



Original Research Article

Intelligent Anomaly Detection in Large-Scale Database Environments: An AI-Enabled Framework for Saudi Digital Infrastructure

Mohammed Afzal Shareef^{1*}

¹Research Scholar, Riyadh, Saudi Arabia

*Corresponding Author

Mohammed Afzal Shareef

Research Scholar, Riyadh, Saudi Arabia

Article History

Received: 27.07.2026

Accepted: 21.09.2026

Published: 22.09.2026

Abstract: Large-scale database environments supporting digital government, financial services, healthcare, smart-city platforms, cloud applications, and national data services generate continuous streams of performance metrics, logs, query statistics, configuration events, and security signals. Static thresholds and manual dashboard inspection are increasingly inadequate because normal behaviour changes with workload cycles, application releases, data growth, infrastructure scaling, and distributed-system dependencies. This paper develops an artificial-intelligence-enabled framework for intelligent anomaly detection in large-scale database environments, with particular attention to Saudi Arabia's rapidly expanding digital infrastructure under Vision 2030. A structured narrative review synthesizes peer-reviewed work on KPI-based database anomaly detection, multivariate time-series modelling, log analytics, deep learning, explainable anomaly detection, AIOps, and database root-cause analysis, together with Saudi policy sources on data and AI, digital government, cybersecurity, and cloud adoption. The proposed framework integrates six layers: governed observability, context-aware data engineering, multi-model anomaly detection, multimodal correlation and root-cause analysis, risk-based alert orchestration, and human-in-the-loop learning. It distinguishes point, contextual, collective, compound, workload, resource, query, availability, configuration, and security-related anomalies, and recommends combining robust statistical baselines with unsupervised machine learning, autoencoders, recurrent or transformer models, and log-sequence analysis rather than relying on a single algorithm. Evaluation is defined using both machine-learning metrics and operational outcomes, including event-level precision and recall, false-alert burden, detection latency, time-to-diagnosis, and stability under concept drift. The framework is aligned with Saudi requirements for responsible AI, resilient digital infrastructure, data protection, and cybersecurity. The paper argues that intelligent anomaly detection should be deployed as an auditable decision-support capability before progressing toward automated remediation, enabling Saudi organizations to improve reliability while preserving governance and expert oversight.

Keywords: Anomaly Detection, Database Systems, Artificial Intelligence, AIOps, Multivariate Time Series, Log Analytics, Root-Cause Analysis, Saudi Vision 2030, Digital Infrastructure, Database Reliability, Explainable AI.

Copyright © 2026 The Author(s): This is an open-access article distributed under the terms of the Creative Commons Attribution 4.0 International License (CC BY-NC 4.0) which permits unrestricted use, distribution, and reproduction in any medium for non-commercial use provided the original author and source are credited.

Citation: Shareef, M. A. (2026). Intelligent Anomaly Detection in Large-Scale Database Environments: An AI-Enabled Framework for Saudi Digital Infrastructure; *Glob Acad J Econ Buss*, 8(5), 1265-1277. <https://doi.org/10.36348/gajeb.2026.v08i05.019>

1. INTRODUCTION

Databases are a foundational layer of digital infrastructure. Government portals, national identity services, payment systems, health platforms, enterprise resource planning, telecommunications, smart-city applications, analytics platforms, and artificial-intelligence services all depend on databases that must remain responsive, correct, secure, and available. The operating challenge is no longer limited to keeping a small number of servers online. Large organizations increasingly operate heterogeneous estates containing relational databases, clustered systems, replicated platforms, cloud-managed services, data warehouses, data lakes, and application-specific data stores. These estates produce high-volume telemetry and interact with application, storage, network, identity, and security layers. A failure that appears as elevated query latency may therefore originate from a changed execution plan, lock contention, a storage bottleneck, a network path, a deployment, a misconfigured parameter, abnormal authentication activity, or an upstream service.

Traditional monitoring is dominated by fixed thresholds, rules, dashboards, and expert interpretation. Such mechanisms remain valuable, but they scale poorly when normal behaviour is seasonal, workload-sensitive, multivariate, and continuously changing. A CPU utilization of 80% may be normal during a known batch window yet abnormal during a quiet period. A modest increase in I/O latency may be harmless on one service but critical when it coincides with rising lock waits, replication delay, and transaction backlog on a high-priority platform. Static rules also generate alert storms because multiple dependent metrics often cross thresholds during a single underlying incident. The result is a mismatch between the volume of observable signals and the human capacity to interpret them in real time.

Artificial intelligence (AI) and machine learning (ML) provide a means to shift monitoring from rigid thresholding toward learned operational context. Research on deep anomaly detection shows that neural models can learn nonlinear representations of normal and abnormal behaviour in complex data, while unsupervised approaches are particularly relevant when labelled failure data are scarce (Pang *et al.*, 2021). Multivariate methods such as USAD, OmniAnomaly, and TranAD demonstrate approaches for modelling temporal dependencies and correlated signals in high-dimensional telemetry (Audibert *et al.*, 2020; Su *et al.*, 2019; Tuli *et al.*, 2022). Database-specific research is also maturing. Gao *et al.*, (2025) synthesize KPI-based performance anomaly detection in databases and emphasize the importance of database context, deployment

conditions, and root-cause diagnosis. DBPA provides a benchmark for reproducible transactional database performance anomalies, while PinSQL demonstrates how anomaly detection can be combined with query-level diagnosis in a cloud database environment (Huang *et al.*, 2023; Liu *et al.*, 2022).

For Saudi Arabia, this topic is strategically relevant. The National Strategy for Data and AI aims to position the Kingdom among leading data-driven economies, while SDAIA's strategic objectives include advanced digital infrastructure, resilient cybersecurity, government cloud services, national platforms, and AI research and adoption (SDAIA, n.d.-a; SDAIA, n.d.-b). Vision 2030 reporting also identifies digital infrastructure, cloud, AI, cybersecurity, and innovation as enablers of the digital and knowledge economy (Saudi Vision 2030, 2026). As digital public services and private-sector platforms scale, the reliability of the underlying database layer becomes a practical requirement for service quality, business continuity, trust, and national digital competitiveness.

This paper develops a conceptual framework for intelligent anomaly detection in large-scale database environments that can be adapted to Saudi digital infrastructure. It addresses four questions: (1) which anomaly types and data sources are most relevant to large database estates; (2) which AI and ML techniques are suitable under real operational constraints; (3) how should detection be connected to diagnosis, prioritization, and human decision-making; and (4) how can the framework align with Saudi digital, AI, data, and cybersecurity governance. The contribution is not a new benchmark result or production deployment. Instead, it is a structured synthesis that converts recent anomaly-detection and AI-for-database research into a deployment-oriented architecture with evaluation criteria, governance controls, and an implementation roadmap.

2. Research Background and Literature Context

2.1 Anomaly Detection in Database Operations

An anomaly is an observation or pattern that deviates sufficiently from expected behaviour to warrant investigation. In database operations, that definition must be contextualized because not every deviation is harmful and not every incident produces a simple outlier. Database telemetry contains natural variability caused by users, batch processing, maintenance, backups, statistics collection, business calendars, software changes, elastic scaling, and data growth. The operational task is therefore to distinguish legitimate variation from behaviour associated with degradation, failure, abuse, or an emerging risk.

A useful taxonomy begins with point, contextual, and collective anomalies. A point anomaly is an isolated observation that is unusual relative to an expected distribution, such as a sudden spike in physical reads. A contextual anomaly is abnormal only within a particular time, workload, or service context; for example, a response time that is acceptable during a known peak may be suspicious

during a low-load window. A collective anomaly emerges from a sequence or group of observations that may appear individually normal but form an abnormal pattern together. Database incidents often fall into the latter two categories because degradation develops through correlated trends rather than a single extreme measurement (Gao *et al.*, 2025; Schmidl *et al.*, 2022).

Table 1: Operational taxonomy of anomalies in large-scale database environments

Anomaly class	Example database signals	Typical operational interpretation	Suitable detection approach
Point	Single CPU, I/O, latency or session spike	Isolated deviation; may be transient or severe	Robust threshold, change-point, outlier model
Contextual	Latency elevated only for a specific workload/time	Value is abnormal relative to business or workload context	Seasonal baseline, context-aware model
Collective	Gradual lock waits + latency + backlog	Abnormal temporal pattern across a window	Multivariate sequence model, forecasting
Compound	CPU + I/O + lock or multi-node symptoms	Concurrent or causally linked anomalies	Ensemble + topology correlation
Query / plan	Execution time, reads, plan hash, frequency shift	Regression, new SQL, cardinality or plan change	SQL fingerprint model + RCA
Availability / replication	Node state, lag, failover, backup failures	Resilience or recovery risk	Rules + temporal correlation
Configuration / change	Parameter, schema, patch, deployment event	Operational drift or change-induced degradation	Change-aware correlation
Security-related	Login, privilege, audit or access pattern changes	Potential abuse, compromise or policy violation	Behavioural anomaly + security rules

Source: author synthesis based on Gao *et al.* (2025), Schmidl *et al.* (2022), DBPA and related reviewed literature.

2.2 KPI-Based and Multivariate Time-Series Detection

Database monitoring systems produce hundreds or thousands of key performance indicators (KPIs). Examples include transactions per second, queries per second, average and percentile response time, CPU, memory, buffer-cache ratios, physical and logical reads, I/O latency, redo or transaction-log generation, active sessions, waits, lock duration, cache eviction, replication delay, storage utilization, and network throughput. The core technical challenge is to learn what combination and temporal pattern of these metrics represents normal operation for a particular service.

Unsupervised machine learning is attractive because labelled incidents are scarce and biased toward previously known failure modes. Isolation-based methods, clustering, one-class classifiers, and density approaches can model normal regions and flag outliers. Deep representation learning extends this idea. Autoencoders learn to reconstruct normal multivariate behaviour and treat high reconstruction error as anomalous. USAD combines autoencoder-based reconstruction with adversarial training for multivariate time-series anomaly detection (Audibert *et al.*, 2020). OmniAnomaly uses a stochastic recurrent neural-network architecture to model temporal dependencies and reconstruction

probabilities in multivariate telemetry (Su *et al.*, 2019). TranAD applies transformer-based attention and self-conditioning to capture wider temporal context and diagnose anomalous dimensions (Tuli *et al.*, 2022). These methods are relevant to database estates because many incidents are preceded by gradual cross-metric changes rather than isolated threshold breaches.

2.3 Log and Event-Sequence Anomaly Detection

Metrics describe how a system behaves numerically, but logs describe events, errors, state transitions, and operational context. Database alert logs, audit logs, listener or connectivity logs, cluster logs, operating-system logs, backup logs, and application logs can reveal symptoms that do not appear clearly in KPIs. Log data are difficult because raw messages are semi-structured, high-volume, and change across software versions.

DeepLog introduced the idea of modelling log sequences with long short-term memory networks so that deviations from learned normal event sequences can be detected (Du *et al.*, 2017). Drain provides an efficient online parser that converts raw log messages into templates, enabling scalable downstream analytics (He *et al.*, 2017). LogAnomaly adds semantic representation of log templates and detects both sequential and

quantitative anomalies (Meng *et al.*, 2019). LogBERT applies a transformer language-model approach to learn patterns in log sequences and identify anomalous events (Guo *et al.*, 2021). These approaches suggest that database anomaly detection should not be limited to numerical KPIs. A strong architecture should fuse time-series signals with parsed logs, change records, query fingerprints, and topology information.

2.4 Database-Specific Diagnosis and AI for DBMS

The database community has increasingly explored the broader interaction between AI and database systems. Zhou *et al.* (2022) describe AI for databases as a research direction encompassing learned optimization, tuning, indexing, monitoring, and other data-management functions. Dritsas and Trigka (2026) similarly review ML integration into modern database systems and identify anomaly detection, self-tuning, query planning, and operational deployment as major areas of development.

Detection alone is insufficient for operations. A system that announces "anomaly detected" without identifying the likely cause can increase workload rather than reduce it. PinSQL is important because it links anomaly detection to the identification of root-cause SQL statements in cloud databases and then connects diagnosis to repair actions (Liu *et al.*, 2022). DBPA is also significant because it formalizes reproducible performance-anomaly scenarios and provides data for studying diagnosis in transactional database systems (Huang *et al.*, 2023). Gao *et al.*, (2025) emphasize that root-cause analysis remains a central research challenge because generic time-series methods often identify abnormal signals without explaining why the anomaly occurred.

Explainability is therefore operationally important. Li *et al.*, (2024) review explainable anomaly detection and note the gap between high-performing models and human-understandable reasons. In database management, Ouared *et al.*, (2023) argue for maintaining the DBA in the loop when ML is used for tuning, because acceptance of automated decisions depends on transparent reasoning. For anomaly detection, an explanation should identify the most influential signals, the time at which they diverged, related changes or SQL, affected dependencies, model confidence, and evidence that supports the recommendation.

2.5 AIOps, Drift, and Production Constraints

AIOps research places anomaly detection inside a larger failure-management lifecycle that includes detection, prediction, diagnosis, and

remediation. Notaro *et al.*, (2021) show that AIOps methods address the scale and complexity of modern operational systems, while Diaz-De-Arcaya *et al.*, (2024) highlight the importance of lifecycle practices, monitoring, model maintenance, and integration between MLOps and AIOps. These perspectives are especially relevant to databases because an anomaly model that performs well during a pilot can degrade after infrastructure changes, workload evolution, software upgrades, or new application features.

3. METHODOLOGY

This study uses a structured narrative review combined with conceptual framework design. It does not claim a PRISMA systematic review or a statistical meta-analysis. The objective is to integrate representative peer-reviewed research on anomaly detection, AIOps, and AI-enabled database operations with official Saudi policy and governance documents, and to convert that evidence into a coherent architecture for large-scale database environments.

The literature selection prioritized five evidence groups. The first group covers database-specific anomaly detection, performance diagnosis, and AI-for-DBMS research. The second covers multivariate time-series anomaly detection and benchmarking. The third covers log parsing and log-sequence anomaly detection. The fourth covers explainable anomaly detection and AIOps lifecycle management. The fifth consists of Saudi official sources concerning Vision 2030, the National Strategy for Data and AI, responsible AI, digital government, cybersecurity, data governance, and cloud transformation. Sources were selected for relevance to operational database monitoring, recency where appropriate, methodological influence, and direct applicability to Saudi digital infrastructure.

The conceptual synthesis followed five steps. First, database anomaly types were mapped to observable data sources and potential detection methods. Second, the strengths and limitations of statistical, classical ML, deep learning, sequence, and multimodal methods were compared. Third, the detection problem was connected to root-cause analysis, prioritization, and human operations workflows. Fourth, lifecycle risks such as drift, explainability, privacy, data quality, and unsafe automation were incorporated. Fifth, the architecture was aligned with Saudi digital and cybersecurity governance. The resulting framework should be interpreted as a deployment blueprint to be validated empirically in a target organization rather than as a claim of universal model superiority.

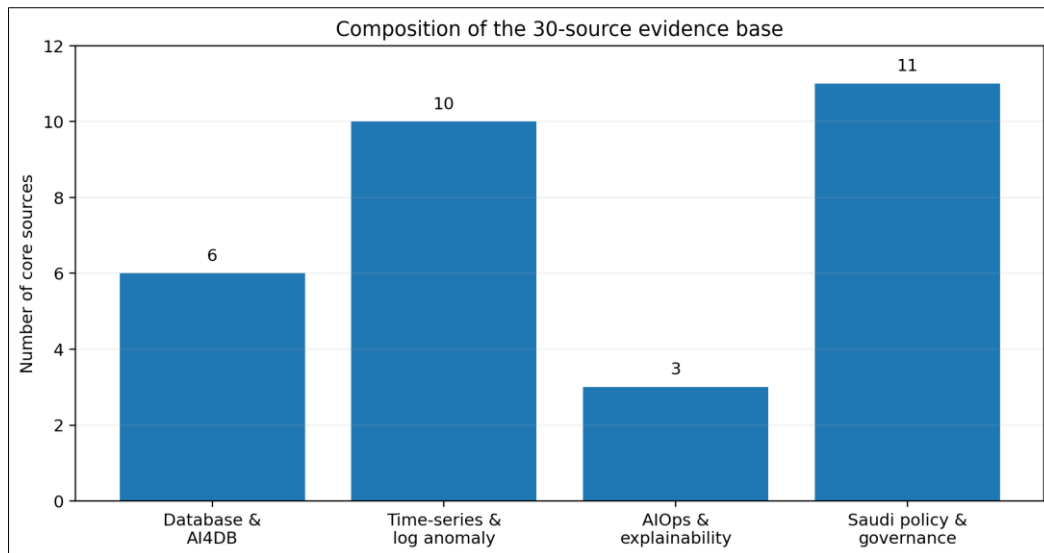


Figure 1: Composition of the review evidence base (n = 30). Source: author synthesis

4. Proposed AI-Enabled Anomaly Detection Framework

The proposed framework contains six interacting layers: governed observability, context-aware data engineering, multi-model anomaly

detection, multimodal correlation and root-cause analysis, risk-based alert orchestration, and human-in-the-loop lifecycle learning. The design assumes heterogeneous database technologies and hybrid infrastructure rather than a single database product.

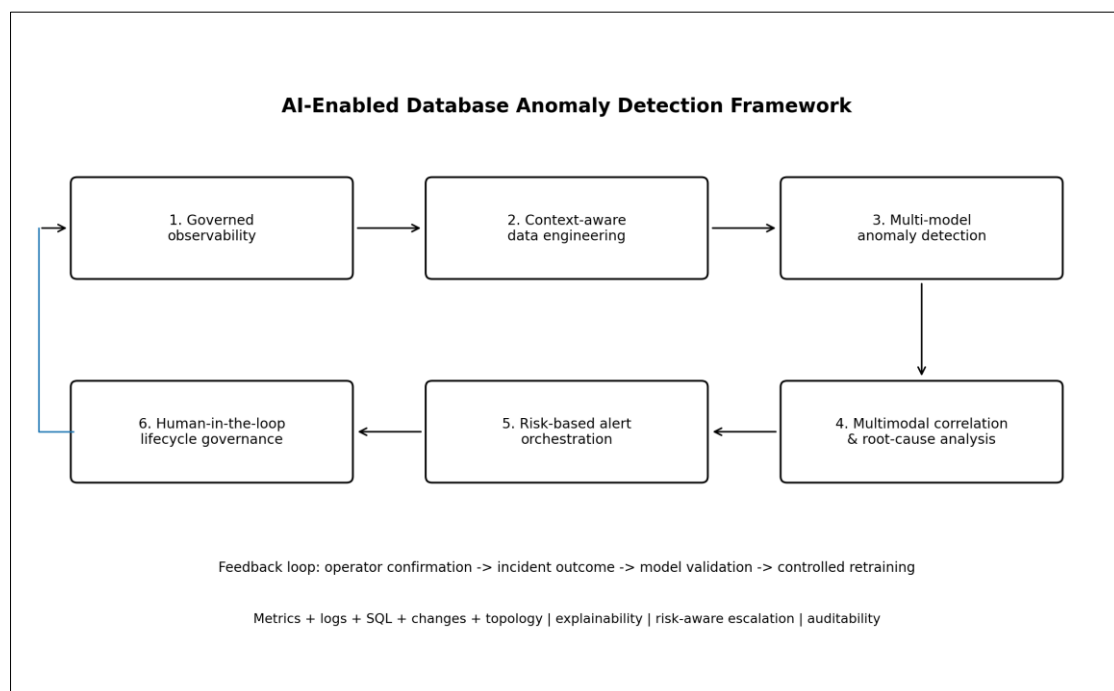


Figure 2: Six-layer AI-enabled anomaly detection framework for Saudi digital infrastructure
Source: author conceptualization based on reviewed literature and Saudi governance requirements

4.1 Layer 1: Governed Observability

The first layer creates a reliable operational data foundation. It collects database KPIs, wait events, query statistics, execution-plan identifiers, lock information, session state, replication status, backup events, cluster state, operating-system and storage metrics, cloud monitoring signals, logs, audit records, configuration snapshots, deployment events,

incident records, and service topology. Collection must be timestamp-synchronized so that changes across layers can be correlated.

A key principle is that telemetry should be semantically normalized. Oracle, PostgreSQL, Microsoft SQL Server, MySQL, and cloud-managed databases expose different native metrics, but many

can be mapped into common concepts such as transaction pressure, read I/O stress, write I/O stress, concurrency pressure, memory pressure, availability state, replication health, and query efficiency. Product-specific detail should be retained for diagnosis, while normalized features support cross-platform models and enterprise-wide dashboards.

Governance begins at ingestion. Sensitive SQL text, audit events, identifiers, and operational logs may contain personal, confidential, or security-relevant data. Access control, encryption, retention, masking or tokenization, and purpose limitation must be defined before telemetry is used for model training. Saudi Data Cybersecurity Controls emphasize protection across the data lifecycle, while SDAIA's data-governance and AI-ethics direction supports responsible processing and oversight (NCA, 2022; SDAIA, 2023).

4.2 Layer 2: Context-Aware Data Engineering

Raw telemetry must be transformed into features that describe behaviour rather than isolated measurements. Useful transformations include rolling means, percentiles, robust dispersion, rate of change, lag features, seasonal residuals, workload-normalized ratios, wait-class percentages, per-execution SQL cost, cache efficiency, lock-wait intensity, replication-lag acceleration, and correlations between resource and query signals. The objective is to represent both absolute state and change dynamics.

Context is essential to reduce false positives. Each observation should be enriched with service identity, environment, criticality, database role, business calendar, maintenance windows, deployment status, batch schedules, backup

windows, expected growth, and known capacity changes. A model that ignores planned maintenance will incorrectly classify many legitimate events as failures. Conversely, the same metric level may deserve different treatment for a low-priority development database and a national service with strict availability requirements.

4.3 Layer 3: Multi-Model Anomaly Detection Engine

No single algorithm is optimal for all database anomalies. The framework therefore uses an ensemble of model families selected according to signal type and operational purpose. Transparent statistical baselines are retained for stable metrics and hard safety limits. Isolation or density methods can identify multivariate outliers with relatively low modelling complexity. Autoencoders can capture nonlinear normal patterns and use reconstruction error as an anomaly score. Recurrent or transformer models can learn temporal dependencies and forecast expected behaviour. Log-sequence models can identify abnormal event patterns. Supervised classifiers can be used for recurring incident types when sufficiently reliable labels exist.

The ensemble should produce calibrated evidence rather than a binary alarm from each model. For a signal vector x at time t , a simplified composite anomaly score may be written as:

$$A(t) = w_1 S_{\text{stat}}(t) + w_2 S_{\text{outlier}}(t) + w_3 S_{\text{recon}}(t) + w_4 S_{\text{forecast}}(t) + w_5 S_{\text{log}}(t),$$

Where each S term is a normalized score from a model family and the weights w are determined by validation, service context, and data quality. The formula is conceptual: production systems should calibrate scores using held-out operational periods and incident data rather than assigning arbitrary weights.

Table 2: Comparison of AI and machine-learning techniques for database anomaly detection

Technique	Best suited data	Main strength	Main limitation / control
Robust statistical baseline	Stable univariate KPIs	Transparent, fast, inexpensive	Limited nonlinear/context modelling
Isolation / density ML	Multivariate KPI snapshots	Works with scarce labels	Scaling and drift sensitivity
Autoencoder / reconstruction	High-dimensional telemetry	Learns nonlinear normal behaviour	Thresholding and explainability
LSTM / recurrent models	Ordered KPI or event sequences	Captures temporal dependence	Training cost; long-range limits
Transformer models	Long multivariate sequences / logs	Broader temporal context and attention	Compute, tuning and explainability
Log-sequence models	Parsed logs and event templates	Finds abnormal event order/semantics	Parser/template drift
Supervised classifier	Well-labelled recurring incidents	Direct incident-type prediction	Label scarcity and class imbalance
Graph / multimodal RCA	Topology + metrics + SQL + changes	Dependency-aware diagnosis	Graph quality and causal ambiguity

Source: author synthesis from Pang et al. (2021), Audibert et al. (2020), Tuli et al. (2022), Guo et al. (2021), Gao et al., (2025) and related sources.

4.4 Layer 4: Multimodal Correlation and Root-Cause Analysis

The fourth layer turns detection into diagnosis. When an anomaly is detected, the system creates an incident window and gathers the changes that occurred before and during that period. It correlates affected metrics with SQL, execution plans, configuration changes, deployments, logs, topology, and dependent services. Temporal order is important: a storage-latency rise preceding database waits has a different interpretation from storage latency increasing after a query storm.

A dependency graph can represent relationships among services, database instances, schemas, queries, hosts, storage volumes, network endpoints, and applications. Root-cause ranking can then combine temporal precedence, anomaly magnitude, graph proximity, known failure patterns, and change events. Query-level diagnosis should distinguish a genuinely causal SQL regression from a query that merely consumed resources after another component degraded. PinSQL illustrates the value of tracing root-cause SQL rather than stopping at the identification of abnormal KPIs (Liu *et al.*, 2022).

Explanations should be concise and evidence based. A useful output is not "model score 0.91" but a statement such as: "High-confidence performance anomaly affecting service X; first divergence detected in read latency at 10:32, followed by buffer-cache misses and query latency; 68% of incremental I/O is associated with SQL fingerprint Q17 introduced after deployment D4; no corresponding CPU anomaly; recommended investigation: plan regression and storage read path." The numerical values in such an explanation would come from the live environment, not from a generic model.

4.5 Layer 5: Risk-Based Alert Orchestration

Anomaly severity should reflect more than model confidence. The framework proposes a risk score combining detection confidence, business criticality, expected service impact, persistence, blast radius, and evidence of an active change or security concern. A conceptual formulation is:

$$R = f(A, C, I, P, B, G),$$

Where A is anomaly confidence, C is service criticality, I is predicted impact, P is persistence, B is blast radius, and G is governance or security sensitivity. The function should be defined by organizational policy and validated through incident history.

The orchestration layer groups correlated anomalies into a single incident, suppresses derivative alerts, and routes the event according to risk. Low-confidence anomalies can be recorded for learning without paging an operator. Medium-risk

anomalies can create a ticket and attach diagnostic evidence. High-risk anomalies can trigger immediate notification, enhanced telemetry collection, and a predefined runbook. Automated remediation should be introduced only after detection and diagnosis have demonstrated sufficient reliability and should initially be limited to reversible, low-risk actions.

4.6 Layer 6: Human-In-The-Loop Learning and Model Governance

DBAs, site-reliability engineers, security teams, and application owners remain part of the control loop. Operators should be able to confirm an anomaly, reject it as expected behaviour, change the root-cause category, identify a missed cause, and record the actual resolution. This feedback improves future models and allows false-alert patterns to be discovered.

Model governance includes versioning, validation datasets, approval records, performance monitoring, drift indicators, retraining triggers, rollback procedures, and audit trails. Explainability should be treated as a control requirement rather than an optional visualization. SDAIA's AI Adoption Framework and National AI Risk Management Framework emphasize governance, reliability, risk identification, treatment, and ongoing monitoring, which supports a lifecycle approach for operational AI (SDAIA, 2024; SDAIA, 2026a).

5. Detection Strategy for Large-Scale Database Environments

5.1 Statistical and Rule-Based Layer

Statistical methods should not be discarded when advanced AI is introduced. They provide low-cost, explainable baselines and enforce non-negotiable limits such as storage exhaustion, failed backup status, or replication states that require immediate attention. Robust statistics are particularly useful when distributions contain occasional spikes. Seasonal baselines can represent known hourly and weekly workload patterns. Change-point methods can identify structural shifts after deployments or configuration changes.

5.2 Unsupervised and Semi-Supervised Learning

Unsupervised learning should be the default starting point for rare and poorly labelled incidents. Isolation methods can detect unusual combinations of metrics; clustering can identify operating regimes; one-class models can learn boundaries around healthy periods; and autoencoders can model high-dimensional normal patterns. Semi-supervised learning can improve performance when a limited set of trustworthy incident labels exists.

5.3 Deep Temporal Models

Recurrent networks and transformers are useful when sequence and long-range context matter. They can learn that a gradual increase in sessions followed by lock waits and rising latency represents a different state from a short independent spike. Transformer attention can capture relationships across longer windows, while recurrent architectures may be more economical for some streaming deployments. Model selection should consider inference latency, training cost, memory, explainability, and data volume, not only F1-score.

5.4 Log and Change-Event Models

Logs provide discrete signals that complement metrics. A log pipeline should parse messages into stable templates, normalize variable fields, preserve important parameters, and detect new or unusual sequences. DeepLog, LogAnomaly, and LogBERT demonstrate different ways to model event sequences and semantics (Du *et al.*, 2017; Meng *et al.*, 2019; Guo *et al.*, 2021). In a database context, the log model should be correlated with patch, deployment, failover, backup, configuration, and security events so that a new message pattern can be interpreted against operational changes.

5.5 Cross-Database and Distributed Anomalies

Large Saudi enterprises may operate many database nodes and multiple geographic or cloud environments. A single-node model can miss anomalies that are visible only through comparison. Peer-group baselines can compare replicas, cluster nodes, or similar services. Distributed anomalies include uneven load, network partitions, replication divergence, region-specific latency, or failures that propagate through shared storage or identity services.

Compound anomalies are particularly difficult because several faults or symptoms can occur simultaneously. Gao *et al.*, (2025) identify compound and distributed scenarios as important future directions for database anomaly detection. Graph-based correlation and topology-aware models are promising because they can represent dependencies rather than treating every metric as independent.

6. Evaluation and Validation

Machine-learning metrics are necessary but not sufficient for operational anomaly detection. Point-wise precision and recall can be misleading when an incident lasts many minutes and produces

hundreds of anomalous timestamps. Evaluation should therefore include event-level metrics that assess whether the system detected the incident, how early it detected it, how many duplicate alerts were generated, and whether the diagnosis was useful.

Precision measures the proportion of detected events that were truly anomalous. Recall measures the proportion of known incidents that were detected. F1-score balances precision and recall, but it should be supplemented by precision-recall curves when anomalies are rare. False-positive rate must be translated into operator burden, such as false alerts per database per week. Detection latency measures the time between the earliest meaningful anomaly and alert generation. Time-to-diagnosis measures how quickly the system identifies a plausible root cause. Root-cause top-k accuracy evaluates whether the actual cause appears among the highest-ranked explanations.

Operational metrics should include mean time to acknowledge, mean time to diagnose, mean time to recover, number of alerts per incident, percentage of incidents with useful early warning, and percentage of model recommendations accepted by experts. Stability under concept drift should be tested across upgrades, workload shifts, seasonal periods, and infrastructure changes. Resource overhead is also important: telemetry collection and model inference must not degrade the database systems being protected.

Evaluation datasets should combine three sources. First, historical incidents provide realistic failure patterns but may contain incomplete labels. Second, controlled fault injection or replay can reproduce known issues such as lock contention, resource saturation, plan regression, or replication delay. DBPA demonstrates the value of reproducible anomaly scenarios for transactional databases (Huang *et al.*, 2023). Third, extended normal-operation periods are necessary to measure false-alert rates under real workload variability.

A staged validation design is preferable. Models should first run in shadow mode without paging operators. Their detections are compared with incident records and expert review. The next stage enables advisory alerts but no automated actions. Only after stable performance is demonstrated should selected low-risk responses be automated. This progression reduces the risk that an inaccurate model creates operational disruption.

Table 3: Recommended evaluation framework for a production anomaly-detection pilot

Evaluation dimension	Recommended metric	Operational meaning
Detection quality	Precision, recall, F1, AUC-PR	Balance missed incidents and false detections
Event quality	Event-level precision / recall	Counts incidents rather than anomalous timestamps
Alert burden	False alerts per database per week	Measures operator fatigue directly
Timeliness	Detection latency / early-warning lead time	Shows whether intervention is possible
Diagnosis	Top-k root-cause accuracy; time-to-diagnosis	Measures usefulness after detection
Operational outcome	MTTA, MTTR, duplicate alerts per incident	Connects models to service reliability
Robustness	Performance across drift / upgrades / seasons	Tests continued validity
Efficiency	Collection overhead, inference latency, compute cost	Ensures monitoring does not harm production

Source: author synthesis; benchmarking principles informed by Schmidl et al. (2022), Wenig et al. (2022), Gao et al., (2025) and DBPA.

7. Application to Saudi Digital Infrastructure

7.1 Government and Citizen Digital Services

Saudi digital-government platforms depend on reliable data services and integrated infrastructure. The Digital Government Authority's standards link digital-transformation planning with Vision 2030 and emphasize coordinated architecture and measurable performance (DGA, 2024). Intelligent database anomaly detection can support these objectives by identifying degradation before it affects beneficiaries, correlating incidents across application and database layers, and reducing the dependence on manual monitoring for large estates.

For citizen-facing services, the most valuable use case is early warning rather than autonomous repair. A model can detect rising transaction latency, replication stress, abnormal connection patterns, or query regressions before service-level thresholds are breached. The orchestration layer can prioritize issues by service criticality and attach a diagnosis package for the responsible team.

7.2 Cloud and Hybrid Data Platforms

Saudi government and private-sector organizations are increasing cloud adoption. DGA guidance identifies cloud computing as an enabler of efficiency, integration, digital services, and secure data processing, while the National Cybersecurity Authority has updated cloud cybersecurity controls for cloud providers and tenants (DGA, 2025; NCA, 2025b). Cloud databases introduce elasticity and managed-service abstraction but also create new observability challenges because infrastructure control and telemetry differ across providers.

The proposed framework supports hybrid environments by separating semantic operational features from product-specific metrics. The model can learn common signals such as transaction pressure and replication health while retaining

provider-specific details for root-cause analysis. Cloud scaling events should be treated as context so that legitimate elasticity is not mistaken for an anomaly.

7.3 Financial, Healthcare, Industrial, and Smart-City Platforms

Financial databases require low latency, transaction integrity, auditability, and strong control over automated actions. Healthcare platforms require availability and data protection. Industrial and smart-city environments may integrate operational technology, sensors, geographic platforms, and real-time analytics. The same anomaly-detection architecture can be adapted by changing feature sets, risk policies, retention, and response thresholds.

8. Governance, Security, and Responsible AI

Saudi Arabia's policy environment makes governance a core component of operational AI. SDAIA's National Strategy for Data and AI positions data and AI as national economic and social assets. Its AI Ethics Principles emphasize responsible use, privacy, governance, and mitigation of negative impacts, while the AI Adoption Framework and National AI Risk Management Framework provide lifecycle-oriented guidance for adoption and risk treatment (SDAIA, 2023; SDAIA, 2024; SDAIA, 2026a). Anomaly detection used in critical infrastructure should therefore be auditable and risk-aware.

Data governance must cover collection, processing, storage, access, retention, and deletion of operational telemetry. SQL text and audit data can contain personal or commercially sensitive information. Training datasets should use the minimum information necessary, and sensitive fields should be masked or tokenized where detailed values

are unnecessary. Access to raw telemetry and model explanations should follow least-privilege principles.

Cybersecurity requirements also intersect with anomaly detection. The NCA Essential Cybersecurity Controls aim to protect national entities' information and technology assets, while the Data Cybersecurity Controls address protection across the data lifecycle and the Cloud Cybersecurity Controls set minimum requirements for cloud services (NCA, 2025a; NCA, 2022; NCA, 2025b). Intelligent anomaly detection can strengthen these objectives by identifying unusual system behaviour, but the AI platform itself becomes part of the attack surface. Model services, feature stores, telemetry pipelines, APIs, and training data require security controls, monitoring, and integrity protection.

Responsible AI also means avoiding false certainty. An anomaly score is evidence, not proof of root cause. Operators should see confidence, contributing signals, model version, data-quality status, and known limitations. High-impact decisions should require human review until the system has demonstrated appropriate reliability. All automated actions should be logged with preconditions, rationale, approval path, execution result, and rollback information.

9. Implementation Roadmap

A practical implementation can be organized into four phases. Phase 1 establishes observability and baseline quality. The organization inventories databases, services, criticality, monitoring coverage, log sources, incident records, and data-governance constraints. Fixed rules are rationalized, telemetry is normalized, and missing monitoring gaps are addressed.

Phase 2 introduces AI-assisted detection in shadow mode. Statistical baselines, an unsupervised multivariate model, and a log-anomaly model can be deployed for a limited set of representative databases. Detections are evaluated against historical incidents and expert review. The objective is to quantify false-alert burden, detection latency, and data-quality limitations before operational paging is enabled.

Phase 3 adds multimodal correlation, root-cause ranking, and advisory alerting. Models are connected to SQL fingerprints, changes, topology, and incident workflows. Explanations and recommended investigation steps are shown to DBAs. Human feedback is captured systematically. Drift monitoring, retraining governance, and model rollback procedures become mandatory.

Phase 4 enables selective closed-loop actions. Only reversible, low-risk operations with strong evidence should be automated initially, such as increasing telemetry, collecting diagnostics, creating tickets, routing incidents, or executing preapproved validation queries. Higher-risk database changes should remain approval-based unless rigorous production evidence supports further autonomy.

Success should be measured as an operational transformation, not merely a model-development exercise. A technically accurate detector that operators do not trust or that creates too many alerts is unsuccessful. The target outcomes are fewer noisy alerts, earlier detection, faster diagnosis, improved service reliability, and better use of expert time.

Table 4: Phased implementation roadmap for intelligent database anomaly detection

Phase	Primary objective	Key deliverables	Exit criterion
1. Foundation	Trusted observability and governance	Inventory, normalized telemetry, retention/access rules, baseline alert cleanup	Critical services have reliable data and known ownership
2. Shadow AI	Validate models without operational impact	Statistical + unsupervised + log models; back testing; false-alert analysis	Stable detection quality and acceptable alert burden
3. Advisory AIOps	Connect detection to diagnosis and workflow	Topology, SQL/change correlation, RCA ranking, explainable tickets, feedback capture	Operators confirm measurable diagnostic value
4. Selective automation	Automate low-risk reversible responses	Preapproved runbooks, action audit, rollback, post-action verification	Automation meets safety and reliability thresholds

Source: author conceptualization based on AIOps lifecycle research and Saudi AI governance guidance.

10. Limitations and Future Research

This paper is conceptual and does not present an empirical deployment in a Saudi production environment. The proposed architecture must therefore be validated using organization-

specific telemetry, workload patterns, incident histories, and governance requirements. Performance results from public anomaly datasets cannot be assumed to transfer directly to enterprise databases.

A second limitation is the scarcity of high-quality labelled database incidents. Incident tickets are often incomplete, root causes may be ambiguous, and "normal" periods may contain hidden degradations. Future research should develop privacy-preserving benchmark datasets, reproducible fault-injection environments, and cross-platform anomaly taxonomies that reflect Oracle, PostgreSQL, SQL Server, cloud databases, and distributed data platforms.

Third, root-cause analysis remains harder than detection. Temporal correlation does not prove causality, and several symptoms may share an upstream cause. Future work should evaluate graph neural networks, causal discovery, knowledge graphs, and retrieval-augmented diagnostic models while maintaining strict evidence controls. Generative AI may help summarize incidents and retrieve runbooks, but it should not be allowed to fabricate causes or execute changes without verification.

Fourth, concept drift and operational change require continued study. Models should learn from legitimate changes without normalizing emerging failures. Adaptive thresholds, online learning, drift detectors, and change-aware retraining policies need evaluation under realistic database lifecycles. Finally, research is needed on Arabic-language operational interfaces, national-sector datasets, and Saudi-specific governance patterns so that intelligent operations can support local capability development as well as technical reliability.

11. CONCLUSION

Large-scale database environments are becoming too dynamic and data-intensive for monitoring strategies based only on static thresholds and manual interpretation. AI-enabled anomaly detection can improve operational awareness by learning normal behaviour, identifying multivariate and sequential deviations, correlating metrics with logs and changes, and prioritizing incidents according to service risk. The literature shows that statistical baselines, unsupervised learning, deep temporal models, log analytics, and database-specific diagnosis each contribute useful capabilities, but no single model solves the full operational problem.

The framework proposed in this paper integrates six layers: governed observability, context-aware feature engineering, multi-model detection, multimodal root-cause analysis, risk-based orchestration, and human-in-the-loop lifecycle governance. Its central principle is that anomaly detection should produce actionable and explainable evidence rather than isolated scores. Machine-learning metrics must be supplemented by

operational measures such as false-alert burden, detection lead time, diagnosis speed, and expert acceptance.

For Saudi Arabia, intelligent database anomaly detection aligns with Vision 2030 priorities around digital and knowledge infrastructure, data and AI leadership, resilient cloud services, cybersecurity, and efficient digital government. The appropriate adoption path is progressive: establish trusted telemetry, validate models in shadow mode, introduce advisory diagnosis, and automate only low-risk actions after evidence of reliability. This approach allows AI to strengthen the resilience of Saudi digital infrastructure while preserving accountability, cybersecurity, data governance, and expert control.

DECLARATIONS

Funding: No external funding was received for this conceptual review.

Conflict of Interest: The author declares no known competing financial interests or personal relationships that could have influenced this work.

Data Availability: No original empirical dataset was generated. The study synthesizes published literature and publicly available Saudi policy and governance documents.

Author Contribution: Conceptualization, literature synthesis, framework design, writing, review and visualization: Mohammed Afzal Shareef.

REFERENCES

- Audibert, J., Michiardi, P., Guyard, F., Marti, S. and Zuluaga, M.A. (2020) 'USAD: UnSupervised Anomaly Detection on Multivariate Time Series', Proceedings of the 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, pp. 3395–3404. <https://doi.org/10.1145/3394486.3403392>.
- Diaz-De-Arcaya, J., Torre-Bastida, A.I., Zárate, G., Miñón, R. and Almeida, A. (2024) 'A Joint Study of the Challenges, Opportunities, and Roadmap of MLOps and AIOps: A Systematic Survey', ACM Computing Surveys, 56(4), Article 84, pp. 1–30. <https://doi.org/10.1145/3625289>.
- Dritsas, E. and Trigka, M. (2026) 'Machine learning in modern database systems: Techniques, architectures, and deployment challenges', Engineering Applications of Artificial Intelligence, 170, 114225. <https://doi.org/10.1016/j.engappai.2026.114225>.
- Du, M., Li, F., Zheng, G. and Srikumar, V. (2017) 'DeepLog: Anomaly Detection and Diagnosis from System Logs through Deep Learning',

- Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, pp. 1285–1298.
<https://doi.org/10.1145/3133956.3134015>.
- Gao, X., Yuan, P., Han, S., Liu, Y., Xia, X., Zhang, H., Cui, J., Li, H. and Zhao, K. (2025) 'Insights into KPI-based performance anomaly detection in database systems: A comprehensive study', *Expert Systems with Applications*, 285, 127959. <https://doi.org/10.1016/j.eswa.2025.127959>.
 - Guo, H., Yuan, S. and Wu, X. (2021) 'LogBERT: Log Anomaly Detection via BERT', 2021 International Joint Conference on Neural Networks (IJCNN), pp. 1–8. <https://doi.org/10.1109/IJCNN52387.2021.9534113>.
 - He, P., Zhu, J., Zheng, Z. and Lyu, M.R. (2017) 'Drain: An Online Log Parsing Approach with Fixed Depth Tree', 2017 IEEE 24th International Conference on Web Services (ICWS), pp. 33–40. <https://doi.org/10.1109/ICWS.2017.13>.
 - Huang, S., Wang, Z., Zhang, X., Tu, Y., Li, Z. and Cui, B. (2023) 'DBPA: A Benchmark for Transactional Database Performance Anomalies', *Proceedings of the ACM on Management of Data*, 1(1), pp. 1–26. <https://doi.org/10.1145/3588926>.
 - Li, Z., Zhu, Y. and van Leeuwen, M. (2024) 'A Survey on Explainable Anomaly Detection', *ACM Transactions on Knowledge Discovery from Data*, 18(1), Article 23, pp. 1–54. <https://doi.org/10.1145/3609333>.
 - Liu, X., Yin, Z., Zhao, C., Ge, C., Chen, L., Gao, Y., Li, D., Wang, Z., Liang, G., Tan, J. and Li, F. (2022) 'PinSQL: Pinpoint Root Cause SQLs to Resolve Performance Issues in Cloud Databases', 2022 IEEE 38th International Conference on Data Engineering (ICDE), pp. 2549–2561. <https://doi.org/10.1109/ICDE53745.2022.00236>.
 - Meng, W., Liu, Y., Zhu, Y., Zhang, S., Pei, D., Liu, Y., Chen, Y., Zhang, R., Tao, S., Sun, P. and Zhou, R. (2019) 'LogAnomaly: Unsupervised Detection of Sequential and Quantitative Anomalies in Unstructured Logs', *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, pp. 4739–4745. <https://doi.org/10.24963/ijcai.2019/658>.
 - Notaro, P., Cardoso, J. and Gerndt, M. (2021) 'A Survey of AIOps Methods for Failure Management', *ACM Transactions on Intelligent Systems and Technology*, 12(6), Article 81. <https://doi.org/10.1145/3483424>.
 - Ouared, A., Amrani, M. and Schobbens, P.-Y. (2023) 'Explainable AI for DBA: Bridging the DBA's experience and machine learning in tuning database systems', *Concurrency and Computation: Practice and Experience*, 35(21), e7698. <https://doi.org/10.1002/cpe.7698>.
 - Pang, G., Shen, C., Cao, L. and van den Hengel, A. (2021) 'Deep Learning for Anomaly Detection: A Review', *ACM Computing Surveys*, 54(2), Article 38, pp. 1–38. <https://doi.org/10.1145/3439950>.
 - Schmidl, S., Wenig, P. and Papenbrock, T. (2022) 'Anomaly Detection in Time Series: A Comprehensive Evaluation', *Proceedings of the VLDB Endowment*, 15(9), pp. 1779–1797. <https://doi.org/10.14778/3538598.3538602>.
 - Su, Y., Zhao, Y., Niu, C., Liu, R., Sun, W. and Pei, D. (2019) 'Robust Anomaly Detection for Multivariate Time Series through Stochastic Recurrent Neural Network', *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 2828–2837. <https://doi.org/10.1145/3292500.3330672>.
 - Tuli, S., Casale, G. and Jennings, N.R. (2022) 'TranAD: Deep Transformer Networks for Anomaly Detection in Multivariate Time Series Data', *Proceedings of the VLDB Endowment*, 15(6), pp. 1201–1214. <https://doi.org/10.14778/3514061.3514067>.
 - Wenig, P., Schmidl, S. and Papenbrock, T. (2022) 'TimeEval: A Benchmarking Toolkit for Time Series Anomaly Detection Algorithms', *Proceedings of the VLDB Endowment*, 15(12), pp. 3678–3681. <https://doi.org/10.14778/3554821.3554873>.
 - Zhou, X., Chai, C., Li, G. and Sun, J. (2022) 'Database Meets Artificial Intelligence: A Survey', *IEEE Transactions on Knowledge and Data Engineering*, 34(3), pp. 1096–1116. <https://doi.org/10.1109/TKDE.2020.2994641>.
 - Digital Government Authority (DGA) (2024) Digital Transformation Basic Standards, Version 3.0. Riyadh: Digital Government Authority. Available at: <https://dga.gov.sa/> (Accessed: 3 September 2026).
 - Digital Government Authority (DGA) (2025) Effective Integration of Cloud Computing in Digital Transformation of the Education Sector, Version 1.0. Riyadh: Digital Government Authority. Available at: <https://dga.gov.sa/> (Accessed: 3 September 2026).
 - National Cybersecurity Authority (NCA) (2022) Data Cybersecurity Controls (DCC-1:2022). Riyadh: National Cybersecurity Authority. Available at: <https://nca.gov.sa/en/regulatory-documents/controls-list/dcc/> (Accessed: 3 September 2026).
 - National Cybersecurity Authority (NCA) (2024) Essential Cybersecurity Controls (ECC 2-2024). Riyadh: National Cybersecurity Authority. Available at: <https://nca.gov.sa/en/regulatory-documents/controls-list/ecc/> (Accessed: 3 September 2026).
 - National Cybersecurity Authority (NCA) (2024) Cloud Cybersecurity Controls (CCC-2:2024).

Riyadh: National Cybersecurity Authority. Available at: <https://nca.gov.sa/en/regulatory-documents/controls-list/ccc/> (Accessed: 3 September 2026).

- Saudi Data & AI Authority (SDAIA) (2023) AI Ethics Principles, Version 1.0. Riyadh: SDAIA. Available at: <https://sdaia.gov.sa/en/SDAIA/about/Documents/ai-principles.pdf> (Accessed: 3 September 2026).
- Saudi Data & AI Authority (SDAIA) (2024) AI Adoption Framework. Riyadh: SDAIA. Available at: <https://sdaia.gov.sa/en/SDAIA/about/Files/AIAdoptionFramework.pdf> (Accessed: 3 September 2026).
- Saudi Data & AI Authority (SDAIA) (2026a) National AI Risk Management Framework.

Riyadh: SDAIA. Available at: <https://sdaia.gov.sa/en/MediaCenter/KnowledgeCenter/Pages/SDAIAPublications.aspx> (Accessed: 3 September 2026).

- Saudi Data & AI Authority (SDAIA) (n.d.-a) National Strategy for Data & AI. Available at: <https://sdaia.gov.sa/en/SDAIA/SdaiaStrategies/Pages/NationalStrategyForDataAndAI.aspx> (Accessed: 3 September 2026).
- Saudi Data & AI Authority (SDAIA) (n.d.-b) Strategic Objectives. Available at: <https://sdaia.gov.sa/en/SDAIA/SdaiaStrategies/Pages/StrategicGoals.aspx> (Accessed: 3 September 2026).
- Saudi Vision 2030 (2026) Vision 2030 Annual Report 2025. Riyadh: Kingdom of Saudi Arabia. Available at: <https://www.vision2030.gov.sa/> (Accessed: 3 September 2026).