ServiceNow Research

Hierarchical Retrieval at Scale

Bridging Interpretability and Efficiency

RETREEVER

ICML 2026

Gupta · Li · Chen · Subakan · Reddy · Taslakian · Zantedeschi

ServiceNow Research · Université Laval · Mila · McGill

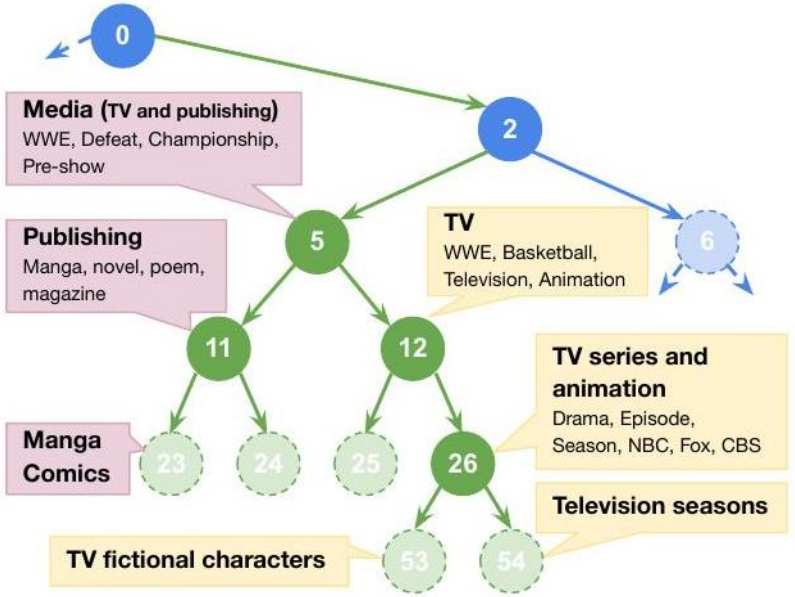
Hierarchical retrieval

Dense retrieval is fast but a black box. Hierarchical methods organize data into an inspectable structure.

What hierarchy gives you

Data organization
nodes of progressively finer semantic groupings, preserving the structure (document sections, entity relationships)

Interpretability
can probe nodes to see what they contain and why they lead to certain content (correct or incorrect)



Hierarchical retrieval

Dense retrieval is fast but a black box. Hierarchical methods organize data into an inspectable structure.

What hierarchy gives you

Data organization
nodes of progressively finer semantic groupings, preserving the structure (document sections, entity relationships)

Interpretability
can probe nodes to see what they contain and why they lead to certain content (correct or incorrect)

But does not scale

LLM-bound
prompt an LLM per node of the hierarchy

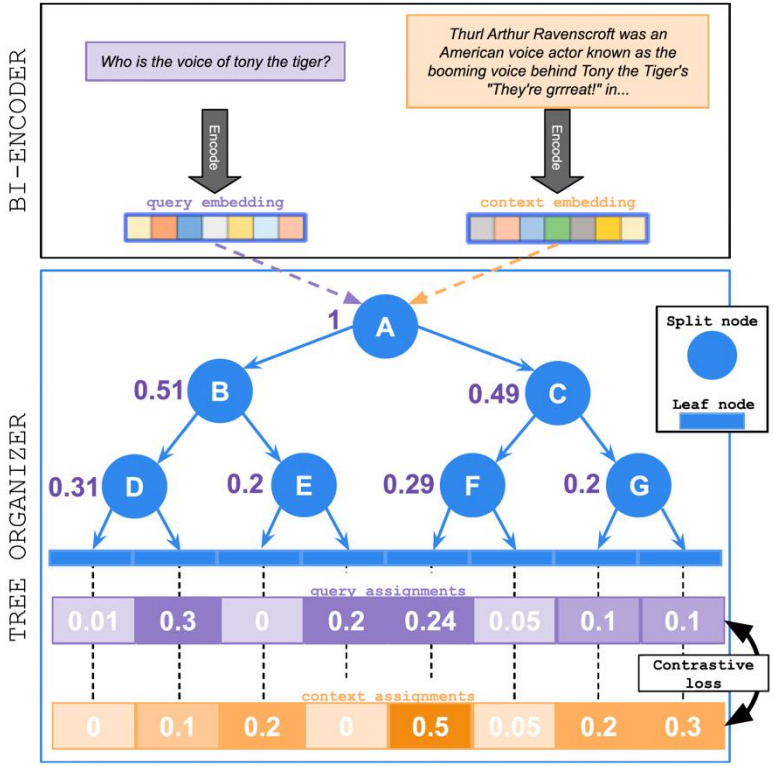
Slower than dense retrieval
RAPTOR takes ~1.7 s/query on NQ

OUR WORK builds and navigates the tree entirely on embeddings — no LLM calls.

*Sarthi, et al ICLR 2024 — **Raptor: Recursive abstractive processing for tree-organized retrieval***

The idea: a tree learned for retrieval

Learn a binary tree that organizes embeddings into semantic hierarchies, trained end-to-end for retrieval.



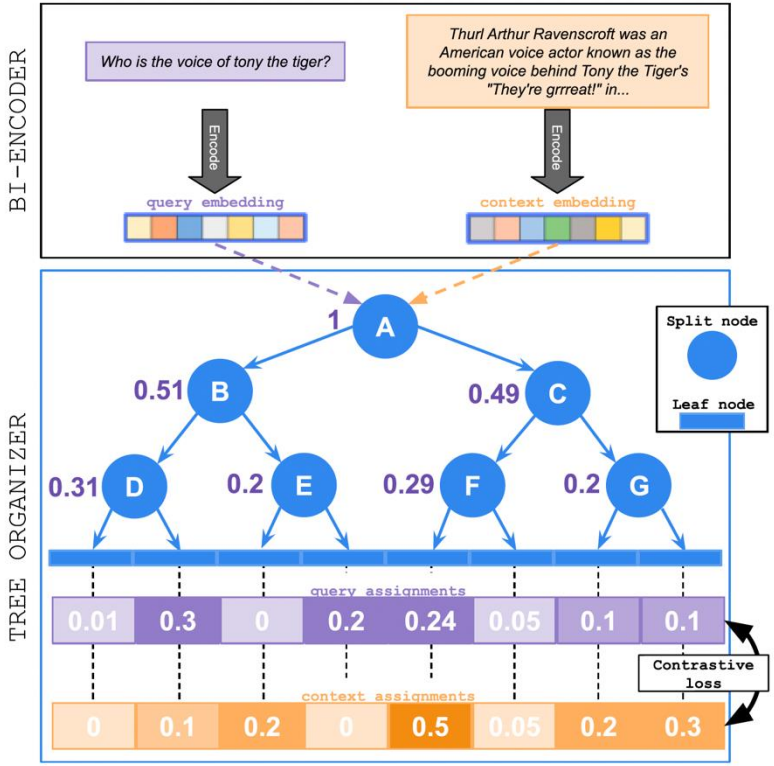
Operates entirely on embeddings
no LLM calls, scalability

Produces coarse-to-fine representations
Can trade speed for accuracy

Documents group semantically
Interpretability

The idea: a tree learned for retrieval

Learn a binary tree that organizes embeddings into semantic hierarchies, trained end-to-end for retrieval.



In practice:

Node outputs a probability of the text belonging to that node

Probability multiply along the path to the leaf (fully differentiable)

Contrastive loss pulls positive (query, reference) pairs to the same leaf

Best accuracy at the lowest latency

No accuracy penalty

Accuracy improves
over base encoder

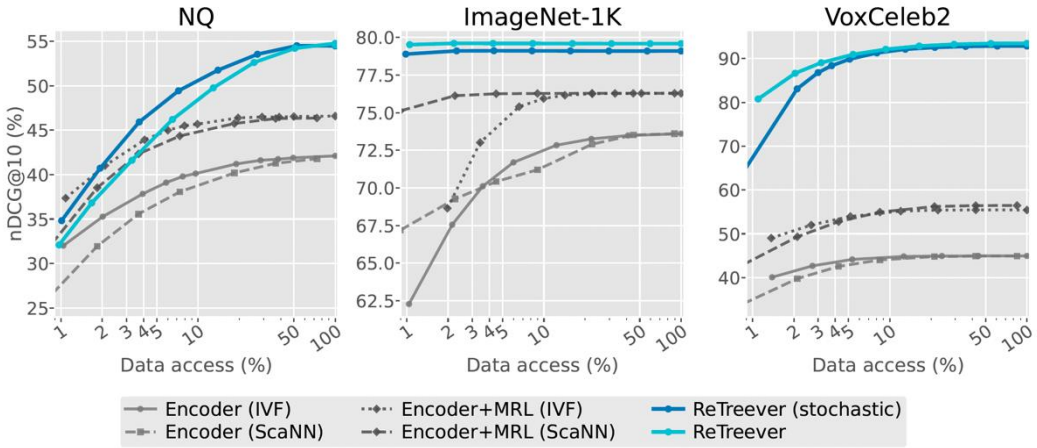
Fast inference

160x faster than RAPTOR
with +30 pts NDCG@10

Better than ANN

Also at low data budgets
vs. IVF and ScaNN

And it works across modalities:



Scales to 124.5M documents

on Reddit Title–Body (~300× bigger than NQ): the learned tree gives a memory-friendly multi-index alternative to global ANN.

ScaNN ran out of memory (>1 TB RAM)

RETREEVER is built in 14 min at 1024 dim, or 2 min at 256 dim.

Tree stays well-used

No empty leaves across 1024.
Routing entropy 8.4 bits (max 10).