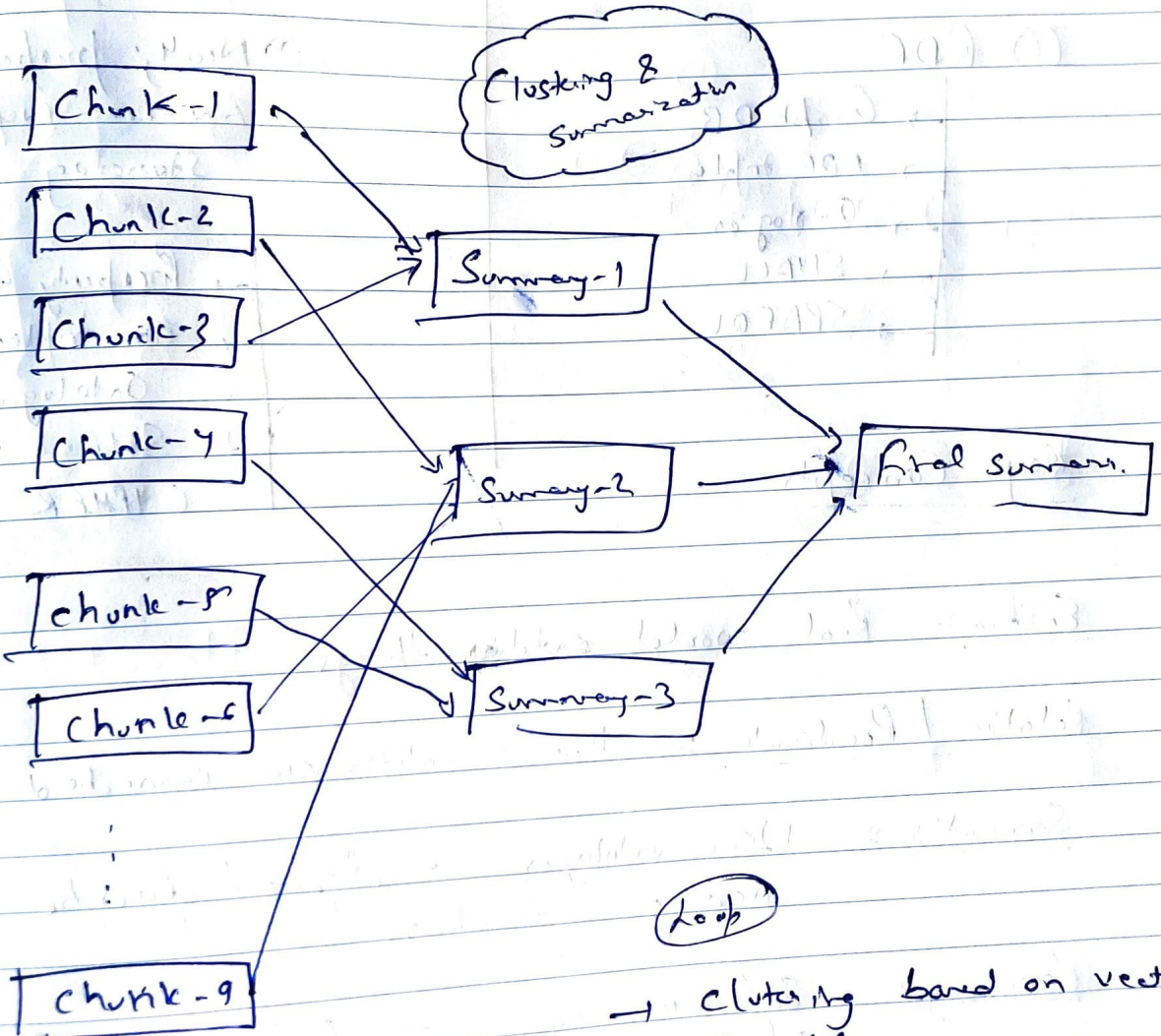
Meeting -

Goal :- GRAPH RAG

Approach :- RAPTOR INFLUENCED KG CONSTRUCTION

RAPTOR



- clustering based on vector Embedding
- Summarization until we get a top level summary.

KG Graph

Structured data → CSV, SQL → KG → KG construction [Easy]

Unstructured data → text, audio, video, images

Tech stack

① RDF
- Graph DB
- RDF triples
- Ontologies
- SHACL
- SPARQL

Tech stack

② Neo4j; property graphs
- Amazon Neptune
- Stardog

- Property Graph DB
- triples
- Ontologies
- SHACL (based on)
- CYPHER

Key Concepts

Entities :- Real world ~~entities~~ things.

Relation / Predicate :- How 2 entities are connected.

Semantics :- User ontologies to define / describe the meaning of thing.

Reasoning :- Inference engine can deduce new facts
{Eg:- If A is a manager, & All managers are employee
A → employee }

Subject → Predicate → Object

(Must be an entity)

(can be a entity or literal)

↓

Not a real world entity (attributes: age, salary)

Experience :- Without Ontology no path to success.

This approach

→ Take all the KG chunks & summaries as nodes in

⇒ Few Admits

- No KG can stand alone to replace text based RAG (Even Microsoft GraphRAG)
- → KG supports text based RAG to improve its performance (How??) ⇒ by retrieving hidden relationships blw entities & thereby foster the context provided to the LLM to generate final answer.

{ chunk-1, chunk-2, ⋮, chunk-n } + { KG } ⇒ Prompt

Retrieve doc + KG

↓

LLM

↓

Answer

Deduplication
&
Relationship
Conflict resolution
⇓
Graph fusion

Current KG Structure

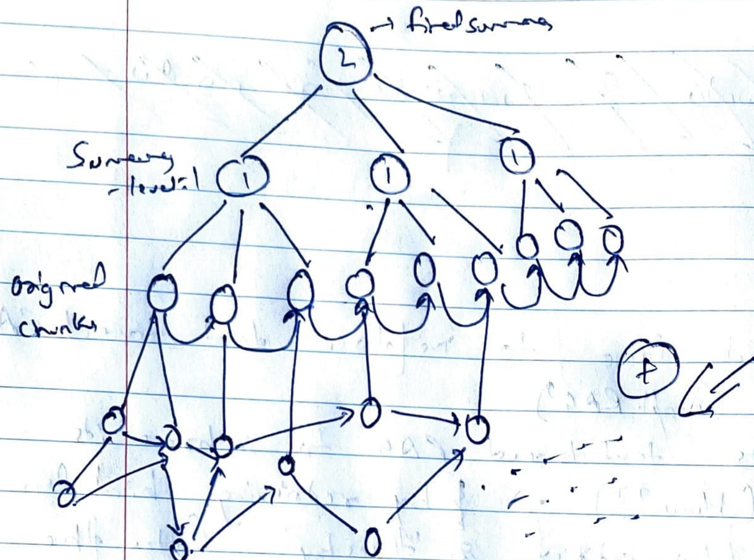
↓

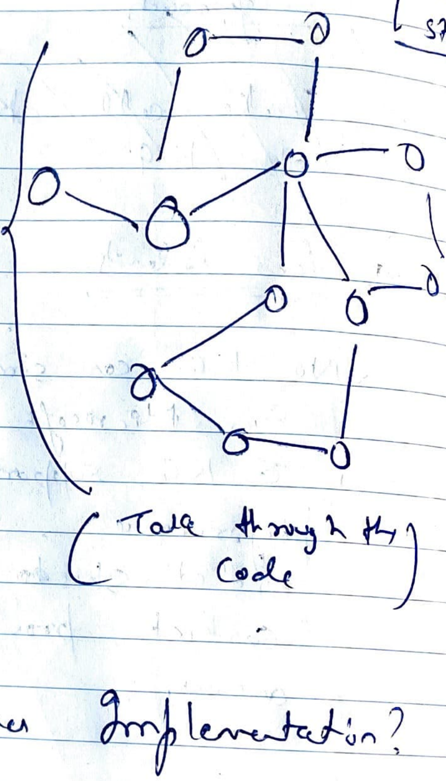
2-Layer KG

Document-structure Graph
(Lexical Graph)

Domain based Graph
(Actual KG)

{ Influenced by Ontologies } ⇓ [W3C Standard]





(Talk through the Code)

Doubts for further Implementation?

① KG for summaries?
(Redundancies in domain KG) ??

② Retrieval strategies

- Vector based
- (KG) cypher based
- both

Vectorbased ⇒ Retrieve relevant chunks (Eg:- 5)

↓

for Each chunk sub KG as an additional context.

& then leverage Graph traversal to find relevant chunks and KG triples that help us to generate better answer.

Cypher based ⇒ Entity Extraction

↓

Dynamic Cypher Query

↓

Retrieves relevant KG triples

↓

as an additional context (Retrieve chunks)