DocProcessor

محتوا

تبدیل مستندات به نقشه ویژگی‌های قابل‌راستی‌آزمایی برای خودکارسازی تضمین کیفیت

خلاصه

DocProcessor یک ماژول مستقل و کاملاً جدا از Go است که مستندات پروژه‌را بارگذاری می‌کند، نقشه‌های ساختاریافته ویژگی‌ها را می‌سازد و پوشش‌راستی‌آزمایی‌را ردیابی می‌کند. این ماژول جهت استخراج هوشمند ویژگی‌ها طراحی شده است، اما شامل استخراج LLM برای کار با عامل‌های اکتشافی برای استفاده آفلاین نیز می‌شود.

توضیح کوتاه

ماژول Go مستقل از پروژه برای پردازش مستندات و استخراج نقشه ویژگی‌ها. این ماژول مستندات‌را به نقشه‌های ساختاریافته ویژگی‌ها تجزیه می‌کند و پیگیری می‌کند که کدام ویژگی‌ها راستی‌آزمایی شده‌اند — برای استخراج هوشمند یا اکتشافات آفلاین — و تضمین می‌کند که LLM با استفاده از عامل‌های خودکارسازی تضمین کیفیت همواره با واقعیت مطابقت دارد و هیچ‌گونه اغراق یا نادرستی در آن‌راه ندارد.

توضیح بلند

هر تیم نرم‌افزاری با یک دروغ آهسته و همیشگی زندگی می‌کند: مستندات ویژگی‌هایی‌را وعده می‌دهد، تست‌ها چیزی نزدیک به آن‌را پوشش می‌دهند و هیچ‌کس نمی‌تواند با اطمینان بگوید که آیا این دو یک محصول‌را توصیف می‌کنند یا نه. DocProcessor برای آشکار و قابل اندازه‌گیری کردن این شکاف وجود دارد. با در اختیار داشتن مستندات یک پروژه، این ماژول یک نقشه ساختاریافته ویژگی‌ها می‌سازد — مدلی شمارش‌شده و قابل خواندن توسط ماشین از تمام آنچه محصول ادعا می‌کند انجام می‌دهد — و پوشش‌راستی‌آزمایی‌را در برابر آن ردیابی می‌کند، به طوری که پرسش «آیا این ویژگی مستندشده واقعاً اثبات شده است؟» دیگر به بحث‌های‌راهرویی ختم نشود و به پرسشی با پاسخ مشخص تبدیل شود.

این ماژول عمداً دوحالته طراحی شده است: زمانی که عامل LLM در دسترس باشد، از آن برای استخراج هوشمند و معنایی ویژگی‌ها استفاده می‌کند و در غیر این صورت به یک تجزیه‌گر اکتشافی برای استفاده کاملاً آفلاین بازمی‌گردد. به این ترتیب، هرگز به وجود یک مدل وابسته نمی‌شود و به یکسان در بدون اتصال به شبکه یا روی لپ‌تاپ توسعه‌دهنده‌ای که در حالت پرواز کار می‌کند، اجرا CI یک وظیفه می‌شود.

Go است از نظر معماری، این ماژول یک ماژول مستقل، ناآشنا به پروژه و کاملاً جدا از هیچ مقدار خاص پروژه‌ای‌را ارسال نمی‌کند و توسط مصرف‌کنندگان به عنوان (CONST-051(B)): یک زیرماژول هم‌سطح با کد اصلی گنجانده می‌شود، بنابراین هر پروژه‌ای می‌تواند آن‌را بدون به ارث

بردن فرضیات دیگران به کار گیرد. همچنین خودرا ملزم به رعایت همان استانداردی می‌کند که بر و قوانین پوشش (CONST-035) دیگران تحمیل می‌کند — ادعاهای خودرا تحت پیمان ضداغراق README قرار می‌دهد، به این معنا که هر قابلیتی که فایل (CONST-048) خودکارسازی کامل آن تبلیغ می‌کند، توسط یک تست خودکار یا اسکریپت چالش مورد آزمایش قرار می‌گیرد تا رفتار واقعی و قابل استفاده توسط کاربر نهایی‌را تأیید کند، نه صرفاً خروجی صفر؛ رشته‌های قابل مشاهده توسط کاربر هدایت می‌شوند. هدف از همه این‌ها ایجاد یک چرخه i18n CONST-046 از طریق درزگاه مترجم آن‌را HelixQA سمت ورودی چرخه تضمین کیفیت است که DocProcessor :بسته است هر ویژگی HelixQA ،تکمیل می‌کند — این ماژول نقشه ویژگی‌هارا از مستندات استخراج می‌کند نقشه‌برداری‌شده‌را با شواهد اجرایی ضبط‌شده اثبات می‌کند و مستندات، تست‌ها و رفتار تحویل‌شده مجبور می‌شوند به جای آنکه به آرامی در هر نسخه از هم دور شوند، به هم نزدیک شوند.

چرا آن‌را ساختیم

مستندات و تست‌ها از هم دور می‌شوند: مستندات ویژگی‌هایی‌را وعده می‌دهند که هیچ تستی آن‌هارا اثبات نمی‌کند و تضمین کیفیت نمی‌تواند به سادگی تشخیص دهد که «کامل» به چه معناست. مستندات‌را به نقشه ویژگی‌های قابل خواندن توسط ماشین تبدیل می‌کند تا DocProcessor پوشش‌راستی‌آزمایی بر اساس آنچه واقعاً وعده داده شده است، اندازه‌گیری شود.

محتوا

چرا یک تغییردهندهٔ بازی است

این ابزار، سؤالی مبهم در تحویل نرم‌افزار — «آیا آنچه تحویل داده‌ایم با آنچه گفته‌ایم تحویل داده‌ایم مطابقت دارد؟» —را به چیزی تبدیل می‌کند که قابل خودکارسازی و بررسی مستمر است، و این کاررا زمانی که مدل در دسترس باشد، و LLM انجام می‌دهد: استخراج AI بدون وابستگی سخت به روش‌های ابتکاری در صورت عدم دسترسی، به‌گونه‌ای که تضمین یکسانی در همهٔ محیط‌ها از یک اجراگر آفلاین تا یک خط لولهٔ کاملاً خودکار برقرار باشد.

نوآوری‌ها

- استخراج نگاشت مستندات به ویژگی‌ها با ردیابی پوشش تأیید.
- یا ابتکاری/آفلاین LLM استخراج دوگانه: مبتنی بر عامل.
- جداسازی بدون نیاز به پیکربندی و مستقل از پروژه (CONST-051(B)).
- با تست‌ها/چالش‌ها پشتیبانی می‌شوند README خودتأیید ضدفریب: ادعاهای فایل (CONST-035/048).

چالش‌ها وراه‌حل‌ها

- با استفاده از یک استخراج‌گر ابتکاری به‌عنوان جایگزین حل شده تا **:عملکرد بدون نیاز به مدل** ماژول به‌صورت آفلاین کار کند.
- با نگاشت ویژگی‌های ساختاریافته و ردیابی پوشش **:هماهنگ نگه‌داشتن مستندات وواقعیت** تأیید حل شده که در چرخهٔ تضمین کیفیت ادغام شده است.
- از طریق جداسازی سخت‌گیرانه و مصرف زیرماژول‌ها در یک کدبیس برابر **:قابلیت استفادهٔ مجدد** حل شده است.

- **اعتبار ادعاهای خود ابزار:** با تست‌ها/چالش‌های ضدفریب برای هر قابلیت تبلیغ‌شده حل شده است.

پشتهٔ فناوری (چرایی و چگونگی)

- **Go (نسخهٔ ۱٫۲۵ به بالا)** — هستهٔ ماژول؛ تحت مجوز آپاچی ۲٫۰.
- **عوامل LLM** — استخراج هوشمند ویژگی‌های معنایی (اختیاری).
- **تجزیه‌گر ابتکاری** — جایگزین استخراج ویژگی‌ها در حالت آفلاین.
- **مترجم چندزبانه (`pkg/i18n`)** — رشته‌های محلی‌سازی‌شدهٔ CONST-046.
- **چارچوب چالش** — تأیید ضدفریب ادعاهای خود ماژول.