

Mis 3: Vector DB search is exact \times
ANN indices (like hnNSW) is approximate ✓
 $(Top-k)***$

Misconception: KNN is just a tool, where you decide:

- i) k value (optimal)
- ii) sim threshold
- iii) filtering logic
- iv) fallback behaviour

* Improving Search Accuracy

- ① Improve Acc by choosing the Better K.

optimal k } → Direct factual — $k = 1-2$
 } → Descriptive — $k = 3-5$
 } → Broad summary — $k = 5-8$

② Sim Threshold :

if sim < 0.55:

③ Query Rewriting : ignore chunk

④ Impact on Retrieval chunks : (Debugging skills) activate different semantic regions of an LLM.

① KNN + Metadata filtering:

→ Improve Precision.

RBAC makes the project real.

Role Based Access Control

allowed



Eg:

Employee

Role: finance Analyst X



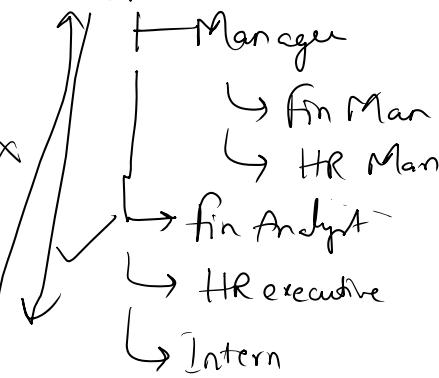
X Permissions :
→ read finance_docs
→ read reports



Role hierarchy

Eg:

Admin



Hierarchy gives us clean permission inheritance.

Level

- read - can view Content
- write - can create/update
- delete - can remove/delete
- admin - full control

* Role Permissions

json structure :

→ {

"roles": {

"admin": {

"inherits": [],

"permissions": ["*"]

wildcard

"manager": {

"inherits": [],

"permissions": [

"read: finance",

"read: hr",

"read: reports"

Pipeline:

User Query



User Role



Allowed Permissions



filter document chunks by metadata



... | | | | | | |

↓
kNN search only on allowed
chunks (top k)
↓
LLM Answer

↑
"read: hr",
"read: reports"
3,
fn
hr...
int - -