Member-only story

Vector Search Is Not All You Need

Anthony Alcaraz · Follow

Published in Artificial Intelligence in Plain English · 6 min read · 4 days ago

Introduction

Retrieval Augmented Generation (RAG) has revolutionized open-domain question answering, enabling systems to produce human-like responses to a wide array of queries. At the heart of RAG lies a retrieval module that scans a vast corpus to find relevant context passages, which are then processed by a neural generative module — often a pre-trained language model like GPT-3 — to formulate a final answer.

While this approach has been highly effective, it's not without its limitations.

One of the most critical components, the vector search over embedded passages, has inherent constraints that can hamper the system's ability to reason in a nuanced manner. This is particularly evident when questions require complex multi-hop reasoning across multiple documents.

Vector search refers to searching for information using vector representations of data. It involves two key steps:

1. **Encoding data into vectors**

First, the data being searched is encoded into numeric vector representations. For text data like passages or documents, this is done using embedding models like BERT or RoBERTa. These models convert text into dense vectors of continuous numbers that represent the semantic meaning. Images, audio, and other formats can also be encoded into vectors using appropriate deep learning models.

2. **Searching using vector similarity**

Once data is encoded into vectors, searching involves finding vectors similar to the vector representation of the search query. This relies on distance metrics like cosine similarity to quantify how close two vectors are and rank results. The vectors with the smallest distance (highest similarity) are returned as the most relevant search hits.

The key advantage of vector search is the ability to search for semantic similarity, not just literal keyword matches. The vector representations capture conceptual meaning, allowing more relevant yet linguistically distinct results to be identified. This enables a higher quality of search compared to traditional keyword matching.

However, transforming data into vectors and searching in high-dimensional semantic space also comes with limitations. Balancing the tradeoffs of vector search is an active area of research.

In this article, we'll dissect the limitations of vector search, exploring why it struggles to capture diverse relationships and intricate interconnections

between documents. We'll also delve into alternative techniques, such as knowledge graph prompting, that promise to overcome these shortcomings.

Understanding the strengths and weaknesses of our current AI tools is essential as they become increasingly integrated into our lives. This article aims to provide a balanced view of where vector search shines and where it falls short in augmenting the reasoning capabilities of large language models.

Semantic Gap Between Questions and Answers

In vector search, both the input question and passages in the corpus are encoded as dense vector representations. Relevant context is retrieved by finding passages with the highest semantic similarity to the question vector.

However, questions often have an indirect relationship to the actual answers they seek.

The vector for "What is the capital of France?" may not necessarily have high similarity to a passage stating "Paris is the most populous city in France".

This semantic gap means that passages containing the answer may be overlooked.

The embeddings fail to capture the inferential link between the question and answer.

Passage Granularity Matters

In vector search systems, passages are typically represented by a single embedding vector. The granularity of these passages can vary.

If the passage is very large, like an entire document, it may encompass multiple concepts. Parts of the passage may be relevant, while other parts are not.

But with a single vector representing the entire passage, it is impossible to distinguish relevant sections from irrelevant ones. The passage as a whole may exhibit only weak similarity to the question vector.

Conversely, using sentence-level chunks can help isolate concepts. But this increases the number of vectors in the index, adding computational overhead.

There are inherent trade-offs between precision and tractability when choosing passage sizes.

Struggle With Complex Reasoning

Some questions demand synthesizing facts spread across multiple documents.

For example, “What is the earliest historical record of winemaking?” may require piecing together dates from various sources.

Vector search is ill-equipped for such multi-hop reasoning. Each passage is scored independently against the question. There is no mechanism to jointly analyze or connect information across separate results.

As questions get more complex, simple similarity search reaches its limits. The system struggles to collect and contextualize facts from different passages.

Black Box Model Workings

In standard vector search pipelines, it is opaque how initial retrieved passages are selected. The rankings depend on the inner workings of the semantic similarity model.

This lack of transparency makes results difficult to explain, verify, and improve. It also limits deployability for business-critical applications.

For increased oversight, the ranking algorithms should provide some interpretability into why certain passages are deemed relevant.

Modeling Diverse Relationships

A core limitation of standard vector search is its singular focus on semantic similarity.

However, real-world reasoning requires modeling diverse relationships between content.

Knowledge graph overcomes this by explicitly encoding various connections into an interconnected graph structure. Specifically:

- *Topical relationships* — Passages are linked if they share rare or key keywords. This captures similarity in the topics discussed.

- *Semantic relationships* — Passage embeddings are compared to connect those with semantic proximity, even if they do not share terms.

This goes beyond surface-level topic matching.

- *Structural relationships* — Passages are connected to the specific sections, pages, or documents they appear in.

This encodes the contextual hierarchy.

- *Temporal relationships* — Passages discussing time-ordered events are linked chronologically.

This represents the flow of events.

- *Entity relationships* — Coreference links are added between passages referencing the same real-world entities.

This allows entity-centric reasoning.

By incorporating these diverse signals beyond just semantic similarity, KGP provides a richer substrate for reasoning about interconnected information.

Structural Relationships

In contrast, standard vector search lacks any notion of these structural relationships. Passages are treated atomically without any surrounding context.

Knowledge graph's modeling of structure relationships merits further discussion. By linking passages to the specific documents or sections they appear in, the contextual hierarchy of the information is encoded.

This enables explicitly reasoning about the section a certain fact is contained in, the document it originates from, and the site it was published on.

Encoding the hierarchical document structure provides useful inductive biases for determining importance, validity, and relevance when reasoning across passages.

Temporal Relationships

This inductive bias is wholly absent in isolated vector search. Vector similarity scores do not factor temporal dynamics in any way. Retrieved passages are disconnected snapshots lacking narrative flow.

The explicit modeling of temporal relationships in KGP also provides significant advantages. Ordering passages chronologically based on the events they describe enables reasoning about unfolding narratives and timelines.

Knowledge graph overcomes this limitation by chaining events based on their relative timing. This unlocks richer reasoning capabilities.

Entity Relationships

With standard vector search, these entity links are not directly modeled. Valuable knowledge around entities is lost in the passage embeddings.

Knowledge graph's ability to connect entity references is a powerful asset. Linking passages that discuss the same real-world entities, concepts, or people allows focused reasoning around those shared elements.

KGP preserves this signal, enabling entity-centric exploration of the knowledge graph. This provides structural advantages when aggregating facts about specific entities across documents.

Conclusion

Vector search enables efficient approximate matching based on semantic similarity. However, it has clear limitations when used in isolation for the retrieval step in RAG systems.

Employing hybrid approaches that combine vector search with graph-based knowledge representation, multi-step reasoning modules, and transparent ranking algorithms can help overcome these weaknesses.

As always, there is no single solution — leveraging a diverse toolkit of techniques is key to robust retrieval for real-world question answering.

Art

Sources :

https://ppl-ai-file-upload.s3.amazonaws.com/web/direct-files/140530/6d15f9ec-46ac-495a-a378-3b3aa977a17a/paste.txt

https://medium.com/thirdai-blog/understanding-the-fundamental-limitations-of-vector-based-retrieval-for-building-llm-powered-48bb7b5a57b3

https://labelbox.com/blog/how-vector-similarity-search-works/

https://www.elastic.co/what-is/vector-search

https://kaushikshakkari.medium.com/open-domain-question-answering-series-part-7-the-rise-of-vector-databases-in-the-world-of-9d848a3f47d5

https://medium.com/@PolonioliAI/limitations-of-vectors-and-neural-search-4d81fd64482f

https://medium.com/vector-database/frustrated-with-new-data-our-vector-database-can-help-e5c430b29be7

https://www.singlestore.com/blog/why-your-vector-database-should-not-be-a-vector-database/

https://clickhouse.com/blog/vector-search-clickhouse-p1

https://www.searchenginejournal.com/semantic-search-with-vectors/467574/

https://www.usenix.org/system/files/osdi23-zhang-qianxi_1.pdf

https://www.infoworld.com/article/3651360/solving-complex-problems-with-vector-databases.html

https://people.eecs.berkeley.edu/~matei/papers/2020/sigir_colbert.pdf

https://blog.futuresmart.ai/gpt-4-semantic-search-and-vector-databases-revolutionizing-question-answering

https://blog.vespa.ai/constrained-approximate-nearest-neighbor-search/

https://www.pinecone.io/learn/vector-search-filtering/

In Plain English

Thank you for being a part of our community! Before you go:

- *Be sure to **clap** and **follow** the writer!* 👏
- *You can find even more content at **PlainEnglish.io*** 🚀
- *Sign up for our **free weekly newsletter**.* 📩
- *Follow us on **Twitter(X)**, **LinkedIn**, **YouTube**,* and ***Discord**.*

AI | Deep Learning | Machine Learning | Software Development | Data Science

Written by Anthony Alcaraz

Follow

2.2K Followers · Writer for Artificial Intelligence in Plain English

Chief AI Officer, ML&LLMOps expert, passionate about decision making.
https://www.linkedin.com/in/anthony-alcaraz-b80763155/ https://aldecis.com/

More from Anthony Alcaraz and Artificial Intelligence in Plain English

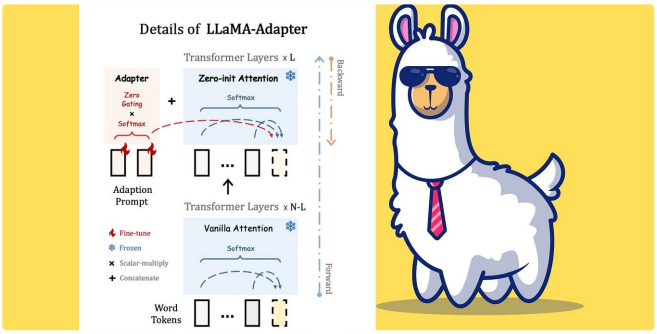
Anthony Alcaraz in GoPenAI

The Complete Overview to Retrieval Augmented Generation...

Retrieval augmented generation (RAG) is an exciting technique that is transforming how...

6 min read · Sep 1

136 2



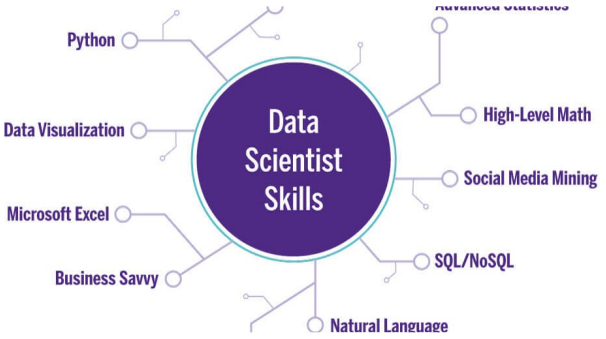
CheeKean in Artificial Intelligence in Plain English

Fine-Tuning LLama2.0 with QLoRA's Single GPU Magic

Efficient Fine-Tuning LLM with Single GPU

11 min read · Aug 1

146 5

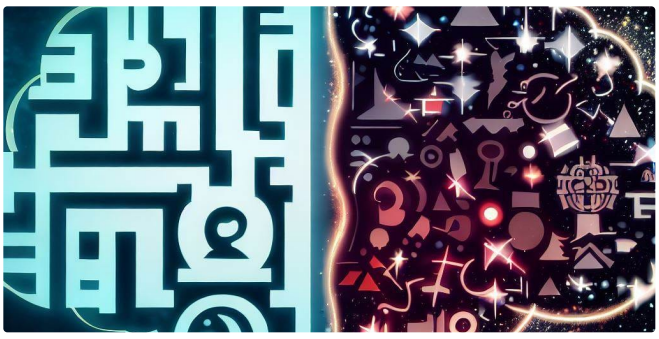


Ritesh Gup... in Artificial Intelligence in Plain Engli...

Master Data Science with This Comprehensive Cheat Sheet

Comprehensive Cheat Sheet for Data Science: Numpy, Pandas, Python, R, ML, DL, NLP, Stat...

6 min read · Jan 29



Anthony Alcaraz in AI Mind

Integrating Knowledge Graphs with Large Language Models for More...

Reasoning—the ability to think logically and make inferences from knowledge—is integr...

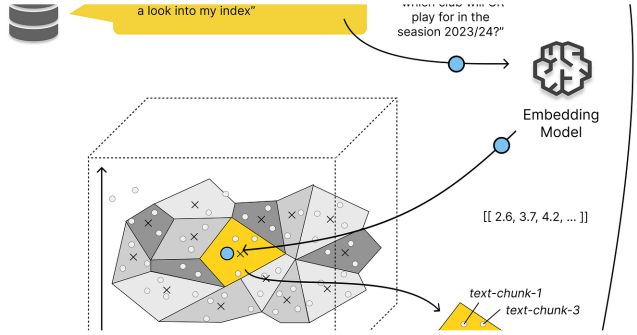
6 min read · Aug 9

1K 14 225 3

See all from Anthony Alcaraz

See all from Artificial Intelligence in Plain English

Recommended from Medium



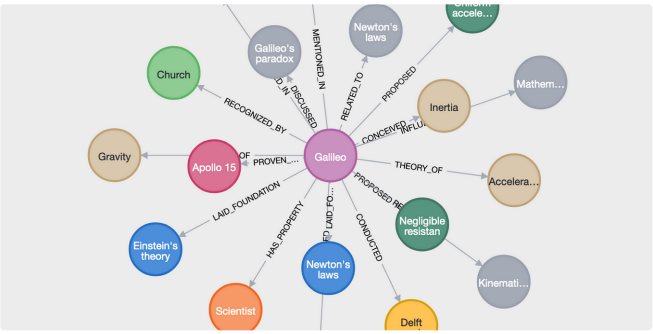
Dominik Polzer in Towards Data Science

All You Need to Know about Vector Databases and How to Use Them ...

A Step-by-Step Guide to Discover and Harness the Power of Vector Databases

· 24 min read · 4 days ago

562 6



Cking

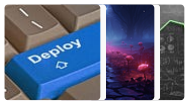
Harnessing the Power of Large Language Models for Knowledge...

· The role of large language models in creating knowledge graphs from unstructured data. ·...

11 min read · Sep 13

10

Lists



Predictive Modeling w/ Python

20 stories · 407 saves

Practical Guides to Machine Learning

10 stories · 472 saves

The New Chatbots: ChatGPT, Bard, and Beyond

13 stories · 123 saves

Natural Language Processing

634 stories · 240 saves

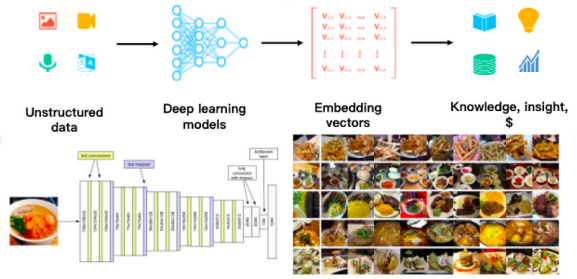
Salvatore Raieli in Level Up Coding

Tabula Rasa: Why Do Tree-Based Algorithms Outperform Neural...

Tree-based algorithms are the winner in tabular data: Why?

· 19 min read · Sep 14

927 11



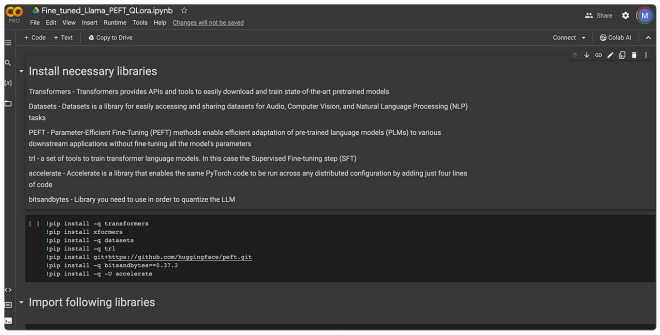
Jayita Bhattacharyya in GoPenAI

Primer on Vector Databases and Retrieval-Augmented Generation...

Vector Databases Generation (RAG) Langchain Pinecone HuggingFace Large...

9 min read · Aug 16

156



Maya Akim

Nick Nolan

Complete Guide to LLM Fine Tuning for Beginners

Fine-tuning a model refers to the process of adapting a pre-trained, foundational model...

5 min read · Aug 13

59

Bankrate Stopped Using AI to Write Articles

Bankrate.com was making headlines in January because they were openly publishin...

· 5 min read · Sep 13

710 10

See more recommendations