Building LLM Powered Solutions

Module 3: Introduction to Semantic Search

Hamza Farooq

Learning Outcomes

- Gain a better understanding of Semantic Search
- Sparse vs. Dense Vectors
- LLM basic Colab
- Euclidean Distance
- Cosine Similarity
- What’s ANN?
- Using FAISS and coding

What is a Retrieval System?

Search and Retrieval Systems are essential tools for information retrieval, enabling fast and precise results with popular methods such as

Euclidean Distance

Cosine Similarity

Approximate Near Neighbors,

Locality Sensitive Hashing,

Hierarchical Navigable Small World Graphs, and

Quantization.

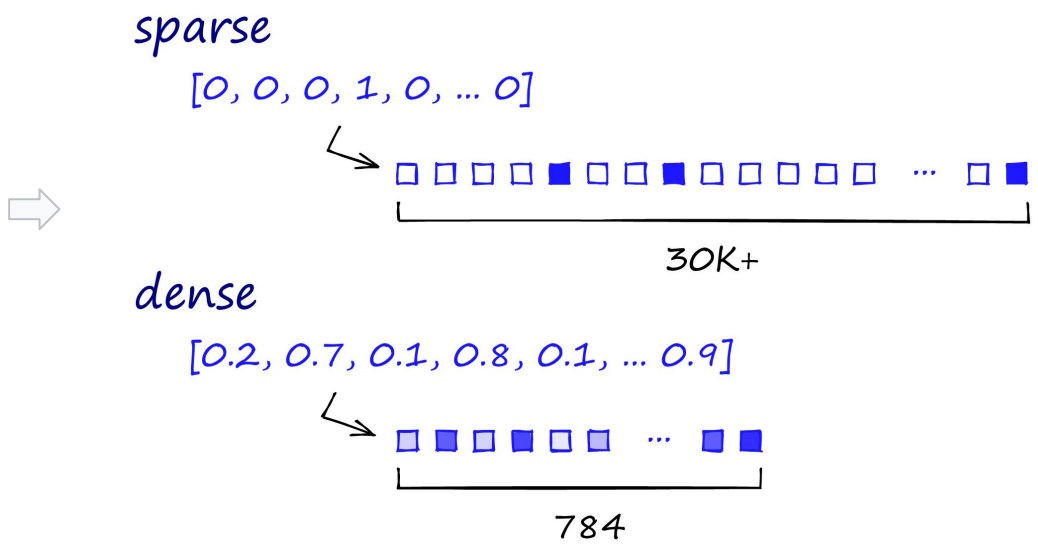
Why Semantic Search?

- Semantic search is a technique that understands the intent and context behind a user's query, rather than relying solely on keyword matching.
- It aims to deliver more accurate and meaningful results by understanding the user's query in a broader context.
- It helps overcome the limitations of traditional keyword-based search by focusing on the user's intent rather than the literal terms used in the query.

why should we represent text using vectors?

for a computer to understand human-readable text, we need to convert our text into a machine-readable format.

```
Bill ran from the
giraffe toward the
dolphin
```



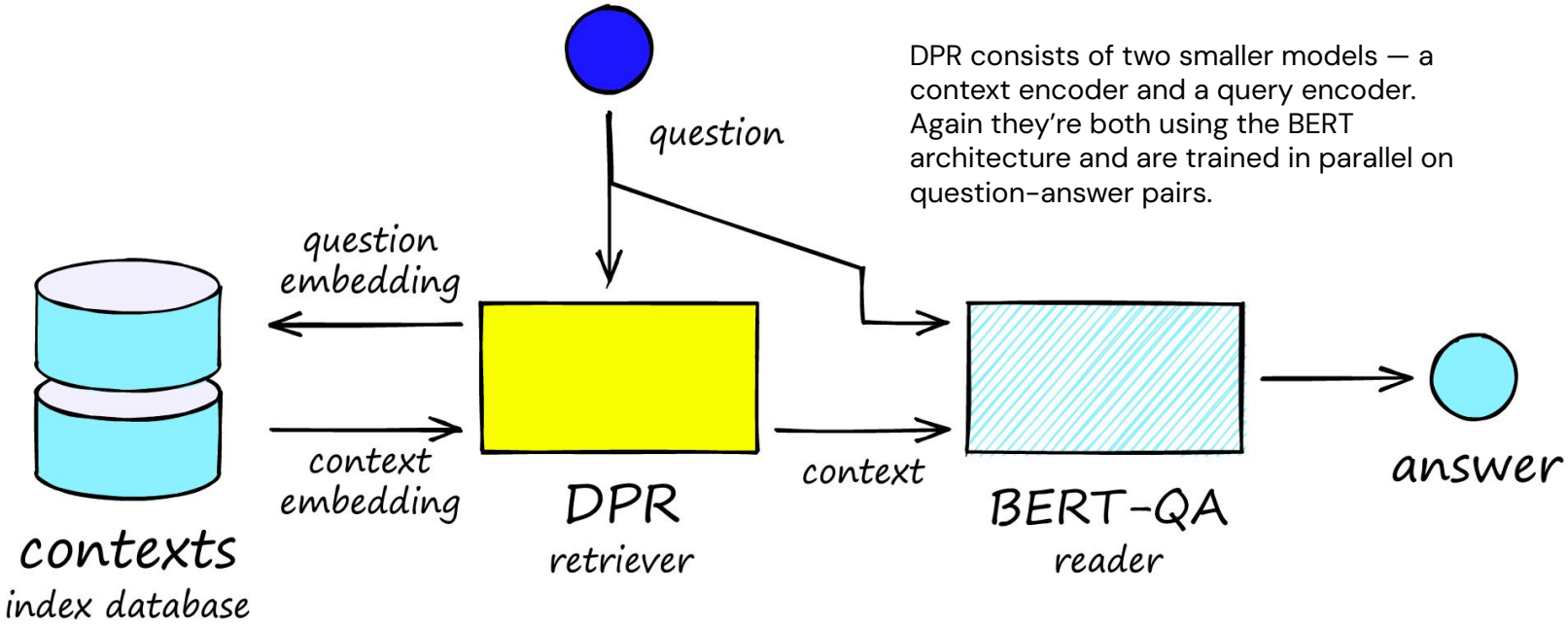
Source

Question Answer is also a Retrieval System

- Another widespread use of transformer models is for questions and answers (Q&A). Within Q&A, there are several different architectures we can use. One of the most common is open domain Q&A (ODQA).
- When doing this, we are making use of three components or models:
    - Some sort of database to store our sentence/paragraphs (called contexts).
    - A retriever retrieves contexts that it sees as similar to our question.
    - A reader model which extracts the answer from our related context(s).

Source

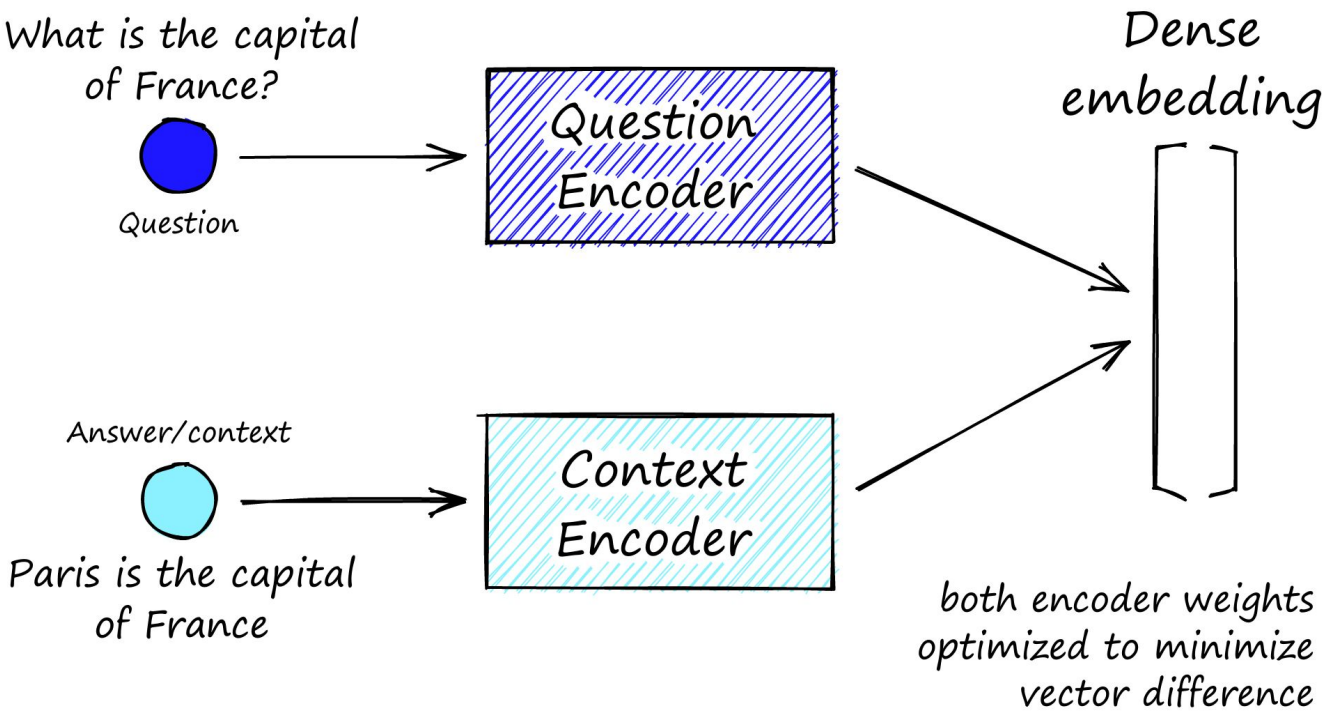
Question Answer is also a Retrieval System



DPR consists of two smaller models — a context encoder and a query encoder. Again they're both using the BERT architecture and are trained in parallel on question-answer pairs.

Source

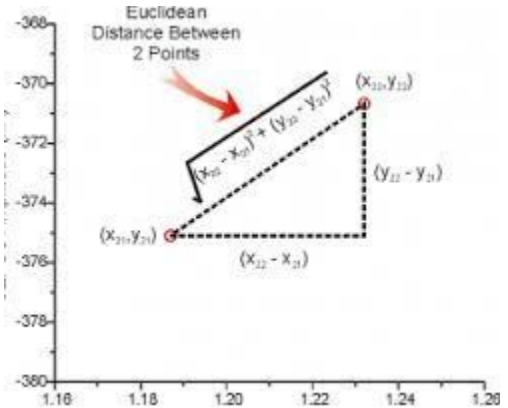
Question Answer is also a Retrieval System



Source

How do we measure distance

IndexFlatL2 measures the L2 (or Euclidean) distance between all given points between our query vector, and the vectors loaded into the index. It's simple, very accurate, but not too fast.

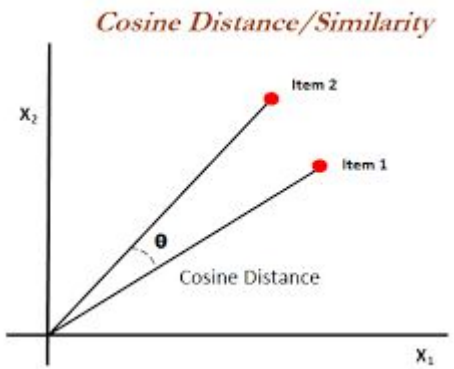


Euclidean distance is a way of measuring the distance between two points in a Cartesian plane. It is calculated by taking the square root of the sum of the squares of the differences between the corresponding coordinates of the two points.

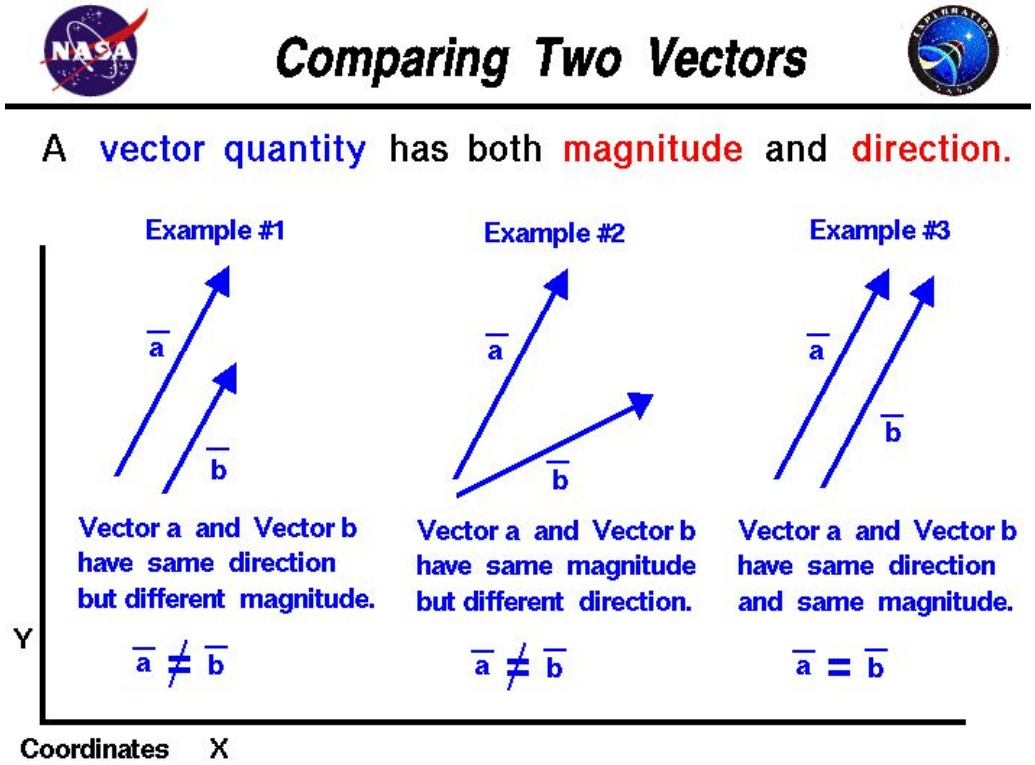
Cosine Similarity

Cosine similarity is a measure of similarity between two vectors. It is calculated by taking the dot product of the two vectors and then dividing by the product of their magnitudes.

In simpler terms, it is a measure of how similar the directions of two vectors are, regardless of their magnitudes.



Magnitude and Direction in a vector



Euclidean distance vs Cosine similarity

- Euclidean distance is typically used when the values of the vectors are important, while cosine similarity is typically used when the directions of the vectors are important.
- For example, Euclidean distance might be used to find the nearest neighbor of a point in a dataset, while cosine similarity might be used to recommend products to a user based on their past purchases.

Euclidean distance vs Cosine similarity

Here are some examples of when to use Euclidean distance and cosine similarity:

Euclidean distance:

- Finding the nearest neighbor of a point in a dataset
- Clustering data points into groups
- Using k-nearest neighbors or support vector machines for machine learning

Cosine similarity:

- Measuring the similarity between two documents, sentences, or words
- Recommending products to a user based on their past purchases
- Ranking search results

Ultimately, the best metric to use depends on the specific application.

Drawback of Cosine and Euclidean

Cons of Euclidean distance when dealing with large search space:

- Sensitivity to dimensionality: Euclidean distance becomes less effective as the number of dimensions increases, known as the "curse of dimensionality."
- Computational complexity: Calculating Euclidean distance requires computing the square root, which can be computationally expensive, especially with large search spaces.
- Lack of normalization: Euclidean distance is not inherently normalized, meaning features with larger magnitudes can dominate the distance calculation.

Drawback of Cosine and Euclidean

Cons of Cosine similarity when dealing with large search space:

- One of the main drawbacks of cosine similarity is that it is not as effective as Euclidean distance at measuring similarity between vectors with different magnitudes. This is because cosine similarity only considers the direction of the vectors, not their magnitudes. This can make it difficult to find similar vectors in a dataset where the magnitudes of the vectors vary widely.
- Another drawback of cosine similarity is that it is not as robust to noise as Euclidean distance. This is because cosine similarity is only affected by the direction of the vectors, not their magnitudes. This means that noise in the vectors can have a significant impact on the cosine similarity between the vectors.

What is noise then?

noise refers to irrelevant or unwanted variations or discrepancies in the data that can potentially impact the accuracy or reliability of the similarity measurement.

Here's an example of two sentences and how noise can affect their similarity measurement using cosine similarity:

- Sentence 1: "The cat chased the mouse."
- Sentence 2: "The cat chased the mouse in the garden."

Noise Example

Sentence 1: "The cat chased the mouse."

Sentence 2: "The cat chased the mouse in the garden."

Sentence 2 (with noise): "The cat chased the elephant in the garden."

What do we do then?

- One approach to deal with large search spaces is reducing vector size by employing techniques like dimensionality reduction or using fewer bits to represent vector values.
- Another approach is to reduce the search scope by clustering or organizing vectors into tree structures based on attributes, similarity, or distance, and then focusing the search on the closest clusters or filtering through the most similar branches.

Hence..

- Using either of these approaches means that we are no longer performing an exhaustive nearest-neighbors search but an approximate nearest-neighbors (ANN) search — as we no longer search the entire, full-resolution dataset.
    - LSH
    - HNSW
    - Quantization

Introducing FAISS

- Faiss is a library — developed by Facebook AI — that enables efficient similarity search.

- So, given a set of vectors, we can index them using Faiss — then using another vector (the query vector), we search for the most similar vectors within the index.

- In vector similarity search, we use an index to store vector representations of the data we intend to search.

Hence..

- Using either of these approaches means that we are no longer performing an exhaustive nearest-neighbors search but an approximate nearest-neighbors (ANN) search — as we no longer search the entire, full-resolution dataset.
    - LSH
    - HNSW
    - Quantization

Homework

Explore all the techniques mentioned here:

- Comprehensive Guide To Approximate Nearest Neighbors Algorithms | by Eyal Trabelsi | Towards Data Science
- Record a one minute video of each member in the group to record and talk about each individual technique, use loom.com to submit link

Thank you.

Appendix