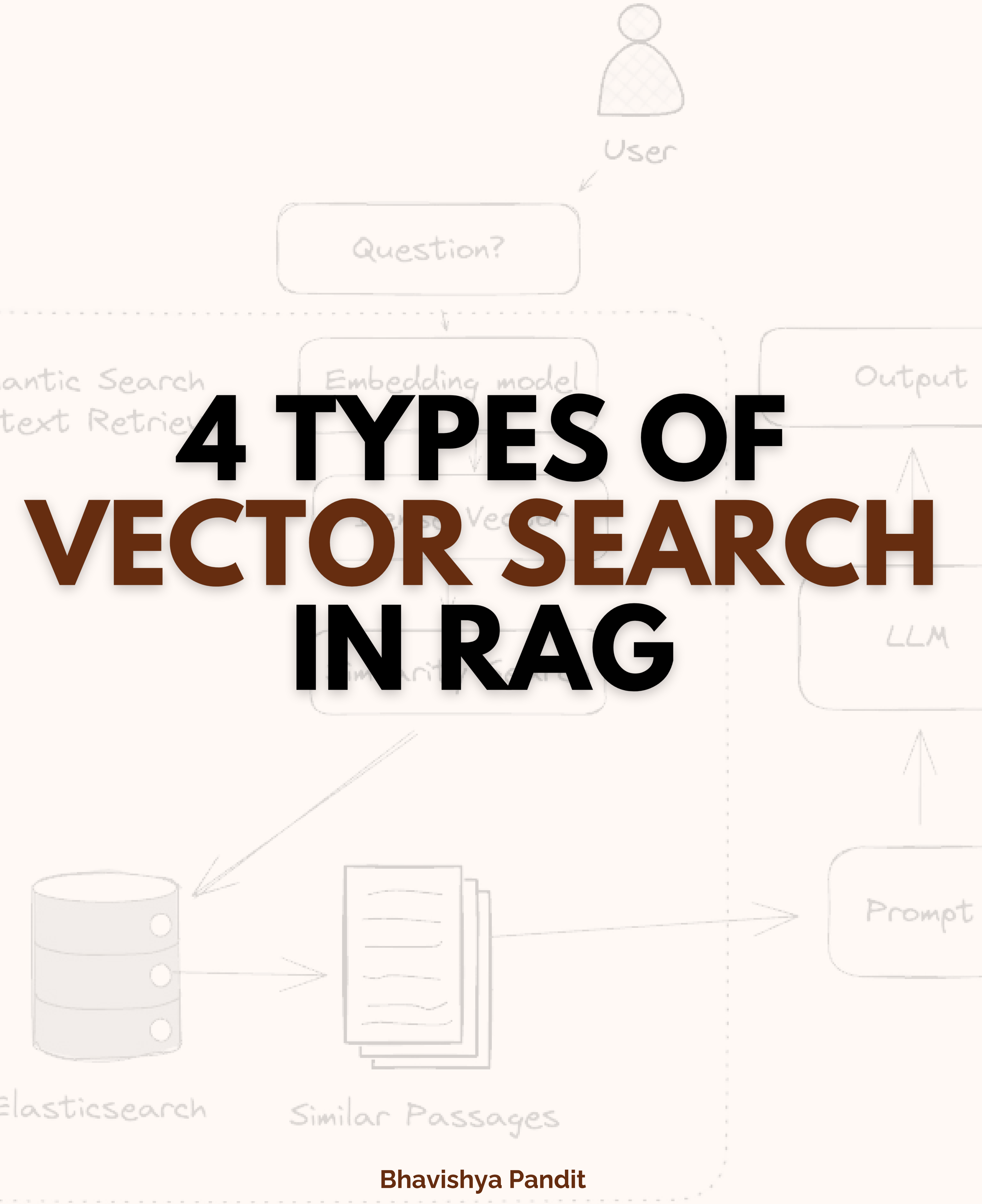
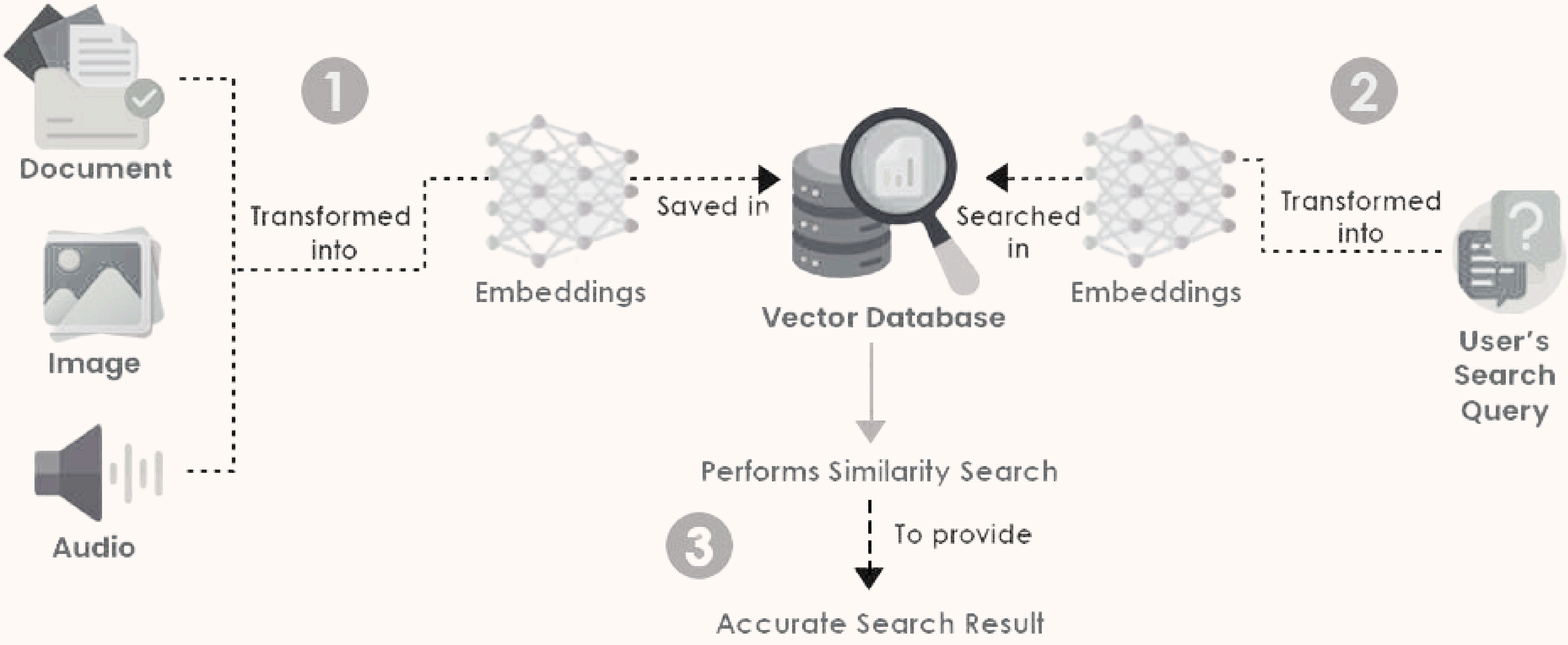
4 TYPES OF VECTOR SEARCH IN RAG



Bhavishya Pandit

WHAT IS VECTOR SEARCH?

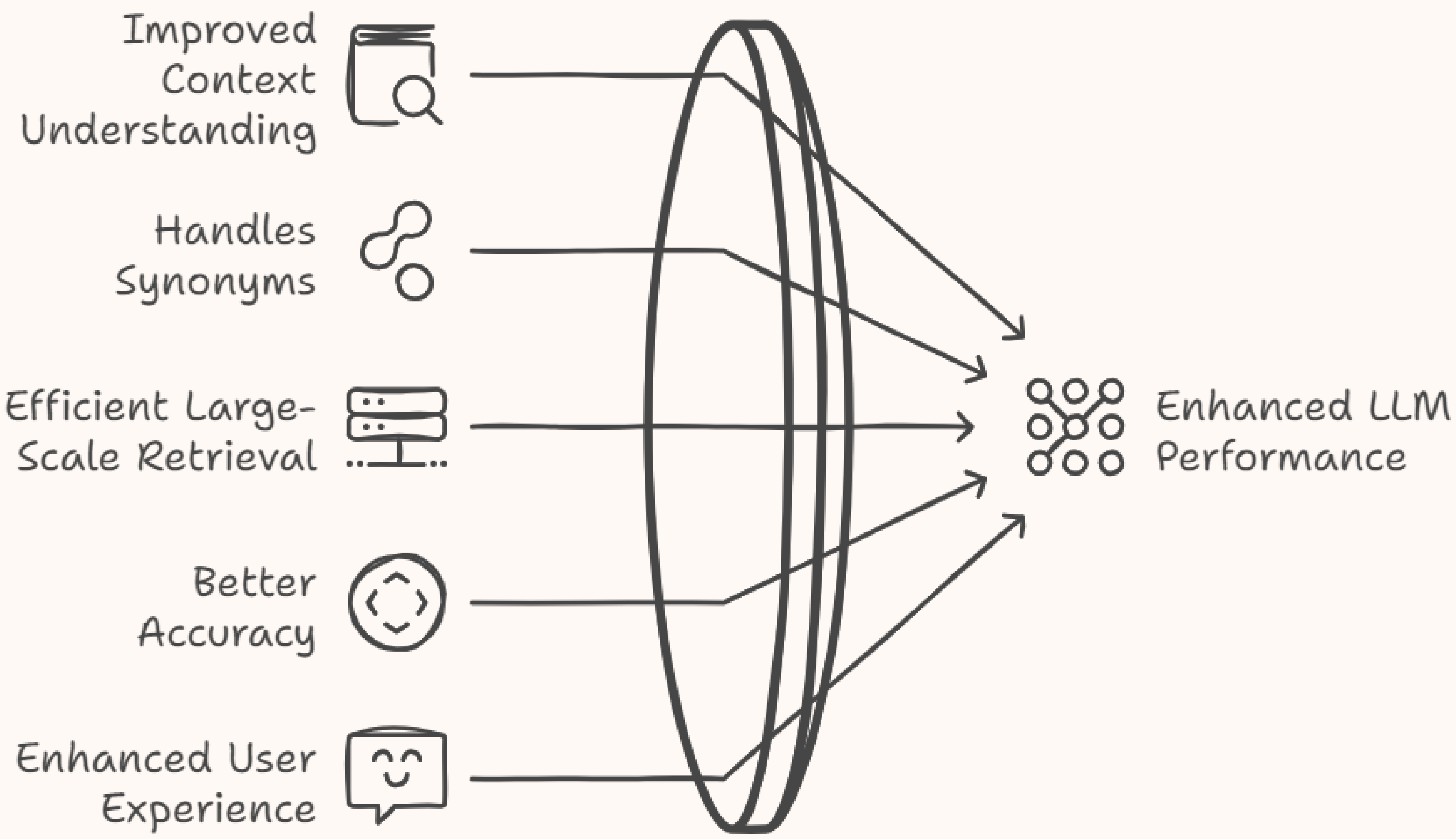


Vector search is used to find relevant information by representing text as vectors (numerical arrays) in high-dimensional space. Instead of keyword matching, it compares the semantic similarity between vectors, enabling more accurate retrieval of contextually similar content.

Vector search leverages embeddings, where words, sentences, or documents are transformed into dense vector representations. By measuring the proximity (e.g., cosine similarity) between these vectors, it identifies the most semantically relevant results, making it particularly effective for tasks like document retrieval, recommendation systems, and question-answering within RAG.

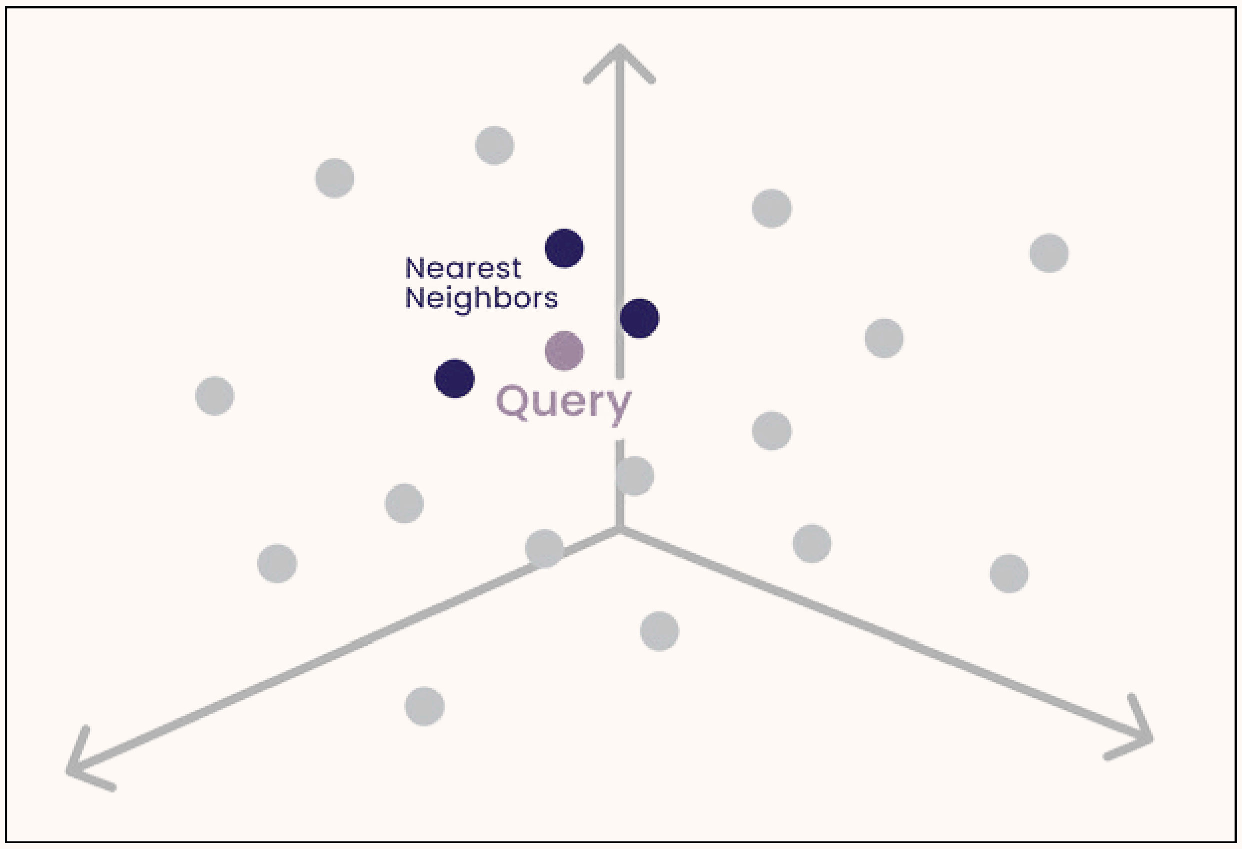
Bhavishya Pandit

WHY IS IT IMPORTANT?



- **Improved Context Understanding**: Vector search helps RAG retrieve information based on semantic meaning, not just exact keywords.
- **Handles Synonyms**: It captures similar meanings even if different words or phrases are used, improving search flexibility.
- **Efficient Large-Scale Retrieval**: It can quickly search vast amounts of data, essential for large datasets and real-time applications.
- **Better Accuracy**: By focusing on vector similarity, it provides more relevant and precise results compared to keyword matching.
- **Enhanced User Experience**: It improves the quality of answers and recommendations by understanding context and intent, leading to more meaningful interactions.

1. EXACT NEAREST NEIGHBOR SEARCH

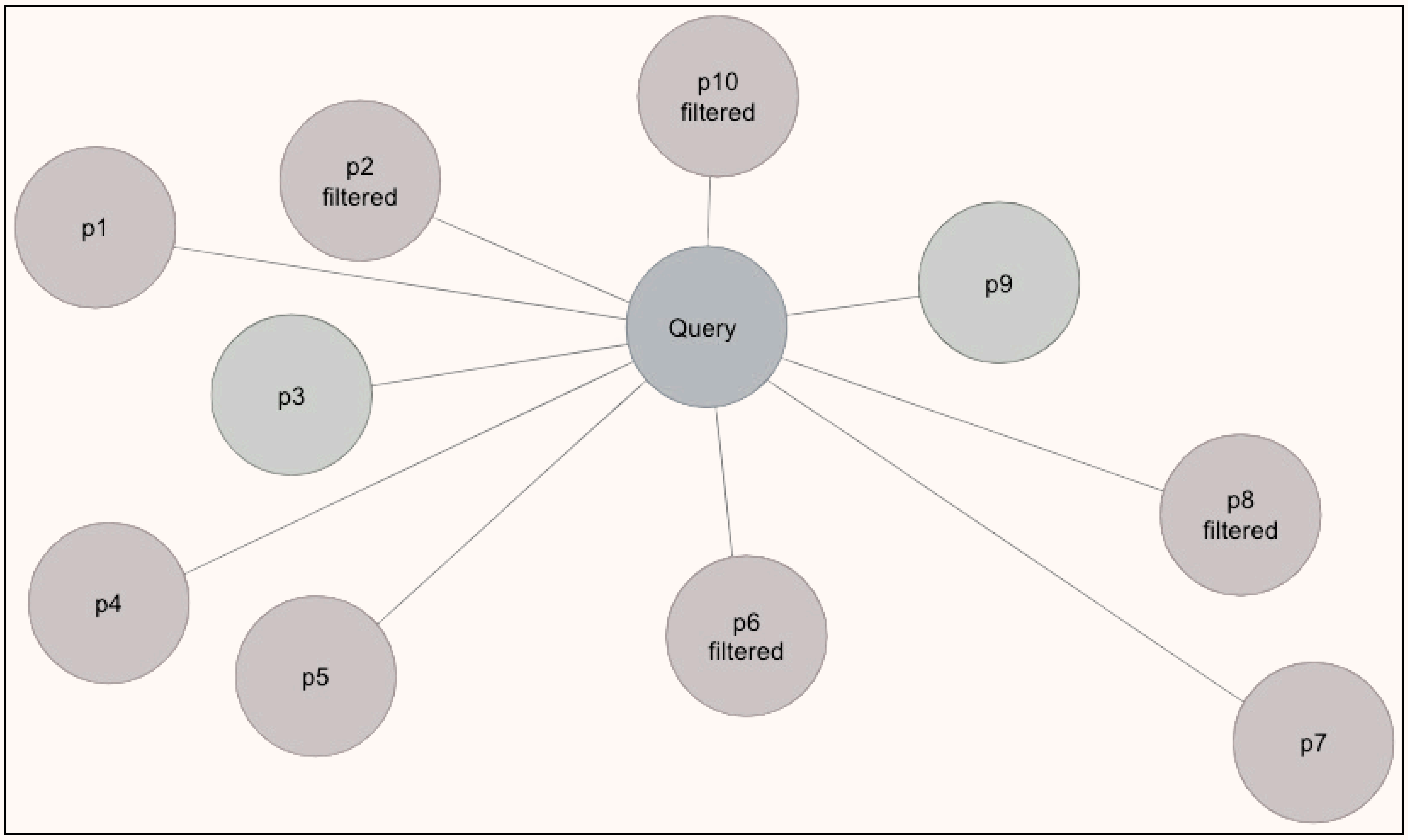


Exact Nearest Neighbor (ENN) Search is a search technique used to find the closest data points (neighbors) to a given query point in high-dimensional space. It compares the distances between vectors and retrieves the exact data points with the smallest distance to the query.

Here's how the Exact Nearest Neighbor (ENN) search algorithm works:

- **Vector Representation**: Data points and the query point are represented as vectors in a high-dimensional space.
- **Distance Calculation**: For each data point in the dataset, the algorithm computes the distance to the query point using a distance metric (e.g., Euclidean distance, Manhattan distance).
- **Find the Closest Points**: The algorithm compares the distances and identifies the data point(s) with the smallest distance to the query.
- **Return Result**: The closest data point(s) is returned as the exact nearest neighbor(s) to the query.

2. APPROXIMATE NEAREST NEIGHBOR (ANN) SEARCH



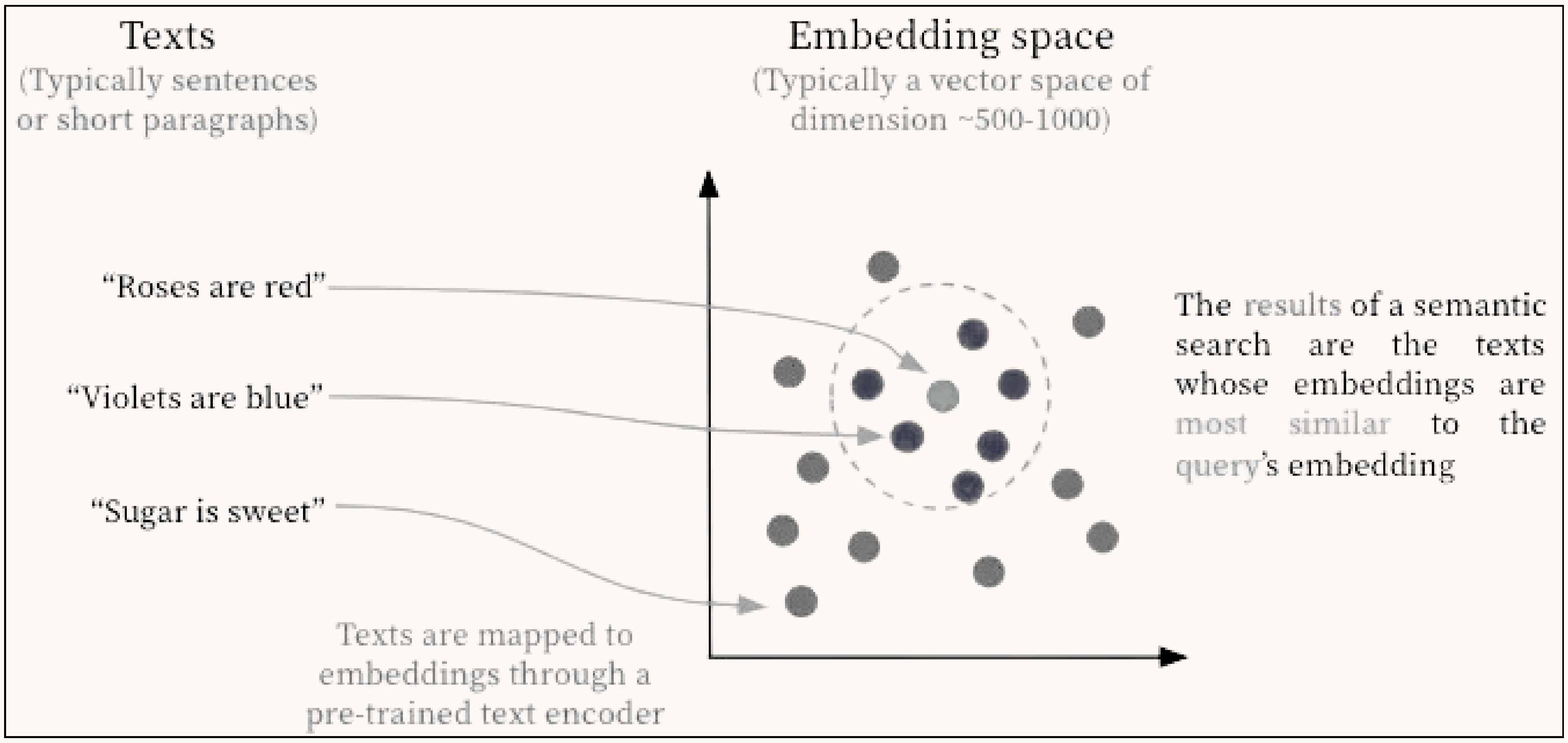
ANN Search is a search technique used to quickly find data points that are close to a query point in high-dimensional space, but with a trade-off in precision. Unlike Exact Nearest Neighbor (ENN) search, ANN aims to find a "good enough" nearest neighbor faster, which is useful for large datasets where exact search can be too slow.

How ANN Search Works :

1. **Vector Representation**: Data points and the query are transformed into vectors in high-dimensional space.
2. **Data Partitioning**: The algorithm partitions the dataset into smaller, manageable subsets using techniques like hashing, tree-based structures, or clustering.
3. **Search within Subset**: The algorithm performs a search within the nearest partition rather than the entire dataset, reducing computational effort.
4. **Return Approximate Neighbor:** Instead of finding the exact nearest neighbor, it returns an approximate neighbor that is close enough, trading off some accuracy for speed.

Bhavishya Pandit

3. SEMANTIC SEARCH

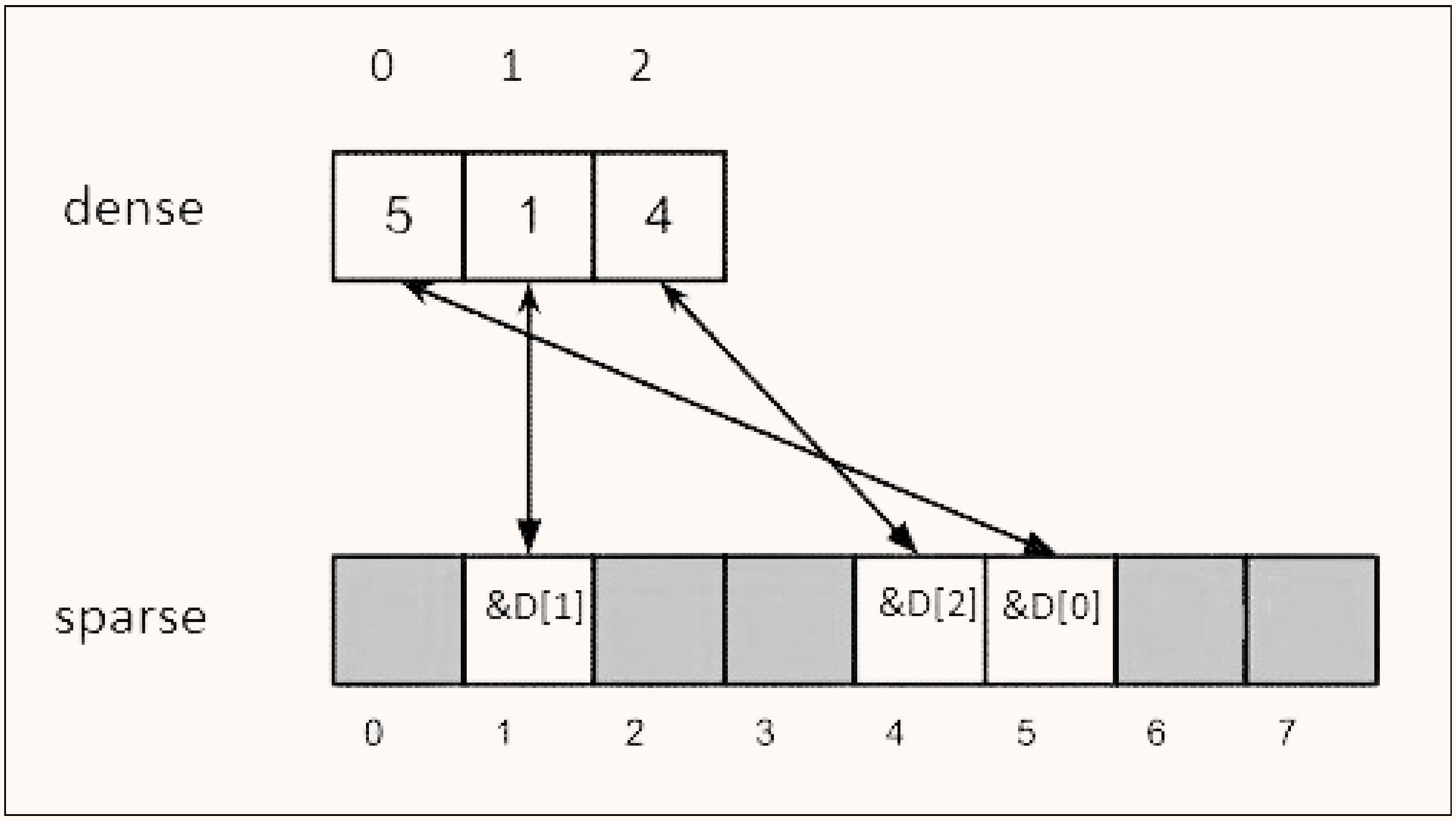

Texts
(Typically sentences or short paragraphs)
Embedding space
(Typically a vector space of dimension ~500-1000)
"Roses are red"
"Violets are blue"
"Sugar is sweet"
The results of a semantic search are the texts whose embeddings are most similar to the query's embedding
Texts are mapped to embeddings through a pre-trained text encoder

Semantic Search is a search technique that focuses on understanding the meaning and context of the query, rather than just matching keywords. It leverages natural language processing (NLP) and machine learning to retrieve results based on the intent and semantic relationships in the text.

How Semantic Search Works :

1. **Text Embeddings**: Queries and documents are converted into dense vector representations (embeddings) that capture their meaning and context.
2. **Similarity Comparison**: The search algorithm compares the vectors of the query and documents to find semantically similar content, often using metrics like cosine similarity.
3. **Contextual Understanding**: It identifies relevant information based on the underlying meaning, even if the exact keywords aren't present.
4. **Return Relevant Results**: The most semantically similar results are ranked and returned, improving accuracy and relevance in search results.

4. SPARSE VECTOR SEARCH

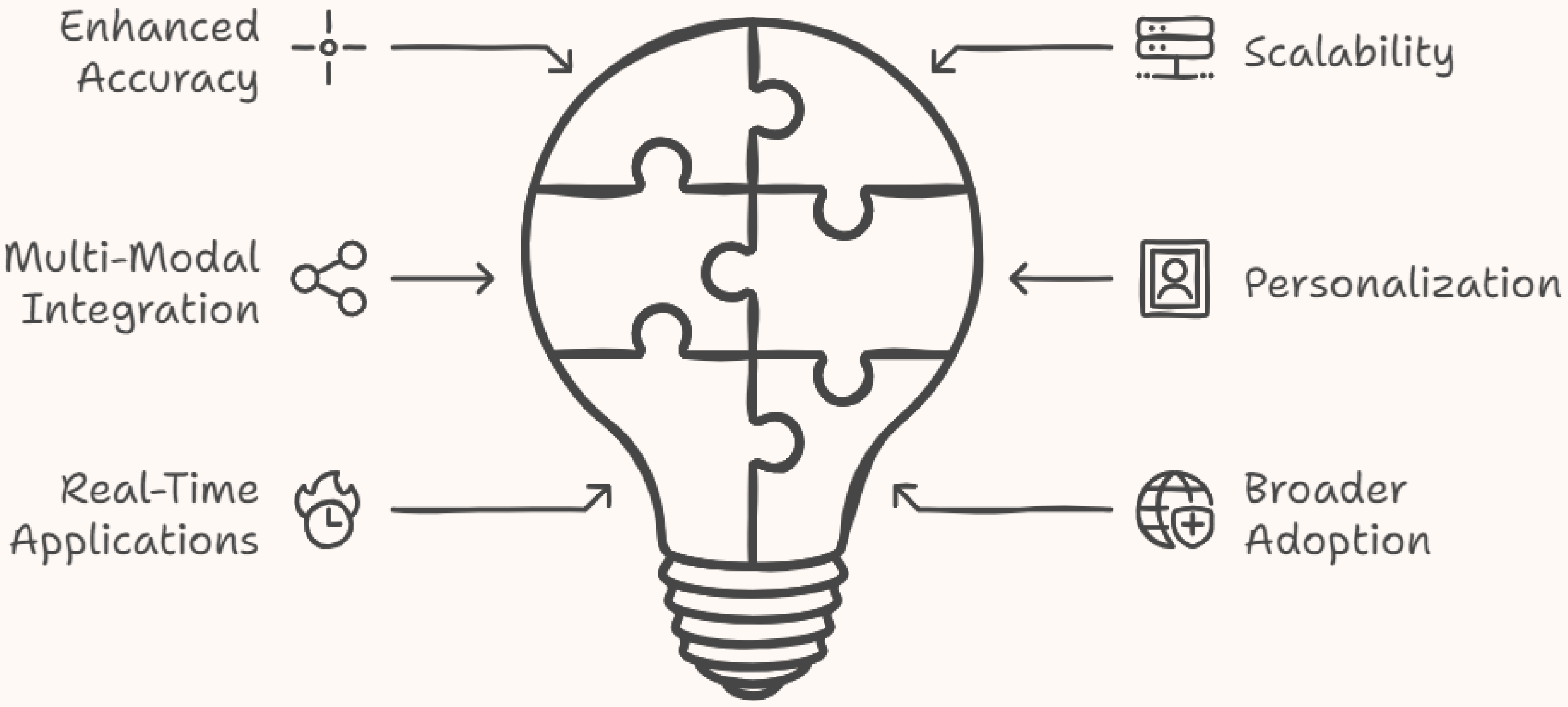


Sparse Vector Search is a search technique used to find relevant data points in high-dimensional space, focusing on vectors that contain a significant number of zeros. This is useful in scenarios where only a small subset of features is relevant, such as in text data represented by term frequency-inverse document frequency or word embeddings.

How Sparse Vector Search Works:

1. **Sparse Representation**: Data points and queries are represented as sparse vectors, where most of the elements are zeros.
2. **Distance Metrics**: The search algorithm employs specialized distance metrics (like cosine similarity or Jaccard index) focusing on the non-zero elements.
3. **Indexing**: Sparse vectors are often indexed using data structures like inverted indices or locality-sensitive hashing to speed up the search process.
4. **Nearest Neighbor Search**: The algorithm identifies the closest data points based on their sparse vector representations, efficiently filtering out irrelevant data.
5. **Return Relevant Results**: The results are ranked and returned based on their similarity to the query vector, emphasizing the most relevant data points despite the sparsity.

FUTURE OF VECTOR SEARCH



- **Enhanced Accuracy**: Improved algorithms and contextual information will lead to more precise and relevant search results.
- **Scalability**: Advanced indexing structures and distributed computing will enable efficient search through massive datasets.
- **Multi-Modal Integration**: Vector search will increasingly support cross-modal data (text, images, audio), providing richer results.
- **Personalization**: Tailored search results based on user preferences and dynamic context adaptation will enhance user experience.
- **Real-Time Applications**: Faster processing speeds will allow for real-time vector search, improving interactions in applications like conversational AI.
- **Broader Adoption**: Industries will leverage vector search for better knowledge management and decision-making.

Bhavishya Pandit

DO YOU KNOW ANY OTHER VECTOR SEARCH TECHNIQUES?

Let me know in the comments

Follow to stay updated on GenAI

SAVE

LIKE

SHARE