPT متن را به صورت ترتیبی تولید می‌کنند و هر کلمه را بر اساس کلمات قبلی پیش‌بینی می‌نمایند.
- مدل‌های خودرمزنگذار مانند BERT با استفاده از ماسک‌کردن کلمات در ورودی، سعی در درک زمینه و پیش‌بینی کلمات ماسک‌شده دارند.

¹Attention Mechanism

²Recursive Neural Networks

³Autoregressive

⁴Autoencoder

۳-۲-۲ مدل بزرگ زبانی Mistral

مدل بزرگ زبانی Mistral [۳۴] از جمله دستاوردهای جدید در حوزه پردازش زبان طبیعی به شمار می‌آید که با بهره‌گیری از معماری ترانسفورمر، عملکرد بالایی در وظایف متعدد زبانی نشان داده است. در ادامه به بررسی جامع این مدل می‌پردازیم.

مدل Mistral بر پایه معماری ترانسفورمر طراحی شده است. همانطور که گفته شد، این معماری با استفاده از مکانیزم توجه قادر است وابستگی‌های طولانی مدت در متون را به صورت موازی پردازش کند. از ویژگی‌های برجسته این مدل می‌توان به استفاده از تعداد پارامترهای بالا^۱ اشاره کرد که موجب بهبود دقت و کیفیت تولید متن می‌شود.

برای رسیدن به عملکرد مطلوب، مدل Mistral با استفاده از مجموعه‌های داده گسترده و متعدد آموخته داده شده است. علاوه بر این، بهره‌گیری از تکنیک‌های تنظیم دقیق^۲ و بهینه‌سازی پیشرفته، موجب شده تا مدل بتواند در وظایف خاص، همچون ترجمه، خلاصه‌سازی و پاسخ به پرسش، عملکرد بهتری از خود نشان دهد.

مدل Mistral در حوزه‌های مختلف پردازش زبان طبیعی کاربرد دارد. از وظایف مهم آن می‌توان به توانایی تولید متونی با کیفیت بالا و طبیعی، ارائه ترجمه‌های دقیق و روان بین زبان‌ها، استخراج اطلاعات کلیدی و ارائه خلاصه‌های مفید از متون طولانی و همچنین درک دقیق سوالات و ارائه پاسخ‌های مرتبط و دقیق اشاره کرد.

با وجود این توانایی‌ها، مدل Mistral دارای نوآوری‌ها و مزایای متعددی نیز است که آن را از سایر مدل‌های زبانی متمایز می‌کند. این نوآوری‌ها شامل بهره‌گیری از معماری ترانسفورمر جهت پردازش موازی و بهبود سرعت محاسبات است، همچنین استفاده از تعداد پارامترهای بالا به افزایش دقت و کیفیت خروجی‌ها منجر می‌شود و قابلیت تنظیم دقیق برای تطبیق با وظایف خاص و کاربردهای صنعتی و پژوهشی را فراهم می‌آورد و به دنبال آن، بهبود چشمگیری در درک وابستگی‌های زبانی و تولید متون طبیعی حاصل می‌شود.

با وجود دستاوردهای قابل توجه، مدل Mistral همچنان با چالش‌هایی همچون مصرف بالای منابع محاسباتی و نیاز به داده‌های آموزشی گسترده مواجه است. پژوهش‌های آتی در زمینه بهبود کارایی، کاهش هزینه‌های محاسباتی و افزایش دقت در کاربردهای خاص، افق‌های روشن‌تری را برای این مدل ترسیم می‌کند.

^۱ در این پژوهش از مدل‌های ۷ میلیارد پارامتری استفاده شده است

^۲Fine-tuning

۳-۲ مهندسی اعلان

مهندسي اعلان^۱ به فرآيند طراحی و بهينه‌سازی ورودی‌های متنی اطلاق می‌شود که به مدل‌های زبانی بزرگ ارائه می‌گردد تا خروجی‌های مطلوب و دقیقی تولید کنند. این ورودی‌ها می‌توانند شامل دستورات، سوالات یا داده‌های زمینه‌ای باشند که به مدل کمک می‌کنند تا پاسخ‌های خود را در چارچوب معنایی و ساختاری مشخصی ارائه دهد. می‌دانیم که مهندسی اعلان تأثیر مستقیمی بر کارایی و دقت مدل‌های زبانی بزرگ دارد از این رو با تدوین اعلان‌های دقیق و مناسب با وظیفه موردنظر، می‌توان رفتار مدل را به گونه‌ای هدایت کرد که خروجی‌های مرتبطتر و با کیفیت‌تری تولید کند. این امر به‌ویژه در شرایطی که داده‌های آموزشی محدود یا ناموجود هستند، اهمیت بیشتری پیدا می‌کند. از مزایای مهندسی اعلان می‌توان به موارد زیر اشاره کرد :

- با استفاده از اعلان‌های دقیق و مناسب می‌توان نتایج مدل را به سمت پاسخ‌های موردنظر هدایت کرد و کنترل بیشتری بر خروجی مدل داشت.
- مهندسی اعلان می‌تواند به کاهش سوگیری‌های موجود در مدل‌های زبانی کمک کند و نتایج منصفانه‌تری ارائه دهد.
- با تدوین اعلان‌های مؤثر، می‌توان زمان و منابع موردنیاز برای رسیدن به نتایج مطلوب را کاهش داد و کارایی را افزایش داد.

در نتیجه، مهندسی اعلان به عنوان ابزاری قدرتمند برای بهبود عملکرد مدل‌های زبانی بزرگ محسوب می‌شود و نقش کلیدی در توسعه و بهره‌برداری مؤثر از این مدل‌ها ایفا می‌کند.

۴-۲ روش‌های دستی در مهندسی اعلان

مهندسي اعلان دستی به فرآيند طراحی و بهينه‌سازی دستی ورودی‌ها (اعلان‌ها) برای هدایت بهتر مدل‌های زبانی بزرگ در تولید پاسخ‌های مطلوب گفته می‌شود. برخلاف روش‌های خودکار که از الگوریتم‌های یادگیری ماشین برای بهینه‌سازی اعلان‌ها استفاده می‌کنند، روش‌های دستی بر دانش زبانی، شهود انسانی و آزمایش‌های مکرر متکی هستند. این تکنیک‌ها در کاربردهای واقعی که نیاز به کنترل دقیق بر خروجی مدل دارند، مانند پاسخ‌گویی به سوالات، خلاصه‌سازی متون و انجام وظایف استدلالی، به کار گرفته می‌شوند.

مهندسي اعلان دستی برای افزایش اثربخشی مدل‌های زبانی ضروری است، به‌ویژه در مواردی که تنظیم و آموزش مجدد مدل امکان‌پذیر نیست. از آنجایی که مدل‌های زبانی پاسخ‌های خود را بر اساس ورودی‌ها

¹Prompt Engineering

تولید می‌کند، حتی تغییرات جزئی در ساختار یا نحوه بیان اعلان‌ها می‌تواند تأثیر قابل‌توجهی بر عملکرد آن‌ها داشته باشد. اعلان‌های طراحی شده به صورت بهینه می‌توانند دقت مدل را افزایش داده، توانایی استدلال آن را بهبود بخشنده و میزان سوگیری در پاسخ‌ها را کاهش دهند، در نتیجه پاسخ‌های دقیق‌تر و متناسب‌تری ارائه کنند.

از ویژگی‌های کلیدی مهندسی اعلان دستی می‌توان به موارد زیر اشاره کرد :

۱. طراحی مبتنی بر دانش انسانی

- برخلاف روش‌های خودکار که از الگوریتم‌های بهینه‌سازی استفاده می‌کنند، مهندسی اعلان دستی بر شهود انسانی و دانش زبانی تکیه دارد.
- درک صحیح از زبان و زمینه موردنظر نقش مهمی در طراحی اعلان‌هایی دارد که مدل را به تولید خروجی‌های مطلوب هدایت می‌کنند.

۲. بهینه‌سازی تدریجی و تکرارشونده

- طراحی اعلان‌های مؤثر نیازمند آزمایش‌های مداوم و اصلاحات متوالی است.
- تنظيمات و تغییرات مداوم در نحوه بیان اعلان به شناسایی بهترین ساختار و سبک ورودی کمک می‌کند.

۳. انعطاف‌پذیری در کاربردهای مختلف

- مهندسي اعلان دستی امکان شخصی‌سازی ورودی‌ها را برای وظایف متنوعی مانند تولید محظوظ، برنامه‌نویسی و استدلال منطقی فراهم می‌کند.
- استراتژی‌های خاصی را می‌توان برای هر کاربرد به کار گرفت، مانند ارائه دستورالعمل‌های گام‌به‌گام، اضافه کردن نشانه‌های متنی یا استفاده از نمونه‌های مشابه.

۴. کنترل و تفسیرپذیری بهتر

- از آنجا که اعلان‌های دستی توسط انسان طراحی می‌شوند، کنترل بیشتری بر رفتار مدل فراهم می‌کنند.
- این روش امکان درک بهتر نحوه پاسخ‌گویی مدل به ورودی‌های مختلف را فراهم کرده و به عیب‌یابی و بهبود عملکرد کمک می‌کند.

با وجود مزایای فراوان، این روش با چالش‌هایی همراه است:

- فرآیند زمانبر: طراحی و بهینه‌سازی دستی اعلان‌ها نیاز به صرف زمان زیادی دارد.
- مقیاس‌پذیری پایین: برخلاف روش‌های خودکار، اعلان‌های دستی به راحتی برای مدل‌ها یا وظایف دیگر تعمیم نمی‌یابند.
- ماهیت مبتنی بر آزمون و خطاب: یافتن اعلان بهینه اغلب نیازمند آزمایش‌های متعدد است که همیشه نتیجه‌های ثابت و پایدار را تضمین نمی‌کند.

در نهایت مهندسی اعلان دستی به عنوان رویکردی بینادین برای بهینه‌سازی تعاملات با مدل‌های زبانی شناخته می‌شود. در بخش‌های بعدی، روش‌های مختلف مهندسی اعلان دستی را بررسی خواهیم کرد و تأثیر هر یک را بر توانایی‌های استدلالی و کیفیت پاسخ‌دهی مدل می‌سنجمیم.

۱-۴-۲ یادگیری درون‌متنی

یادگیری درون‌متنی^۱ از قابلیت‌های برجسته مدل‌های زبانی بزرگ است که به آن‌ها اجازه می‌دهد بدون نیاز به بازآموزی^۲ یا تنظیم مجدد وزن‌ها، تنها با دریافت چند نمونه در ورودی، وظایف جدید را تشخیص دهند و به درستی انجام دهند. این روش به مدل کمک می‌کند با تحلیل مثال‌های ارائه‌شده در ورودی، پاسخ‌هایی متناسب با همان زمینه تولید کند. از مزایای مهم این روش، افزایش انعطاف‌پذیری در مواجهه با وظایف و موضوعات جدید است که باعث می‌شود مدل‌های زبانی به سرعت خود را با زمینه‌های مختلف تطبیق دهند. با حذف نیاز به بازآموزی برای هر وظیفه جدید، این روش می‌تواند بهینه‌سازی قابل توجهی در مصرف منابع و زمان ایجاد کند. از سوی دیگر، این قابلیت نقش مهمی در بهبود کیفیت تعاملات میان انسان و ماشین ایفا می‌کند. مدل‌ها با درک بهتر زمینه و مثال‌های ارائه‌شده، پاسخ‌هایی طبیعی‌تر و دقیق‌تر تولید می‌کنند که باعث افزایش رضایت کاربران می‌شود.

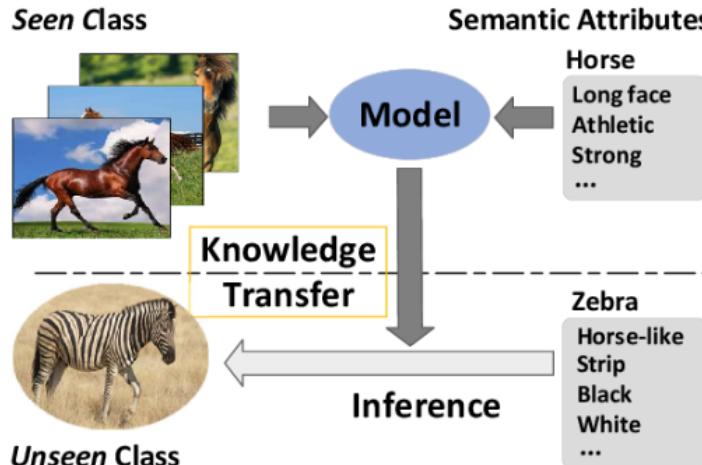
با وجود مزایای فوق، یادگیری درون‌متنی با چالش‌های نیز همراه است. عملکرد مدل‌ها به شدت به کیفیت و تنوع داده‌ها و مثال‌های ورودی وابسته است. مثال‌های ناقص یا ناسازگار می‌توانند منجر به تولید پاسخ‌های نادرست شوند. همچنین، در برخی موقعیت‌های پیچیده یا مبهم، مدل ممکن است در درک دقیق زمینه دچار مشکل شود.

از روش‌های یادگیری درون‌متنی می‌توان به یادگیری بدون نمونه و یادگیری با نمونه‌های کم اشاره کرد. در یادگیری بدون نمونه [۱۸]^۳ مدل‌ها وظیفه دارند اشیاء یا مفاهیمی را که در طول آموختش با آن‌ها رویرونشده‌اند شناسایی و دسته‌بندی کنند. برخلاف یادگیری نظارت‌شده سنتی که نیاز به مثال‌های برچسب‌خورده

¹In-Context Learning

²Training

³Zero-Shot Learning (ZSL)



شکل ۲-۲ مثالی از یادگیری بدون نمونه در یادگیری درون متنی

برای هر کلاس دارد، یادگیری بدون نمونه به مدل‌ها این امکان را می‌دهد که پیش‌بینی‌هایی در مورد دسته‌های دیده نشده در زمان آموزش با استفاده از اطلاعات کمکی انجام دهنند. برای شناسایی کلاس‌های دیده نشده در زمان آموزش، مدل‌های یادگیری بدون نمونه روابطی بین کلاس‌های مشاهده شده (دیده شده) و کلاس‌های مشاهده نشده (دیده نشده) برقرار می‌کنند. این ارتباط معمولاً از طریق ویژگی‌های مشترک یا اطلاعات معنایی که این دو به هم پیوند می‌دهند، تسهیل می‌شود. از طرفی وجود اطلاعات کمکی برای یادگیری بدون نمونه حیاتی است و داده‌های توصیفی در مورد کلاس‌های دیده شده و دیده نشده از ارائه می‌دهد. این اطلاعات می‌توانند شامل ویژگی‌هایی مانند توضیحات متنی، جاسازی‌های معنایی یا سایر داده‌های مرتبط باشد که به مدل کمک می‌کند تا کلاس‌ها را درک کرده و از هم تمایز دهد.

چگونگی عملکرد یادگیری بدون نمونه بدین صورت است که در غیاب مثال‌های برچسب‌خورده برای کلاس‌های دیده نشده، برای پر کردن شکاف به اطلاعات کمکی اتکا می‌کنند. به عنوان مثال، در طبقه‌بندی تصاویر، مدلی که بر اساس گونه‌های خاص حیوانات آموزش دیده باشد، می‌تواند از توضیحات متنی گونه‌های دیده نشده برای شناسایی صحیح آن‌ها استفاده کند. با درک اینکه یک گورخر مشابه یک اسب است اما با خطوط راهراه، مدل می‌تواند گورخرها را بدون دیدن آن‌ها در طول آموزش شناسایی کند (تصویر ۲-۲).

یادگیری بدون نمونه در حوزه‌های مختلفی کاربرد داشته است، از جمله:

- طبقه‌بندی تصویر: شناسایی اشیاء یا گونه‌هایی که در داده‌های آموزشی وجود ندارند.
- بخش‌بندی معنایی: بخش‌بندی اشیاء دیده نشده در تصاویر بر اساس ویژگی‌های یادگرفته شده.
- پردازش زبان طبیعی: انجام کارهایی مانند طبقه‌بندی متن یا شناسایی موجودیت‌ها بدون مثال‌های صریح.
- زیست‌شناسی محاسباتی: پیش‌بینی عملکرد ژن‌ها یا پروتئین‌هایی که قادر نشانه‌های آزمایشی هستند.

از چالش‌های مهم یادگیری بدون نمونه می‌توان به احتمال اشتباه کردن مدل در دسته‌بندی کلاس‌های دیده نشده اشاره کرد، به ویژه زمانی که این کلاس‌ها ویژگی‌های مشابه با کلاس‌های دیده شده دارند. تحقیقات جاری به دنبال افزایش کارایی و دقت مدل‌های یادگیری بدون نمونه است و روش‌هایی مانند تولید داده‌های مصنوعی برای کلاس‌های دیده نشده و بهبود کیفیت اطلاعات کمکی را بررسی می‌کنند.

در طرف دیگر ماجرا، برای یادگیری با نمونه‌های کم [۱۹] مدل‌ها طوری طراحی می‌شوند که بتوانند بر اساس تعداد بسیار محدودی از نمونه‌های آموزشی، یادگیری کرده و پیش‌بینی‌های دقیقی انجام دهنند. این رویکرد به ویژه در موقعیت‌هایی مفید است که جمع‌آوری مجموعه‌داده‌های بزرگ عملی یا مقرر نبهر صرفه نیست.

در یادگیری با نمونه‌های کم، مدل‌ها در مرحله استنتاج تنها تعداد کمی نمونه برچسب‌خورده (که اغلب به آن مجموعه پشتیبان^۱ گفته می‌شود) برای هر کلاس جدید دریافت می‌کنند. این تعداد محدود از نمونه‌ها به مدل امکان می‌دهد تا به سرعت خود را تطبیق داده و برای کلاس‌های نادیده شده پیش‌بینی انجام دهد. یک تنظیم رایج در یادگیری با نمونه‌های کم شامل الگوی K-shot N-way است که در آن 'N' نشان‌دهنده تعداد کلاس‌های جدید و 'K' نشان‌دهنده تعداد نمونه‌های برچسب‌خورده موجود برای هر کلاس است. برای مثال، در یک سناریوی 5-way 1-shot، 5 کلاس جدید وجود دارد که هر کدام تنها یک نمونه برچسب‌خورده دارند.

مدل‌های یادگیری با نمونه‌های کم معمولاً از تکییک‌های فرا-یادگیری^۲ استفاده می‌کنند که به آن "یادگیری برای یادگیری"^۳ نیز گفته می‌شود. در این چارچوب، مدل در طیف وسیعی از وظایف آموزش می‌بیند تا یک استراتژی برای تطبیق سریع با وظایف جدید با حداقل داده‌ها را یاد بگیرد. در زمان استنتاج، مدل با استفاده از همان تعداد اندک نمونه‌ها، پارامترهای خود را به‌طور مؤثری تنظیم می‌کند و می‌تواند دسته‌بندی‌های جدید را شناسایی و طبقه‌بندی کند.

از کاربردهای یادگیری با نمونه‌های کم می‌توان به موارد زیر اشاره کرد:

- یادگیری با نمونه‌های کم به مدل‌ها امکان می‌دهد تصاویر را با استفاده از تنها چند نمونه برچسب‌خورده در دسته‌های جدید طبقه‌بندی کنند که این موضوع به ویژه در حوزه‌هایی مانند تصویربرداری پزشکی که برچسب‌گذاری داده‌ها بسیار زمان بر است، اهمیت دارد.
- در حوزه پردازش زبان طبیعی، یادگیری با نمونه‌های کم می‌تواند در وظایفی مانند طبقه‌بندی متن و تحلیل احساسات به کار رود و به مدل‌ها کمک کند تا با حداقل داده متنی، موضوعات جدید را پردازش و درک کنند.

¹support set

²meta Learning

³Learning to Learn

- ربات‌ها می‌توانند با استفاده از یادگیری با نمونه‌های کم وظایف جدید در زمینه دستکاری اشیاء یا تطبیق با محیط‌های تازه را یاد بگیرند، که این موضوع نیاز به آموزش مجدد گستردۀ را کاهش داده و امکان استقرار سریع در محیط‌های پویا را فراهم می‌آورد.

یکی از چالش‌های اصلی در یادگیری با نمونه‌های کم این است که اطمینان حاصل شود مدل‌ها خود را با داده‌های محدود به خوبی تعمیم می‌دهند و دچار بیش‌برازش^۱ نمی‌شوند. پژوهشگران در حال بررسی روش‌های مختلفی از جمله افزایش داده^۲، یادگیری انتقالی^۳ و الگوریتم‌های پیشرفته فرا-یادگیری^۴ برای بهبود عملکرد مدل‌های یادگیری با نمونه‌های کم هستند.

برای فهم شفاف تر روش یادگیری درون متنی میتوان از یک چهارچوب ریاضیاتی [۲۰] کمک گرفت. در این چارچوب ریاضیاتی می‌توان یادگیری درون متنی را به عنوان یک استنتاج بیزی ضمنی^۵ تفسیر کرد. در چارچوب آن‌ها، در طی پیش‌آموزش^۶، مدل‌های زبانی بزرگ با استنتاج مفاهیم پنهان در اعلان، که روابط معنایی و نحوی مختلفی را در متن در بر می‌گیرد، یاد می‌گیرند که توکن‌های بعدی را پیش‌بینی کنند. در زمان استنتاج^۷، زمانی که مدلی با یک اعلان شامل مثال‌های ورودی-خروجی مواجه می‌شود، یک مفهوم پنهان مشترک بین این مثال‌ها را شناسایی می‌کند. این فرآیند شناسایی با استنتاج بیزی هم راست است، جایی که مدل بر اساس داده‌های مشاهده شده، باورهای خود را به روزرسانی می‌کند تا پیش‌بینی انجام دهد.

$$P(\text{مفهوم} | \text{اعلان}) = \int P(\text{مفهوم} | \text{اعلان}, \text{خروجی}) P(\text{خروجی}) \quad (1-2)$$

همانطور که در معادله ۱-۲ مشاهده می‌شود، از منظر بیزی، یادگیری درون متنی با پیدا کردن احتمال خروجی به شرط اعلان برابر است که انجام استنتاج توسط مدل برای یافتن یک مفهوم پنهان را در بر می‌گیرد که این مفهوم با وظیفه مورد نظر همخوانی دارد. با دریافت یک اعلان، مدل توزیع پسین^۸ را بر روی مفاهیم پنهان ممکن استنتاج می‌کند و مفهومی را انتخاب می‌کند که به بهترین نحو مثال‌های ارائه شده را توضیح می‌دهد. این فرآیند مشابه به روزرسانی بیزی است که در آن باورهای قبلی در پرتو شواهد جدید تنظیم می‌شوند تا پیش‌بینی‌های آگاهانه‌تری انجام گیرد.

¹overfitting

²data augmentation

³transfer learning

⁴meta Learning

⁵implicit Bayesian inference

⁶pre-training

⁷inference time

⁸posterior

۲-۴-۲ روش زنجیره تفکر

روش زنجیره تفکر^۱ [۱] تکنیکی است که با هدف تقویت توانایی استدلال مدل‌های زبانی بزرگ طراحی شده و آن‌ها را راهنمایی می‌کند تا هنگام حل مسائل پیچیده، گام‌های میانی استدلالی تولید کنند. این رویکرد مدل‌ها را تشویق می‌کند تا وظایف را به بخش‌های متوالی و مرحله‌به‌مرحله تقسیم کرده و در نتیجه به خروجی‌هایی دقیق‌تر و قابل تفسیرتر برست.

این روش که توسط پژوهشگران گوگل معرفی شد، شامل ارائه نمونه‌هایی به مدل‌ها است که هم شامل مسئله و هم شامل راه حل گام‌به‌گام و دقیق هستند. به عنوان مثال، وقتی یک مسئله کلامی ریاضی به مدل داده می‌شود، مدل با استفاده از زنجیره تفکر تشویق می‌شود که محاسبات و مراحل منطقی منتهی به پاسخ نهایی را به صورت شفاف بیان کند. این روش نشان داده که عملکرد مدل‌ها را در وظایف نیازمند به محاسبات ریاضی، استدلال مبتنی بر عقل سليم و استدلال نمادین به شکل چشمگیری بهبود می‌دهد. به طور خاص، یک مدل با ۵۴۰ میلیارد پارامتر با استفاده از CoT موفق شد به دقتی فراتر از حد استاندارد در دیتابست GSM8K برای مسائل ریاضی دست پیدا کند و حتی بهتر از نسخه‌های تنظیم شده ۳-GPT عمل کند.

اثربخشی روش زنجیره تفکر بهویژه در مدل‌های بزرگ‌تر بیشتر مشهود است. مدل‌هایی با بیش از ۱۰۰ میلیارد پارامتر در مواجهه با زنجیره تفکر توانایی‌های نوظهوری در زمینه استدلال چند مرحله‌ای از خود نشان می‌دهند و می‌توانند مسائل چند گامی را به شکل موثرتری حل کنند. این تکنیک نه تنها دقت را افزایش می‌دهد، بلکه شفافیت فرآیند استدلال مدل را نیز بهبود می‌بخشد؛ چرا که هر گام به صورت صریح ارائه می‌شود.

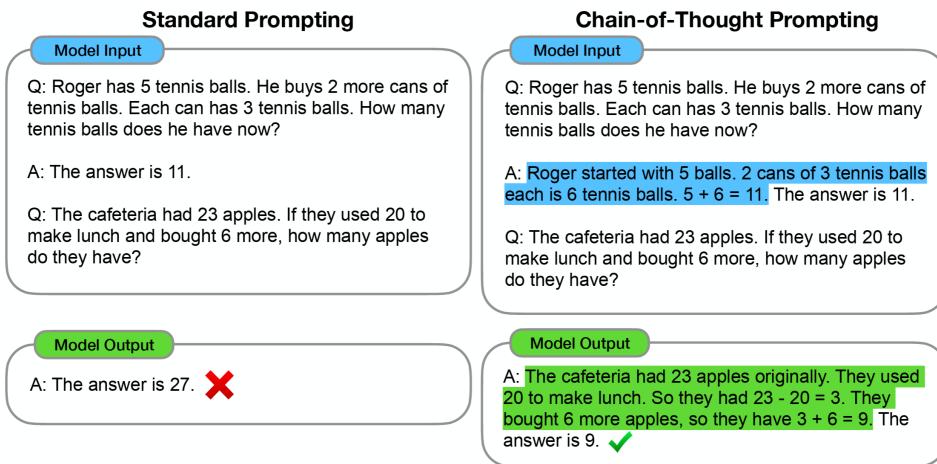
در شکل ۳-۲ یک نمونه از تولید جواب با استفاده از روش زنجیره تفکر آورده شده است. در این مثال به عنوان ورودی یک سوال و جواب نمونه به همراه راه حل قدم به قدم مسئله نیز آورده شده است تا مدل به سمت تولید راه حل سوق داده شود.

۳-۴-۲ روش استدلال بدون دیدن نمونه آموزشی

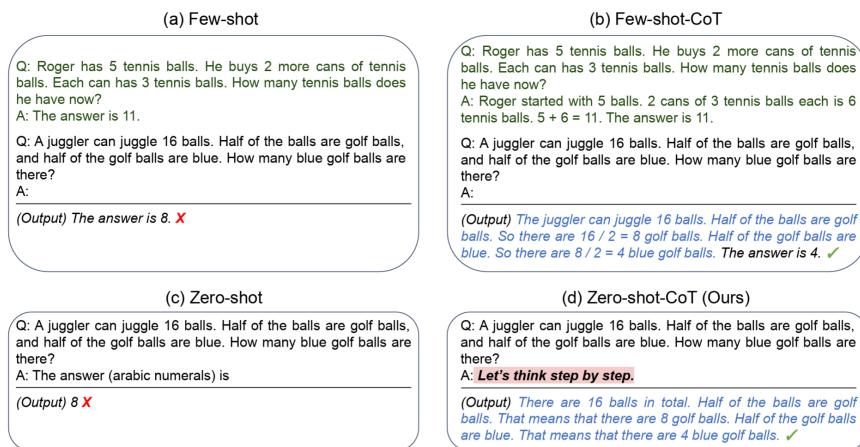
در مقاله‌ی استدلال بدون دیدن نمونه آموزشی^۲ [۲۱]، پتانسیل مدل‌های بزرگ زبانی را برای انجام استدلال از طریق تغییرات ساده در اعلان‌ها و بدون افزودن نمونه‌ای از مثال‌های حل شده، بررسی شده است. با اضافه کردن عبارت «بایاید مرحله به مرحله فکر کنیم» به انتهای یک اعلان، مدل‌هایی مانند ۳-GPT و PaLM [۲۲] به‌طور قابل توجهی عملکرد بهتری در دیتابست‌های مختلف استدلالی ارائه می‌دهند. این روش بر توانایی ذاتی مدل‌ها در پردازش و تولید متن شبیه به انسان تکیه دارد و آن‌ها را تشویق می‌کند تا

¹Chain-of-Thought

²Large Language Models are Zero-Shot Reasoners



شکل ۲ - ۳ مثالی از روش زنجیره تفکر برای حل یک سوال از دیتاست GSM8K



شکل ۴ - ۲ مقایسه روش استدلال بدون دیدن نمونه آموزشی و روش زنجیره تفکر

دباهات منطقی از تفکرات را که منجر به پاسخ نهایی می‌شود، تولید کنند. سادگی و کارآمدی این تکنیک، توانایی‌های استدلالی نهفته در مدل‌های بزرگ زبانی را آشکار می‌کند که می‌توان آن‌ها را بدون نیاز به آموزش وسیع و خاص برای هر وظیفه فعال کرد. این یافته نشان می‌دهد که با مهندسی اعلان مناسب، این مدل‌ها می‌توانند طیف وسیعی از وظایف را به طور مؤثرتری انجام دهند و نیاز به دیتاست‌های بزرگ و برچسب‌گذاری شده و همچنین فرایندهای زمانبر تنظیم دقیق مدل‌ها را کاهش دهند. در شکل ۴ - ۲ نمونه از حل مسئله دیتاست GSM8K آورده شده است و جواب این روش با روش زنجیره تفکر مقایسه شده است.

۴-۴-۲ روش برنامه تفکر

روش برنامه تفکر^۱ [۲] یک روش پیشرفته در مهندسی اعلان است که برای بهبود توانایی‌های استدلال عددی در مدل‌های زبانی بزرگ طراحی شده است. برخلاف تکنیک زنجیره تفکر که در آن مدل هم استدلال و هم محاسبات را در قالب متن تولید می‌کرد، در برنامه تفکر این دو فرآیند از یکدیگر جدا می‌شوند. در این روش، مدل استدلال خود را به صورت کد قابل اجرایی (معمولًاً با زبان‌هایی مانند پایتون) بیان می‌کند و این کد توسط یک مفسر خارجی اجرا می‌شود تا پاسخ نهایی به دست آید. این جداسازی باعث می‌شود محاسبه و استدلال از یکدیگر تفکیک شوند.

از مزایای روش برنامه تفکر می‌توان به موارد زیر اشاره کرد:

- با واگذاری محاسبات به یک مفسر خارجی، احتمال بروز خطاهای عددی که ممکن است در صورت انجام محاسبه و استدلال توسط خود مدل رخ دهد، کاهش می‌یابد و دقت بالاتر می‌رود.
- مطالعات تجربی نشان داده‌اند که برنامه تفکر می‌تواند عملکرد مدل را در وظایف عددی پیچیده به طور قابل توجهی افزایش دهد. برای مثال، آزمایش‌ها نشان داده‌اند که PoT نتایجی هم‌سطح با بهترین روش‌های موجود در دیتاست‌های مسائل ریاضی و نزدیک به بهترین عملکردها در دیتاست‌های مالی داشته است.
- نمایش مراحل استدلال به صورت کد، شفافیت فرآیند حل مسئله را افزایش می‌دهد و بررسی و درک هر مرحله را ساده‌تر می‌کند.

در شکل ۵-۲ حل مسئله دنباله فیبونانچی و پیداکردن ۵۰ امین عضو این دنباله، با استفاده از روش زنجیره تفکر و روش برنامه تفکر بررسی شده است. به دلیل طولانی بودن مراحل حل مسئله، روش زنجیره تفکر نتوانسته جواب درست را پیدا کند ولی روش برنامه تفکر ابتدا یک کد برای حل این مسئله ارائه داده است و پس از اجرای این کد با مفسر، به جواب صحیح رسیده است.

۵-۴-۲ روش بهینه سازی با اعلان

همانطور که گفته شد مدل‌های زبانی بزرگ تا به اینجا برای وظایفی مانند تولید متن، ترجمه و خلاصه‌سازی به کار گرفته می‌شدند. با این حال، تحقیقات اخیر ظرفیت این مدل‌ها را به عنوان بهینه‌سازها مورد بررسی قرار داده‌اند و از قابلیت‌های استدلالی آن‌ها برای حل مسائل پیچیده بهینه‌سازی بهره برده‌اند.

¹Program-of-Thought

Question: In Fibonacci sequence, it follows the rule that each number is equal to the sum of the preceding two numbers. Assuming the first two numbers are 0 and 1, what is the 50th number in Fibonacci sequence?

The first number is 0, the second number is 1, therefore, the third number is $0+1=1$. The fourth number is $1+1=2$. The fifth number is $1+2=3$. The sixth number is $2+3=5$. The seventh number is $3+5=8$. The eighth number is $5+8=13$.
..... (Skip 1000 tokens)
The 50th number is 32,432,268,459.

CoT

```
length_of_fibonacci_sequence = 50
fibonacci_sequence = np.zeros(length_of_)
fibonacci_sequence[0] = 0
fibonacci_sequence[1] = 1
for i in range(3, length_of_fibonacci_sequence):
    fibonacci_sequence[i] = fibonacci_sequence[i-1] +
    fibonacci_sequence[i-2]
ans = fibonacci_sequence[-1]
```

PoT

 32,432,268,459



 python

 12,586,269,025



Question: Ketty saves 20000 dollars to the bank. After three years, the sum with compound interest rate is 1000 dollars more than the sum with simple interest rate. What is the interest rate of the bank?

Assuming the interest rate is x . The sum after two years with simple interest rate is $20000 + x * 20000 * 3 = 20000 + 60000x$. The sum after two years with compound interest rate is $20000 * (1 + x)^3 = 200000 + 60000 * x + 60000x^2 + 20000x^3$. The difference can be written as $60000x^2 + 20000x^3 = 1000$. In order to solve x , we can use the quadratic formula. $x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$, ..., $x = \frac{(-20000 \pm 6160)}{120000}$, $x = -0.051333$.

CoT

```
interest_rate = Symbol('x')
sum_in_two_years_with_simple_interest = 20000 +
interest_rate * 20000 * 3
sum_in_two_years_with_compound_interest = 20000 * (1 +
interest_rate)**3
# Since compound interest is 1000 more than simple interest.
ans = solve(sum_after_in_yeras_with_compound_interest -
sum_after_two_years_in_compound_interest - 1000,
interest_rate)
```

PoT

 -0.051333



 python



 x = 0.24814



شکل ۲-۵ مقایسه روش برنامه تفکر و روش زنجیره تفکر

یکی از رویکردهای برجسته در این زمینه، روش بهینه سازی با اعلان^۱ [۲۳] است. این روش از مدل‌های زبانی بزرگ برای تولید راه حل‌های مسائل بهینه سازی بر اساس اعلان‌های زبان طبیعی استفاده می‌کند. این فرآیند شامل تکرارهای پی‌درپی است که در آن، اعلان جدید با راه حل‌ها و ارزیابی‌های قبلی طراحی می‌شود و در تکرارهای بعدی، راه حل‌های بهتری پیشنهاد می‌دهد. این روش در کاربردهای مختلفی همچون رگرسیون خطی^۲، مسئله فروشنده دوره‌گرد^۳ و حتی بهینه سازی اعلان برای خود مدل‌های زبانی بزرگ مؤثر واقع شده است. جالب توجه اینکه این روش توانسته تا ۸٪ بهبود عملکرد در دیتاست GSM8K و تا ۵۰٪ بهبود در وظایف دشوار^۴ داشته باشد و حتی از اعلان‌های طراحی شده توسط انسان نیز پیشی بگیرد.

در روش بهینه سازی با اعلان، مسئله بهینه سازی با زبان طبیعی توصیف می‌شود تا مدل بتواند بافت و هدف مسئله را درک کند. همانطور که در شکل ۶-۲ در هر مرحله از بهینه سازی، مدل با توجه به اعلان شامل راه حل‌ها و ارزیابی‌های قبلی، راه حل‌های جدیدی تولید می‌کند. این راه حل‌ها ارزیابی شده و سپس برای تکرار بعدی در اعلان گنجانده می‌شوند و این چرخه بهبود مستمر ادامه پیدا می‌کند.

از مزایای استفاده از مدل‌های زبانی بزرگ به عنوان بهینه ساز می‌توان به موارد زیر اشاره کرد:

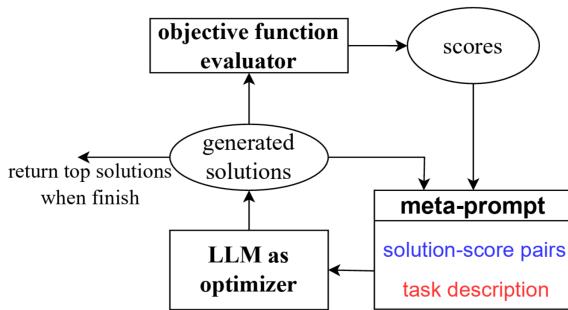
- مدل‌های زبانی بزرگ به دلیل توانایی درک و تولید متى مشابه انسان، می‌توانند به انواع مختلف مسائل بهینه سازی که با زبان طبیعی توصیف می‌شوند، پاسخ دهنند.

¹ Optimization by PROMPTing

² Linear Regression

³ Traveling Salesman Problem

⁴ Big-Bench Hard



شکل ۲-۶ دیاگرام روش بهینه سازی با اعلان

- همچنین این مدل‌ها قادر به انجام بهینه‌سازی در مسائلی هستند که اطلاعات گرادیان در دسترس نیست یا به دست آوردن آن دشوار است و به همین دلیل رویکردی بدون نیاز به مشتق ارائه می‌دهند.
- از طرفی این مدل‌ها می‌توانند اعلان‌های خود را بهینه کنند و بدون نیاز به آموزش مجدد، عملکرد خود را در وظایف خاص بهبود دهند.

۶-۴-۲ روش برنامه‌ریزی و حل

روش برنامه‌ریزی و حل^۱ یک تکنیک پیشرفته در مهندسی اعلان است که بهبود عملکرد مدل‌های زبانی بزرگ را در حل مسائل پیچیده مورد هدف قرار می‌دهد. در این روش، مدل ابتدا یک برنامه‌ریزی^۲ انجام داده و سپس بر اساس آن، به حل مسئله^۳ می‌پردازد.

برخلاف روش‌های سنتی که مدل را مستقیماً درگیر حل مسئله می‌کنند، این روش باعث کاهش نرخ خطای شده و دقت پاسخ‌های مدل را افزایش می‌دهد. روش برنامه‌ریزی و حل بهویژه در مسائلی که نیاز به چندین مرحله استدلالی دارند، مانند حل مسائل ریاضی، تحلیل منطقی و برنامه‌ریزی وظایف، بسیار کارآمد است. این روش شامل دو مرحله اصلی است:

- برنامه‌ریزی^۴: مدل یک برنامه کلی برای حل مسئله ارائه می‌دهد، شامل مراحل موردنیاز برای رسیدن به پاسخ.
- حل مسئله^۵: مدل بر اساس برنامه تولیدشده، گام به گام راه حل را پیاده‌سازی کرده و پاسخ نهایی را استخراج می‌کند.

¹ Plan-and-Solve (PS)

² Planing

³ Solve

⁴ Plan Phase

⁵ Solve Phase

این تفکیک دو مرحله‌ای، عملکرد مدل را بهبود می‌بخشد زیرا ابتدا ساختار حل مسئله مشخص شده و سپس محاسبات انجام می‌شود. در ادامه یک مسئله و راه حل این روش برای آن مسئله را بررسی می‌کنیم.

مسئله: سن علی ۳ برابر سن برادرش است. ۴ سال پیش، مجموع سن آن‌ها ۲۰ سال بوده است. سن هر یک را مشخص کنید.

۱. اجرای مرحله برنامه‌ریزی:

- تعریف متغیرها: فرض کنیم سن برادر علی را X در نظر بگیریم.
- رابطه کنونی: سن علی برابر با $3X$ است.
- رابطه در گذشته: ۴ سال پیش، سن برادر علی برابر $4 - X$ و سن علی برابر $4 - 3X$ بوده است.

• معادله کلی:

$$(X - 4) + (3X - 4) = 20$$

• حل معادله و یافتن مقدار X

۲. اجرای مرحله حل:

$$X - 4 + 3X - 4 = 20$$

$$4X - 8 = 20$$

$$4X = 28$$

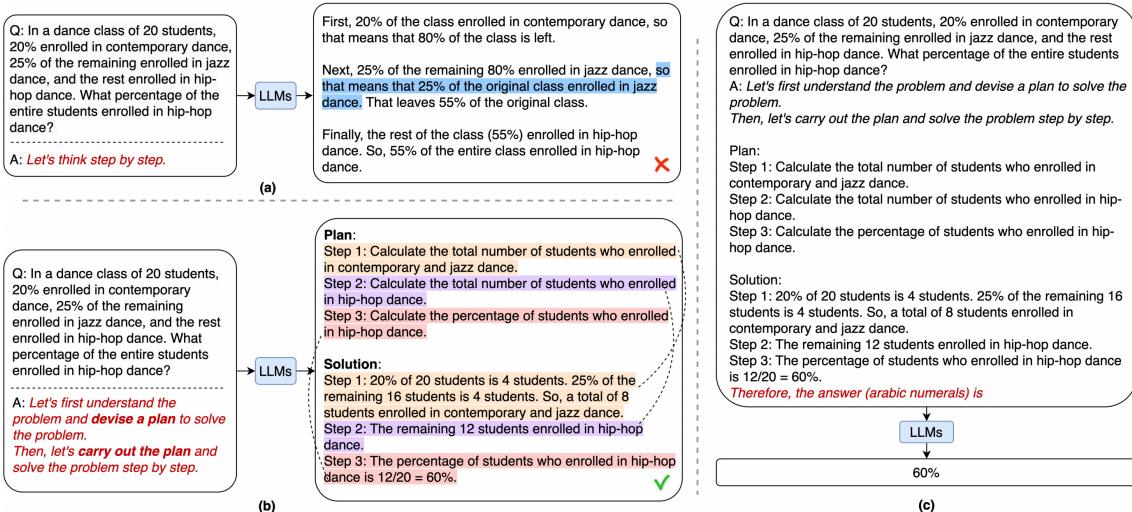
$$X = 7$$

در نتیجه :

• سن برادر علی : ۷ سال

• سن علی : $7 \times 3 = 21$ سال

با بررسی این مثال می‌توان متوجه افزایش دقت حل مسائل چند مرحله‌ای با استفاده از روش برنامه‌ریزی و حل شد و همچنین عملکرد برنامه ریزی و سپس عمل را مشاهده کرد که مشابه سیستم تفکر انسانی است. در شکل ۷-۲ یک مثال دیگر از این روش آورده شده است.



شکل ۷-۲ یک مثال از روش برنامه‌ریزی و حل

۵-۲ بهینه‌سازی خودکار اعلان‌ها

بهینه‌سازی خودکار اعلان‌ها یک رویکرد نوین در مهندسی اعلان است که با بهره‌گیری از الگوریتم‌های جستجو و بهینه‌سازی، به صورت سیستماتیک بهترین ورودی‌های متنی را برای مدل‌های زبانی شناسایی و تولید می‌کند. در این روش به جای تکیه بر شهود انسانی و روش‌های آزمون و خطأ، از تکنیک‌هایی مانند الگوریتم‌های تکاملی، جستجوی تصادفی و یادگیری تقویتی استفاده می‌شود. این الگوریتم‌ها فضای اعلان‌ها را مورد بررسی قرار داده و بر اساس معیارهای مشخصی مانند دقیق، انسجام و ارتباط معنایی، به انتخاب بهینه‌ترین اعلان‌ها می‌پردازند.

یکی از مزایای مهم بهینه‌سازی خودکار اعلان‌ها این است که در مقیاس‌های بزرگ و در زمان‌های کوتاه، قادر به دستیابی به نتایج بهینه می‌باشد. بهینه‌سازی خودکار پرامت‌ها با ارائه چارچوبی سیستماتیک، امکان ارزیابی سریع و انتخاب اعلان‌های بهینه را فراهم می‌آورد. این دو رویکرد می‌توانند مکمل یکدیگر عمل کرده و در کاربردهایی که نیاز به سرعت و دقیقی بالا دارند، مانند پردازش دسته‌جمعی داده‌های متنی یا تنظیم خودکار اعلان‌ها برای وظایف متنوع، عملکرد بهتری ارائه دهند.

۵-۱ روش زنجیره تفکر خودکار

از اولین جرقه‌های خودکارسازی مهندسی اعلان، می‌توان به خودکارسازی تولید زنجیره‌های استدلالی در روش زنجیره تفکر اشاره کرد. روش زنجیره تفکر خودکار^۱ [۲۴] به طور خودکار زنجیره‌هایی از استدلال‌ها و سوالات را برای ساخت نسخه‌ها ایجاد می‌کند. این روش شامل دو مرحله اصلی است. مرحله اول خوشبندی

^۱ AUTOMATIC CHAIN-OF-THOUGHT PROMPTING (Auto-CoT)

سوالات و تقسیم سوالات یک مجموعه داده به چندین خوش و مرحله دوم نمونه‌گیری از نسخهای انتخاب یک سوال نماینده از هر خوش و تولید زنجیره استدلال برای آن است. روند کلی این روش در شکل ۸-۲ نشان داده شده است.

در مرحله اول، از آنجایی که خوشبندی متنی بر تنوع می‌تواند از گمراهی ناشی از شباهت جلوگیری کند، این روش تحلیل خوشبندی را برای مجموعه‌ای از سوالات Q انجام می‌دهد. بدین صورت که ابتدا برای هر سوال در Q ، یک بردار نمایشی با استفاده از Sentence-BERT [۲۵] محاسبه می‌شود و سپس این بردارهای متنی میانگین‌گیری شده و یک بردار با اندازه ثابت برای هر سوال ایجاد می‌شود. پس از آن، با استفاده از الگوریتم خوشبندی k-means، سوالات به k خوش تقسیم می‌شوند. در نهایت برای هر خوش i ، سوالات آن خوش را به صورت یک لیست مرتب شده $[q_1^{(i)}, q_2^{(i)}, \dots, q_n^{(i)}]$ بر اساس فاصله از مرکز خوش به ترتیب صعودی مرتب می‌شوند.

در مرحله دوم، برای سوالات نمونه‌گیری شده، زنجیرهای استدلال تولید می‌شوند و نسخهایی که با معیارهای انتخاب مطابقت دارند استخراج می‌شوند. به طور دقیق‌تر، برای هر خوش i ، یک نسخه به صورت $d^{(i)}$ (ترکیبی از سوال، استدلال و پاسخ) ساخته می‌شود. برای خوش i ، سوالات مرتب شده در لیست $q^{(i)} = [q_1^{(i)}, q_2^{(i)}, \dots]$ تا زمانی که معیارهای انتخاب برآورده شوند، بررسی می‌شوند. به عبارت دیگر، سوالی که به مرکز خوش نزدیک‌تر است، زودتر بررسی می‌شود. فرض کنید سوال $q_j^{(i)}$ که j -امین سوال نزدیک به مرکز خوش i است، در حال بررسی باشد. یک ورودی به شکل زیر ساخته می‌شود:

$$[Q: q_j^{(i)} \ A: [P]]$$

سپس یک نسخه کاندید برای خوش i به صورت زیر ساخته می‌شود:

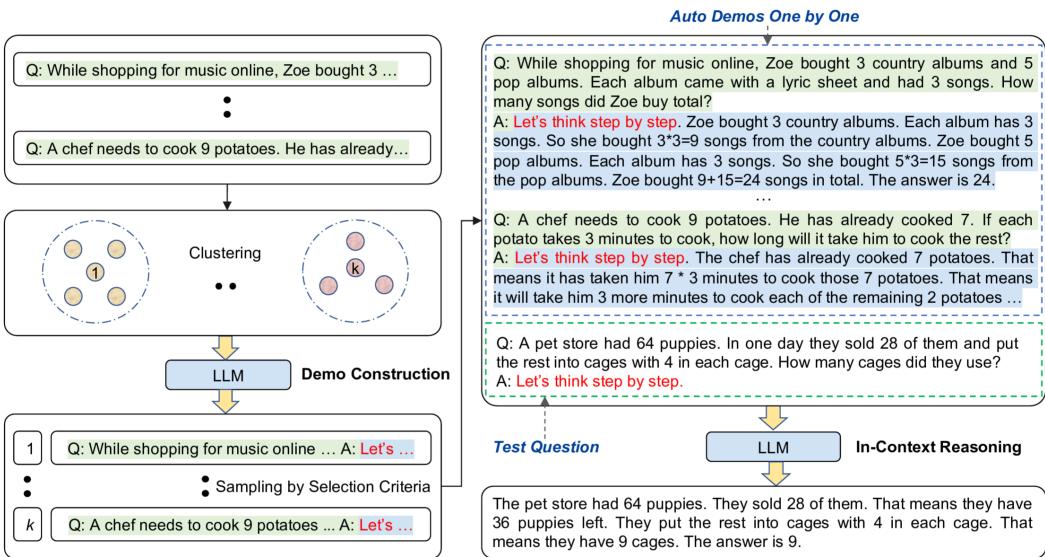
$$[Q: q_j^{(i)} \ A: r_j^{(i)} \ a_j^{(i)}]$$

مشابه معیارهای مورد استفاده در نسخه‌های دستی روش زنجیره تفکر [۱]، معیار انتخاب این الگوریتم نیز بر قوانین ساده‌ای تکیه دارد تا سوالات و استدلال‌های ساده‌تر را انتخاب کند: نسخه $d_j^{(i)}$ به عنوان $d^{(i)}$ انتخاب می‌شود اگر سوال $q_j^{(i)}$ دارای کمتر از ۶ توکن و استدلال $r_j^{(i)}$ دارای کمتر از ۵ گام استدلالی باشد.

۲-۵-۲ روش مهندس اعلان خودکار

روش مهندس اعلان خودکار ^۱ [۲۶] از خود مدل‌های زبانی بزرگ برای خودکارسازی فرآیند ایجاد و بهینه‌سازی اعلان‌ها استفاده می‌کند. در این روش، یک مدل زبانی، مجموعه‌ای از اعلان‌های مختلف تولید

^۱Auto Prompt Engineer



شکل ۸-۲ دیاگرام روش auto-CoT

می‌کند و سپس این اعلان‌ها بر اساس کیفیت پاسخ‌هایی که از یک مدل دیگر دریافت می‌شود، ارزیابی می‌شوند. این فرآیند به صورت تکراری ادامه می‌یابد تا مؤثرترین اعلان‌ها شناسایی شوند. نتایج در ۲۴ وظیفه مختلف در حوزه پردازش زبان طبیعی نشان داده که اعلان‌های تولید شده توسط این روش در اغلب موارد عملکرد بهتری نسبت به روش‌های قبلی داشته‌اند و در ۱۹ مورد از ۲۴ وظیفه، کیفیتی معادل یا نزدیک به اعلان‌های طراحی شده توسط انسان ارائه کرده‌اند.

تأثیرات این روش قابل توجه است؛ این روش نیاز به مهندسان اعلان انسانی را کاهش می‌دهد، فرآیند توسعه را سریع‌تر می‌کند و سازگاری مدل‌های زبانی بزرگ با وظایف مختلف را افزایش می‌دهد. این پیشرفت نشان‌دهنده ظرفیت مدل‌های بزرگ زبانی برای نه تنها انجام وظایف، بلکه بهینه‌سازی دستورالعمل‌های خودشان نیز هست و گامی مهم به سوی سامانه‌های هوشمند خودمختارتر به شمار می‌رود.

۳-۵-۲ روش مولد اعلان

روش مولد اعلان^۱ [۴] یک سیستم خوددارجاعی^۲ و تکاملی برای بهبود خودکار اعلان‌های مورد استفاده در مدل‌های زبانی بزرگ است. این روش با الهام از الگوریتم‌های تکاملی و فرآیندهای خودبهبوددهی^۳، به جای اتکا بر اعلان‌های دستی و مهندسی شده، به طور خودکار اعلان‌هایی را تولید و اصلاح می‌کند که می‌توانند عملکرد مدل را در حل مسائل مختلف بهبود بخشنند. ویژگی کلیدی این روش این است که نه تنها

¹Promptbreeder

²self-referential

³self-improvement

اعلان‌های وظیفه^۱ را تکامل می‌دهد، بلکه اعلان‌های تغییر^۲ را نیز که برای تغییر اعلان‌های مورد جستجو استفاده می‌شوند، بهبود می‌بخشد.

این الگوریتم از یک فرآیند تکاملی مبتنی بر جمعیت^۳ استفاده می‌کند. همانطور که در شکل ۹-۲ نشان داده شده است، در ابتدا یک مجموعه از اعلان‌های اولیه به همراه دستورات تغییر تولید می‌شود. سپس، در هر نسل از فرآیند تکامل:

۱. ارزیابی سازگاری^۴: اعلان‌ها بر اساس عملکردشان در پاسخ‌دهی به مجموعه‌ای از سؤالات آموزشی ارزیابی می‌شوند.

۲. انتخاب^۵: دو اعلان به صورت تصادفی انتخاب شده و مقایسه می‌شوند؛ اعلانی که عملکرد بهتری داشته باشد، انتخاب می‌شود.

۳. اعمال تغییرات^۶: اعلان انتخاب شده با استفاده از عملگرهای تکاملی تغییر داده می‌شود. این تغییرات شامل تولید نسخه‌های جدید از اعلان، اصلاح بر اساس الگوهای موفق، و یا حتی بهبود خود دستور تغییر است.

۴. جایگزینی^۷: نسخه‌ی بهبودیافته جایگزین اعلان با عملکرد ضعیفتر شده و این فرآیند در نسل‌های بعدی تکرار می‌شود.

عملگرهای تکاملی شامل جهش مستقیم^۸، جهش مبتنی بر توزیع^۹، ابرجهش^{۱۰}، جهش لامارکین^{۱۱}، و ترکیب اعلان‌ها^{۱۲} هستند که هر یک روش‌های متفاوتی را برای اصلاح و بهبود اعلان‌ها ارائه می‌دهند. با ادامه این فرآیند در طی چندین نسل، اعلان‌ها به تدریج بهینه شده و به عملکرد بهتری در مدل‌های زبانی منجر می‌شوند.

روش مولد اعلان در آزمایش‌های مختلف نشان داده است که می‌تواند از روش‌های متداول مهندسی اعلان مانند CoT و PS عملکرد بهتری داشته باشد. همچنین، این سیستم در حوزه‌های مختلف مانند حل

¹Task Prompts

²Mutation Prompts

³population base

⁴Fitness Evaluation

⁵Selection

⁶Mutation

⁷Replacement

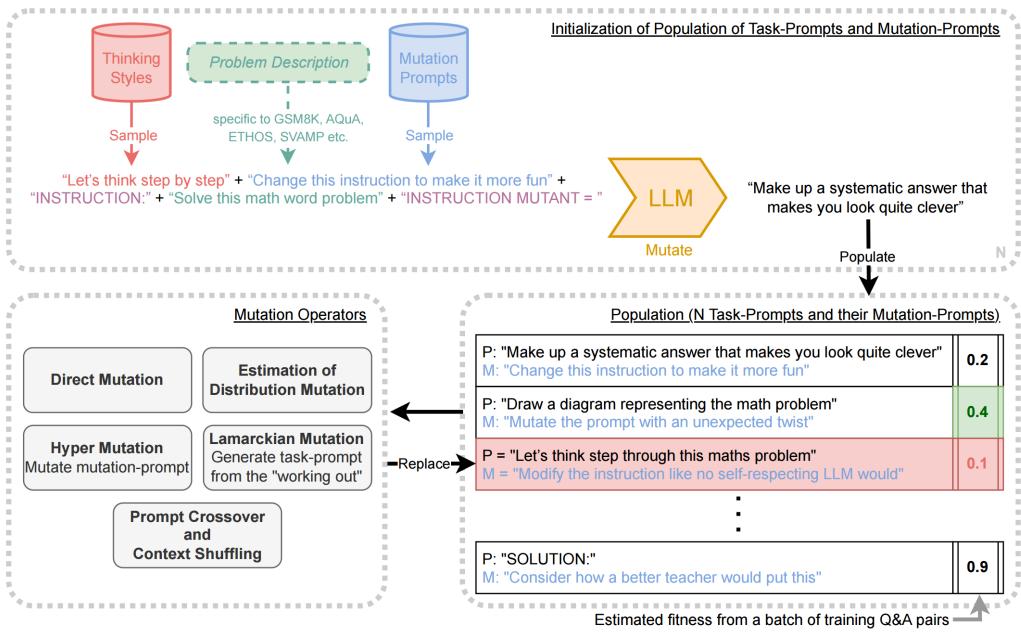
⁸Direct Mutation

⁹Estimation of Distribution (EDA) Mutation

¹⁰Hypermutation

¹¹Lamarckian Mutation

¹²Prompt Crossover and Context Shuffling



شکل ۹-۲ دیاگرام روش مولد اعلان

مسائل ریاضی، استدلال عمومی^۱، و کلاس‌بندی گفتار نفرت‌آمیز^۲ بهبود چشمگیری ایجاد کرده است. مهم‌ترین مزیت آن این است که به بروزرسانی پارامترهای مدل نیازی ندارد و تنها از طریق اصلاح اعلان‌ها، کارایی مدل را افزایش می‌دهد.

این روش نشان‌دهنده قدرت بهینه‌سازی خودکار اعلان‌ها و حرکت بهسوی سیستم‌های هوش مصنوعی خودبهبوددهنده و خودارجاعی است که می‌توانند با حداقل مداخله انسانی، عملکرد خود را در طی زمان بهبود بخشنند.

۶-۲ جمع‌بندی مباحث ارائه شده

این فصل با هدف بررسی پیشینه تحقیق و تبیین مفاهیم بنیادی مرتبط با موضوع پژوهش تدوین گردید. در این فصل، تلاش شد تا ضمن ارائه چارچوب نظری جامع، زمینه مناسبی برای درک بهتر مسأله تحقیق و مسیر پژوهش فراهم شود. با توجه به گستره و عمق موضوعات مطرح شده، می‌توان این فصل را به عنوان شالوده‌ای برای تحلیل‌ها و مطالعات تخصصی‌تر در فصول بعدی قلمداد نمود.

در این فصل، نخست به معرفی و تحلیل مدل‌های زبان بزرگ پرداخته شد و روند تکاملی این مدل‌ها از آغاز تا وضعیت کنونی آن‌ها مورد بررسی قرار گرفت. در ادامه، به معماری‌های رایج و کاربردهای متنوع این مدل‌ها اشاره شد و اهمیت آن‌ها در پیشبرد حوزه‌های مختلف پردازش زبان طبیعی و سامانه‌های مبتنی بر

¹commonsense reasoning

²hate speech classification

هوش مصنوعی تبیین گردید.

سپس، به موضوع مهندسی اعلان و نقش آن در بهره‌برداری مؤثر از مدل‌های زبان بزرگ پرداخته شد. در این بخش، ابتدا روش‌های سنتی مهندسی اعلان، شامل استفاده از یادگیری درون متنی، روش زنجیره تفکر، استدلال بدون نمونه آموزشی، برنامه تفکر، بهینه سازی با اعلان و روش برنامه‌ریزی و حل به تفصیل بررسی شدند. در ادامه، به معرفی و تحلیل رویکردهای نوین در بهینه‌سازی خودکار اعلان‌ها پرداخته شد. از جمله این روش‌ها می‌توان به زنجیره تفکر خودکار، مهندسی اعلان خودکار و روش مولد اعلان اشاره نمود که هریک با هدف افزایش دقت و توان استدلال مدل‌های زبانی توسعه یافته‌اند.

بر مبنای مباحث ارائه شده، آشکار شد که مهندسی اعلان و بهینه‌سازی آن از مهم‌ترین مؤلفه‌ها در افزایش کارایی مدل‌های زبان بزرگ به شمار می‌رود. این امر اهمیت طراحی الگوریتم‌های نوین و کارآمد برای بهبود ساختار و محتوای اعلان‌ها را دوچندان می‌سازد.

در فصل آتی، به تشریح کامل الگوریتم پیشنهادی پرداخته خواهد شد. این الگوریتم که مولد اعلان ساده نام دارد، یکی از رویکردهای نوین در زمینه بهینه‌سازی و تکامل خودکار اعلان‌ها محسوب می‌شود و مبتنی بر اصول الگوریتم‌های تکاملی و بهبود مستمر اعلان‌ها از طریق تولید، ارزیابی و گرینش نسل‌های مختلف اعلان‌ها طراحی شده است. در فصل آینده، بهطور دقیق به ساختار، مراحل اجرایی و مزایای این الگوریتم در مقایسه با سایر روش‌ها پرداخته خواهد شد.

فصل سوم:

روش‌های پیشنهادی

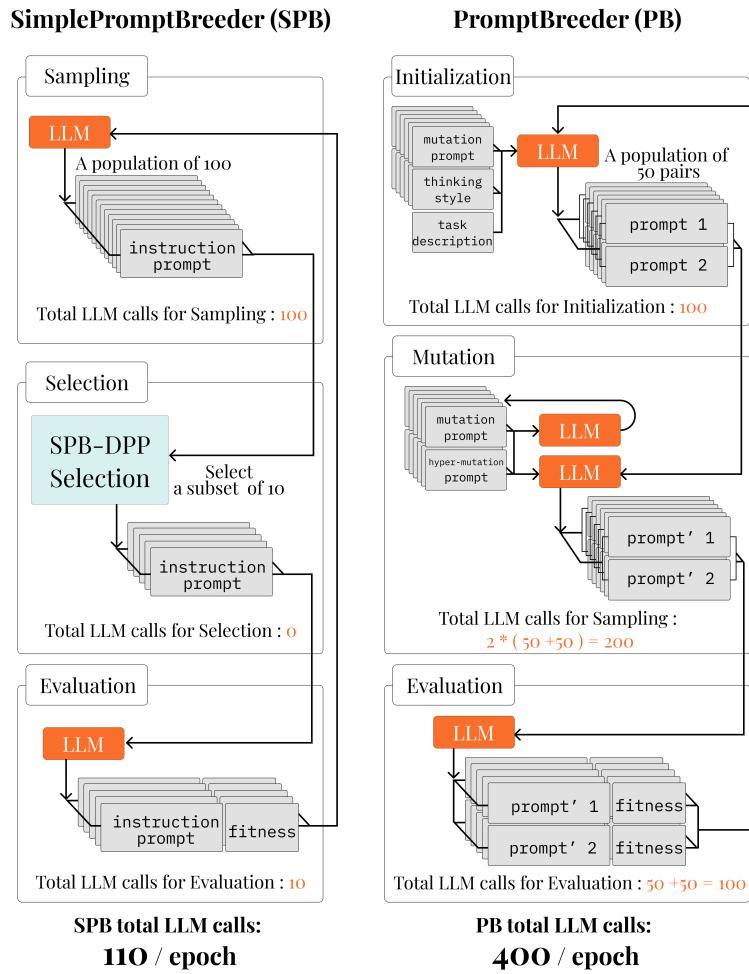
همانطور که گفته شد، عملکرد مدل‌های زبانی بزرگ بهشت به مهندسی اعلان وابسته است، چرا که دستورات به طور مستقیم بر تفسیر مدل از وظایف و همچنین نحوه تولید خروجی تاثیر می‌گذارند. بنابراین، دستورات نقش حیاتی در اثربخشی مدل‌های زبانی بزرگ ایفا می‌کنند، فرآیندی که معمولاً از طریق آنچه به عنوان مهندسی دستور شناخته می‌شود، بهینه‌سازی می‌گردد. استراتژی‌های پرکاربردی مانند زنجیره تفکر [۱] اثربخشی خود را در بهبود توانمندی‌های استدلالی مدل‌های زبانی با تجزیه مسائل پیچیده به گام‌های میانی نشان داده‌اند [۲۷]، [۲۸]، [۲۹]. با این حال، استراتژی‌های دستی طراحی شده توسط انسان، به دلیل وابستگی به شهود انسانی، غالباً برای وظایف خاص دامنه محور، بهینه نبوده و نمی‌توانند به طور کامل از ظرفیت مدل‌های پایه بهره‌برداری کنند. از سوی دیگر، با پیشرفت مداوم مدل‌های زبانی و تغییر قابلیت‌های آن‌ها، مؤثرترین دستورات نیز ممکن است دستخوش تغییر شوند. به طور کلی، مهندسی اعلان که به صورت دستی انجام می‌شود، فرآیندی زمان‌بر بوده و نتایجی کمتر از حد بهینه به همراه دارد. این محدودیت‌ها موجب شده است که تولید خودکار دستورات به عنوان یک حوزه تحقیقاتی مهم در هوش مصنوعی مطرح گردد [۳۱]، [۳۲]. همانطور که در فصل قبل دیدیم، الگوریتم‌های مختلفی برای بهینه‌سازی دستورات معرفی شده‌اند؛ از جمله بهینه‌سازی با اعلان [۲۳]، مولد اعلان [۴] و سایر روش‌ها [۳۳]. در این میان، روش مولد اعلان با استفاده از یک چارچوب تکاملی و خودارجاعی، از طریق تکرار فرآیند جهش و ارزیابی یک جمعیت از دستورات وظیفه، به بهینه‌سازی دستورات می‌پردازد. اگرچه نشان داده شده است که روش مود اعلان برای وظایف استدلالی و طبقه‌بندی مؤثر است، اما پیچیدگی طراحی آن موجب کاهش قابلیت تعمیم این روش می‌شود. افزون بر این، عدم تنوع کافی در دستورات تولید شده می‌تواند چالش‌زا باشد. وابستگی به چندین لایه بهینه‌سازی، سازوکارهای خودارجاعی و حتی به کارگیری تکنیک‌هایی برای فریب مدل زبانی [۱] ممکن است موجب عدم موفقیت این روش هنگام به کارگیری بر روی مدل‌های زبانی جدید شود.

در این مطالعه، جایگزینی ساده‌تر و شفاف‌تر برای مولد اعلان تحت عنوان مولد اعلان ساده^۱ پیشنهاد می‌شود که ضمن حفظ نقاط قوت اصلی، کاستی‌های آن را نیز برطرف می‌سازد. با ساده‌سازی چارچوب تکاملی بهینه‌سازی دستورات، روش پیشنهادی، ضمن کاهش پیچیدگی، قابلیت تفسیر و مقیاس‌پذیری را نیز بهبود می‌بخشد (همان‌طور که در شکل ۱-۳ نشان داده شده است). این رویکرد در تلاش است تا تولید خودکار دستورات را کاربردی‌تر و در دسترس‌تر سازد.

مولد اعلان ساده به صورت تکرارشونده دستورات را بررسی می‌کند و با بهره‌گیری مستقیم از یک مدل احتمالاتی به نام فرآیندهای نقطه‌ای دترمینانی، به بهینه‌سازی تنوع و کیفیت آن‌ها می‌پردازد. مدل‌های

^۱صفحه ۷ مقاله مولد اعلان : «توجه داشته باشید که ما به مدل زبانی 'دروغ' گفته‌ایم و بیان کرده‌ایم که ترتیب نزولی است.»

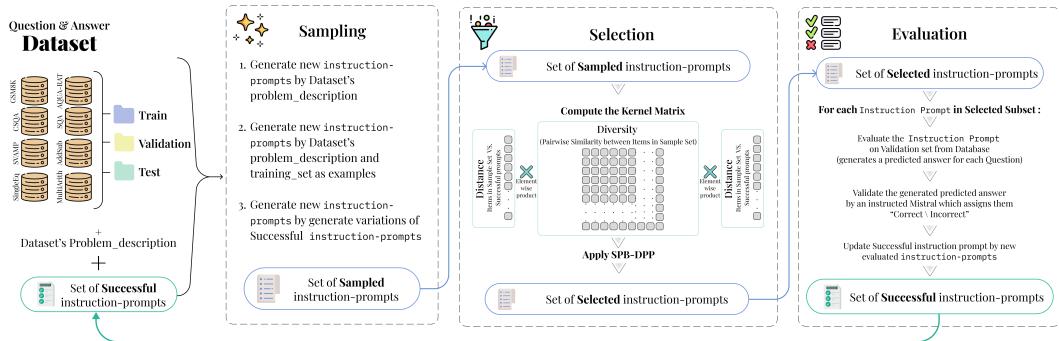
²SimplePromptBreeder



شکل ۱-۳ مقایسه پیجیدگی محاسباتی روش مولد اعلان و روش مولد اعلان ساده

احتمالی فرآیندهای نقطه‌ای دترمینانی، به گونه‌ای طراحی شده‌اند که راهکارهای بهینه‌ای برای مسئله انتخاب در فضای دستورات ارائه دهند به گونه‌ای که هم با کیفیت و هم متنوع باشند.

فرضیه تحقیق ما این است که استفاده از فرآیندهای نقطه‌ای دترمینانی می‌تواند تعداد دفعات فراخوانی مدل زبانی بزرگ را برای یافتن دستورات بهینه کاهش داده و در عین حال، سطح عملکرد رقابتی را حفظ نماید. سادگی و ظرفت ریاضی این رویکرد همچنین می‌تواند موجب افزایش استحکام آن در برابر انواع مدل‌های زبانی شود که این روش بر روی آن‌ها اعمال می‌گردد. از طرفی ساختار ریاضی این روش، آن را به جایگزینی عملی و قابل تفسیر برای روش‌های پیچیده تبدیل می‌کند.



شکل ۲-۳ دیاگرام روش مولد اعلان ساده

۲-۳ روش مولد اعلان ساده

برای استفاده از مدل‌های زبانی بزرگ با دستورالعمل^۱ مانند Mistral^۲، لازم است اعلان‌ها در قالب خاصی سازماندهی شوند تا حالت مکالمه‌ای^۳ شبیه‌سازی شود. این قالب بر پایه‌ی ساختار کلید-مقدار استوار است، که در آن هر کلید، نقش^۴ را مشخص می‌کند و مقدار متناظر، محتوا یا پیام مربوط به آن نقش^۵ را شامل می‌شود. سه نقش رایج در این قالب وجود دارد: سیستم، دستیار و کاربر. برای نقش سیستم، محتوا به عنوان دستورالعمل‌هایی برای مدل زبانی عمل می‌کند تا وظایف مشخصی را انجام داده و به اعلان‌های کاربر پاسخ دهد. این دستورالعمل‌ها با عنوان اعلان دستوری^۶ شناخته می‌شوند و تأثیر بسزایی بر کیفیت پاسخ‌های تولید شده توسط مدل دارند.

برای کشف این نوع اعلان‌های دستوری، از یک الگوریتم تکاملی استفاده می‌کند که به طور کارآمد یک جستجوی محلی^۷ را در فضای بینهایت اعلان‌ها انجام می‌دهد.

همان‌طور که در شکل ۲-۳ نشان داده شده است، مولد اعلان ساده یک چرخه سه مرحله‌ای را به صورت

تکراری دنبال می‌کند:

- **نمونه‌گیری^۸**: مولد اعلان ساده با تولید مجموعه متنوعی از اعلان‌های دستوری شروع می‌کند و اطلاعاتی مانند توصیف وظایف^۹، مثال‌ها یا تغییرات مبتنی بر اعلان‌های موفق قبلی را در نظر می‌گیرد.

- **انتخاب^{۱۰}**: از میان اعلان‌های تولید شده، مولد اعلان ساده زیرمجموعه‌ای متنوع از اعلان‌های با

¹instructed LLMs

²chat-mode

³key-value

⁴role

⁵instruction prompt

⁶Local Search

⁷Sampling

⁸task-description

⁹Selection

کیفیت را انتخاب می‌کند، به گونه‌ای که هم شباخت به اعلان‌های موفق قبلی حفظ شود و هم تنوع میان اعلان‌های انتخاب شده رعایت گردد.

- **ارزیابی^۱** : اعلان‌های انتخاب شده بر روی یک مجموعه داده کوچک ارزیابی می‌شوند و بهترین‌ها به مجموعه نامزدها^۲ افزوده می‌شوند. این مجموعه به صورت تکراری بهروز شده و روند انتخاب در آینده را هدایت می‌کند.

در طول چندین تکرار^۳، مولد اعلان ساده اعلان‌های موجود در مجموعه نامزدها را پالایش می‌کند تا اطمینان حاصل شود که این اعلان‌ها مؤثر و متناسب با وظیفه موردنظر هستند.

الگوریتم ۱ - مولد اعلان ساده

Input:

problem_description: A description of the problem domain.

epochs: Number of epochs to repeat the searching method.

C_size: Maximum size of the candidate set C .

DPP_selection_size: Number of prompts selected by DPP.

Dataset: A dataset of Q&A pairs relevant to the problem domain.

Output: C (a set of k best evaluated instruction prompts).

- 1: Initialize $C \leftarrow \emptyset$ {Start with an empty candidate set}
 - 2: **for** epoch = 1 to epochs **do**
 - 3: Generate sample set S from problem_description and Dataset.training_data
 - 4: Select set P of prompts from S by SPB-DPP , $|P| = DPP_selection_size$
 - 5: Evaluate prompts in P
 - 6: Merge P with C
 - 7: Select the k best prompts from C and set them as C for the next epoch
 - 8: **end for**
 - 9: **Return** C {Set of k best evaluated prompts}
-

۱-۲-۳ نمونه‌گیری

مرحله نمونه‌گیری مشابه فرآیند ساختن جمعیت اولیه^۴ در الگوریتم‌های تکاملی عمل می‌کند. در این مرحله، مجموعه‌ای از دستورالعمل‌های اولیه بالقوه بر اساس توصیف مسئله، جفت‌های پرسش و پاسخ موجود در مجموعه داده به عنوان نمونه، و همچنین دستورالعمل‌های موفق استخراج شده از تکرارهای قبلی الگوریتم تولید می‌گردد.

هر مجموعه داده دارای توصیف مسئله مشخصی است که در جدول ۱-۳ آورده شده‌اند.

¹Evaluation

²Candidate Set

³epoch

⁴initialization

توصیف مسئله	مجموعه داده
Solve the math word problem, giving your answer as an arabic numeral.	SVAMP, SINGLEEQ, ADDSUB, GSM8K, MULTIARITH
Solve the multiple choice math word problem, choosing (A),(B),(C),(D) or (E).	AQUA-RAT
Solve the multiple choice math word problem, choosing (A),(B),(C),(D) or (E).	CSQA
Work out an answer to the commonsense reasoning question above, and then answer yes or no.	SQA

جدول ۳ - ۱ توصیف مسئله برای مجموعه داده های مختلف

نخستین رویکرد نمونه‌گیری برگرفته از روش یادگیری بدون نمونه [۱۸] است که در آن نمونه‌ای صریح از مجموعه داده در متن اعلان ارائه نمی‌شود و تولید پاسخ صرفاً مبتنی بر توصیف مسئله انجام می‌پذیرد. به عنوان مثال، در مجموعه داده GSM8K توصیف مسئله به صورت زیر بیان شده است: "مسئله کلمه‌ای ریاضی را حل کن و پاسخ را به صورت یک عدد عربی ارائه بده." ^۱ در رویکرد دوم نمونه‌گیری، علاوه بر توصیف مسئله، از چندین پرسش و پاسخ به عنوان مثال‌هایی از خروجی مورد انتظار استفاده می‌شود که در روش مولد اعلان ساده این تعداد ۵ عدد است. این رویکرد مبتنی بر یادگیری با نمونه‌های کم [۱۹] بوده و با ارائه نمونه‌هایی از خروجی مورد انتظار، به مدل زبانی کمک می‌کند تا درک دقیق‌تری از ماهیت مسئله داشته باشد. به کارگیری این نمونه‌ها موجب تولید دستورالعمل‌های اولیه مرتبط‌تر و متناسب‌تر با مسئله خواهد شد.

در نهایت، در سومین رویکرد نمونه‌گیری، به مدل زبانی دستور داده می‌شود تا با استفاده از دستورالعمل‌های موفق و ارزیابی شده از تکرارهای پیشین، که به عنوان مجموعه کاندیدا شناخته می‌شوند، تغییرات و نسخه‌های جدیدی از این دستورالعمل‌ها تولید نماید.

در مجموع، بهره‌گیری از این سه رویکرد نمونه‌گیری منجر به ایجاد مجموعه‌ای متنوع و جامع از دستورالعمل‌های اولیه بالقوه می‌گردد. جزئیات بیشتر در خصوص این سه رویکرد نمونه‌گیری در ضمیمه B: دستورالعمل‌های اولیه برای سه رویکرد نمونه‌گیری ارائه شده است.

۲-۲-۳ انتخاب

تعیین دقت یک دستورالعمل مستلزم محاسبات قابل توجهی است، چرا که برای ارزیابی، لازم است هر دستورالعمل بر روی مجموعه آزمون اجرا شود. از این رو، ضروری است تا زیرمجموعه‌ای از دستورالعمل‌های نمونه‌گیری شده از مرحله قبلی، که دارای بیشترین پتانسیل هستند، برای ارزیابی انتخاب شوند. این زیرمجموعه

^۱Solve the math word problem, giving your answer as an arabic numeral.

باید متنوع باشد تا امکان کاوش بیشتر فراهم شود و در عین حال به دستورالعمل‌های موفق پیشین و مجموعه کاندید نزدیک باقی بماند. یک روش ساده نظیر انتخاب صرفاً دستورالعمل‌هایی با بیشترین دقت، منجر به کاهش تنوع و همگرایی سریع به سمت بهینه محلی خواهد شد. برای مقابله با این مشکل، روش مولد اعلان ساده از مدل احتمالاتی فرآیندهای نقطه‌ای دترمینانی بهره می‌گیرد.

مدل فرآیندهای نقطه‌ای دترمینانی [۶]، مدل‌های احتمالی هستند که برای انتخاب زیرمجموعه‌هایی متنوع و در عین حال مرتبط از یک مجموعه بزرگ‌تر طراحی شده‌اند. این مدل‌ها در ابتدا برای فیزیک کوانتم توسعه یافته‌اند، اما بعداً در مسائل یادگیری ماشین نظیر خلاصه‌سازی اسناد، سیستم‌های توصیه‌گر، انتخاب ویژگی و انتخاب زیرمجموعه‌های داده به کار گرفته شده‌اند [۵]. نکته مشترک در این کاربردها، بهینه‌سازی همزمان دو معیار تنوع^۱ و کیفیت^۲ است. در فرآیندهای نقطه‌ای دترمینانی، یک ماتریس کرنل^۳ نیمه مثبت معین^۴ L تعریف می‌شود، که هر درایه L_{ij} میزان شباهت^۵ بین عناصر^۶ i و j را نشان می‌دهد. احتمال انتخاب زیرمجموعه‌ای $Y \subseteq S$ ، که Y مجموعه اصلی است، متناسب با دترمینان زیرماتریس L_S خواهد بود، که متناظر با سطرها و ستون‌های شاخص‌گذاری شده توسط S است:

$$P(S) \propto \det(L_S) \quad (1-3)$$

در این روش، دترمینان به عنوان معیار تنوع عمل می‌کند و زیرمجموعه‌هایی با عناصر نامشابه‌تر را ترجیح می‌دهد. به طور شهودی، زیرمجموعه‌هایی با عناصر نامشابه‌تر دارای دترمینان بزرگ‌تری هستند، چرا که بردارهای متعامد^۶ یا تقریباً متعامد^۷ در ماتریس کرنل، به دترمینان بزرگ‌تری منجر می‌شوند.

مدل فرآیندهای نقطه‌ای دترمینانی شرطی، مدلی احتمالی است که به زیرمجموعه‌ای از آیتم‌ها، با توجه به ورودی یا قیودی مشخص، احتمال تخصیص می‌دهد. در روش مولد اعلان ساده، فرآیندهای نقطه‌ای دترمینانی شرطی توازن میان کیفیت (برحسب شباهت به دستورالعمل‌های موفق) و تنوع (برحسب عدم شباهت بین اعضاء زیرمجموعه) را برقرار می‌سازد.

به زبان ریاضی، احتمال انتخاب زیرمجموعه Y از مجموعه اصلی (X) با توجه به ورودی X متناسب با دترمینان یک ماتریس کرنل نیمه مثبت معین $(L_Y(X))$ خواهد بود.

فرض کنید \mathcal{X} فضای ورودی (مثلاً مجموعه نمونه‌ها) و (X) مجموعه آیتم‌ها با توجه به ورودی X

¹Diversity

²Quality

³Kernel Matrix

⁴Positive semi-definit

⁵Similarity

⁶orthogonal

⁷near-orthogonal

(مثلاً مجموعه دستورالعمل‌های نمونه‌گیری شده) باشد. تعریف رسمی به صورت زیر خواهد بود:
تعریف: یک فرآیندهای نقطه‌ای دترمینانی شرطی به صورت $P(\mathbf{Y} = Y|X)$ مدلی احتمالاتی است که به هر زیرمجموعه ممکن $Y \subseteq \mathcal{Y}(X)$ احتمال تخصیص می‌دهد و به صورت زیر تعریف می‌شود:

$$P(\mathbf{Y} = Y|X) \propto \det(L_Y(X)), \quad (2-3)$$

که در آن $L_Y(X)$ ماتریس کرنل نیمه مثبت معینی با ابعاد $|\mathcal{Y}(X)| \times |\mathcal{Y}(X)|$ و وابسته به X است.
ثابت نرمال‌سازی فرآیندهای نقطه‌ای دترمینانی شرطی به صورت کارآمد با رابطه $\det(L(X) + I)$ محاسبه می‌شود. اگر تنوع برابر ϕ و کیفیت برابر q باشد، با استفاده از تجزیه کیفیت-تنوع، خواهیم داشت:

$$L_{ij}(X) = q_i(X)\phi_i(X)^T\phi_j(X)q_j(X) \quad (3-3)$$

که در آن $\phi_i(X) \in \mathbb{R}^D$ و $q_i(X) \in \mathbb{R}^+$ تعیین شده‌اند و هر دو به X وابسته هستند.

روش مولد اعلان ساده، تجزیه‌ای از فرآیندهای نقطه‌ای دترمینانی شرطی به کار گرفته است که به طور صریح توازن میان تنوع و کیفیت آیتم‌ها را نشان می‌دهد. برای این منظور، ساخت ماتریس کرنل بر مبنای دو مؤلفه کیفیت و تنوع انجام می‌شود. پارامتر تنوع از طریق شباهت زوجی میان دستورالعمل‌های موجود در مجموعه نمونه^۱ با نام Φ از ابعاد $n \times n$ سنجیده می‌شود. برای محاسبه Φ ، ابتدا یک ماتریس صفر از ابعاد $n \times n$ ایجاد می‌شود و سپس برای هر جفت از دستورالعمل‌ها (p_i, p_j) در مجموعه transformer sentence $S = \{p_1, p_2, \dots, p_n\}$ استخراج می‌شوند. سپس شباهت کسینوسی بین e_i و e_j مطابق رابطه زیر محاسبه می‌شود:

$$\text{similarity}(e_i, e_j) = \frac{\sum_{k=1}^d e_i^k \cdot e_j^k}{\sqrt{\sum_{k=1}^d (e_i^k)^2} \cdot \sqrt{\sum_{k=1}^d (e_j^k)^2}} \quad (4-3)$$

که در آن d ابعاد بردارهای عددی است. در نهایت:

$$\Phi[i][j] = \text{similarity}(e_i, e_j) \quad (5-3)$$

¹ Sample set

از طرف دیگر، ماتریس Q_d که نشان دهنده پارامتر کیفیت است، از ابعاد $1 \times n$ است که فاصله هر دستورالعمل را از مجموعه دستورالعمل های موفق $S_{success}$ می سنجد. فرآیند محاسبه Q_d با محاسبه شباهت کسینوسی بین هر $p_j \in S_{success}$ و هر $p_i \in S$ با استفاده از رابطه ۴-۳ آغاز می شود. سپس:

$$\mu_i = \frac{1}{m} \sum_{k=1}^m s_{ik} \quad (6-3)$$

و انحرافات مثبت از میانگین محاسبه می گردد:

$$s'_{ik} = \max(0, s_{ik} - \mu_i) \quad (7-3)$$

که سپس با تابع softmax نرمال سازی می شود:

$$w_{ik} = \frac{\exp(s'_{ik})}{\sum_{j=1}^m \exp(s'_{ij})}. \quad (8-3)$$

وزن نهایی هر دستورالعمل p_i به صورت زیر محاسبه می شود:

$$W_i = \sum_{k=1}^m w_{ik} \quad (9-3)$$

و در درایه i ام قطر اصلی Q_d قرار می گیرد.
در نهایت، این روش ماتریس کرنل را به صورت زیر می سازد:

$$L = Q_d \otimes \Phi \otimes Q_d \quad (10-3)$$

که در آن نماد \otimes به معنای ضرب عنصر به عنصر^۱ است. سپس، از این ماتریس در فرآیندهای نقطه ای دترمینانی برای انتخاب زیرمجموعه بهینه از نظر تنوع و کیفیت استفاده می شود. در این فرآیند، زیرمجموعه هایی که نقاط انتخابی آنها پراکندگی بیشتری دارند احتمال بالاتری خواهند داشت.

¹ element-wise product

۳-۲-۳ ارزیابی

برای ارزیابی^۱ از معیار دقت^۲ استفاده شده است. برای بدست آوردن دقت اعلان های زیرمجموعه^{*} انتخاب شده توسط فرآیندهای نقطه ای دترمینانی ، زیرمجموعه ای شامل ۱۰۰ جفت پرسش و پاسخ از هر مجموعه داده به عنوان مجموعه اعتبارسنجی^۳ مورد استفاده قرار می گیرد. هر دستور آموزشی در زیرمجموعه انتخاب شده برای هدایت مدل زبانی بزرگ به کار می رود. سپس این مدل زبانی هدایت شده، برای هر پرسش در مجموعه اعتبارسنجی، پاسخ هایی تولید می کند که به آنها پاسخ های پیش بینی شده گفته می شود. این پاسخ های پیش بینی شده سپس نیاز به اعتبارسنجی دارند. برای اعتبارسنجی، از رویکردی تحت عنوان مدل زبانی به عنوان داور^۴ استفاده می کنیم که در آن، پاسخ پیش بینی شده با پاسخ واقعی مقایسه می شود. ما از مدل Mistral به عنوان داور استفاده می کنیم.

جريان کلی این فرآیند در شکل ۳-۳ با استفاده از نمونه ای از مجموعه داده GSM8K نشان داده شده است. همانطور که دیده می شود، پس از ارزیابی دستورات آموزشی انتخاب شده، آنها با دستورات آموزشی موفق از آخرین دوره آموزشی ادغام می شوند. علاوه بر این، بهترین دستورات آموزشی برتر (k مورد بر اساس دقت آزمایشی) به عنوان دستورات آموزشی موفق جدید انتخاب می شوند تا در تکرار های بعدی الگوریتم مورد استفاده قرار گیرند. مقدار k در روش مولد اعلان ساده برابر ۱۰ است، این مقدار با آزمون و خطای مختلف بدست آمده است.

¹Evaluation

²Accuracy

³Validation set

⁴LLM-as-a-Judge

For each **INSTRUCTION-PROMPT** on GSM8K dataset :



Guidelines

- Understand the Task: Use the problem_description to understand the overall goal. Supplement this understanding with the two provided examples to clarify the problem's scope.
- ⋮

Edge Cases:

- Handle problems that involve multiple types of operations (e.g., addition, subtraction, multiplication, and division).
- Handle problems that require rounding or estimating values.
- Handle problems that involve fractions or decimals.

Ask all questions in validation set from the instructed Mistral

GSM8K



Validation set

Question: Every day, Wendi feeds each of her chickens three cups of mixed chicken feed, containing seeds, mealworms and vegetables to help keep them healthy. She gives the chickens their feed in three separate meals. In the morning, she gives her flock of chickens 15 cups of feed. In the afternoon, she gives her chickens another 25 cups of feed. How many cups of feed does she need to give her chickens in the final meal of the day if the size of Wendi's flock is 20 chickens?

Answer: 20

Analyze the problem_description

⋮

Solution:

- Calculate the total amount of feed given so far by adding the amount of feed given in the morning and afternoon ($15 + 25 = 40$ cups).
- Determine the amount of feed each chicken has received so far by dividing the total amount of feed given so far by the number of chickens ($40 / 20 = 2$ cups per chicken).
- Calculate the amount of feed needed for the final meal by subtracting the amount of feed each chicken has received so far from the amount of feed each chicken needs in a day ($3 - 2 = 1$ cup per chicken).
- Multiply the amount of feed needed for the final meal by the number of chickens ($1 * 20 = 20$ cups).
- The final meal of the day requires 20 cups of feed.



Evaluate the this answers with evaluation instructed Mistral



By the provided information, Task:

- Compare the Predicted Answer with the Actual Answer.
- Determine if the Predicted Answer matches the Actual Answer exactly.
- Output "CORRECT" if the answers match, otherwise output "INCORRECT".

Response Format:

{CORRECT or INCORRECT}



CORRECT

شكل ٣-٣ دیاگرام مرحله ارزیابی روش مولد اعلان ساده

فصل چهارم: نتایج و بحث

۱-۴ مقدمه

در این فصل به معرفی مجموعه داده‌های استفاده شده و همچنین شرایط و جزئیات هر آزمایش پرداخته می‌شود. در نهایت نتایج حاصل از هر آزمایش آورده شده است.

۲-۴ مجموعه داده‌های مورد استفاده

روش مولد اعلان ساده را در مجموعه‌ای گسترده از وظایف مرتبط با استدلال حسابی^۱ و استدلال مبتنی بر درک مسئله^۲ مورد ارزیابی قرار داده‌ایم. برای این منظور از هشت مجموعه داده‌ی عمومی، شامل [A] SingleEq [۱۳] AQuA-RAT [۹] AddSub [۱۰] MultiArith [۱۴] SVAMP [۱۵] StrategyQA [۱۲] CommonsenseQA [۱۱] و به همراه استفاده شده است.

همانطور که در جدول ۱-۴ می‌بینید، روش مولد اعلان ساده با روش‌های زنجیره تفکر، برنامه‌ریزی و حل با اعلان بهبود یافته و بدون آن، مهندس اعلان خودکار و بهینه‌سازی با اعلان‌ها مورد مقایسه قرار گرفته است. بالاترین دقت برای هر مجموعه داده به صورت فونت برجسته مشخص شده است. نتایج مدل زبان PaLM-۰۰۳ text-davinci-۰-۰ مستقیماً از مقاله‌ی برنامه‌ریزی و حل [۳] استخراج شده و نتایج مدل زبان L-۲ نیز به طور مستقیم از مقاله‌ی مولد اعلان [۴] گرفته شده‌اند. جهت تضمین مقایسه‌ای عادلانه، تمامی روش‌ها با مدل زبانی Mistral ارزیابی شده‌اند و سپس نتایج مقایسه شده است.

Results from	Method	LLM	MultiArith	SingleEq	AddSub	SVAMP	SQA	CSQA	AQuA-RAT	GSM&K
PS [۱۳]	COT	text-davinci	83.8	88.1	85.3	69.9	63.8	65.2	38.9	56.4
	POT	text-davinci	92.2	91.7	85.1	70.8	-	-	43.9	57.0
	PS	text-davinci	87.2	89.2	88.1	72.0	-	-	42.5	58.2
	PS+	text-davinci	91.8	94.7	92.2	75.7	65.4	71.9	46.0	59.3
PB[۱۴]	PS	PaLM ۰-L	97.7	90.6	72.4	83.8	50.0	77.9	40.2	59.0
	PS+	PaLM ۰-L	92.5	94.7	74.4	86.3	50.1	73.3	39.4	60.5
	APE	PaLM ۰-L	95.8	82.2	72.2	73.0	38.4	67.3	45.7	77.9
	OPRO	PaLM ۰-L	-	-	-	-	-	-	-	۷۸.۰
	PB	PaLM ۰-L	99.7	96.4	87.8	90.2	71.8	85.4	62.2	83.9
ساده (us) اعلان مولد	CoT	Mistral	79.0	85.0	78.0	64.0	54.0	51.0	33.0	72.0
	PS	Mistral	68.0	91.0	78.0	68.0	61.0	43.0	36.0	66.0
	PS+	Mistral	82.0	84.0	80.0	62.0	44.0	27.0	26.0	60.0
	APE	Mistral	78.0	86.0	84.0	69.0	54.0	51.0	36.0	66.0
	OPRO	Mistral	85.0	84.0	78.0	61.0	53.0	35.0	34.0	67.0
	ساده (ours) اعلان مولد	Mistral	92.0	94.0	89.0	93.0	92.0	91.0	93.0	84.0

جدول ۱-۴ مقایسه دقت روش مولد اعلان ساده با سایر روش‌های موجود

۳-۴ متغیرهای روش مولد اعلان ساده

از آنجایی که مولد اعلان ساده یک الگوریتم تکاملی است، این الگوریتم را در بازه ۲۰ تا ۳۰ تکرار ۳ اجرا می‌کنیم؛ تا جایی که در آخرین تکرارها، مجموعه کاندیدا تغییر نخواهد کرد. در هر دوره، جمعیتی

^۱ arithmetic reasoning

^۲ commonsense reasoning

^۳ Epoch

متشكل از ۱۲۰ تا ۱۵۰ اعلان دستوری، بسته به مجموعه کاندیدا، تولید می‌شود. همانطور که در فصل قبل گفته شد، این اعلان‌ها بر اساس توضیحات مسئله و یک مجموعه آموزشی شامل ۲۰ سوال و پاسخ از مجموعه داده، نمونه‌گیری می‌شوند. از میان این جمعیت، زیرمجموعه‌ای شامل ۱۰ اعلان دستوری انتخاب شده و بر روی یک مجموعه اعتبارسنجی که شامل ۲۰ سوال و پاسخ متمایز از مجموعه آموزشی است، ارزیابی می‌گردد.

پس از اتمام تمامی دوره‌ها، اعلان‌های دستوری با بهترین عملکرد بر روی مجموعه‌ای شامل ۱۰۰ سوال و پاسخ که به طور تصادفی از مجموعه داده انتخاب شده و از هر دو مجموعه آموزشی و اعتبارسنجی متمایز می‌باشد، مورد آزمون قرار می‌گیرند.

پارامتر توضیحات مسئله به مجموعه داده بستگی دارد که در جدول ۳-۱ توضیح متناظر با هر مجموعه داده آورده شده است.

در این پژوهش، از مدل زبانی VB-Mistral [۳۴] برای تولید و آزمون اعلان‌های دستوری استفاده می‌شود. همچنین، به منظور تبدیل اعلان‌های دستوری به بردار عددی، از مدل تبدیل جمله‌ی bert-base-^۱ موجود در سایت HuggingFace nli-mean-tokens بهره گرفته شده است.

۴-۴ روش‌های مرجع

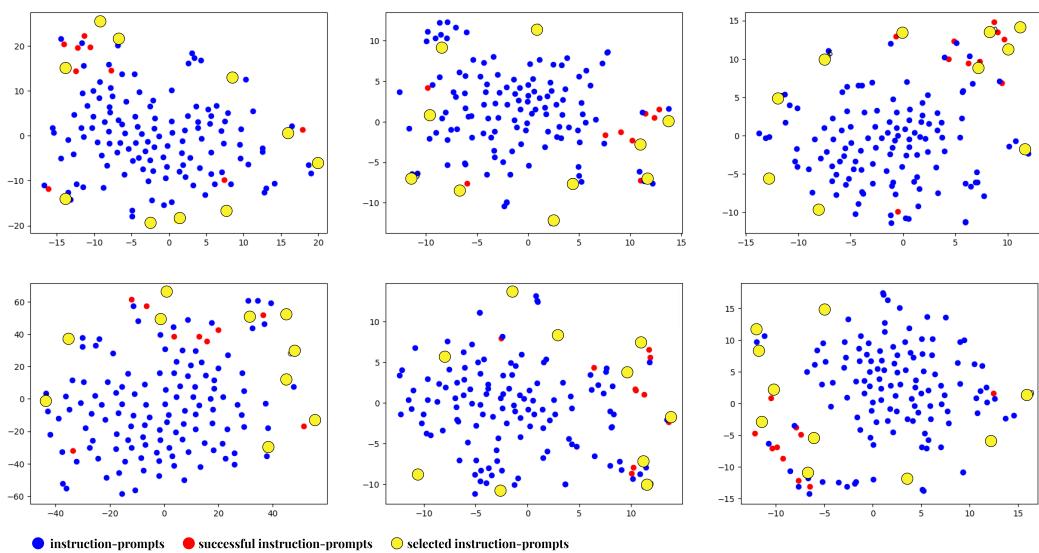
نتایج روش مولد اعلان ساده با استفاده از پیشرفته‌ترین روش‌های مهندسی که در فصل‌های قبلی به تفصیل توضیح داده شده‌اند، مورد مقایسه قرار گرفته‌اند. به‌منظور تضمین مقایسه‌ای عادلانه، در صورت امکان، این روش‌ها با استفاده از مدل پایه Mistral مجددًا اجرا شده‌اند زیرا امکان اجرای روش مولد اعلان ساده با مدل‌های text-davinci-۰۰۳ و PaLM-L وجود ندارد چرا که این مدل‌ها به صورت عمومی در دسترس نمی‌باشند و روش‌های دسترسی به آن‌ها منسخ شده است. اعلان‌های دستوری این روش‌ها به‌طور مستقیم از مقاله مولد اعلان استخراج شده‌اند؛ اما اعلان‌های تولید شده توسط مولد اعلان مورد آزمون قرار نگرفته‌اند زیرا آن‌ها به گونه‌ای برای PaLM ۲ بهینه‌سازی شده‌اند که استفاده از آن‌ها در مدل Mistral مقایسه‌ای منصفانه ارائه نخواهد داد.

لازم به یادآوری است که برای هر مجموعه داده و هر روش، از دقت به عنوان معیار ارزیابی استفاده شده است.

^۱Sentence Embedding

۵-۴ نمایش دو بعدی اعلان ها

برای درک بهتر عملکرد مدل احتمالاتی فرآیندهای نقطه‌ای دترمینانی در طول تکرارها، در شکل ۱-۴ توزیع دو بعدی بردار های عددی اعلان ها را نمایش داده‌ایم. برای کاهش ابعاد این بردار های عددی، ابتدا از روش کاهش بعد PCA [۳۵] برای کاهش ابعاد به ۵۰ و سپس از روش t-SNE [۳۶] برای کاهش ۵۰ بعد به ۲ بعد استفاده کردیم. همانطور که دیده می‌شود، توزیع یکنواخت دستورالعمل های انتخاب شده در نواحی مختلف فضای جستجو نمایش داده شده است که بیانگر توازن میان تنوع و کیفیت در این روش است.



شکل ۱-۴ نمایش دو بعدی بردار اعلان ها برای ارزیابی انتخاب در روش مولد اعلان ساده

۶-۴ نتایج

نتایج اجرای روش مولد اعلان ساده بر روی مجموعه داده های مذکور نشان می‌دهد که این روش از پیشرفته‌ترین روش‌های مهندسی برتری دارد. همانطور که در جدول ۱-۴ مشاهده می‌شود، مولد اعلان ساده در تمامی مجموعه داده ها از سایر روش ها برتر است.

ابتدا عملکرد هر روش را در میان مدل های زبانی مختلف ارزیابی کردیم. مشاهده می‌شود که اکثر روش ها دقت مشابهی در میان مدل های مختلف دارند، اگرچه بهبودهایی در عملکرد هنگام تغییر از text-Mistral به PaLM 2-L یا davinci-003 موجود در دقت و ساختار مدل های زبانی نسبت داد.

برای تضمین صحت مقایسه، از روش برنامه‌ریزی و حل به عنوان نقطه مرجع استفاده شده و بهبود نسبی

مولد اعلان و مولد اعلان ساده محاسبه گردیده است. در روش مولد اعلان، با استفاده از فرمول زیر به بهبود نسبی ۲۳.۴٪ نسبت به روش برنامه‌ریزی و حل در میان مجموعه داده‌ها دست یافتیم:

$$\text{بهبود نسبی مولد اعلان} = \frac{1}{8} \left[\frac{99.7}{97.7} + \frac{96.4}{90.6} + \frac{87.8}{72.4} + \frac{90.2}{83.8} + \frac{71.8}{50.0} + \frac{85.4}{77.9} + \frac{62.2}{40.2} + \frac{83.9}{59.0} \right],$$

در همین حال، برای مولد اعلان ساده با فرمول زیر، به بهبود نسبی ۵۴.۷٪ نسبت به روش برنامه‌ریزی و حل در میان مجموعه داده‌ها دست یافتیم:

$$\text{بهبود نسبی مولد اعلان ساده} = \frac{1}{8} \left[\frac{92.0}{68.0} + \frac{94.0}{91.0} + \frac{89.0}{78.0} + \frac{93.0}{68.0} + \frac{92.0}{61.0} + \frac{91.0}{43.0} + \frac{93.0}{36.0} + \frac{84.0}{66.0} \right],$$

۷-۴ سربار محاسباتی

همانطور که در بخش پارامتر های روش مولد اعلان ساده توضیح داده شد، مولد اعلان ساده به طور مشابه با مولد اعلان در بازه ۲۰ تا ۳۰ تکرار اجرا می شود. در هر تکرار، مولد اعلان ساده جمعیتی متشكل از ۱۰۰ تا ۱۲۰ اعلان دستوری تولید می کند که برای هر یک، یک فراخوانی مدل زبانی مورد نیاز است. در مقابل، مولد اعلان با جمعیتی متشكل از ۵۰ واحد عمل می کند؛ هر واحد شامل یک جفت اعلان است که منجر به ۱۰۰ فراخوانی مدل زبانی برای تولید آنها می شود. در گام بعدی، برای انتخاب و جهش، مولد اعلان از رویکرد ترکیبی جهش و ابرجهش استفاده می کند که برای هر آیتم از واحدهای جمعیت، حداقل دو فراخوانی مدل زبانی نیاز دارد. بنابراین، تعداد کل فراخوانی های مدل زبانی در این مرحله برای مولد اعلان به صورت زیر محاسبه می شود:

$$50 \times 2 \times 2 = 200.$$

در مقابل، مولد اعلان ساده این محاسبات را با استفاده از محاسبات روش DPP انجام می دهد و از فراخوانی های اضافی مدل زبانی در این گام صرف نظر می کند. در انتهای ارزیابی، مولد اعلان کل جمعیت را مورد ارزیابی قرار داده و به هر واحد نمره می دهد.

از آنجا که هر واحد در جمعیت مولد اعلان شامل یک جفت اعلان است، ارزیابی ۵ واحد نیازمند:

$$50 \times 2 = 100$$

فراخوانی مدل زبانی می‌باشد. اما مولد اعلان ساده تنها ۱۰ اعلان دستوری از میان ۱۰۰ اعلان برای ارزیابی انتخاب می‌کند و هر اعلان یک فراخوانی مدل زبانی نیاز دارد؛ بنابراین:

$$10 \times 1 = 10.$$

از سوی دیگر، اندازه‌ی مجموعه داده اعتبارسنجی برای مولد اعلان شامل ۱۰۰ جفت سوال و پاسخ است، در حالی که برای مولد اعلان ساده به ۲۰ کاهش یافته است. علاوه بر این، مولد اعلان ساده پس از شناسایی اعلان‌های با عملکرد برتر، گام آزمون را به منظور تضمین مقایسه‌ای منصفانه اضافه می‌کند؛ این گام اعلان‌های انتخاب شده را بر روی مجموعه داده‌ای شامل ۱۰۰ جفت سوال و پاسخ مانند مولد اعلان ارزیابی می‌کند.

تعداد کل فراخوانی‌های مدل زبانی برای مولد اعلان و مولد اعلان ساده در هر دوره به صورت زیر بیان می‌شود:

$$\text{کل فراخوانی‌های مدل زبانی برای مولد اعلان} = 100 + 200 + 100 = 400,$$

$$\text{کل فراخوانی‌های مدل زبانی برای مولد اعلان ساده} = 110 \text{ تا } 120 + 10 = 110 \text{ تا } 130.$$

بنابراین، مولد اعلان ساده سربار محاسباتی را تقریباً به میزان ۳ برابر کاهش می‌دهد.

۸-۴ خروجی‌ها

با بررسی خروجی‌های روش مولد اعلان ساده، مشاهده شد که اعلان‌های دستوری با بهترین عملکرد برای هر مجموعه داده، در قالب چارچوبی صریح ساختاربندی شده‌اند. این چارچوب شامل چند بخش است: بخش اول، توضیح مفصل وظیفه می‌باشد که با دقت به تشریح آن می‌پردازد و در برخی موارد شامل دستورالعمل‌هایی است که جنبه‌های مختلف وظیفه را توضیح می‌دهند. بخش دوم این اعلان، مسیر حل مسئله با توضیحات گام به گام جهت رفع مشکلات موجود در حوزه مربوطه است. در مرحله بعد، تعدادی مثال ارائه شده است که به عنوان نمایش نمونه از مجموعه آموزشی با ارائه راه حل کامل و گام به گام، جهت هدایت روند تفکر مدل زبانی به کار رفته‌اند.

برای مثال، اعلان دستوری با بهترین عملکرد برای مجموعه داده MultiArith در ابتدا وظیفه را به‌طور

کامل تعریف می‌کند، سپس پنج مثال به همراه راه حل کامل آن‌ها ارائه می‌دهد و در نهایت، دستورالعمل‌هایی برای حل مسائل در این حوزه بیان می‌نماید. متن کامل این اعلان دستوری در کادر ۴-۸ آورده شده است.

بهترین اعلان دستوری تولید شده برای مجموعه داده MultiArith

Task: Analyze the given math problem, break down the information, and compute the answer step by step. Present the final answer as an Arabic number.

Examples:

- . ۱ Understand that April makes \$9 for every rose she sells. Estimate how many roses April started with, then calculate how many roses she ended with, in order to figure out the revenue she generated.
 - Compute the difference between the number of roses April started with and the number of roses she ended with to find the quantity of roses April sold.
 - Multiply the number of roses sold by the cost per rose to determine the earnings incurred.
- . ۲ Understand the scenario in which Megan prepared 68 cupcakes, and her brother, Todd, consumed 32 of them. Find out how many packages Megan can assemble with 6 cupcakes per package.
 - Subtract the number of cupcakes Todd consumed from the total number of cupcakes Megan prepared to determine the remaining amount of uneaten cupcakes.
 - Divide the quantity of uneaten cupcakes by the number of cupcakes needed to make one package to find the number of packages Megan can assemble.
- . ۳ Identify the situation in which Katie organized and counted 9 albums with a consistent number of pictures in each album. Determine the number of pictures in each

album.

- Sum the total number of pictures Katie has to obtain the grand total.

- Divide the grand total by the number of albums to figure out the number of pictures in each album.

.^f Grasp the details that Tiffany earns 6 points for each treasure she discovers. Figure out the point score Tiffany received from finding treasures on the first and second levels.

- Add the number of treasures Tiffany discovered on the first and second levels to get the sum of treasures found.

- Multiply the total number of treasures by the points per treasure to learn the point score Tiffany attained.

.^d Recall the situation in which a waiter manages 33 customers. Estimate how many customers the waiter is still serving after some prospective diners left and others joined the restaurant.

- Subtract the number of customers who left from the current number of customers to determine the number of customers still at the restaurant.

- Add the number of new customers who arrived to find the updated number of customers the waiter is currently serving.

Instructions:

.ⁱ Explore the given problem to recognize the necessary computations needed to derive the solution.

.^r Break down the problem into more manageable parts for better organization and clarity.

. \natural Calculate the solutions step by step.

. \diamond Deliver the numerical solution.

فصل پنجم:

نتیجه گیری و پیشنهادات آتی

۱-۵ نتیجه‌گیری

در این پژوهش، با هدف ارتقای فرآیند تولید و بهینه‌سازی اعلان‌ها برای مدل‌های زبانی بزرگ، روشی نوین و کم‌هزینه با عنوان مولد اعلان شاده^۱ طراحی و پیاده‌سازی گردید. دلیل اصلی انجام این تحقیق، چالش‌های محاسباتی موجود در روش‌های پیشین نظری مولد اعلان [۴] بود که به سبب پیچیدگی‌های ذاتی خود، در جستجوی فضای وسیع اعلان‌ها با محدودیت‌های جدی مواجه بودند. به همین منظور، تلاش شد تا با ارائه رویکردی مبتنی بر جستجوی محلی، کارایی و بهره‌وری فرآیند بهینه‌سازی اعلان‌ها به شکل محسوسی افزایش یابد.

نتایج حاصل از آزمایش‌ها نشان می‌دهد که روش مولد اعلان شاده، ضمن حفظ دقت مطلوب، توانسته است به میزان قابل توجهی هزینه‌های محاسباتی را در مقایسه با الگوریتم‌های تکاملی مشابه کاهش دهد. این دستاورده، مؤید اثربخشی روش پیشنهادی در تسهیل و تسريع فرآیند مهندسی اعلان در زمینه‌های پژوهشی و کاربردی می‌باشد. به طور کلی، روش مولد اعلان شاده به عنوان ابزاری کارآمد، قادر است به نیازهای پروژه‌ها و سامانه‌هایی که با محدودیت منابع محاسباتی روبرو هستند، پاسخ موثری ارائه دهد.

۲-۵ پیشنهادات آتی

با توجه به نتایج مثبت حاصل از این پژوهش و همچنین چالش‌های موجود در زمینه بهینه‌سازی اعلان‌ها برای مدل‌های زبانی، پیشنهاد می‌شود مسیرهای زیر در مطالعات آتی مورد توجه قرار گیرد:

۱. **گسترش مطالعات بر روی داده‌های گسترده‌تر و متنوع‌تر:** بررسی عملکرد الگوریتم مولد اعلان شاده بر روی مجموعه داده‌هایی با ابعاد و تنوع بیشتر، می‌تواند میزان پایداری و تعیین‌پذیری این الگوریتم را در محیط‌های عملیاتی واقعی مورد ارزیابی قرار دهد.

۲. **افزودن شاخص‌های تکمیلی برای سنجش کیفیت و تنوع:** در ادامه پژوهش، می‌توان با افزودن معیارهای جدید جهت سنجش تنوع و کیفیت اعلان‌ها، الگوریتم را در برابر خطر همگرایی زودهنگام و تولید اعلان‌های یکنواخت مقاوم‌تر ساخت و توازن مطلوبی میان اکتشاف و بهره‌برداری برقرار نمود.

۳. **سازگاری با معماری‌های نوین مدل‌های زبانی:** با توجه به ظهور معماری‌های نوین در حوزه مدل‌های زبانی نظری معماری‌های مبتنی بر Sparse Attention و یا مدل‌های چندوجهی^۲، توسعه نسخه‌های بهینه‌شده از مولد اعلان شاده متناسب با این معماری‌ها می‌تواند گامی مؤثر در راستای افزایش انعطاف‌پذیری و قابلیت‌های این الگوریتم باشد.

¹Simple Prompt Breeder

²multimodal

۴. بررسی و پیاده‌سازی سایر روش‌های جستجوی محلی: از آنجا که مولد اعلان شاده ماهیتاً مشابه الگوریتم Hill-Climbing عمل می‌نماید، پیشنهاد می‌شود در مطالعات آتی از رویکردهای جایگزین نظیر Tabu Search یا Simulated Annealing تنوع در فرآیند جستجو، به بهبود کارایی الگوریتم در مسائل بیچیده‌تر منجر شود.

۵. ادغام با رویکردهای مبتنی بر یادگیری تقویتی: به منظور بهبود تدریجی و هدفمند سیاست‌های جستجو، می‌توان مولد اعلان شاده را با الگوریتم‌های یادگیری تقویتی ترکیب نمود و از این طریق، فرآیند بهینه‌سازی اعلان‌ها را با دریافت بازخوردهای پویا از محیط و مدل هدف، هوشمندانه‌تر و اثربخش‌تر ساخت.

در نهایت، نتایج این پژوهش زمینه‌ساز ارائه یک چارچوب توسعه‌پذیر برای بهینه‌سازی اعلان‌ها در مدل‌های زبانی بزرگ بوده و می‌تواند بستر مناسبی برای تحقیقات و کاربردهای آینده در حوزه مهندسی اعلان فراهم آورد.

كتاب نامه

- [1] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models, 2023.
- [2] Wenhui Chen, Xueguang Ma, Xinyi Wang, and William W. Cohen. Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks, 2023.
- [3] Lei Wang, Wanyu Xu, Yihuai Lan, Zhiqiang Hu, Yunshi Lan, Roy Ka-Wei Lee, and Ee-Peng Lim. Plan-and-solve prompting: Improving zero-shot chain-of-thought reasoning by large language models, 2023.
- [4] Chrisantha Fernando, Dylan Banarse, Henryk Michalewski, Simon Osindero, and Tim Rocktäschel. Promptbreeder: Self-referential self-improvement via prompt evolution, 2023.
- [5] Alex Kulesza. Determinantal point processes for machine learning. *Foundations and Trends® in Machine Learning*, 5(2–3):123–286, 2012.
- [6] Alexei Borodin. Determinantal point processes, 2009.
- [7] Subhro Roy and Dan Roth. Solving general arithmetic word problems, 2016.
- [8] Rik Koncel-Kedziorski, Hannaneh Hajishirzi, Ashish Sabharwal, Oren Etzioni, and Siena Dumas Ang. Parsing algebraic word problems into equations. *Transactions of the Association for Computational Linguistics*, 3:585–597, 2015.
- [9] Mohammad Javad Hosseini, Hannaneh Hajishirzi, Oren Etzioni, and Nate Kushman. Learning to solve arithmetic word problems with verb categorization. In Alessandro Moschitti, Bo Pang, and Walter Daelemans, editors, *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 523–533, Doha, Qatar, October 2014. Association for Computational Linguistics.
- [10] Arkil Patel, Satwik Bhattacharya, and Navin Goyal. Are nlp models really able to solve simple math word problems?, 2021.

- [11] Mor Geva, Daniel Khashabi, Elad Segal, Tushar Khot, Dan Roth, and Jonathan Berant. Did aristotle use a laptop? a question answering benchmark with implicit reasoning strategies, 2021.
- [12] Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. CommonsenseQA: A question answering challenge targeting commonsense knowledge. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4149–4158, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.
- [13] Wang Ling, Dani Yogatama, Chris Dyer, and Phil Blunsom. Program induction by rationale generation : Learning to solve and explain algebraic word problems, 2017.
- [14] Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems, 2021.
- [15] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2023.
- [16] Gokul Yenduri, Ramalingam M, Chemmalar Selvi G, Supriya Y, Gautam Srivastava, Praveen Kumar Reddy Maddikunta, Deepti Raj G, Rutvij H Jhaveri, Prabadevi B, Weizheng Wang, Athanasios V. Vasilakos, and Thippa Reddy Gadekallu. Generative pre-trained transformer: A comprehensive review on enabling technologies, potential applications, emerging challenges, and future directions, 2023.
- [17] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding, 2019.
- [18] Soravit Changpinyo, Wei-Lun Chao, Boqing Gong, and Fei Sha. Classifier and exemplar synthesis for zero-shot learning, 2019.

- [19] Archit Parnami and Minwoo Lee. Learning from few examples: A summary of approaches to few-shot learning, 2022.
- [20] Sang Michael Xie, Aditi Raghunathan, Percy Liang, and Tengyu Ma. An explanation of in-context learning as implicit bayesian inference, 2022.
- [21] Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners, 2023.
- [22] Rohan Anil, Andrew M. Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Paschos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, Eric Chu, Jonathan H. Clark, Laurent El Shafey, Yanping Huang, Kathy Meier-Hellstern, Gaurav Mishra, Erica Moreira, Mark Omernick, Kevin Robinson, Sebastian Ruder, Yi Tay, Kefan Xiao, Yuanzhong Xu, Yujing Zhang, Gustavo Hernandez Abrego, Junwhan Ahn, Jacob Austin, Paul Barham, Jan Botha, James Bradbury, Siddhartha Brahma, Kevin Brooks, Michele Catasta, Yong Cheng, Colin Cherry, Christopher A. Choquette-Choo, Aakanksha Chowdhery, Clément Crepy, Shachi Dave, Mostafa Dehghani, Sunipa Dev, Jacob Devlin, Mark Diaz, Nan Du, Ethan Dyer, Vlad Feinberg, Fangxiaoyu Feng, Vlad Fienber, Markus Freitag, Xavier Garcia, Sebastian Gehrmann, Lucas Gonzalez, Guy Gur-Ari, Steven Hand, Hadi Hashemi, Le Hou, Joshua Howland, Andrea Hu, Jeffrey Hui, Jeremy Hurwitz, Michael Isard, Abe Ittycheriah, Matthew Jagielski, Wenhao Jia, Kathleen Kenealy, Maxim Krikun, Sneha Kudugunta, Chang Lan, Katherine Lee, Benjamin Lee, Eric Li, Music Li, Wei Li, YaGuang Li, Jian Li, Hyontaek Lim, Hanzhao Lin, Zhongtao Liu, Frederick Liu, Marcello Maggioni, Aroma Mahendru, Joshua Maynez, Vedant Misra, Maysam Moussalem, Zachary Nado, John Nham, Eric Ni, Andrew Nystrom, Alicia Parrish, Marie Pellat, Martin Polacek, Alex Polozov, Reiner Pope, Siyuan Qiao, Emily Reif, Bryan Richter, Parker Riley, Alex Castro Ros, Aurko Roy, Brennan Saeta, Rajkumar Samuel, Renee Shelby, Ambrose Slone, Daniel Smilkov, David R. So, Daniel Sohn, Simon Tokumine, Dasha Valter, Vijay Vasudevan, Kiran Vodrahalli, Xuezhi Wang, Pidong Wang, Zirui Wang, Tao Wang, John Wieting, Yuhuai Wu, Kelvin Xu, Yunhan Xu, Linting Xue, Pengcheng Yin, Jiahui Yu, Qiao Zhang, Steven Zheng, Ce Zheng, Weikang Zhou, Denny Zhou, Slav Petrov, and Yonghui Wu. Palm 2 technical report, 2023.

- [23] Chengrun Yang, Xuezhi Wang, Yifeng Lu, Hanxiao Liu, Quoc V. Le, Denny Zhou, and Xinyun Chen. Large language models as optimizers, 2024.
- [24] Zhuosheng Zhang, Aston Zhang, Mu Li, and Alex Smola. Automatic chain of thought prompting in large language models, 2022.
- [25] Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China, November 2019. Association for Computational Linguistics.
- [26] Yongchao Zhou, Andrei Ioan Muresanu, Ziwen Han, Keiran Paster, Silviu Pitis, Harris Chan, and Jimmy Ba. Large language models are human-level prompt engineers, 2023.
- [27] Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models, 2023.
- [28] Boshi Wang, Sewon Min, Xiang Deng, Jiaming Shen, You Wu, Luke Zettlemoyer, and Huan Sun. Towards understanding chain-of-thought prompting: An empirical study of what matters. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2717–2739, Toronto, Canada, July 2023. Association for Computational Linguistics.
- [29] Seungone Kim, Se June Joo, Doyoung Kim, Joel Jang, Seonghyeon Ye, Jamin Shin, and Minjoon Seo. The cot collection: Improving zero-shot and few-shot learning of language models via chain-of-thought fine-tuning, 2023.
- [30] Guhao Feng, Bohang Zhang, Yuntian Gu, Haotian Ye, Di He, and Liwei Wang. Towards revealing the mystery behind chain of thought: A theoretical perspective, 2023.

- [31] Aman Madaan and Amir Yazdanbakhsh. Text and patterns: For effective chain of thought, it takes two to tango, 2022.
- [32] Hamed Taherkhani, Melika Sepindband, Hung Viet Pham, Song Wang, and Hadi Hemmati. Epic: Cost-effective search-based prompt engineering of llms for code generation, 2024.
- [33] Martina Saletta and Claudio Ferretti. Exploring the prompt space of large language models through evolutionary sampling. In *Proceedings of the Genetic and Evolutionary Computation Conference*, GECCO '24, page 1345–1353, New York, NY, USA, 2024. Association for Computing Machinery.
- [34] Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7b, 2023.
- [35] Andrzej Maćkiewicz and Waldemar Ratajczak. Principal components analysis (pca). *Computers and Geosciences*, 19:303–342, 3 1993.
- [36] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(86):2579–2605, 2008.

پیوست‌ها

توضیحات تکمیلی

در پیوست در ابتدا، اعلان های دستوری که برای تنظیم کردن مدل زبانی استفاده شده اند، آورده شده اند که شامل سه بخش نمونه‌گیری روش مولد اعلان ساده، ارزیابی روش مولد اعلان ساده و اعلان های دستوری سایر روش ها می‌باشد. سپس اعلان های دستوری تولید شده توسط مولد اعلان ساده برای هر مجموعه داده آورده شده اند.

اعلان های دستوری برای نمونه‌گیری روش مولد اعلان ساده

در روش مولد اعلان ساده از سه رویکرد برای نمونه‌گیری اعلان های دستوری جدید استفاده کردیم، رویکرد اول نمونه‌گیری براساس توضیح مسئله مربوط به مجموعه داده بود. اعلان دستوری برای این رویکرد در کادر ۲-۵ آورده شده است.

اعلان دستوری برای نمونه‌گیری براساس توضیح مسئله

Given a task description, produce a detailed system prompt to guide a language model in completing the task effectively.

Guidelines:

- **Understand the Task:** Grasp the core objective, goals, and expected output of the problem as described in the problem description. Identify any implicit requirements or constraints.
- **Minimal Changes:** Since this is a zero-shot approach, use only the information available in the problem description without assuming any additional context or knowledge. Clarify instructions where needed, but avoid adding new elements unless absolutely necessary for comprehension.
- **Reasoning Before Conclusions:** Guide the model to break down the problem step by step before arriving at any conclusions. Structure the prompt to ensure that reasoning is fully explored before the final solution is given.

- Reverse the order if reasoning is provided after conclusions in any sample content. Always start with the reasoning.
- **Clarity and Conciseness:** Make sure that the prompt uses clear, specific language. The instructions should avoid unnecessary complexity or ambiguity.
- **Examples:** Since no examples are provided in a zero-shot context, ensure the problem is fully explained with placeholders for any variables or specifics that may vary.
- **Formatting:** Use markdown for readability. Present steps clearly and in order.
- **Preserve User Content:** Focus entirely on the problem description without bringing in external examples, but structure it logically.

Steps:

- . ۱ Parse the problem description.
- . ۲ Identify key variables or constraints.
- . ۳ Guide the model to explore reasoning steps (list if applicable).
- . ۴ Ensure any assumptions or logical pathways are clearly outlined.

Output Format: The output should be structured as detailed paragraphs or step-by-step instructions, depending on the problem complexity.

Notes: Edge cases: Ensure prompts remain flexible for a variety of inputs, even though no examples are provided.

در رویکرد دوم از مدل زبانی خواسته می‌شد که براساس توضیح مسئله و چند نمونه مثال به همراه جواب از مجموعه داده اقدام به تولید اعلان‌های دستوری مناسب کند، اعلان دستوری برای این امر در کادر ۲-۵ آورده شده است.

Given a problem description and two example Q&A pairs, produce a detailed system prompt to guide a language model in completing the task effectively.

Guidelines

- **Understand the Task:** Use the problem description to understand the overall goal. Supplement this understanding with the two provided examples to clarify the problem's scope.
- **Minimal Changes:** Incorporate key elements from the examples into the prompt, while maintaining the structure of the problem description. Only adjust where clarity or better instruction flow is necessary.
- **Reasoning Before Conclusions:** Guide the model to analyze the examples and reasoning patterns within the example Q&As. Ensure that prompts encourage reasoning steps before arriving at final answers.
 - Reverse the reasoning order if necessary to ensure it starts with analysis.
- **Examples:** Highlight the key learning points or steps from each of the two examples. Use placeholders [in brackets] to allow flexibility for future examples.
- **Clarity and Conciseness:** Be specific in what needs to be done, and avoid vague or generalized instructions. Ensure the combination of the problem description and examples provides enough guidance.
- **Formatting:** Use markdown for clear structure, with sections for example-based learning, reasoning, and solution paths.

- **Preserve User Content:** Include both the problem description and examples faithfully, without losing important context.

Steps

- . ۱ Analyze the problem description.
- . ۲ Examine the example Q&As for patterns in reasoning and solutions.
- . ۳ Synthesize the information to produce a prompt that mirrors the examples while remaining flexible for new problems.

Output Format

The output should be a structured set of instructions, with examples embedded for illustration. Use a mix of bullet points and paragraphs for clarity.

Examples

Provide example reasoning paths based on the given Q&A pairs.

Notes

Edge cases: Address how prompts should handle examples that deviate from common patterns found in the provided examples.

در رویکرد سوم نمونه‌گیری، از مدل زبانی خواسته شد که با الهام از اعلان‌های دستوری موفق، اقدام به تولید اعلان‌های دستوری جدید و مشابه با پرایپت‌های موفق کند، اعلان دستوری مربوط به این رویکرد در کادر ۲-۵ آورده شده است.

اعلان دستوری برای نمونه‌گیری براساس اعلان‌های دستوری موفق

Given a successful prompt, produce variations of the prompt while maintaining the original task's goals and structure.

Guidelines

- **Understand the Task:** Start by identifying the core objective of the successful prompt. Determine why it was effective in completing the

task and maintain this focus.

- **Minimal Changes:** Focus on slight variations in wording, structure, or approach to maintain effectiveness. Do not change the task's essence or main steps unless necessary for clarity.
- **Reasoning Before Conclusions:** Ensure that all variations continue to follow reasoning-first structures. If the original prompt placed conclusions before reasoning, reverse the order for variations.
- **Clarity and Conciseness:** Variations should remain clear and to the point, without introducing ambiguity or confusion.
- **Examples:** Highlight variations with slight changes in phrasing, while retaining the core elements of the original prompt.
- **Formatting:** Keep formatting consistent across variations. Use bullet points or markdown headings to segment the variations clearly.
- **Preserve User Content:** Maintain the overall flow and details of the successful prompts, making variations in small increments.

Steps

- . 1 Analyze the successful prompt to identify key elements that make it work.
- . 2 Create multiple variations by adjusting wording, step order, or clarity points.
- . 3 Ensure each variation follows the same reasoning and solution path, with slight differences in phrasing or structure.

Output Format

Output only one variation in the given prompt form, with minor changes to structure, wording, or instruction flow.

Notes

Edge cases: Test how different variations might perform across a range of inputs. Identify possible weaknesses in certain phrasing and adjust accordingly.

ارزیابی

همانطور که در فصل ۳ بخش ارزیابی توضیح داده شد، هر اعلان دستوری تولید شده روی مجموعه داده ارزیابی می‌شود و برای سوال موجود در آن مجموعه داده، پاسخی تولید می‌کند. سپس نیاز است که این پاسخ با پاسخ واقعی مقایسه شود. این مقایسه و ارزیابی توسط مدل زبانی از طریق اعلان دستوری ۲-۵ به عنوان دستورالعمل انجام می‌شود.

اعلان دستوری برای ارزیابی جواب تولید شده

By the provided information, Task:

- . ۱ Compare the Predicted Answer with the Actual Answer.
- . ۲ Determine if the Predicted Answer matches the Actual Answer exactly.
- . ۳ Output "CORRECT" if the answers match, otherwise output "INCORRECT".

Response Format:

{CORRECT or INCORRECT}

در ادامه یک سوال و جواب از مجموعه داده GSM8K به عنوان مثال و همچنین جواب تولید شده توسط بهترین اعلان دستوری برای بررسی نحوه پاسخدهی مدل زبانی آورده شده است.

Question: Henry made two stops during his 60-mile bike trip. He first stopped after 20 miles. His second stop was 15 miles before the end of the trip. How many miles did he travel between his first and second stops?

Answer: Henry traveled 20 miles + 15 miles = 20 + 15 = 35 miles, not counting the distance between stops.

Henry traveled 60 miles - 35 miles = 60 - 35 = 25 miles between his first and second stop.

Final Answer : 25

Predicted-Answer:

Analyze the Problem Description:

The problem involves finding the distance traveled between two stops during a bike trip. The initial stop was after 20 miles, and the second stop was 15 miles before the end of the trip.

Example-Based Learning:

- Identify the key components of the problem: initial stop distance, final stop distance, and total trip distance.
- Understand the order of operations: find the distance between the two stops first, then subtract this value from the total trip distance to find the distance traveled between the stops.

Reasoning:

- Find the distance between the two stops: $60 - (20 + 15) = 60 - 35 = 25$ miles.
- The distance traveled between the first and second stops is 25 miles.

Solution Paths:

- Find the distance between the two stops by subtracting the sum of the initial and final stop distances from the total trip distance.
- The distance traveled between the first and second stops is the calculated distance between the two stops.

Example Q&A Pair 1 : Ken created a care package to send to his brother, who was away at boarding school. Ken placed a box on a scale, and then he poured into the box enough jelly beans to bring the weight to 2 pounds. Then, he added enough brownies to cause the weight to triple. Next, he added another 2 pounds of jelly beans. And finally, he added enough gummy worms to double the weight once again. What was the final weight of the box of goodies, in pounds?

Example Q&A Pair 2 : James creates a media empire. He creates a movie for \$2000. Each DVD costs \$6 to make. He sells it for 2.5 times that much. He sells 500 movies a day for 5 days a week. How much profit does he make in 20 weeks?

Prompt and Example:

Prompt: Given a problem that involves finding the distance traveled between two stops during a trip, solve it by finding the distance between the two stops first, then subtracting this value from the total trip distance to find the distance traveled between the stops.

Example: Henry made two stops during his 60-mile bike trip. He first stopped after 20 miles. His second stop was 15 miles before the end of the trip. How many miles did he travel between his first and second stops?

Solution:

- Find the distance between the two stops: $60 - (20 + 15) = 60 - 35 = 25$ miles.
- The distance traveled between the first and second stops is 25 miles.

سایر روش ها

در جدول ۱، اعلان‌های دستوری به کاررفته برای هدایت مدل زبانی Mistral به‌منظور ایفای نقش به‌عنوان روش‌های زنجیره تفکر، برنامه‌ریزی و حل، برنامه‌ریزی و حل پیشرفته، مهندس اعلان خودکار و بهینه‌سازی با اعلان ارائه شده است. این اعلان‌ها برگرفته از مقاله مولد اعلان [۴] می‌باشند.

Method	Instruction Prompt
CoT	“Let’s think step by step.”
PS	“Let’s first understand the problem and devise a plan to solve the problem. Then, let’s carry out the plan and solve the problem step by step.”
PS+	“Let’s first understand the problem, extract relevant variables and their corresponding numerals, and make a plan. Then, let’s carry out the plan, calculate intermediate variables (pay attention to correct numerical calculation and commonsense), solve the problem step by step, and show the answer.”
APE	“Let’s work this out in a step by step way to be sure we have the right answer.”
OPRO	“Take a deep breath and work on this problem step-by-step.”

جدول ۱ اعلان‌های دستوری برای سایر روش‌ها جهت مقایسه نتایج



بسم الله تعالى

فرم نایید اطلاعات بولیدات علمی * مستخرج از پایان نامه دانشجویان کارشناسی ارشد

نایید عبداللهی کرمانی شماره دانشجویی: ۵۰۱۱۵۶۰۴ نام داشتکده: فنی مهندسی
خانوادگی دانشجو:

رسنده و تراییس: مهندسی کامپیوئر - هویس مصنوعی یام استاد راهنمای دکتر مهدی افتخاری

عنوان پایان نامه: خودکارسازی مهندسی اعلان: تولید اعلان‌های دستوری برای مدل های بزرگ زبانی جهت حل مسائل پردازش زبان طبیعی

دربیت	کی	Less is More: Prompt Optimization with SimplePromptBreeder	عنوان تولیدات علمی مرجع تایید کنندگان / نام مجله	مشخصات تولیدات علمی توضیحات
IJCAI2025				ارسال شده

(سروف) یک و پنجم دهم (حداد) ۱.۵ (سروف) با نمره (حداد) ۱۰۰٪ تولیدات علمی فوتو با نمره (حداد) ۱۰۰٪ تولیدات علمی فوتو با نمره (حداد) ۱۰۰٪

نام و نام خانوادگی مدبیر گروه/ رئیس بخش:
نام و نام خانوادگی نماینده هیأت داوران:

၁၃

امضيات

بازیچه

تاریخ
امضاء:

* تولیدات علمی شامل مقاله، اختراع، ساخت دستگاه، اکتشاف، ثبت اثر بدعی هنری می باشد که از پایان نامه استخراج شده باشد.

Abstract

The performance of Large Language Models relies heavily on effective prompt engineering. Hand-crafted methods such as Plan-and-Solve have improved reasoning but often remain inefficient of ensuring diversity in generated prompts, limiting effectiveness. PromptBreeder addressed this limitation by introducing a self-referential self-improvement mechanism that evolves prompts using a binary tournament genetic algorithm. This method allows PB to search the prompt space iteratively, optimizing for diversity and performance. While PB achieved significant advancements in prompt generation, it introduced a new challenge of increased computational overhead and complexity, making it less practical for many applications. To address these issues, we propose SimplePromptBreeder (SPB) that optimizes prompts through a local search strategy. It employs Determinantal Point Processes to select diverse and high-quality prompts, directly balancing performance and diversity without relying on complex self-referential mechanisms. We evaluated SPB on eight benchmarks namely MultiArith, SingleEq, AddSub, SVAMP, SQA, CSQA, AQuA-RAT, and GSM8K, demonstrating strong reasoning performance and explainability. By achieving results comparable to PromptBreeder with significantly reduced computational requirements, SPB offers a practical and efficient solution for automated prompt generation.

Keywords:Large Language Models, prompt engineering, Instruction prompt, Determinant point processes, Diversity, Quality



Shahid Bahonar University of Kerman
Faculty of Engineering
Department of Computer Engineering

**Auto prompt engineering : generate instruction prompts for
LLMs to solve NLP tasks.**

Prepared by:
Nahid Abdollahi

Supervisor:
Dr. Mahdi Eftekhari

Advisor:
Dr. Sahar Vahdati

**A Thesis Submitted as a Partial Fulfillment of the Requirements for the
Degree of Master of Science in Computer Engineering (M. Sc.)**

April 2025