

Retrieval Augmented Generation

CSC401/2511 – Natural Language Computing
Lecture 15
Spring 2026
Presenter: Ken Shi



What we have learnt so far

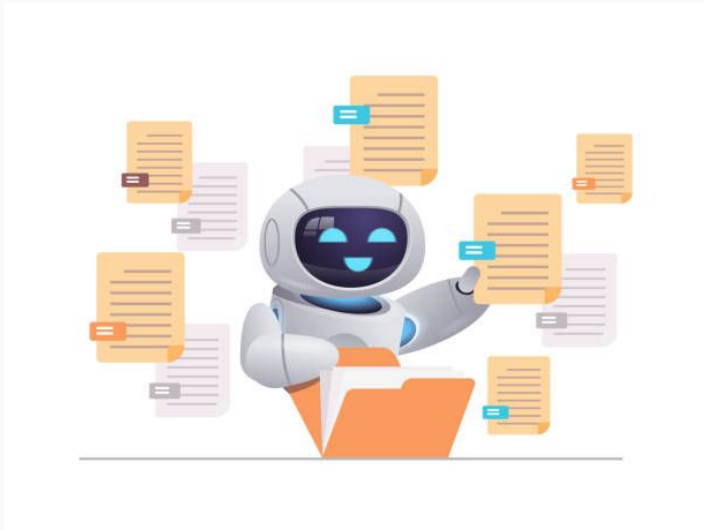
- Language Modeling: Pre-training, Fine-tuning, In-context Learning...
- Language Representation Models: embedding models, encoder-based LMs
- Generative Language Models: autoregressive models, decoder-based LMs
- Information Retrieval: query, knowledge base, search



What problems do we have with these

Plain IR systems:

- ... serves retrieved documents flat
- ... may not *interpret* queries accurately

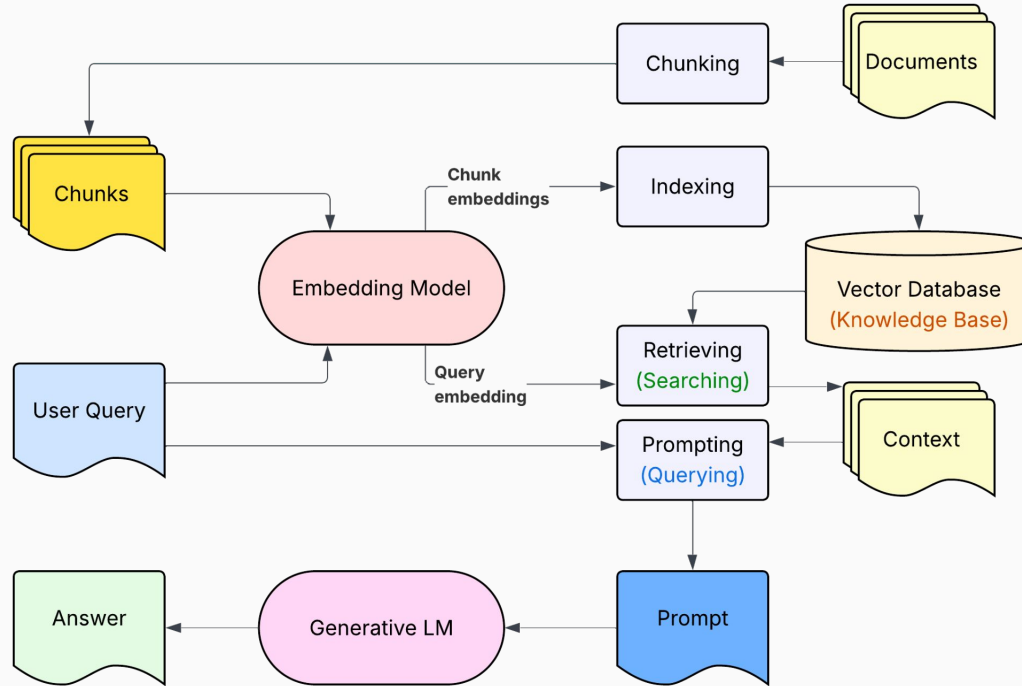


Generative LMs (that we know so far):

- Hallucinations
- Responses are not specific enough



Retrieval-Augmented Generation (RAG)





Chunking

- **Chunk size:** usually 200-300 tokens (moderate)
- **Chunking iteration:** sliding window, with overlap of ~50 tokens
- **Structure-aware Strategies:** Smarter chunking respects document structure.

e.g. :

- splitting on headings and sentences;
- first breaking at paragraph or section boundaries, and then by sentence;
- avoiding mid-sentence splits, etc.

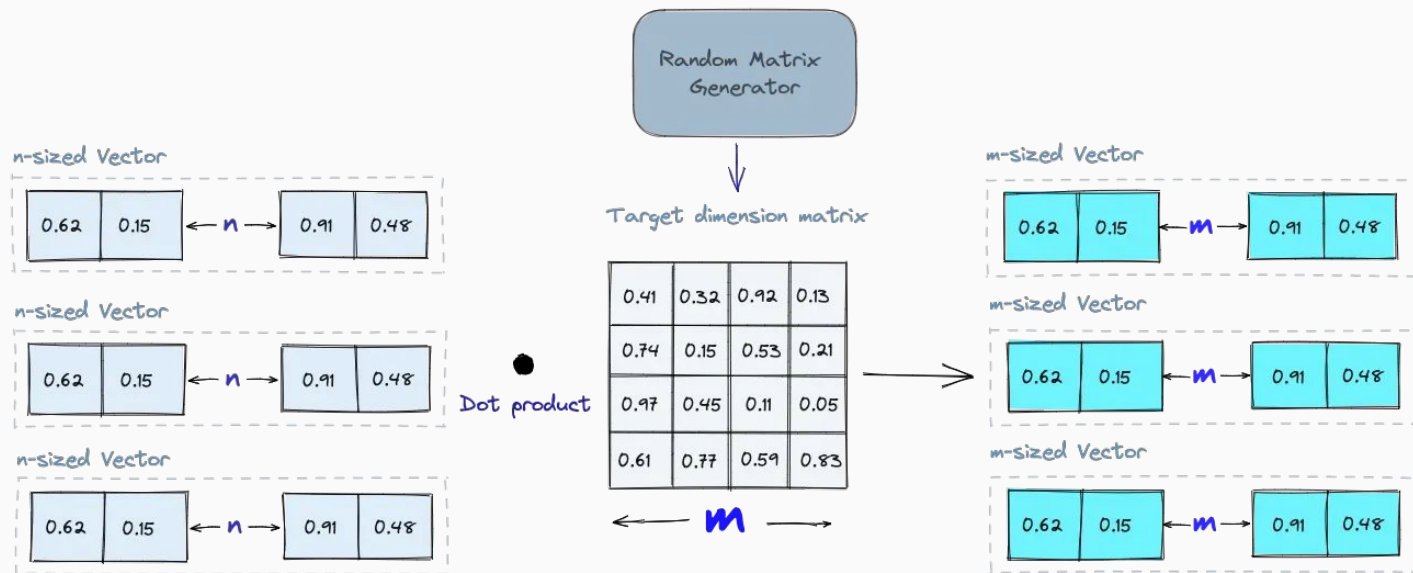


Indexing

Indexing: constructing the vector database from the documents (embeddings) via certain indexing algorithms.

- Indexing maps the vectors to a data structure that will enable faster searching.
- That usually means compressing the vectors to a lower / smaller dimensional space.
- Seems familiar?

Indexing: Random Projection

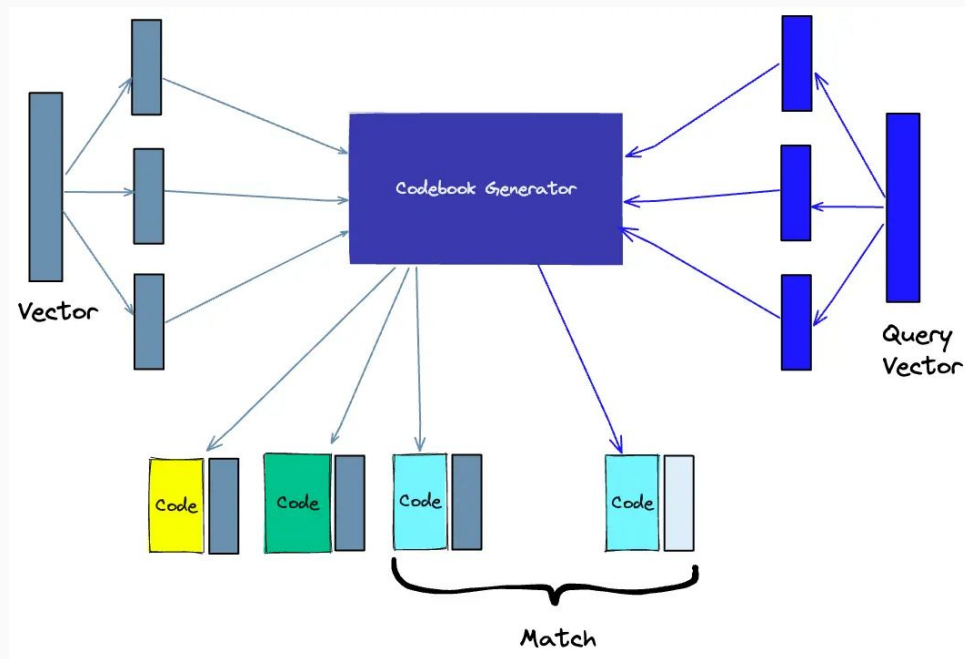


(Figure borrowed from Pinecone)

Indexing: Product Quantization (PQ)

General Process:

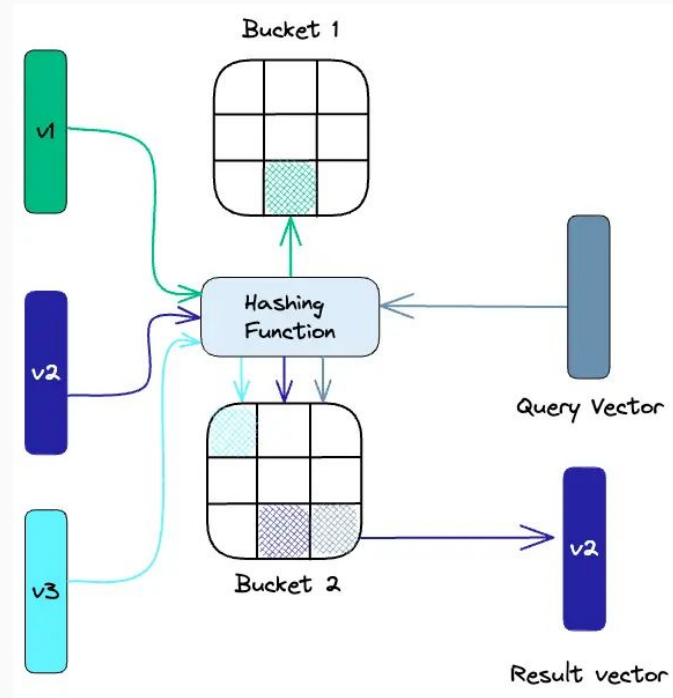
- **Splitting:** chop vectors into segments;
- **Training:** build a “codebook” for each segment
- **Encoding:** assign a specific code to each segment



(Figure borrowed from Pinecone)

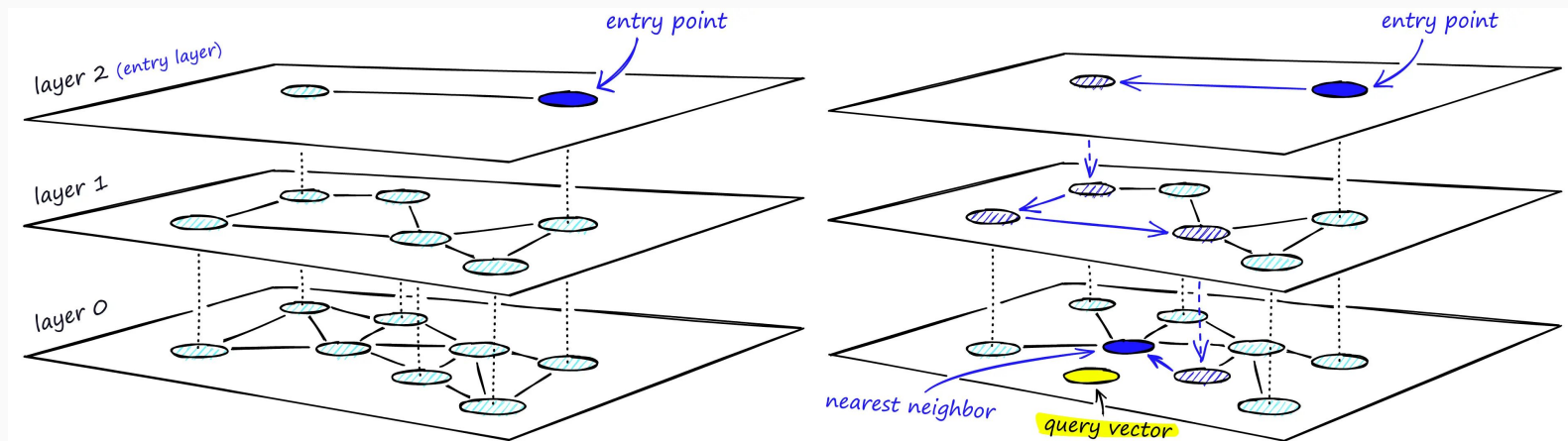
Indexing: Locality-Sensitive Hashing (LSH)

- LSH maps similar vectors into “buckets” using a set of hashing functions.
- The same set of hashing functions is used for fetching the query.



(Figure borrowed from Pinecone)

Indexing: Hierarchical Navigable Small Worlds (HNSW)



(Figures borrowed from Pinecone)



Retrieving

For each of the aforementioned algorithms, we need a way to fetch context from the vector database that is specific to how indexing was done:

- **Random Projection:** compute similarity between projected vectors;
- **PQ:** match codebooks;
- **LSH:** first map query to a “bucket”, then find the closest matches within that “bucket”;
- **HNSW:** Approximate Nearest Neighbour (ANN)



Prompting

Other than the “prompt engineering” on the natural language level, we also have to figure out the exact context we are presenting to the LLM.

Strategies include:

- **Hierarchical chunks:** for retrieved chunks, further compute the cosine similarity between those and the query;
- **Context Window:** extend retrieved chunks to include context before and after the chunks.
- **Hybridization with classic IR system:** remember systems like BM25? We can create a simple mixture model that unifies their results with our RAG results.



RAG Evaluation: Retrieval

Context Precision: to compute, we need query + contexts + reference answer:

$$\text{Context Precision@}K = \frac{\sum_{k=1}^K (\text{Precision@}k \times v_k)}{\text{Total number of relevant items in the top } K \text{ results}}$$

$$\text{Precision@}k = \frac{\text{true positives@}k}{(\text{true positives@}k + \text{false positives@}k)}$$

Where K is the total number of chunks in `contexts` and $v_k \in \{0, 1\}$ is the relevance indicator at rank k .

Context Recall: to compute, we need contexts + ground truth answers:

$$\text{Context Recall} = \frac{|\text{GT sentences that can be attributed to context}|}{|\text{sentences in GT}|}$$



Context Precision: example

Question: Where is France and what is its capital?

Ground truth: France is in Western Europe and its capital is Paris.

High context precision: ["France, in Western Europe, encompasses medieval cities, alpine villages and Mediterranean beaches. Paris, its capital, is famed for its fashion houses, classical art museums including the Louvre and monuments like the Eiffel Tower", "The country is also renowned for its wines and sophisticated cuisine. Lascaux's ancient cave drawings, Lyon's Roman theater and the vast Palace of Versailles attest to its rich history."]

$$P @ 1 = 1, P @ 2 = 0.5 \quad \Rightarrow \quad CP @ 2 = (1 \times 1 + 0.5 \times 0) / 1 = 1$$

Low context precision: ["The country is also renowned for its wines and sophisticated cuisine. Lascaux's ancient cave drawings, Lyon's Roman theater and", "France, in Western Europe, encompasses medieval cities, alpine villages and Mediterranean beaches. Paris, its capital, is famed for its fashion houses, classical art museums including the Louvre and monuments like the Eiffel Tower"]

$$P @ 1 = 0, P @ 2 = 0.5 \quad \Rightarrow \quad CP @ 2 = (0 \times 0 + 0.5 \times 1) / 1 = 0.5$$



Context Recall: example

Question: Where is France and what is its capital?

Ground truth: France is in Western Europe and its capital is Paris.

High context recall: France, in Western Europe, encompasses medieval cities, alpine villages and Mediterranean beaches. Paris, its capital, is famed for its fashion houses, classical art museums including the Louvre and monuments like the Eiffel Tower.

$$\text{CR} = 2 / 2 = 1.0$$

Low context recall: France, in Western Europe, encompasses medieval cities, alpine villages and Mediterranean beaches. The country is also renowned for its wines and sophisticated cuisine. Lascaux's ancient cave drawings, Lyon's Roman theater and the vast Palace of Versailles attest to its rich history.

$$\text{CR} = 1 / 2 = 0.5$$



RAG Evaluation: Generation

Faithfulness: We are NOT looking for the veracity of the claims, but whether the LLM has responded based on the context:

$$\text{Faithfulness score} = \frac{|\text{Number of claims in the generated answer that can be inferred from given context}|}{|\text{Total number of claims in the generated answer}|}$$

Answer Relevancy: compare the embeddings between **generated reverse question** and **the original query**

$$\text{Answer Relevancy} = \frac{1}{N} \sum_{i=1}^N \cos(E_{g_i}, E_o)$$



Faithfulness: example

Question: Where and when was Einstein born?

Context: Albert Einstein (born 20 March 1879) was a German-born theoretical physicist, widely held to be one of the greatest and most influential scientists of all time

High faithfulness answer: Einstein was born in Germany on 20th March 1879.

Low faithfulness answer: Einstein was born in Germany on 14th March 1879.

Please note: Albert Einstein was actually born on 14th March 1879, *factually*.



Answer Relevancy: example

Question: **Where is France** and **what is its capital?**

Low relevance answer: **France is in western Europe.**

High relevance answer: **France is in western Europe** and **Paris is its capital.**