Engineer

Retrievalgestützte Generierung

Antworten mit pro Frage abgerufenen Dokumenten erzeugen

Was es ist

Retrievalgestützte Generierung beantwortet eine Frage mithilfe eines Sprachmodells in zwei Schritten. Zuerst durchsucht die Anwendung einen Dokumentbestand nach den Textstellen, die die Antwort am ehesten enthalten. Dann legt sie diese neben die Frage in die Eingabe des Modells. Das Modell schreibt seine Antwort daraus und nicht aus dem, was es beim Training aufgenommen hat. Der englische Name Retrieval-Augmented Generation, meist zu RAG verkürzt, stammt aus einem Forschungsartikel von 2020 über wissensintensive Sprachaufgaben, etwa die Beantwortung offener Fragen.

Der Artikel beschrieb zwei Arten von Gedächtnis. Ein vortrainiertes Modell bewahrt das Gelernte in seinen Parametern auf. Die Autoren nannten das parametrisches Gedächtnis. Ein getrennter Dokumentindex, den ein trainierter Retriever durchsucht, ist nichtparametrisches Gedächtnis. Ihr System verband ein vortrainiertes Sequenz-zu-Sequenz-Modell mit einem dichten Vektorindex der Wikipedia. Sie verglichen zwei Arten, die Suchergebnisse zu verwenden: Die eine stützt die ganze Antwort auf dieselben abgerufenen Textstellen, die andere kann für jedes erzeugte Token andere Textstellen heranziehen. Auf drei Benchmarks zur Beantwortung von Fragen ohne festgelegtes Themengebiet erzielte das System neue Bestwerte. Die Autoren fanden seine erzeugten Texte spezifischer, vielfältiger und sachlich richtiger als die eines vergleichbaren Modells ohne Retrieval.

Eine Anwendung kann dasselbe Muster aus bereits vorhandenen Teilen aufbauen: einem Suchindex über ihre eigenen Dokumente und einem großen Sprachmodell, das sie über eine API aufruft. Die Retrieval-Hälfte ist gewöhnliche Suche, und beide Suchfamilien kommen hier vor. Sparse Retrieval gleicht Wörter ab, so wie eine Volltextsuche Dokumente mit einer Bewertungsfunktion wie BM25 einordnet. Dense Retrieval verwandelt die Frage und jede Textstelle in Zahlenvektoren und liefert die Textstellen, deren Vektoren am nächsten liegen.

Das ist Vektorsuche. Ein Artikel von 2020 zur dichten Suche nach Textstellen fand, dass ein dichter Retriever, der mit einer überschaubaren Zahl von Frage-Textstellen-Paaren trainiert wurde, eine starke BM25-Vergleichsmethode um 9 bis 19 Prozentpunkte übertraf: Gemessen wurde, wie oft die richtige Textstelle unter den ersten zwanzig Ergebnissen erschien. Ein System kann beide Verfahren ausführen und ihre Ergebnislisten zusammenführen.

Das Konzept für eine Entscheidung nutzen

Nutzen Sie „Was ist retrievalgestützte Generierung?“ für eine konkrete Entscheidung. Wir können helfen, das Konzept mit dem System in Verbindung zu bringen, das Sie entwickeln oder einkaufen. Ihre Datensätze sind für KI-Agenten erreichbar, jeder Kunde ist nur einmal erfasst, niemand sieht mehr als erlaubt und jede Antwort lässt sich bis zur Quelle zurückverfolgen.

Unsere Leistungen entdecken Kontakt aufnehmen →

Die Annahmen hinter dem Begriff prüfen

Nutzen Sie die Erklärung und die verlinkten Arbeiten, um Fragen für Ihr Projekt zu formulieren. Wir können die Annahmen gemeinsam mit Ihnen prüfen, bevor ein Werkzeug ausgewählt wird.
Geografische Datenpipeline 50-mal schneller — 50x schnellere Geodaten-Pipelines

Geografische Datenpipeline 50-mal schneller Kontakt aufnehmen →

Warum es wichtig ist

Ein Sprachmodell kennt nur, was in seinen Trainingstexten stand. Diese enden zu einem festen Datum und enthielten nie Ihre Verträge, Preislisten oder das Protokoll der letzten Woche. Der RAG-Artikel nennt zwei Grenzen von Wissen in Parametern: Die Herkunft einer Antwort ist schwer nachzuweisen, und das Wissen ist schwer zu aktualisieren. Retrieval setzt bei beidem an. Die Antwort kann die erhaltenen Textstellen zitieren, sodass ein Leser die Quelle öffnen und überprüfen kann. Neues Wissen erreicht das Modell, sobald ein Dokument in den Index gelangt, ganz ohne Trainingslauf.

Auch die Frage, wer was sehen darf, landet wieder an der richtigen Stelle. Retrieval ist eine Suche, und eine Suche kann mit den Rechten der fragenden Person laufen. Das Modell bekommt dann nur Textstellen, die diese Person selbst hätte öffnen dürfen. Ein Modell, das mit denselben Dokumenten trainiert wurde, hat eine solche Grenze nicht: Alles Gelernte steht allen zur Verfügung, die es benutzen.

Der Preis ist, dass die Antwort nicht besser als das Suchergebnis sein kann. Ein Modell mit falschen Textstellen schreibt daraus eine flüssige Antwort. Bekommt es nichts Brauchbares, antwortet es möglicherweise trotzdem aus dem Gedächtnis. Der größte Teil der Arbeit an einem retrievalgestützten System fließt deshalb in die Daten und die Suche, nicht in das Modell.

Wie es in der Praxis aussieht

Der folgende Ablauf verfolgt eine Frage durch ein retrievalgestütztes System. Die Suche läuft mit den Berechtigungen des fragenden Nutzers. Jede Textstelle kommt mit ihrer Quelldatei und Seite, und die Anweisungen verlangen, nur aus den Textstellen zu antworten:

```
Frage: Welche Lieferbedingungen gelten für Auftrag 4471?
Nutzer:   sales-3 (Rolle: sales, Region: north)

Abgerufen und nach den Leserechten von sales-3 gefiltert:
  [1] contracts/framework-2025.pdf, Seite 4
      "Die Ware wird an das Lager des Käufers geliefert,
       sofern der Auftrag nichts anderes festlegt."
  [2] orders/4471.txt, Zeile 12
      "Lieferbedingungen: gemäß Rahmenvertrag."

Anweisungen:
  Antworten Sie nur anhand der Textstellen unten; zitieren Sie sie als [n].
  Falls sie die Frage nicht beantworten, sagen Sie das.

Antwort:
  Auftrag 4471 verweist auf den Rahmenvertrag [2]. Gemäß diesem Vertrag
  wird die Ware an das Lager des Käufers geliefert [1].
```

Jeder Teil dieses Ablaufs besteht aus Daten, die das System prüfen kann. Ein Test kann bestätigen, dass für eine bekannte Frage die richtigen Textstellen zurückkamen. Ein Skript kann prüfen, dass jedes `[n]` in der Antwort auf eine tatsächlich abgerufene Textstelle verweist, und ein Mensch kann beim Lesen die Seite 4 des Vertrags öffnen.

Die nötige Vorarbeit behandelt die Seite dieser Website über Unternehmensdaten für KI-Agenten. Sie beschreibt, wie alte PDFs, Scans und E-Mail-Archive in durchsuchbaren Text überführt werden, dessen Quelle daneben erhalten bleibt, und wie jede Zeile ihre Herkunft behält: System, Ladevorgang, Datei und Seite. So kann ein Agent den Datensatz hinter einer Zahl zitieren und ein Mensch ihn öffnen. Die Seite legt auch die Regeln in die Daten statt in den Prompt: Der Agent handelt mit den Berechtigungen seines Nutzers, niemals denen eines Administrators, und jeder Lesezugriff wird mit der Person protokolliert, für die er ausgeführt wurde.

Wo es schiefgeht

Die Suche findet die Antwort nicht, das Modell antwortet trotzdem. Wird die richtige Textstelle nicht abgerufen, bekommt das Modell dennoch etwas und schreibt dennoch eine Antwort. Messen Sie Retrieval für sich: Halten Sie echte Fragen mit den Textstellen fest, die sie beantworten, und prüfen Sie, wie oft diese unter den Ergebnissen auftauchen, die das Modell erhält. Weisen Sie es an, ausdrücklich zu sagen, wenn die Textstellen die Frage nicht beantworten, und nehmen Sie solche Fragen in die Testfälle auf.

Textstellen schneiden die Antwort in zwei Hälften. Werden Dokumente in Stücke fester Länge zerlegt, trennt das eine Tabelle von ihrer Überschrift und eine Klausel von ihrer Ausnahme. Jede allein gefundene Hälfte führt in die Irre. Teilen Sie entlang der vorhandenen Dokumentstruktur, ihrer Abschnitte, Seiten und Tabellenzeilen, und bewahren Sie Titel und Fundort des Dokuments bei jedem Teil auf.

Der Index ignoriert, wer fragt. Ein Index, den ein Dienst mit Leserechten auf alles aufbaut, liefert Textstellen aus Dokumenten, die die fragende Person nicht öffnen darf. Das Modell fasst sie für diese Person zusammen. Filtern Sie schon in der Suche, im Index oder in der Datenbank nach den Nutzerrechten, bevor etwas das Modell erreicht. Eine Regel im Prompt ist keine Zugriffskontrolle.

Anweisungen sind in ein Dokument eingeschleust. Abgerufener Text stammt von jemand anderem, und das Modell kann eine Textstelle nicht zuverlässig von einer Anweisung unterscheiden. Das NIST-Profil für generative KI beschreibt indirekte Prompt-Injektion als Angriffe auf Anwendungen mit Sprachmodellen durch Daten, die wahrscheinlich abgerufen werden. Ein Artikel von 2023 zeigte solche Angriffe auf reale Systeme. Behandeln Sie jede abgerufene Textstelle als nicht vertrauenswürdige Eingabe. Lassen Sie keine Textstelle ohne Kontrolle im Code eine Aktion auslösen, und geben Sie den aufrufbaren Werkzeugen des Modells nicht mehr Rechte als dem Nutzer.

Der Index bleibt zurück. Ein Dokument wird geändert oder gelöscht, doch der Index enthält weiterhin die alten Textstellen. Das System antwortet dann selbstsicher auf Grundlage eines Vertrags, der nicht mehr gilt. Aktualisieren Sie den Index bei jeder Quellenänderung, entfernen Sie Textstellen beim Löschen ihres Dokuments und zeigen Sie das Datum jeder Quelle neben der Antwort an.

Quellenverweise belegen nicht, was die Antwort behauptet. Ein Modell kann Textstelle [2] für einen Satz zitieren, den Textstelle [2] nicht stützt. NIST merkt an, dass auch die erzeugten Quellenverweise selbst erfunden sein können. Prüfen Sie im Code, dass jeder Verweis auf eine abgerufene Textstelle zeigt. Lassen Sie Menschen im Rahmen regelmäßiger Tests stichprobenweise Antworten mit ihren zitierten Textstellen vergleichen.

Weiterführende Lektüre

- Retrievalgestützte Generierung für wissensintensive Sprachverarbeitungsaufgaben — Der Artikel von 2020, der RAG-Modelle einführte und ihre beiden Varianten bei Fragebeantwortung und Generierung verglich.
- Dichte Textstellensuche für die Beantwortung von Fragen ohne festgelegtes Themengebiet — Ein mit Frage-Textstellen-Paaren trainierter dichter Retriever, gemessen an BM25 beim Finden der Textstelle mit der Antwort.
- NIST AI 600-1: Profil für generative künstliche Intelligenz — NISTs Risikoprofil für generative KI zu erfundenen Aussagen, Datenschutz und indirekter Prompt-Injektion.
- Nicht das, wofür Sie sich angemeldet haben: Reale Anwendungen mit integrierten Sprachmodellen durch indirekte Prompt-Injektion kompromittieren — Wie Anweisungen in abgerufenen Daten eine Anwendung übernehmen, mit Angriffen auf reale Systeme.

Glossar

Für Ihre tatsächlichen Rahmenbedingungen auswählen

Bringen Sie einen repräsentativen Datensatz, die Fragen, die er beantworten muss, und den aktuellen Engpass mit. Wir können vereinbaren, wie Qualität, Durchsatz und Wiederherstellung gemessen werden.

Preise ansehen Kontakt aufnehmen →

Die Begriffe praktisch nutzbar machen

Fragen Sie, wie „Was ist retrievalgestützte Generierung?“ auf Ihre Organisation zutrifft. Ein konkretes Beispiel aus Ihrem bestehenden System reicht aus, um das Gespräch zu beginnen.

Unsere Leistungen entdecken

welcome@engineer.company +45 50 10 61 84 Kopenhagen, Dänemark

Nehmen Sie diese Seite mit

Kontaktkarte

Mit Sorgfalt und Liebe gestaltet von D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Copyright © 2025 – heute · Engineer ApS · Alle Rechte vorbehalten.

Gedruckt von https://engineer.company/de/glossary/retrieval-augmented-generation/