

Memory for LLMs

An IR Problem

Antonio Mallia

IR-RAG @ SIGIR'2025

July 17, 2025

About me



✉ me@antoniomallia.it
🏠 www.antoniomallia.it

X @antonio_mallia
Cat amallia
in in/antoniomallia

I am currently a **Staff Research Scientist** at Pinecone.

Prior to this, I served as an Applied Scientist on the Artificial General Intelligence (AGI) team at Amazon.

I hold a Ph.D. from New York University, where my research focused on efficient web retrieval methodologies.



Pinecone

Founded by Edo Liberty

Created the vector database category

Pinecone's mission is to make AI knowledgeable

Pinecone is the leading vector database for building accurate and performant

AI applications at scale in production.

Go and check it out: <https://www.pinecone.io/>

When we think about search

Information retrieval (IR) is finding material (usually documents) of an unstructured nature (usually text) that satisfies an information need from within large collections.

The screenshot shows a Google search results page with a dark theme. The search query "keynote speaker ir-rag 2024" is entered in the search bar. Below the search bar, the "Tutti" filter is selected. The results list three entries:

- Coda**
https://coda.io › ir-rag-sigir24 · IR-RAG @ SIGIR24: Information Retrieval's Role in ...
Opening. 9:00 AM ; Keynote speaker (1). 9:30 AM. Nicola Tonellotto, University of Pisa ; Coffee break.
10:30 AM. ☕☕☕☕ ; Poster presentations. 11:00 AM ; Lunch ...
- ECIR 2024**
https://www.ecir2024.org › ke... · Traduci questa pagina · Keynote Speakers
ECIR 2024 Keynote Speakers: Prof. Charles LA Clarke, University of Waterloo, Evaluating Generative Information Retrieval Systems.
- SIGIR 2024**
https://sigir-2024.github.io › ke... · Traduci questa pagina · Keynote
This talk was the acceptance speech upon receiving the ACM SIGIR 2024 Gerard Salton award. It recaps my information retrieval career and offers my opinions.

Nowadays, Information Retrieval goes beyond building just search engines!

When we think about RAG

ChatGPT 4o ▾

↑ Share

Who was the keynote speaker at IR-RAG last year?

Last year's IR-RAG workshop — the first edition held at SIGIR 2024 on July 18, 2024 — featured two keynote speakers:

- **Nicola Tonellotto** (University of Pisa) opened the workshop, delivering the first keynote. coda.io +8
- **Yuhao Zhang** (Samaya AI) gave the second keynote of the day. coda.io

If you had one particular speaker in mind, feel free to ask—I'm happy to dive deeper!



Sources

RAG Breaks the Old Rules

In RAG, the retriever isn't helping a user – it's helping a model. That changes everything.

Traditional IR



User browses ranked list



Optimize for top-k relevance (MAP, NDCG)



Queries are short, keyword-based



Focus on document ranking

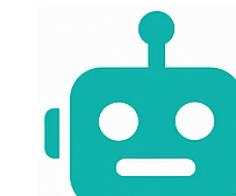


Evaluation: clicks, rank positions



Retrieval ends with the user

RAG Systems



LLM consumes retrieved content



Optimize for grounding and synthesis



Queries are natural, long-form, ambiguous



Focus on passage evidence selection



Evaluation: faithfulness, hallucination rate



Retrieval fuels generation

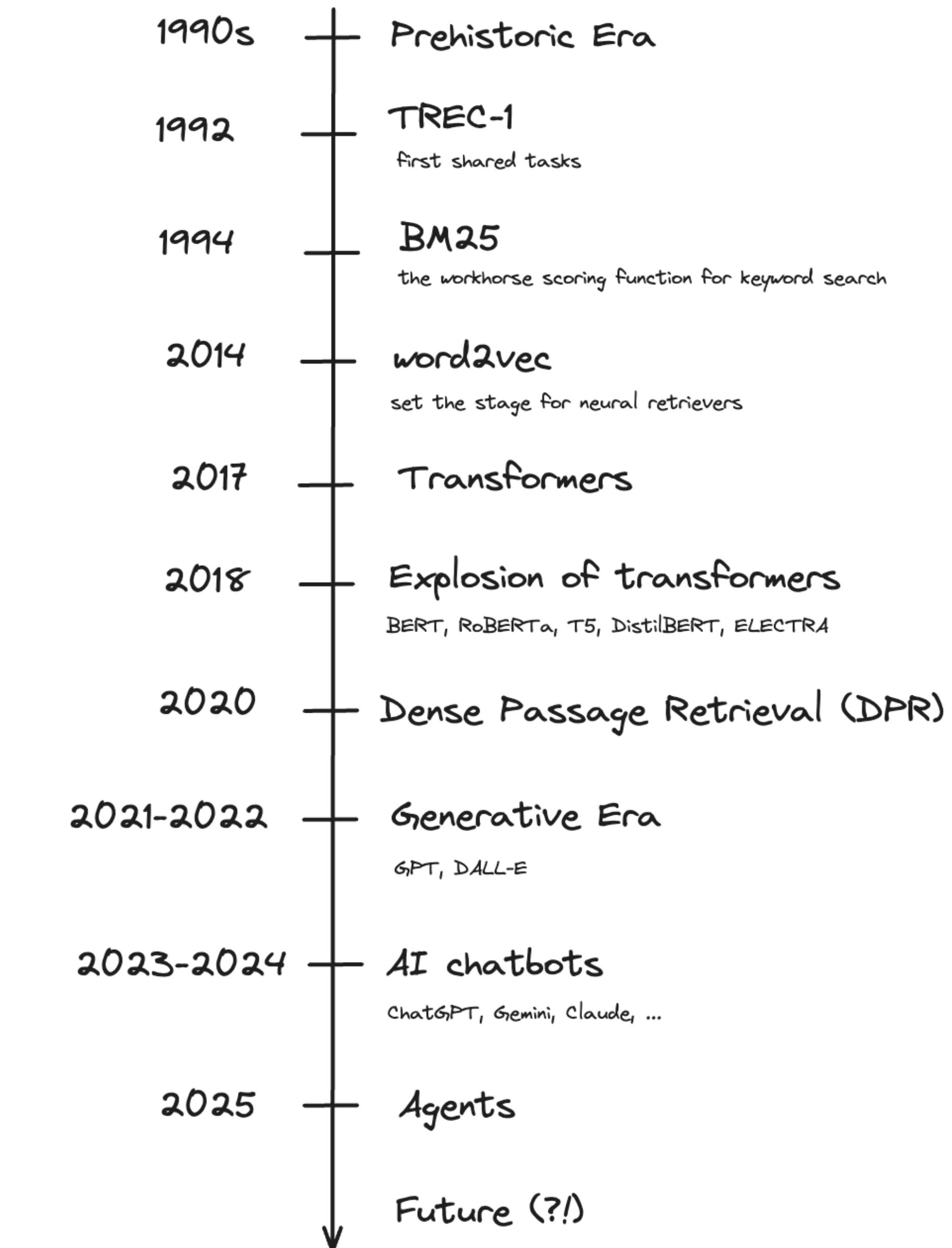
From TREC-1 to Agents: 30 Years of Retrieval & Generation

- Understand Progression

- Recognize Key Innovations

- Appreciate Impact on Applications

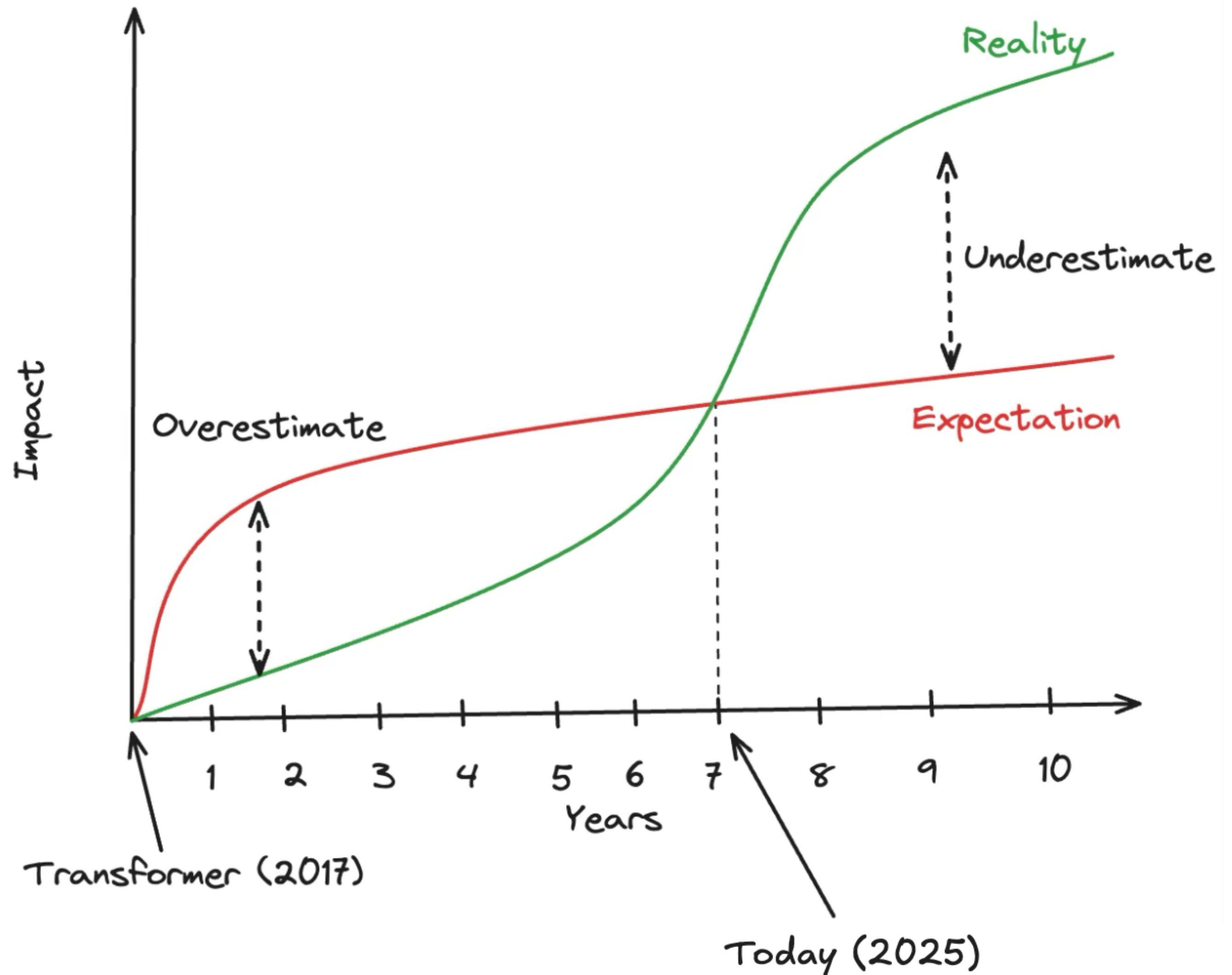
- Guide Future Research



Why Now?

We always overestimate the change that will occur in the next few years and underestimate the change that will occur in the next ten.

Bill Gates, The Road Ahead



Large Language Models (LLMs)

Enhanced versions of the Transformers, often featuring millions or billions of parameters

Generally trained on vast quantities of diverse textual data, such as web corpora

They exhibit new capabilities as their scale increases, such as chain-of-thought reasoning

They require significant computational resources

Trained once, then adapted to various tasks without needing retraining

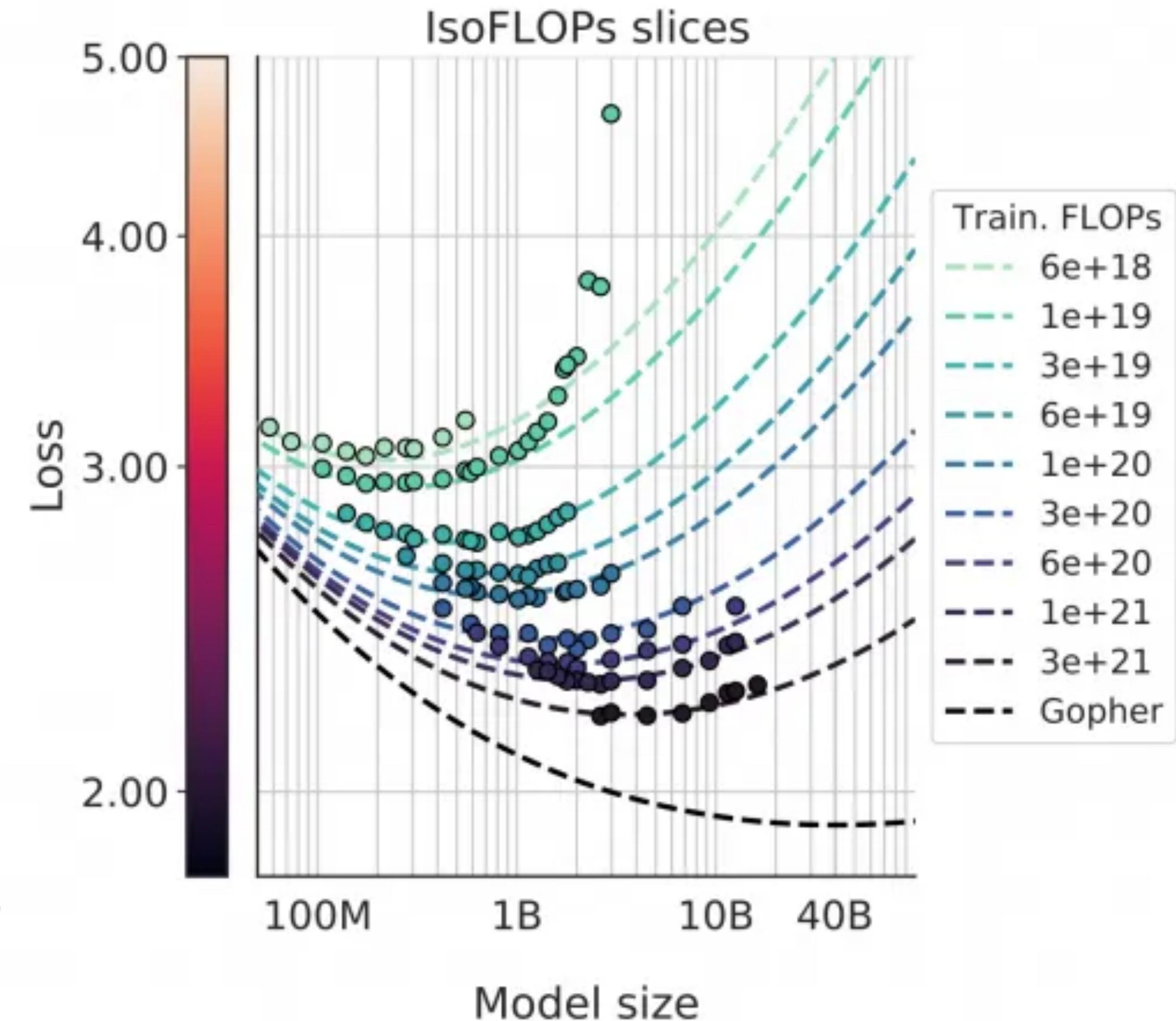


LLM Scaling Laws

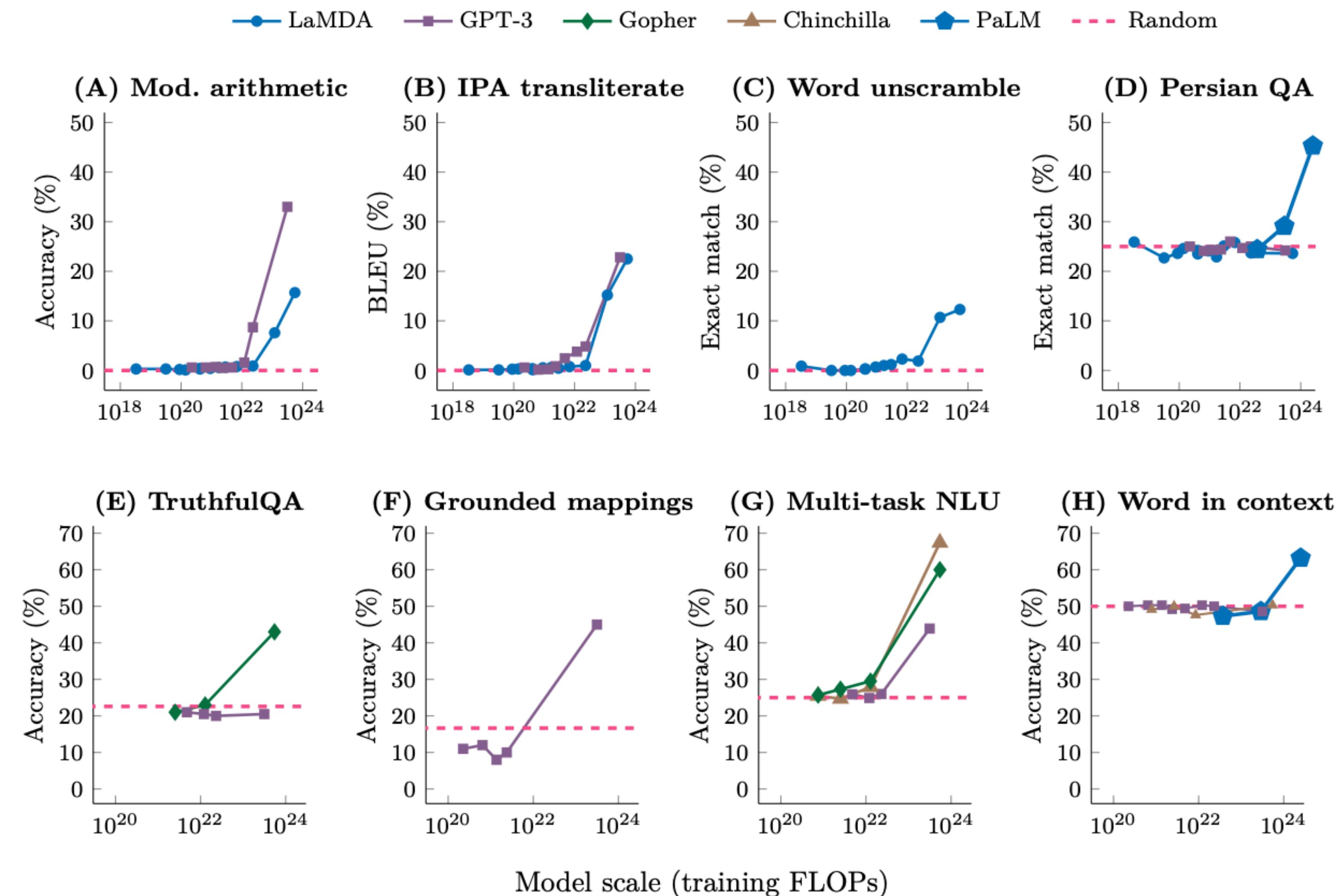
The performance of an LLM is a function of:

- N – the number of parameters in the network
- D – the amount of text we train on

We get more “intelligence” for free with scaling!



Emergent Abilities



Emergent Abilities

Emergent abilities may go beyond scaling!

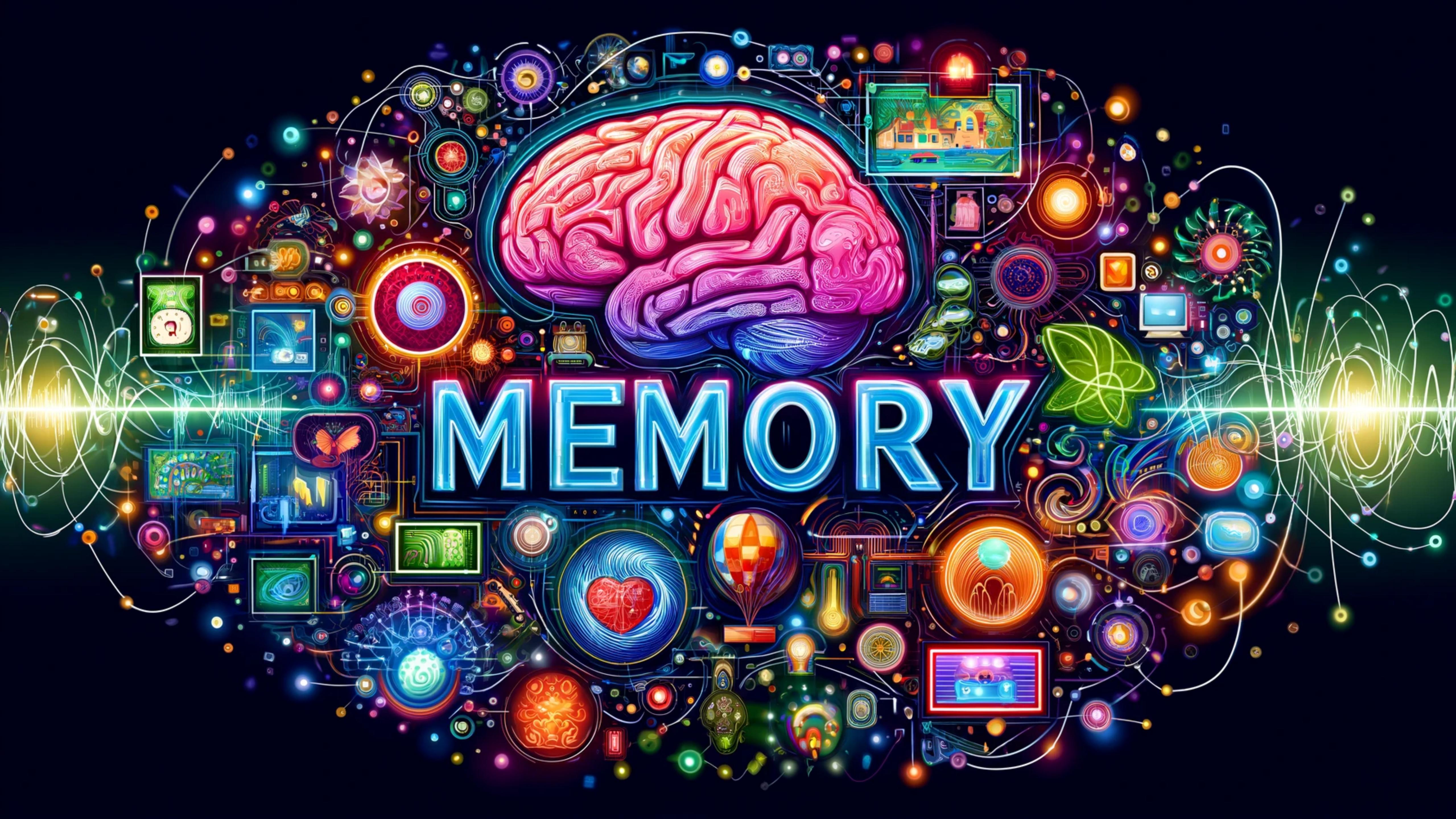
Emergent abilities are also influenced by:

- new architectures
- higher-quality data
- access to external knowledge and tools
- enhanced training procedures

The future of LLM development may hinge on finding ways to enable smaller models develop new abilities



MEMORY



We Want Reasoning, Not Memorization

The model weights are serving dual purposes

Reasoning is the ability to pattern match across a diversity of input examples

The utility of **memorization** is actually relatively low

Choose a large model that can reason and memorize, versus just a small model that can reason - which do I pick?

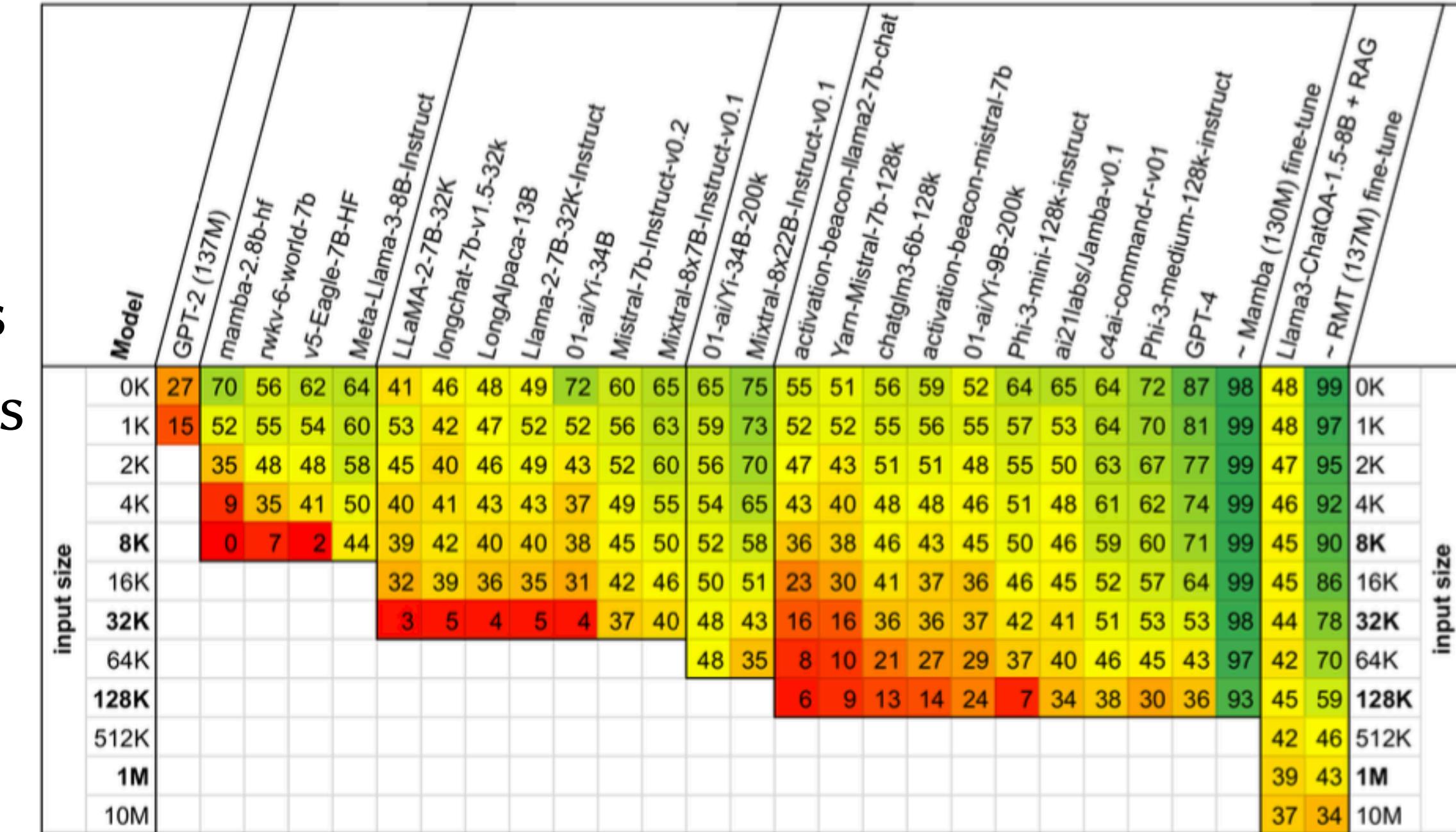


Can long context replace the need for external memory?

“Lost-in-the-Middle” effect. Liu et al. (2023) show models lose ≥ 30 pp when the same answer-bearing sentence is moved from the start or end to the middle context.

The **BABILong benchmark** scatters one- or two-fact answers across 100k -10M token documents; popular long-context LLMs tap only 10–20 % of the window.

Retrieval still outperforms pure long context. RAG baselines in BABILong retain ≈ 60 % accuracy regardless of input length, highlighting that targeted retrieval + generation remains more reliable than simply extending windows.



Retrieval-Augmented Generation (RAG)

Combines retrieval of external information with generation capabilities of language models.

Incorporates up-to-date and specific information not contained within the model

Reduces hallucinations by grounding responses in real data

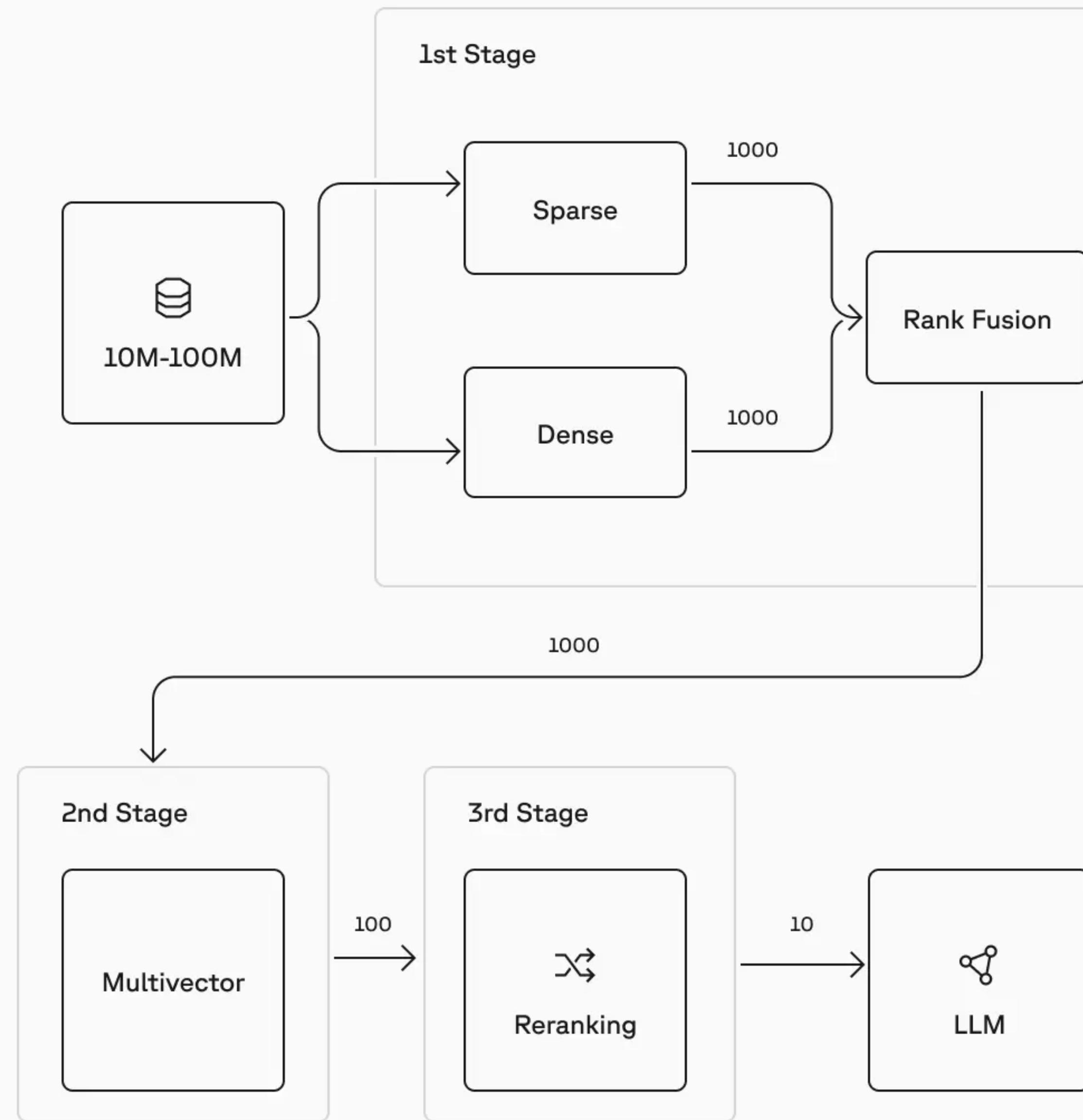


Include citations to the original sources, enhancing credibility and traceability

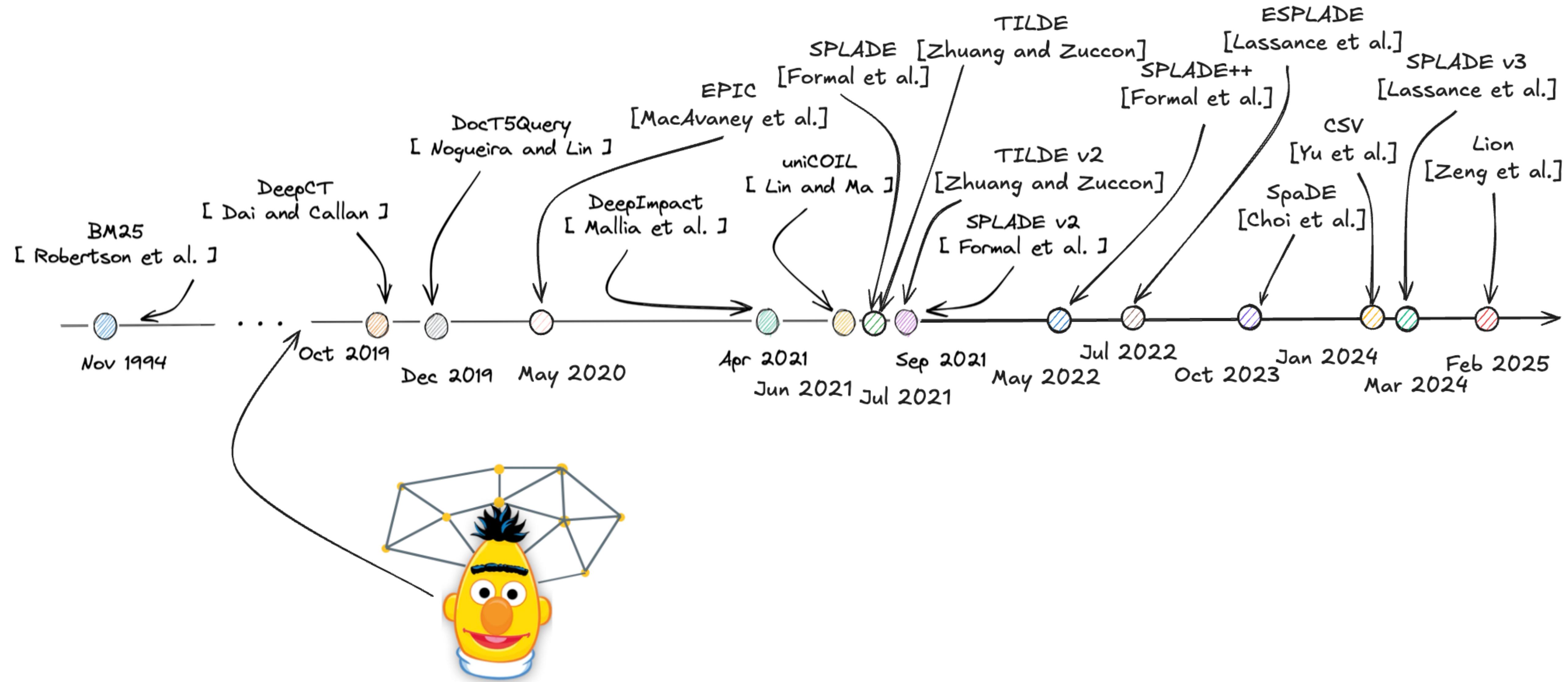
Allows models to leverage vast external databases without the need for extensive retraining

Model “does not know when it does not know”

A Standard RAG Pipeline



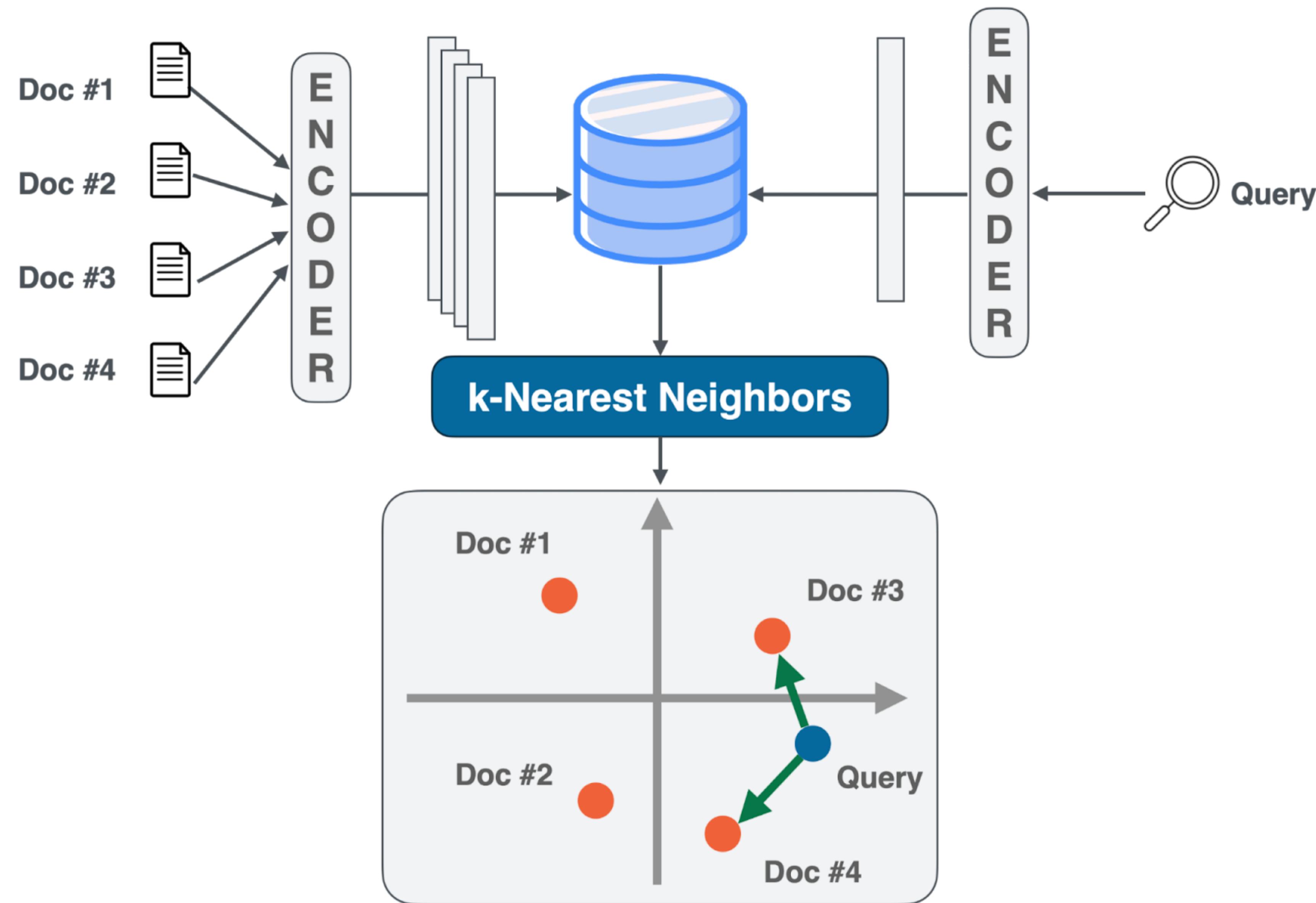
Sparse Retrieval



Semantic Similarity



Dense Retrieval



Hybrid Retrieval

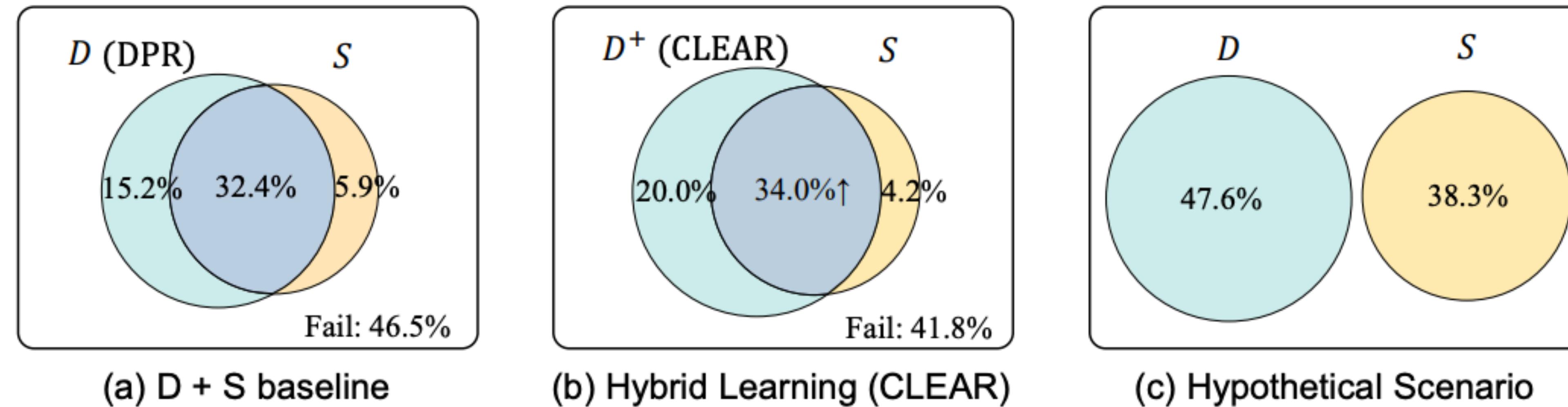
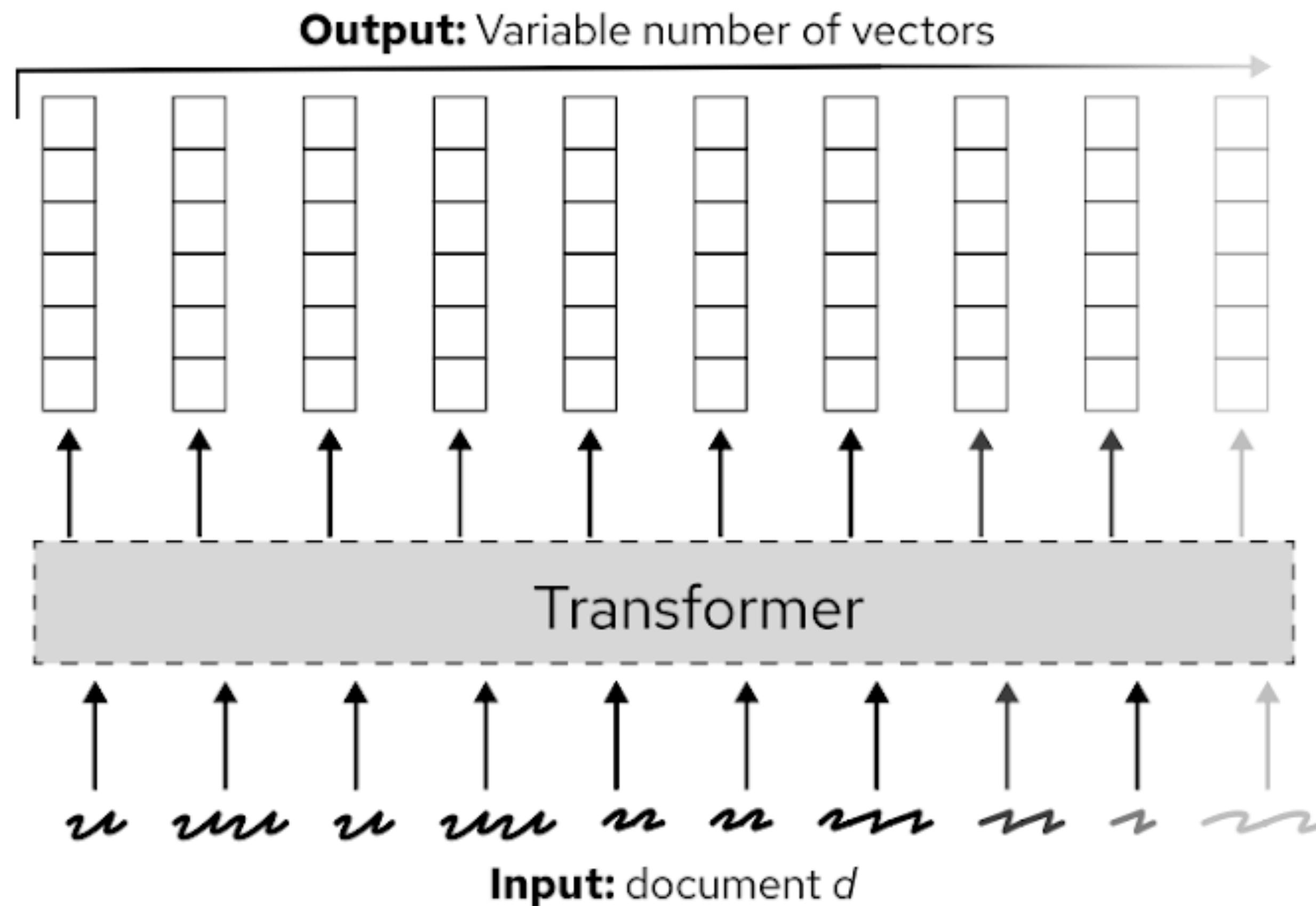


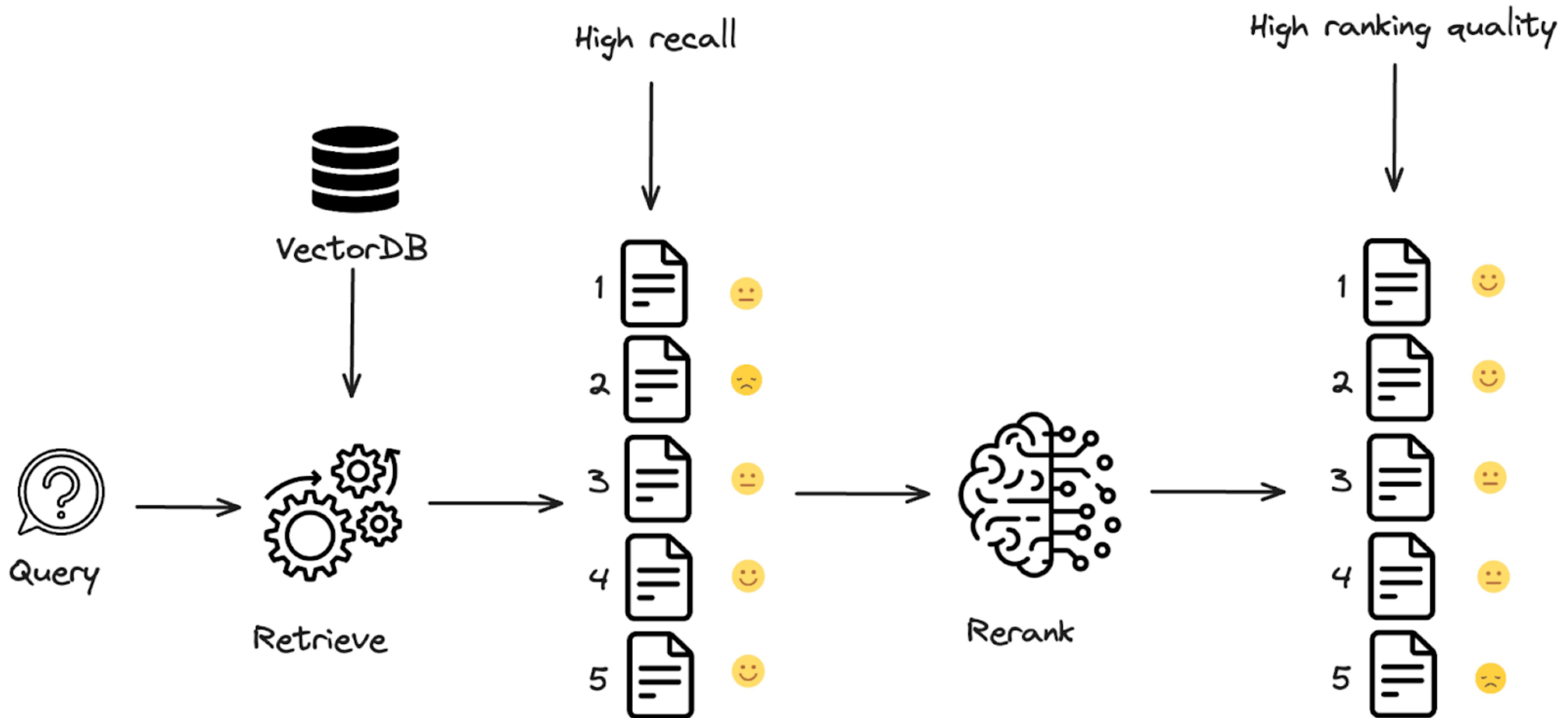
Figure 1: Recall@10 on Natural Questions. In the venn diagram, (a) shows BM25+DPR baseline and (b) shows CLEAR using residual margin. (c) is a hypothetical scenario, identical to (a) but without the intersection

Multi-vector Models

CoBERT



Reranking



From Naive RAG to Advanced RAG

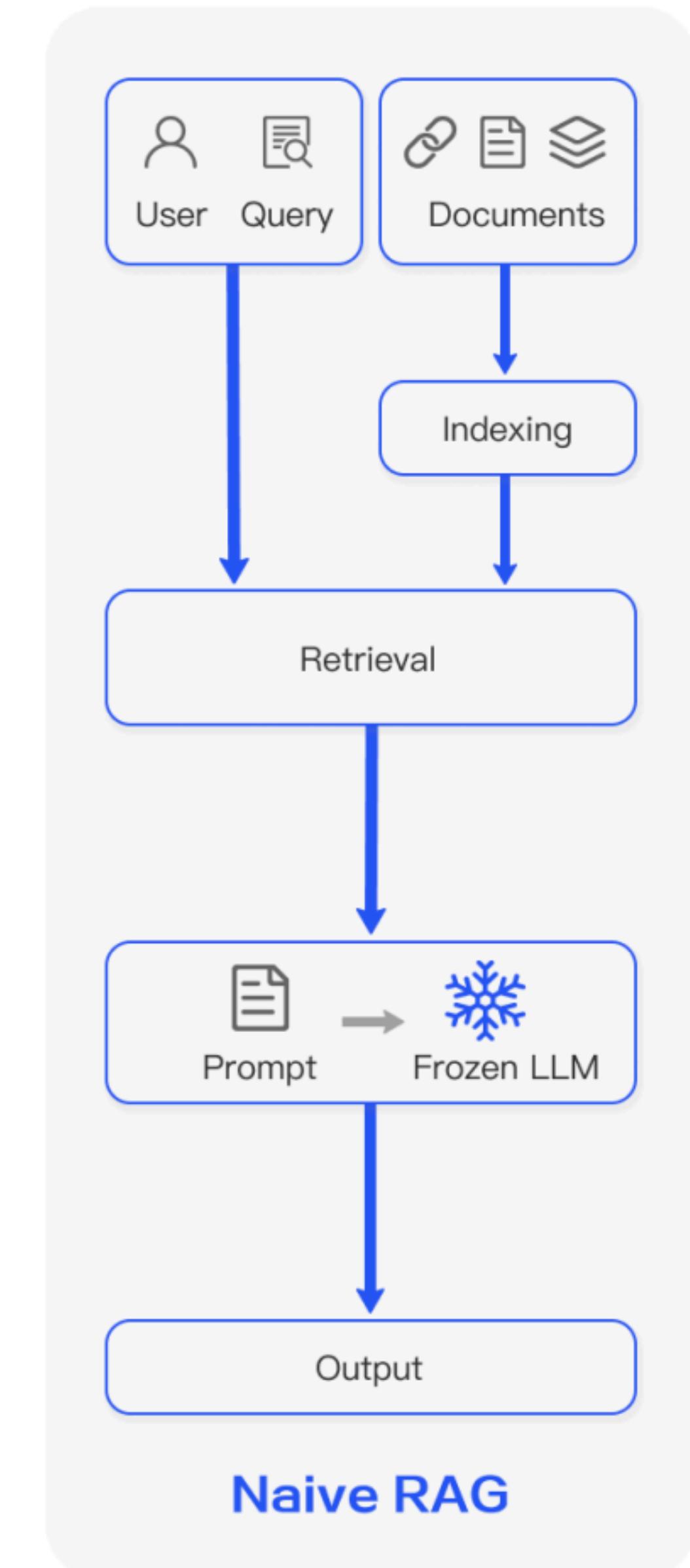
The gains are real

BUT

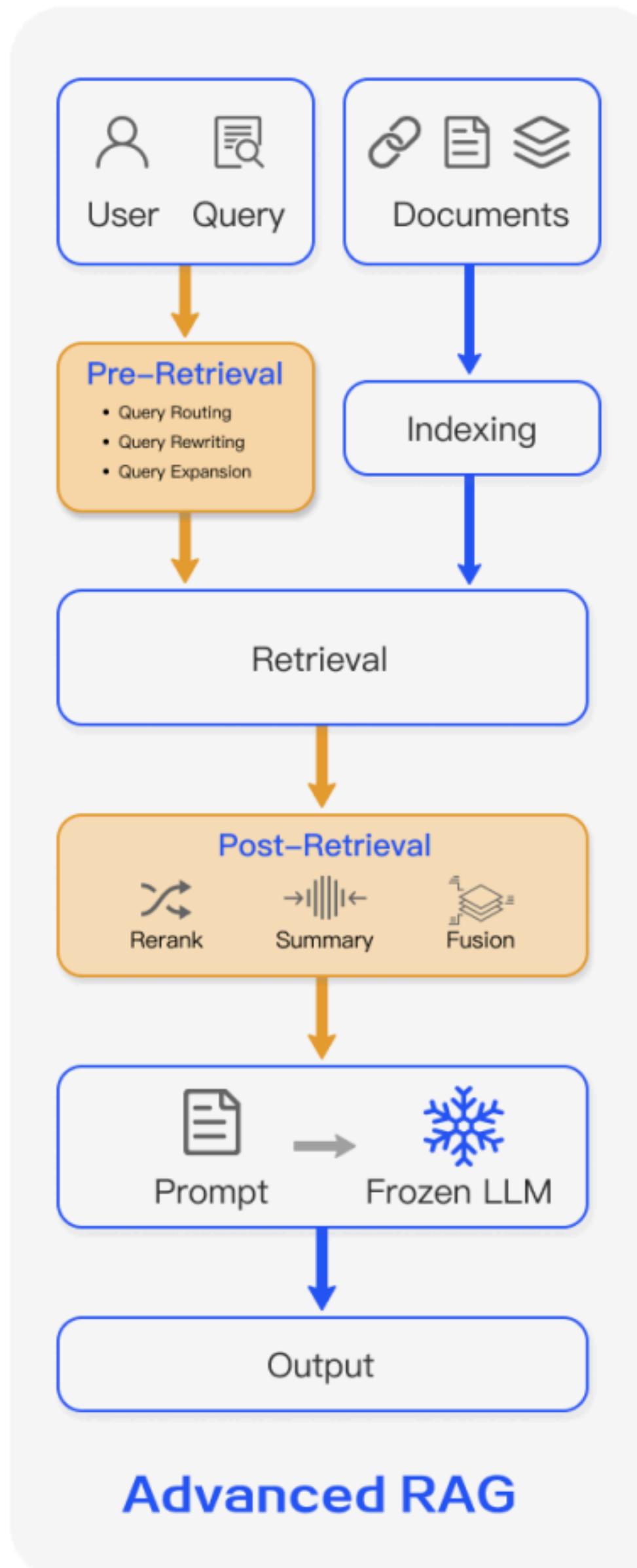
so are the trade-offs:

- latency
- engineering complexity
- infra spend
- operational risk.

At some point we need a new architectural leap rather than endlessly bolting on more patches.



Naive RAG



Advanced RAG

The Query-Length Explosion

Rewrite–Retrieve–Read–style rewriters expand a 5-word question into 40 – 80 token paraphrases.

Ma et al. "Query rewriting in retrieval-augmented large language models." EMNLP 2023.

HyDE fabricates up to 512 tokens per “pseudo-document” (often 8 per query) before the first retrieval call.

Gao et al. "Precise zero-shot dense retrieval without relevance labels." ACL. 2023.

Question-Decomposition (QD) RAG breaks one complex query into 3 – 6 sub-queries.

Ammann et al. "Question Decomposition for Retrieval-Augmented Generation." arXiv:2507.00355 (2025).

LevelRAG adds a high-level planner that iteratively rewrites until “coverage” is met.

Zhang et al. "LevelRAG: Enhancing Retrieval-Augmented Generation with Multi-hop Logic Planning over Rewriting Augmented Searchers." arXiv:2502.18139 (2025).

Context Pruning

Token costs shift left – the retriever now dominates the token bill

Question: Can you eat pumpkin every day?

Retrieved context:

Pumpkin is at its peak in the fall. Eating pumpkin every day can help reduce inflammation, strengthen your immune system and promote a health. It may also help lower blood pressure. However, if you eat too much, you may experience diarrhea from a high dose of fiber. Read on to learn all about pumpkin's nutrition and health benefits.

Relevance score:

0.96

PROVENCE

Provence removes sentences that are irrelevant to the user question

...and also outputs a relevance score

What Do We Measure?

Retrieval scientist holds a ruler labeled “**nDCG**”: classic IR relevance.

Answer scientist kneels with a clipboard “**EM / F1 / BLEU**”: correctness.

Attribution scientist peers through a magnifier: **faithfulness** checks.



Hallucination by Distraction: Why Non-Relevant Docs Matter

Non-relevant ≠ neutral

Bad abandonment studies in classical IR show users quit when “**egregiously non-relevant**” results appear.

They distinguish plausibly vs egregiously off-topic docs, a distinction most RAG pipelines still ignore.

Moffat & Wicaksono, SIGIR ’18

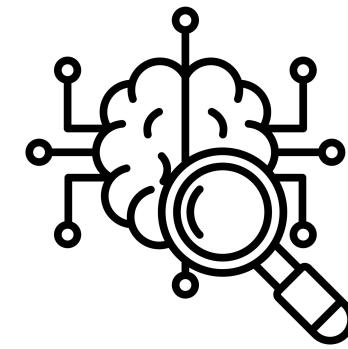
Accuracy can recover if the off-topic doc is replaced by pure random noise (dilutes distraction)

Cuconasu et al., SIGIR 2024

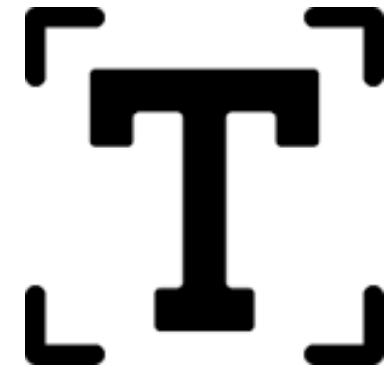
Reranker abstention: recent work adds a reject option for low-confidence passages instead of a rank order.

Gisserot-Boukhlef et al., arXiv:2402.12997 2024

Problems with RAG



Lack of seamless interaction between retriever and generator



Text as context is not *first-class citizen* for an LLM



Corpus is almost static (minimal changes) processed by LLM over and over



Hard to define search relevance for an LLM as we do for users

Let's go back to the LLMs

The Transformer

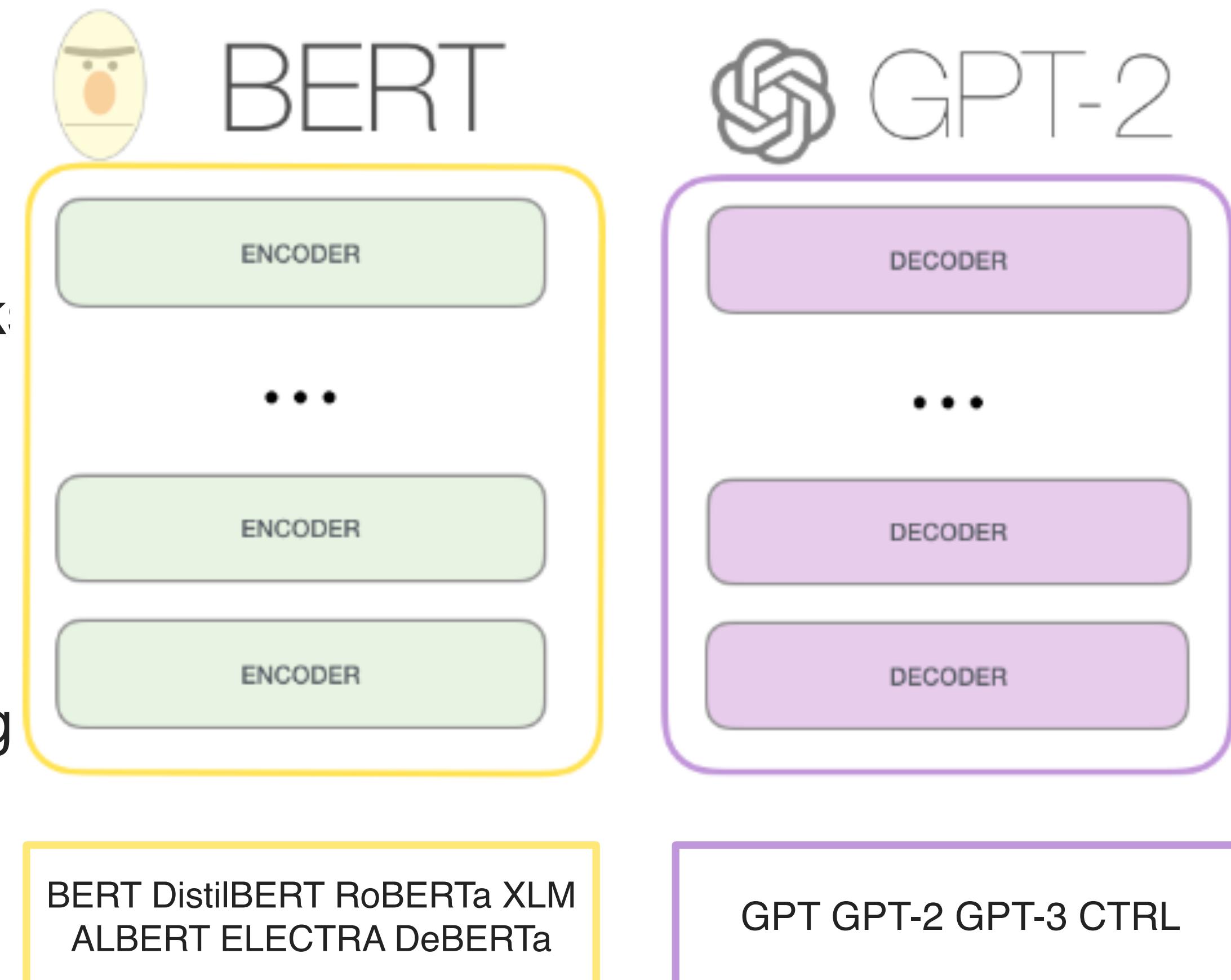
The original transformer model is composed of an **encoder** to process the input sequence and a **decoder** to generate the output sequence.

Encoder-only transformers can solve various NLU tasks

Trained using Masked Language Modeling (MLM) task.

Decoder-only transformers are mostly utilized for text-g

Trained using next token prediction.



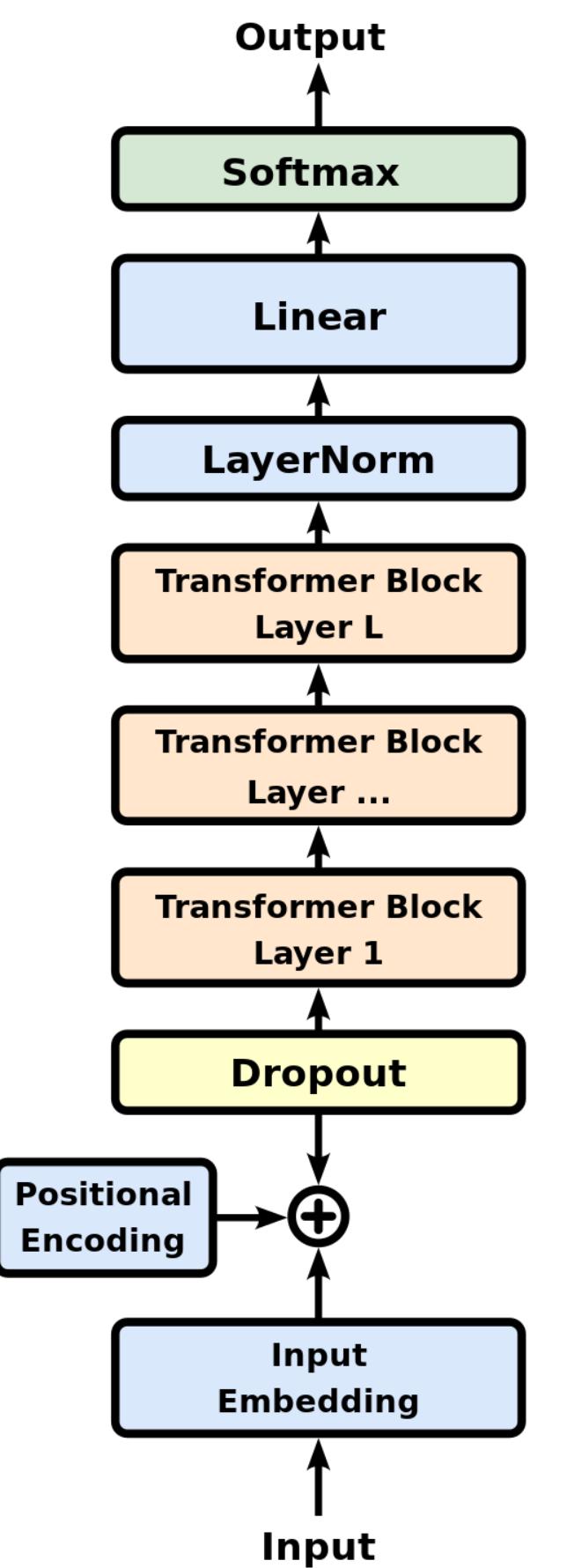
Generative Pre-trained Transformers (GPT)

GPT leverages the **decoder-only** transformer architecture

It consists of several **stacked transformer decoder layers**

GPT uses positional encodings to inject information about the relative position of tokens in the sequence

A softmax is applied to projected output to obtain a **probability distribution over the vocabulary**



Decoder Transformer Block

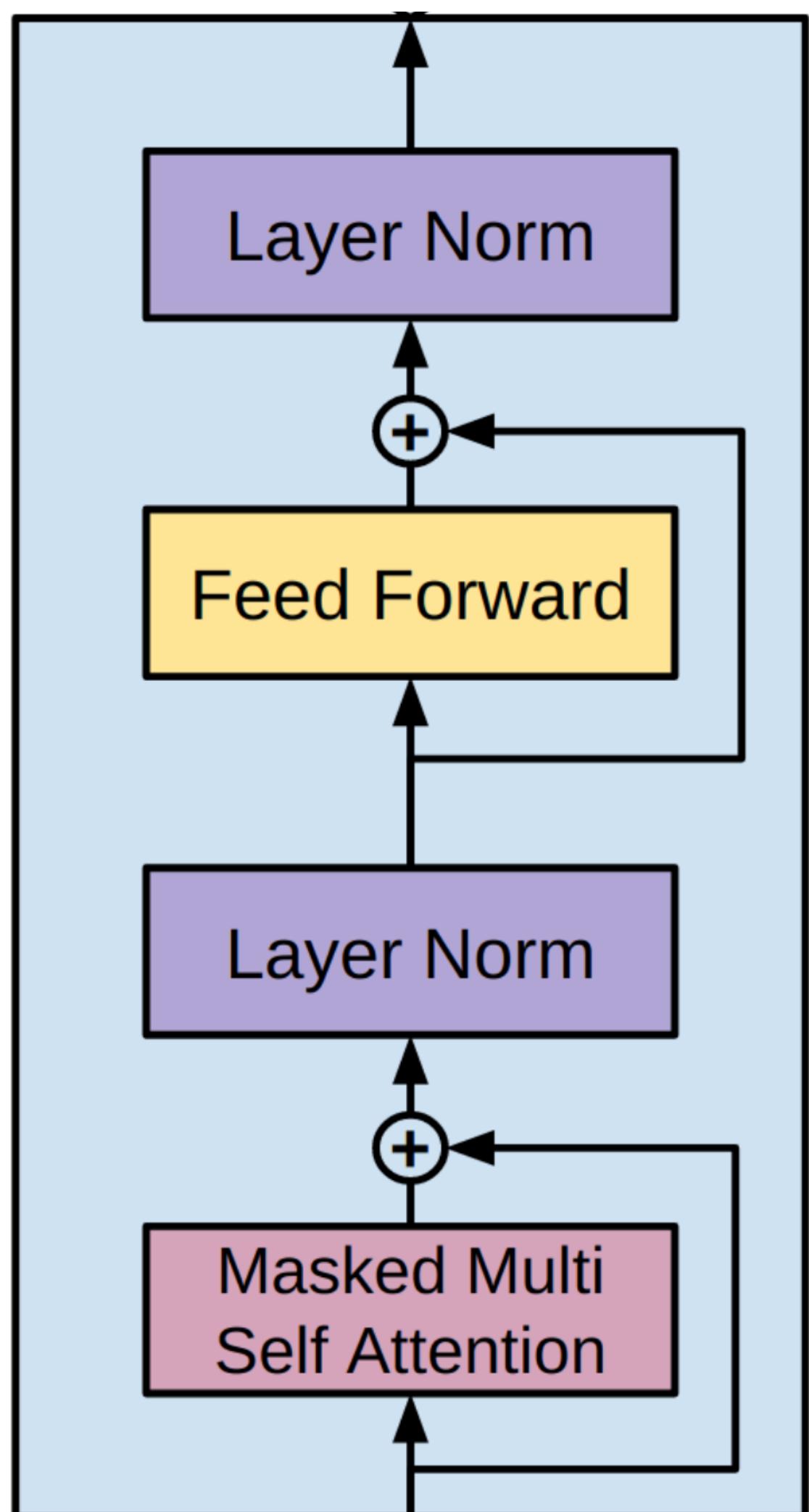
A GPT model is composed of multiple Decoder Blocks stacked on top of each other, forming **layers**.

The depth of the model, determined by the number of these layers, directly influences its capacity

The **masked self-attention** allows each position in the input sequence to attend to all other positions and ensures that the prediction for a given token only depends on known outputs.

The **feed-forward neural network** helps in learning complex patterns and representations

Each sub-layer is followed by **layer normalization** and residual connections, who help stabilize training and improve gradient flow



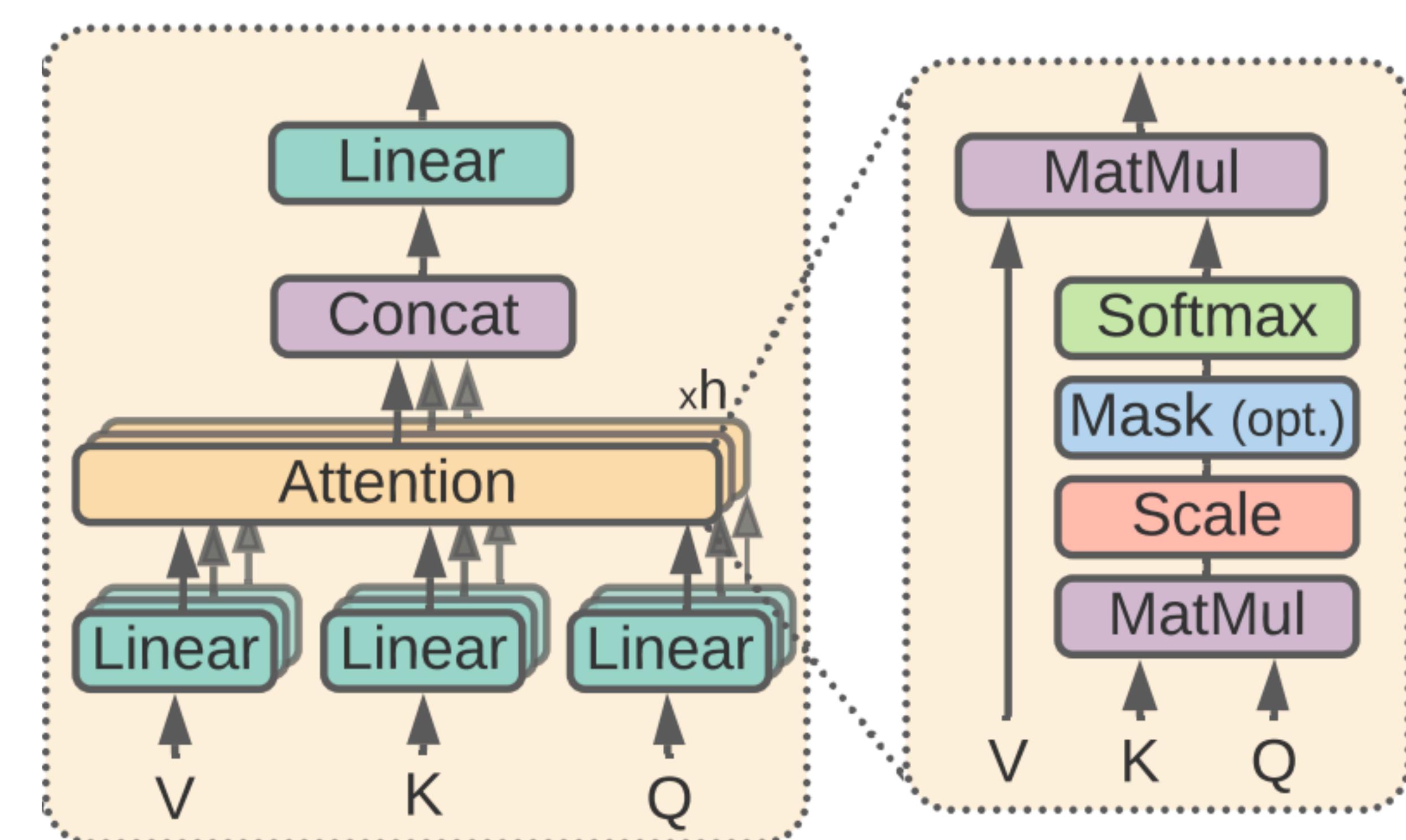
Attention Mechanism

It captures dependencies and **relationships between words** in a sentence

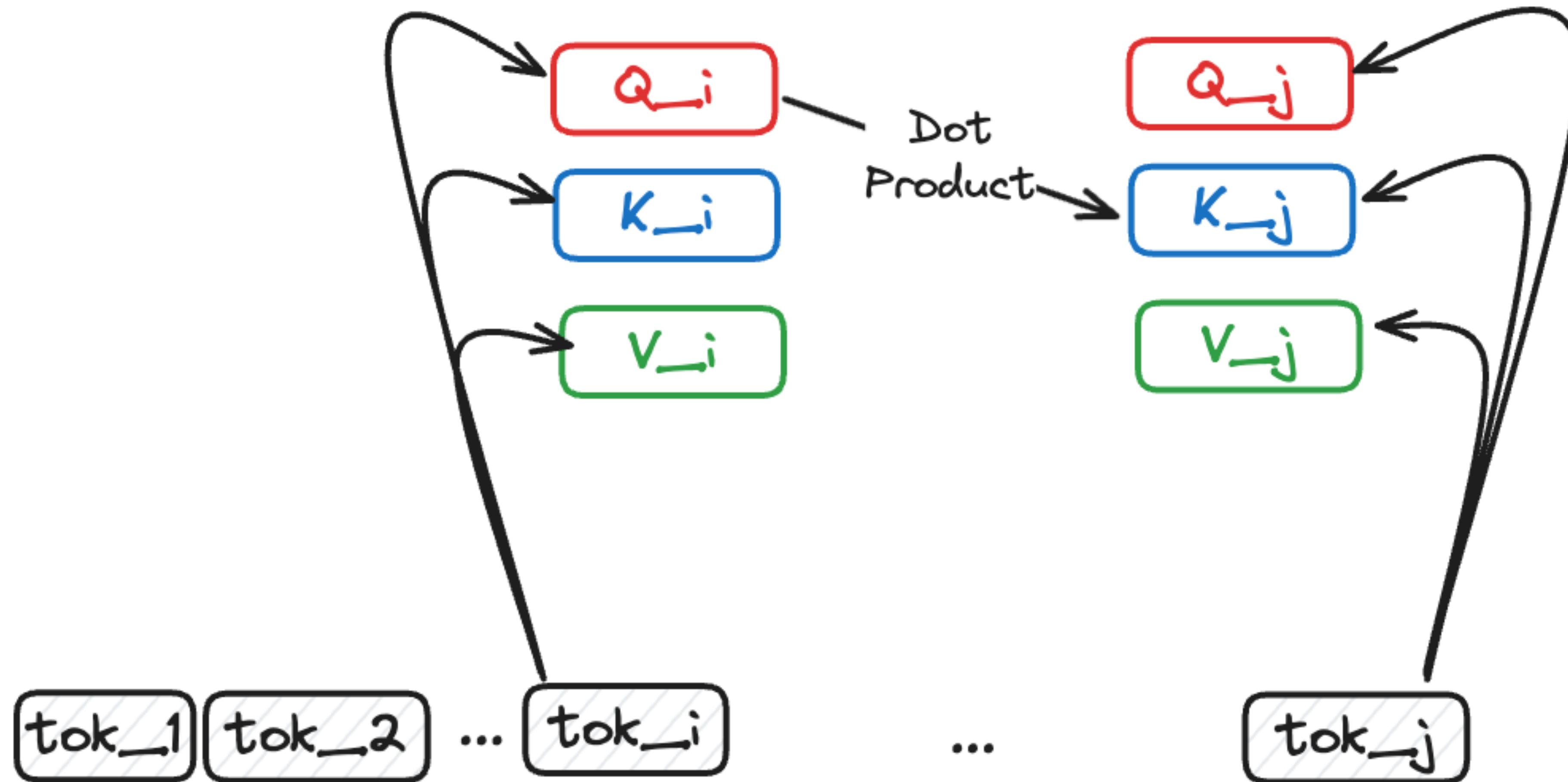
Multi-Head Attention is an extension of the attention mechanism where multiple "heads" operate in parallel

The attention mechanism involves computing **attention scores** using queries (Q), keys (K) and values (V)

The outputs from several attention heads are concatenated and processed through a linear layer



Attention Computation

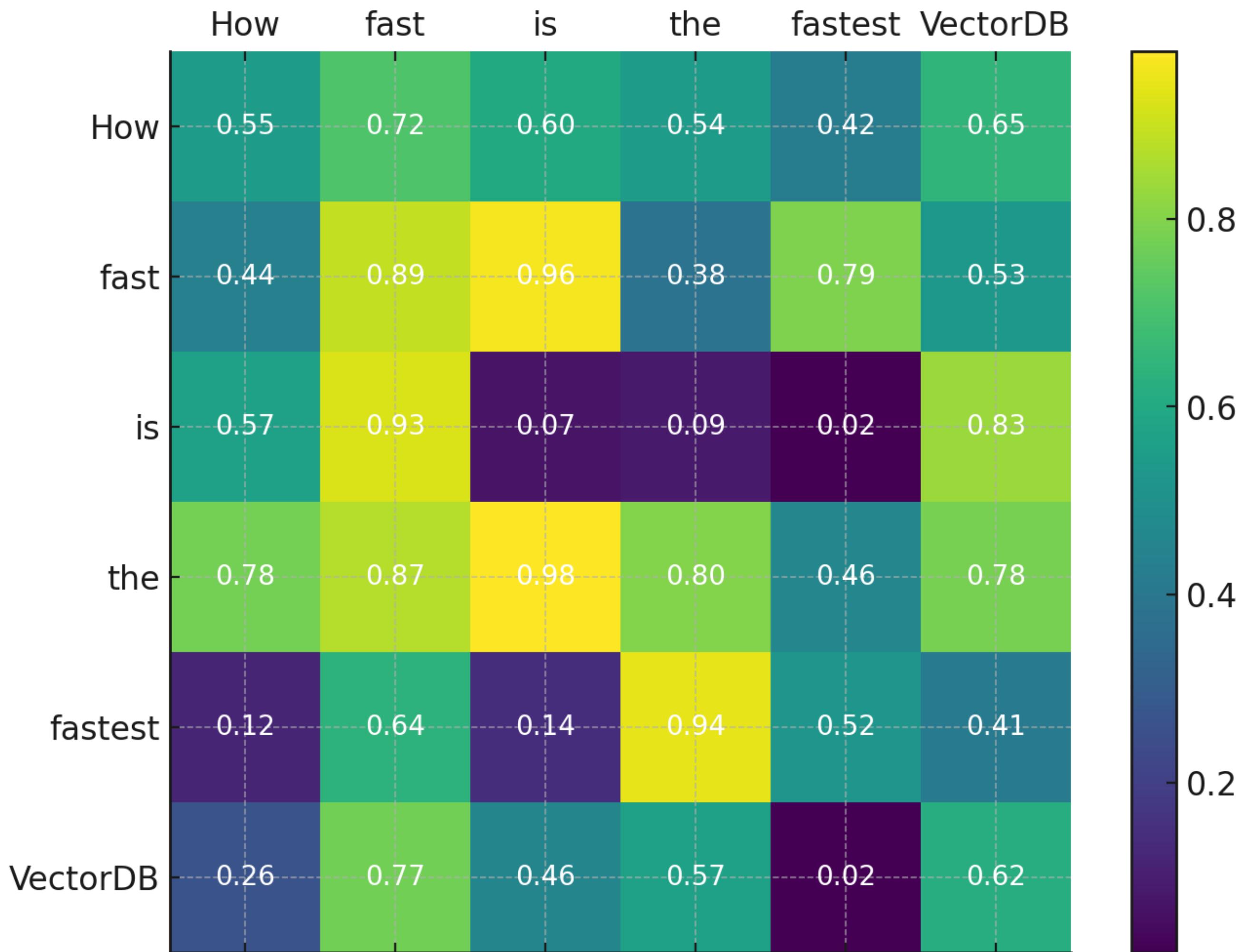


Attention Matrix

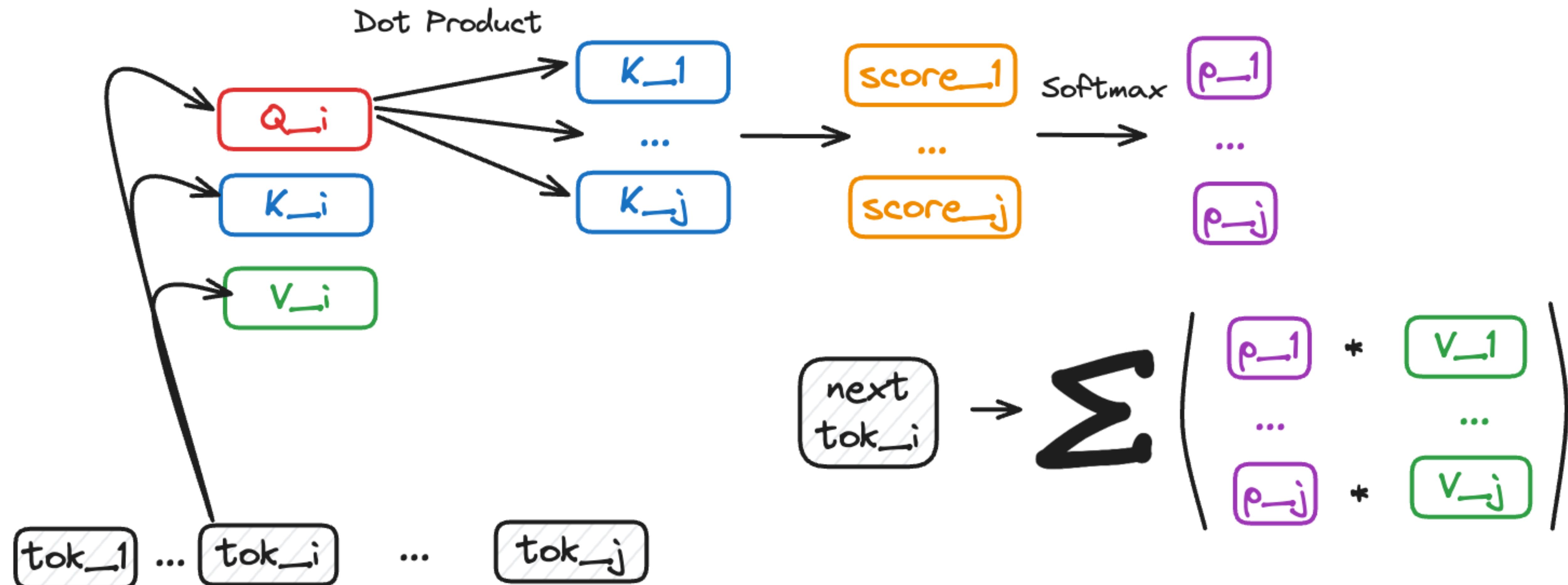
How much **focus or attention** each word gives to every other word when processing the sentence

Each value represents the **attention score** between the corresponding row and column words

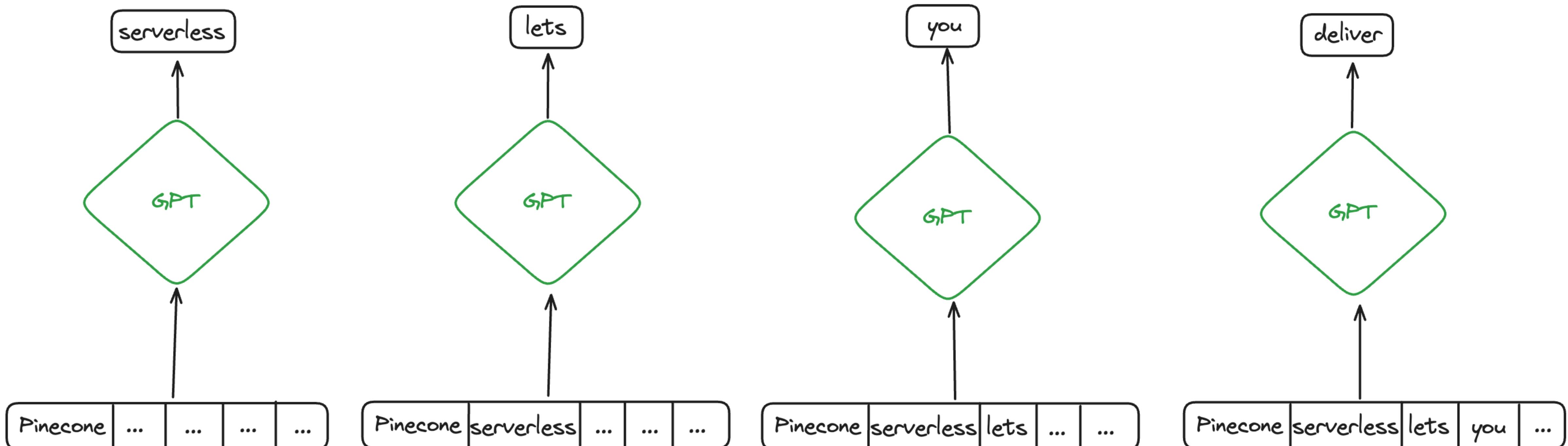
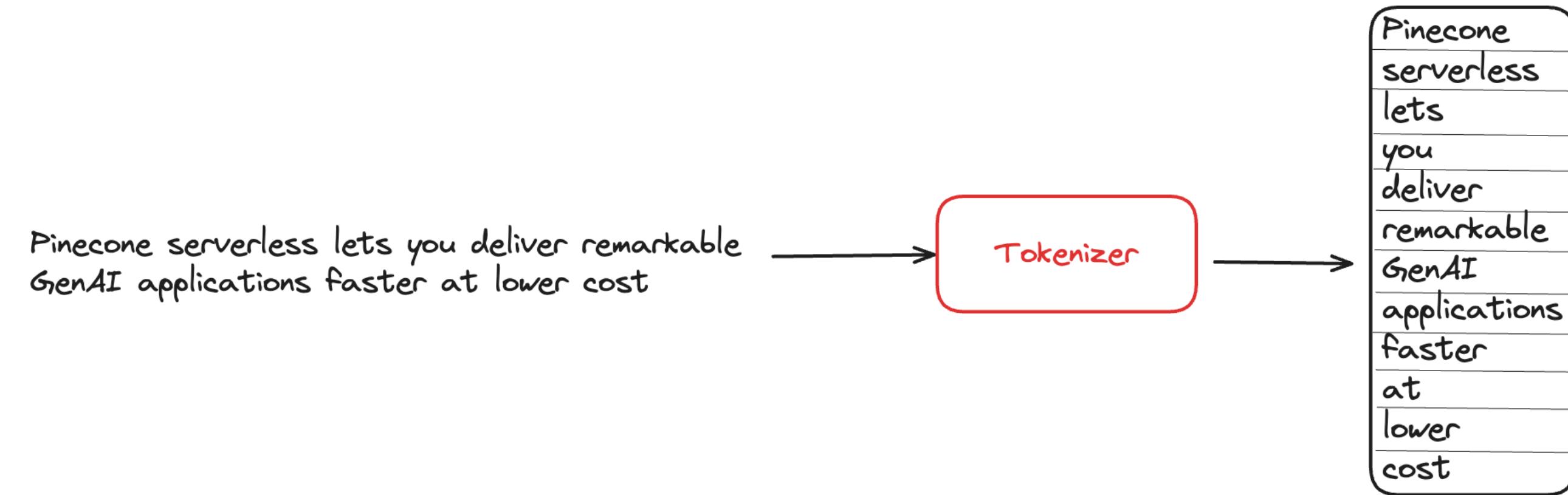
"fast" pays high attention to "is", conversely "is" pays minimal attention to "fast" showcasing the **directional nature of attention**



Next Token Score



Next Token Prediction



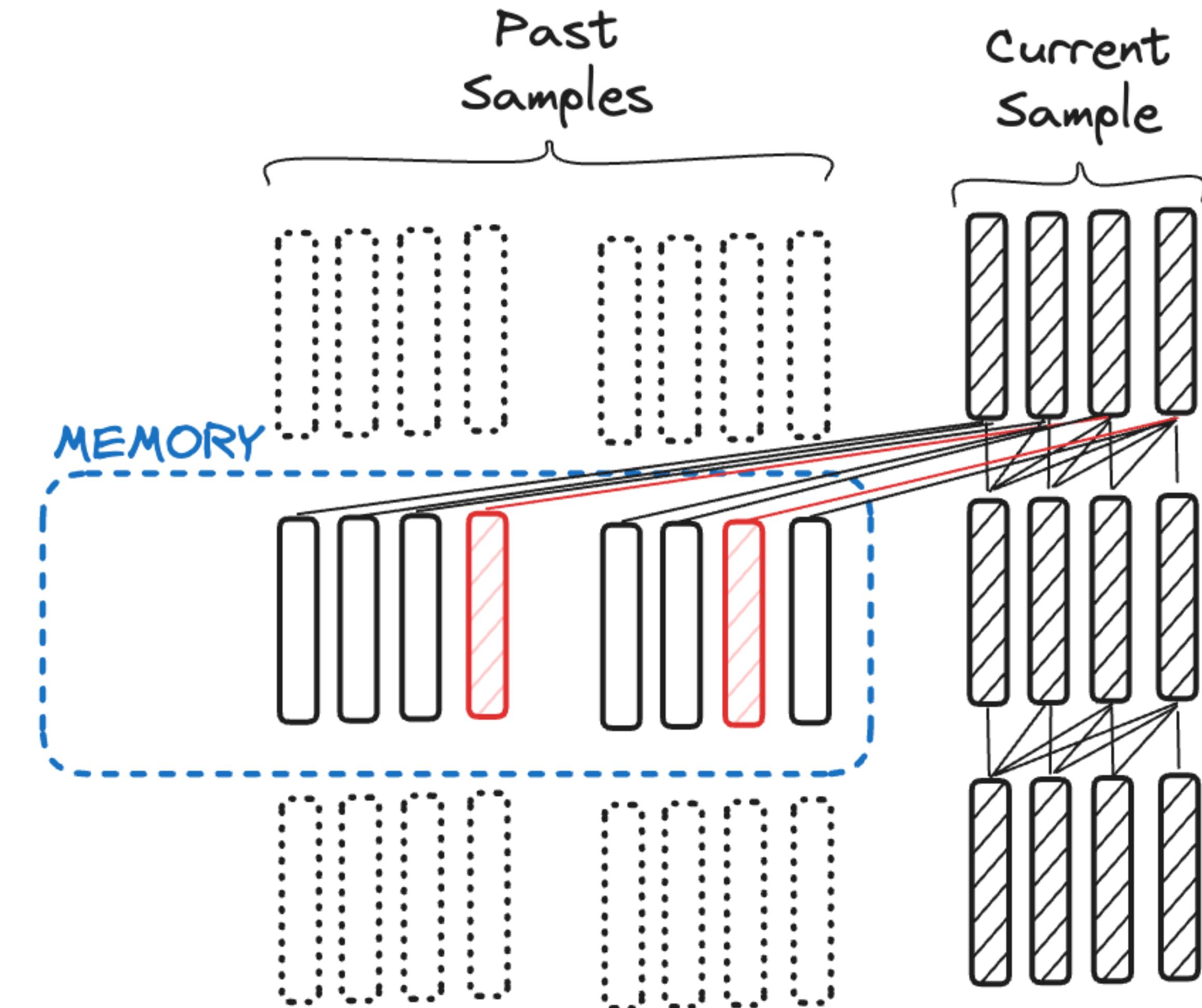
Memorizing Transformers

External Memory: Extends a standard Transformer with external memory.

kNN Lookup: During inference, the model performs an approximate k-nearest-neighbor search in the memory to retrieve relevant past contexts by matching keys.

Leverage past knowledge: attends to both the normal sequence context and the retrieved memory key–value pairs

Extended Context: Effectively gives the model a much longer memory beyond the fixed context window.



Other Attention Mechanisms

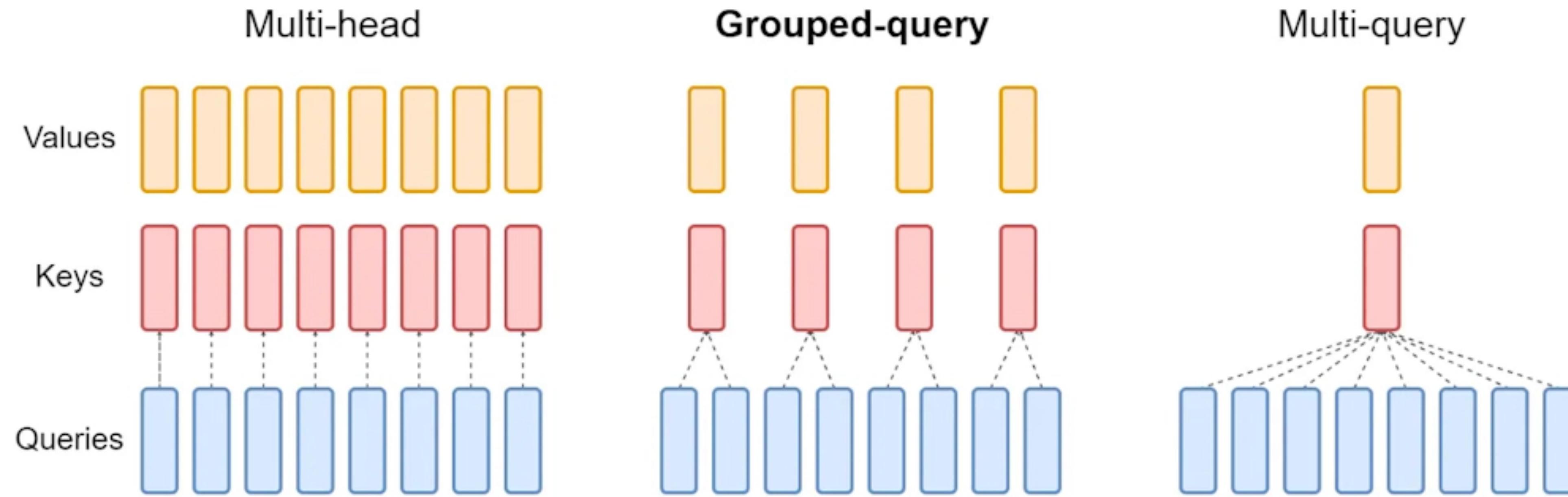


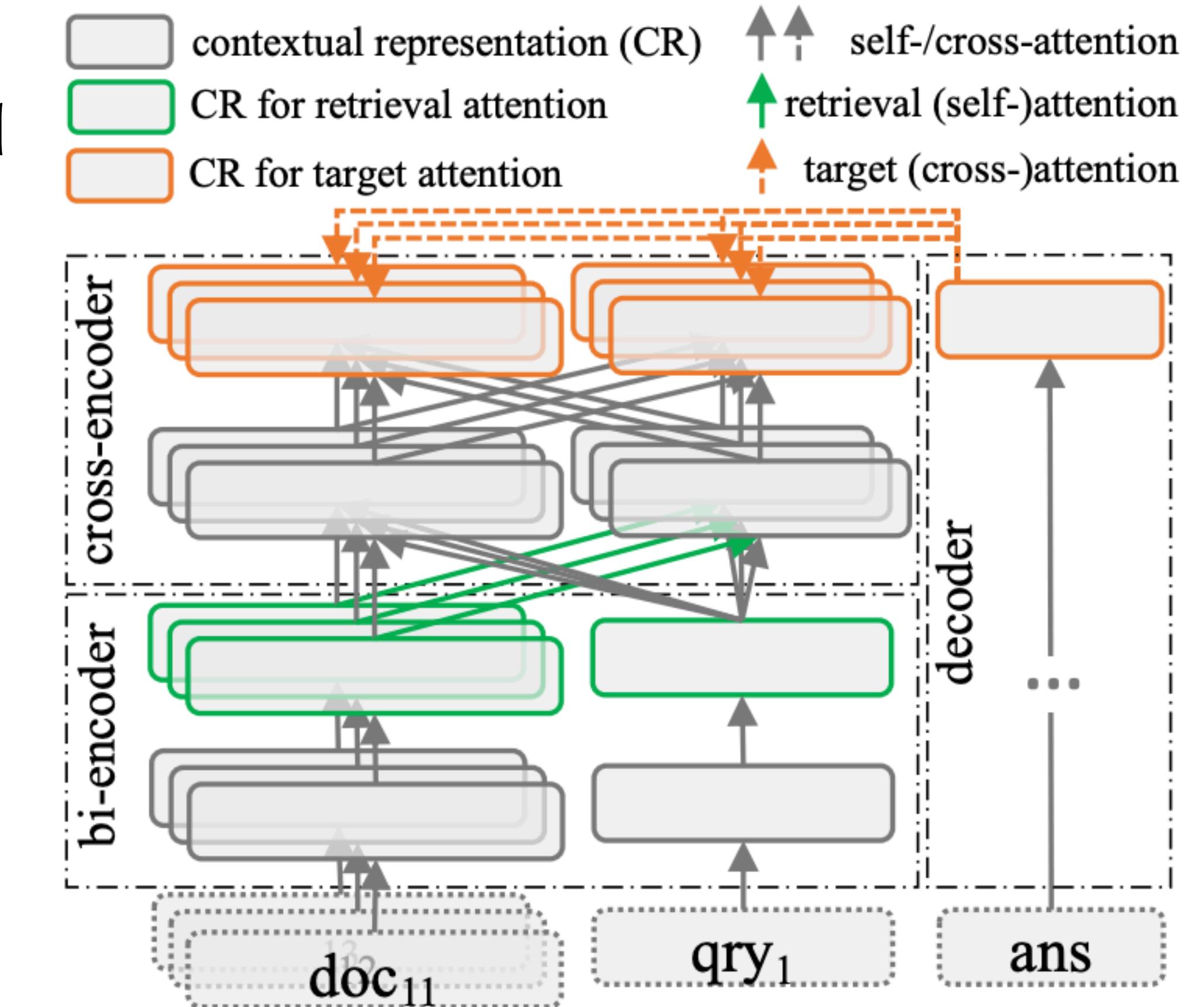
Figure 2: Overview of grouped-query method. Multi-head attention has H query, key, and value heads. Multi-query attention shares single key and value heads across all query heads. Grouped-query attention instead shares single key and value heads for each *group* of query heads, interpolating between multi-head and multi-query attention.

One Transformer, Two Jobs

Act as a retriever and as a cross-encoder reader - all inside a single model

Retrieval is just another attention head: the model retrieves and reads in one forward pass.

End-to-end supervision: no separate IR labels



Memory: A Retrieval Problem

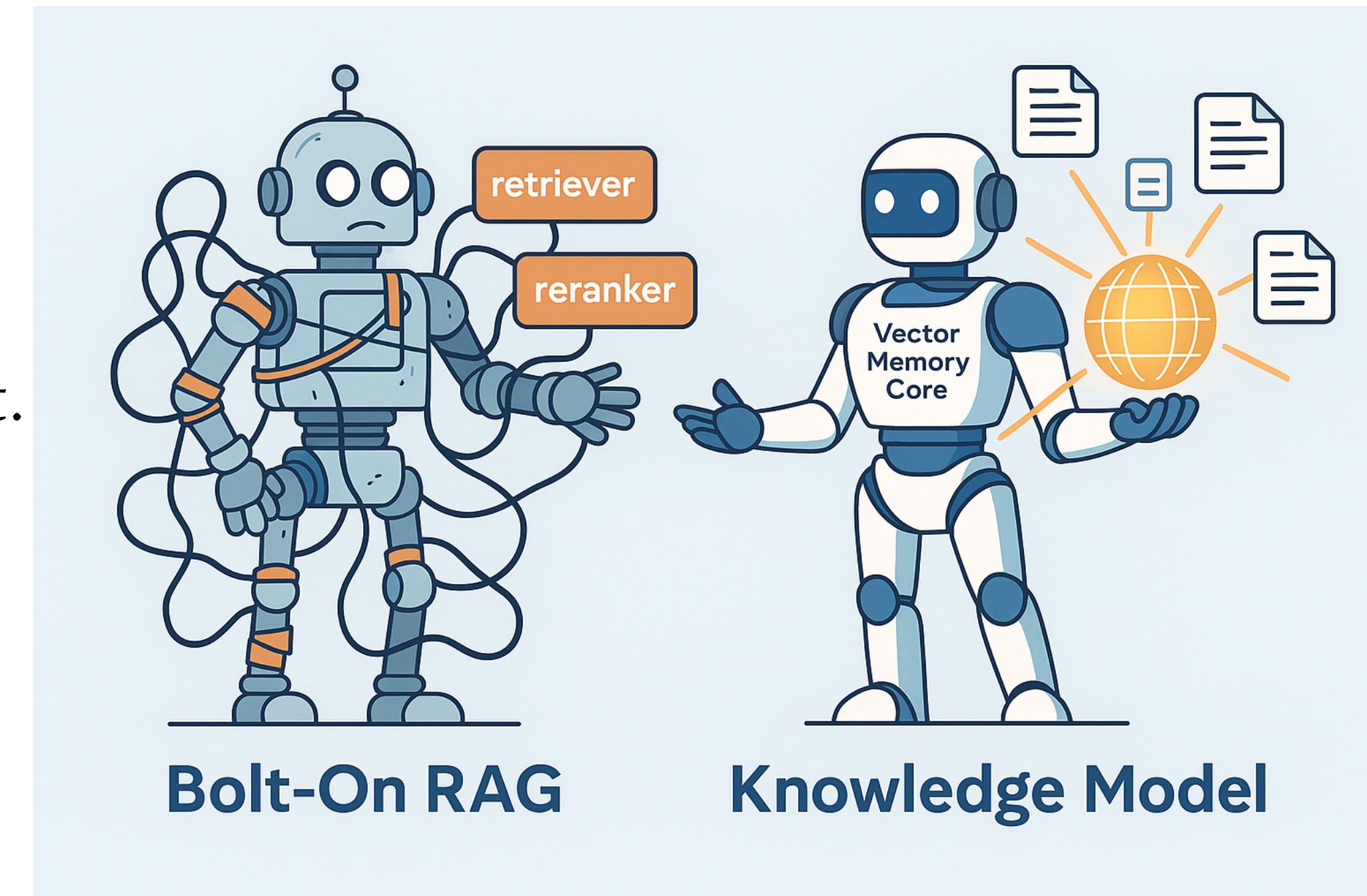
Retrieval = Attention. Embed large-scale retrieval inside the Transformer stack.

Built-in memory layer. The external datastore becomes an intrinsic component of the network..

Trillions-parameter capability, without trillion-parameter cost.

End-to-end trainable. Gradients flow through memory.

Rich research agenda: multi-query & group attention for paragraph-level keys, dynamic memory refreshing, diversified retrieval, and agentic “think-retrieve-think” loops that refine knowledge mid-generation.



Thanks!

Any questions?