World Journal of Advanced Engineering Technology and Sciences
eISSN: 2582-8266
Cross Ref DOI: 10.30574/wjaets
Journal homepage: https://wjaets.com/

(REVIEW ARTICLE)

Check for updates

ML-driven data engineering pipeline for health informatics

NISHANTH JOSEPH PAULRAJ *

Thermo Fisher Scientific, USA.

World Journal of Advanced Engineering Technology and Sciences, 2025, 15(02), 765-773

Publication history: Received on 27 March 2025; revised on 03 May 2025; accepted on 06 May 2025

Article DOI: https://doi.org/10.30574/wjaets.2025.15.2.0629

Abstract

This article presents a comprehensive framework for implementing machine learning-driven data engineering pipelines in healthcare informatics. Healthcare data presents unique challenges including high dimensionality, heterogeneity across sources, missing values, temporal dependencies, and strict privacy requirements. To address these challenges, we propose a four-layer architecture comprising data ingestion, data processing, ML modeling, and model management components. The pipeline leverages Apache Spark and Delta Lake for robust data processing, modern ML frameworks for predictive modeling, and MLflow for model lifecycle management. It demonstrates the practical application of this architecture through a sepsis risk prediction use case, highlighting how temporal patterns in clinical data can be leveraged for early intervention. The article also explores deep learning approaches for genomic data analysis and discusses critical implementation challenges including data privacy, class imbalance, model explainability, and model drift. Throughout, It emphasizes best practices that balance technical performance with clinical utility and regulatory compliance, providing a roadmap for healthcare organizations seeking to implement scalable ML solutions.

Keywords: Healthcare Data Engineering; Machine Learning Pipelines; Clinical Predictive Modeling; Model Lifecycle Management; Sepsis Prediction

1. Introduction

Healthcare organizations are increasingly leveraging machine learning to extract actionable insights from clinical and genomic datasets. The healthcare analytics market continues to expand rapidly as institutions recognize the value of data-driven decision making. Analysis of electronic health record (EHR) data has demonstrated significant potential for improving patient outcomes through predictive modeling. Research has shown that deep learning approaches applied to EHR data can accurately predict important clinical outcomes including in-hospital mortality, 30-day readmission, and prolonged length of stay with high accuracy [2]. Studies utilizing large-scale healthcare datasets comprising adult patients and numerous data points have demonstrated the feasibility of building models that achieve high accuracy for mortality prediction and readmission prediction [2]. Building robust data engineering pipelines that can handle the volume, variety, and velocity of healthcare data while maintaining compliance with regulatory requirements presents significant challenges. This article explores the architecture and implementation of a scalable ML-driven data engineering pipeline specifically designed for health informatics applications.

1.1. The Challenge of Healthcare Data

Healthcare data presents unique challenges that necessitate specialized approaches to data engineering. Patient records contain hundreds of variables across different timeframes, creating high-dimensional datasets that are difficult to process using traditional methods. The heterogeneity of healthcare data further complicates analysis, as information comes from diverse sources including EHRs, laboratory systems, imaging repositories, and genomic sequencing platforms. The MIMIC-III database exemplifies this complexity, containing data from distinct adult patients across

* Corresponding author: NISHANTH JOSEPH PAULRAJ.

numerous ICU stays, with information spanning demographics, vital signs, laboratory tests, medications, and clinical notes [4].

Missing values represent another significant challenge, as clinical data often contains gaps due to variations in care processes and documentation practices. Healthcare data also exhibits strong temporal dependencies, with health outcomes influenced by sequences of events and time intervals between interventions. Finally, strict regulatory compliance requirements under frameworks such as HIPAA and GDPR necessitate careful attention to data privacy and security throughout the pipeline. These characteristics require specialized pipelines that can ingest, transform, and analyze healthcare data while maintaining data integrity and facilitating model development.

Table 1 Healthcare Data Challenges and Solutions [4]

Challenge	Solution Approach
High Dimensionality	Feature selection, dimensionality reduction techniques
Heterogeneity	Unified data schema, standardized formats (FHIR)
Missing Values	Imputation strategies, algorithms tolerant to missing data
Temporal Dependencies	Time-series modeling, sequence analysis approaches
Privacy Requirements	De-identification, role-based access, audit logging

2. Architecture overview

Our pipeline architecture consists of four core components designed to address the unique challenges of healthcare data processing. The data ingestion layer captures information from various clinical and genomic sources, enabling comprehensive analysis across data modalities. The data processing layer performs ETL operations using Apache Spark and Delta Lake, leveraging Spark's unified API for batch, streaming, and interactive queries [3]. Apache Spark provides substantial performance improvements over traditional Hadoop MapReduce for healthcare workloads through its in-memory processing capabilities [3].

The ML modeling layer implements prediction models using frameworks such as XGBoost, LightGBM, or PyTorch, selected based on the specific requirements of the healthcare prediction task. Finally, the model management layer handles versioning, tracking, and deployment with MLflow, an open-source platform designed to accelerate the machine learning lifecycle through standardized experiment tracking, model packaging, and deployment [5]. This layered architecture provides the flexibility and scalability required for complex healthcare analytics while maintaining the rigorous standards necessary for clinical applications.

Table 2 Healthcare Data Engineering Pipeline Components [5]

Layer	Key Technologies	Primary Functions	Healthcare Benefits
Data Ingestion	FHIR APIs, Apache Spark	Extract data from clinical and genomic sources	Interoperability, scalable processing
Data Processing	Delta Lake, Spark ETL	Clean, transform, feature engineering	ACID transactions, audit capabilities
ML Modeling	XGBoost/LightGBM, PyTorch	Prediction models for clinical outcomes	Interpretability, complex pattern recognition
Model Management	MLflow	Versioning, deployment, monitoring	Regulatory compliance, lifecycle management

2.1. Data Ingestion Layer

The ingestion layer handles the extraction of data from multiple sources including electronic health records, laboratory information systems, and genomic sequencing platforms. For clinical data, we implement FHIR-compatible APIs to ensure interoperability with existing healthcare IT infrastructure. The ingestion process utilizes Apache Spark's distributed computing capabilities to efficiently process large volumes of healthcare data in parallel. For genomic data,

specialized parsing tools handle the complexity of formats such as VCF (Variant Call Format) files, which contain detailed information about genomic variants identified through sequencing.

The ingestion layer incorporates robust data validation to ensure that incoming information meets quality standards before entering the pipeline. All data is persisted using Delta Lake, which provides ACID transaction guarantees essential for maintaining the consistency of healthcare information [1]. Delta Lake's time travel capabilities enable access to previous versions of data, facilitating audit processes and compliance with healthcare regulations [1]. This approach ensures that the data ingestion process maintains the provenance and lineage information critical for clinical applications.

2.2. Data Processing Layer

The processing layer handles data cleaning, transformation, and feature engineering to prepare healthcare information for analysis. Apache Spark's distributed processing framework enables efficient handling of the large-scale data typical in healthcare environments. The processing layer implements specialized techniques for handling missing values, a common challenge in clinical datasets. Temporal feature engineering is particularly important for healthcare applications, as the progression and timing of clinical events significantly impact patient outcomes.

Delta Lake provides several key advantages for healthcare data processing that address the unique requirements of clinical information. Its ACID transactions ensure data consistency even with concurrent operations, a critical requirement in healthcare environments where multiple systems may simultaneously access and modify patient information [1]. The time travel capability enables access to previous versions of data, facilitating compliance with healthcare regulations that require comprehensive audit trails. Delta Lake's schema evolution capabilities adapt to changing data structures without disrupting the pipeline, an important consideration given the evolving nature of healthcare documentation [1]. Finally, data quality enforcement through constraints ensures that only valid information enters the analytical pipeline, maintaining the integrity necessary for clinical decision support.

2.3. ML Modeling Layer

The modeling layer implements machine learning models tailored to healthcare prediction tasks. For clinical outcome prediction, gradient boosting frameworks such as XGBoost and LightGBM offer strong performance while maintaining interpretability, an essential requirement for healthcare applications. Deep learning approaches using PyTorch provide additional capabilities for handling complex, unstructured data such as medical imaging and clinical notes.

Feature preparation for healthcare models requires careful attention to data characteristics such as temporal dependencies and missing values. The modeling process follows established best practices including proper train-test splitting, hyperparameter optimization, and rigorous evaluation. Model evaluation metrics are selected based on the clinical context, with emphasis on measures that reflect the real-world impact of prediction errors. For critical care applications such as sepsis prediction, high sensitivity is prioritized given the severe consequences of missed cases.

Research has demonstrated that deep learning models trained on EHR data can achieve high accuracy for important clinical outcomes. Models applied to datasets from multiple medical centers have achieved strong performance for predicting inpatient mortality and predicting readmissions [2]. These results highlight the potential of ML-driven approaches to provide actionable insights for healthcare providers.

2.4. Model Management Layer

The model management layer handles the operationalization of healthcare prediction models, ensuring that they can be reliably deployed and monitored in clinical environments. MLflow provides comprehensive capabilities for experiment tracking, model versioning, and deployment that address the unique requirements of healthcare applications [5]. Its language-agnostic design supports multiple ML frameworks, enabling flexibility in model development while maintaining standardized processes for deployment [5].

The management layer implements rigorous version control for models and feature transformations, ensuring that the provenance of predictions can be traced throughout the model lifecycle. Automated monitoring identifies potential issues such as model drift, triggering retraining processes when performance degrades below specified thresholds. This approach ensures that healthcare prediction models maintain their accuracy over time despite changing clinical patterns and patient populations.

Integration with clinical workflows is carefully designed to present predictions in a manner that supports rather than replaces clinical decision-making. The management layer includes capabilities for generating explanations that help healthcare providers understand the factors contributing to specific predictions, enhancing trust and facilitating appropriate interpretation in the clinical context.

3. Use case: sepsis risk prediction

Sepsis is a life-threatening condition characterized by a dysregulated host response to infection that can lead to organ dysfunction and death. Early detection and intervention are critical to improving patient outcomes, with research demonstrating that prompt treatment significantly reduces mortality. Our pipeline implementation for sepsis prediction leverages data from multiple sources to identify patients at risk before clinical manifestations become apparent.

The data ingestion process captures vital signs, laboratory results, medications, and demographic information from the electronic health record. Feature engineering focuses on temporal patterns in vital signs and laboratory values, with particular attention to trends that may indicate developing infection or organ dysfunction. The MIMIC-III database has been instrumental in developing such models, providing comprehensive data from distinct adult patients across numerous ICU stays [4]. This rich dataset includes the temporal sequence of interventions, vital signs, and laboratory results necessary for developing accurate sepsis prediction models.

Model selection for sepsis prediction emphasizes both accuracy and interpretability, with gradient boosting frameworks such as LightGBM providing a good balance of these characteristics. Research using deep learning approaches applied to EHR data has demonstrated the potential to predict sepsis with high accuracy, achieving performance comparable to that seen in mortality prediction models [2]. The model produces a continuous risk score that can be integrated into clinical workflows to alert medical staff when a patient's risk exceeds specified thresholds, enabling early intervention before sepsis becomes clinically apparent.

3.1. Challenges and Best Practices

Healthcare data pipelines must adhere to strict privacy regulations that protect patient information. Implementing robust data governance frameworks ensures that all processing complies with relevant legislation such as HIPAA and GDPR. Appropriate de-identification techniques are applied where necessary, although maintaining the utility of data for analysis while ensuring privacy presents ongoing challenges. Comprehensive audit logs document all data access and transformations, providing the transparency required for healthcare applications. Role-based access controls restrict data visibility based on user responsibilities, implementing the principle of least privilege essential for clinical data security.

Many healthcare prediction tasks involve rare events, creating challenges related to class imbalance. Techniques such as synthetic sampling, cost-sensitive learning, and specialized evaluation metrics address this issue. For healthcare applications, proper evaluation requires metrics that reflect the clinical impact of different types of errors. Precision-recall curves and area under the precision-recall curve (PR AUC) often provide more informative assessments than traditional ROC curves for imbalanced healthcare datasets.

Clinical applications require interpretable predictions that healthcare providers can understand and trust. Techniques such as SHAP (SHapley Additive exPlanations) values quantify the contribution of individual features to specific predictions, enhancing transparency. Feature importance visualization presents this information in an accessible format for clinical users, supporting appropriate interpretation of model outputs. Case-based reasoning provides explanations by identifying similar historical cases, a form of explanation that aligns well with clinical reasoning patterns.

Healthcare patterns change over time due to factors such as evolving treatment guidelines, changing patient populations, and technological advances. Continuous monitoring of model performance identifies degradation before it impacts clinical care. Automated retraining pipelines update models when drift exceeds specified thresholds, maintaining accuracy despite changing conditions. Version control for models and feature transformations ensures that all predictions can be traced to specific model versions, maintaining the provenance information critical for clinical applications.

3.2. Deep Learning Applications for Genomic Data Analysis

Deep learning approaches have demonstrated remarkable success in analyzing complex genomic datasets, with PyTorch emerging as a preferred framework due to its flexibility and robust gradient computation capabilities. Recent advances in genomic deep learning architectures have shown significant improvements in variant classification tasks, with convolutional neural networks demonstrating particular efficacy for sequence-based predictions. The computational efficiency of these approaches has been extensively documented in comprehensive benchmarking studies published in Cell Patterns, with PyTorch implementations showing superior performance on memory-intensive genomic tasks compared to alternative frameworks [6]. The scalability of these frameworks makes them particularly suitable for whole-genome analysis workflows, where processing requirements can easily exceed traditional computing constraints.

The implementation of genomic neural networks requires careful architectural design to address the unique characteristics of genomic data. Convolutional neural network architectures optimized for genomic sequence analysis typically incorporate specialized layers that capture the unique characteristics of biological sequences. These architectures have been validated on diverse genomic datasets, with performance characteristics showing robust generalization across different sequence analysis tasks. The convolutional layer captures local sequential patterns in genomic data, which is particularly important for identifying motifs associated with regulatory elements or splice sites. Research published in Cell Patterns has demonstrated that these architectural designs achieve significant accuracy improvements when applied to challenging genomic prediction tasks, with performance gains attributed to the ability of convolutional layers to identify biologically relevant sequence patterns without explicit feature engineering [6]. The pooling operations reduce dimensionality while preserving essential features, addressing the computational challenges posed by the high-dimensional nature of genomic data.

For model training, PyTorch's dynamic computation graph facilitates implementation of complex training procedures that can incorporate biological constraints or domain knowledge. Training efficiency has been extensively optimized in recent implementations, with techniques such as mixed-precision training showing particular promise for accelerating genomic deep learning without sacrificing accuracy. Comprehensive evaluations published in the literature have validated that these optimization approaches can reduce training time substantially while maintaining model quality, enabling researchers to iterate more rapidly on complex genomic prediction tasks [6].

3.3. Model Management Layer for Healthcare Applications

The effective deployment of healthcare machine learning models requires robust management infrastructure that addresses the unique requirements of clinical applications. MLflow has emerged as a leading platform for this purpose, with adoption rates increasing significantly across healthcare organizations seeking to operationalize machine learning for clinical applications. Recent surveys have documented the benefits of structured model management approaches for healthcare applications, highlighting improvements in deployment efficiency, regulatory compliance, and maintenance overhead [8].

The implementation of MLflow for healthcare model management involves several key components including tracking servers, model registries, and deployment pipelines. This implementation approach integrates with the broader MLOps ecosystem, supporting rigorous deployment requirements essential for healthcare applications. The model registry component is particularly important for healthcare applications, as it provides the version control and lineage tracking required for regulatory compliance under frameworks such as FDA's proposed regulatory approach for AI/ML-based Software as a Medical Device. Recent publications in JAMIA have emphasized the importance of comprehensive provenance tracking for clinical ML applications, noting that systems without adequate lineage documentation face significant challenges during regulatory review processes [8]. The structured staging approach used in modern MLflow implementations aligns with best practices for clinical model deployment, facilitating controlled rollout and validation before widespread clinical implementation.

Performance characteristics of MLflow-managed healthcare models have been documented in recent literature, confirming that well-implemented model management infrastructures can maintain the reliability requirements necessary for clinical applications. The ability to rapidly transition between model versions becomes particularly important when performance degradation is detected or when clinical guidelines change, scenarios that occur regularly in healthcare environments. Implementation patterns documented in JAMIA highlight the importance of automated deployment pipelines with comprehensive testing, noting that organizations following these practices reported significantly fewer adverse events related to ML-based decision support [8].

3.4. Use Case: Sepsis Risk Prediction

Sepsis represents a significant healthcare challenge globally, with substantial impact on patient outcomes and healthcare resources. The timely prediction of sepsis remains a high-priority target for machine learning applications in healthcare, with potential to significantly improve patient outcomes through earlier intervention. A systematic review and meta-analysis of machine learning approaches for sepsis prediction evaluated numerous studies representing many patient encounters, finding substantial variability in methodological approaches and reported performance [7]. The analysis found that gradient boosting methods were most commonly employed, followed by random forests and neural networks, with no clear superiority of any single algorithm across all clinical contexts.

Our pipeline implementation for sepsis prediction leverages multiple data sources to identify at-risk patients before clinical manifestations become apparent:

The data ingestion process incorporates electronic health record data spanning vital signs, laboratory results, medication records, and demographic information. The systematic review of sepsis prediction models identified that the most effective implementations incorporated temporal features derived from these data sources, with particular emphasis on subtle changes in vital signs and laboratory values that precede obvious clinical deterioration [7]. Feature engineering focuses on temporal patterns in vital signs and laboratory values, with comprehensive meta-analysis confirming that models incorporating trend analysis demonstrated superior performance compared to those using only static measurements, with significant AUROC improvements when temporal derivatives were included [7].

Model selection for sepsis prediction emphasizes both accuracy and interpretability, with gradient boosting frameworks such as LightGBM providing a good balance of these characteristics. The systematic review and meta-analysis determined that gradient boosting models achieved strong AUROC performance across multiple clinical settings, with relatively consistent performance across different hospital types [7]. These methods demonstrated particular strength in maintaining specificity while improving sensitivity compared to traditional screening tools.

Real-time scoring capabilities ensure timely risk assessment during patient stays, critical requirement given evidence that each hour of delayed sepsis treatment is associated with increased mortality. The meta-analysis found that prediction windows of several hours provided optimal balance between advance notice and prediction accuracy, with performance degrading substantially for longer prediction horizons [7]. Implementations successfully achieving this prediction window reported end-to-end latency metrics that included data collection, feature extraction, model inference, and alert generation, with comprehensive systems maintaining appropriate latencies for the majority of predictions.

Key features for sepsis prediction have been extensively validated across multiple studies, with the systematic review identifying vital sign abnormalities, laboratory markers, and demographic risk factors as consistent predictors across diverse clinical populations [7]. The model outputs a continuous risk score that can be integrated into clinical workflows to alert medical staff when a patient's risk exceeds specified thresholds, enabling early intervention before sepsis becomes clinically apparent.

4. Challenges and Best Practices in Healthcare ML

4.1. Data Privacy and Compliance

Healthcare data pipelines must adhere to strict privacy regulations that protect patient information while enabling analytics that improve care quality. Recent publications in JAMIA have emphasized the growing complexity of healthcare data governance, noting that organizations must balance competing requirements for data accessibility, security, and regulatory compliance [8]. Implementation of robust data governance frameworks provides the foundation for responsible healthcare analytics, with structured approaches demonstrating measurably improved compliance outcomes compared to ad-hoc strategies. These frameworks establish clear data stewardship responsibilities, implement consistent data handling protocols, and provide audit mechanisms that support regulatory requirements.

Application of appropriate de-identification techniques represents a critical component of healthcare data protection strategies, with recent literature emphasizing the importance of contextually appropriate methods. Research published in JAMIA has highlighted the evolution of healthcare de-identification approaches beyond simple rule-based methods, with increased adoption of statistical disclosure limitation techniques that provide mathematically rigorous privacy guarantees [8]. These approaches enable analytics while maintaining regulatory compliance, though implementation

complexity and computational requirements present ongoing challenges for resource-constrained healthcare organizations.

Comprehensive audit capabilities provide essential transparency for healthcare data initiatives, with recent publications emphasizing that effective auditing extends beyond simple access logging to include purpose specification, authorization verification, and anomaly detection. Organizations implementing comprehensive audit frameworks report significantly improved ability to demonstrate regulatory compliance during formal evaluations, with structured audit capabilities supporting both routine monitoring and focused investigations of potential data misuse [8]. The implementation of role-based access controls with fine-grained permissions has become standard practice for healthcare analytics environments, though challenges remain in harmonizing access models across disparate systems within complex healthcare enterprises.

4.2. Class Imbalance

Many healthcare prediction tasks involve rare events, creating challenges related to class imbalance that must be addressed to develop effective models. The systematic review and meta-analysis of sepsis prediction models identified class imbalance as a universal challenge, with sepsis typically represented in only a small fraction of hospitalized patients in included studies [7]. This imbalance creates substantial methodological challenges, with naive models often demonstrating high accuracy but poor sensitivity for the critical minority class. The meta-analysis found that several approaches have demonstrated effectiveness in addressing this challenge, with results varying based on the specific clinical context and dataset characteristics.

Synthetic sampling techniques have shown promise for improving model performance on imbalanced healthcare datasets, though implementation approaches vary widely across studies. The systematic review identified that among studies employing synthetic sampling, a significant majority utilized SMOTE or its variants, with the remainder employing alternative approaches including generative adversarial networks for synthetic data creation [7]. Performance improvements associated with synthetic sampling varied substantially based on baseline model characteristics and implementation details, with notable sensitivity improvements at comparable specificity levels.

Table 3 Class Imbalance Techniques in Healthcare ML [7]

Technique	Application
Synthetic Sampling	Generation of minority class examples (SMOTE/ADASYN)
Cost-Sensitive Learning	Higher penalties for missing critical conditions
Evaluation Metrics	PR AUC, F1 score over standard accuracy
Threshold Optimization	Adjusted decision boundaries for clinical relevance

Selection of appropriate evaluation metrics for imbalanced healthcare datasets has been extensively discussed in recent literature, with consensus emerging around the limitations of accuracy for rare-event prediction tasks. The systematic review found that a large majority of recent sepsis prediction studies reported area under the precision-recall curve (AUPRC) in addition to traditional AUROC, reflecting growing recognition of AUPRC's superior ability to assess performance on imbalanced datasets [7]. Stratified performance reporting has similarly gained traction, with increased emphasis on understanding model behavior across important patient subgroups rather than focusing exclusively on aggregate performance metrics.

Cost-sensitive learning approaches demonstrate particular promise for clinical applications where false negatives and false positives have substantially different implications. The meta-analysis found that approximately half of reviewed studies employed some form of cost-sensitive training, though methodology and effectiveness reporting varied widely [7]. This heterogeneity complicates direct comparison, though studies providing detailed methodology generally reported improved clinical utility compared to models without cost-sensitive adjustments.

4.3. Model Explainability

Clinical applications require interpretable predictions that healthcare providers can understand and trust, a requirement extensively documented in recent healthcare informatics literature. Research published in JAMIA has highlighted the importance of explanation quality in determining clinical adoption of machine learning tools, with surveys indicating that inadequate explainability represents a primary barrier to implementation for many healthcare

organizations [8]. These findings underscore the importance of incorporating explainability considerations throughout the model development lifecycle rather than treating them as post-hoc additions.

Comprehensive literature reviews have documented diverse approaches to healthcare model explainability, with techniques varying based on model architecture, clinical domain, and intended user characteristics. Recent publications have emphasized the importance of aligning explanation methods with clinical reasoning patterns, noting that explanations that contradict established domain knowledge face significant adoption barriers regardless of technical sophistication [8]. This alignment requires deep collaboration between technical and clinical stakeholders throughout the development process, with iterative refinement based on clinician feedback.

Implementation of feature importance visualization has become standard practice for explainable healthcare AI, though approaches vary widely in their sophistication and clinical utility. Recent literature has highlighted the evolution of these visualizations beyond simple importance rankings to incorporate contextual information that supports clinical interpretation [8]. These advanced approaches present model outputs within familiar clinical frameworks, emphasizing factors that align with established medical knowledge while providing appropriate context for novel or unexpected predictors.

Case-based explanations provide an alternative approach that aligns particularly well with clinical reasoning patterns, presenting predictions in relation to similar historical cases rather than abstract feature contributions. Research published in JAMIA has documented increased clinician comfort with this explanation paradigm, particularly among specialists with extensive domain expertise [8]. The effectiveness of case-based approaches depends heavily on implementation quality, with considerations including appropriate similarity metrics, case selection strategies, and presentation formats that highlight clinically relevant similarities and differences.

4.4. Model Drift

Healthcare patterns change over time due to factors such as evolving treatment guidelines, changing patient populations, and technological advances, creating challenges for machine learning models that must maintain performance despite these shifts. Recent literature has extensively documented the prevalence and impact of model drift in healthcare applications, with research in JAMIA highlighting that performance degradation occurs across diverse clinical domains and modeling approaches [8]. This universal challenge necessitates structured approaches for monitoring and maintaining healthcare models throughout their lifecycle.

Implementation of continuous monitoring represents the foundation of effective drift management, providing visibility into model performance across relevant dimensions and enabling timely response when degradation occurs. Recent publications have documented diverse approaches to healthcare model monitoring, with emphasis on metrics aligned with clinical utility rather than purely technical performance measures [8]. Effective implementations typically incorporate both population-level metrics and stratified analysis across important patient subgroups, enabling early detection of performance disparities that might otherwise remain hidden in aggregate statistics.

Table 4 Model Drift Management Framework [8]

Component	Implementation
Performance Monitoring	Tracking clinical and technical metrics
Distribution Analysis	Detecting shifts in patient data patterns
Retraining Process	Threshold-based triggers, validation protocol
Version Control	Comprehensive model lineage and metadata

Establishment of automated retraining pipelines based on drift detection has emerged as best practice for maintaining healthcare model performance, though implementation approaches vary based on organizational capabilities and application characteristics. Research published in JAMIA has documented the relationship between retraining frequency and performance stability, noting that optimal schedules vary substantially based on factors including data volume, condition stability, and algorithmic approach [8]. Organizations implementing automated retraining capabilities report improved performance stability compared to those relying on manual processes, though substantial implementation challenges remain related to validation requirements and clinical workflow integration.

Comprehensive version control for models and feature transformations provides the foundation for responsible model updating, ensuring that all predictions can be traced to specific model versions throughout the clinical lifecycle. Recent literature has emphasized the importance of maintaining complete lineage information for healthcare models, noting that this capability supports both technical troubleshooting and regulatory compliance [8]. Organizations implementing robust version control report improved ability to diagnose performance issues and respond effectively to regulatory inquiries, capabilities that become increasingly important as machine learning applications expand across healthcare domains.

5. Conclusion

ML-driven data engineering pipelines for health informatics require specialized architectures that address the unique challenges of healthcare data. By leveraging Apache Spark's unified engine for big data processing, Delta Lake's transaction guarantees and versioning capabilities, and MLflow's standardized approach to the machine learning lifecycle, organizations can build scalable solutions for critical use cases such as readmission prediction, sepsis risk assessment, and laboratory anomaly detection. The architecture presented in this article provides a foundation that can be adapted to various healthcare prediction tasks while maintaining the necessary standards for reliability, interpretability, and regulatory compliance. Deep learning approaches have demonstrated considerable promise for complex tasks including genomic data analysis, though implementation requires careful consideration of model explainability to ensure clinical adoption. Addressing challenges related to data privacy, class imbalance, model interpretability, and model drift remains essential for successful deployment in healthcare environments. As healthcare continues to generate increasing volumes of data across diverse systems, robust data engineering pipelines will play a crucial role in extracting actionable insights that improve clinical decision-making and patient care. Organizations implementing these solutions must balance technical innovation with practical clinical requirements, ensuring that machine learning systems augment rather than replace clinical expertise while maintaining the highest standards of data stewardship and regulatory compliance.

References

[1] Hanza Parayil Salim, “A Comparative Study of Delta Lake as a Preferred ETL and Analytics Database,” International Journal of Computer Trends and Technology, 2025, Available: https://ijcttjournal.org/2025/Volume-73%20Issue-1/IJCTT-V73I1P108.pdf

[2] Alvin Rajkomar, et al, “Scalable and accurate deep learning for electronic health records,” January 2018, npj Digital Medicine, Available: https://www.researchgate.net/publication/322695006_Scalable_and_accurate_deep_learning_for_electronic_health_records

[3] Matei Zaharia, et al, “Apache spark: A unified engine for big data processing,” November 2016, Available: https://www.researchgate.net/publication/310613994_Apache_spark_A_unified_engine_for_big_data_processing

[4] Alistair Edward William Johnson, et al, “MIMIC-III, a freely accessible critical care database,” May 2016, Scientific Data, Available: https://www.researchgate.net/publication/303499206_MIMIC-III_a_freely_accessible_critical_care_database

[5] Andrew Chen, et al, “Developments in MLflow: A System to Accelerate the Machine Learning Lifecycle,” June 2020, Online, Available: https://www.researchgate.net/publication/342250885_Developments_in_MLflow_A_System_to_Accelerate_the_Machine_Learning_Lifecycle

[6] Hao Wang, et al, “Cropformer: An interpretable deep learning framework for crop genomic prediction,” 10 March 2025, Online, Available: https://www.sciencedirect.com/science/article/pii/S2590346224006448

[7] Lucas Fleuren, et al, “Machine learning for the prediction of sepsis: a systematic review and meta-analysis of diagnostic test accuracy,” January 2020, Intensive Care Medicine, Available: https://www.researchgate.net/publication/338731040_Machine_learning_for_the_prediction_of_sepsis_a_systematic_review_and_meta-analysis_of_diagnostic_test_accuracy

[8] Adam Paul Yan, et al, “A roadmap to implementing machine learning in healthcare: from concept to practice,” 2025 Jan, NIH, Available: https://pmc.ncbi.nlm.nih.gov/articles/PMC11788154/