# بررسی و پیاده‌سازی جست‌وجوی برداری

لینک نوتبوک کولب: اینجا

نام گروه

Tensor Titans


اعضای گروه

غزل عسکری

یاسمین صرافی

سپیده سلیمانیان

امیرحسین رجبی

محمدرضا ویلانی


تاریخ تنظیم سند

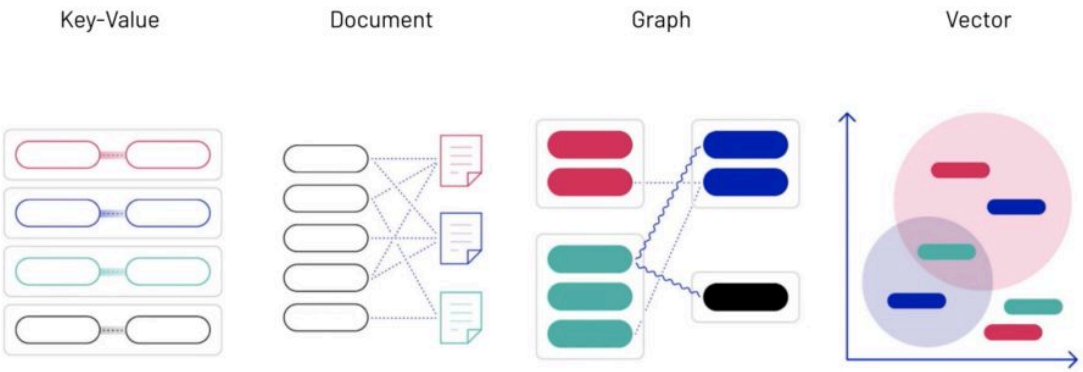
شهریور ۱۴۰۳

## پایگاه‌های داده برداری[1] [2]:

در دنیای هوش مصنوعی (AI)، حجم زیادی از داده‌ها نیاز به مدیریت و پردازش کارآمد دارند. با ورود به برنامه‌های پیشرفته‌تر هوش مصنوعی مانند تشخیص تصویر، جستجوی صوتی یا سیستم‌های پیشنهاددهنده، ماهیت داده‌ها پیچیده‌تر می‌شود.

امبدینگ یکی از ابزارهای بسیار متنوع در پردازش زبان طبیعیست که طیف گسترده‌ای از کاربردها را پشتیبانی می‌کند. به طور کلی، امبدینگ نمایش‌های عددی از اشیاء پیچیده‌تر مانند متن، تصاویر، صدا و غیره است. به طور خاص، این اشیاء به صورت بردارهای n-بعدی نمایش داده می‌شوند.

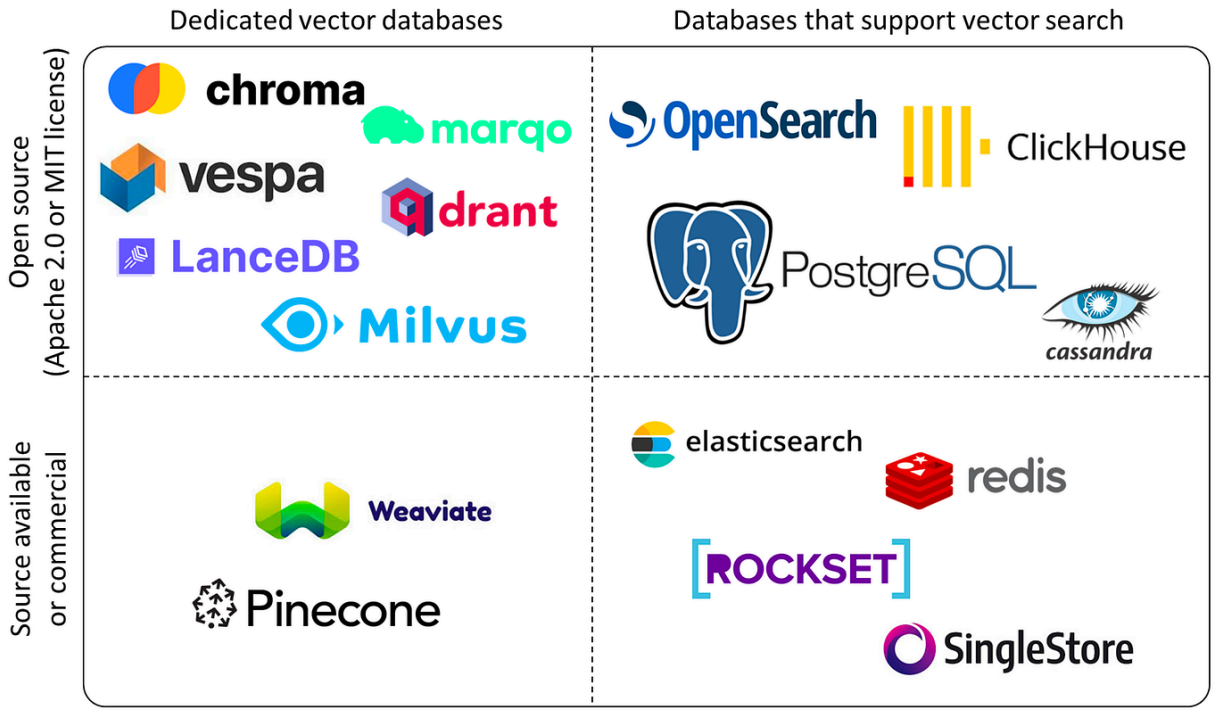
**اینجاست که پایگاه‌های داده برداری وارد می‌شوند.** برخلاف پایگاه‌های داده سنتی که مقادیر اسکالر را ذخیره می‌کنند، پایگاه‌های داده برداری به طور خاص برای مدیریت نقاط داده چندبعدی طراحی شده‌اند که اغلب به آن‌ها بردار گفته می‌شود. این بردارها که داده‌ها را در ابعاد مختلف نشان می‌دهند، می‌توانند به عنوان پیکان‌هایی تصور شوند که به سمت خاصی با شدت خاصی در فضا اشاره دارند.




مزیت اصلی یک پایگاه داده برداری توانایی آن در یافتن و بازیابی داده‌ها با سرعت و دقت بر اساس **نزدیکی یا شباهت برداری** است. این امکان جستجوهای مبتنی بر ارتباط معنایی یا مفهومی را فراهم می‌کند، نه فقط اتکا بر تطابق‌های دقیق یا معیارهای ثابت، همانطور که در پایگاه‌های داده معمولی اتفاق می‌افتد. پایگاه‌های داده برداری از تکنیک‌های جستجوی خاصی به نام جستجوی نزدیک‌ترین همسایه تقریبی (ANN) استفاده می‌کنند که شامل روش‌هایی مانند هشینگ و جستجوهای مبتنی بر گراف می‌شود. [3]

این امر برای بسیاری از موارد استفاده بسیار مهم است و به عنوان ستون فقرات سیستم‌های پیشنهاددهنده، بازیابی داده، یادگیری one-shot یا few-shot، شناسایی داده‌های نامتعارف، جستجوی شباهت، تشخیص پارافریز، خوشه‌بندی، طبقه‌بندی و بسیاری موارد دیگر عمل می‌کند.

## انواع دیتابیس‌های برداری:




**سوال اول: استفاده از امبدینگ‌های مختلف چقدر بر روی بازدهی کار شما موثر است(FastText, BERT Models, etc.)؟**

مدل‌های مختلف امبدینگ به طور قابل‌توجهی بر جستجوی برداری از نظر عملکرد، کیفیت و مقیاس‌پذیری تأثیر می‌گذارند. انتخاب مدل به تعادلی بین موارد زیر بستگی دارد:

- کارایی جستجو (بردارهای با ابعاد پایین، محاسبه سریع)
- کیفیت جستجو (درک مفهومی و خاص دامنه)
- نیازهای موارد استفاده (real-time بودن، مقیاس بزرگ، یا دغدغه‌ی interpretability).

در اینجا چگونگی تأثیر مدل‌های مختلف امبدینگ بر جستجوی برداری بررسی شده است:

1. **تأثیر بر عملکرد جستجو: ابعاد بردارها**
مدل‌های مختلف امبدینگ‌هایی با ابعاد متفاوت تولید می‌کنند. امبدینگ‌های با ابعاد بالا (مثل 768 یا 1024 بُعد از مدل‌های مشابه BERT) می‌توانند الگوها و جزئیات پیچیده‌تری را نمایش دهند اما ممکن است کارایی جستجو را کاهش دهند. امبدینگ‌های با ابعاد پایین (مثل 300 از Word2Vec یا GloVe) ممکن است جستجوها را سریع‌تر کنند اما ممکن است شباهت‌های ظریف را از دست بدهند.

2. **کیفیت نمایش و تأثیر بر کیفیت جستجو:**
مدل‌های مختلف در توانایی‌شان برای به‌دست‌آوردن اطلاعات مفهومی یا معنایی متفاوت هستند. امبدینگ‌های مفهومی معمولاً در جستجوهای برداری زمانی که شباهت معنایی مهم است (مثل بازیابی داکیومنت‌ها، پاسخ به سوالات) عملکرد بهتری دارند. اما ممکن است هزینه محاسباتی بیشتری به همراه داشته باشند.
- **مدل‌های امبدینگ کلاسیک**: مدل‌هایی مانند Word2Vec، GloVe، یا FastText کلمات را بر اساس الگوهای هم‌رخدادی نمایش می‌دهند اما ممکن است در برخورد با چندمعنایی (کلماتی با چندین معنی) و مترادف‌ها دچار مشکل شوند.
- **مدل‌های امبدینگ مفهومی:** مدل‌هایی مانند BERT، RoBERTa، یا Sentence-BERT امبدینگ‌هایی تولید می‌کنند که به کانتکست کلمات حساس هستند و بهبود شباهت معنایی و مدیریت چندمعنایی را فراهم می‌کنند. Sentence-BERT، به عنوان مثال، به طور خاص برای شباهت معنایی در سطح جمله بهینه‌سازی شده است.

3. **داده‌های آموزشی و خاص‌بودن دامنه:**
مدل‌های از پیش آموزش‌دیده معمولاً بر روی مجموعه‌داده‌های عمومی (مثل ویکی‌پدیا، Common Crawl) آموزش داده شده‌اند که ممکن است برای دامنه‌های خاص (مثل پزشکی، حقوقی) عملکرد بهینه‌ای نداشته باشند. اگر جستجوی برداری شما شامل دامنه‌های خاص باشد، استفاده از یک مدل امبدینگ تنظیم‌شده یا خاص دامنه می‌تواند به‌طور قابل‌توجهی ارتباط جستجو را بهبود بخشد.
- **مدل‌های عمومی**: مدل‌هایی که بر روی متن عمومی آموزش دیده‌اند (مثل GloVe، BERT) ممکن است جزئیات خاص دامنه را دریافت نکنند.
- **مدل‌های خاص دامنه**: مدل‌هایی که بر روی مجموعه‌داده‌های خاص تنظیم شده‌اند (مثل BioBERT، LegalBERT) امبدینگ‌هایی تولید می‌کنند که بهتر بازتاب‌دهنده روابط خاص دامنه هستند.

4. **تأثیر بر پوشش جستجو: مدیریت کلمات خارج از واژگان (OOV)**
مدل‌های قدیمی مانند Word2Vec یا GloVe واژگان ثابتی دارند، به این معنا که هر کلمه‌ای که در زمان آموزش دیده نشده است (OOV) بردار متناظر نخواهد داشت. برای جستجوهایی که شامل کلمات نادر، اصطلاحات عامیانه یا واژگان جدید هستند، مدل‌هایی با قابلیت subword مانند FastText و مدل‌های مفهومی مانند BERT کلمات OOV را بهتر مدیریت می‌کنند زیرا برای واحدهای زیربخش (مثل n-گرم‌های حرفی) امبدینگ‌هایی ایجاد می‌کنند و پوشش برای کلمات نادر یا جدید را بهبود می‌بخشند.

5. **تأثیر بر دقت جستجو: معیارهای شباهت برداری**
مدل‌های امبدینگ مختلف ممکن است با معیارهای خاص شباهت برداری (مثل شباهت کسینوسی، فاصله اقلیدسی) بهتر هماهنگ شوند. به عنوان مثال، امبدینگ‌های Sentence-BERT با شباهت کسینوسی خوب کار می‌کنند زیرا برای آن بهینه‌سازی شده‌اند.

6. **تأثیر بر شفافیت جستجو: قابلیت فهم و سوگیری**
برخی امبدینگ‌ها ممکن است سخت‌تر قابل فهم باشند، به ویژه آن‌هایی که توسط مدل‌های دیپ‌لرنینگ مانند BERT تولید شده‌اند. این می‌تواند فهم دلیل بازیابی نتیجه خاص را دشوار کند. علاوه بر این، داده‌های آموزشی برخی مدل‌ها ممکن است سوگیری‌هایی را وارد کنند که بر خروجی جستجو تأثیر بگذارد.  اگر شفافیت مهم است، ممکن است امبدینگ‌های قابل‌فهم‌تر (مثل Word2Vec) را ترجیح دهید. در برنامه‌های حساس به سوگیری، درک داده‌های آموزشی و سوگیری‌های مدل برای نتایج منصفانه جستجو حیاتی است.

7. **کارایی محاسباتی و تأثیر بر مقیاس‌پذیری:**
مدل‌های مدرن مانند BERT یا RoBERTa از نظر محاسباتی برای تولید و ذخیره امبدینگ‌ها پرهزینه هستند، به ویژه اگر نیاز به تولید امبدینگ‌ها به صورت لحظه‌ای برای پرس‌وجوها باشد. امبدینگ‌های از پیش‌محاسبه‌شده از مدل‌های ساده‌تر (مثل Word2Vec، FastText) برای جستجوهای برداری با مقیاس بزرگ سریع‌تر هستند. بنابراین در برنامه‌هایی که نیاز به مدیریت پرس‌وجوهای لحظه‌ای دارند، مدل‌های محاسباتی کارآمدتر یا امبدینگ‌های از پیش‌محاسبه‌شده ممکن است عملی‌تر باشند.

در حال حاضر، بسیاری از مدل‌های پیشرفته امبدینگ‌هایی با 1024 بعد تولید می‌کنند که هرکدام در float32 کدگذاری شده‌اند، به این معنی که به ازای هر بُعد 4 بایت نیاز دارند. بنابرین برای انجام بازیابی بر روی 250 میلیون بردار، به حدود 1 ترابایت حافظه نیاز دارید!

جدول زیر یک نمای کلی از مدل‌های مختلف، اندازه ابعاد، نیازهای حافظه و هزینه‌ها ارائه می‌دهد. هزینه‌ها با نرخ تخمینی 3.8 دلار به ازای هر گیگابایت در ماه با استفاده از x2gd instances در AWS محاسبه شده است.

| Embedding Dimension | Example Models | 100M Embeddings | 250M Embeddings | 1B Embeddings |
|---|---|---|---|---|
| 384 | all-MiniLM-L6-v2<br>bge-small-en-v1.5 | 143.05GB<br>$543 / mo | 357.62GB<br>$1,358 / mo | 1430.51GB<br>$5,435 / mo |
| 768 | all-mpnet-base-v2<br>bge-base-en-v1.5<br>jina-embeddings-v2-base-en<br>nomic-embed-text-v1 | 286.10GB<br>$1,087 / mo | 715.26GB<br>$2,717 / mo | 2861.02GB<br>$10,871 / mo |
| 1024 | bge-large-en-v1.5<br>mxbai-embed-large-v1<br>Cohere-embed-english-v3.0 | 381.46GB<br>$1,449 / mo | 953.67GB<br>$3,623 / mo | 3814.69GB<br>$14,495 / mo |
| 1536 | OpenAI text-embedding-3-small | 572.20GB<br>$2,174 / mo | 1430.51GB<br>$5,435 / mo | 5722.04GB<br>$21,743 / mo |
| 3072 | OpenAI text-embedding-3-large | 1144.40GB<br>$4,348 / mo | 2861.02GB<br>$10,871 / mo | 11444.09GB<br>$43,487 / mo |

## سوال دوم: چالش چانکینگ را چطور حل کردید؟

در پایگاه‌های داده جستجوی برداری، «چانکینگ» به فرآیند تجزیه داکیومنت‌ها یا داده‌های بزرگ به قطعات کوچک‌تر و مدیریت‌پذیر (یا "چانک‌ها") اشاره دارد تا کارایی جستجو و دقت بازیابی را بهبود بخشد. در زمینه جستجوی برداری، که در آن هر بخش به یک بردار (نمایش عددی داده) تبدیل می‌شود، چانکینگ کمک می‌کند که سیستم به جای اینکه کل داکیومنت‌ها را به عنوان یک واحد در نظر بگیرد، بخش‌های کوچک‌تری از اطلاعات را ایندکس و بازیابی کند.

## مزایای چانکینگ:

1. **بهبود دقت جستجو:**
داکیومنت‌های بزرگ اغلب موضوعات یا ایده‌های متعددی را پوشش می‌دهند و در نظر گرفتن آن‌ها به عنوان یک واحد می‌تواند نتایج جستجو را ضعیف کند. برای بسیاری از جستجوها، کاربران به دنبال جملات یا پاراگراف‌های خاصی در یک داکیومنت بزرگ هستند. چانکینگ تضمین می‌کند که سیستم بتواند نتایجی را بازگرداند که از نظر مفهومی مرتبط‌تر هستند، زیرا هر چانک ممکن است شامل یک موضوع یا ایده متمرکز باشد.

2. **مدیریت بهتر حافظه و ذخیره‌سازی:**
امبدینگ‌های برداری (که نمایش‌های عددی از چانک‌ها هستند) اندازه ثابتی دارند. اگر یک داکیومنت بیش از حد بزرگ باشد، ایجاد یک بردار برای کل داکیومنت ممکن است به از دست رفتن اطلاعات یا امبدینگ‌های بیش از حد پیچیده منجر شود. با چانکینگ، امبدینگ‌ها متمرکزتر و کوچک‌تر می‌شوند و سیستم کارآمدتر می‌شود.

3. **بازیابی سریع‌تر:**
جستجوی چانک‌های کوچک‌تر امکان جستجوهای سریع‌تری را فراهم می‌کند. به جای جستجو در بردارهای بزرگ و پیچیده، سیستم فقط نیاز به جستجو در بردارهای کوچک‌تر مربوط به چانک‌ها دارد، که به روند بازیابی سرعت می‌بخشد.

4. **مدیریت داکیومنت‌های بلند:**
برخی از سیستم‌های جستجوی برداری محدودیتی در اندازه متنی که می‌تواند در یک بردار امبدینگ شود دارند. برای داکیومنت‌های که از این محدودیت‌ها فراتر می‌روند، چانکینگ ضروری می‌شود تا از برش و از دست دادن اطلاعات مهم جلوگیری شود.

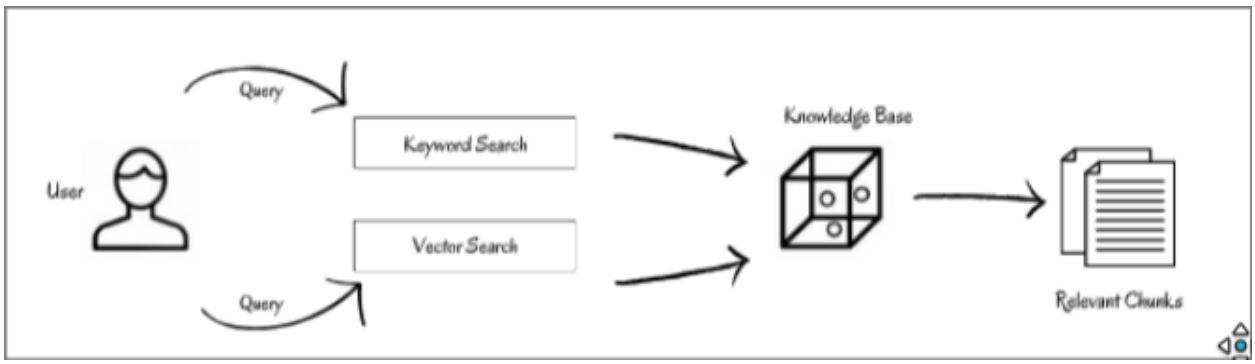
**در پیاده‌سازی ما، از آنجا که با یک فایل CSV کار می‌کردیم و هر ردیف حداکثر ۲ تا ۳ جمله داشت، هر ردیف به عنوان یک چانک در نظر گرفته شد.**

## سوال سوم: آیا می‌توان با روش‌هایی مانند Hybrid Search بازدهی بازیابی اطلاعات با کمک وکتورها را افزایش داد؟ توضیح دهید.

جستجوی بر اساس شباهت برداری و جستجوی مبتنی بر کلمات کلیدی، دارای نقاط قوت و ضعف خاص خود هستند. جستجوی شباهت برداری در مواجهه با پرسش‌هایی که حاوی اشتباهات تایپی هستند، عملکرد بهتری دارد، زیرا این اشتباهات معمولاً قصد کلی جمله را تغییر نمی‌دهند. با این حال، جستجوی شباهت برداری در تطابق دقیق با کلمات کلیدی، اختصارات و نام‌ها که ممکن است در امبدینگ‌های برداری به همراه کلمات اطراف گم شوند، به خوبی عمل نمی‌کند. در این موارد، جستجوی مبتنی بر کلمات کلیدی عملکرد بهتری دارد.

با این وجود، جستجوی مبتنی بر کلمات کلیدی به اندازه جستجوی شباهت برداری در یافتن نتایج مرتبط بر اساس روابط معنایی یا مفاهیم عملکرد خوبی ندارد. این روابط تنها از طریق امبدینگ‌های کلمات قابل دسترسی هستند. برای مثال، یک جستجوی کلمه کلیدی ممکن است عبارات "شیر سلطان جنگل است" و "شیر آب را ببند" را به یکدیگر ربط دهد، حتی بااینکه هیچ ارتباط معنایی بین این اصطلاحات وجود ندارد؛ چیزی که جستجوی شباهت برداری به خوبی به آن حساس است.

در جستجوی ترکیبی الگوریتم‌های جستجوی برداری و جستجوی مبتنی بر کلمات کلیدی را ترکیب می‌کنیم تا از نقاط قوت هر کدام استفاده کنیم و محدودیت‌های آن‌ها را کاهش دهیم. بنابراین، جستجوی مبتنی بر کلمات کلیدی می‌تواند از جستجوی برداری بهره‌مند شود، اما رویکرد رایج این است که این دو را ترکیب نکنیم، بلکه به صورت جداگانه با استفاده از روش‌های مختلف پیاده‌سازی کنیم. [4]




## منابع:

1. https://www.datacamp.com/blog/the-top-5-vector-databases
2. https://huggingface.co/blog/embedding-quantization
3. https://www.elastic.co/what-is/vector-search
4. Optimizing RAG with Hybrid Search & Reranking
https://superlinked.com/vectorhub/articles/optimizing-rag-with-hybrid-search-reranking