

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ



دانشگاه اصفهان

دانشکده مهندسی کامپیوتر

گروه مهندسی نرم افزار

رساله کارشناسی رشته‌ی رشته‌ی مهندسی کامپیوتر گرایش هوش مصنوعی

استفاده از کاننیزرها بر اساس سریع و مدله

مدل‌ها بر زبان بزرگ

استاد راهنما:

دکتر زهرا زجاجی

دانشجو:

سهیل سلیمی

دی ۱۴۰۲

تقديم به

خودم

چکیده

مدل های زبانی بزرگ^۱ مدل هایی هستند که با استفاده از تکنیک های یادگیری عمیق بر روی داده های متنی بزرگ آموزش دیده اند و قادر به تولید متن های شبیه به انسان و انجام وظایف مختلف بر اساس ورودی ارائه شده هستند. این مدل ها می توانند برای تولید محتوای خلاق، ترجمه زبان ها، پاسخ به سوالات و انجام وظایف دیگر مورد استفاده قرار گیرند. اما اجرای این مدل ها در محیط های واقعی با چالش هایی مانند نیاز به منابع محاسباتی زیاد، حفظ حریم خصوصی و امنیت داده ها، و مسئولیت اخلاقی مواجه است. در این مقاله، ما یک روش برای استفاده از کانتینر ها برای اجرای سریع و محلی مدل های زبانی بزرگ را ارائه می دهیم. کانتینر ها امکان ایجاد و اجرای محیط های نرم افزاری مستقل و قابل حمل را فراهم می کنند. ما نشان می دهیم که چگونه می توان با استفاده از کانتینر ها، مدل های زبانی را بدون نیاز به یک سرویس ابری، بر روی دستگاه های محلی اجرا کرد. ما مزایا و چالش های این روش را بررسی می کنیم و چندین مورد کاربردی را نشان می دهیم. ما نتایج آزمایش های خود را بر روی چندین مدل زبانی معروف و چندین وظیفه زبانی ارائه می دهیم و نشان می دهیم که این روش می تواند کارایی و دقت بالایی را حفظ کند. ما همچنین چندین جهت برای کارهای آینده در این زمینه پیشنهاد می دهیم.

کلیدواژه ها: ۱- کانتینر ها و مدل های زبانی بزرگ ۲- اجرای محلی و سریع مدل های زبانی ۳- بهینه سازی و امنیت مدل های زبانی ۴- مورد کاربردی مدل های زبانی بزرگ

¹ Large language model

فهرست مطالب

صفحه	عنوان
۱	۱: مطالب اصلی
۱	۱-۱ ادبیات موضوع
۱	۱-۱-۱ کانتینر ها
۲	۲-۱-۱ کانتینرها چه مزایایی دارند؟
۳	۲-۱ روش های پیشین
۳	۱-۲-۱ استفاده از ماشین های مجازی
۳	۲-۲-۱ سرویس های ابری
۴	۳-۲-۱ استفاده مستقیم از مدل های زبانی بزرگ
۴	۳-۱ روش پیشنهادی
۴	۱-۳-۱ استفاده از کانتینر ها
۵	۲-۳-۱ مدیریت کانتینر های ساخته شده برای مدل های زبانی بزرگ
۶	منابع و مآخذ
۶	پیوست ها
۶	پ-۱ جزئیات معادله ها
۷	پ-۲ اثبات روابط ریاضی
۸	واژه نامه فارسی به انگلیسی
۱۰	واژه نامه انگلیسی به فارسی
۱۲	علایم و اختصارات

عنوان

صفحه

عنوان

صفحه

فهرست تصاویر

صفحه

عنوان

فهرست جداول

صفحه	عنوان
۴	جدول ۱-۱: هزینه استفاده از سرویس های ابری Open Ai

فهرست الگوریتم‌ها

صفحه

عنوان

فصل اول

مقدمه

۱-۱ پیش‌گفتار

مدل‌های زبانی بزرگ^۱ در سال‌های اخیر توانایی شگفت‌انگیزی در وظایف پردازش زبان طبیعی و فراتر از آن نشان داده‌اند. این موفقیت مدل‌های زبانی بزرگ‌ها منجر به ورود تعداد زیادی از پژوهش‌های علمی در این زمینه شده است. این پژوهش‌ها موضوعات متنوعی را شامل می‌شوند، از جمله نوآوری‌های معماری، راهبردهای بهتر آموزش، بهبود طول متن، تنظیم دقیق، مدل‌های زبانی بزرگ‌های چندحالتی، رباتیک، مجموعه داده‌ها، معیارهای ارزیابی، کارایی و بیشتر. با توجه به توسعه سریع روش‌ها و پیشرفت‌های مداوم در پژوهش مدل‌های زبانی بزرگ‌ها، درک تصویر کلی از پیشرفت‌ها در این راستا بسیار چالش‌برانگیز شده است.

در این مقاله، ما به بررسی چالش‌ها و راه‌حل‌های مربوط به اجرای سریع و محلی مدل‌های زبانی بزرگ‌ها می‌پردازیم. ما نشان می‌دهیم که چگونه می‌توان با استفاده از کانتینر‌ها، یک روش مدیریت بسته‌بندی و انتشار نرم‌افزار، مدل‌های زبانی بزرگ‌ها را بر روی رایانه‌های شخصی یا سرورهای خود اجرا کرد. این به معنای این است که نیازی به تکیه بر یک سرویس ابری برای استفاده از آنها نیست،

¹Large language model

که می تواند چندین مزیت داشته باشد، از جمله: حفظ حریم خصوصی و امنیت داده ها: وقتی یک مدل های زبانی بزرگ محلی را اجرا می کنید، داده های شما هرگز از دستگاه شما خارج نمی شود. این می تواند برای داده های حساس، مانند سوابق بهداشتی یا داده های مالی، مهم باشد. در دسترس بودن آفلاین: مدل های زبانی بزرگ محلی می توانند آفلاین استفاده شوند، که به این معنی است که شما می توانید از آنها حتی اگر اتصال اینترنت نداشته باشید، استفاده کنید. این می تواند برای کار بر روی پروژه های در مناطق دور افتاده یا برای برنامه هایی که نیاز به در دسترس بودن همیشگی دارند، مفید باشد. سفارشی سازی: مدل های زبانی بزرگ محلی می توانند برای وظایف یا حوزه های خاص تنظیم دقیق شوند. این می تواند آنها را دقیق تر و کارآمدتر برای وظایفی که شما نیاز دارید انجام دهند، کند. مدل های زبانی بزرگ ها می توانند بر روی انواع پلتفرم های سخت افزاری، از جمله واحد پردازش مرکزی^۲ ها و واحد پردازش گرافیکی^۳ ها اجرا شوند. با این حال، مهم است توجه داشته باشید که مدل های زبانی بزرگ محلی می توانند بسیار هزینه بر برای اجرا باشند، بنابراین شما ممکن است به یک رایانه قدرتمند برای استفاده از آنها به طور موثر نیاز داشته باشید. برای اجرای یک مدل های زبانی بزرگ محلی، شما باید نرم افزار لازم را نصب کنید و فایل های مدل را دانلود کنید. پس از انجام این کار، شما می توانید مدل را شروع کنید و از آن برای تولید متن، ترجمه زبان ها، پاسخ به سوالات و انجام وظایف دیگر استفاده کنید.

^۲CPU

^۳GPU

فصل دوم

مطالب اصلی

۱-۲ ادبیات موضوع

۱-۱-۲ کانتینرها

کانتینر سازی یک روش مدیریت بسته بندی و انتشار نرم افزار است که به شکل مجازی برنامه های کاربردی را برای استقرار، بسته بندی و ایزوله می کند^۱. کانتینرها از هسته سیستم عامل استفاده می کنند تا برنامه های کاربردی را از سخت افزار و سیستم عامل میزبان جدا کنند. این باعث می شود که برنامه ها قابل انتقال، سبک و سریع باشند^۲.

مدیریت کانتینر یک فرآیند است که به ایجاد، اجرا، مانیتورینگ، توقف و حذف کانتینرها می پردازد^۳. برای مدیریت کانتینرها، نیاز به ابزارهایی است که به عنوان ارکستراسیون کانتینر شناخته می شوند. این ابزارها به مدیریت کانتینرهای متعدد بر روی یک یا چند سرور کمک می کنند. برخی از این ابزارها عبارتند از:

- Docker: این ابزار یک پلتفرم کانتینر سازی است که به ساخت، اجرا و اشتراک گذاری کانتینرهای برنامه ای کمک می کند.

- Kubernetes: این ابزار یک سیستم ارکستراسیون کانتینر اپن سورس و رایگان است که اولین

نسخه های آن در کمپانی گوگل طراحی شد. این ابزار به مدیریت، مقیاس بندی و به روز رسانی کانتینرهای برنامه ای بر روی یک خوشه از سرورها کمک می کند.

OpenShift - این ابزار یک پلتفرم کانتینر سازی تجاری است که بر پایه داکر و کوبرنتیس ساخته شده است. این ابزار به توسعه، اجرا و مدیریت کانتینرهای برنامه ای در محیط های ابری یا محلی کمک می کند.

کانتینرهای برنامه ای و کانتینرهای سیستمی دو نوع کانتینر هستند که بر اساس نوع برنامه های کاربردی که اجرا می کنند، تفاوت دارند. کانتینرهای برنامه ای، مثل داکر، فایل ها، وابستگی ها و کتابخانه های یک برنامه را برای اجرا در یک سیستم عامل کپسوله می کنند^۲. این کانتینرها فقط یک برنامه را اجرا می کنند و نیازی به یک سیستم عامل مهمان ندارند. کانتینرهای سیستمی، مثل LXC یا LXD، یک سیستم عامل کامل را برای اجرا چندین برنامه در یک کانتینر کپسوله می کنند. این کانتینرها مانند یک ماشین مجازی عمل می کنند، اما با استفاده از هسته سیستم عامل میزبان به جای یک هایپروایزر.

۲-۱-۲ کانتینرها چه مزایایی دارند؟

برخی از مزایای کانتینرها عبارتند از:

- سرعت و کارایی: کانتینرها به دلیل حجم کم و استفاده بهینه از منابع سخت افزاری، سریع تر و کارآمدتر از ماشین های مجازی هستند. کانتینرها می توانند در چند ثانیه ایجاد، اجرا و حذف شوند، در حالی که ماشین های مجازی ممکن است چند دقیقه زمان ببرند.
- انتقال پذیری و توزیع پذیری: کانتینرها می توانند بر روی هر دستگاهی که دارای نرم افزار کانتینر سازی است، اجرا شوند. این به این معنی است که شما می توانید یک کانتینر را بر روی یک رایانه شخصی، یک سرور، یک ابر یا یک دستگاه IoT اجرا کنید. همچنین، شما می توانید کانتینرها را به راحتی بین محیط های مختلف منتقل یا توزیع کنید.
- ایزولاسیون و امنیت: کانتینرها از یکدیگر و از سیستم عامل میزبان جدا هستند. این به این معنی است که اگر یک کانتینر دچار خرابی یا حمله شود، تاثیری ندارد.

۲-۲ روش های پیشین

۱-۲-۲ استفاده از ماشین های مجازی

ماشین مجازی یک نرم افزار است که به شما اجازه می دهد یک سیستم عامل دیگر را در داخل سیستم عامل فعلی خود اجرا کنید. برای استفاده از مدل های زبانی بزرگ ها، شما نیاز دارید که یک ماشین مجازی با سیستم عامل ویندوز را نصب کنید و سپس نرم افزار مدل های زبانی بزرگ را در آن اجرا کنید. این روش دارای برخی مزایا و معایب است. برخی از مزایای استفاده از ماشین مجازی عبارتند از:

- شما می توانید از مدل های زبانی بزرگ ها بدون نیاز به خرید یک کامپیوتر ویندوز استفاده کنید.
- شما می توانید به راحتی بین سیستم عامل های مختلف جابجا شوید و فایل های خود را به اشتراک بگذارید.
- شما می توانید تنظیمات و پیکربندی های مختلف را برای ماشین مجازی خود انجام دهید و در صورت لزوم به حالت قبل بازگردانید.
- برخی از معایب استفاده از ماشین مجازی عبارتند از:
 - شما نیاز دارید که فضای حافظه و پردازنده کافی را برای اجرای ماشین مجازی فراهم کنید، در غیر این صورت سرعت و عملکرد آن کند خواهد شد.
 - شما نیاز دارید که یک نسخه قانونی از سیستم عامل ویندوز را تهیه و فعال کنید، در غیر این صورت با مشکلات قانونی و امنیتی روبرو خواهید شد.
 - شما نمی توانید از برخی قابلیت های سخت افزاری کامپیوتر خود، مانند دوربین، صدا، چاپگر و غیره، در محیط مجازی استفاده کنید، مگر اینکه درایور های مناسب را نصب کنید.

۲-۲-۲ سرویس های ابری

سرویس های ابری را می توان برای استفاده از مدل های زبانی بزرگ ها به عنوان یک راه حل مقیاس پذیر و انعطاف پذیر در نظر گرفت. با استفاده از سرویس های ابری، می توان از منابع محاسباتی و ذخیره سازی بدون نگرانی از محدودیت های سخت افزاری بهره برد. همچنین، می توان با استفاده از سرویس های ابری، مدل های زبانی بزرگ ها را به صورت خودکار و پویا مدیریت کرد و به روز رسانی کرد. با این حال، استفاده از سرویس های ابری نیز مشکلات خود را دارد. برخی از مشکلات عبارتند از:

- حفظ امنیت و حریم خصوصی داده ها و مدل های زبانی بزرگ ها در فضای ابری
- تضمین کیفیت سرویس و عملکرد مناسب مدل های زبانی بزرگ ها در شرایط نامطلوب شبکه
- هزینه بالای استفاده از سرویس های ابری برای برخی از کاربردهای مدل های زبانی بزرگ ها
- عدم وجود استانداردهای یکسان و قابل تبادل بین سرویس دهندگان مختلف ابری

جدول ۱-۲ - هزینه استفاده از سرویس های ابری Open Ai

Output	Input	Model
\$0.06/ 1K tokens	\$0.03/ 1K tokens	gpt-4
\$0.12/ 1K tokens	\$0.06/ 1K tokens	gpt-4-32k

با توجه به جدول ۱-۱ برای تولید ۱۰۰۰ توکن شما باید حدود \$ 58 پرداخت کنید.

۳-۲-۲ استفاده مستقیم از مدل های زبانی بزرگ

برای استفاده مستقیم از مدل های زبانی بزرگ نیاز مشکلات زیر را به همراه دارد

- نیاز به فرد متخصص
- عدم مقیاس پذیری
- امکان استفاده آن در سیستم عامل ها مختلف وجود ندارد

۳-۲ روش پیشنهادی ۱-۳-۲ استفاده از کانتینر ها

کانتنر ها راهی برای بسته بندی و اجرای برنامه های کامپیوتری هستند که می توانند در محیط های مختلف اجرا شوند. کانتنر ها مزایایی مانند سادگی، قابلیت حمل و نقل، امنیت و کارایی دارند.

برای انتشار مدل های زبانی بزرگ، کانتنر ها می توانند راه حل مناسبی باشند. چون:

- کانتنر ها می توانند مدل ها را به صورت جداگانه و مستقل از سخت افزار و سیستم عامل اجرا کنند. این به این معنی است که مدل ها را نیازی نیست برای هر پلتفرم یا دستگاه جدید تغییر داد یا تطبیق داد.

- کانتنر ها می توانند مدل ها را به صورت خودکار و پویا مقیاس بزرگ کنند. این به این معنی است که بر اساس نیاز و تقاضای کاربران، تعداد و منابع کانتنر ها را می توان افزایش یا کاهش داد.

- کانتنر ها می توانند مدل ها را به صورت امن و قابل اعتماد اجرا کنند. این به این معنی است که کانتنر ها محافظت شده از دسترسی های غیرمجاز یا خطای سخت افزار یا نرم افزار هستند.

برای استفاده از کانتنر ها برای انتشار مدل های زبانی بزرگ، لازم است چند قدم را طی کنیم:

- ابتدا باید یک تصویر ^۱ کانتنر را بسازید. تصویر کانتنر شامل کدهای، پکیج های، پیکربندی های و داده های لازم برای اجرای مدل است.

- سپس باید تصویر کانتنر را در یک رجیستر ^۲ آپلود کنید. رجیستر یک سرویس ذخیره سازی است که تصویر کانتنر را در دسترس قرار می دهد.

- در نهایت باید یک نمونه ^۳ از تصویر کانتنر را در یک سرویس حمل و نقل ^۴ درخواست کنید. سرویس حمل و نقل چگونگی و کجای اجرای نمونه را تعیین می کند.

ولی این مراحل دسترسی سریع را برای کاربر اینجا نمی کند. چون همیچنان کار با این نمونه ساخته شده مشکل است.

۲-۳-۲ مدیریت کانتنر های ساخته شده برای مدل های زبانی بزرگ

¹image

²registry

³instance

⁴transport

پیوست‌ها

پ-۱ جزئیات معادله‌ها

لورم ایپسوم متن ساختگی با تولید سادگی نامفهوم از صنعت چاپ و با استفاده از طراحان گرافیک است. چاپگرها و متون بلکه روزنامه و مجله در ستون و سطرآنچنان که لازم است و برای شرایط فعلی تکنولوژی مورد نیاز و کاربردهای متنوع با هدف بهبود ابزارهای کاربردی می باشد. کتابهای زیادی در شصت و سه درصد گذشته، حال و آینده شناخت فراوان جامعه و متخصصان را می طلبد تا با نرم افزارها شناخت بیشتری را برای طراحان رایانه ای علی الخصوص طراحان خلاق و فرهنگ پیشرو در زبان فارسی ایجاد کرد. در این صورت می توان امید داشت که تمام و دشواری موجود در ارائه راهکارها و شرایط سخت تایپ به پایان رسد و زمان مورد نیاز شامل حروفچینی دستاوردهای اصلی و جوابگوی سوالات پیوسته اهل دنیای موجود طراحی اساسا مورد استفاده قرار گیرد.

لورم ایپسوم متن ساختگی با تولید سادگی نامفهوم از صنعت چاپ و با استفاده از طراحان گرافیک است. چاپگرها و متون بلکه روزنامه و مجله در ستون و سطرآنچنان که لازم است و برای شرایط فعلی تکنولوژی مورد نیاز و کاربردهای متنوع با هدف بهبود ابزارهای کاربردی می باشد. کتابهای زیادی در شصت و سه درصد گذشته، حال و آینده شناخت فراوان جامعه و متخصصان را می طلبد تا با نرم افزارها شناخت بیشتری را برای طراحان رایانه ای علی الخصوص طراحان خلاق و فرهنگ پیشرو در زبان فارسی ایجاد کرد. در این صورت می توان امید داشت که تمام و دشواری موجود در ارائه راهکارها و شرایط سخت تایپ به پایان رسد و زمان مورد نیاز شامل حروفچینی دستاوردهای اصلی و جوابگوی سوالات پیوسته اهل دنیای موجود طراحی اساسا مورد استفاده قرار گیرد.

$$p(r) = C_k \frac{N}{\pi a^2} \left[1 - \left(\frac{r}{a} \right)^k \right]^{\frac{1}{k}} \quad (\text{پ-۱})$$

است.

پ-۲ اثبات روابط ریاضی

لورم ایپسوم متن ساختگی با تولید سادگی نامفهوم از صنعت چاپ و با استفاده از طراحان گرافیک است. چاپگرها و متون بلکه روزنامه و مجله در ستون و سطرآنچنان که لازم است و برای شرایط فعلی تکنولوژی مورد نیاز و کاربردهای متنوع با هدف بهبود ابزارهای کاربردی می باشد. کتابهای زیادی در شصت و سه درصد گذشته، حال و آینده شناخت فراوان جامعه و متخصصان را می طلبد تا با نرم افزارها شناخت بیشتری را برای طراحان رایانه ای علی الخصوص طراحان خلاق و فرهنگ پیشرو در زبان فارسی ایجاد کرد. در این صورت می توان امید داشت که تمام و دشواری موجود در ارائه راهکارها و شرایط سخت تایپ به پایان رسد و زمان مورد نیاز شامل حروفچینی دستاوردهای اصلی و جوابگوی سوالات پیوسته اهل دنیای موجود طراحی اساسا مورد استفاده قرار گیرد.



شکل پ-۱ - تصویر مفهومی

لورم ایپسوم متن ساختگی با تولید سادگی نامفهوم از صنعت چاپ و با استفاده از طراحان گرافیک است. چاپگرها و متون بلکه روزنامه و مجله در ستون و سطرآنچنان که لازم است و برای شرایط فعلی تکنولوژی مورد نیاز و کاربردهای متنوع با هدف بهبود ابزارهای کاربردی می باشد. کتابهای زیادی در شصت و سه درصد گذشته، حال و آینده شناخت فراوان جامعه و متخصصان را می طلبد تا با نرم افزارها شناخت بیشتری را برای طراحان رایانه ای علی الخصوص طراحان خلاق و فرهنگ پیشرو در زبان فارسی ایجاد کرد. در این صورت می توان امید داشت که تمام و دشواری موجود در ارائه راهکارها و شرایط سخت تایپ به پایان رسد و زمان مورد نیاز شامل حروفچینی دستاوردهای اصلی و جوابگوی سوالات پیوسته اهل دنیای موجود طراحی اساسا مورد استفاده قرار گیرد.

واژه‌نامه فارسی به انگلیسی

Reasoner	استدلال‌گر
Nominals	اسمی
Bug	اشکال
Debugging	اشکال‌زدایی
Assertional	اعلانی
Tableau Algorithm	الگوریتم تابلو
NP-Hard	ان‌پی-سخت
Ontology	آنتولوژی
Profile Ontology	آنتولوژی نمایه
Cohesion	انسجام
Representation	بازنمایی
Transitivity	تراگذری
Diagnosis	تشخیص
Conflict	تضاد
Single-Thread	تک-نخی
Inconsistency	تناقض
Bottom Concept	ته‌مفهوم
Recall	جامعیت
Coupling	چسبندگی
Multi-Thread	چند-نخی
Cache	حافظه پنهان
Linked Data	داده‌های پیوندی
Precision	دقت
Pinpointing	ریشه‌یابی
Sub-Class	زیرکلاس
Coherency	سازگاری
Consistency	سازگاری
Top Concept	سرمفهوم
Breadth-First	سطح-اول
Ontology Expressivity	سطح بیان آنتولوژی
Correctness	صحت
Formal	صوری
Conciseness	عدم ابهام
Completeness	کمال
Axiom	گزاره
Class Disjointness Axiom	گزاره کلاس مجزا
Suspected Axiom	گزاره مشکوک

Gold Standard	گنجینه
Variety	گونه
Hit Set	مجموعه پوشش
Specification	مشخصه‌سازی
Unsatisfiability	مصادق‌ناپذیری
F-Measure	معیار-اف
Description Logics	منطق توصیفی
Incoherency	ناسازگاری
Terminological	واژگانی
Object Property	ویژگی شی

واژه‌نامه انگلیسی به فارسی

Assertional	اعلانی
Axiom	گزاره
Bottom Concept	ته‌مفهوم
Breadth-First	سطح-اول
Bug	اشکال
Cache	حافظه پنهان
Class Disjointness Axiom	گزاره کلاس مجزا
Coherency	سازگاری
Cohesion	انسجام
Completeness	کمال
Conciseness	عدم ابهام
Conflict	تضاد
Consistency	سازگاری
Correctness	صحت
Coupling	چسبندگی
Debugging	اشکال‌زدایی
Description Logics	منطق توصیفی
Diagnosis	تشخیص
F-Measure	معیار-اف
Formal	صوری
Gold Standard	گنجینه
Hit Set	مجموعه پوشش
Incoherency	ناسازگاری
Inconsistency	تناقض
Linked Data	داده‌های پیوندی
Multi-Thread	چند-نخی
Nominals	اسمی
NP-Hard	ان‌پی-سخت
Object Property	ویژگی شی
Ontology Expressivity	سطح بیان آنتولوژی
Ontology	آنتولوژی
Pinpointing	ریشه‌یابی
Precision	دقت
Profile Ontology	آنتولوژی نمایه
Reasoner	استدلال‌گر
Recall	جامعیت
Representation	بازنمایی

Single-Thread	تک-نخی
Specification	مشخصه‌سازی
Sub-Class	زیر کلاس
Suspected Axiom	گزاره مشکوک
Tableau Algorithm	الگوریتم تابلو
Terminological	واژگانی
Top Concept	سرمفهوم
Transitivity	تراگذری
Unsatisfiability	مصادق‌ناپذیری
Variety	گونه

علايم و اختصارات

A-Box	Assertional Box
ACL	Attributive Concept Language with Complements
AML	Agreement Maker Light
DL	Description Logics
EDAM	EMBRACE Data And Methods
HST	Hit Set Tree
LOV	Lined Open Vocabulary
MUPS	Minimum Unsatisfiability Preserving Set
OAEI	Ontology Alignment Evaluation Initiative
ORE	Ontology Repair and Enrichment
RaDON	Repair and Diagnosis in Ontology Networks
RDF	Resource Description Framework
RIO	Regularity Inspector for Ontologies
SPARQL	SPARQL Protocol and RDF Query Language
T-Box	Terminological Box
W3C	World Wide Web Consortium
XML	Extensible Markup Language

Abstract

Large-scale linguistic models (مدل های بزرگ) are models that are trained using deep learning techniques on large textual data and are capable of producing human-like text and performing various tasks based on the input provided. These models can be used to generate creative content, translate languages, answer questions and perform other tasks. But the implementation of these models in real environments faces challenges such as the need for large computing resources, data privacy and security, and ethical responsibility. In this paper, we present a method for using containers to run مدل های بزرگ quickly and locally. Containers provide the possibility to create and run independent and portable software environments. We show how to use containers to run language models on local machines without the need for a cloud service. We review the advantages and challenges of this method and show several practical cases. We present the results of our experiments on several well-known language models and several linguistic tasks, and show that this method can maintain high efficiency and accuracy. We also suggest several directions for future work in this area.

Keywords: 1- First Keyword, 2- Second Keyword, 3- Third Keyword, 4- Fourth Keyword, 5- Fifth Keyword

سال



University of Isfahan
Faculty of Computer Engineering
Department of Software Engineering

Ph.D. Thesis

Thesis English Title Supervisor:

Dr. Supervisor First and Last Name

By:

Student First and Last Name

Month Year