



Building an "Ask My Document" Feature in Oracle APEX: AI-Powered RAG in Oracle 23ai

Karkuvelraja Thangamariappan



Karkuvelraja
Thangamariappan
ACE Pro

Karkuvelraja Thangamariappan

Title : **ORACLE ACE PRO**

Company : **VIBHATHI LABS**

Date : **26-03-2025**



400+ technical experts helping peers globally

The **Oracle ACE Program** recognizes and rewards community members for their technical and community contributions to the Oracle community

3 membership tiers



For more details on Oracle ACE Program:
ace.oracle.com



Nominate
yourself or someone you know:

ace.oracle.com/nominate

Connect: aceprogram_ww@oracle.com

Facebook.com/OracleACEs

[@oracleace](https://twitter.com/oracleace)





- You have a **100-page document** and need insights on a specific topic.
- How do you find the information quickly?

Think About This...

What challenges do you face when searching documents?

The Problem Statement

Challenges in Traditional Search

- Exact word matching, not meaning
- Misses' relevant information
- Struggles with large documents

What If There's a Better Way?



What if you could just ask your document a question, and it gives you the answer just like a human expert?

Outcome of This Presentation

By the end of this session, you will learn:

- ✓ How AI can process documents intelligently
- ✓ How Oracle's inbuilt AI functions make this possible
- ✓ How you can implement this in your own projects

Oracle DB 23ai and Its Capabilities

Key Features for AI and Modern Applications

- *AI Vector Search*: Store and retrieve vector embeddings for semantic search
- *DBMS_VECTOR_CHAIN Package*: Built-in functions for AI-driven text processing
- *Document Understanding*: Extract, chunk, and process text from documents
- *AI-Optimized Query Execution*: Enhanced performance for AI workloads

Vector Search: Beyond Keywords

Unlocking Semantic Understanding in Data

- Converts data into high-dimensional vectors
- Enables semantic search for more relevant results
- Uses similarity measures like cosine distance
- Enhances AI-driven search and recommendations

RAG (Retrieval Augmented Generation)

Combining Retrieval and Generation for Accuracy

- Combines information retrieval with generative AI
- Fetches relevant data before generating responses
- Improves accuracy and context awareness
- Used in AI-driven chatbots, search, and documentation assistants

How It Works

Steps to Extract the Text and convert to vector

Upload Document – Oracle extracts text

Partition into Chunks – Break into meaningful sections

Convert Text to Vectors – Uses ONNX model

Data Setup Complete – Ready for intelligent search

Note: You must load an ONNX model into your database to enable text-to-vector conversion.

The Next Steps

Fetching Data & AI Response

User Input Query – Search for specific content

Convert Input to Vector – For similarity matching

Semantic Search – Finds nearest matches

Retrieve & Sort Records – Fetch relevant document chunks

Send Data to AI – Pass search results + user query

AI Understands & Responds – Generates a meaningful answer

Steps of Implementation

Extract Text from PDF and Convert it to Chunks and then to vector

Imagine you have a table with uploaded documents. This code extracts the text, partitions it into chunks, converts it into vectors, and stores it in the VECTOR_STORE table.

Inbuilt functions used in this step:

- `dbms_vector_chain.utl_to_text(<FILE_CONTENT>)`
 - Converts the PDF file to the text
- `dbms_vector_chain.utl_to_chunks(<CONVERTED_TEXT>, json({'by':'words','max':'300','split':'sentence','normalize':'all'})`
 - Splits a large plain text document into smaller chunks of text.
- `dbms_vector_chain.utl_to_embeddings(<CHUNKED_TEXT>, json({'provider':'database','model':'<ONNX_MODEL_NAME>'})`
 - Converts the chunked text to the vectors

Semantic Search and AI Response

Convert the user question to vector and find similar vectors

User question is converted to a vector and compared with stored vectors to find the most relevant content

Inbuilt functions used in this step:

- `VECTOR_EMBEDDING(<ONNX_MODEL_NAME> USING '<USER_QUESTION>' AS data)`
 - Converts the user question into vectors
- `VECTOR_DISTANCE(<USER_QUESTION_VECTOR>, <CONTEXT_VECTOR>, COSINE)`
 - Compares the user question with the document data and provides the distance

Semantic Search and AI Response

Generate AI-Powered Response

The retrieved text chunks and the user's question are sent to an AI model. The AI processes the context and generates a coherent response based on the most relevant information extracted from the stored document data.

Note: Gemini is Used in the provided example.

Supported Models are:



Semantic Search and AI Response

Inbuilt function used to get response from AI

```
DBMS_VECTOR_CHAIN.UTL_TO_GENERATE_TEXT (  
    DATA          IN CLOB,  
    PARAMS        IN JSON default NULL  
) return CLOB;
```

This chainable utility function to generate a text response for a given prompt or an image, by accessing third-party text generation models.

Params Json Example

```
{  
    "provider"      : "ocigenai",  
    "credential_name" : "<CREDENTIAL_NAME>",  
    "url"           : "https://inference.generativeai.us-example.com/chat",  
    "model"         : "cohere.command-r-16k",  
    "chatRequest"   : { "maxTokens" : 256 }  
}
```

Application Demo

Experience the Process in Action

The screenshot displays the AskMyDoc web application interface. At the top, there is a navigation menu with a hamburger icon and the text 'AskMyDoc'. Below this, the main header area features the 'AskMyDoc' logo and the tagline 'Upload your documents and let AI answer your questions with clarity and precision, directly from the content.' To the right of the header is an 'Upload Document' button with an upward arrow icon.

The central part of the interface shows a table listing uploaded files. Each row contains the file name, file size, file type, and an 'Ask AI' button with a speech bubble icon.

File Name	File Size	File Type	
~temp_1734669170479-compressed.pdf	1074922 KB	application/pdf	Ask AI
SQL Developer.configuration.pdf	379577 KB	application/pdf	Ask AI
wordpress-pdf-invoice-plugin-sample.pdf	43627 KB	application/pdf	Ask AI
KOOS-JR.pdf	140950 KB	application/pdf	Ask AI
Subbaraju Omkaram.pdf	50443 KB	application/pdf	Ask AI

At the bottom of the screenshot, the Windows taskbar is visible, showing the Start button, search bar, and various application icons. The system tray on the right indicates the language is set to 'ENG INL', the date is '17-01-2024', and the time is '22:27'.

Key Takeaways

What we learnt in this session

- ✓ Extracting text from a PDF
- ✓ Splitting text into chunks
- ✓ Converting chunks into vectors
- ✓ Finding relevant data using vector search
- ✓ Sending data to AI for a meaningful response

References and Credits

Resources and Acknowledgements

[AI Vector Search - Complete RAG Application using PL/SQL in Oracle Database 23ai](#)

[DBMS_VECTOR_CHAIN](#)

[DMBS_VECTOR](#)

Learn more & connect with the Oracle ACE Program



Program Details

ace.oracle.com

Nomination

ace.oracle.com/nominate

ACEs in Action Blog

blogs.oracle.com/ace

Twitter

@oracleace

Facebook

The Oracle ACE Program

Linkedin

bit.ly/OracleACEs

Email

ACEprogram_ww@oracle.com

Thank you

Thank You

Q & A
Love to Connect

