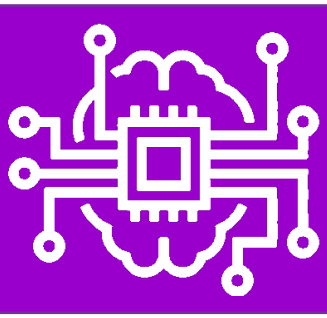

عنوان مقاله: RAG (Retrieval-Augmented Generation) چیست و چگونه آن را دستیار AI می دانیم؟

نویسنده مقاله: تیم فنی نیک‌آموز

تاریخ انتشار: ۱۰ خرداد ۱۴۰۳

منبع: https://nikamooz.com/what-is-retrieval-augmented-generation/

RAG رویکردی در توسعه هوش مصنوعی است که به مدل‌های زبانی بزرگ (LLM) کمک می‌کند تا به سؤالات تخصصی کاربران، با سرعت و دقت بالایی پاسخ دهد. فرآیند کارکرد RAG جالب توجه و جذاب است که در این مقاله، به‌سراغ تشریح آن خواهیم رفت. علاوه‌بر ارائه پاسخ جزئی‌تر به سؤال Retrieval-Augmented Generation چیست؟، مباحث دیگری همچون تاریخچه، نحوه کارکرد، مزایا و معایب و آینده RAG را نیز بررسی خواهیم کرد تا درنهایت بتوانیم نحوه کارکرد Generative AI و آینده آن را بهتر درک کنیم.

RAG (Retrieval-Augmented Generation) چیست؟

مدل‌های زبانی بزرگ ((Large language models (LLMs)، که هوش مصنوعی برای شناسایی و تولید متن است، این روزها درحال تبدیل‌شدن به ستون فقرات اکثر سازمان‌ها هستند. آن‌ها می‌توانند طیف گسترده‌ای از سؤالات انسان‌ها را پاسخ دهند؛ اما در ارائه جواب‌های صحیح و قابل استناد، ضعف دارند. استفاده از رویکرد RAG برای پوشش این ضعف به‌وجود آمد.

درواقع RAG (Retrieval-Augmented Generation) شیوه‌ای است که اطلاعات فعلی یا مرتبط با موضوع را از یک پایگاه داده خارجی استخراج کرده و در اختیار هوش مصنوعی مبتنی‌بر LLM قرار می‌دهد؛ درست زمانی که کاربر دستور خاصی به AI می‌دهد و مدل باید پاسخ را ایجاد کند. پس از دریافت درخواست کاربر توسط مدل، RAG وارد عمل می‌شود و اطلاعات را از منبعی موثق که در یک پایگاه داده برداری (Vector Database) ذخیره شده‌اند، بازیابی می‌کند. درنهایت، هر پاسخ با معنا و مفهومی نسبتاً صحیح بازیابی می‌شود و احتمال خطای مدل کاهش می‌یابد.

تاریخچه و توسعه RAG

ریشه‌های RAG (تولید مبتنی‌بر بازیابی) به اوایل دهه ۱۹۷۰ بازمی‌گردد که در آن زمان، سیستم‌های پاسخ‌گویی به پرسش توسعه یافتند. عبارت Retrieval-Augmented Generation برای اولین‌بار در مقاله Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks توسط محققان فیس‌بوک، دانشگاه College London و دانشگاه نیویورک معرفی شد. نویسنده اصلی این مقاله Patrick Lewis بود و در این تحقیق نشان داد که این شیوه می‌تواند در سناریوهای مختلف با مزایای مستقیم برای جامعه، مانند پاسخ‌گویی دقیق به سؤالات پزشکی، کاربرد داشته باشد.

توسعه RAG در اواسط دهه ۱۹۹۰ با سرویس Ask Jeeves آغاز شد. در سال ۲۰۱۱، هوش مصنوعی واتسون IBM با استفاده از RAG در مسابقه Jeopardy پیروز شد و شهرت یافت. امروزه شرکت‌های بزرگی مانند گوگل، مایکروسافت، OpenAI و آمازون از این رویکرد استفاده می‌کنند و تقاضا برای آن نیز درحال افزایش است.

RAG چگونه کار می کند؟

RAG (Retrieval-Augmented Generation) شامل سه جزء حیاتی است که در ادامه آن‌ها را نام برده و توضیح خواهیم داد:

- **بازیابی (Retrieval)**

این مؤلفه به مدل کمک می‌کند تا اطلاعات مربوطه را از پایگاه دانش خارجی، مانند یک پایگاه داده برداری، برای هر درخواست کاربر دریافت کند. Retrieval نقش بسیار حیاتی را در پاسخ‌دهی AI ایفا می‌کند؛ زیرا اولین گام در تنظیم پاسخ‌های معنی‌دار و درست است.

- **افزایش (Augmentation)**

این بخش شامل بهبود و افزودن زمینه مرتبط بیشتر به پاسخ بازیابی‌شده است. هدف از Augmentation این است که پاسخ معنادارتر و درست‌تری توسط AI به درخواست کاربر ارائه شود.

- **تولید (Generation)**

درنهایت، یک خروجی نهایی با کمک مدل LLM به کاربر ارائه می‌شود. LLM از دانش خود و زمینه ارائه‌شده استفاده می‌کند تا پاسخی مناسب و دقیق به درخواست کاربر ارائه دهد.

کاربردهای RAG (Retrieval-Augmented Generation) چیست؟

اساسی‌ترین مورد از کاربردهای RAG، استفاده در برنامه‌های پشتیبانی از مشتریان است. در این برنامه‌ها، داده‌های مشتریان در پایگاه داده برداری ذخیره می‌شوند. هنگامی که اپراتور درخواست خود را به ماشین می‌دهد، مناسب‌ترین پاسخی که مرتبط با تاریخچه مشتری و محصولات یا خدمات سازمان است را می‌گیرد. در این میان، هیچ پاسخ عمومی توسط ماشین ارائه نمی‌شود. Retrieval-Augmented Generation می‌تواند در سناریوهایی که به پاسخ‌های دقیق و آگاهانه نیاز است، بسیار مفید باشد؛ ازجمله:

- سیستم‌های پاسخ‌گویی به سؤالات عمومی و فنی کاربران
- تولیدمحتوا متنی و کمک به نویسندگان و محتوانویسان فضای آنلاین
- تحقیقات پزشکی
- مترجم زبان
- ابزارهای صنعت آموزش
- پاسخ به سؤالاتی با گستره وسیع و زمینه‌های متنوع
- چت‌بات‌هایی با نیاز به ارائه پاسخ‌های پویا
- موقعیت‌هایی که با داده‌های جدید بسیاری سروکار دارند

مزایا و معایب Retrieval-Augmented Generation

اگر بخواهیم پاسخ «RAG چیست؟» را بهتر درک کنیم، به تشریح مزایا و معایب آن نیاز داریم که در ادامه، آن‌ها را کامل بررسی می‌کنیم:

مزایای RAG

کلیدی‌ترین مزیت RAG این است که به مدل اجازه می‌دهد تا اطلاعات صحیح را به‌صورت بلادرنگ از منابع خارجی دریافت کند. درنتیجه، مدل می‌تواند پاسخی جدیدتر و سازگارتر با درخواست کاربر ارائه دهد. این موضوع زمانی اهمیت زیادی پیدا می‌کند که هوش مصنوعی نیاز به ارجاع به جزئیات خاصی دارد؛ جزئیاتی که هنگام آموزش مدل (Model Training) ارائه نشده بود. موقعیت‌هایی مانند بررسی حقایق جدید یا پاسخ‌دادن به سؤالات درباره رویدادهای اخیر، ازجمله مهم‌ترین مزیت RAG هستند.

RAG مزایای باورنکردنی دیگری نیز دارد. پس چند مورد بسیار جذاب و قابل‌توجه را در ادامه با شما به‌اشتراک خواهیم گذاشت.

- **مقیاس‌پذیری**

رویکرد RAG به مهندسان هوش مصنوعی اجازه می‌دهد تا با کمک به‌روزرسانی یا افزودن داده‌های خارجی / سفارشی به پایگاه داده برداری، دانش و طیف پاسخ‌گویی مدل‌ها را بزرگ‌تر کنند.

- **حافظه کارآمد**

مدل‌های سنتی، مانند GPT، محدودیت‌هایی برای جمع‌آوری اطلاعات تازه و به‌روز داشته و عملکرد بهینه‌ای در استفاده از حافظه خود ندارند؛ اما RAG از پایگاه داده‌های خارجی استفاده می‌کند که به مدل اجازه می‌دهد تا درصورت نیاز، با سرعت بالا، اطلاعات تازه، به‌روز و دقیق را دریافت کند.

- **انعطاف‌پذیری**

با به‌روزرسانی یا گسترش اطلاعات در منبع دانش خارجی، می‌توان RAG را برای ایجاد هر برنامه هوش مصنوعی با انعطاف‌پذیری بالا تطبیق داد.

- **سادگی در پیاده‌سازی**

آقای لوئیس و سه نفر از نویسندگان در وب‌سایت متا، در **مقاله Retrieval-Augmented Generation چیست**، می‌گویند: توسعه‌دهندگان می‌توانند این فرآیند را با حداقل پنج خط کد پیاده‌سازی کنند؛ بنابراین، می‌توان با سرعت بیشتر و هزینه کمتر نسبت به آموزش مجدد یک مدل، منابع جدید را کشف و بازیابی کرد.

معایب RAG

- بدون داده‌های باکیفیت، صحت و دقت خروجی ممکن است آسیب ببیند.
- ایجاد یک پایگاه دانش مطلوب، نیازمند زمان و سازمان‌دهی قابل‌توجهی است.
- وجود تعصب‌ها یا سوگیری‌های انسانی در داده‌های آموزشی می‌تواند بر خروجی‌ها تأثیر منفی بگذارد.
- خطر توهم مدل حتی با بهبود دقت در RAG می‌تواند رخ دهد.

چالش ها و محدودیت های RAG

درحالی‌که RAG یک رویکرد بسیار مفید برای توسعه هوش مصنوعی است، اما نقص‌هایی نیز دارد. شاید بزرگ‌ترین چالش این باشد که توسعه‌دهنده باید یک پایگاه دانش گسترده از محتوای با کیفیت بالا برای مرجع ایجاد کند.

به همین ترتیب، توسعه‌دهندگان باید وجود هرگونه سوگیری یا پیش‌داوری پایگاه دانش را درنظر داشته باشند. درنهایت، باید بگوییم که Retrieval-Augmented Generation نمی‌تواند خطرات هذیان‌گویی مدل را به‌طور کامل ازبین ببرد. محدودیت‌های RAG نیز نکته مهمی است که باید نسبت‌به آن‌ها، به‌شکلی محتاطانه عمل کنیم. این محدودیت‌ها را در لیست زیر ارائه داده‌ایم:

- امکان ارائه پاسخ‌های نه‌چندان مرتبط با درخواست کاربر
- عدم درک اطلاعات بازیابی‌شده
- مشکلات در مرحله بازیابی، مانند واکشی نادرست یا کم‌عمق اطلاعات
- شکاف بین داده‌های خام و درک متن به‌دلیل ضعف در مرحله افزایش (Augmentation)

این یک فرآیند دشوار است؛ زیرا داده‌ها باید به‌دقت تنظیم شوند. اگر کیفیت داده‌های ورودی پایین باشد، این امر بر دقت و قابلیت اطمینان خروجی، تأثیر منفی می‌گذارد.

مقایسه RAG (Retrieval-Augmented Generation) با سایر روش های تولید متن

در این بخش به مقایسه RAG با سایر روش‌های تولید متن خواهیم پرداخت. ابتدا هرکدام از روش‌ها را نام برده و توضیح می‌دهیم، سپس به مقایسه هر روش با Retrieval-Augmented Generation خواهیم پرداخت.

مقایسه RAG با N-gram و CRFs

این دو رویکرد در دسته مدل‌های آماری ((Statistical Language Models (SLMs) جا می‌گیرند. تلاش این دو گروه آن است که با کمک‌گرفتن از مجموعه داده‌های متنی بزرگ، الگوها و ساختارهای زبان انسانی را شناسایی کنند.

RAG ازنظر تولید خروجی با دو شیوه N-gram و CRFs تفاوت دارد؛ زیرا از پایگاه دانش و افزودن اطلاعات دقیق برای ارائه خروجی استفاده می‌کند، اما N-gram و CRFs الگوها و ساختارهای زبان انسانی را با کمک مجموعه داده‌های متنی بزرگ شناسایی می‌کنند.

مقایسه RAG با LSTMs و RNNs

Long Short-Term Memory Networks (LSTMs) و Recurrent Neural Networks (RNNs) در گروه شبکه‌های عصبی (Neural Networks) قرار دارند. این دو شیوه از شبکه‌های عصبی مصنوعی برای شناسایی الگوهای داده استفاده می‌کنند.

تفاوت RAG با این دو رویکرد این است که الگوهای داده را شناسایی نمی‌کند؛ بلکه تلاش دارد محتوای دقیق و به‌روزی را به مدل بازگرداند.

مقایسه RAG با GPT و BERT

Generative Pretrained Transformer (GPT) و Bidirectional Encoder Representations from Transformers (BERT) اعضای خانواده Transformer-based Models هستند. مدل‌های مبتنی‌بر ترانسفورماتور می‌توانند در تولید متن خلاقانه و متنوع، مؤثر باشند؛ زیرا قادر به کشف الگوها و ساختارهای پیچیده در داده‌های آموزشی خودشان هستند .

در مقایسه RAG با GPT و BERT، باید بگوییم که این مدل می‌تواند دقت و قابل اتکا بودن اطلاعات مدل‌های مبتنی‌بر ترانسفورماتور را افزایش دهد.

ابزارها و پلتفرم های موجود برای پیاده سازی RAG

ابزارها و پلتفرم های موجود برای پیاده سازی RAG به سه گروه تقسیم می‌شوند:

۱. دسته اول LLM هایی را پوشش می‌دهد که از RAG ، برای بهبود دقت و کیفیت خروجی خود استفاده می‌کنند.
۲. گروه دوم به کتابخانه‌ها و چهارچوب‌های RAG اشاره دارد که می‌توانند برای LLM ها اعمال شوند.
۳. دسته نهایی نیز شامل مدل‌ها و کتابخانه‌های یکپارچه‌سازی است که می‌توانند با یکدیگر ادغام شده و یا با LLM ترکیب شوند تا مدل‌های RAG بهینه‌تری را بسازند.

ابزارها و پلتفرم های موجود برای پیاده سازی RAG در LLM

- **Azure machine learning:** یادگیری ماشینی Azure که با استفاده از ابزار Azure AI Studio یا نوشتن کدهای Azure ، RAG را در هوش مصنوعی می‌گنجاند.
- **افزونه ChatGPT Retrieval:** کمپانی OpenAI یک افزونه بازیابی را برای ChatGPT ارائه داده است. با کمک این افزونه، می‌توان پاسخ‌های بهبودیافته را از این چت‌بات دریافت کرد.
- **افزونه HuggingFace Transformer:** افزونه‌ای که یک ترانسفورماتور برای تولید مدل‌های RAG ارائه می‌دهد.
- **IBM Watsonx.ai:** این مدل می‌تواند الگوی RAG را برای تولید خروجی دقیق و واقعی اجرا کند.
- **هوش مصنوعی متا:** این سیستم برای کارهایی طراحی شده است که هم نیاز به بازیابی اطلاعات از یک مجموعه بزرگ و هم ایجاد پاسخ‌های منسجم دارد.

کتابخانه ها و چهارچوب های RAG

- **FARM:** یک چهارچوب داخلی از کمپانی Deepset برای ایجاد زیرساخت NLP مبتنی‌بر ترانسفورماتور ازجمله RAG.
- **Haystack:** چهارچوب RAG End-to-End که توسط Deepset برای جستجوی اسناد ارائه شده است.
- **REALM:** یک جعبه ساخته‌شده توسط گوگل برای پاسخ‌گویی به سؤالات گسترده و دامنه‌باز (Open-Domain) با RAG.

کتابخانه های یکپارچه سازی برای RAG

این کتابخانه‌ها اجزای ماژولار و زنجیره‌های از پیش پیکربندی‌شده را برای برآورده‌کردن الزامات برنامه‌های کاربردی خاص در حین سفارشی‌سازی مدل‌ها فراهم می‌کنند. کاربران می‌توانند این فریم‌ورک‌ها را با پایگاه داده برداری ترکیب کنند تا از RAG در LLM های خود بهره ببرند.

آینده RAG (Retrieval-Augmented Generation) چگونه است؟

سرمایه‌گذاری شرکت‌های بزرگ روی مدل‌های زبانی LLM نشان می‌دهد که آینده RAG ، درخشان و روبه‌پیشرفت خواهد بود. همچنین افزایش داده‌ها و تسلط دانشمندان داده به حوزه Big Data، این امید را به‌وجود می‌آورد که پاسخ‌دهی هوش مصنوعی به سؤالات تخصصی، با نتایج مطلوب‌تری همراه خواهد بود.

تقریباً هر کسب‌وکاری می‌تواند دستورالعمل‌های فنی یا خط‌مشی، ویدئوها یا گزارش‌های خود را به منابعی به نام پایگاه‌های دانش (Knowledge Bases) تبدیل کند. این اتفاق، قدرت و توانایی LLM ها را به‌شکل چشم‌گیری افزایش خواهد داد. این منابع می‌توانند کاربردهای RAG در زمینه‌های تخصصی مانند پشتیبانی از مشتریان، آموزش کارمندان و بهره‌وری توسعه‌دهندگان را تثبیت کنند. دلیل دیگر ما برای امیدواری نسبت‌به آینده RAG، استفاده کمپانی‌های بزرگ مانند AWS ، IBM Glean، Google، Microsoft، NVIDIA، Oracleو Pinecone از این رویکرد است.

جمع بندی: RAG چیست؟

در دنیایی که به‌روز ماندن اهمیت زیادی پیدا کرده است، RAG رویکردی قابل اعتماد برای آگاه نگهداشتن مدل‌های LLM و ارائه پاسخ‌های دقیق ارائه می‌دهد. در مقایسه این رویکرد با سایر روش‌های پاسخ‌دهی توسط AI، می‌توان گفت که چنانچه Retrieval-Augmented Generation با سایر روش‌ها، مانند GPT و N-gram، ترکیب شود، نتایج دقیق‌تر و مطابق با واقعیت را با سرعت بالاتری ارائه خواهد داد.