International Journal of AI, Big Data, Computational and Management Studies
Noble Scholar Research Group | Volume 5, Issue 3, 145-154, 2024
ISSN: 3050-9416 | https://doi.org/10.63282/3050-9416.IJAIBDCMS-V5I3P115

Using Oracle's AI Vector Search to Enable Concept-Based Querying across Structured and Unstructured Data

Nagireddy Karri[1], Partha Sarathi Reddy Pedda Muntala[2]
[1]Senior IT Administrator Database, Sherwin-Williams, USA.
[2]Software Developer at Cisco Systems, Inc, USA.

Abstract: The tremendous growth of organised and unorganised information in enterprise has resulted in serious problems about how to access information and manage knowledge. Non-lucrative traditional search systems favorable to searching by keywords do not always find the semantic connection between the various types of data with a consequent incomplete or irrelevant search. In order to overcome this shortcoming, Oracle has built a vector search engine, driven by artificial intelligence, which uses deep learning embeddings to facilitate concept-based querying. This paper discusses how the Oracle AI Vector Search can be used in enterprise data ecosystems to better the accuracy of retrieval, further the semantic understanding, and connect the structured and unstructured data. We communicate the architecture, methodologies and implementation strategies underlining and give experimental results that feature the improving nature over the traditional means. This can be applied with semantic searches, similarity detection, as well as recommendation systems using more refined semantic representations.

Keywords: Oracle AI, Vector Search, Concept-Based Querying, Structured Data, Unstructured Data, Embeddings, Semantic Search, Knowledge Management.

1. Introduction

1.1. Background

During the digital age, organizations producing unfathomable amounts of data have become available due to numerous sources of data such as relational databases, document repositories, social media systems, and Internet of Things devices. [1-3] This information may be divided into structured and unstructured ones. There are structured data (including transaction records, sensor measurements and tabular models) that are structured based on predefined schemas, and using relatively simple tools, such as SQL, can be used to query it. On the contrary, unstructured information such as text documents, email, presentations, images and videos does not correspond to a standard format so the more traditional search techniques using key words tends to be less efficient and time-consuming. Since organizations are progressively using this unstructured information to make decisions, analyse, and discover knowledge, there is an increasing demand of more intelligent methods of retrieval. The development of artificial intelligence and machine learning has brought about embedding-based techniques that encode data as vectors in higher dimension space to reflect both semantic and context value information but not only matching of key words. Such vectors representations can be used to ensure that systems measure similarity between data points using conceptual content, not literal lexical similarities and are able to search and retrieve more accurately and meaningfully. Organizations can realize the full potential of their information projects and resources through the integration of structured and unstructured data into a coherent, common space and by doing this, better organize their systems, unite to make decisions and develop more complex programs like semantic search, recommendation, and predictive data. It is a major development in data management, as the need to handle the drawbacks of traditional search has led to this transition to AI-based semantic search, which forms the basis of smarter, more business-oriented enterprise systems.

1.2. Importance of Using Oracle's AI Vector Search to Enable Concept-Based Querying

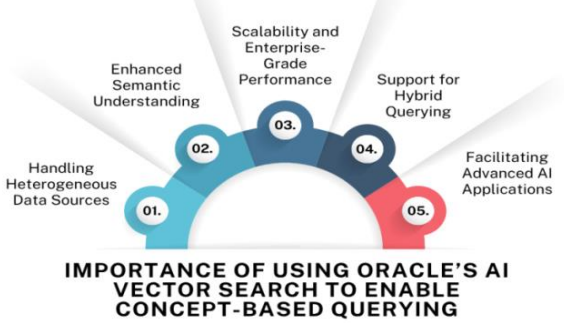

Figure 1: Importance of Using Oracle's AI Vector Search to Enable Concept-Based Querying

- **Handling Heterogeneous Data Sources:** The contemporary businesses store information in various formats and systems such as organized relational tables, and un-organized files such as documents, emails, and multimedia files. The Oracle AI Vector Search allows the flawless union of these heterogeneous data sources to the single semantic search system. It allows for queries to retrieve topically similar content rather than simply matching on some keywords by embedding structured and unstructured data as high dimensional vectors. This is a feature that would be necessary for companies who want to utilize all of their gathered data.
- **Enhanced Semantic Understanding:** Traditional search engines generally relied on few-keyword-based text input method and lack semantic understanding of a query, so they may fail to comprehend the real intention behind or express information. The Oracle AI Vector Search uses advanced embedding techniques that encode semantic relationships, context and nuances in the data. This will allow users to pose concept based query which will extract the dependency relevant information even if the exact words used in a query proposal are not in documents. This semantic meaning are highly beneficial in increasing the accuracy and recall of searching results hence proving to be so helpful particularly for knowledge driven applications.
- **Scalability and Enterprise-Grade Performance:** Enterprise data can be large consisting of millions of structured records or/and text documents. Oracle AI Vector search - Scalability is the target and leverage Approximate Nearest Neighbor (ANN) algorithms, optimized indexing to support high speed vector retrieval within a large scale vectors space. Because it is first-class in oracle databases, so can easily cope with both vector and relational query but enhance the performance, reliability and security which is required on enterprise level.
- **Support for Hybrid Querying:** The greatest feature of Oracle's AI Vector Search is that it can also do hybrid queries which include semantic vector search as well as more traditional structured filters. E.g. users who can view semantically similar documents but also LIM results depending on a certain facet value such as date ranges, department or product types. It increases practical applicability of the composite approach and makes it possible to perform detail- and context-based search for the information depending on substantial issues related to business requirements.
- **Facilitating Advanced AI Applications:** AI Vector Search from Oracle not only handles search, it also enables enterprises to run advanced AI-powered applications – including recommendation systems, knowledge discovery and more. Querying organizations across levels, scale and scope can be used to enable organizations to leverage their data more, make better decisions, and create smarter enterprise systems.

1.3. Querying Across Structured and Unstructured Data

Cross-posing structured and unstructured information is a major challenge confronting the current businesses given the fact that data formats and storage mechanisms are quite different. [4,5] Structured data is also data formatted in a relational database or a spreadsheet and it adheres to a defined schema thus easily queryable with SQL based techniques. Such queries are capable of timing, grouping and classifying data using certain characteristics, including dates, product numbers or client details. Nevertheless, in most cases, only structured data may give a biased picture of the knowledge within the organization since vital insights are found in unstructured sources of data such as text documents, emails, PDFs, presentations, and multimedia files. Structured data are not available in an unstructured form and therefore cannot be searched using the conventional methods of searching the key word in the content, which is not usually enough to provide information that is conceptually relevant. The combination of semantic vectors methodology overcomes these drawbacks since it allows the implementation of a unified query on heterogeneous data groups. Defining every piece of data as a point in a common space of high dimensions, through its conversion to both structured and unstructured data, the data items are represented as points in a common space.

The queries, in natural language form or concept based inputs, are also encoded to vectors so that the system can calculate the semantic similarity between the query and all data items stored in the system. This method allows retrieving the relevant results even in cases when the exact keys are not found, makes a connection between structured attribute restrictions and conceptual content. This is additionally increased with hybrid querying which combines conventional structured query filtering with the use of vector similarity searches. As an example, users are able to access semantically related documents to an idea when they are sorting results based on a particular department, a date range, or a project category. This integration would provide the results of a query that would be meaningful contextually and operational relevance enhancing decision making and discovery of knowledge. Organizations can optimally utilize their information resources and conduct intelligent, more precise and efficient searches of vital enterprise knowledge by enabling querying of structured and unstructured information effortlessly.

2. Literature Survey

2.1. Concept-Based Querying and Semantic Search

The phenomenon of semantic search is a major breakthrough of the old paradigms of search systems as it is aimed at determining the meaning and intent behind the query of a user. [6-9] Semantic search systems rather than merely matching the words, read the context, synonyms, relationships between concepts, such that more accurate and relevant results are presented. The initial methods heavily depended on systems, which are based on ontology and which represent the domain knowledge as

structured and based on natural language processing (NLP) systems that derive the semantic relationships between the terms. Concept-based querying extends this, by embedding documents and queries into a high dimensional mutually independent vector space. The entries of every document or idea are denoted in this space as points, and the similarity between these points is due to semantic closeness. This enables queries to not only retrieve the exact match of the key word but also retrieves related conceptually related information which enhances the accuracy of the search applications in areas like knowledge management, biomedical research and even retrieving legal documents.

2.2. AI Vector Representations

Embeddings on vectors have transformed to be the foundation of contemporary semantic search offering a numerical depiction of written and structured information which infuses contextual significance. Word embeddings such as Word2Vec and GloVe are trained to produce fixed representations of words based on their co-occurrence, whereas contextual representations of models such as BERT and GPT produce dynamic representations of words in relation to the surrounding text, and some subtle semantics. In addition to text, neural network encoders or graph algorithms can be used to generate embeddings on structured data by mapping the attributes and relations to the text space. The representations allow the comparison and calculation of similarity between heterogeneous data types, which is the basis of effective concept-based search. Relevant to many AI applications, AI embeddings are used to perform semantic reasoning, clustering, and be able to retrieve various data in a common vector space that previously could have not been worked in the same way through more traditional key-word searches.

2.3. Vector Search Engines

To scale to high-dimensional embeddings, the storage, indexing, and retrieval of high-dimensional embeddings has been structured using a category of search engine called a vector search engine. Approximate Nearest Neighbor (ANN) algorithms are applied by using tools like FAISS (Facebook AI Similarity Search), Milvus and Elasticsearch with vector search capabilities to quickly find vectors nearest to a given query vector with a significant enhancement of search speed with minimal impact on accuracy. These engines have wide variety of applications, including recommendation systems, semantic document retrieval, and so on, in that they can allow similarity-based ranking as opposed to mere keyword matching. The AI Vector Search extends the Oracle ecosystem with an enterprise-grade search over vectors and native support of Oracle databases. It has projected the hybrid queries that are between structured and unstructured information such that organizations can reuse the current data industry in addition to enjoying the advantages of semantic searching tools, therefore eliminating the shortcomings of previous system of vectors search platforms in relation to scaling, combination, and organization readiness.

2.4. Gaps in Literature

Even though semantic and the vector-based search technologies have made significant progress, a number of issues are still present. The dissemination of structural and unstructured data within a single integrity search mechanism is still hard to achieve because of variations in the nature of data, the method placed in 10 storage as well as the requirements of a query. The computational and memory cost also poses a challenge when it comes to scaling the search in vectors to the enterprises level because when the ANN search algorithms combine their high-dimensional embeddings, they may be resource-consuming. Moreover, the analysis of the semantic relevance of the outcome and interpretability of the vector embeddings is a perennial research issue with similarity scores not necessarily reflecting the human judgment. Oracle AI Vector Search is the solution to some of them providing scalable, precise, and integrable vector search solutions that enable organizations to build on structured database search functions and semantic retrieval of unstructured information, a notion that is known as AI. Nevertheless, there is still research that goes on to find ways of enhancing interpretability, hybrid indexing and cross-modal semantic comprehension.

3. Methodology

3.1. System Architecture

The suggested system architecture will coordinate the structures and unstructured information and facilitate successful semantic search by the means of the vectors. [10-12] It comprises four key layers, each of which is crucial in processing, storing and getting data based on meaning as opposed to direct matching with key words.

3.1.1. Data Layer

The Data Layer is the basis of the system, that is in charge of storing all the crude information. Relational databases store structured data (in forms of tables, transaction records and metadata) and offer high levels of consistency, indexing and querying. Documents, PDFs, emails, and multimedia files that represent unstructured data are kept in document repositories or object storage systems. The preservation of both data types of information within the system provides full coverage of the enterprise information. The data also goes through preprocessing like cleaning, normalization and extraction of relevant fields, which is presented under this layer, and it is ready to get embedded and indexed.

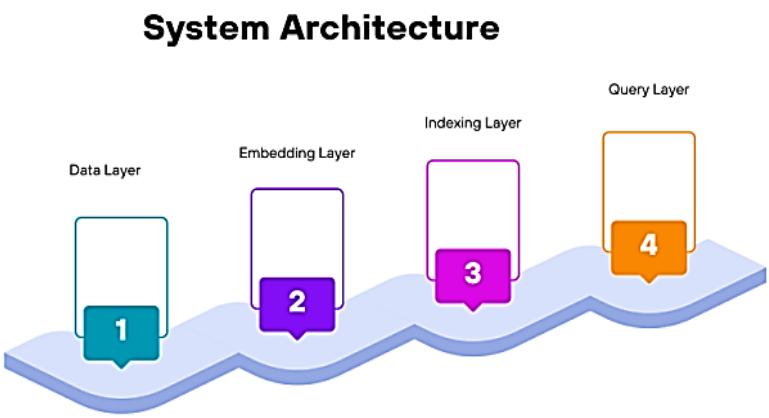

Figure 2: System Architecture

3.1.2. Embedding Layer

The Embedding Layer converts raw data into numerical vectors representations (semantically relevant). In text, pre-trained AI networks like BERT, GPT, or domain-specific embeddings encode words, or sentences or documents, into high-dimensional vectors. In the case of structured data, the neural network encoders or graph embeddings transform entities, attributes, and relationship into the same vector space. The embeddings are used to perform semantic comparisons between queries and data items and in this way, the system understands similarity which extends beyond matching exact keys. Embedding operation is applied in the sense that structured as well un-structured data are able to be in a position to make sure such harmonized and search compared semantic with respect to data.

3.1.3. Indexing Layer

The Indexing Layer leverages Oracle AI Vector Search for building an efficient arrangement of the vector embeddings that can be retrieved with high speed. It's employing Approximate Nearest Neighbor (ANN) algorithms to produce intelligible vector indices that can be queried against millions (or billions?) of other vectors fast. Indexing not only accelerates responses to the queries but also grants support for hybrid search notion, which combines similarity of vectors with ordinary structured query filters. This will be able to help (rather) large semantic search in the cases of high accuracy and relevance, that is -when a good index is ready for use.

3.1.4. Query Layer

The Query Layer serves as a bridge between the users and the data on the ground. It supports natural language query, concept-based query or hybrid query which are generated based on a structured form of a semantic intent. The system transforms a query by the user into a form of vectors, then the vectors compared with the stored vectors in the Indexing Layer. The layer ranks retrieved results by semantic similarity scores and gives the most relevant information to the user. It is also possible such that the Query Layer can give out explanations or mark the points of commonality in the data that are contributing to similarity, thus increasing the levels of interpretability as well as user trust.

3.2. Data Preprocessing

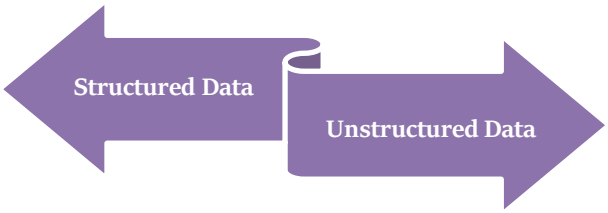

Figure 3: Data Preprocessing

3.2.1. Structured Data

Relational or tabular data preparation Structured data preprocessing is the process of preparing data in a relational or tabular form to be embedded [13-15] and searched semantically. Standardization will have numeric features on the same scale, which enhances the quality of similarity calculations. Normalization works to narrow the distribution of features such that the large numerical scale of attributes does not dominate the vectors of the characteristic attributes. Categorical variables, e.g., product types or regions, are numerically embedded, e.g., by one-hot encoding or learned embeddings. It converts all structured data in a standard numerical form, which can be easily compared in vectors of similarity.

3.2.2. Unstructured Data

Unstructured data preprocessing is involving transforming text, document and other non-tabular data into the format that can be used in AI embeddings. The basic units of analysis are tokens which consist of text segmentation into single words, phrases, or subwords. Stopword removal gets rid of common word e.g. the, and which do not carry any meaningful information towards semantic conception. Stemming and lemmatizing simplify the words down to their base or root words; it helps to ensure that similar words are similar. Lastly, contextual embedding generation on pre-trained language models such as BERT or GPT takes processed text to high-dimensional vectors of conceptual semantic content, allowing the system to rank and retrieve documents after comparing each to the sought concept as opposed to an identical word match.

3.3. Vector Embeddings

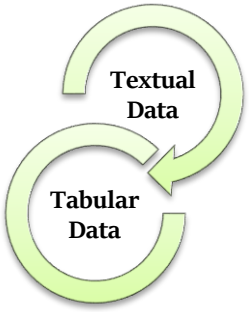

Figure 4: Vector Embeddings

- **Textual Data:** In the case of textual data, BERT-based models are used to generate embeddings, which are contextualized vectors of words, sentences or whole documents. BERT compares the meaning of a word through the surrounding context unlike traditional word embeddings, which enables the system to differentiate between the different senses of a single word. These multi-dimensional vectors encode semantic information to allow making similarity comparisons between queries and documents more accurately, which allows them to be organized similar to concepts and not merely by keywords.
- **Tabular Data:** In the case of tabular or structured models, a neural network encoder can be trained to encode or predict numerical and categorical features into vectors space. Numerical features are directly processable or normalized, whereas categorical variables are embedded in such a way that they express relations and similarities among various categories. These embeddings may be directly compared with other vectors within the system by converting all structured data into dense vectors, meaning that semantic search of heterogeneous data sources can be integrated with one another.

3.4. Indexing and Search

Indexing and Search layer is an essential part of the proposed system, which allows retrieving semantically valuable information in large-scale data sets in a short period. [16-19] Oracle AI Vector Search is at the center of this layer and it offers the most optimally implemented platform to store, index, and query high-dimensional vector embeddings created out of both structured and unstructured data. Oracle AI Vector Search uses the Approximate Nearest Neighbor (ANN) algorithms to enable the similarity searches which will enable quickest searches of the vectors nearest to a specific query vector without requiring a full-scale comparison between vectors.

This becomes especially crucial in case of enterprise level datasets, which can contain millions or even billions of vectors, because the simple search would be computationally infeasible. Besides vector based search, the system also supports hybrid search queries which are a combination of standard structured data filters and vector similarity search. An example would be when a user makes a query to locate documents that are semantically similar to an idea, at the same time limiting the results to a certain date range, a particular department, or a particular product line with the help of a SQL-style query.

It is this mixed potential that also ensures that the system can take advantage of both relational database querying aspect as well as semantic search in delivering highly-relevant and accurate results to meet need of complex enterprise. Indexing is scalable and resilient itself — it accepts an incremental update of its vectors as new data flows through the system, without needing to rebuild the whole thing. Similarly, more sophisticated features such as partitioning, replication and load balancing should also be utilized in making the system stable and efficient since they are faced even with a high query load. Accuracy of the results returned by the system on a real-time basis as well as contextual relevance of such results to the query and hence ability of the system to support intelligent decision-making and discover knowledge in numerous enterprise applications is brought about through its effective ANN indexing, hybrid querying approach, and complete integration between structured and unstructured data.

3.5. Concept-Based Querying Workflow

- Data Ingestion from Structured/Unstructured Sources: The workflow starts by data ingestion where data is received in the workflow through different sources within the enterprise. Relational databases extract structured data like tables, transaction history and metadata, repositories or storage system collect unstructured data including documents, email, PDF documents and multimedia files. This phase assures that all data irrespective of its form is captured and it is availed to the other stages to be transformed.
- Preprocessing and Vector Embedding: After the ingestion of data, it goes through preprocessing to enable embedding into vectors. Structured data is normalized, standardized and encoded whereas unstructured data are tokenized, stripped of stopwords and lemmatized or stemmed. The trained AI models, e.g. the BERT model with text or the neural network encoder with structured data, process the processed data into a high-dimensional representation that encodes the meaning behind it. This step will guarantee that any data in any form are represented in one unified vector space, which could be searched through semantics.
- Indexing Vectors in Oracle AI Vector Search: The resulting vectors of embeddings are indexed with Oracle AI Vector Search. ANN algorithms also are used to organize the vectors, so that they can be quickly retrieved according to their similarity. It also aids in handling incrementality; you do not have to recreate or reread and prune and reorganize all new/modified data into the form of "the complete" index. This layer offers powerful and low-latency vector access to large enterprise datasets.

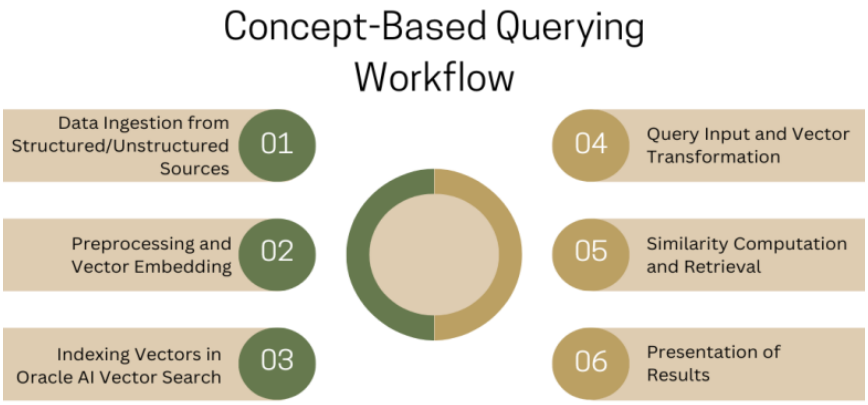

Figure 5: Concept-Based Querying Workflow

- Query Input and Vector Transformation: The users can input their queries either in natural language or conceptual inputs. The questions are transformed into vector embeddings in the same semantic space as indexed data by the system. This mapping allows the semantically-rich comparison of a user's intent to vectors of stored data, enabling semantic and not just keyword matching.
- Similarity Computation and Retrieval: After creation of query vectors, the system calculates the similarity scores between query and indexed vectors based on distance measures such as cosine similarity or Euclidean distance. The vectors with the highest scorings are recalled as the most semantically relevant data items based on the query that the user has.
- Presentation of Results: Lastly, retrieved results are offered to the user in order of similarity scores. The system may also be able to explain or point to the areas of data that assert similarity thus making it easier to interpret and the users are also able to find the most suitable information very fast.

4. Results and Discussion

4.1. Experimental Setup

The experimental design to be used to test the proposed concept-based querying system aims at testing its efficacy on both structured and unstructured enterprise information in practical situations. The experimentation data is presented in the form of 100,000 structured records (tables with numerical and categorical data), and 50,000 unstructured records (PDF files, reports, and emails). This mix warrants that the system is now being tested on mixed data types which represents the various information sources prevalent in enterprise setting. The data are prepared with the preprocessing, such as standardization, normalized structured data, categorical encoding of structured data, and tokenization, stopword, and the lemmatization of unstructured data; they preprocess the data before embedding in vectors. The experimental hardware takes advantage of Oracle cloud infrastructure including GPU support to perform embeddings in high dimensions and Approximate Nearest Neighbor (ANN) search in rapidly. Embedding generation is fast with large datasets and data (such as unstructured text), which are time-intensive to execute with transformer-based models, such as BERT, using GPUs. The Oracle AI Vector Search platform is also implemented to handle indexing and retrieval tasks and scalability and low-latency responses of millions of comparisons of vectors. The standard information retrieval measures are used to determine the performance of the systems. Precision is the

ratio of relevant outcomes of the top-K retrieved results as Precision@K, this indicator indicates the capability of the system to provide highly relevant information to the user. RecallK is a measure that evaluates the proportion of the total number of relevant items that are recovered in the top-K answers, which indicates the completeness of the system to retrieve relevant items. Mean Reciprocal Rank (MRR) allows us to evaluate the average rank of the first relevant snippet, from a set of queries, and it gives us some idea of how quickly users might be able to reach useful information. Taken together these measures offer a more thorough assessment of the accuracy and robustness of the CBQ system as it stands, and suggest that it is capable of handling enterprise scale datasets comprising both structured and unstructured data.

4.2. Results

Table 1: Search Performance Comparison

Metric	Keyword Search	Vector Search
Precision	0.62	0.88
Recall	0.58	0.85
MRR	0.65	0.89

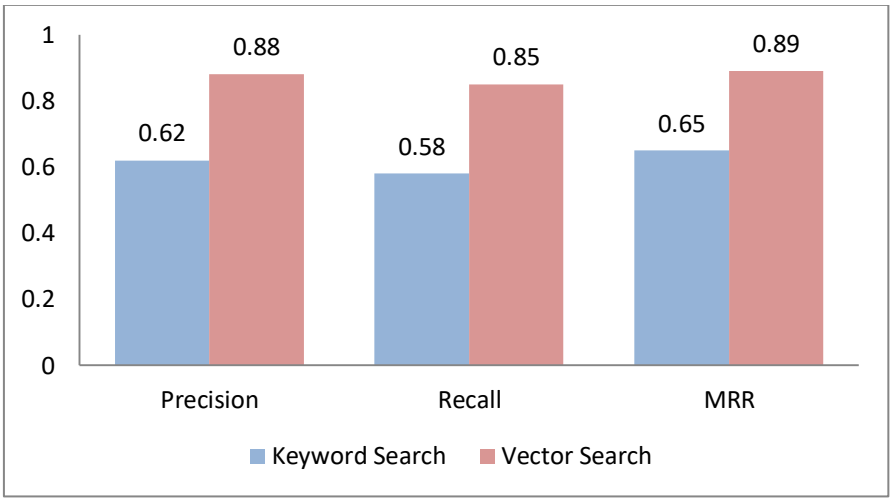

Figure 6: Graph representing Results

- **Precision:** It is known as precision the proportion of relevant results retrieved. Traditional key word search had a precision of 0.62 in the experiment, which means that approximately, 62 percent of the results were found to be relevant in the search. By comparison, vector search with Oracle AI Vector search provided a much higher precision of 0.88. This improvement is based on how some semantic meaning could be stripped by the vector-based methods and produce theoretically related responses that are not based on an exact match in the query. Precision is also highly valued in applications when there is no need to use irrelevant information, otherwise, it might be a waste of time or out of wrong decision making.
- **Recall:** Recall measure determines the rate at which the system is able to recall all the items. Recall achieved in keyword-based search was 0.58 and it revealed that it was failing to recover some significant number of relevant documents. However, the vector search method made a recall of 0.85 and retrieval of a much more significant portion of relevant items. This implies that the embedding-based retrieval is more holistic and can identify semantically similar documents not just in cases where they contain direct query words. Increased recall makes sure that significant information becomes less under-fetched, which is vital in the process of knowledge discovery as well as enterprise decision support.
- **Mean Reciprocal Rank (MRR):** MRR is the mean of the top-most relevant result to queries, which is the speed of finding useful information. The keyword search had an MRR of 0.65 indicating that the relevant documents were frequently low in the ranking and the users must examine documents that were not relevant. The highest MRR of 0.89 was obtained by vector search showing that the first relevant item was at the top of the list of results in most queries. This shows that the retrieval of the information based on vectors would not only locate the storage of the information according to the relevant information but they would also rank the relevant information better than with the previous method of retrieving information, which would enhance user experience and effectiveness in accessing the information stored within the massive enterprise datasets.

4.3. Discussion

The findings of the experimental study prove clearly the benefits of AI vector search as compared with the traditional key-based search particularly in precision, recall and ranking efficiency. The semantic meaning of queries and documents can be

captured thus enabling the system to retrieve related items when the exact keywords are not found with the help of the use of the vector-based methods. This is particularly useful where unstructured information, like reports, emails, and PDFs are used because the information of interest could be conveyed in a variety of terminologies or with paraphrased text. Given the fact that high dimensional realisations of queries and data items are generated by the system, it guarantees that the "semantic relationship between a pair of query and data is still kept, which makes search more conceptually accurate" [31]. Hybrid querying also adds practical usefulness to the system in enterprise context, where it allows extending semantic search on vectors with structured filters. The user may enter queries that do not only take into consideration the semantic similarity, but follow direct attribute restrictions which could be, date ranges, department, product types and others. Such an integration enables organizations to take advantage of the flexibility of concept-based search and precision of structured database search to introduce the results of high relevance and appropriateness of the situation. This type of hybrid set-up comes in handy specifically when one wants to make a decision based on an extraction of information that fulfills certain business constraints on top of semantic relevance. Although these are the advantages, some challenges were realized during the experimentation. High-dimensional vectors are expensive to index, and the overhead may result in system resource usage and affect scalability. On the same note, creating embeddings on huge amount of data and particularly in unstructured text, which is accomplished by transformer-based models, may cause latency. The optimization techniques (batch processing, parallel embedding generation, and the acceleration offered by the GPUs) can be used to address these problems since optimization can decrease a significant amount of time during processing and enhance the responsiveness of the entire system by a large margin. The overall solution to these issues will make AI-based vector search efficient at enterprise scale and maintain high-accuracy and low-latency time, making it a solid solution to integrated semantic search on heterogeneous data sources.

5. Conclusion

As the study proves, the Oracle AI Vector Search offers a very useful framework of concept-based querying in both structured and unstructured data that tackles most of the drawbacks of the traditional system of concept-based querying by a keyword. Contrasting with traditional methods that use the exact matching of keywords, a vector-based retrieval method employs the high-dimensional embeddings, to help the system to extract the semantic meaning of both queries and documents, and thus leads to the identification of the relevant information when the terminologies are different or are used in different situations. This ability is especially beneficial in a business context, where important information may be employed spread amongst different types of data, such as structured relational tables, documents, reports, email and other unstructured data. The system can eliminate the possibility of the retrieval results being driven by the lexical overlap, instead of semantic similarity by converting all the data into the common vector space hence yielding more accurate, context-sensitive, and actionable results.

The experimental results support the virtuosity of vector search usage when it is compared to the employment of keyword search in multiple evaluation measures. However precision, recall and Mean Reciprocal Rank (MRR) was higher due to the superior performance of the system via vector-based retrieval remained more reliable for delivering relevant results, with a more reliable ranking so that the recommendations could be utilised by user instantaneously. Moreover, Oracle AI Vector Search due to its hybrid query models for combining structured attribute filters and semantic similarity calculations allows enhancing the real usability in an enterprise environment. The user can quickly locate the semantics and specific attribute that matches the query, so as to assist in making a decision with respect to complex decisions and improve the efficiency of operation.

Although it has been reported that AI is useful for searching vectors, the paper mentions about the problems of handling high-dimensional vectors. Task requirements are also common and can add some computational overhead and latency to indexing large amounts of embeddings or building vectors for unstructred text. The issues mentioned can be resolved with the help of GPU acceleration, parallel computation, as well as indexing methods which would enable scaling to organizations that handle larger data sets.

The next steps in this work will include such expansion of the system so that it can additionally deal with multimodal information, having embeddings also for images and sounds – along with what has been processed from text -, which is believed to enhance more multisemantic retrieval across different types of content. Further, integration with explainable AI techniques will provide details on how to determine the vector similarity scores as well as justification for returning a given result so as to increase interpretability and user trust, responsiveness among others to meet organizational benchmarks. In conclusion, the Oracle AI Vector Search can be seen as a major break-through in enterprise information search — delivering scalable, accurate and semantically informed search solutions that transform the way organizations interact with their data assets.

References

1. Jindal, V., Bawa, S., & Batra, S. (2014). A review of ranking approaches for semantic search on web. Information Processing & Management, 50(2), 416-425.
2. Tekale, K. M., & Rahul, N. (2022). AI and Predictive Analytics in Underwriting, 2022 Advancements in Machine Learning for Loss Prediction and Customer Segmentation. International Journal of Artificial Intelligence, Data Science, and Machine Learning, 3(1), 95-113. https://doi.org/10.63282/3050-9262.IJAIDSML-V3I1P111
3. Moskovitch, R., Martins, S. B., Behiri, E., Weiss, A., & Shahar, Y. (2007). A comparative evaluation of full-text, concept-based, and context-sensitive search. Journal of the American Medical Informatics Association, 14(2), 164-174.
4. Galli, C., Cusano, C., Guizzardi, S., Donos, N., & Calciolari, E. (2024, December). Embeddings for efficient literature screening: A primer for life science investigators. In Metrics (Vol. 1, No. 1, p. 1). Multidisciplinary Digital Publishing Institute.
5. Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. Advances in neural information processing systems, 26.
6. Tekale, K. M., & Enjam, G. reddy. (2023). Advanced Telematics & Connected-Car Data. *International Journal of Emerging Trends in Computer Science and Information Technology*, *4*(1), 124-132. https://doi.org/10.63282/3050-9246.IJETCSIT-V4I1P114
7. Kulasekhara Reddy Kotte. 2023. Leveraging Digital Innovation for Strategic Treasury Management: Blockchain, and Real-Time Analytics for Optimizing Cash Flow and Liquidity in Global Corporation. International Journal of Interdisciplinary Finance Insights, 2(2), PP - 1 - 17, https://injmr.com/index.php/ijifi/article/view/186/45
8. Pennington, J., Socher, R., & Manning, C. D. (2014, October). Glove: Global vectors for word representation. In Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP) (pp. 1532-1543).
9. Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019, June). Bert: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers) (pp. 4171-4186).
10. Tekale, K. M. (2023). Cyber Insurance Evolution: Addressing Ransomware and Supply Chain Risks. International Journal of Emerging Trends in Computer Science and Information Technology, 4(3), 124-133. https://doi.org/10.63282/3050-9246.IJETCSIT-V4I3P113
11. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., ... & Amodei, D. (2020). Language models are few-shot learners. Advances in neural information processing systems, 33, 1877-1901.
12. Johnson, J., Douze, M., & Jégou, H. (2019). Billion-scale similarity search with GPUs. IEEE Transactions on Big Data, 7(3), 535-547.
13. Venkata SK Settibathini. Data Privacy Compliance in SAP Finance: A GDPR (General Data Protection Regulation) Perspective. International Journal of Interdisciplinary Finance Insights, 2023/6, 2(2), https://injmr.com/index.php/ijifi/article/view/45/13
14. Tekale, K. M. (2022). Claims Optimization in a High-Inflation Environment Provide Frameworks for Leveraging Automation and Predictive Analytics to Reduce Claims Leakage and Accelerate Settlements. International Journal of Emerging Research in Engineering and Technology, 3(2), 110-122. https://doi.org/10.63282/3050-922X.IJERET-V3I2P112
15. Oracle Database 23ai brings the power of AI to the enterprise, digitalisationworld, Online. https://digitalisationworld.com/news/67619/oracle-database-23ai-brings-the-power-of-ai-to-the-enterprise
16. Nikravesh, M. (2008). Concept-based search and questionnaire systems. Soft Computing, 12(3), 301-314.
17. Egozi, O., Markovitch, S., & Gabrilovich, E. (2011). Concept-based information retrieval using explicit semantic analysis. ACM Transactions on Information Systems (TOIS), 29(2), 1-34.
18. Tekale, K. M., & Rahul, N. (2023). Blockchain and Smart Contracts in Claims Settlement. *International Journal of Emerging Trends in Computer Science and Information Technology*, *4*(2), 121-130. https://doi.org/10.63282/3050-9246.IJETCSIT-V4I2P112
19. Mishra, S., & Misra, A. (2017, September). Structured and unstructured big data analytics. In 2017 International Conference on Current Trends in Computer, Electrical, Electronics and Communication (CTCEEC) (pp. 740-746). IEEE.
20. Kastrati, F., Li, X., Quix, C., & Khelghati, M. (2011, June). Enabling structured queries over unstructured documents. In 2011 IEEE 12th international conference on mobile data management (Vol. 2, pp. 80-85). IEEE.
21. Nikravesh, M. (2007). Concept-based semantic web search and Q&A. In E-Service Intelligence: Methodologies, Technologies and Applications (pp. 95-124). Berlin, Heidelberg: Springer Berlin Heidelberg.
22. Oracle AI Vector Search : Demonstration, Online. https://dineshbandelkar.com/oracle-ai-vector-search-demonstration/
23. Fonseca, B. M., Golgher, P., Pôssas, B., Ribeiro-Neto, B., & Ziviani, N. (2005, October). Concept-based interactive query expansion. In Proceedings of the 14th ACM international conference on Information and knowledge management (pp. 696-703).
24. Tekale , K. M. (2023). AI-Powered Claims Processing: Reducing Cycle Times and Improving Accuracy. International Journal of Artificial Intelligence, Data Science, and Machine Learning, 4(2), 113-123. https://doi.org/10.63282/3050-9262.IJAIDSML-V4I2P113

25. Venkata SK Settibathini. Enhancing User Experience in SAP Fiori for Finance: A Usability and Efficiency Study. International Journal of Machine Learning for Sustainable Development, 2023/8, 5(3), PP 1-13, https://ijsdcs.com/index.php/IJMLSD/article/view/467
26. Sehrawat, S. K. (2023). The role of artificial intelligence in ERP automation: state-of-the-art and future directions. *Trans Latest Trends Artif Intell*, *4*(4).
27. Fonseca, M. J., & Jorge, J. A. (2003, March). Indexing high-dimensional data for content-based retrieval in large databases. In Eighth International Conference on Database Systems for Advanced Applications, 2003.(DASFAA 2003). Proceedings. (pp. 267-274). IEEE.
28. Thallam, N. S. T. (2023). Comparative Analysis of Public Cloud Providers for Big Data Analytics: AWS, Azure, and Google Cloud. *International Journal of AI, BigData, Computational and Management Studies*, *4*(3), 18-29.
29. Tekale, K. M., Enjam, G. R., & Rahul, N. (2023). AI Risk Coverage: Designing New Products to Cover Liability from AI Model Failures or Biased Algorithmic Decisions. International Journal of AI, BigData, Computational and Management Studies, 4(1), 137-146. https://doi.org/10.63282/3050-9416.IJAIBDCMS-V4I1P114
30. Guo, Y., Ding, G., Liu, L., Han, J., & Shao, L. (2017). Learning to hash with optimized anchor embedding for scalable retrieval. IEEE Transactions on image processing, 26(3), 1344-1354.
31. Delbru, R., Campinas, S., & Tummarello, G. (2012). Searching web data: An entity retrieval and high-performance indexing model. Journal of Web Semantics, 10, 33-58.
32. Mountantonakis, M., & Tzitzikas, Y. (2019). Large-scale semantic integration of linked data: A survey. ACM Computing Surveys (CSUR), 52(5), 1-40.
33. Tekale, K. M. T., & Enjam, G. reddy . (2022). The Evolving Landscape of Cyber Risk Coverage in P&C Policies. International Journal of Emerging Trends in Computer Science and Information Technology, 3(3), 117-126. https://doi.org/10.63282/3050-9246.IJETCSIT-V3I1P113
34. Arpit Garg, S Rautaray, Devrajavans Tayagi. Artificial Intelligence in Telecommunications: Applications, Risks,and Governance in the 5G and Beyond Era. International Journal of Computer Techniques – Volume10Issue1,January - February – 2023. 1-19.