MEDICA AI

(i) Problem Definition

Ineffective manual inventory management in health facilities does not adequately account for dynamic factors such as patient demand, supplier lead times, medicine shelf life and external shortage alerts. Existing systems rather operate reactively rather than proactively.

Pharmaceutical supply chains, particularly at the retail and hospital pharmacy level face two significant interconnected challenges.

Medicine Expiry: An estimated of $6 billion worth of medicines are discarded annually due to expiration, resulting to substantial financial losses and contributing to environmental waste.

Medicine shortage: Frequent and unpredictable shortages of essential medicines disrupt patient care, delay treatments, and force healthcare providers to use fewer effective alternatives.

(ii) Core objectives

1. Enhance predictability capability and proactive Decision-making

- Achieve a demand forecasting accuracy for over 90% for fast moving items and over 75% for slow moving, intermittent-demand items.
- Ensure that over 85% of potential stockouts and expiries are flagged with a high-risk alert at least 3 months in advance and 4 weeks, allowing sufficient time for intervention.
- Automatically generate a daily “Actionable insights” report for pharmacists, highlighting top-priority ordering and dispensing actions.

2. Optimize Inventory Efficiency by Dynamically Balancing Stock levels.

The core function of this Medica AI tool is to solve the paradox of having too much and too little stock simultaneously. This objective directly tackles the root cause of both expiries and shortages.

- Reduce volume of expired drugs by at least 30% within the first year of implementation.
- Decrease the frequency of stockouts for mission-critical medications by at least 50%.
- Improve the overall inventory turnover ratio, indicating faster conversion of stock into sales without gaps.

3. Maximize Usability and Build trust Among End-users.

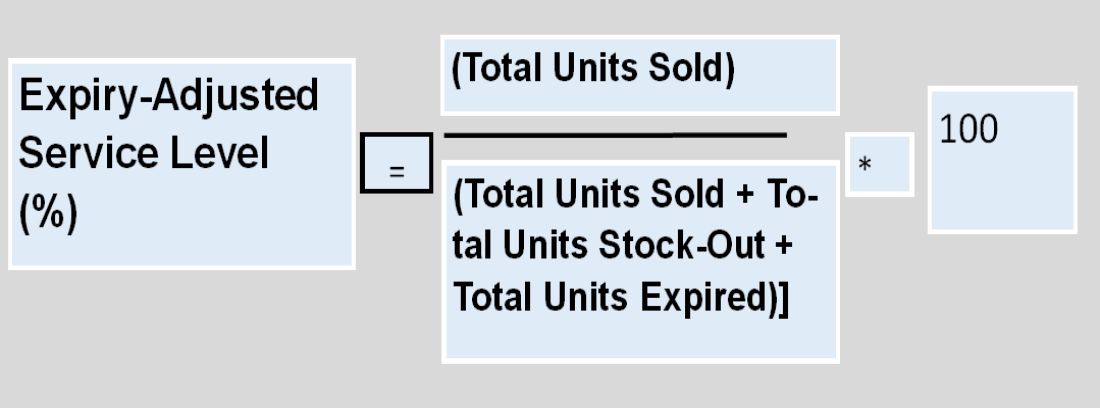
This ensures this AI tool is designed to be a helper and assistive partner to users.

- Achieve user adoption rate where over 90% of generated recommendations are accepted and acted upon by pharmacy staff within 6 months.
- Providing clear, plain language reasoning that links directly to the data (eg. supplier delays, demand spikes)
- Ensure the system reduces the time spent on manual inventory planning and ordering by at least 50%, freeing up staff for patient-facing duties.

Key Performance Indicator:

Medica AI uses a powerful metric that balances the dual goals of preventing drug shortages and minimizing expiries. It modifies the traditional "Service level" metric by penalizing the operation for inventory that expires.

The formula is:



- **Total Units Sold:** The number of units dispensed to patients.
- **Total Units Stock-Out:** The number of units demanded but not available (a proxy for lost sales).
- **Total Units Expired:** The number of units removed from inventory due to expiration.

Why this is the Ideal KPI (Key Performance Indicator):

Directly Tied to Core Mission: It quantitatively measures the system's success at it's primary success at it's primary purpose: getting the right drug to the right patient at the right time, without waste.

Holistic measure: It doesn't allow improving one problem by making the other worse. A pharmacy could achieve a 99% traditional service level by massively over-ordering, but this would lead to high expiries, crashing this new KPI.

It forces Intelligent Optimization: It rewards the AI tool for making nuanced decisions. For example, it might accept a slightly higher risk of a stockout for a very slow-moving , short dated item if the alternative is near-certain expiry.

(A) Sample Pharmacy data (Before Medica AI):

- Units Sold: 9,500
- Units Stock-Out: 500
- Units Expired: 1,000

- **Expiry-Adjusted Service Level** = 9,500 / (9,500 + 500 + 1,000) = 9,500 / 11,000 = **86.4%**

(B) (After Medica AI Implementation):

- Units Sold: 9,800 (more patients served)
- Units Stock-Out: 150 (fewer shortages)
- Units Expired: 200 (dramatically fewer expiries)
- **Expiry-Adjusted Service Level** = 9,800 / (9,800 + 150 + 200) = 9,800 / 10,150 = **96.6%**

The 10.2 percentage point increase vividly demonstrates the Medica AI's success. The pharmacy is now serving more patients while simultaneously drastically reducing both waste and stockouts. This single performance Indicator captures the entire value proposition of the Medica AI.

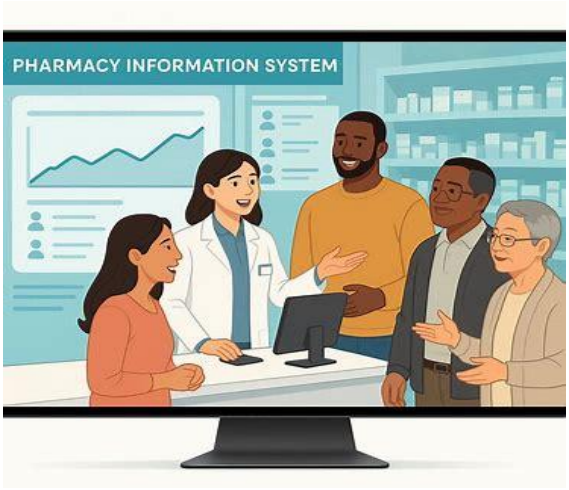
Data sources for Medica AI

To effectively predict demand, calculate risk and prescribe optimal orders, the Medica AI tool must be fed from a variety of internal and external data streams.

1. Internal Data sources. (The core Foundation)

These are the proprietary, high frequency data sources within pharmacy own systems.

(a) Pharmacy Information System/ Inventory Management Systems:



a. **Historical Sales/ Dispensing data:** Date, drug, quantity, prescription ID (anonymized).

Current inventory levels: Real time stock counts for each SKU.

Batch-Level Data: The most critical data for expiry management, including Batch Number, Manufacture Date and Expiration Date.

Purchase Order History: Records of all past orders (dates, quantities, suppliers, costs) and their received dates to calculate supplier lead times.

Supplier Master List: Details of all approved vendors and their performance history.

Electronic Health Records (EHR) – (If accessible and anonymized):

Diagnosis codes: Aggregated and anonymized data on patient diagnoses can provide early signals for disease outbreaks within the community the pharmacy serves.

Prescribing Physician Data: Understanding which doctors are prescribing what can help identify localized trends.

b. External Data Sources (The contextual Intelligence)

These sources provide the external context that turns internal data actionable predictions.

Public Health & Epidemiological data.

Local and State Health Department Reports: Data on reportable diseases, outbreaks (e.g., influenza, malaria, viruses) and public health alerts.

Supply chain & Regulatory Data:

FDA Drug Shortages Database: The authoritative source for manufacturer-reported shortages and discontinuations in the WORLD.

Supplier & Market Data:

Supplier portals: Direct integration with major distributors to pull real time data on their stock levels, allocation status and confirmed lead times.

Price Indexes: Tracks market prices, which can be an indicator of supply tightness or surplus.

News & Social Media feeds (Unstructured Data):

Data Integration and Application Table

Data Source Category	Specific Example	How it Addresses the Problem
Internal: Inventory	Batch Expiration Dates	**Core input for Expiry Risk Score.** Enables FEFO optimization.
Internal: Sales	Historical Dispensing Data	**Trains the demand forecasting model.** Identifies seasonal and trend-based patterns.
External: Regulatory	FDA Shortage Database	**Directly feeds the Shortage Risk Score.** Provides early warning for supply disruption.
External: Public Health	CDC Flu View Report	**Improves demand forecast accuracy** for antivirals, antibiotics, and analgesics.
External: News/Social	News on Factory Closure	**Provides "early warning" signals** for potential shortages long before official alerts.
External: Supplier	Real-time Distributor Stock	**Refines the Optimal Order Recommendation** by showing what is actually available to order.

By strategically integrating these diverse data sources, the Medica AI tool transitions from a simple inventory tracker to predictive and prescriptive intelligence system, capable of navigating the complex realities of the Pharmaceutical supply chain.

Potential Data Biases in the Medica AI Tool.

1. **Digital Footprint Bias:**

This Bias arises from over-reliance on digitally trackable events, missing crucial "offline" context.

Bias Source: Over-dependence on sales and electronic order data.

Manifestation & Impact:

Missing the" Why": The data shows what was dispensed, but not why. A Pharmacist's clinical decision to switch a patient to an alternative due to minor shortage is not recorded. The model might interpret this as reduced demand for the original drug, leading to a future under-order just as the shortage ends and demand returns.

Ignoring Verbal Intelligence: A pharmacist hearing from a sales rep about an impending manufacturing issue is verbal intelligence. This "soft data" is not in any structured feed and would be ignored by the AI, potentially causing it to be blindsided by a shortage.

2. **Algorithmic & Feedback Loop Bias:**

This is a risk created by the AI's own operations.

Self-Fulfilling: If the AI predicts a high risk of shortage for a drug and every pharmaceutical outlet using the system simultaneously increases their orders, this collective action can cause the very shortage it predicted.

Confirmation Bias: The AI model may become over-confident in its predictions. If it starts to deprioritize a drug it deems "low-risk," the resulting lack of sales data for that drug will reinforce its belief that the drug is unimportant, potentially leading to a critical stockout later.

(iii). Outline 3 preprocessing steps (e.g., handling missing data, normalization)

Effective preprocessing is critical to prepare real-world, heterogenous for AI modeling in Medica AI.

1. Temporal Alignment and Aggregation
   Data sources vary widely in granularity—sales are transactional by second, inventory snapshots are daily, external reports may be weekly or monthly, and news feeds are continuous streams. All must be unified on a consistent time scale (e.g., daily or weekly). Transactional sales are aggregated accordingly, and other data aligned to these periods to enable meaningful time-series forecasting and capture causal relationships, such as the impact of flu trends on medicine demand.

2. Domain-Aware Handling of Missing Data
   Missing data is inevitable and context-sensitive. Sales gaps may indicate closures or lack of demand; critical fields like expiration dates cannot be imputed reliably and must be flagged for manual review to preserve model accuracy. External feeds missing values are imputed using time-based carry-forward or flagged as missing to avoid distorting temporal patterns. This approach minimizes bias and prevents unsafe expiry predictions.

3. Normalization, Scaling, and Encoding
   Raw features differ in scale and type—units sold, lead times, prices, categorical drug and supplier IDs, cyclical time features. Robust scaling or standardization normalizes numerical data while being resilient to outliers. Categorical data benefits from target encoding or one-hot encoding to capture demand-specific signals. Cyclical features like day of week are encoded via sine/cosine transforms to preserve circularity. This ensures the model fairly weights feature and generalizes across products, improving prediction reliability.

Chosen Model: Gradient Boosting with XGBoost

Justification for Selection

Although problems of this nature might initially suggest the use of complex neural networks, **XGBoost** (Extreme Gradient Boosting) offers a more effective balance between performance, practicality, and interpretability, making it the most suitable choice for this use case.

1. Exceptional Performance on Tabular Data with Mixed Features
The dataset in this project is primarily structured, containing numerical variables (such as historical sales, lead times, and inventory levels), categorical variables (like supplier ID, drug category, and drug name), and cyclical variables (such as day of the week and month). XGBoost performs exceptionally well with this type of heterogeneous, tabular data. Unlike neural networks that often require complex architectures or embeddings, XGBoost can naturally process preprocessed numerical and categorical inputs (e.g., one-hot encoded features) and efficiently learn nonlinear relationships without extensive feature engineering.

2. Robustness Against Noise and Missing Data
Pharmaceutical sales data often contain noise, outliers (such as one-time bulk purchases), and missing entries. XGBoost's ensemble-based structure, consisting of multiple decision trees, ensures that individual anomalies do not overly influence the model's overall performance. Additionally, it has a built-in mechanism to manage missing values by determining the most suitable direction for data splits, a more advanced and adaptive approach compared to traditional imputation methods.

3. Strong Predictive Power with Limited Data
In scenarios involving new or slow-moving drugs, available historical data may be limited. Unlike deep learning models, which typically require vast datasets, XGBoost performs remarkably well on medium-sized datasets, such as those common in pharmaceutical retail. Its gradient boosting framework efficiently identifies complex patterns and interactions even in relatively small datasets, producing high predictive accuracy without excessive computational or data demands.

4. Interpretability and Transparency
Trust and explainability are critical when working with pharmacists and decision-makers. XGBoost provides clear insights into its predictions through **feature importance** rankings, allowing users to identify which variables—such as average sales, supplier delays, or flu activity—most influence results. Furthermore, tools like **SHAP (SHapley Additive exPlanations)** enable detailed, instance-level explanations. For example, the model can specify that a high shortage risk (e.g., 85%) stems from a three-week supplier delay (+40%), an increase in regional flu cases (+30%), and a 10% decline in inventory (+15%). This transparency fosters trust and ensures that the model's outputs can be confidently used in decision-making.

5. Efficiency and Ease of Deployment
Finally, XGBoost's computational efficiency makes it ideal for production environments. It can rapidly generate predictions for thousands of SKUs, even under cloud-based cost constraints.

Because prediction merely involves traversing a set of decision trees, latency remains low, supporting real-time decision systems.

In summary, **XGBoost** provides the ideal blend of speed, accuracy, robustness, and interpretability, making it the most pragmatic and trustworthy model for pharmaceutical sales forecasting and shortage prediction.

Time Series–Based Data Splitting Strategy

The main principle behind this approach is simple: the model must always be evaluated on data from a future period relative to the training period. This setup mimics real-world forecasting, where we use historical information to predict upcoming outcomes.

To achieve this, a **nested, time-based split with a rolling origin** is applied — a best-practice method for time series forecasting tasks.

Step 1: Arrange Data Chronologically
All records must first be ordered by time. For every (pharmacy, drug) combination, weekly or periodic observations should appear in correct chronological order to maintain the temporal sequence.

Step 2: Create the Main Splits (Train, Validation, Test)
The timeline is divided into three key segments:

- **Training Set (≈60–70%)**: The earliest data used to learn trends, seasonality, and overall patterns.
- **Validation Set (≈15–20%)**: The next segment, used for model selection and hyperparameter tuning (e.g., XGBoost's *max_depth* or *learning_rate*).
- **Test Set (≈15–20%)**: The most recent data, reserved for the final evaluation to measure true future performance.

Timeline Example:
```
[---Training (60%)---][---Validation (20%)---][---Test (20%)---]
```

Step 3: Apply a Rolling (Nested) Validation Approach
A static validation set may not capture real-world variability, so a **rolling origin** method is used within the training and validation segments. The model is repeatedly trained on earlier data and validated on successive time windows to simulate progressive retraining.

Why This Approach Matters for Medica AI
This method prevents **data leakage** by ensuring future information never influences past predictions. It also tests **temporal generalization**, answering whether the model can accurately forecast upcoming demand based on historical patterns. Additionally, it detects **concept drift**, highlighting if model accuracy declines due to market changes or new external factors.

By applying this structured, time-aware validation strategy, the **Medica AI** can deliver reliable, real-world performance in predicting drug shortages and expirations.

Hyperparameters used:

In the Medica AI system, tuning **XGBoost hyperparameters** is essential to achieve a balance between model complexity, predictive accuracy, and generalization. Two of the most critical parameters to optimize are **`max_depth`** and **`scale_pos_weight`**, both of which directly influence model performance and reliability.

1. max_depth

- **Definition:** Determines how deep each decision tree in the ensemble can grow. Deeper trees can represent more complex feature interactions.
- **Purpose:** Controls the trade-off between underfitting and overfitting.
- **If set too high:** The model becomes overly complex and may capture random noise in the data—such as isolated sales spikes—leading to poor generalization on new, unseen data.
- **If set too low:** The model becomes too simplistic and fails to capture key feature relationships, like the combined effect of supplier delays and seasonal demand fluctuations.
- **Goal for Medica AI:** Identify a moderate `max_depth` value that captures meaningful patterns in drug demand and supply risk without memorizing irrelevant historical noise.

2. scale_pos_weight

- **Definition:** Adjusts the balance between positive and negative classes in classification problems, such as predicting shortage or expiry risks.
- **Purpose:** Compensates for class imbalance, where "No Risk" instances far outnumber actual "High Risk" events.
- **Without tuning:** The model might default to predicting "No Risk" for most cases, appearing accurate but failing to detect critical shortage or expiry events.
- **Goal for Medica AI:** Increase the weight of the minority "Risk" class, ensuring the model pays closer attention to these rare yet high-impact occurrences. This helps improve recall and ensures the system effectively flags potential shortages or expiries before they occur.

By thoughtfully tuning **`max_depth`** and **`scale_pos_weight`**, Medica AI's XGBoost model can achieve a strong balance between accuracy and practicality—delivering reliable predictions that help prevent costly supply chain disruptions.

4. Evaluation & deployment

Evaluation Metrics:

Given the critical and imbalanced nature of pharmaceutical supply chain data, the evaluation of the **Medica AI** must focus on detecting rare but high-impact events rather than achieving high overall accuracy. Two metrics—**F2-Score** and **Pinball Loss**—are especially effective for this purpose.

1. Shortage and Expiry Risk Classification – F2-Score

- **Definition:**
  The **F2-Score** is a weighted harmonic mean of **Precision** and **Recall**, placing greater emphasis on Recall. It is calculated as:

$$F2 = \frac{5 \times (\text{Precision} \times \text{Recall})}{4 \times \text{Precision} + \text{Recall}}$$

  It Matters Because:

  - **Emphasizes Critical Errors:** In this context, a *False Negative* (predicting "No Risk" when a shortage or expiry actually happens) is far more damaging than a *False Positive*. The F2-Score penalizes such missed detections more heavily, ensuring the model prioritizes capturing true risk events.
  - **Balances Recall and Precision:** While it's crucial to identify every possible shortage or expiry (high Recall), too many false alerts can erode user confidence. The F2-Score maintains a balance, valuing Recall twice as much as Precision for optimal performance.
  - **Suitable for Imbalanced Data:** Since shortage and expiry cases are relatively rare, accuracy can be misleading. The F2-Score provides a more reliable assessment of model effectiveness in these scenarios.

2. Demand Forecasting – Pinball Loss (Quantile Loss)

- **Definition:**
  Pinball Loss evaluates the accuracy of quantile forecasts, which estimate a range of possible outcomes (e.g., 10th to 90th percentile) rather than a single value.
- **Why It Matters:**
  - **Improves Inventory Management:** Forecasting errors carry asymmetric costs. Overestimations can lead to wastage and expired stock, while underestimations can cause stockouts and lost sales. Pinball Loss helps optimize forecasts to reduce the total cost of these errors.
  - **Supports Risk-Based Decisions:** By predicting demand intervals (e.g., "There's an 80% probability demand will fall between 70 and 130 units"), managers can

make informed decisions based on their risk tolerance. Hospitals may prefer safer upper estimates, while retailers might choose more cost-efficient midpoints.
    - **Measures Predictive Reliability:** A lower Pinball Loss means not only accurate predictions but also trustworthy uncertainty intervals—crucial for generating dependable risk assessments and guiding stock management decisions.

Concept drift? Monitoring post-deployment?

Concept drift occurs when the relationship between input features and the target variable changes over time, reducing a model's predictive accuracy. In **Medica AI**, this means that patterns linking sales history, supplier lead times, or flu trends to outcomes like demand, shortage, or expiry risk evolve, making the model's earlier learning less reliable.

Examples of Concept Drift in the Medica AI:

- **Sudden Drift:** A pandemic changes prescription habits and supply reliability, making pre-pandemic data obsolete.
- **Gradual Drift:** A new pharmacy gradually reduces local demand, causing consistent overestimation.
- **Seasonal Drift:** An unusually severe flu season drives demand far beyond historical trends.

Monitoring Concept Drift Post-Deployment

To maintain model reliability, the PCO should use a **multi-layered drift monitoring system** that includes:

1. **Performance Monitoring:**
    - Regularly track F2-Score (for risk classification) and Pinball Loss (for demand forecasting) over recent data.
    - Compare results to baseline validation metrics.
    - If F2-Score drops by >15% or Pinball Loss rises sharply, it indicates drift and the need for retraining.
2. **Data Drift Monitoring:**
    - Detect early signs of change by comparing current feature distributions with training data using tests such as:
        - **KS Test** (continuous data)
        - **PSI** (overall distribution stability)
        - **Chi-Square Test** (categorical data)

    - Watch for shifts in feature importance or prediction distributions as early warnings.
3. **“Canary” Drugs:**
    - Monitor a few representative drugs (e.g., fast-moving, specialty, OTC).
    - If predictions for these drugs become unreliable, it suggests broader concept drift.

Monitoring and Retraining Cycle:
Once drift is detected:

1. **Alert and Diagnose** the cause (e.g., new supplier, demand shift).
2. **Retrain** the model using recent data.
3. **Validate** improvements on a hold-out set.
4. **Redeploy** the updated model.

By continuously monitoring data and performance, **Medica AI** ensures the Medica AI stays adaptive, accurate, and trustworthy—effectively responding to evolving pharmaceutical market dynamics.

Describe 1 technical challenge during deployment.

A key technical challenge during deployment is **scalability**. The AI tool must process vast, continuously updating data—from inventory, suppliers, and hospitals—while generating timely predictions. Handling thousands of drugs in real time requires optimized data pipelines, efficient model serving, and auto-scaling infrastructure to prevent latency, bottlenecks, or outdated shortage forecasts.

Part 2: Case Study Application

Problem Definition:
Unplanned patient readmissions within 30 days of discharge pose a major challenge in healthcare, serving as both a key quality indicator and a source of financial strain. These readmissions often reflect shortcomings in discharge planning, follow-up care, or unresolved medical issues. The central goal is to identify high-risk patients *before* they are readmitted, allowing healthcare providers to intervene early, enhance recovery, and reduce unnecessary hospitalizations.

Objectives:

- **Accurate Prediction:** Build an AI model capable of precisely identifying patients at high risk of 30-day readmission using relevant clinical and demographic data.
- **Actionable Intervention:** Integrate predictive insights into clinical workflows to prompt timely actions—such as follow-up calls, medication reviews, or coordinated home care.
- **Resource Optimization:** Enable hospitals to deploy care management resources efficiently, focusing on patients most likely to benefit from targeted support.

Stakeholders:

- **Primary:** Patients and clinical staff (doctors, nurses, and case managers) who use the insights to guide care decisions.
- **Secondary:** Hospital administrators and insurers who benefit from reduced penalties and improved cost-efficiency.
- **Tertiary:** Families and public health systems that gain from reduced hospital congestion and better overall patient outcomes.

Data strategy:

Proposed Data Sources

Electronic Health Records (EHRs):

- Patient demographics (age, gender, location)
- Medical history and comorbidities (e.g., Charlson Comorbidity Index)
- Current admission details (diagnosis, length of stay, ICU admission)
- Vital signs, lab results, and medication lists
- Procedures conducted during hospitalization
- Discharge disposition (home, rehabilitation, or nursing facility)

Claims and Billing Data:

- Insurance coverage and type
- Past healthcare utilization (hospitalizations, ED visits)
- Historical billing and admission records

Social Determinants of Health (SDOH):

- Community-level census data (income, education, poverty rates)
- Housing and transportation stability
- Access to community healthcare resources

Operational and Workflow Data:

- Care team composition and staffing ratios
- Completeness of discharge documentation
- Follow-up appointment scheduling and adherence

Ethical Considerations

Algorithmic Bias and Equity:
The model may unintentionally overestimate or underestimate risk for certain demographic or socioeconomic groups, perpetuating health inequities.
Mitigation: Perform fairness audits across subgroups, ensure diverse data representation, apply causal inference to control for confounding social factors, and carefully manage sensitive variables like race or zip code.

Clinical Over-Reliance:
Clinicians might rely too heavily on AI predictions, neglecting clinical judgment for patients flagged as “low-risk.”

Mitigation: Keep the AI as a decision-support tool, mandate partial manual review, include interpretable explanations for risk scores, and train users on limitations and appropriate use.

Preprocessing Pipeline Design

Step 1 – Data Integration & Target Definition:
Merge EHR, claims, and SDOH data by patient ID; ensure only discharge-time data is used. Define the binary target variable: *readmitted_within_30_days (1/0).*

Step 2 – Handling Missing Data:

- Apply MICE for clinical data imputation.
- Use geographic averages for missing SDOH values.
- Flag records missing key vitals or labs for review.

Step 3 – Feature Engineering:

- *Clinical Complexity:* number of medications, comorbidities, lab variability, and vital trends.
- *Healthcare Utilization:* recent admissions, ED visits, and time since last hospitalization.
- *Social Complexity:* isolation index, transportation burden, and local pharmacy density.
- *Care Transition:* discharge completeness, delay between discharge order and execution, and follow-up timing.

Step 4 – Feature Selection & Transformation:
Remove redundant features, handle imbalance with SMOTE, normalize skewed data using quantile transformation, and one-hot encode categorical variables.

Step 5 – Monitoring & Feedback Loop:
Automate data quality checks, integrate updated readmission outcomes, and retrain the model periodically based on performance drift.

Reflection (5 points):

- **What was the most challenging part of the workflow? Why?**
- **How would you improve your approach with more time/resources?**

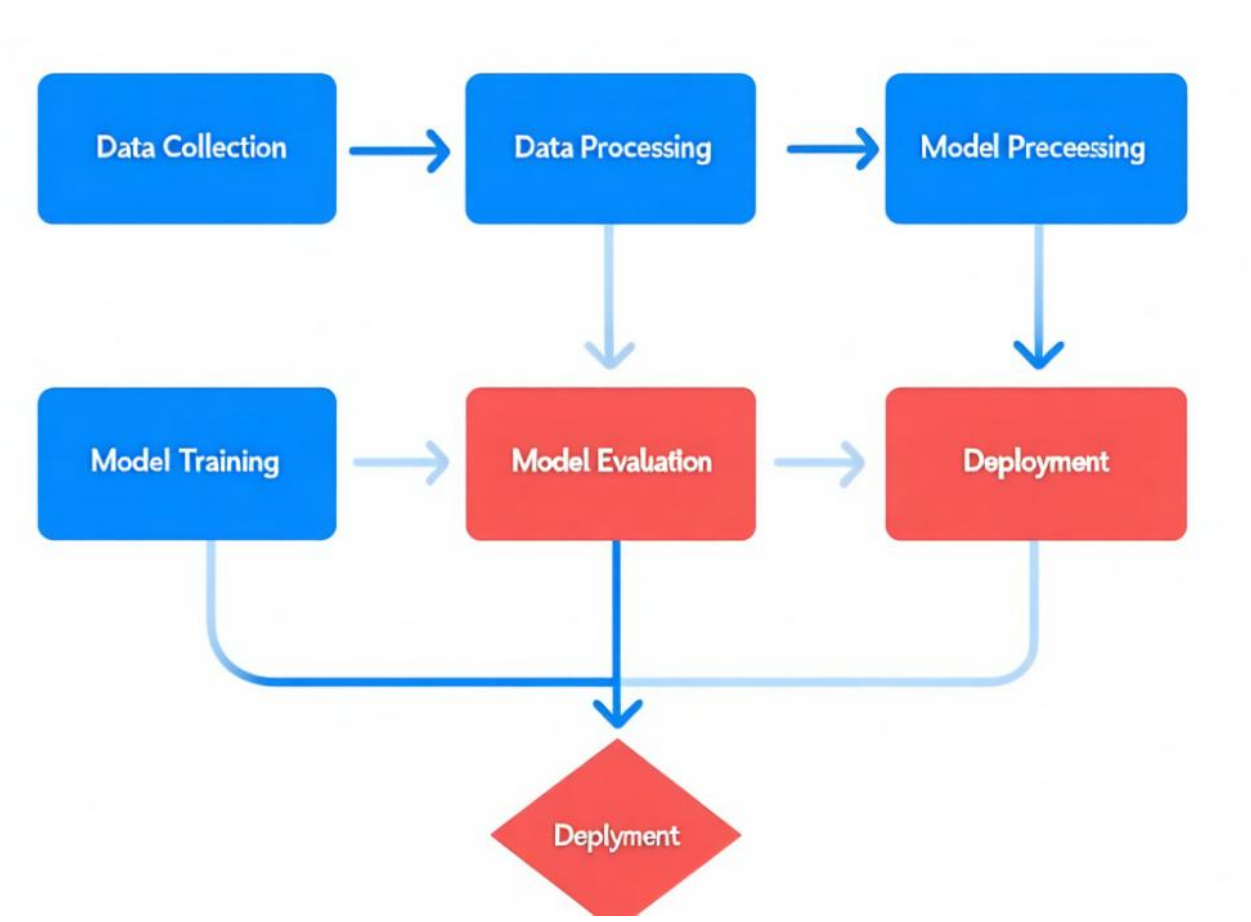
The most difficult aspect of the workflow is managing **ethical challenges**, especially **algorithmic bias** and **clinical deskilling**. Unlike technical tasks such as data integration or feature engineering, ethical concerns lack clear-cut solutions. Algorithmic bias was particularly complex because it stems from systemic inequalities embedded in historical data. Simply removing sensitive variables like race doesn't eliminate bias—proxies such as zip code or insurance type can still reflect inequities. The challenge was building a model that detects real clinical and social risks while avoiding discrimination. Achieving fairness required constant auditing, interpretation, and adjustment to balance accuracy with ethical responsibility.

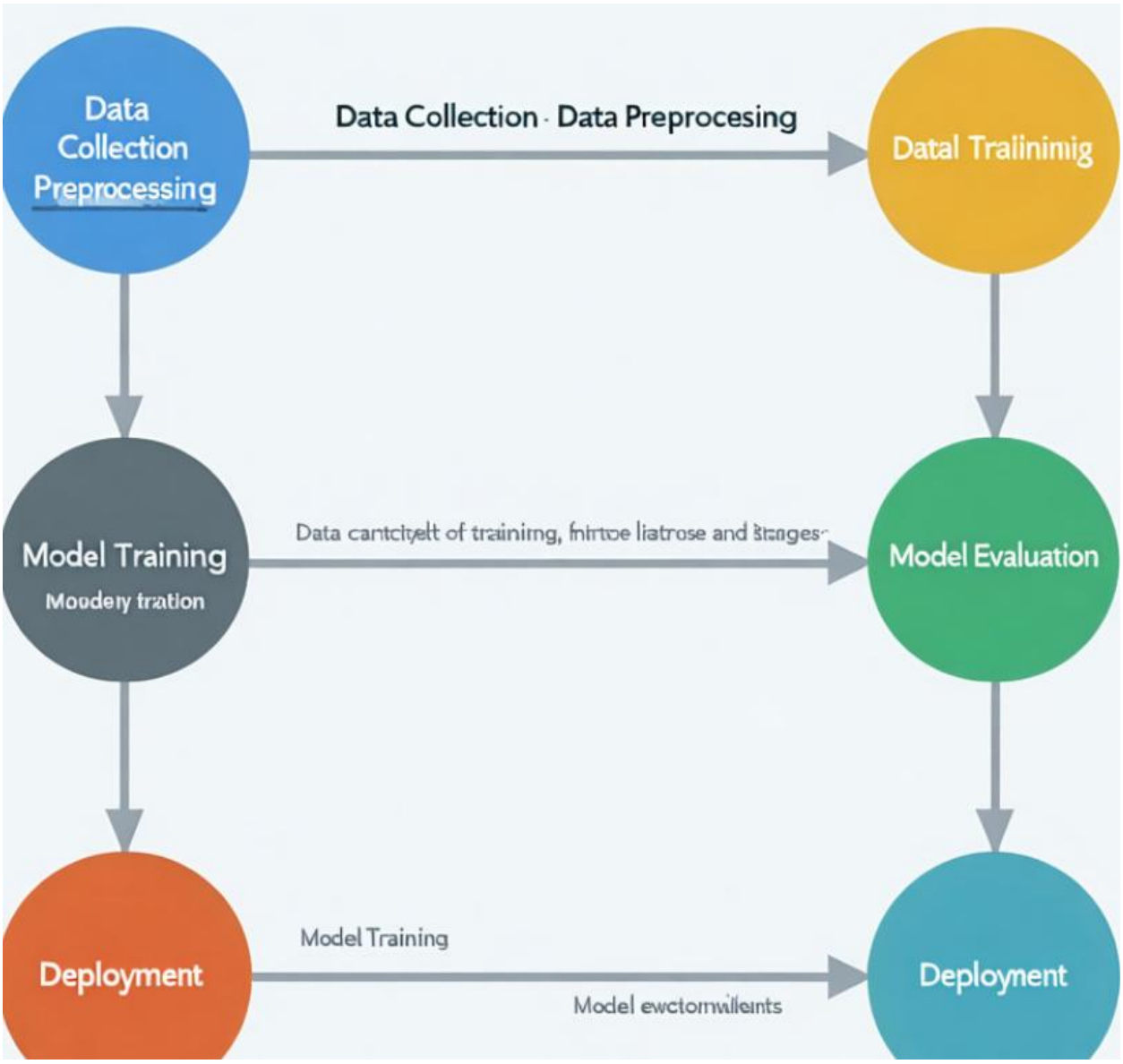
Preventing clinical deskilling was another challenge. Over-reliance on AI can erode clinicians' judgment if they trust the system blindly. Ensuring the model remained a **decision-support tool** demanded thoughtful design, clear model explanations, and ongoing staff training.

If given more time and resources, I would enhance the system in three key ways:

1. **Data Fidelity:** Integrate richer, real-time data from community partners, wearables, and patient-reported outcomes for a more complete recovery view.
2. **Causal and Explainable AI:** Use causal modeling and explainable AI to clarify *why* patients are at risk and guide actionable interventions.
3. **Real-World Validation:** Conduct longer A/B trials and build digital simulations to test recommendations safely before full deployment.
4. **Small funding to expand.** In making the idea a reality.

1. **Diagram (5 points):**
    - **Sketch a flowchart of the AI Development Workflow, labeling all stages**.




Data Collection Preprocessing
Data Collection · Data Preprocesing
Datal Trailinimig
Model Training
Model Evaluation
Deployment
Model Training
Deployment