استفاده از کانتینر ها برای اجرای سریع و محلی مدل های زبانی بزرگ

سهیل سلیمی - استاد زجاجی

see live at soheilsalimidev.github.io/nosql-presentation

1. استفاده از کانتینر ها برای اجرای سریع و محلی مدل های زبانی بزرگ
2. ادبیات موضوع
3. بیان مسله
    1. اما مدل های رایگان هم وجود دارد!
    2. هدف های این پژوه
4. ضرورت انجام مساله
5. پروژه های پیشن
    1. مشکلات ollma
6. کاربرد پژوهش
7. روش پیشنهادی
    1. روش پیشنهادی
    2. مزیت های این روش

ادبیات موضوع

- **مدل زبان** آماری یک توزیع احتمال روی دنباله‌ی کلمات است.
- مدل **زبانی بزرگ** یا به اختصار ال‌ال‌ام (به انگلیسی: LLM)، یک مدل زبانی متشکل از یک شبکه عصبی با پارامترهای زیادی است که بر روی مقادیر زیادی متن بدون برچسب با استفاده از یادگیری خود نظارتی یا یادگیری نیمه‌نظارتی آموزش داده شده‌است.[1]
- **کانتینر** یک واحد نرم‌افزاری است که شامل یک برنامه و تمام وابستگی‌های آن است. کانتینرها می‌توانند بر روی هر سیستم عاملی که داکر را اجرا می‌کند، به صورت مستقل و یکنواخت اجرا شوند.

بیان مسله

از انجا که راه اندازی و استفاده از مدل های زبانی یزرگ کاری زمان بر است و به همین علت معمولا مجبور به استفاده از سرویس هایی هستم که این مدل ها را در اختیار ما قرار می دهدند.

قیمت های سرویس های مدل های زبانی

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

به زبان دیگر شما برای پردازش 130,000 کلمه نیاز به پرداخت حدودا $58.50 هست

اما مدل های رایگان هم وجود دارد!

- **Llama 2**, The most popular(free) model for general use.
- **codellama**, A large language model that can use text prompts to generate and discuss code.
- **mistral**, Mistral is a 7.3B parameter model, distributed with the Apache license. It is available in both instruct (instruction following) and text completion.

هدف های این پرژه

- نصب و استفاده راحت از مدل های زبانی بزرگ
- کاهش هزینه‌های محاسباتی و انتقال داده‌ها با استفاده از منابع محلی یا شبکه‌های خصوصی
- افزایش امنیت و حفظ حریم خصوصی با جلوگیری از ارسال داده‌ها به سرورهای ابری یا سرویس‌های آنلاین
- افزایش کنترل و انعطاف‌پذیری با امکان تغییر و بهینه‌سازی مدل‌ها بر اساس نیازها و شرایط

ضرورت انجام مساله

با پیشرفت و توسعه مدل های زبان و افزایش توانایی ان ها در حل مشکلات خیلی از اپ و شرکت ها می خواهند از این مدل ها استفاده کنند. ولی همانطور که گفته شد استفاده از این مدل ها کار ساده ای نیست یا با هزینه زیادی باید این کار را انجام داد.

پروژه های پیشن

Ollma

Get up and running with large language models locally

مشکلات ollma

- به علت نوع طراحی در بعضی اوقات ممکن است وابستگی های مدل با یک دیگر به تداخل بخورد
- تمرکز روی مدل های Llama
- نداشتن gui برای ارتباط با مدل
- کند بودن

کاربرد پژوهش

افراد و شرکت ها به راحتی می توانند از این مدل استفاده کنند, بدون نیاز به اینکه وارد جزئیات پیاده سازی این مدل ها شوند.

درآمد از پروژه

- ارائه خدمات تولید محتوا با استفاده از مدل‌های زبانی گسترده
- ارائه خدمات بهینه‌سازی و تغییر مدل‌های زبانی گسترده
- ذخیره و نگه داری مدل های گسترده کاربران

روش پیشنهادی

- کانتینر کردن این LLM که این کار را طبق [2] Open Container Initiative انجام می دهیم.
- اتصال LLM به یک اجراکننده کانتینر مانند youki
- ساخت یک فایل تنظیمات برای تنظیم وابستگی های مورد نیاز مدل
- تنظیم پروتوکل ها و تنظیمات مربوط به [3] HTTP و [4] gRPC

روش پیشنهادی

برنامه ما یک کانتینر از نوع youki را با استفاده از یک LLM و ورودی‌های مربوطه ایجاد می‌کند و مطمین که بخشی از برنامه ما قابل اتصال به این LLM است. که این برنامه ما روی پورت‌هایی گوش می‌دهد و امکان دسترسی به LLM را فراهم می‌کند. این پورت‌ها شامل دو نوع هستند: یکی برای وب که کاربران می‌توانند با یک رابط گرافیکی ساده با آن ارتباط برقرار کنند و یکی برای gRPC که اپلیکیشن‌هایی که نیاز به استفاده از LLM در شبکه داخلی خود دارند، از آن استفاده می‌کنند. با این روش، کاربران می‌توانند فقط با دانلود این کانتینر، به راحتی از LLM بهره ببرند، بدون اینکه نیاز به نصب وابستگی‌ها و تنظیمات پیچیده خاصی داشته باشند.

مزیت های این روش

- به علت کانتینری بود همه جا و سریع قابل اجرا هستند
- نیاز به پرداخت هزینه ای برای استفاده از انها نیست
- حفظ امنیت داده های کاربر
- رسیدن به یک API واحد برای مدل های زبانی

- \[1\] Goled, Shraddha (May 7, 2021). "Self-Supervised Learning Vs Semi-Supervised Learning: How They Differ". Analytics India Magazine
- \[2\] O. Initiative, Open container initiatives. 2020.
- \[3\] D. Gourley and B. Totty, HTTP: the definitive guide. “ O’Reilly Media, Inc.,” 2002.
- \[4\] X. Wang, H. Zhao, and J. Zhu, “GRPC: A communication cooperation mechanism in distributed systems,” ACM SIGOPS Operating Systems Review, vol. 27, no. 3, pp. 75–86, 1993.

Thank You

Hope you have good day