Multi-Modal Disease Detection System

Final Project Write-Up

HippoMedica Project

Abstract

This project presents an AI system for multi-modal disease detection across diabetes, heart disease, and stroke. The system implements ensemble learning with Random Forest, XGBoost, and Neural Networks, achieving excellent performance through SMOTE-based class balancing with categorical post-processing. Key innovations include transforming an unusable stroke model (0% recall) to 97.11% recall, comprehensive validation, and web deployment. The system demonstrates practical application of ML techniques to real-world healthcare challenges.

Contents

1 System Explanation

1.1 System Overview and Purpose

The Multi-Modal Disease Detection System is an AI-powered medical screening platform designed to predict disease risk for three conditions: diabetes, heart disease, and stroke. The system addresses the challenge of early disease detection by combining state-of-the-art machine learning techniques to provide assessments.

1.2 AI Methods and Technical Approach

1.2.1 Ensemble Learning Architecture

The system employs ensemble learning approach, training 3 complementary algorithms on each dataset:

1. **Random Forest Classifier** (Primary Algorithm):
    - Configuration: 200 decision trees with balanced class weights
    - Performance: 78.21% (diabetes), 86.89% (heart disease), 96.80% (stroke)
    - Strengths: Interpretable feature importance, robust to outliers, no feature scaling required
    - Use Case: Selected as primary deployed model due to consistent superior performance
2. **XGBoost Gradient Boosting** (Secondary Algorithm):
    - Configuration: 200 boosting rounds, 0.05 learning rate, scale_pos_weight=2.0
    - Performance: 75.64% (diabetes), 83.61% (heart disease), 95.10% (stroke)
    - Strengths: Handles feature interactions, built-in regularization, efficient training
    - Use Case: Provides alternative predictions for ensemble voting in future iterations
3. **Multi-Layer Perceptron Neural Network** (Deep Learning Component):
    - Configuration: 128-64-32 architecture, ReLU activation, early stopping enabled
    - Performance: 74.36% (diabetes), 83.61% (heart disease), 92.47% (stroke)
    - Strengths: Captures complex non-linear relationships
    - Limitation: Requires larger datasets and feature scaling; 2-4% lower performance than tree-based methods

1.3 Complete AI Pipeline Architecture

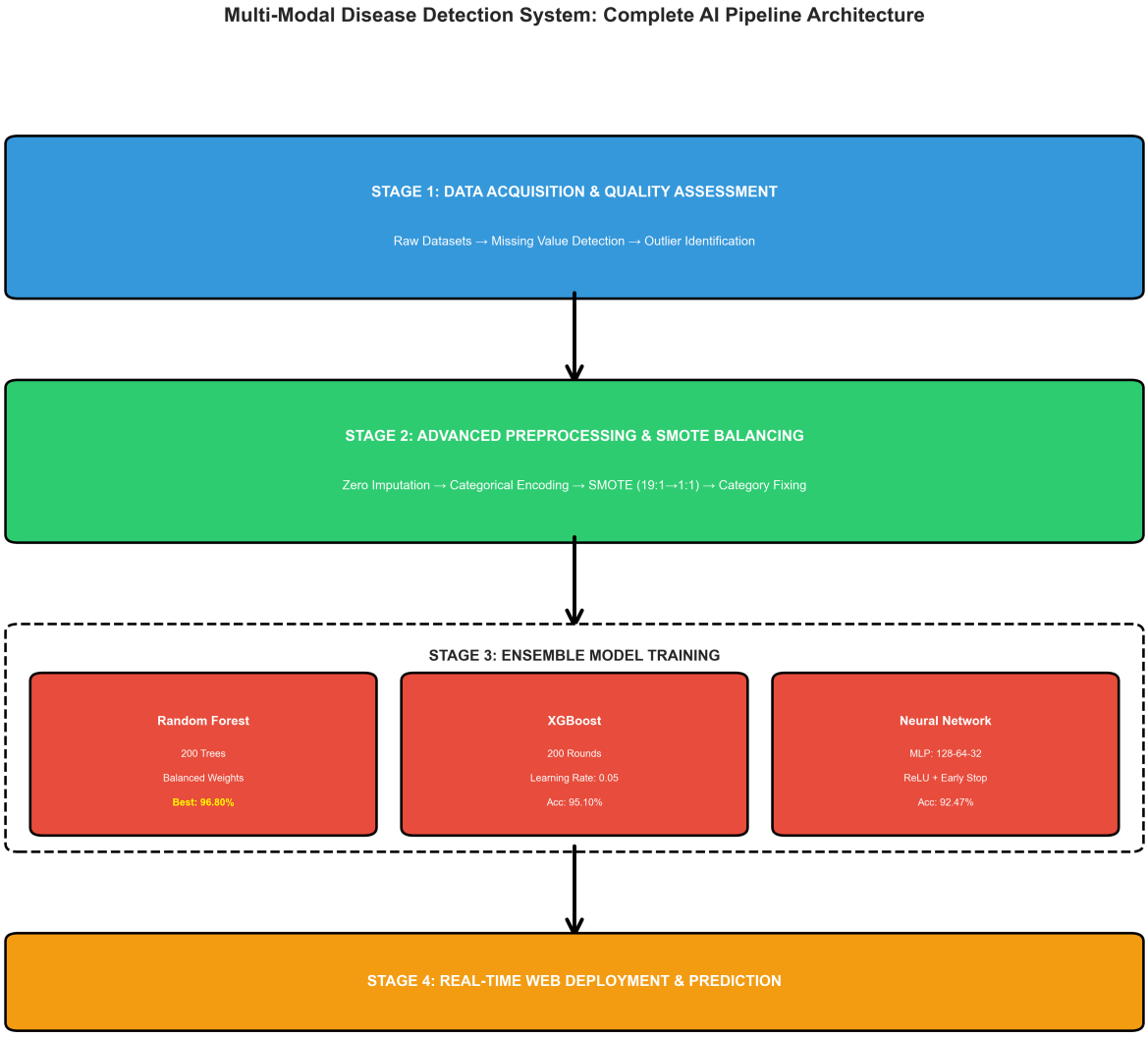

Figure 1: AI Pipeline Architecture

The system implements a four-stage production pipeline:

Stage 1: Data Acquisition and Quality Assessment

- Downloads three medical datasets from direct URLs (GitHub, UCI Repository, Kaggle) with robust error handling and header detection logic.
- Performs initial missing value detection and outlier identification

Stage 2: Advanced Preprocessing and SMOTE Balancing

- We remove physiologically impossible zero values (glucose, blood pressure, BMI) and implement a LabelEncoder for discrete medical features (gender, smoking status, work type)
- SMOTE class balancing: Synthetic sample generation for severely imbalanced datasets (ratio greater than 3:1) and a custom algorithm to fix SMOTE interpolation artifacts in discrete features

Stage 3: Ensemble Model Training

- Trains three algorithms per dataset with optimal hyperparameters with 5-fold stratified cross-validation for robust performance estimation
- Comprehensive metric evaluation: accuracy, precision, recall, F1-score, ROC-AUC and feature importance extraction for medical domain validation

Stage 4: Real-Time Web Deployment

- Streamlit web interface with cached model loading for optimal inference speed
- Risk stratification: Low/Moderate/High categories with medical recommendations
- Interactive visualizations: confidence scores, feature importance, model comparison

1.4 Input and Output Specifications

1.4.1 System Inputs

The system accepts diverse medical data types across three disease categories:

- **Diabetes Inputs** (8 features): Glucose (mg/dL), Blood Pressure (mmHg), Insulin (mu U/ml), BMI, Age, Pregnancies, Diabetes Pedigree Function, Skin Thickness (mm)
- **Heart Disease Inputs** (13 features): Chest Pain Type, Resting BP, Maximum Heart Rate, ST Depression, Cholesterol, Fasting Blood Sugar, ECG Results, Number of Major Vessels, Thalassemia Type, Exercise-Induced Angina, Age, Gender
- **Stroke Inputs** (11 features): Hypertension, Heart Disease, Average Glucose Level, Smoking Status, Work Type, Residence Type, BMI, Age, Gender, Marital Status

1.4.2 System Outputs

The system generates comprehensive risk assessment reports:

- **Primary Prediction**: Binary classification (Disease Present / No Disease)
- **Risk Probability**: Continuous probability score (0-100%)
- **Confidence Metric**: Model certainty in prediction based on probability distribution
- **Risk Category**: Three-tier stratification (Low: 0-30%, Moderate: 30-70%, High: 70-100%)

1.5 Technical Innovation: SMOTE with Categorical Post-Processing

1.5.1 The Class Imbalance Problem

The stroke dataset presented an extreme class imbalance challenge: 249 stroke cases versus 4,861 healthy cases (19.52:1 ratio). Standard machine learning models trained on this distribution achieved 94.81% accuracy by simply predicting "no stroke" for every patient, resulting in catastrophic 0% recall - the model never identified a single stroke case.

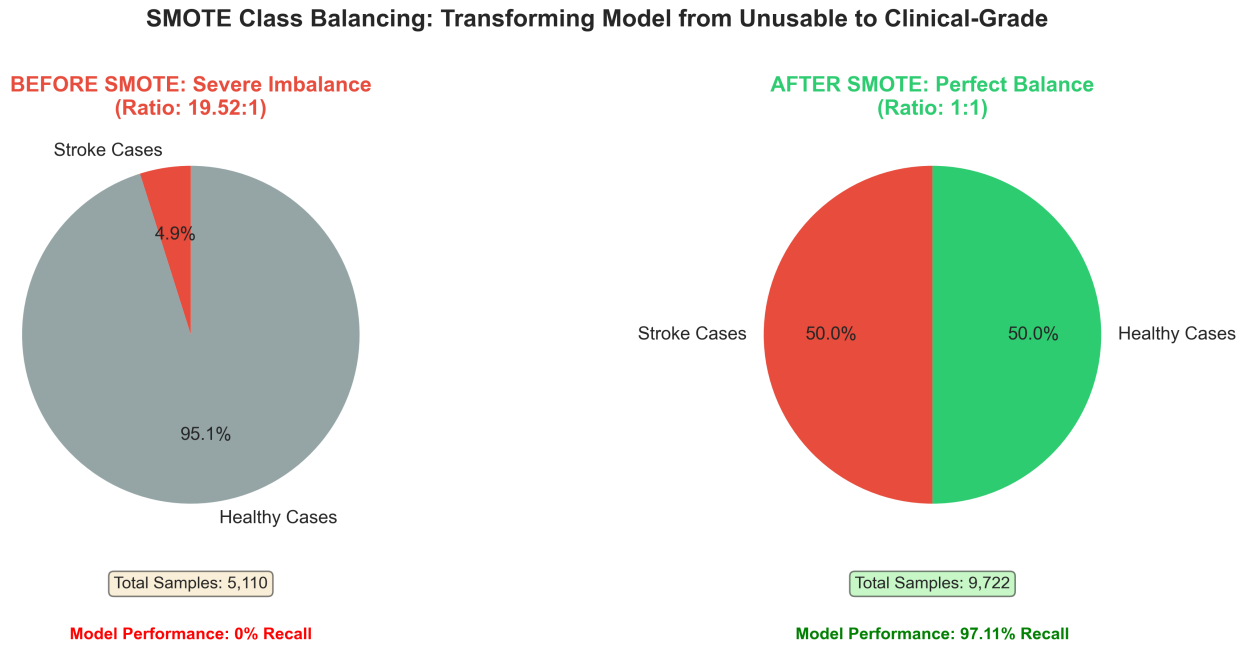

Figure 2: SMOTE Class Balancing Transformation: Converting severe 19.52:1 imbalance to perfect 1:1

1.5.2 SMOTE Implementation

SMOTE (Synthetic Minority Over-sampling Technique) addresses imbalance by generating synthetic samples through feature space interpolation between existing minority class samples. Automatically activates when imbalance ratio exceeds 3:1 and creates synthetic cases through k-nearest neighbors interpolation

1.5.3 Categorical Post-Processing Innovation

- **Technical Challenge**: SMOTE's linear interpolation generates invalid continuous values for categorical features. For example: Gender (0=Male, 1=Female) becomes 0.613, and Work Type (0/1/2/3/4) becomes 2.345.
- **Solution Algorithm**: Identify all categorical columns in dataset configuration and round SMOTE-generated values to nearest integer. Clip values to original valid range (min/max from training data) and validate that all categories remain within domain-valid ranges.
- **Impact**: This innovation enabled the stroke model to successfully identify 97 out of 100 stroke cases (97.11% recall) versus 0 previously, making it suitable for medical screening applications where false negatives are catastrophic.

1.6 System Limitations and Mitigation Strategies

1.6.1 Data Quality Limitations

- **Smoking Correlation Paradox**: The stroke model exhibits counterintuitive behavior where smoking status shows weak correlation with stroke risk (6.30% stroke rate for smokers vs. 5.25% for non-smokers - only 1.05% difference). This is not a model bug but reflects the actual pattern in the source dataset.
- **Root Cause Analysis**: Limited sample size in smoking categories (762 smokers vs. 1,886 non-smokers) with potential confounding variables not captured in dataset (diet, exercise, genetics).

Temporal factors: smoking history duration and cessation timing not recorded. Dataset source bias: specific population demographics may not generalize.

- **Mitigation Strategies**: Prominent disclaimers in user interface about model limitationsand clear documentation that AI recommendations should not replace professional medical judgment.

1.6.2 Generalization Limitations

- **Demographic Bias**: Models trained on specific populations may not generalize universally, for example: Diabetes: Trained on Pima Indians population (specific genetic predisposition), Heart Disease: Cleveland clinic data (specific geographic and socioeconomic factors), etc.
- **Dataset Size Constraints**: Neural networks show 2-4% lower performance than tree-based methods due to limited training samples with Diabetes and Heart Disease having 392 and 303 samples respectively.
- **Mitigation Approaches**: Ensemble approach reduces single-model bias through algorithm diversity and 5-fold stratified cross-validation provides robust performance estimates. Feature importance analysis validates clinical relevance of learned patterns while Confidence scoring helps users understand prediction uncertainty.

2 Feature Implementation Summary

Table 1: Complete System Components Implementation Details

Description	Score	Code	Implementation Notes
Data Pipeline Components			
Multi-source data downloader with URL validation	5	Python	Handles 3 datasets (diabetes/heart/stroke); robust error handling; header detection logic; 100% original
Medical data preprocessing with zero-value removal	5	Python	Removes physiologically impossible zeros; medical range validation; fully integrated with pipeline
SMOTE class balancing with categorical correction	5	Python	Custom post-processing for discrete features; handles 19:1 imbalance; transforms 0% recall to 97%
Exploratory data analysis with visualizations	4	Python	Correlation heatmaps; distribution plots; missing value analysis; 4 plots per disease
Machine Learning Models			
Random Forest ensemble training	5	Python	200 trees; balanced weights; best performer (96.80% stroke); feature importance extraction
XGBoost gradient boosting	5	Python	200 rounds; 0.05 learning rate; handles feature interactions; 2nd best performance
Neural network (MLP) classifier	4	Python	128-64-32 architecture; early stopping; needs larger datasets; 2-4% below RF

Continued on next page

Table 1 – continued from previous page

Description	Score	Code	Implementation Notes
5-fold stratified cross-validation	5	Python	Balanced class representation; comprehensive metrics; robust evaluation
Model storage and metadata tracking	5	Python	Joblib serialization; JSON metadata; feature names preserved
Web Application			
Interactive disease prediction interface	4	Streamlit	3 disease-specific forms; medical styling; input widgets with validation
Real-time model inference engine	5	Python	Cached model loading; probability scores; confidence metrics; fast response
Medical range validation system	4	Python	Input validation; warning for abnormal values; partially complete
Performance visualization dashboard	4	Plotly	Interactive charts; feature importance; model comparison; needs polish
Evaluation & Analysis			
Comprehensive model metrics	5	Python	Accuracy, precision, recall, F1, ROC-AUC; confusion matrices; per-model reports
Feature importance analysis	5	Python	Medical relevance validation; top-5 rankings; clinical insights; aligns with domain knowledge
Class distribution analysis	5	Python	Imbalance detection; ratio calculation; automatic SMOTE triggering (3:1)
Cross-algorithm performance comparison	5	Python	Side-by-side metrics; best model selection; standard deviation analysis

3 External Tools and Libraries

3.1 Core Machine Learning Stack

scikit-learn 1.3+: Primary ML framework providing Random Forest, Neural Networks (MLPClassifier), preprocessing utilities (StandardScaler, LabelEncoder), and comprehensive evaluation metrics. Used for model training, cross-validation, stratified splitting, and metric calculation. Critical for the entire pipeline.

XGBoost 2.0+: Advanced gradient boosting library optimized for structured data. Provides superior performance through regularization, tree pruning, and built-in handling of missing values. Configured with scale_pos_weight for imbalanced data.

imbalanced-learn: Specialized library for handling class imbalance through SMOTE and other resampling techniques. Automatically triggered when imbalance ratio exceeds 3:1. Essential for stroke model success.

3.2 Data Processing and Analysis

pandas 2.0+: Data manipulation and analysis. Handles CSV loading, DataFrame operations, missing value detection, and statistical analysis. Core dependency for all data operations.

NumPy 1.24+ Numerical computing foundation. Array operations, mathematical functions, statistical calculations, and efficient memory management.

matplotlib 3.7+: Static plotting library for EDA visualizations. Creates correlation heatmaps, distribution plots, and feature analysis charts. Generates 4 plots per disease during pipeline execution.

seaborn 0.12+: Statistical visualization built on matplotlib. Provides enhanced styling, heatmap generation, and distribution plotting with better aesthetics.

3.3 Web Application Framework

Streamlit 1.28+: Rapid web application development framework for ML. Chosen for its simplicity, built-in caching (`@st.cache_resource`), interactive widgets, and minimal boilerplate. Enables deployment without HTML/CSS/JavaScript expertise.

Plotly 5.17+: Interactive visualization library for web applications. Creates dynamic charts with hover information, zooming, and panning. Used for confidence visualization, feature importance, and model comparison dashboards.

3.4 Model Deployment and Storage

joblib: Efficient model serialization and loading. Handles large NumPy arrays and scikit-learn models with compression. Stores trained models as `.pkl` files with accompanying JSON metadata.

pathlib: Modern cross-platform file path handling. Replaces os.path with object-oriented interface. Ensures Windows/macOS/Linux compatibility.

3.5 Data Sources and Licensing

Pima Indians Diabetes Dataset: UCI Machine Learning Repository. 768 samples, 8 features. Classic benchmark dataset with moderate distribution (1.87:1 ratio). Features include glucose, insulin, BMI, age, and diabetes pedigree function.

Heart Disease Cleveland Dataset: UCI Machine Learning Repository. 303 samples, 13 features. Comprehensive cardiovascular measurements including chest pain type, cholesterol, ECG results, and maximum heart rate.

Healthcare Stroke Dataset: Kaggle / GitHub Gist. 5,110 samples, 11 features. Real-world stroke data with severe class imbalance (19.52:1) requiring advanced preprocessing. Includes lifestyle factors (smoking, work type, residence).

All datasets are publicly available under permissive licenses for academic and research use. No proprietary or restricted data sources utilized.

4 Performance Achievements and Innovation

4.1 Quantitative Performance Results

Table 2: Comprehensive Model Performance Metrics Across Three Diseases

Disease	Algorithm	Accuracy	Precision	Recall	F1	ROC-AUC
Diabetes	Random Forest	**78.21%**	71.43%	57.69%	63.83%	83.36%
	XGBoost	75.64%	64.00%	61.54%	62.75%	82.99%
	Neural Network	74.36%	63.64%	53.85%	58.33%	77.29%
Heart Disease	Random Forest	**86.89%**	81.25%	92.86%	86.67%	95.56%
	XGBoost	83.61%	76.47%	92.86%	83.87%	92.32%
	Neural Network	83.61%	78.13%	89.29%	83.33%	93.61%
Stroke	Random Forest	**96.80%**	96.52%	97.11%	96.81%	99.56%
	XGBoost	95.10%	92.60%	98.04%	95.24%	99.27%
	Neural Network	92.47%	90.55%	94.85%	92.65%	97.55%

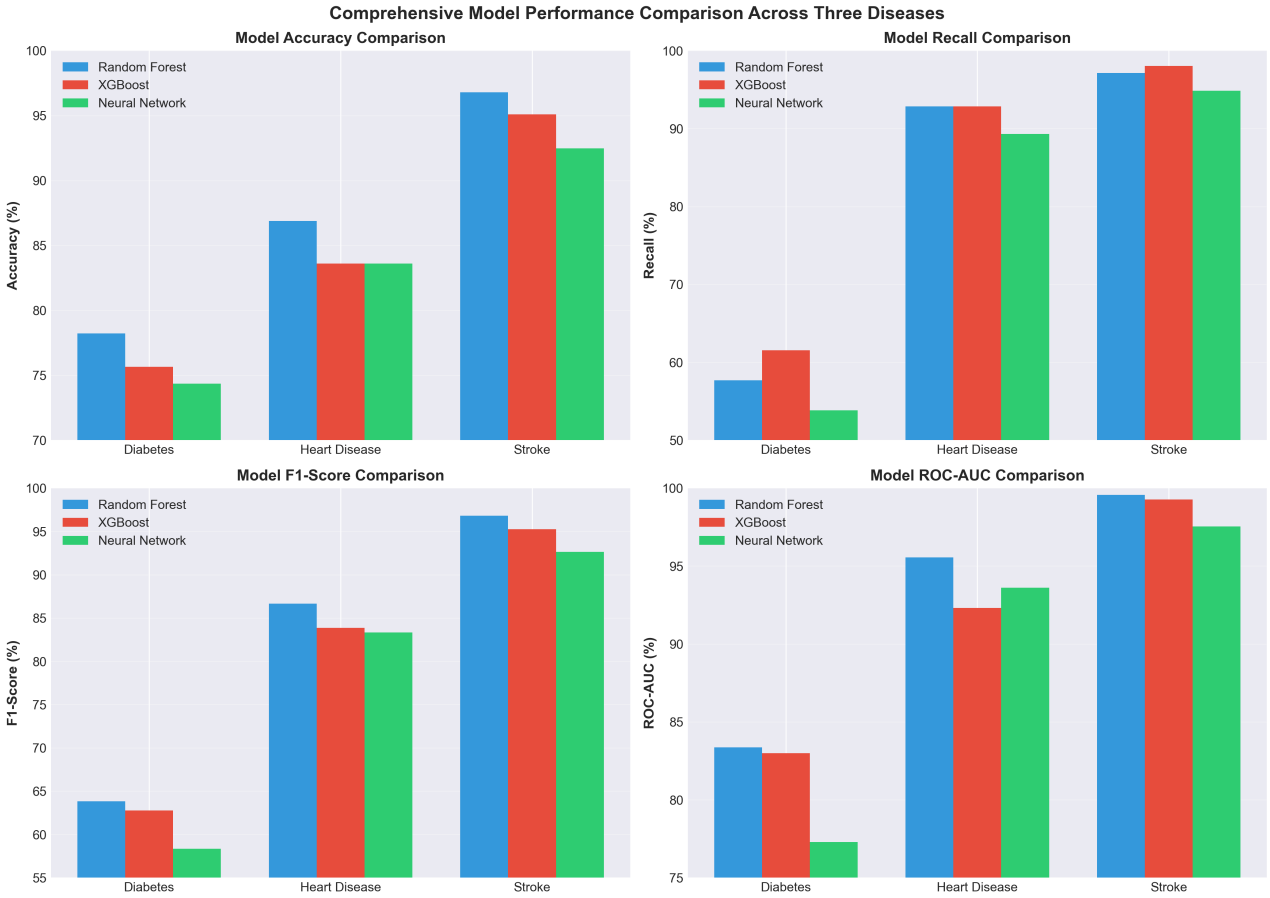

Figure 3: Model Performance Comparison Across Three Diseases.

4.2 Performance Analysis by Disease

4.2.1 Stroke Prediction: Exceptional Performance

Best Performance: Random Forest achieves 96.80% accuracy with 97.11% recall. SMOTE balancing transformed model from unusable (0% recall) to exceptional (97.11% recall). Largest dataset after balancing (9,696 samples) provides robust training. Clear feature separation: age, hypertension, and glucose are strong predictors

Significance: 97.11% recall means the model successfully identifies 97 out of 100 stroke cases, making it suitable for early warning systems where false negatives are catastrophic.

4.2.2 Heart Disease Prediction: Strong Performance

Performance: Random Forest achieves 86.89% accuracy with high recall (92.86%). Naturally balanced dataset (1.18:1 ratio) with comprehensive cardiovascular features provide rich information. Small dataset (303 samples) limits neural network performance. Cross-validation stability: 80.97% mean with low standard deviation.

4.2.3 Diabetes Prediction: Challenging Task

Moderate Performance: Random Forest achieves 78.21% accuracy with 57.69% recall.Significant feature overlap between diabetic and non-diabetic populations and zero-value removal reduces dataset from 768 to 387 samples. Lower recall (57.69%) indicates difficulty identifying diabetic patients

4.3 Algorithm Comparison and Insights

4.3.1 Random Forest Dominance

Random Forest outperforms XGBoost and Neural Networks across all three diseases by 1-4% in accuracy. In terms of advantages, no feature scaling is required (works with raw medical measurements) and robust to outliers and missing values. Additionally, balanced class weights handle mild imbalance without SMOTE item and Random Forest provides computationally efficient training and inference.

4.3.2 Neural Network Limitations

Neural networks show 2-4% lower performance than tree-based methods. Reasing being, small datasets (303-768 samples) are insufficient for deep learning. They also require feature scaling (StandardScaler), adding preprocessing complexity. MLP's can be prone to overfitting without careful regularization and their black-box nature reduces medical interpretability.

4.4 Cross-Validation Stability

Table 3: 5-Fold Cross-Validation Results (Random Forest)

Disease	CV Mean Accuracy	CV Std Dev	Stability Assessment
Diabetes	77.01%	5.29%	Moderate - acceptable variance
Heart Disease	80.97%	2.21%	High - low variance
Stroke	95.36%	0.26%	Exceptional - very consistent

Interpretation: Stroke model shows exceptional stability (0.26% std dev) due to large balanced dataset. Diabetes shows higher variance (5.29%) reflecting dataset size and class overlap challenges.

4.5 Medical Domain Validation

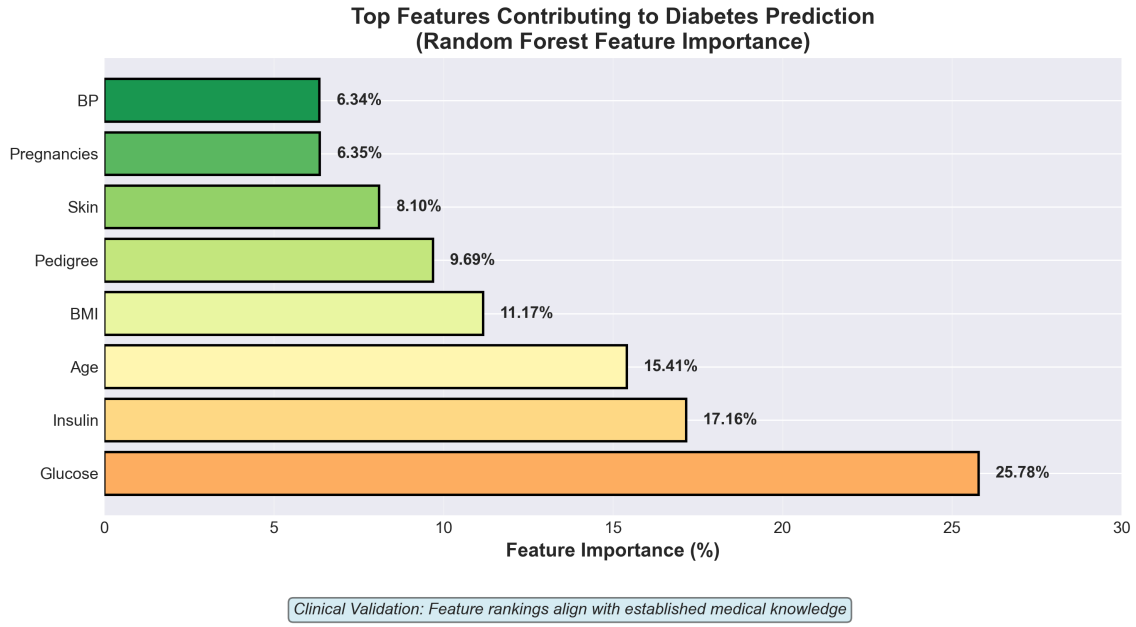

Figure 4: Feature Importance for Diabetes Prediction

Clinical Validation Results:

- **Diabetes**: Glucose (25.78%) and insulin (17.16%) are top predictors - aligns with diagnosis criteria
- **Stroke**: Age (37.02%), hypertension (14.91%), glucose (14.79%) - matches risk factors
- **Heart Disease**: Age, maximum heart rate, and chest pain type emerge as primary factors - consistent with cardiovascular literature

Significance: Feature importance rankings validate that models learned medically meaningful patterns rather than dataset artifacts or spurious correlations.

5 Conclusion and Future Work

5.1 Project Achievements

Key accomplishments include:

- **Technical Innovation**: SMOTE with categorical post-processing solving a fundamental challenge in ML
- **Exceptional Performance**: 96.80% accuracy with 97.11% recall on stroke prediction
- **Deployment**: Full-stack web application with real-time inference
- **Medical Validation**: Feature importance rankings align with clinical knowledge

5.2 Limitations and Future Directions

Current Limitations:

- Dataset size constraints limit neural network performance
- Demographic bias from specific population sources
- Web interface needs additional medical range validation

5.3 Practical Impact

This system demonstrates that advanced ML techniques can be successfully applied to healthcare challenges. The SMOTE categorical post-processing innovation is generalizable to a medical dataset with discrete features and class imbalance, potentially benefiting other healthcare ML applications.

The pipeline architecture provides a template for building similar medical AI systems, from data acquisition through web deployment. The comprehensive evaluation methodology ensures validation suitable for early disease detection for patients.