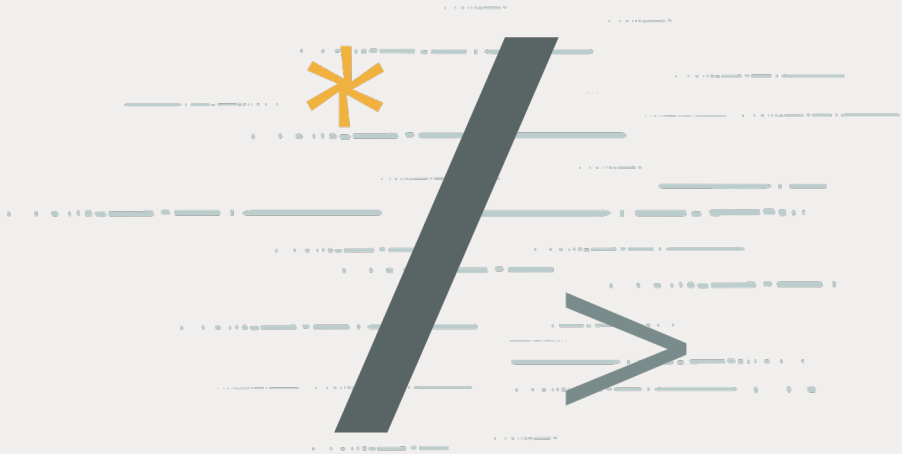
Develop AI/ML-Driven Apps with Autonomous Database 23ai

Paul Parkinson
Architect and Developer Advocate
Oracle Database
August 6th 2024

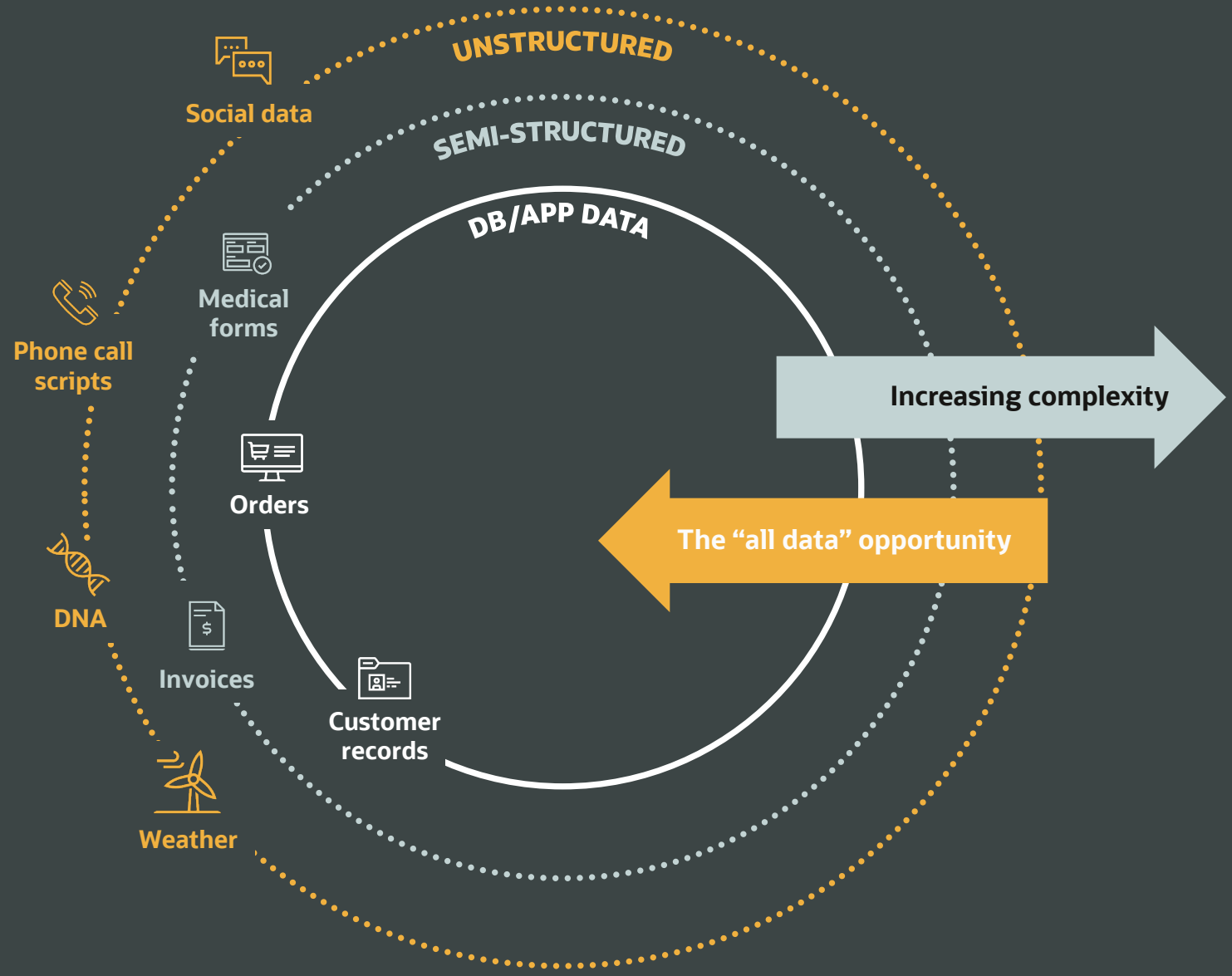
Safe harbor statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, timing, and pricing of any features or functionality described for Oracle's products may change and remains at the sole discretion of Oracle Corporation.



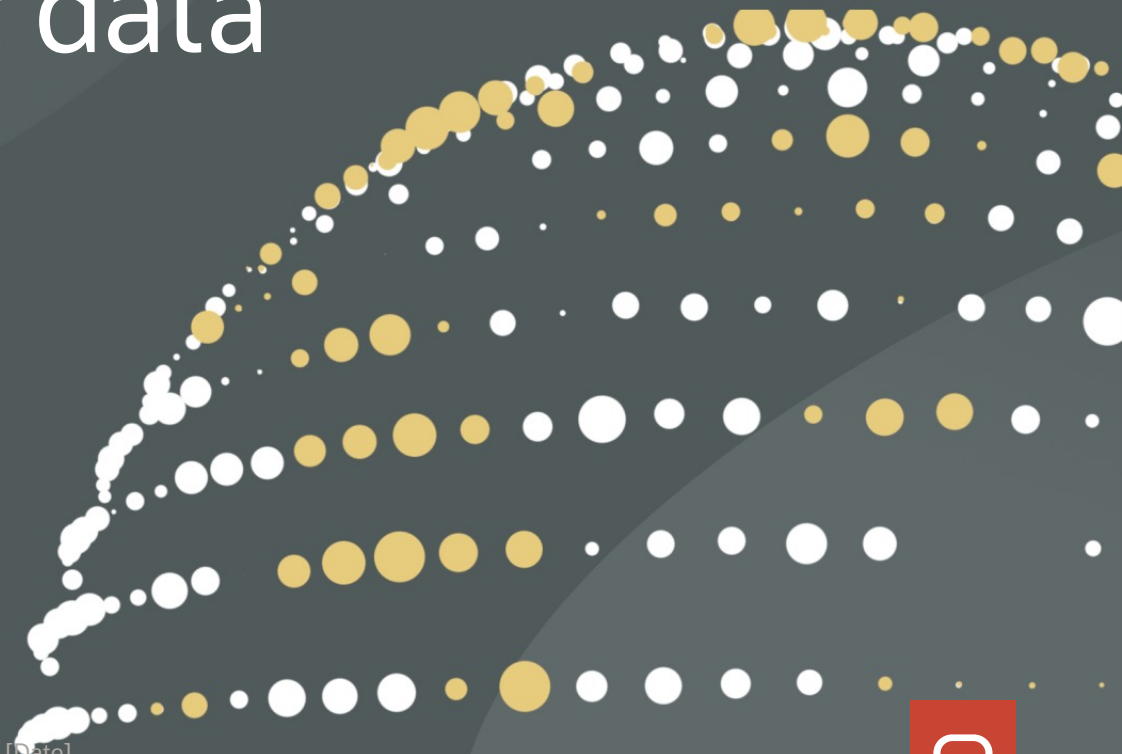
Copyright © 2024, Oracle and/or its affiliates

It all starts with data



Copyright © 2024, Oracle and/or its affiliates

Oracle brings AI to your data



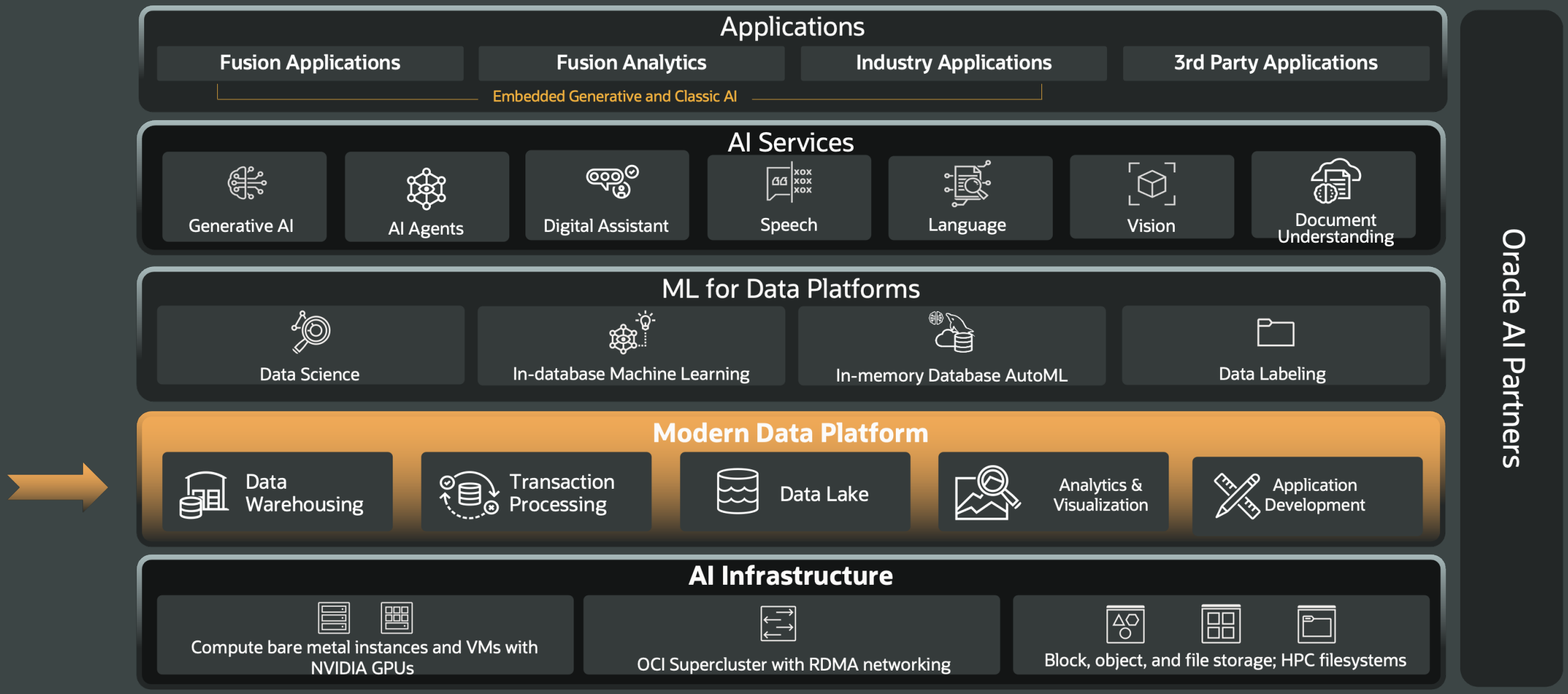
Copyright © 2024, Oracle and/or its affiliates

Agenda

- OCI AI Services Oracle AI Stack
- Oracle Multi-purpose/Converged Database
- Select AI (NL2SQL)
- Vector and RAG
- AI Agents
- AI Playground
- OML4PY
- Spatial AI
- Graph ML and RAG

Copyright © 2024, Oracle and/or its affiliates

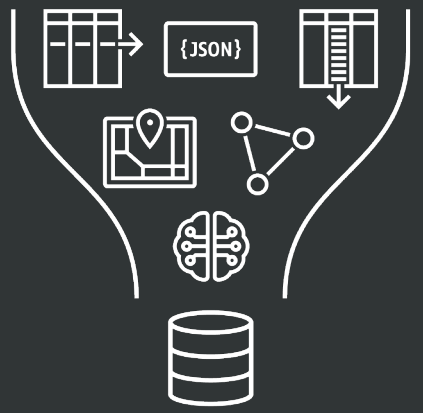
The Oracle AI stack



Copyright © 2024, Oracle and/or its affiliates

How we deliver the Vision

Complete and Simple Platform for All Data Management Needs



Converged Database

Complete: all modern data types, workloads, and development styles

Simple: Add a SQL Statement, not a database to support any need of modern applications

Running on

Autonomous Database

Powerful: All the benefits of converged database running on Exadata

Simple: Fully-managed cloud service

Language and API support

OCI…

- SDKs for JavaScript, Java, Python, .NET, PL/SQL, as well as REST.

Oracle Database…

- Support for all languages as well as REST.
- Run Java, JavaScript, and Python within the database itself.
- DBMS_CLOUD and DBMS_CLOUD_AI PL/SQL packages to call other OCI AI, etc. services
- Outbound REST calls to external services including multi-cloud, huggingface, etc.

Most AI apps will involve adding AI to existing apps vs writing from scratch.

Copyright © 2024, Oracle and/or its affiliates

What if you could…

Converse in natural language with your database

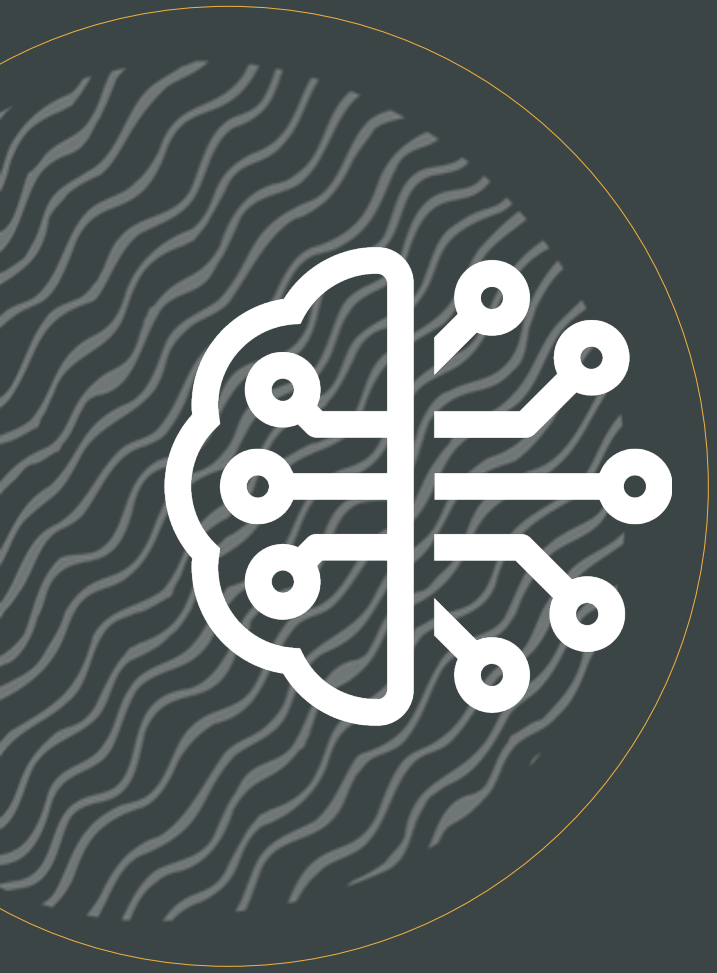
Combine semantic search on unstructured data with searches on business data

Get better answers from Generative AI by augmenting it with data in your database

Copyright © 2024, Oracle and/or its affiliates

Autonomous Database Select AI

Simplest way to get answers about your business



Select AI allows you to use your natural language to query data

- No need to understand where and how your data is stored to gain insights

Copyright © 2024, Oracle and/or its affiliates

LLMs infer the intent of our natural language question

We infer a lot from human language

what are our total streams for each tom hanks movie this month?

- total streams → total number of movie views
- for each → breakout views by movie
- tom hanks → tom hanks is an actor
- this month → understanding of time

Copyright © 2024, Oracle and/or its affiliates

Putting it all together

what are our total streams for each tom hanks movie this month?

what are our total streams for each tom hanks movie this month?

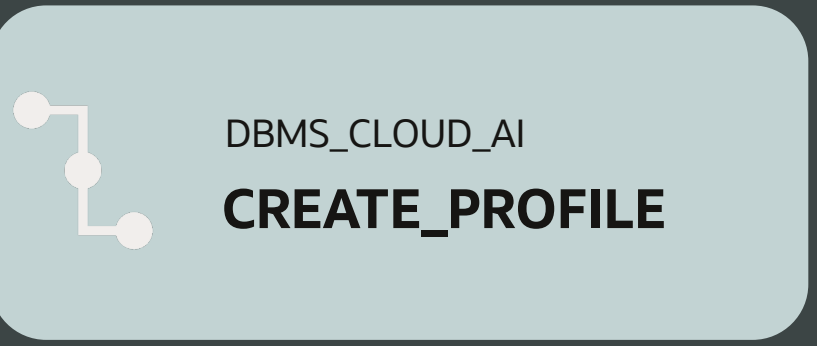
MOVIE_TITLE	TOTAL_STREAMS
Forrest Gump	888.00
Sleepless in Seattle	176.00
Splash	117.00
Angels & Demons	293.00
Saving Private Ryan	494.00
Who Killed the Electric Car?	9.00
Philadelphia	189.00
Cast Away	567.00
Big	189.00
The Great Buck Howard	4.00

```
SELECT
  m.title AS movie_title,
  COUNT(s.views) AS total_streams
FROM movie m
  JOIN sales_sample s ON m.movie_id = s.movie_id
  JOIN actors a ON m.movie_id = a.movie_id
WHERE a.actor = 'Tom Hanks'
  AND EXTRACT(MONTH FROM s.day_id) =
      EXTRACT(MONTH FROM SYSDATE)
GROUP BY m.title
```

LLMs are remarkable at inferring intent (and getting better)
But they are not perfect
It is **very important** to verify the results

How Select AI Works?

Create a profile describing your LLM & the metadata such as schemas, tables, etc.

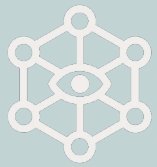



```
BEGIN DBMS_CLOUD_AI.CREATE_PROFILE(
      profile_name => 'openai_gpt35',
     attributes => '{
     "provider"       : "openai",
     "credential_name": "OPENAI_CRED",
     "object_list"    : [
     {"owner": "MOVIESTREAM", "name": "movies"},
     {"owner": "MOVIESTREAM", "name": "streams"},
     {"owner": "MOVIESTREAM", "name": "cust_ext"},
     {"owner": "MOVIESTREAM", "name": "pizza_shop"},
     {"owner": "MOVIESTREAM", "name": "actors"},
     {"owner": "MOVIESTREAM", "name": "genre"},
     {"owner": "MOVIESTREAM", "name": "cust_seg"},
     {"owner": "MOVIESTREAM", "name": "cust_contact"}
                        ] }' );
END;
/
```


Copyright © 2024, Oracle and/or its affiliates

Configure Select AI

Perform AI translation with your existing profile



DBMS_CLOUD_AI

Generate

```
l_response := DBMS_CLOUD_AI.GENERATE(
   prompt       => '<task + data>',
   profile_name => 'openai_gpt35',
   action => 'chat' / 'showsql' / 'narrate'
);



SELECT DBMS_CLOUD_AI.GENERATE(
   prompt       => 'How many customers do we have',
   profile_name => 'openai_gpt35',
   action       => 'showsql'
) FROM dual;
```


Copyright © 2024, Oracle and/or its affiliates

Adding context to users question

The Prompt Context is compiled and stored into a Page Item - P6_*PROMPT*_CONTEXT

```
:P6_FINAL_PROMPT := apex_string.join_clobs(
     apex_t_clob(
        '<CONTEXT>', :P6_PROMPT_CONTEXT||chr(10),
        '<QUESTION> Based on the context above answer the following question: '||:P6_PROMPT||CHR(10),
        'If this question cannot be answered based on above context say - "Information not found!"') );


--Example query for generating context
SELECT 'Overview of the school : '|| OVERVIEW_PARAGRAPH ||chr(10) || chr(13)||
       'The following Language Courses are taught here : '||LANGUAGE_CLASSES||chr(10) || chr(13)||
       'The following Advanced Placement Courses are taught : '||ADVANCEDP_COURSES||chr(10) || chr(13)||
       'The following is the Diversity in Admission Policy for school: '||diadetails||chr(10) || chr(13)||
       'Extra curricular activities are available: '|| extracurricular_activities|| chr(10) ||chr(13)||
       'The Public Schools Athletic League (PSAL) sports for boys: '||PSAL_SPTS_BOYS || chr(10) ||chr(13)||
       'The Public Schools Athletic League (PSAL) sports for girls: '||PSAL_STS_GIRLS || chr(10) ||chr(13)||
       'Other facilities in this school: '||addtl_info1 || chr(10) ||chr(13)||
       'The following academic oppurtunities are available : '||academic_opps || chr(10)||chr(13)
       as prompt_context
FROM high_schools WHERE id = :P6_ID;
```


Copyright © 2024, Oracle and/or its affiliates

Finding what you're looking for can be challenging…

I have a photo of an artwork, but don't know the artist or work's name

I got this error message ORA-XXXX, but don't know how to resolve it

I heard a catchy tune and can even hum a few bars, but don't know the title or artist

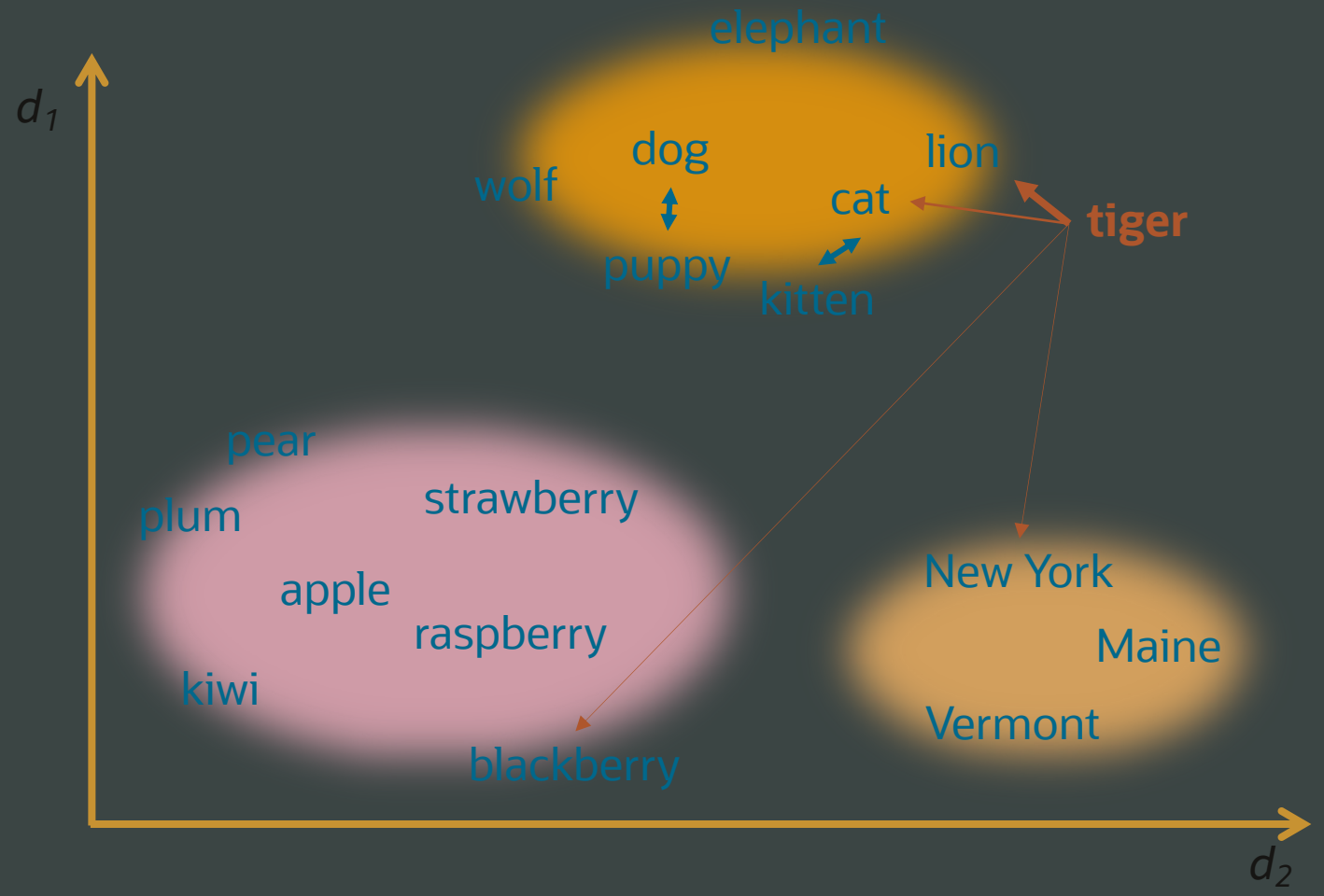
Copyright © 2023, Oracle and/or its affiliates

A technology called AI Vector Search enables end-user questions to be mapped to relevant data in your database enabling semantic searches on unstructured data.

 Copyright © 2023, Oracle and/or its affiliates

An intuitive example

Word relatedness in two dimensions



Copyright © 2023, Oracle and/or its affiliates

Embeddings for images

Convert an image to one or more vector embeddings and search vector database

0.5, 1.5, 2.6, -1.1, …

Compare these against other vectors in the database to find similar content

Copyright © 2023, Oracle and/or its affiliates

Introducing Oracle AI Vector Search

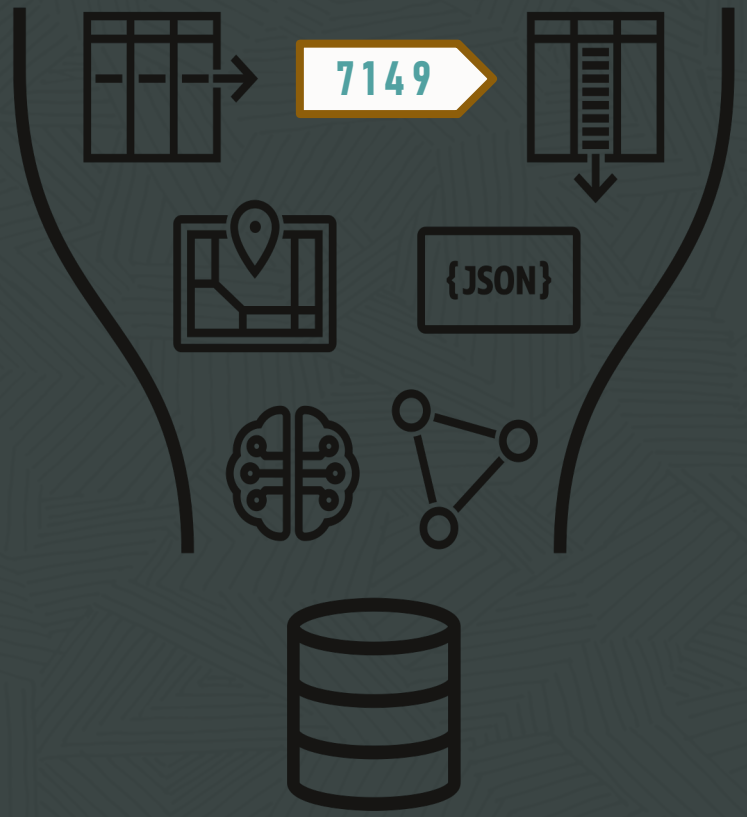
New set of capabilities coming in Oracle Database 23ai

Process vector search and other workloads in same Oracle converged database

Designed to be simple to use and easy to understand

- **New** VECTOR data type for storing vectors
- **New** SQL syntax and functions express similarity search with ease
- **New** Approximate search indexes packaged and tuned for high performance and quality

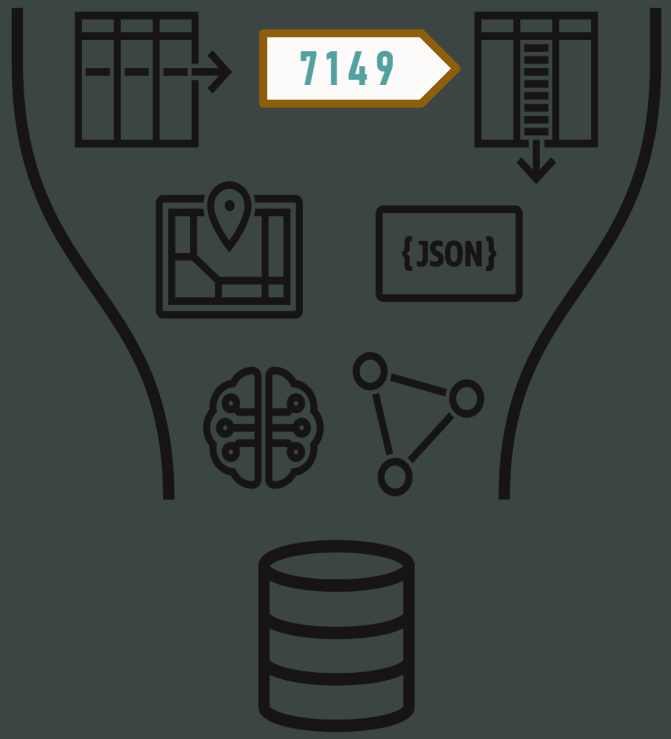
Perform vector search in queries alongside business data about customers and products



Converged Database

Copyright © 2024, Oracle and/or its affiliates

Oracle Database 23ai can store vectors using a new vector data type



```
CREATE TABLE house_for_sale (house_id     number,
                             price        number,
                             city         varchar2(400),
                             house_vector vector
);
```

Allows finding data that is semantically similar to an input

Models are pluggable

No ML expertise required

DBAs and Developers can learn and use AI Vector Search in minutes

Find houses that are similar to this picture

```
SELECT   …
FROM     house_for_sale
ORDER BY vector_distance(house_vector, :input_vector);
```

Allows queries that combine AI vector search with business data about customers and products

Combines customer data, product data, and AI search in a few lines of SQL!

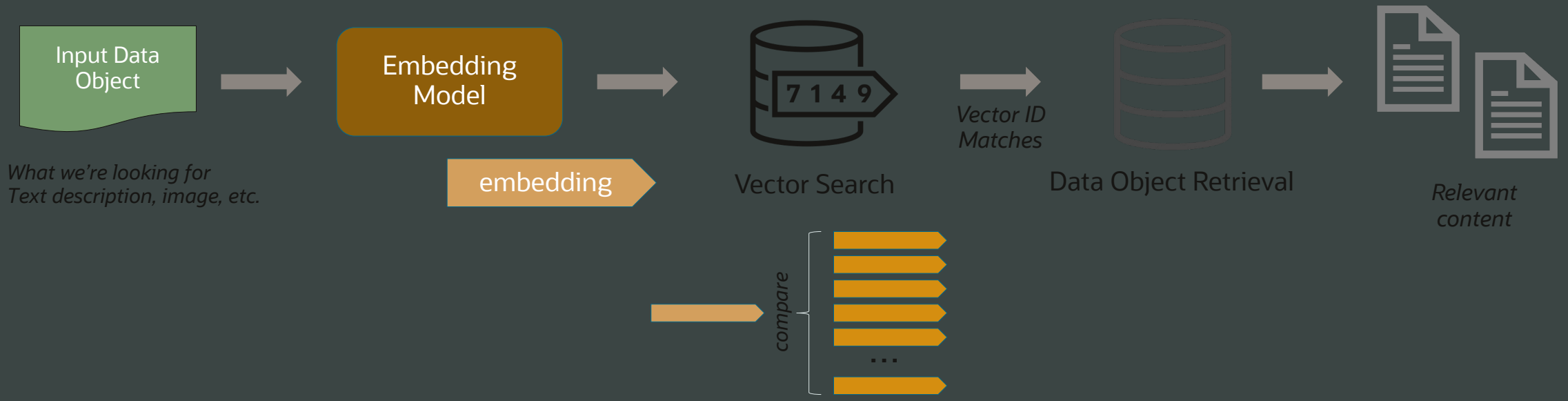
A single integrated solution, all data is fully consistent

Find houses that are similar to this picture and match the customer's preferred city and budget

```
SELECT …
FROM   house_for_sale
WHERE  price <= (SELECT budget      FROM customer …)
AND    city  in (SELECT search_city FROM customer …)
ORDER BY vector_distance(house_vector, :input_vector)
FETCH APPROX FIRST 10 ROWS ONLY
WITH TARGET ACCURACY 95 PERCENT;
```

Simple vector search

Pipeline



Copyright © 2023, Oracle and/or its affiliates

Retrieval Augmented Generation (RAG)

LLMs may have information that is outdated, only public, can not be referenced to a source (provenance), provide hallucinations, …

AI Vector Search can augment Generative AI by retrieving additional, often private, content needed to answer questions more accurately

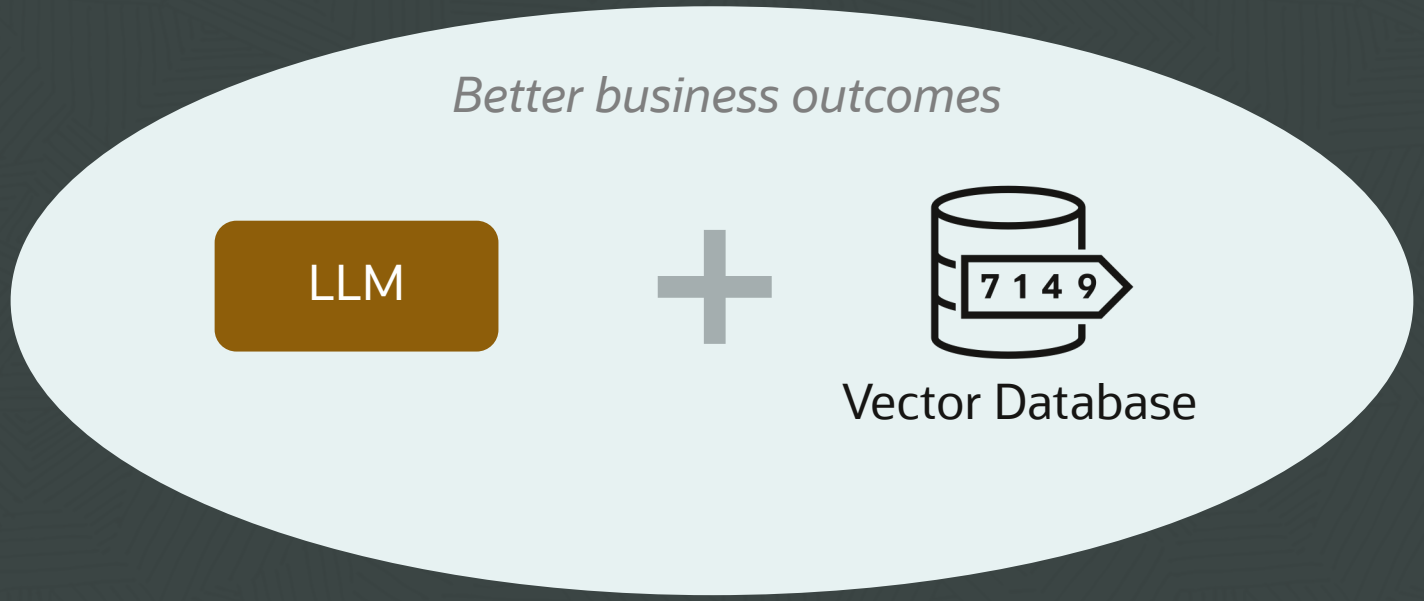
RAG addresses this by reusing inference model with additional data but without the need to retrain.

Oracle Database Vector AI allow for the rapid coding of new data, and searches against that data to feed into the LLM.

Copyright ©o 2023, Oracle and/or its affiliates - Confidential: Internal Only

Role of vector databases with LLMs

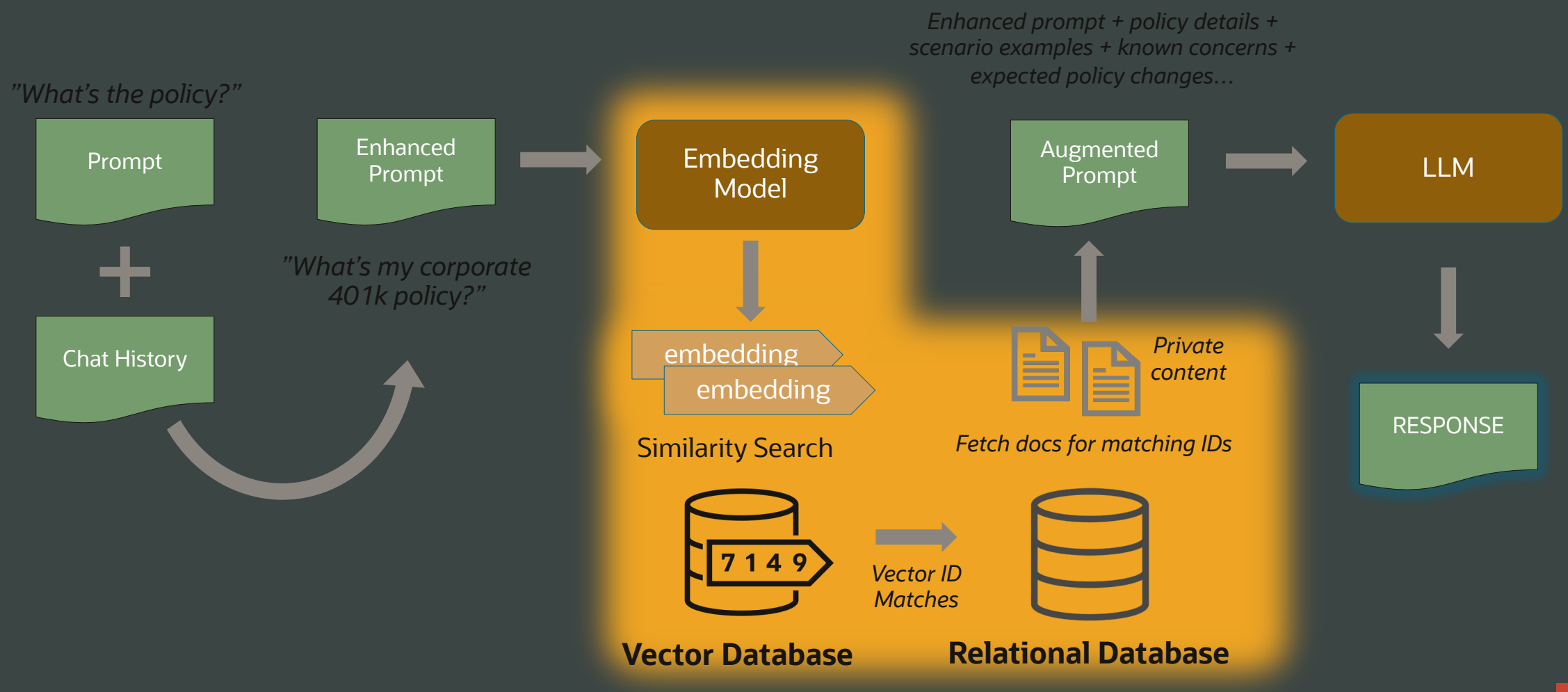
Avoid using sensitive customer data for LLM training and fine-tuning
Cheaper than fine-tuning LLMs, which can be expensive to update
Real-time updated knowledgebase
Cache previous LLM prompts/responses to improve performance and reduce costs



Copyright © 2023, Oracle and/or its affiliates

LLM-based chatbot with “enterprise knowledge”

Extended pipeline using retrieval architecture

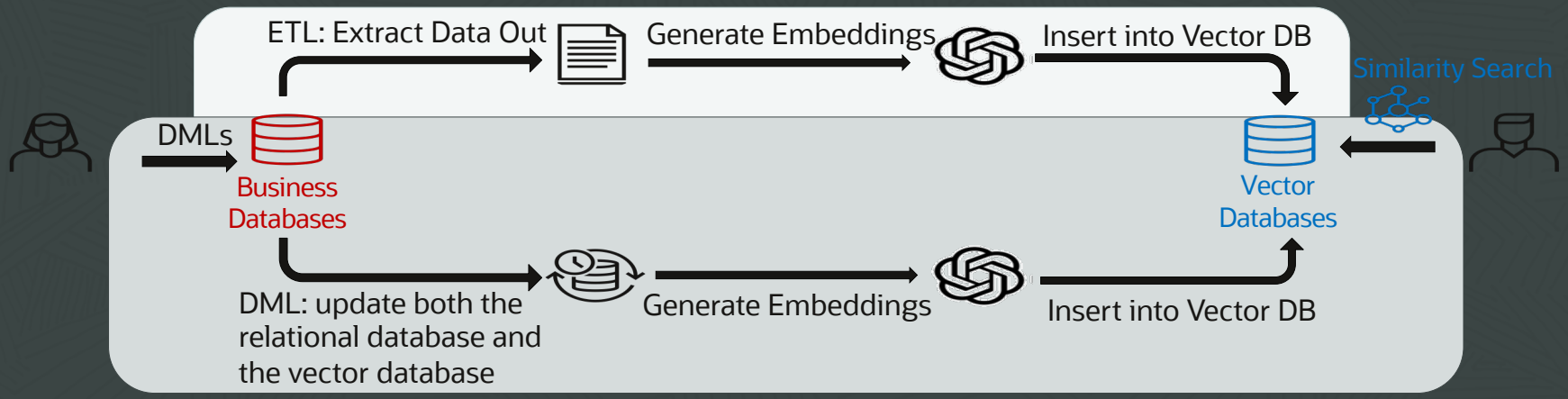


 Copyright © 2023, Oracle and/or its affiliates

Oracle's Intrinsic Advantage with Vector Search

Oracle already holds **most of the world's operational enterprise data**

It's an **order of magnitude easier** to **add vector embeddings to enterprise data** than replicating enterprise data to a separate vector database to perform similarity search



Problems:
Data Fragmentation
Data Copy
Data Divergence

It's easy to add vector search capability into a converged, enterprise-grade business database

- *But is it easy to add enterprise-grade relational database functionality into a vector database?*

Copyright © 2023, Oracle and/or its affiliates

AI Vector Search | Complex, Converged SQL

Oracle is a converged database that supports all types of workloads and data models:

- Graph, Text, JSON, Spatial, Relational, etc.

But it also support the full gamut of SQL, including complex operators and functionality:

- Window analytic functions, stored procedures, aggregation

What you get is support for **Complex, Converged SQL**:

Show me the top 3 photos, grouped by year, over the past 5 years, based on similarity to a provided query image. The photos should have been taken within20 miles of San Francisco, and have been viewed by at least 100 different people.

No purpose-built Database (let alone Vector Db) can do this.

 Copyright © 2023, Oracle and/or its affiliates

Oracle AI Vector Search is Fully Integrated

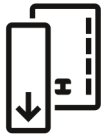
Seamless Integration with core database features for enterprise-grade performance and reliability

RAC

Sharding

Partitioning

Exadata



Transactions

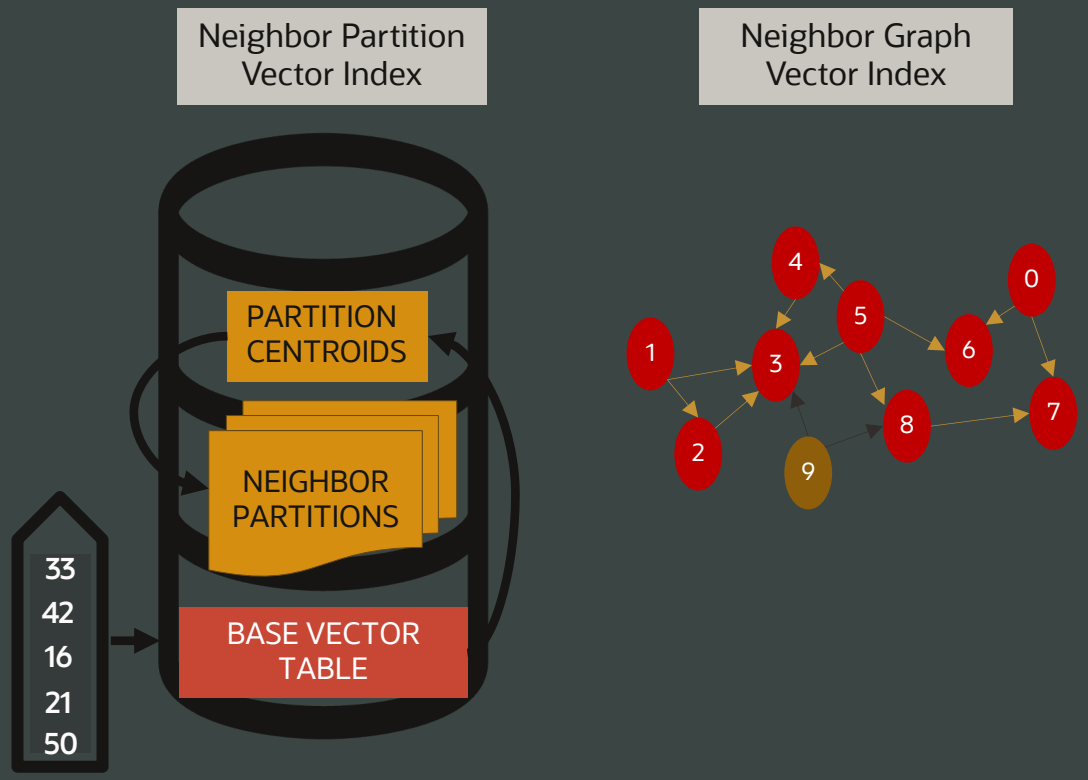
Parallel Execution

Analytics

Security

Copyright © 2024, Oracle and/or its affiliates

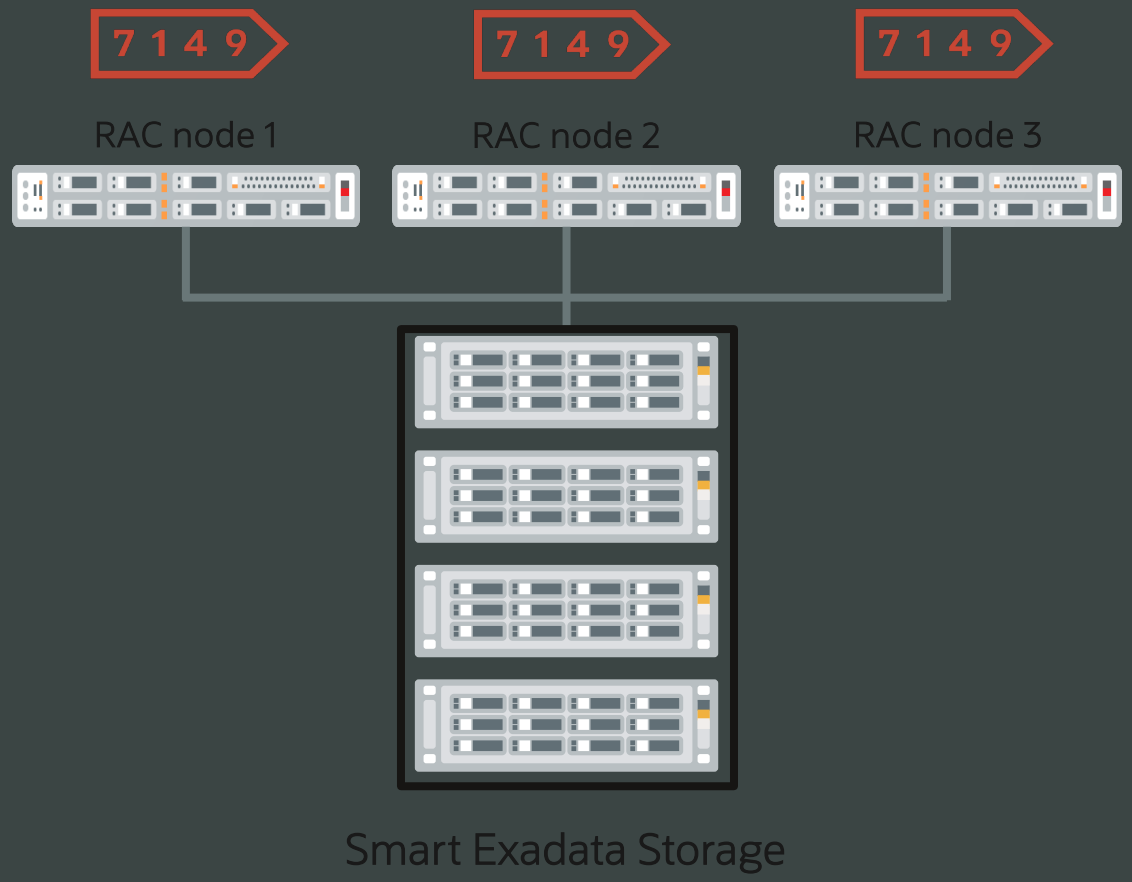
AI Vector Search | Transactions



Oracle's AI Vector Search Indexes maintain transactional consistency with DML activity

Copyright © 2023, Oracle and/or its affiliates

AI Vector Search | Scale-Out with RAC

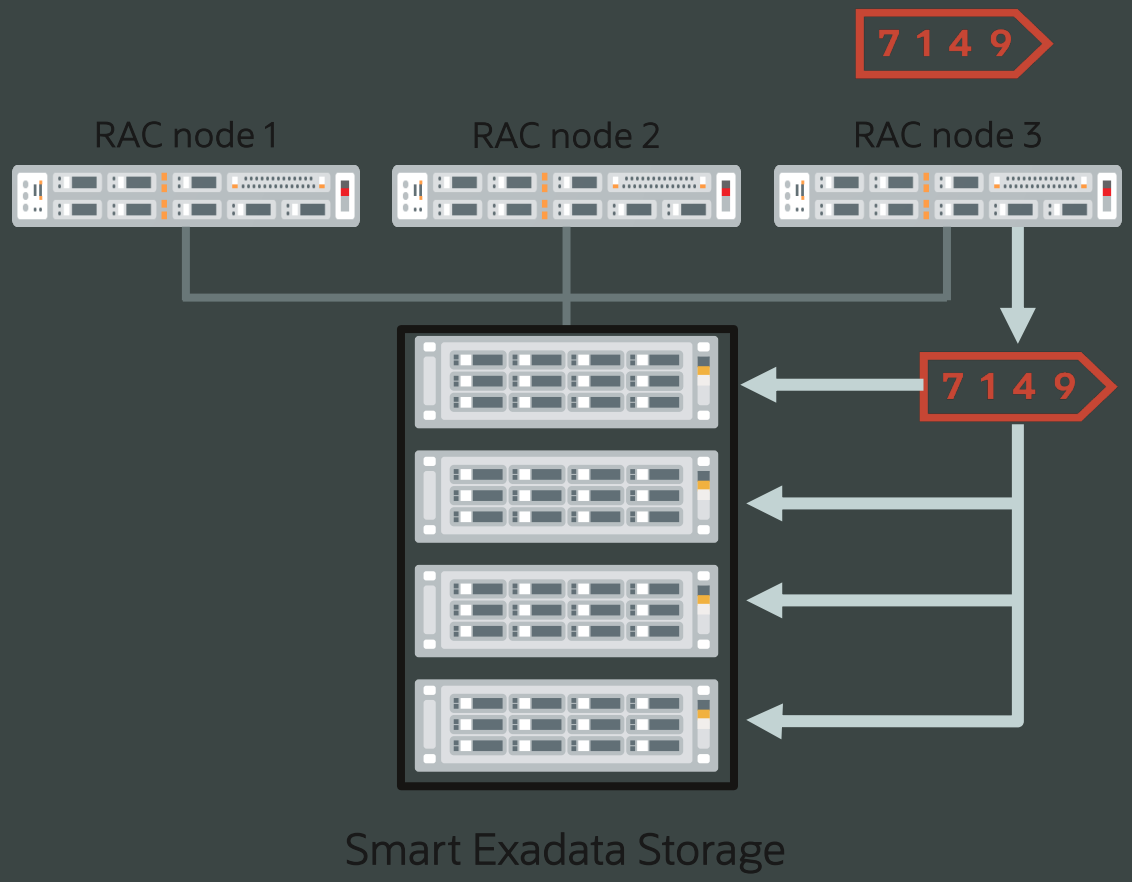


Oracle transparently scales vector processing across the computers in a RAC cluster

With full data consistency

Copyright © 2023, Oracle and/or its affiliates

AI Vector Search | Scale-Out with Exadata



Oracle AI Vector search can be transparently offloaded to smart Exadata storage for faster search

Copyright © 2023, Oracle and/or its affiliates

Vector search continues to evolve…

Copyright © 2023, Oracle and/or its affiliates

OCI Generative AI Agents

Why use OCI Generative AI Agents?

Converse directly with your enterprise knowledge bases

Leverage a conversational interface that lets anyone query enterprise data stores in natural language.

Get up-to-date results

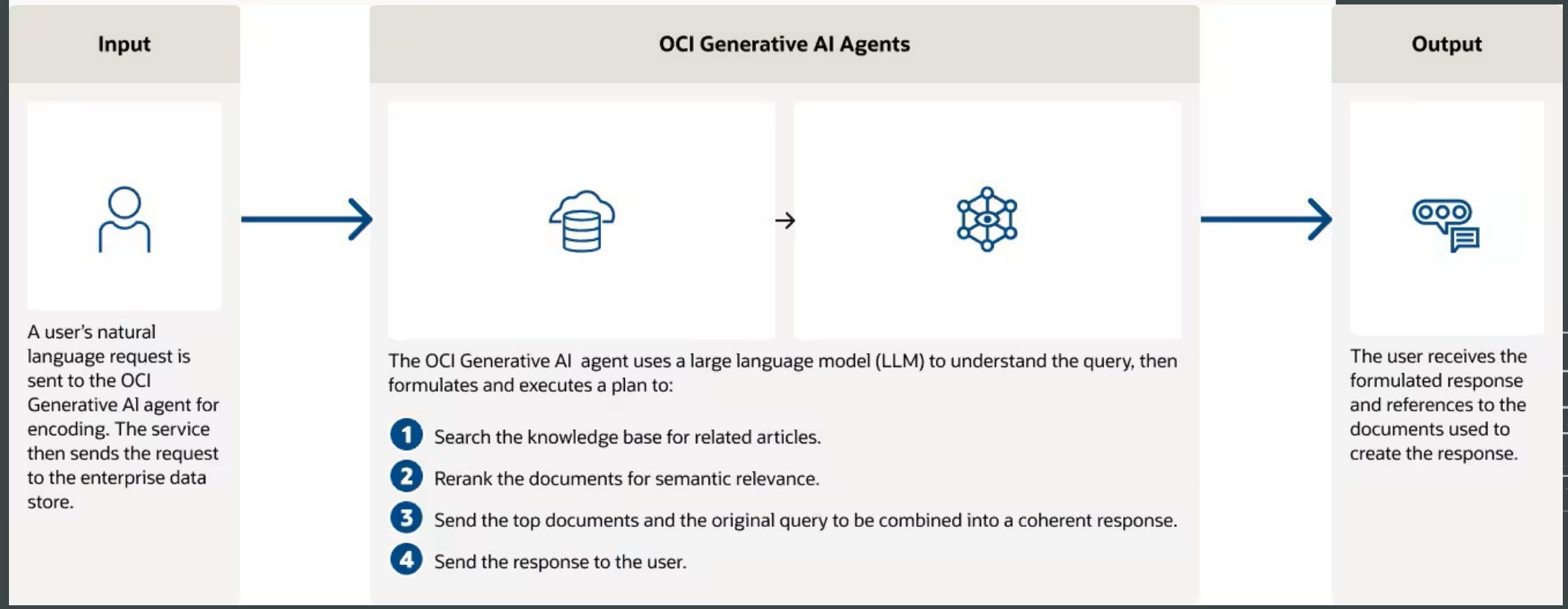
Receive dynamic, real-time responses, even on fast-changing enterprise data stores.

Embedded for actionable results

Create and embed custom AI agents into enterprise business applications and processes.

Scale your search

Get faster results through RAG with OpenSearch, with AI Vector Search in Oracle Database 23ai.



Copyright © 2024, Oracle and/or its affiliates

The fallacy of the “quick” chatbot

Many articles present a simple chatbot that can be created very quickly but these are not production-ready – why not?

It’s not as easy as you may think…

Many AI projects fail miserably. So today, let’s have a look at the recent news on McDonald's AI fiasco at its drive thrus.

An AI-powered voice ordering technology was developed in partnership with IBM and launched at 100 McDonald’s drive-thru locations across the United States.

But a series of viral TikTok videos started appearing recently where the AI failed to comprehend different accents, got confused by background noise and misunderstood and messed up orders.

Case in point: it gave one person hundreds of dollars worth of chicken nuggets and gave another customer an ice cream cone topped with bacon. 🤢

Suffice it to say no brand wants this kind of ridicule as publicity.

McDonald's just announced that it will shut off the AI system by the end of July.

What are some common challenges?

- Choosing the right model
- Context window size
- The quality of the corpus – structure, content, formatting, size, presence of distractors
- The preciseness or ambiguousness of the question
- The quality of the prompt
- Domain-specific specialized language not seen in model pre-training
- Dealing with information after the model pre-training was completed

There are some aspects of working with LLMs that are extremely expensive (in terms of time and/or money)

- Testing/evaluating model performance with human feedback or human-created datasets
- Fine tuning models
- Creating embeddings for large corpora

LLMs tend to “make up” answers if they don’t know the answer – formally called “hallucination”

Success factors

Creating a production-ready chatbot

What are the important choices and success factors?

- Model choice for embedding and inferencing
- Chunking strategy and size
- Vector similarity algorithm
- Preparation of the corpus
- Prompt engineering
- Model parameters
- Memory
- Re-ranking
- Agents
- Evaluation

This is an active area of research in industry and academia and there are many promising emerging techniques including:

- Re-ranking
- Prompt tuning
- Sentence Window Retrieval
- Retrieval Augmented Fine Tuning
- Parameter Efficient Fine Tuning
- Low Rank Adaptation

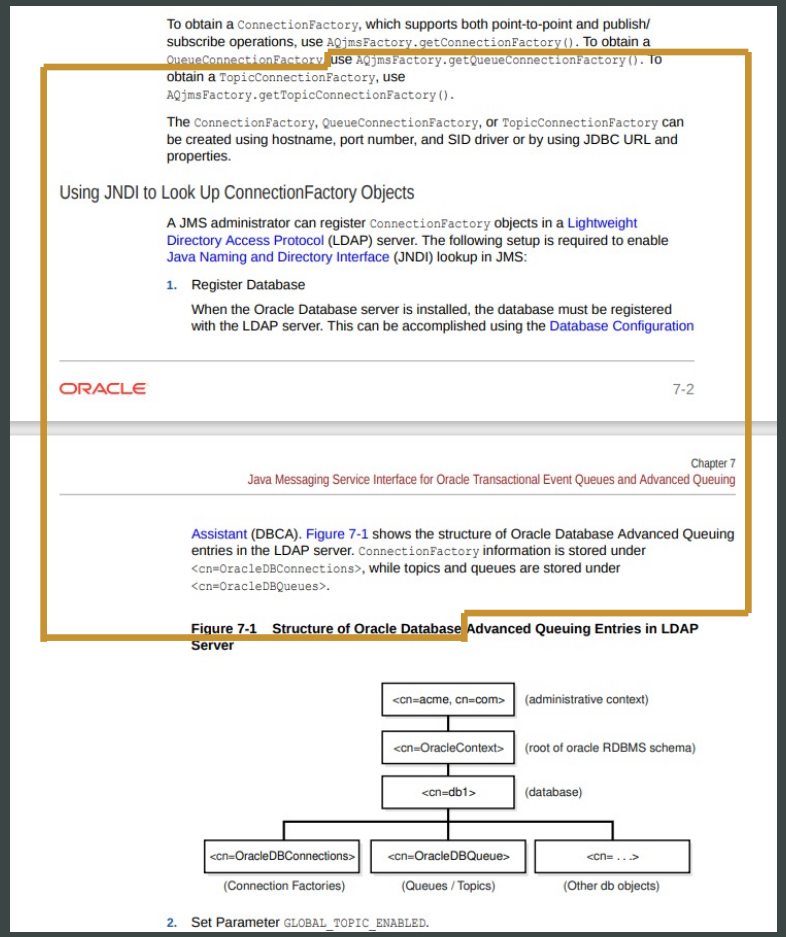
Model choice

LLMs are trained for specific tasks and using defined datasets

- Models may be optimized for embeddings, inferencing, summarization, entity extraction, or other tasks
- Some models are general purpose, i.e., can perform different tasks

Model Type	Optimized Tasks	Examples
Embedding-focused LLMs	Generating embeddings for downstream tasks	BERT, RoBERTa, ALBERT, DistilBERT
Inference-focused LLMs	Fast and efficient inference	GPT-3, GPT-Neo, T5
General-purpose LLMs	Broad applicability across various tasks	GPT-3, GPT-Neo, T5, BERT, RoBERTa
Multimodal LLMs	Handling text and multimodal inputs	CLIP, ALIGN, LXMERT, UNITER
Multilingual LLMs	Handling multiple languages	mBERT, XLM-R, mT5, MarianMT
Domain-specific LLMs	Tailored for specific domains or tasks	BioBERT (biomedical), SciBERT (scientific text)

Chunking



To obtain a `ConnectionFactory`, which supports both point-to-point and publish/subscribe operations, use `AQjmsFactory.getConnectionFactory()`. To obtain a `QueueConnectionFactory`, use `AQjmsFactory.getQueueConnectionFactory()`. To obtain a `TopicConnectionFactory`, use `AQjmsFactory.getTopicConnectionFactory()`.

The `ConnectionFactory`, `QueueConnectionFactory`, or `TopicConnectionFactory` can be created using hostname, port number, and SID driver or by using JDBC URL and properties.

Using JNDI to Look Up ConnectionFactory Objects

A JMS administrator can register `ConnectionFactory` objects in a Lightweight Directory Access Protocol (LDAP) server. The following setup is required to enable Java Naming and Directory Interface (JNDI) lookup in JMS:

1. Register Database

   When the Oracle Database server is installed, the database must be registered with the LDAP server. This can be accomplished using the Database Configuration

ORACLE 7-2

Chapter 7
Java Messaging Service Interface for Oracle Transactional Event Queues and Advanced Queuing

Assistant (DBCA). Figure 7-1 shows the structure of Oracle Database Advanced Queuing entries in the LDAP server. `ConnectionFactory` information is stored under `<cn=OracleDBConnections>`, while topics and queues are stored under `<cn=OracleDBQueues>`.

Figure 7-1 Structure of Oracle Database Advanced Queuing Entries in LDAP Server

2. Set Parameter `GLOBAL_TOPIC_ENABLED`.

Chunking is more difficult that you'd expect, for example consider this chunk of a PDF

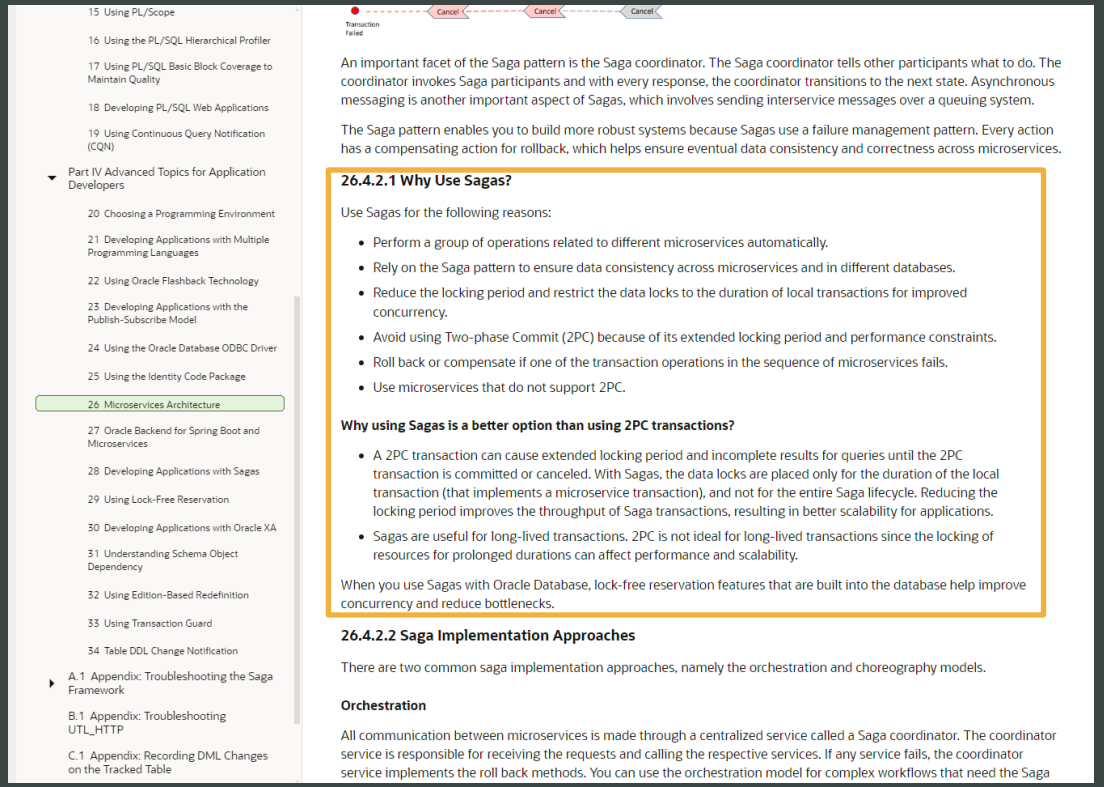
- It spans multiple topics which may not be related
- It includes headings, page headers and footers in the middle of the text
- It includes part of some sentences
- It does not understand the content of images at all

The quality of the chunks has a direct impact on the quality of the context provided to the LLM

Structured Chunking

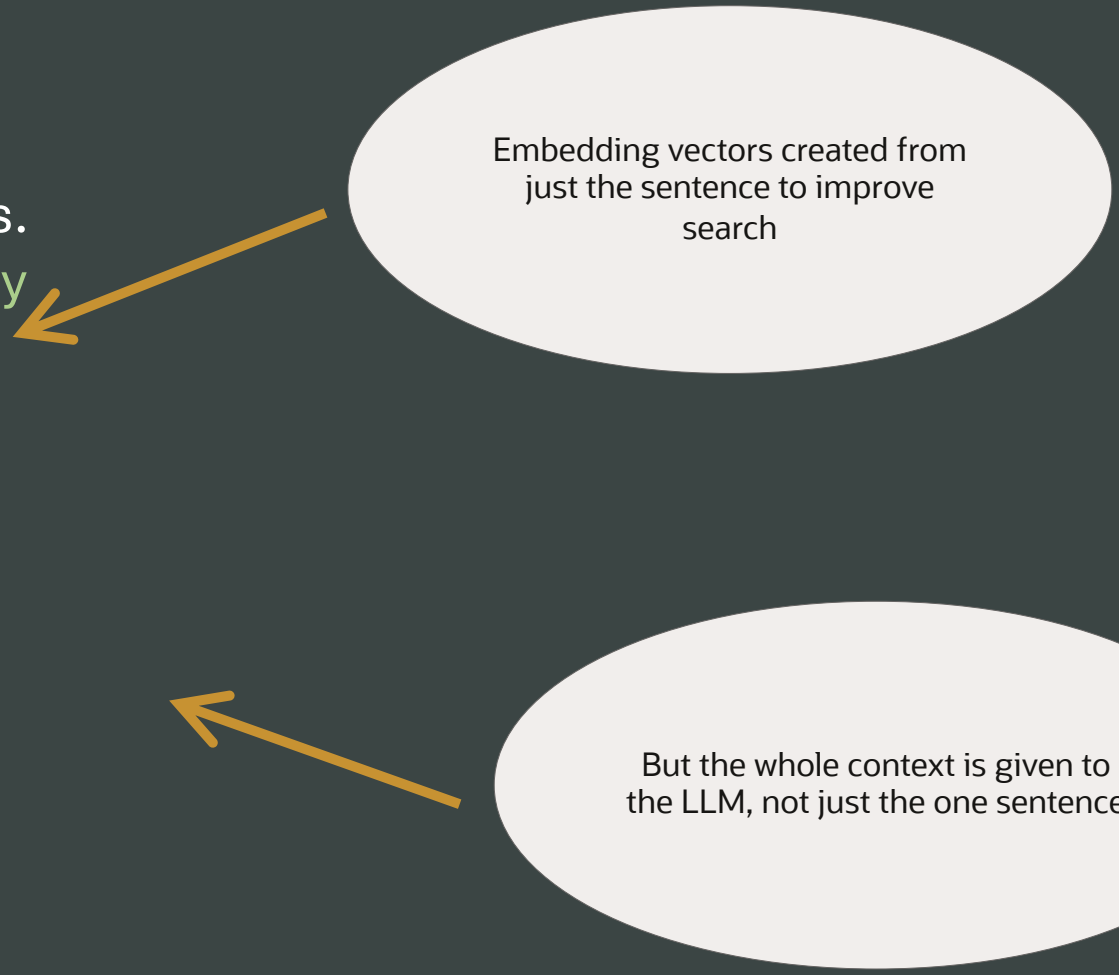
For example, using HTML headings to understand the structure of the document

- Captures a single topic per chunk
- Retains the hierarchical structure of the document
- Avoids inclusion of spurious text like page headers, footers, etc.
- May be further split into text chunks if too large for the chosen model



Sentence Window Retrieval

She lay on the operating table unconscious. Jones made a deep cut in her leg with a very sharp knife. He was able to remove the bullet and repair the damage to the blood vessels in the area. Ultimately, this saved her life.



Understanding the context window

Each LLM has a specific size context window which determines the largest amount of information that it can work with

For example, GPT-3's context window is 2000 tokens

Context window
System instructions
Question
Context (documents from the RAG)
Memory (chat history, usually summarized)
Agent work area
Response

Suppose the window was 4k tokens in size, this could be used as follows:

- 512 tokens for chat memory
- 512 tokens for agent work area
- 512 tokens for system instructions, question and response

This leaves enough room for only three documents from the RAG if the chunk size is 800 tokens – smaller chunks means more documents but less context

Prompt engineering

- Models expect specific prompt formatting including delimiters and placeholders
- The prompt can give the model extra information about the task to be performed
- Prompts have significant impact on the quality of responses
- Significant effort can go into optimizing the prompt for the intended use case

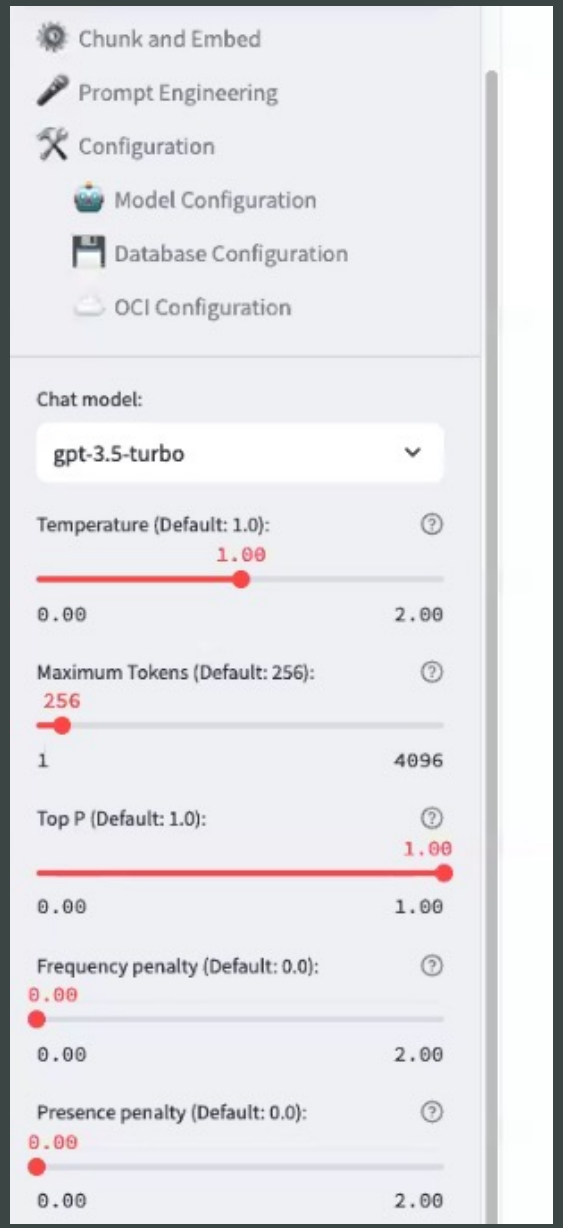
```
<|begin_of_text|><|start_header_id|>system<|end_header_id|>

You are a friendly, helpful and honest assistant.
You are given parts of a long document and a question. Use the
provided document parts to answer the question.
If you do not know the answer, just say "I don't know". Do not make up
an answer.
After your answer, tell the human which document parts you used to
answer the question.

Question: {question}

Context:
{context}<|eot_id|><|start_header_id|>assistant<|end_header_id|>
```

Example of a RAG prompt for Llama 3

Model parameters

Models have parameters that affect the responses created, including:

- Temperature controls how “creative” the response is
- Top K, Top P, frequency penalty, presence penalty, and others control how many candidates are considered when choosing the next token
- Maximum tokens controls the length of the response



Is there an approach that avoids the downsides?

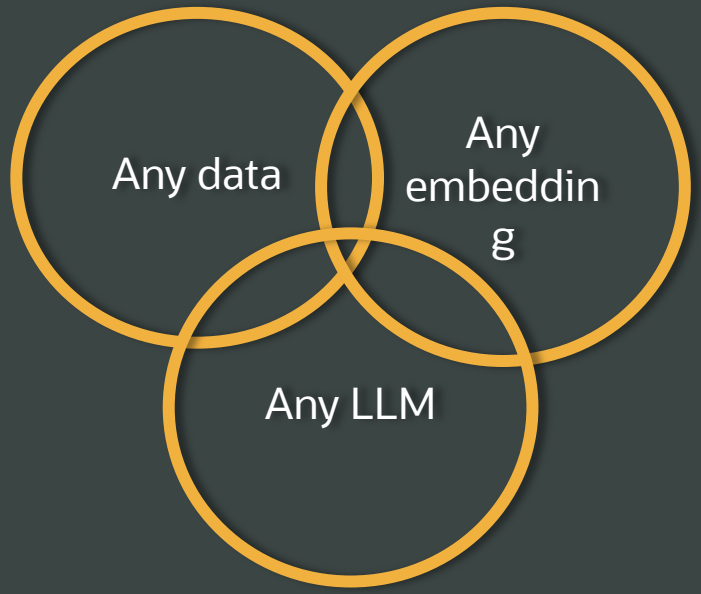
The most important success factor in developing a real chatbot is ***experimentation*** and ***iteration***

Oracle AI Playground

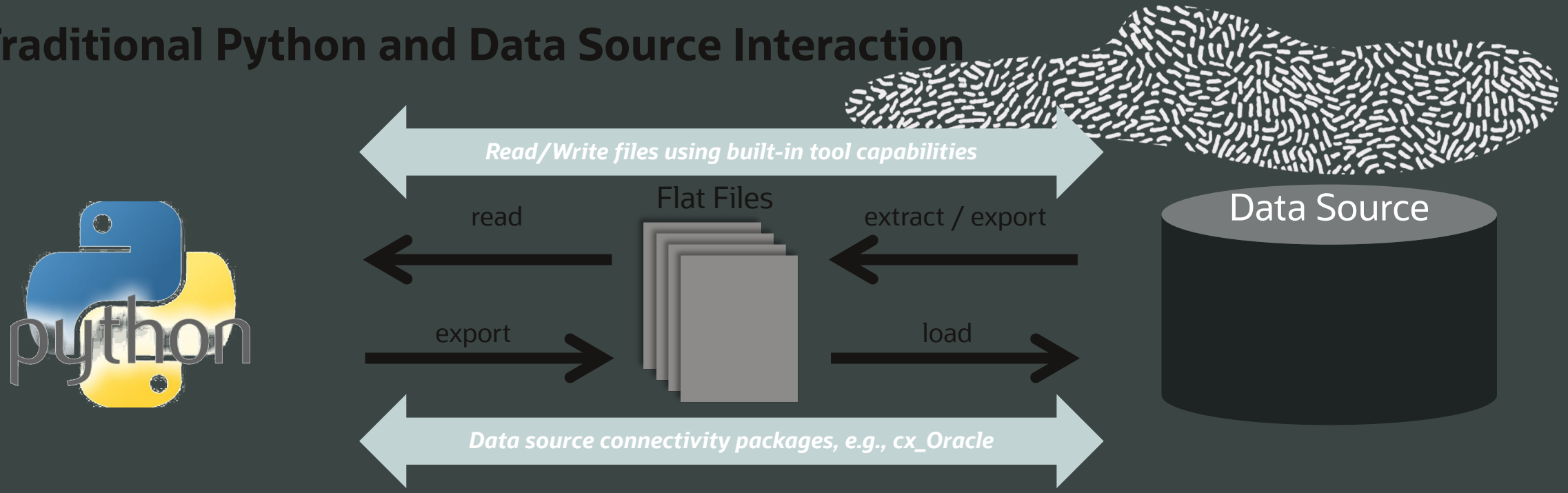
Developer release coming soon…

ORACLE

AI Playground



Traditional Python and Data Source Interaction



Access latency
Paradigm shift: Python → *Data Access Language* → Python
Memory limitation – data size, in-memory processing
Single threaded
Issues for backup, recovery, security
Ad hoc production deployment

Copyright © 2024, Oracle and/or its affiliates

Oracle Machine Learning for Python (OML4Py)

Empower data scientists and Python users on Oracle Database and Autonomous Database

Leverage your database as a high-performance compute engine

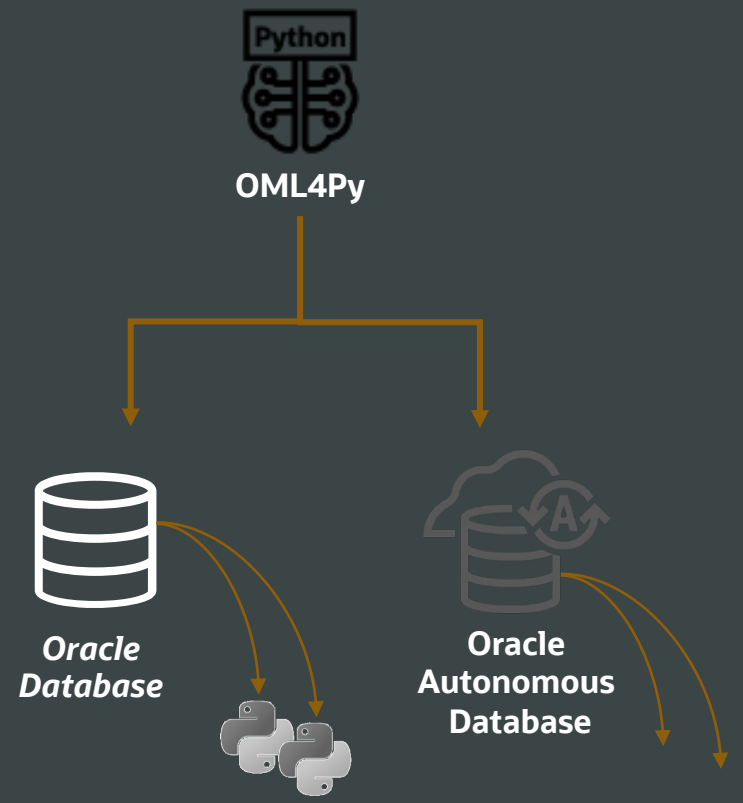
Use in-database optimized ML algorithms

Store and manage Python objects and user-defined functions in your database

Augment solutions with third-party Python packages

Invoke user-defined Python functions from SQL and REST

Use AutoML API with in-database ML algorithms



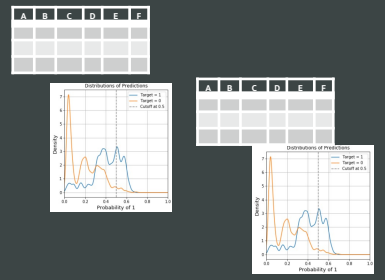
Copyright © 2024, Oracle and/or its affiliates | Confidential: Internal/Restricted/Highly Restricted

matplotlib

pandas

scikit learn

{REST:API}
SQL



```
def my_function (arg):
    import sklearn
    ...
    return result
```

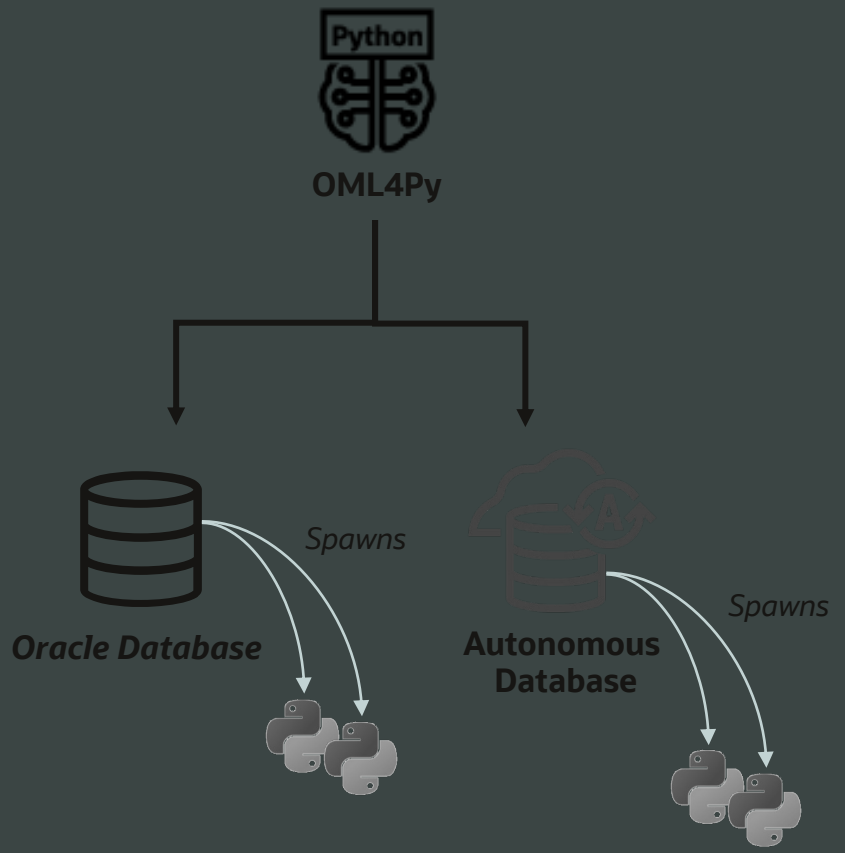
python

Copyright © 2022, Oracle and/or its affiliates

Embedded Execution

Example of parallel partitioned data flow using third party package using OML4Py

```
# user-defined function using sklearn
def build_lm(dat):
  from sklearn import linear_model
  lm = linear_model.LinearRegression()
  X = dat[['PETAL_WIDTH']]
  y = dat[['PETAL_LENGTH']]
  lm.fit(X, y)
  return lm

# select column(s) for partitioning data
index = oml.DataFrame(IRIS['SPECIES'])

# invoke function in parallel on IRIS table
mods = oml.group_apply(IRIS, index,
                       func=build_lm,
                       parallel=2)
mods.pull().items()
```



Copyright © 2024 Oracle and/or its affiliates.

Custom third-party packages on ADB via OML Notebooks

Expanding ADB as a platform for data science and machine learning

Support 3rd party Python package installation and conda environment creation

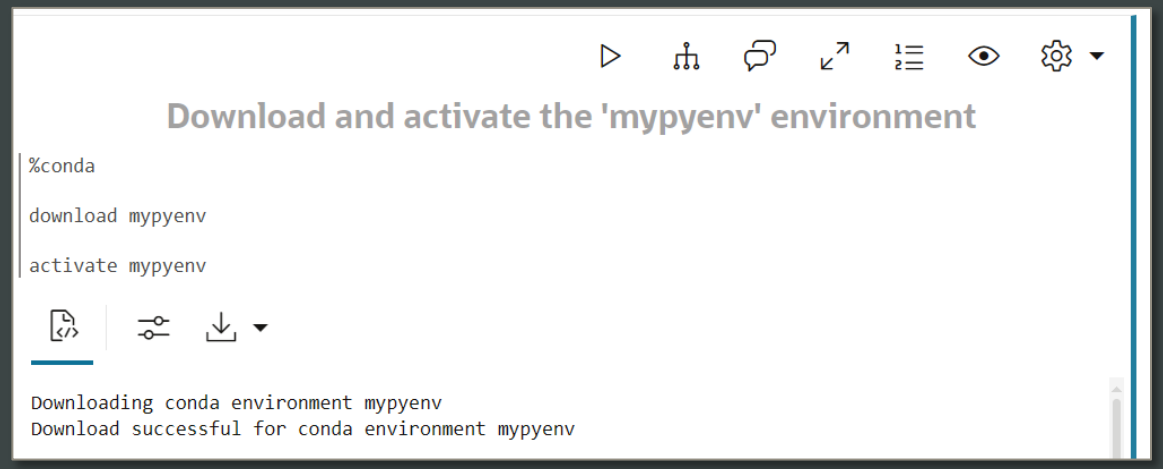
- **conda** - open-source package and environment management system
- Admins install third-party packages and manage conda environments
- Users download and activate conda environments from Object Storage
- Environments run in a separate container for security

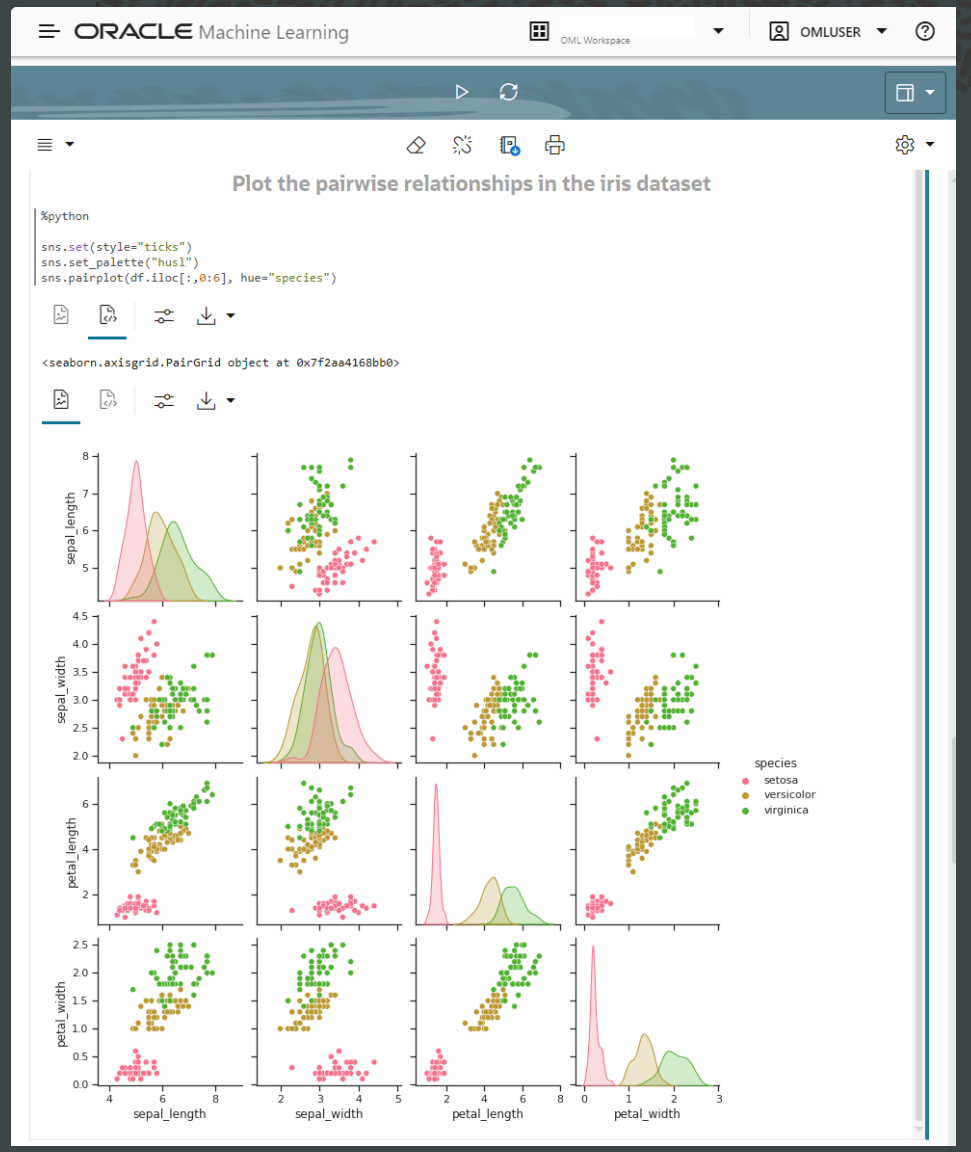
Use with Python interpreters in OML Notebooks

Use with embedded execution in OML4Py

Copyright © 2024, Oracle and/or its affiliates

Conda environment usage via OML Notebooks on ADB



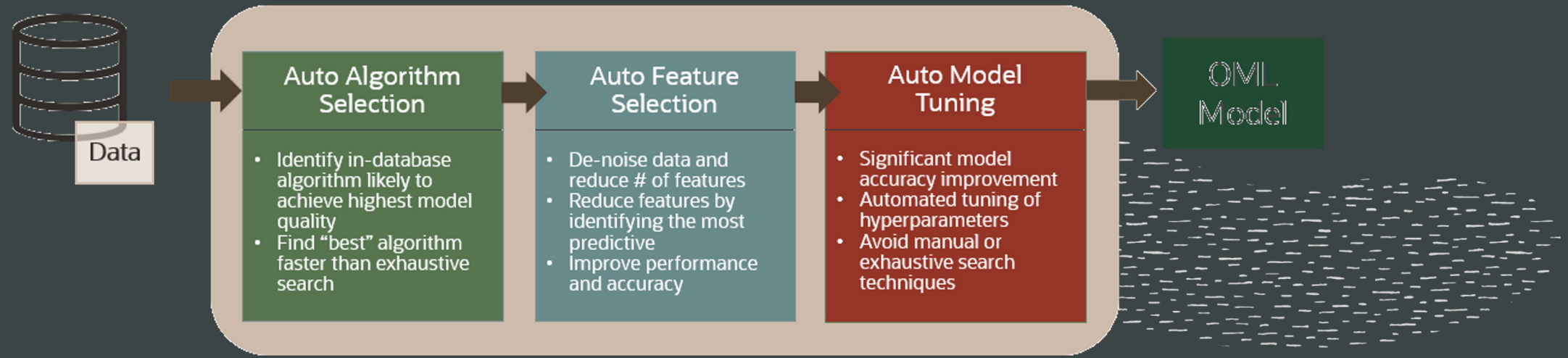


Copyright © 2024, Oracle and/or its affiliates

OML4Py AutoML

Enhance data scientist productivity and enable Python users with ML

- Eliminate repetitive tasks of model building/evaluation to increase user productivity
- Enable non-expert users to leverage machine learning
- Apply ML to the ML process to reduce algorithm and hyperparameters search space and reduce compute time and cost



Copyright © 2022 Oracle and/or its affiliates.

OML4Py AutoML API

Algorithm Selection

```
%python

as_wine_cl = automl.AlgorithmSelection(mining_function='classification',
                                       score_metric='accuracy', parallel=2)

wine_alg_ranking_cl = as_wine_cl.select(WINE_X_cl, WINE_y_cl, k=4)

print("Ranked algorithms:\n", win

selected_wine_alg_cl = next(iter(
print("Best algorithm: ", selecte
```

```
Ranked algorithms:
 [('svm_gaussian', 0.9825515947467
2114714554), ('rf', 0.949547038327
Best algorithm:  svm_gaussian
```

Feature Selection

```
%python

fs_wine_cl = automl.FeatureSelection(mining_function = 'classification',
                                     score_metric = 'accuracy', parallel=2)

selected_wine_features_cl = fs_wine_cl.reduce(selected_wine_alg_cl,
                                              WINE_X_cl, WINE_y_cl)

WINE_X_reduced_cl = WINE_X_cl[:,selected_wine_features_cl]

print("Selected columns:", W
print("Number of columns:")
"{}  reduced to {}".format(l
```

FINISHED

```
Selected columns: ['alcohol',
ls', 'color_intensity', 'hue'
Number of columns:
'13  reduced to 9'
```

Model Tuning

```
%python


mt_wine_cl = automl.ModelTuning(mining_function = 'classification', parallel=2)

results_cl = mt_wine_cl.tune(selected_wine_alg_cl, WINE_X_reduced_cl, WINE_y_cl)
tuned_model_cl = results_cl['best_model']
tuned_model_cl


Algorithm Name: Support Vector Machine

Mining Function: CLASSIFICATION
```

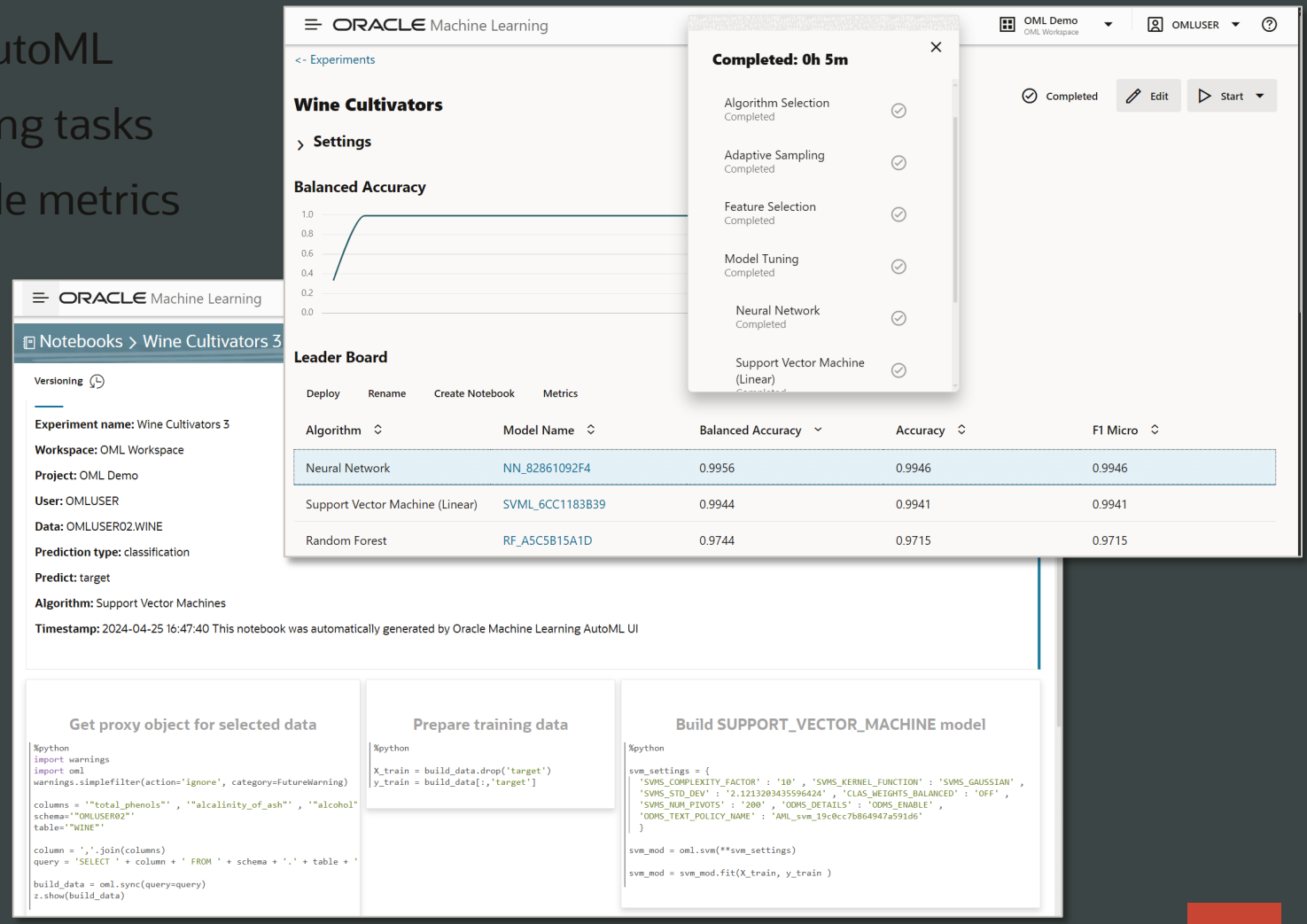
FINISHED

Copyright © 2024, Oracle and/or its affiliates

Generate notebooks with Python code using AutoML UI

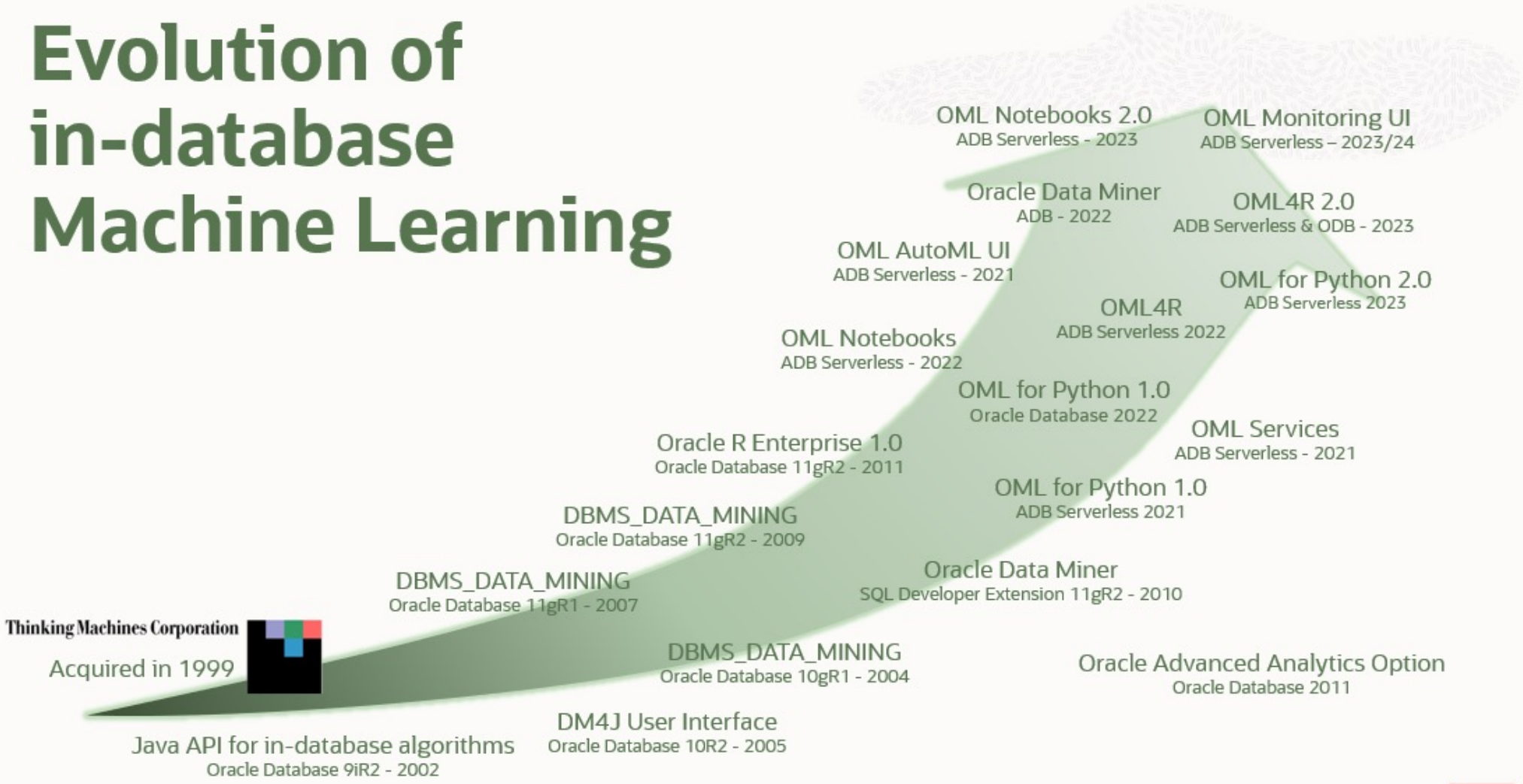
Enhance data scientist productivity and enable non-expert data professionals

- Accelerate ML projects with no-code AutoML
- Automate repetitive and time-consuming tasks
- Compare multiple models using multiple metrics
- Generate editable notebooks for desired models based on OML4Py
- Deploy ML models immediately using SQL or to OML Services as REST endpoints



Copyright © 2024, Oracle and/or its affiliates

Evolution of in-database Machine Learning



Copyright © 2024, Oracle and/or its affiliates

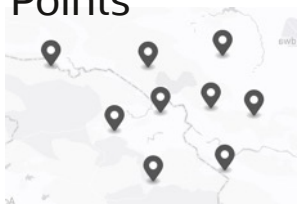
Copyright © 2024, Oracle and/or its affiliates

Oracle Spatial

Native spatial type and spatial analysis in Autonomous Database

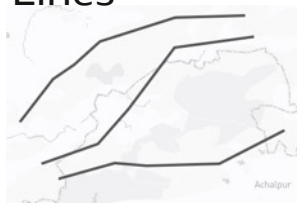
Points

Address locations
GPS locations
Earthquake locations



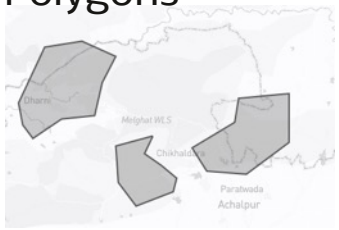
Lines

Roads
Utility lines
Storm path



Polygons

Counties
Flood zone
Sales regions



Which address locations are within a flood zone?

Which utility lines are within a mile of an earthquake location?

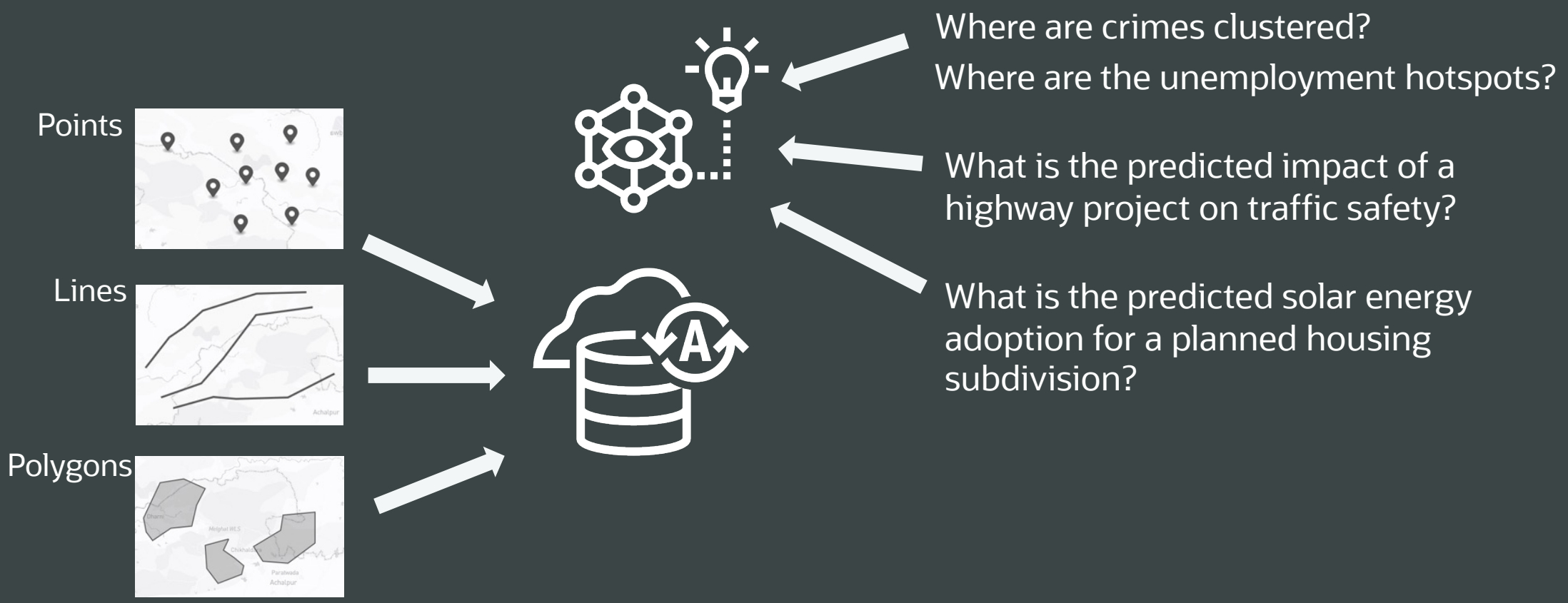
How far from a storm path is the nearest emergency response center?

Also 3D, point clouds, raster imagery, spatial networks and more

Copyright © 2024, Oracle and/or its affiliates

Oracle Spatial AI

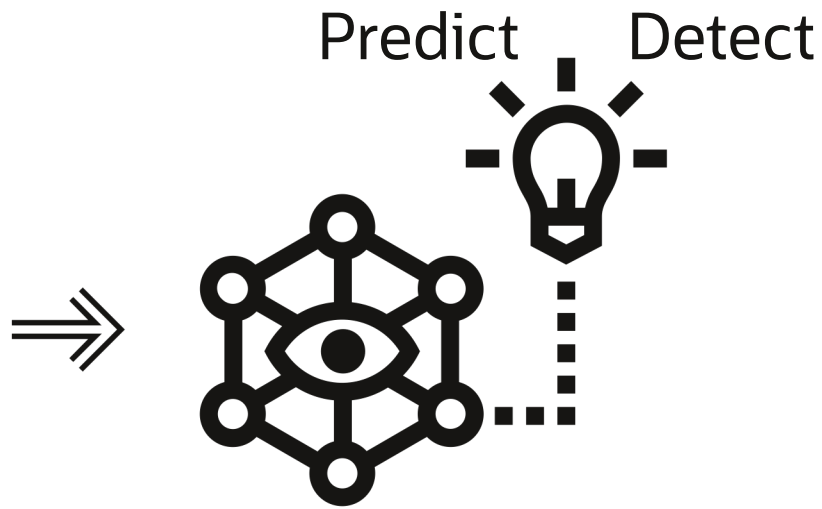
Spatial algorithms in Oracle Machine Learning for Python (OML4Py)



Copyright © 2024, Oracle and/or its affiliates

Spatial AI

"Derive insights from spatial data"

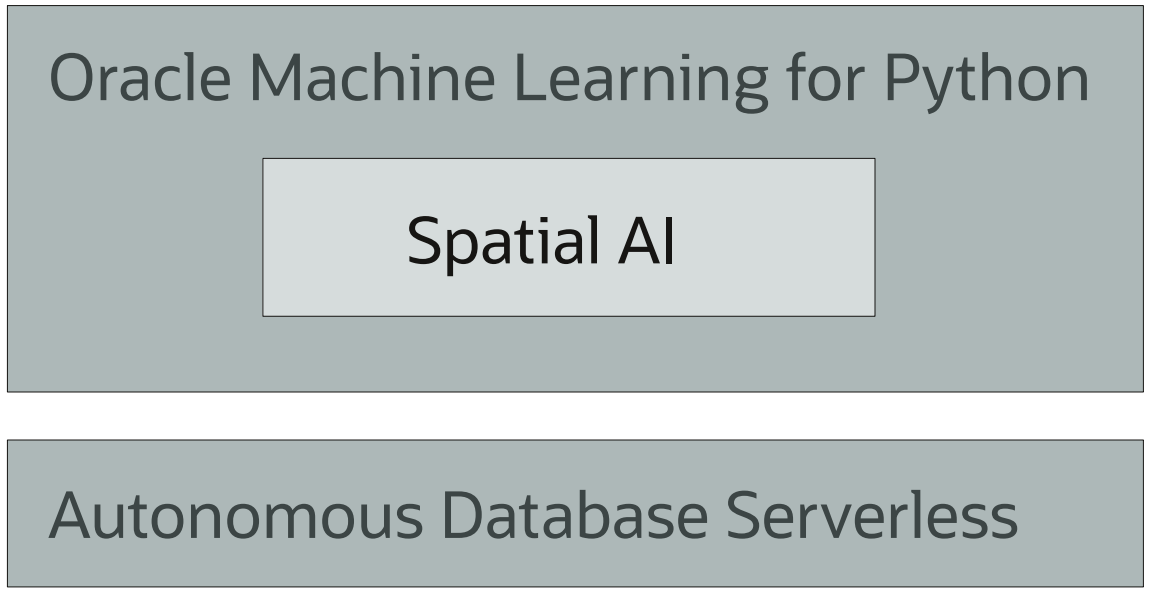


Copyright © 2024, Oracle and/or its affiliates

Common scenarios addressed by spatial algorithms

Scenario	Jargon	Example
Predictive influence of nearby values	“Spatial dependence”	Property values influenced by nearby values
Regional variation of predictive factors	“Spatial heterogeneity”	Predictive factors for vehicle purchase vary regionally
Geographic concentration of observations	“Spatial clustering”	Crime clusters, unemployment hotspots
Geographic alignment of observations	“Spatial colocation”	Successful sandwich shops tend to be in close proximity to desert shops

Oracle Spatial AI



- Python API for common Spatial SQL operators and functions
- Spatial pre/post processing
  - Feature engineering
  - Gap filling
  - Spatial weights/spatial lag
- Spatial algorithms
  - Regression
  - Classification
  - Clustering
  - Anomaly detection
  - Colocation
- Spatial pipeline
- Python, REST, SQL APIs to invoke models

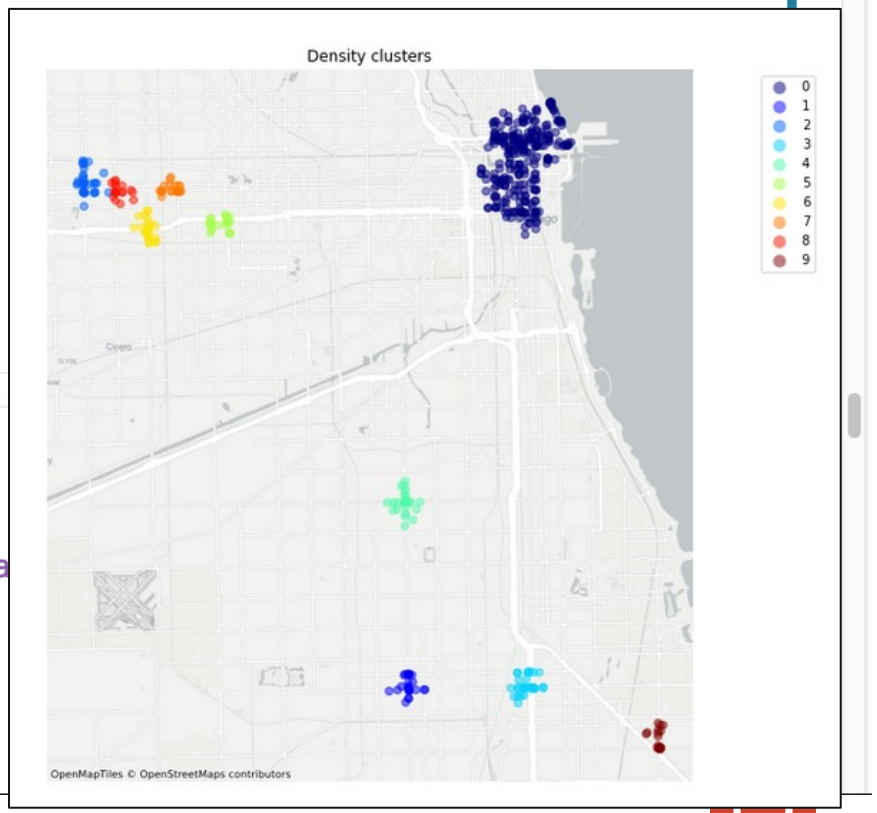
Detect density clusters in data from Oracle Spatial

Versioning low Zeppelin

```
%python

X = accidents_injury_sdf[['geometry']].to_crs("EPSG:3857")
labels = dbscan.fit_predict(SCoordTransformer().transform(X))
np.unique(labels)
```

```
array([-1,  0,  1,  2,  3,  4,  5,  6,  7,  8,  9])
```

```
%python

_, axs = plt.subplots(figsize=(10, 10))
plot_clusters(X, labels, title='Density clusters', with_noise=False, with_bounds=Fa
```



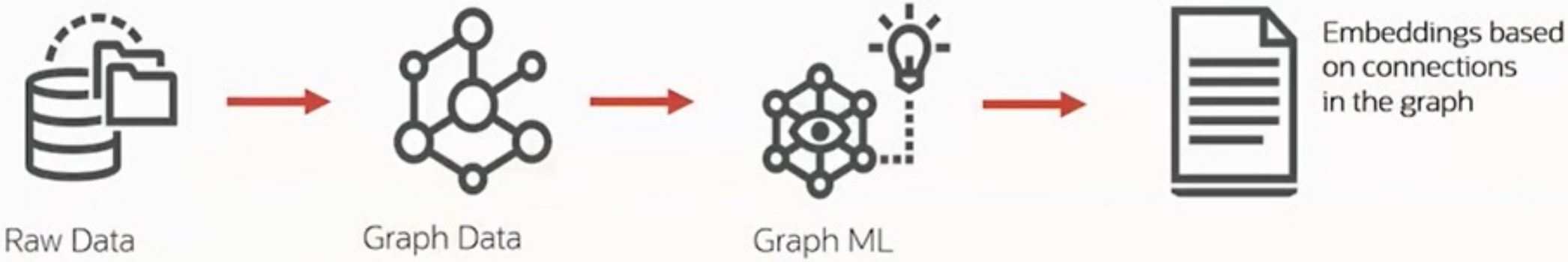
Copyright © 2024, Oracle and/or its affiliates

Spatial AI

- Library added to OML4Py on ADB-S
- Python API for common Oracle Spatial operations
- Spatial pre/post-processing
- Spatial ML algorithms and APIs

Copyright © 2024, Oracle and/or its affiliates

Graphs and Machine Learning



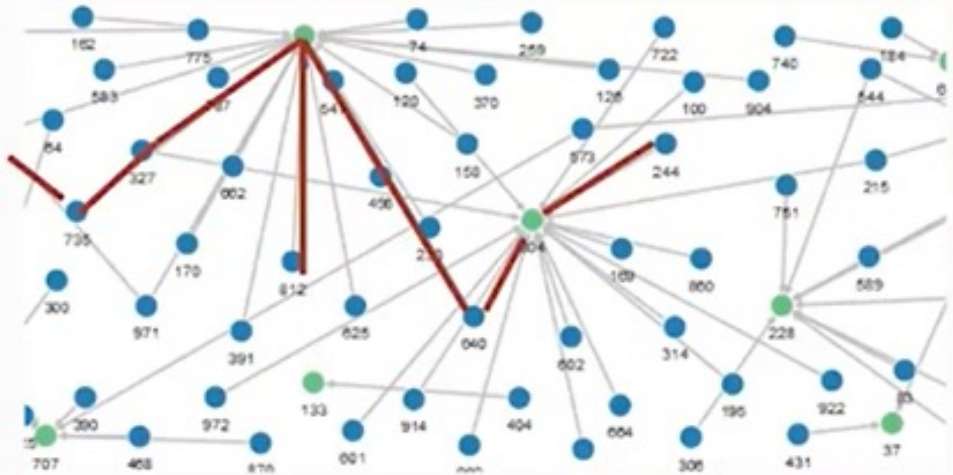
Embeddings capture

- Graph topology
- Node and edge features
- How information propagates in a graph: How node and edge features are influenced by neighbors

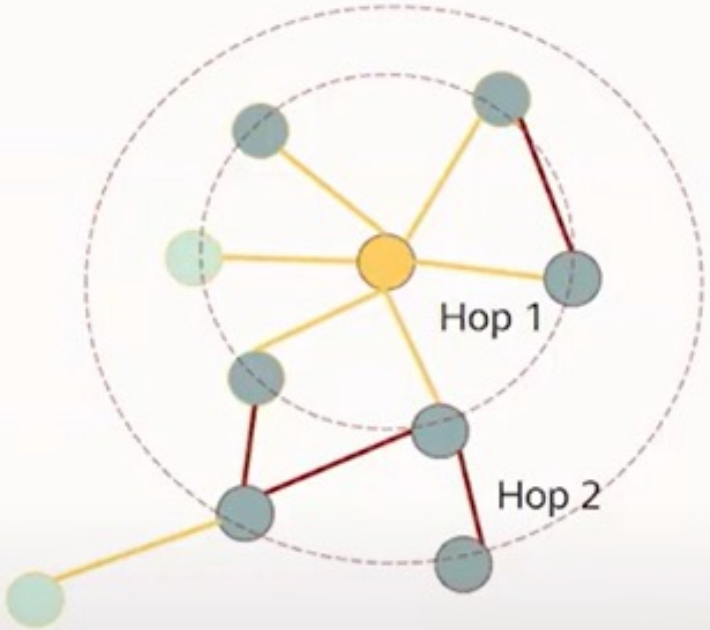
Copyright © 2024, Oracle and/or its affiliates

Graphs and Machine Learning

DeepWalk: Follow walks from each node



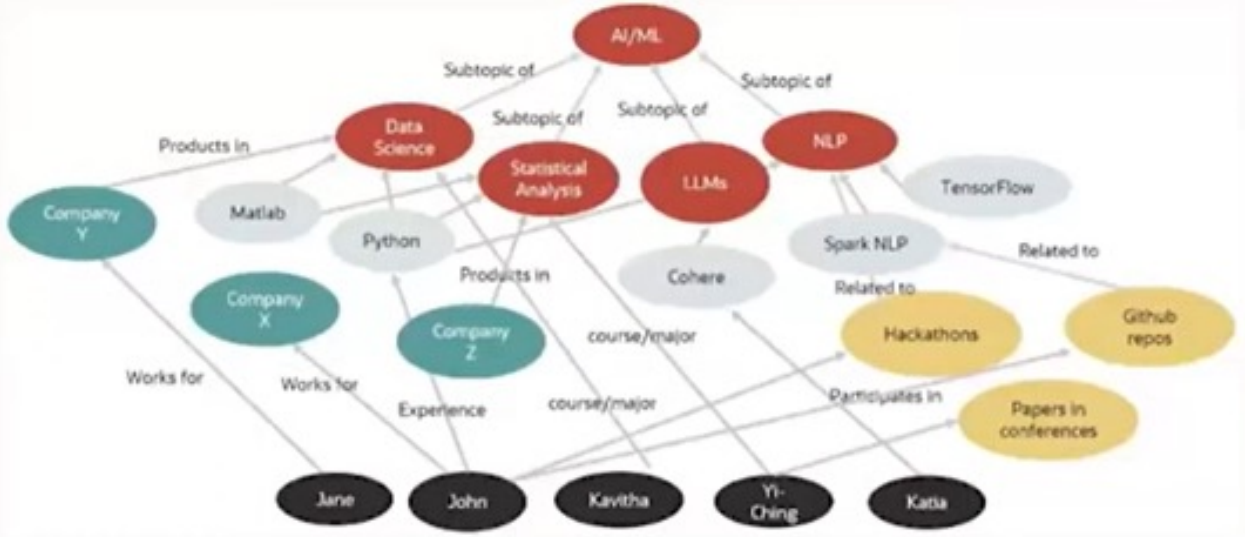
GraphWise: Sample neighbors recursively to learn a function



Copyright © 2024, Oracle and/or its affiliates

Knowledge Graphs and Retrieval Augmented Generation (RAG)

- Knowledge graphs capture expert information about a domain
- Generate embeddings from the knowledge graph using Graph ML techniques
- Use vector search for fast retrieval of top results
- Combine with user query in a prompt to LLMs



Copyright © 2024, Oracle and/or its affiliates

Recap, what we covered…

- OCI AI Services Oracle AI Stack
- Oracle Multi-purpose/Converged database
- Select AI (NL2SQL)
- Vector and RAG
- AI Agents
- AI Playground
- OML4PY
- Spatial AI
- Graph ML and RAG

Copyright © 2024, Oracle and/or its affiliates

Thanks for Attending!

Oracle Database 23ai

AI Solutions Hub

LiveLabs workshops

Paul Parkinson
Architect and Developer Advocate
Oracle Database

Develop AI/ML-Driven Apps with Autonomous Database 23ai

Copyright © 2024, Oracle and/or its affiliates