**HOSPITAL READMISSION PREDICTION: AI PROJECT FINAL REPORT**

Team Members: 5-member collaborative project

Project: AI System for Predicting Patient Readmission Risk

Date: July 2025

**Table of Contents**

1. Executive Summary

2. Part 1: Short Answer Questions

3. Part 2: Case Study Application

4. Part 3: Critical Thinking

5. Part 4: Reflection & Workflow

6. Appendices

**Executive Summary**

This project addresses the critical healthcare challenge of predicting patient readmission risk within 30 days of discharge. Our team developed a comprehensive AI solution that combines advanced machine learning techniques with healthcare domain expertise to create a practical, deployable system.

**Key Achievements:**

- Problem Definition: Clearly defined scope with measurable objectives and stakeholder identification

- Data Strategy: Comprehensive approach to data collection, preprocessing, and ethical considerations

- Model Development: Robust machine learning pipeline with appropriate evaluation metrics

- Deployment Planning: HIPAA-compliant integration strategy with monitoring protocols

- Quality Assurance: Complete workflow documentation with reflection and improvement strategies

Expected Impact: 15% reduction in 30-day readmission rates, improved patient outcomes, and significant cost savings for healthcare institutions.

Part 1: Short Answer Questions

1. Problem Definition

Hypothetical AI Problem: Predicting Hospital Readmission Risk

Three Objectives:

1. Primary: Develop a machine learning model achieving >80% recall for identifying high-risk patients

2. Secondary: Create an interpretable model providing actionable clinical insights

3. Operational: Integrate seamlessly into existing hospital workflows with minimal disruption

Two Stakeholders:

1. Healthcare Providers (Doctors, Nurses): Need accurate predictions for clinical decision-making

2. Hospital Administrators: Require system to reduce readmission rates and associated costs

Key Performance Indicator (KPI): F1-Score > 0.75 (balancing precision and recall for healthcare context)

2. Data Collection & Pre-processing

Two Data Sources:

1. Electronic Health Records (EHR): Patient demographics, medical history, treatment data

2. Hospital Information Systems: Admission/discharge data, length of stay, procedures performed

Potential Bias in Data:

- Socioeconomic Bias: Patients from lower-income areas may have limited access to follow-up care, leading to higher readmission rates that could bias the model against these populations

Three Pre-processing Steps:

1. Missing Data Handling: Use median imputation for numerical variables, mode for categorical variables

2. Feature Normalization: Apply Standard Scaler to numerical features to ensure equal weight consideration

3. Categorical Encoding: Use one-hot encoding for medical specialties and diagnosis codes

3. Model Development (8 points)

Chosen Model: Random Forest Classifier

Justification:

- Handles both numerical and categorical data effectively

- Provides feature importance for clinical interpretability

- Robust to overfitting with healthcare datasets

- Can capture non-linear relationships in patient data

Data Splitting Strategy:

- Training Set: 70% of data for model training

- Validation Set: 15% for hyperparameter tuning

- Test Set: 15% for final performance evaluation

Two Hyperparameters to Tune:

1. n\_estimators (100-500): Controls number of trees, affects model complexity and performance

2. max\_depth (5-20): Prevents overfitting by limiting tree depth, crucial for generalization

4. Evaluation & Deployment (8 points)

Two Evaluation Metrics:

1. Recall: Critical for healthcare as missing high-risk patients (false negatives) has severe consequences

2. F1-Score: Balances precision and recall, suitable for imbalanced healthcare datasets

Concept Drift: Changes in patient demographics, treatment protocols, or hospital policies that affect model performance over time.

Monitoring Post-Deployment:

- Weekly performance metric tracking

- Automated alerts for significant performance drops

- Periodic model retraining with fresh data

Technical Challenge - Scalability:

- Managing real-time prediction requests as hospital data volume grows

- Ensuring computational resources can handle increased load during peak times

- Implementing efficient data pipelines for continuous model updates

Part 2: Case Study Application

Problem Scope (5 points)

Problem Definition: The hospital requires an AI system to predict patient readmission risk within 30 days of discharge. This predictive model will help healthcare providers identify high-risk patients who may need additional care, follow-up appointments, or modified discharge plans to reduce readmission rates.

Objectives:

1. Primary Objective: Develop a machine learning model that accurately predicts 30-day readmission risk with at least 80% recall

2. Secondary Objective: Create an interpretable model that provides actionable insights for healthcare providers

3. Operational Objective: Integrate the prediction system seamlessly into existing hospital workflows

Stakeholders:

- Primary: Healthcare Providers, Hospital Administrators, Patients

- Secondary: Data Science Team, IT Department, Insurance Providers, Regulatory Bodies

Data Strategy (10 points)

Data Sources:

1. Electronic Health Records (EHRs): Patient demographics, medical history, medications

2. Hospital Administrative Data: Length of stay, procedures, discharge information

3. Demographic Data: Age, gender, socioeconomic factors

4. Clinical Data: Lab results, vital signs, diagnosis codes

Two Ethical Concerns:

1. Patient Privacy: Ensuring HIPAA compliance and secure handling of sensitive health information

2. Algorithmic Bias: Preventing discrimination based on race, ethnicity, or socioeconomic status

Preprocessing Pipeline:

1. Data Cleaning: Remove duplicates, handle missing values, validate data integrity

2. Feature Engineering: Create interaction terms, aggregate temporal features, encode categorical variables

3. Data Validation: Check for outliers, validate clinical ranges, ensure data quality

Model Development (10 points)

Selected Model: Random Forest Classifier

Justification:

- Excellent performance on healthcare datasets

- Provides feature importance for clinical interpretability

- Handles mixed data types effectively

- Robust to overfitting

Deployment

Integration Steps:

1. API Development: Create RESTful API using Flask/FastAPI

2. EHR Integration: Connect with hospital's existing electronic health record system

3. User Interface: Develop clinician dashboard for risk visualization

4. Automated Triggers: Set up prediction generation after discharge data entry

5. Audit Trail: Implement comprehensive logging for compliance and monitoring

HIPAA Compliance:

- Data Encryption: Use HTTPS for all data transmission

- Access Controls: Implement role-based access with authentication

- Audit Logging: Track all data access and model predictions

- Secure Storage: Use HIPAA-compliant cloud services (AWS HealthLake, Google Cloud Healthcare)

Optimization

Overfitting Mitigation Method: Cross-validation with k-fold (k=5) to ensure model generalizes well to unseen data, combined with early stopping criteria and regularization techniques.

Part 3: Critical Thinking

Ethics & Bias

Impact of Biased Training Data:

Biased training data could lead to systematic underprediction of readmission risk for certain demographic groups, potentially resulting in inadequate follow-up care and worse health outcomes for these patients.

Mitigation Strategy:

Implement fairness-aware machine learning techniques, including:

- Regular bias audits using demographic parity metrics

- Diverse training data collection from multiple hospitals

- Algorithmic fairness constraints during model training

- Continuous monitoring of prediction disparities across demographic groups

Trade-offs

Interpretability vs. Accuracy:

In healthcare, interpretability is crucial for clinical adoption, even if it means slightly lower accuracy. Clinicians need to understand model reasoning to trust and act on predictions. Random Forest provides a good balance, offering feature importance while maintaining competitive performance.

Limited Computational Resources Impact:

With limited resources, simpler models (logistic regression, decision trees) become more attractive despite potentially lower accuracy. The trade-off involves balancing model complexity with available computational power and maintenance requirements.

Part 4: Reflection & Workflow

Reflection

Most Challenging Aspect:

Ensuring seamless integration and consistency across all team members' contributions. Coordinating diverse expertise, maintaining technical consistency, and balancing individual contributions with overall document quality presented significant challenges.

Improvements with More Resources:

- Enhanced collaboration tools and standardized templates

- Quality assurance processes with peer review cycles

- Technical infrastructure for shared repositories

- Expert consultation for clinical and technical validation

Workflow Diagram

The AI Development Lifecycle includes 13 key stages:

1. Problem Definition → 2. Data Collection → 3. Data Pre-processing → 4. Feature Engineering → 5. Model Development → 6. Model Training → 7. Model Evaluation → 8. Model Validation → 9. Deployment Planning → 10. System Integration → 11. Production Deployment → 12. Monitoring & Maintenance → 13. Model Retraining

Key Decision Points:

- Performance gates ensure quality standards

- Drift detection triggers model updates

- Quality assurance checkpoints maintain project integrity

Appendices

A. Team Member Contributions

- Member 1: Problem framing and data strategy

- Member 2: Model development and training

- Member 3: Evaluation, deployment, and monitoring

- Member 4: Critical thinking and ethical considerations

- Member 5: Final compilation, reflection, and workflow diagram

B. Technical Specifications

- Dataset: 25,001 patient records with 17 features

- Target Variable: Binary readmission classification

- Model: Random Forest Classifier

- Performance Targets: F1-Score > 0.75, Recall > 0.80

C. Deployment Architecture

- API Framework: Flask/FastAPI

- Database: HIPAA-compliant cloud storage

- Monitoring: Automated performance tracking

- Security: Role-based access control, encryption