VERITAS

illuminate 2024

uConnect Hackathon 2024

Anomaly Detection for structured workloads like databases

SPate

Samruddhi Prashant Pate

Business Case Slide

Idea Description	
Problem Statement	Anomaly Detection for structured workloads like databases
Solution Benefits	The solution enhances cyber resiliency by proactively identifying and mitigating potential threats to database integrity and security, thereby minimizing risks of data breaches and system downtime.
Category	AI/ML-driven Cyber Resiliency Solution.
Theme	AI/ML, Cyber Resiliency

Key Points

Data Collection:

- Capture database activity logs, including queries, transactions, and system events.

Feature Engineering:

- Extract relevant features such as query frequency, data access patterns, and transaction volume.

Model Training:

- Utilize machine learning algorithms (e.g., Random Forest, Support Vector Machines) to train the anomaly detection model.

Real-time Monitoring:

- Continuously monitor database activities and compare them against learned patterns.

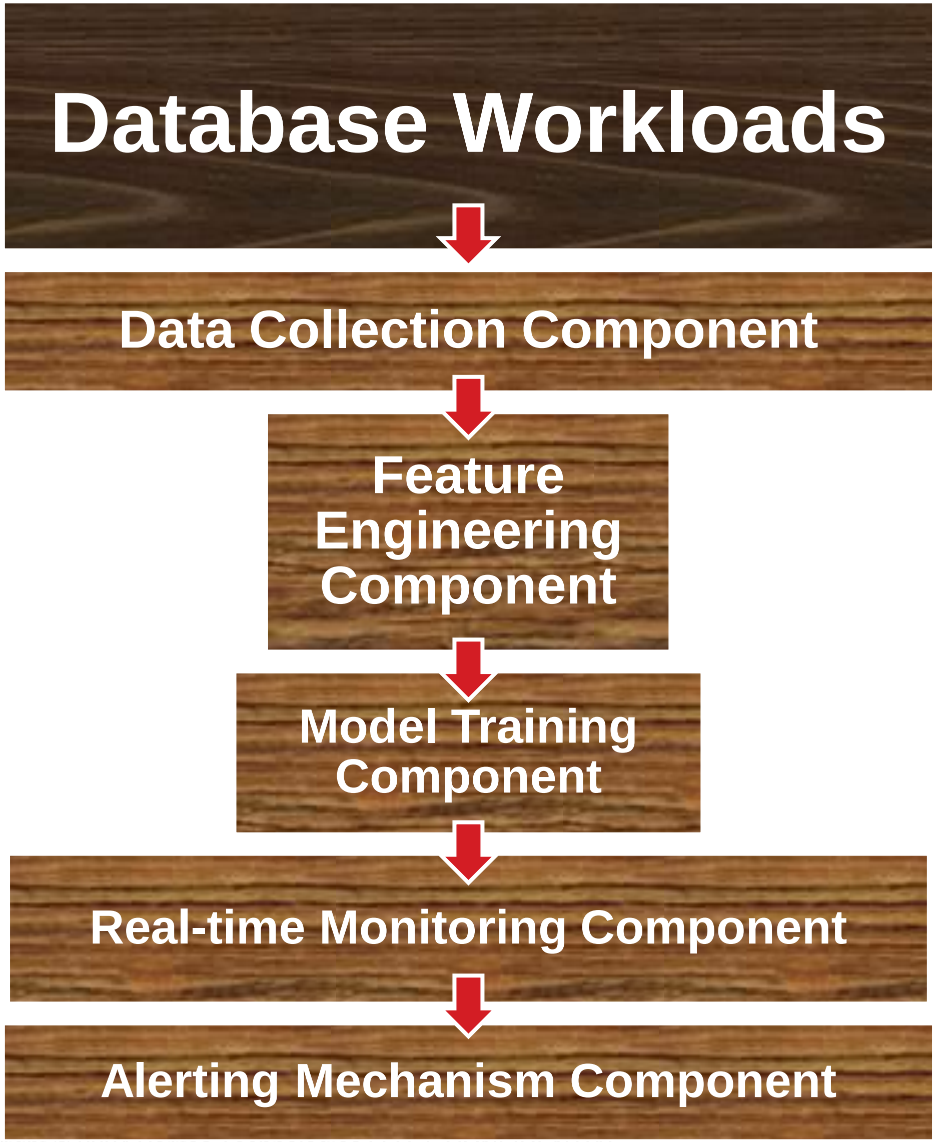
Alerting Mechanism:

- Trigger alerts and notifications for detected anomalies, including severity levels and recommended actions.

Adaptive Learning:

- Incorporate feedback loops to adapt the model to evolving database environments and usage patterns.

Architecture / Design Diagram



Database Workloads

Data Collection Component

Feature Engineering Component

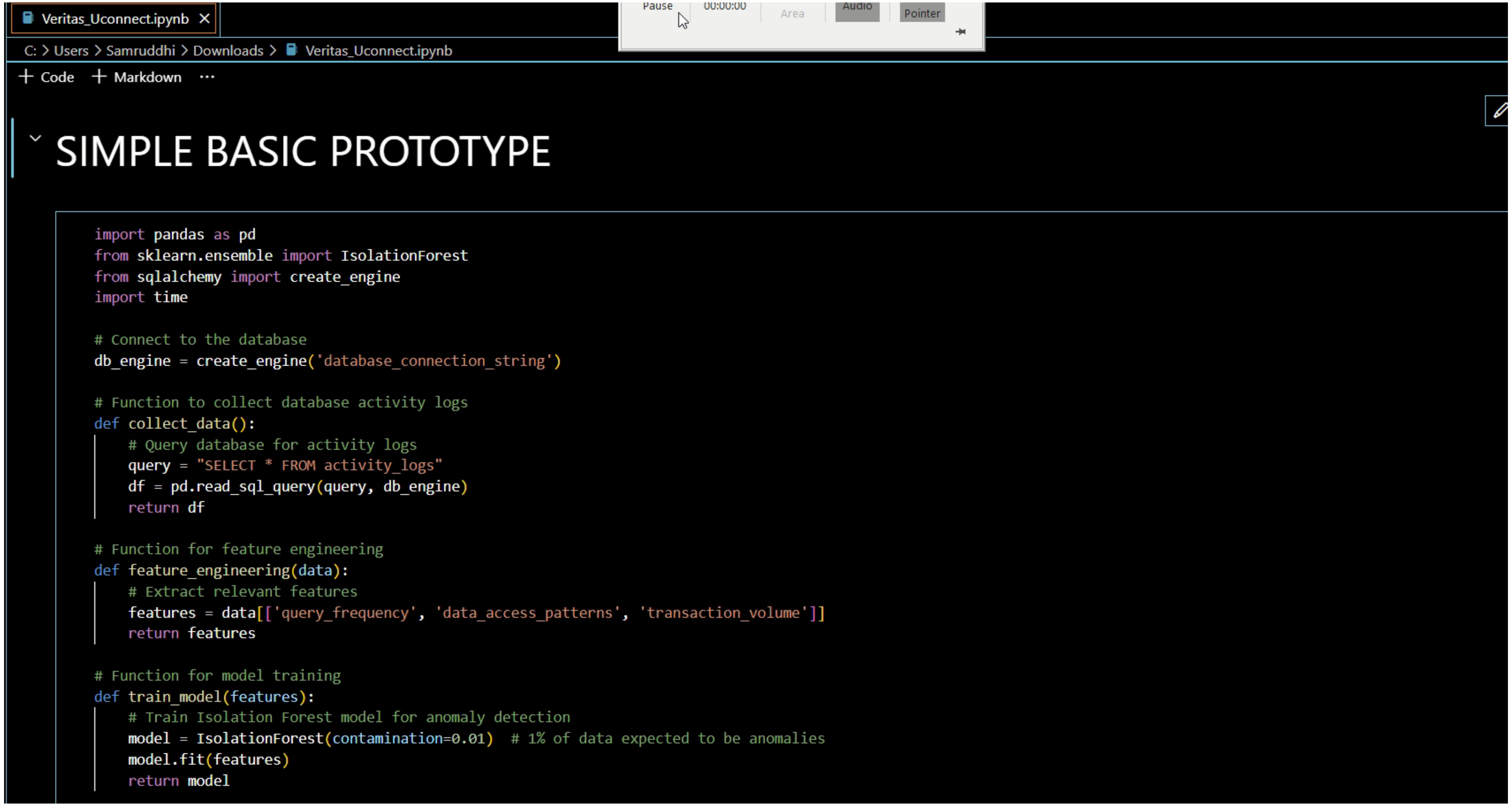
Model Training Component

Real-time Monitoring Component

Alerting Mechanism Component

- **Database Workloads:** Represents various structured workloads like MySQL, PostgreSQL, MongoDB, etc.
- **Data Collection Component:** Responsible for capturing database activity logs including queries, transactions, and system events.
- **Feature Engineering Component:** Extracts relevant features from the collected data such as query frequency, data access patterns, and transaction volume.
- **Model Training Component:** Utilizes machine learning algorithms to train the anomaly detection model based on the extracted features.
- **Real-time Monitoring Component:** Monitors database activities in real-time and compares them against learned patterns.
- **Alerting Mechanism Component:** Triggers alerts and notifications for detected anomalies, indicating severity levels and recommended actions.

Demo – Early Preview of the Project Recording (For Judging as a Reference material)



SIMPLE BASIC PROTOTYPE

```python
import pandas as pd
from sklearn.ensemble import IsolationForest
from sqlalchemy import create_engine
import time

# Connect to the database
db_engine = create_engine('database_connection_string')

# Function to collect database activity logs
def collect_data():
    # Query database for activity logs
    query = "SELECT * FROM activity_logs"
    df = pd.read_sql_query(query, db_engine)
    return df

# Function for feature engineering
def feature_engineering(data):
    # Extract relevant features
    features = data[['query_frequency', 'data_access_patterns', 'transaction_volume']]
    return features

# Function for model training
def train_model(features):
    # Train Isolation Forest model for anomaly detection
    model = IsolationForest(contamination=0.01)  # 1% of data expected to be anomalies
    model.fit(features)
    return model
```

Future Scope

Enhanced Model Capabilities:

- Integrate advanced AI/ML techniques such as deep learning for improved anomaly detection accuracy.

Scalability and Performance Optimization:

- Optimize the system for handling large database sizes and diverse deployment methods.

Integration with Security Frameworks:

- Integrate with existing security frameworks and tools for comprehensive cyber resiliency.

User-friendly Interfaces:

- Develop user-friendly interfaces and dashboards for easy monitoring and management.

Compliance and Audit Support:

- Implement features to facilitate compliance requirements and audit trails for regulatory standards.

Team Photo

Individual participation

Name : Samruddhi Prashant Pate

VERITAS

illuminate 2024

Thank you