Faculty of Engineering
Cairo University

Advanced Database Systems

#CMP4030

Project Design Document

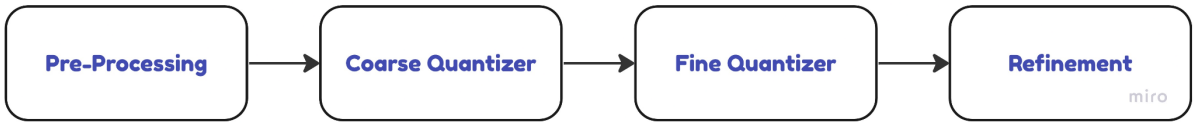
Team Members

Name	Email
Yasmine Ashraf Ghanem	yasmine.ghanem01@eng-st.cu.edu.eg
Yasmin Abdullah Nasser Elgendi	yasminelgendi@gmail.com
Sarah Mohamed Hossam Elzayat	sarahelzayat@outlook.com
Ahmed Sayed Sayed Madbouly	ahmedmadbouly186@gmail.com

Indexing System

The idea was to build a composite indexing algorithm to capture the advantages and limit the shortcomings of each algorithm creating a more efficient index structure that approximates the nearest neighbor concept.

1. **Architecture**



a. *Pre-processing*
   Also referred to as vector transformation where the main idea is to reduce the dimensionality of the vectors in the dataset for a faster indexing and searching technique.

b. *Coarse Quantizer*
   The main algorithm that divides vectors into subdomains to limit the search scope when searching for similar vectors resulting in a faster search.

c. *Fine Quantizer*
   Further compression of index size by applying finer compression of vectors into smaller domains. For example, by giving each vector a code.

d. *Refinement ?*
   Post-processing step done at search time for which it rearranges the results according to the original flat vectors.

2. Algorithms & Data Structures

a. *Pre-processing*
(lesa msh 3arfa awi mehtag el hayakhodha yedawar aktar)
The algorithms available:
PCA, OPQ, L2 Norm

b. *Coarse Quantizer* ***(Choose one)***
LSH: For the coarse quantizer the algorithm used is Locality Sensitive Hashing (LSH). LSH is a hashing technique that uses hash tables and hash functions to group similar vectors together in a hash bucket which limits the search scope.

HNSW: For the coarse quantizer the algorithm used is Hierarchical Navigable Small World (HNSW). The idea is to use graphs to represent vectors as nodes with the similar nodes linked together through branches referred to as "friends". The hierarchical part is represented by dividing the nodes into layers depending on the **Probability Skip List** that gives some nodes higher probability to be in a certain layer.

IVF: For the coarse quantizer the algorithm used is Inverted File Index (IVF).By using clusters and inverted lists it allocates each vector to a cluster where it is closest to its centroid using some measure (cosine similarity ghaleban or distance)

c. *Fine Quantizer*
Using Product Quantization we aim to compress the vector within each hash bucket/graph layer/ivf cell for memory usage reduction without changing its position.

d. *Refinement*
RFlat algorithm

Reasoning & Trade-offs

Algorithm	Reasoning	Trade-offs
LSH	1. The use of hash functions can result in a faster search as opposed to other techniques 2. Scales well with large datasets	1. Higher level of approximation where the accuracy/recall depends on the use of a good hash function. 2. Uses less memory.
HNSW	1. Provides more accurate results and hence higher accuracy/recall and captures both the local and global relationship of data. 2. Handles large datasets and high dimensionality	1. Building the hierarchical graph structure when indexing the vectors can be computationally expensive. 2. Higher memory usage for higher accuracy/recall
IVF	1. Combining IVF with other quantization techniques improves efficiency. It can be organized hierarchically for better performance. 2. Relatively simple and easy to build. 3. Suitable for large scale vectors.	1. Index size can become relatively large depending on the input parameters. 2. Easy to build and when combined with quantization can yield efficient results.

Conclusion

---

1. Data Characteristics
    - LSH may be suitable for datasets with inherent locality, where similar vectors are likely to be close in space.
    - HNSW might perform well for datasets with complex relationships that can be captured in a graph structure.
    - IVF is effective when data can be efficiently quantized and organized hierarchically.
2. Search Requirement
    - If an approximate nearest neighbor search is acceptable, LSH and HNSW are strong candidates.
    - IVF is suitable when accurate results are desired, and the hierarchical organization allows for efficient pruning.
3. Scalability

    LSH, HNSW, and IVF can all scale to large datasets, but the efficiency may depend on the specific characteristics of the data.
4. Dimensionality
    - LSH and HNSW are known for their effectiveness in high-dimensional spaces.
    - IVF, especially when combined with product quantization, can handle high-dimensional data efficiently.