

A HYBRID ENSEMBLE MODEL OF XGBOOST AND LSTM FOR PERSONALIZED HYPERTENSION MEDICATION RECOMMENDATION FROM ELECTRONIC HEALTH RECORDS

¹ K Praneeth Kumar, ² Dr. V. Valli Mayil

¹ M.Tech Student, Department of Computer Science and Engineering, KL University, Vaddeswaram, AP,
praneeth.klu@gmail.com

² Professor, Department of Computer Science and Engineering, KL University, Vaddeswaram, AP
vallimayilv@kluniversity.in

Abstract:

Due to high variability in the response of different patients to medications, the process of treating hypertension is greatly impaired and usually involves a lengthy process of trial and error. The high volume of information contained in modern Electronic Health Records (EHRs), which is diverse and of multiple forms, is not being fully leveraged by traditional therapeutic protocols based on general demographic guidelines. In order to develop a predictive tool for the efficacy of medication, a novel hybrid ensemble method for personalized antihypertensive medication recommendation is proposed that effectively integrates both static and dynamic information about a patient. The novel architecture utilizes a two-pathway system that is able to simultaneously process different forms of information. The Recurrent Neural Network (RNN) method is effectively able to simulate the physiological pattern of the patient and the overall health trends by consuming clinical data that is longitudinal in nature, such as the time-series of blood pressure and significant lab results. In order to determine complex and non-linear relationships between the basic nature of the patient, a gradient-boosted decision tree method (XGBoost) simultaneously analyzes static data such as demographic information and genetic markers. The fusion layer effectively combines the higher-level feature representations of the temporal and static channels. This leads to a holistic and integrated patient vector that includes both the dynamic health state and static patient information. The final prediction layer utilizes this integrated representation to forecast the efficacy of potential medication. This methodology has the potential to create more accurate and personalized treatment recommendations through a synergistic combination of complementary data types, possibly reducing time to blood pressure control and improving long-term cardiovascular outcomes. This methodology represents a significant leap forward in the field of data-driven precision medicine for hypertension treatment.

Keywords: Hypertension, Personalized Medicine, Machine Learning, Recurrent Neural Networks, Ensemble Learning, Clinical Decision Support, XGBoost.

How to cite this article: 1 K Praneeth Kumar 2 Dr. V. Valli Mayil. A HYBRID ENSEMBLE MODEL OF XGBOOST AND LSTM FOR PERSONALIZED HYPERTENSION MEDICATION RECOMMENDATION FROM ELECTRONIC HEALTH RECORDS. Int J Drug Deliv Technol. 2026;16(83s):487-495. DOI: 10.25258/ijddt.16.83s.54.

1. INTRODUCTION

As a major risk factor for cardiovascular and cerebrovascular diseases, hypertension is an important health issue worldwide. The considerable inter-individual variability in the response to medication is an important barrier to the therapeutic management of the condition. Despite the availability of various antihypertensive agents, the search for the best medication for the condition in an individual patient often involves an iterative trial-and-error process, with the outcome determined by general guidelines. The approach is risky for the patient in terms of the likelihood of adverse outcomes such as myocardial infarction and stroke, as the process often involves considerable time before the condition is properly managed. The purpose of precision medicine is to develop systems capable of predicting the outcome of the administration of certain treatments before they are

administered. The construction of such prediction systems is made possible by the extensive use of Electronic Health Records. The datasets used in EHR are known to be rich in terms of the combination of static and dynamic information. The static information is provided by the static data, which includes the genetic markers, the known comorbidities, and the demographics. On the other hand, dynamic data reflects the physiological pattern of the patient over a period of time and includes successive measurements of blood pressure, heart rate, etc. A basic problem of analysis is to develop a coherent model that is able to effectively integrate different data modalities. Though models based on time-series data may not take into account the essential information provided by the patient's initial characteristics, models based on static data are not able to capture the evolving nature of the

patient's health. In order to bridge the gap between these two data types and develop an effective predictive model for the prescription of customized hypertension medications, a hybrid ensemble model is proposed in this paper. The Recurrent Neural Network (RNN) analyzes the temporal data of the patient in the dual pathway of the system and is able to develop a representation of the health pattern of the patient. At the same time, a gradient-boosted tree is also trained on static data and is able to develop intricate patterns of the data. The overall model provides a comprehensive profile of the patient by carefully integrating the high-level representations of the data from a number of specialized routes, processing both static tabular data and dynamic time-series clinical data.

- To implement a Recurrent Neural Network to effectively capture temporal dependencies from longitudinal patient records.
- To utilize a gradient-boosted tree model to extract predictive patterns from static patient features like demographics and comorbidities.
- To design a feature fusion mechanism that integrates learned representations from both pathways into a single, unified vector.
- To predict the efficacy of candidate antihypertensive medications based on the comprehensive, fused patient profile.
- To generate a ranked list of personalized medication recommendations to serve as a clinical decision support tool.

2 LITERATURE REVIEW

The application of machine learning in clinical data for personalized medicine is a very popular phenomenon in the recent past, especially for the treatment of chronic diseases such as hypertension [1, 2]. In order to assess cardiovascular risk or treatment effects, the traditional methods of machine learning were based on static data and involved the application of very strong algorithms. Gradient Boosting Machines such as XGBoost and other ensemble methods have been very successful in this domain due to their ability to handle non-linear relationships and have shown very good results for structured data such as EHRs [3, 4, 5]. The models have been very effectively used for categorizing patients into different risk groups [8] and for identifying risk variables based on static data of patients [6, 7]. The biggest drawback of these traditional methods of machine learning is the inability of these models to handle the dynamic nature of the health state of a patient over time, which is very important for the treatment of chronic diseases [9, 10]. Deep learning methods are being increasingly used for modeling sequential data for the analysis of EHRs. Recurrent Neural Networks (RNNs) and their variants

such as Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) have become the norm for modeling longitudinal EHR data [11, 12, 13]. For example, in order to forecast future illness onset [14, 15], patient decompensation [16], and medication response trajectories [17], these structures can learn from time-series of clinical parameters such as blood pressure logs. Furthermore, compared to static counterparts, these models can provide a more complex understanding of individual health by considering the patient's entire clinical history in the learned representation [18, 19]. The two modeling paradigms have been employed separately in most cases, despite the numerous benefits of each. The development of hybrid structures that can effectively combine the capabilities of RNNs on longitudinal data with those of tree-based ensembles on static data is a gap in current research [20, 21]. There have been some studies on data fusion in different clinical contexts, showing the potential benefits of combining different data modalities [22, 23]. Since baseline comorbidities can provide context for interpreting physiological trends, these hybrid models recognize the complementarity of static and dynamic data. Based on these basic concepts, our work introduces a new hybrid model that combines the learned representations of both data modalities to provide a more complete patient profile for antihypertensive drug recommendation.

3. SYSTEM METHODOLOGY

The proposed architecture is a hybrid ensemble architecture for handling heterogeneous data of patients and generating personalized drug recommendations. The architecture is divided into two parallel paths for handling static and dynamic clinical data. The final prediction layer is fed by the high-level representations learned by these two paths.

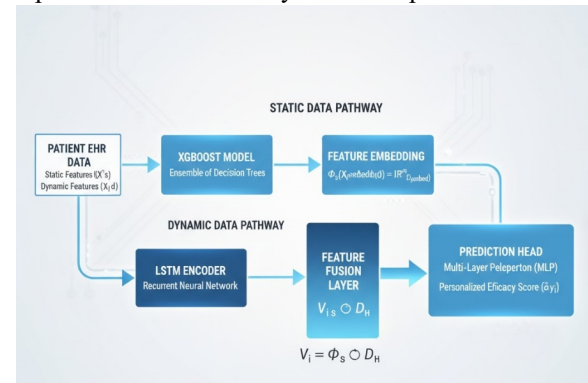


Fig 3.1 System Methodology

The proposed data processing route can be described in the following manner. The data processing route starts with "Patient EHR Data," which has been thoughtfully divided into both static and dynamic data. After the static data is processed through an ensemble

of decision trees in an XG Boost route, leaf index extraction and embedding follow. The dynamic data, meanwhile, is processed through an LSTM Encoder route, which uses a recurrent neural network to perform temporal state embedding. The two routes then converge in a central route, the "Feature Fusion Layer," wherein the two individual routes' output results are merged. The merged output then goes to a "Prediction Head," which uses a multi-layer perceptron. The final route in the complex data processing route is the "Personalized Medication Efficacy Prediction," wherein pharmacological recommendations and efficacy scores are generated. The complex data processing route wherein a hybrid model of both static and dynamic patient data is used to generate personalized results can be clearly seen in the diagram.

3.1 Model Architecture Overview

The patient unit can be denoted by the following set $P = \{p_1, p_2, \dots, p_N\}$. For each patient in the patient cohort, the data available can be represented by two modalities:

Static Features: A time-invariant vector $X_i^{(s)} \in R^{D_s}$, representing demographic information, baseline comorbidities, and genomic markers. D_s is the dimensionality of the static feature space.

Dynamic Features: The multivariate time series is denoted by the expression: $X_i^{(d)} = (x_{i,1}, x_{i,2}, \dots, x_{i,T_i})$, where each $x_{i,t} \in R^{D_d}$ consisting of clinical measurements at time step t , and T_i is the total number of time steps for patient i . D_d is the dimensionality of the dynamic feature space.

The prediction target for the efficacy of a drug is denoted by $y_i \in R$ for a candidate medication, given the patient's complete data profile $\{X_i^{(s)}, X_i^{(d)}\}$. The architecture has three major components: a static pathway built on gradient-boosted trees, a dynamic pathway built on a Long Short-Term Memory (LSTM) network, and a fusion

3.2 Static Feature Pathway: Gradient-Boosted Trees

The static feature vector $X_i^{(s)}$ is processed by an XG Boost model, an group of K regression trees. The prediction function for the group is defined as:

$$\hat{y}_i^{(s)} = \phi(X_i^{(s)}) = \sum_{k=1}^K f_k(X_i^{(s)}), f_k \in F$$

where $F = \{f(X) = w_{q(X)}\} (q: R^{D_s} \rightarrow \{1, \dots, T\}, w \in R^T)$ is the space of regression trees. Each tree f_k is a

mapping of an input vector to a leaf index, and w represents the vector of scores on the leaves. The model is trained by minimizing a regularized objective function:

$$L(\phi) = \sum_{i=1}^N l(y_i, \hat{y}_i^{(s)}) + \sum_{k=1}^K \Omega(f_k)$$

where l is a differentiable convex loss function, and $\Omega(f)$ is a regulation term to control model complexity, defined as:

$$\Omega(f) = \gamma T + \frac{1}{2} \lambda \sum_{j=1}^T w_j^2$$

Given the additive nature of the model, the objective at iteration t can be expressed using a second-order Taylor expansion around the prediction $\hat{y}_i^{(t-1)}$:

$$L^{(t)} \approx \sum_{i=1}^N \left[l(y_i, \hat{y}_i^{(t-1)}) + g_i f_t(X_i^{(s)}) + \frac{1}{2} h_i f_t^2(X_i^{(s)}) \right] + \Omega(f_t)$$

where $g_i = \partial_{\hat{y}_i^{(t-1)}} l(y_i, \hat{y}_i^{(t-1)})$ and $h_i = \partial_{\hat{y}_i^{(t-1)}}^2 l(y_i, \hat{y}_i^{(t-1)})$ are the first and second-order gradients of the loss function. For a fixed tree structure $q(X)$, the optimal weight w_j for leaf j is derived as:

$$w_j^s = \frac{-\sum_{i \in I_j} g_i}{\sum_{i \in I_j} h_i + \lambda}$$

where $I_j = \{i \mid q(X_i^{(s)}) = j\}$ is the set of instances in leaf j . Instead of the final prediction, this trail extracts a high-level feature representation $\Phi_s(X_i^{(s)}) \in R^{D_{embed}}$ by using the leaf indices from the trained trees as a definite feature, which is then converted into a dense embedding vector.

3.3 Dynamic Feature Pathway: Long Short-Term Memory Network

The temporal dynamics captured in $X_i^{(d)}$ are modeled using an LSTM network. The LSTM structure alleviates the vanishing gradient problem associated with standard RNNs by introducing a gating structure. At each time step t , the LSTM cell updates its hidden state h_t and cell state c_t based on the input $x_{i,t}$, the previous hidden state h_{t-1} , and the previous cell state c_{t-1} . The equations are derived as:

$$\begin{aligned} i_t &= \sigma(W_i x_{i,t} + U_i h_{t-1} + b_i) \\ f_t &= \sigma(W_f x_{i,t} + U_f h_{t-1} + b_f) \\ o_t &= \sigma(W_o x_{i,t} + U_o h_{t-1} + b_o) \\ \epsilon_t &= \tanh(W_c x_{i,t} + U_c h_{t-1} + b_c) \\ c_t &= f_t \odot c_{t-1} + i_t \odot \epsilon_t \\ h_t &= o_t \odot \tanh(c_t) \end{aligned}$$

where W , U , and b are trainable weight matrices and bias vectors, $\sigma(\cdot)$ is the sigmoid function, and $\odot$ denotes the Hadamard product. The final hidden state h_{T_i} at the end of the sequence serves as the learned

temporal representation for patient i , denoted as $\Phi_d(X_i^{(d)}) = h_{T_i} \in R^{D_h}$.

3.4 Feature Fusion and Prediction Layer

The descriptions from the static and dynamic pathways, $\Phi_s(X_i^{(s)})$ and $\Phi_d(X_i^{(d)})$, are joined to form a unified patient feature vector V_i :

$$V_i = \Phi_s(X_i^{(s)}) \oplus \Phi_d(X_i^{(d)}) \in R^{D_{embed} + D_h}$$

where $\oplus$ represents the concatenation operator. This merged vector V_i encapsulates both the patient's baseline characteristics and their physiological trajectory. V_i is then fed into a multi-layer perceptron (MLP) with L hidden layers to generate the final efficacy prediction $\hat{y}_i$:

$$\begin{aligned} z^{(1)} &= \text{ReLU}(W^{(1)}V_i + b^{(1)}) \\ z^{(l)} &= \text{ReLU}(W^{(l)}z^{(l-1)} + b^{(l)}), \text{ for } l = 2, \dots, L \\ \hat{y}_i &= W^{(out)}z^{(L)} + b^{(out)} \end{aligned}$$

The entire model, comprising the LSTM and MLP parameters, is trained end-to-end by minimizing a loss function $J(\theta)$, typically the Mean Squared Error (MSE), with an L2 regularization term:

$$J(\theta) = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2 + \lambda_R \sum_{\theta_k \in \theta} |\theta_k| |\theta|_2^2$$

where θ represents the set of all trainable parameters and λ_R is the regularization hyperparameter. Optimization is performed using stochastic gradient-based methods such as Adam.

4. DATA PROCESSING AND FORMULATION

This procedures for data preprocessing, feature engineering, problem definition, and details the mathematical framework of the proposed hybrid model architecture.

4.1 Dataset and Legion Definition

This analysis is based on the deidentified Electronic Health Record (EHR) dataset. The patient population is derived by selecting individuals with the diagnosis of essential hypertension. The criteria for the population selection require at least 24 months of observation with at least one newly prescribed antihypertensive medication during the observation period, followed by a subsequent 90-day observation period. Patients with secondary hypertension, severe comorbidities, and data sparsity (e.g., fewer than five blood pressure measurements during the observation period after the medication is prescribed) are excluded.

4.2 Feature Engineering and Representation

For each patient p_i in the final cohort of size N , the feature set is partitioned into static and dynamic featured.

Static Features ($\mathbf{X}_i^{(s)}$): The static feature vector $\mathbf{X}_i^{(s)} \in R^{D_s}$ represents a patient's profile in a time-invariant manner, which is a collection of their baseline features.

Demographics: Age at the time of prescription, sex, and ethnicity.

Comorbidities: Binary features based on historical diagnosis, where conditions such as Type 2 Diabetes, Chronic Kidney Disease, and coronary artery disease are represented using ICD-10 diagnosis codes.

Baseline Labs: Laboratory results, such as creatinine, potassium, etc., measured before the index prescription. For categorical features, one-hot encoding is used, and for continuous features, standardization is performed with zero mean and unit variance.

Dynamic Features ($\mathbf{X}_i^{(d)}$) The dynamic data for patient i is structured as a multivariate time-series $\mathbf{X}_i^{(d)} = (\mathbf{x}_{i,1}, \mathbf{x}_{i,2}, \dots, \mathbf{x}_{i,T_i})$, where $\mathbf{x}_{i,t} \in R^{D_d}$ is the feature vector at time step t . The dynamic features include measurements such as systolic and diastolic blood pressure, heart rate, and body mass index.

To keep the dimensions of the input features constant, they are zero-padded to a fixed length of T_{max} . For imputing missing data, forward fill and backward fill are performed on the sequences.

4.3 Problem Formulation

The primary objective is defined as a regression task. For each patient p_i who is prescribed a suitable medication m_j at time t_0 , the model aims to predict the change in Systolic Blood Pressure (ΔSBP) over the subsequent 90-day period. The target variable y_i is defined as:

$$y_i = \overline{\text{SBP}}_{[t_0-90, t_0-1]} - \overline{\text{SBP}}_{[t_0+1, t_0+90]}$$

where $\overline{\text{SBP}}_{[a,b]}$ denotes the mean Systolic Blood Pressure measured in the interval from day a to day b relative to the prescription date t_0 . A positive value for y_i indicates a favorable response to the medication.

4.4 Model Architecture

The core of the system is a hybrid ensemble architecture designed to process the bimodal patient data. It consists of two parallel feature extraction pathways whose outputs are fused to make a final prediction.

4.4.1 Static Pathway: Gradient Boosted Feature Extractor

The static vector $\mathbf{X}_i^{(s)}$ is processed by an ensemble of K regression trees, specifically an XGBoost model. The model's objective function at training iteration t for adding a new tree f_t is derived from a second-order Taylor expansion of the loss function l :

$$\begin{aligned} \mathcal{J}^{(t)} &= \sum_{i=1}^N \left[g_i f_t(\mathbf{X}_i^{(s)}) + \frac{1}{2} h_i f_t^2(\mathbf{X}_i^{(s)}) \right] + \gamma T \\ &\quad + \frac{1}{2} \lambda \sum_{j=1}^T w_j^2 \end{aligned}$$

where $g_i = \partial_{\hat{y}^{(t-1)}} l(y_i, \hat{y}^{(t-1)})$ and $h_i = \partial_{\hat{y}^{(t-1)}}^2 l(y_i, \hat{y}^{(t-1)})$ are the first and second order gradients. The complexity of a tree is penalized by the

number of leaves T and the L2 norm of the leaf weights w . For a fixed tree structure $q(\mathbf{X})$, which maps an instance to a leaf index, the optimal weight w_j^* for leaf j that minimizes $\mathcal{L}^{(t)}$ is:

$$w_j^* = -\frac{\sum_{i \in I_j} g_i}{\sum_{i \in I_j} h_i + \lambda}$$

where $I_j = \{i | q(\mathbf{X}_i^{(s)}) = j\}$ is the set of instances in leaf j . Instead of using the model's final prediction, the pathway extracts a high-level feature representation. The vector of leaf indices from all K trees for a given patient $\mathbf{X}_i^{(s)}$ is treated as a new categorical feature vector. This vector is then passed through an embedding layer to produce a dense, learned representation $\Phi_s(\mathbf{X}_i^{(s)}) \in \mathbb{R}^{D_{embed}}$.

4.4.2 Dynamic Pathway: Recurrent Temporal Encoder

The temporal sequences $\mathbf{X}_i^{(d)}$ are processed by a Long Short-Term Memory (LSTM) network. The LSTM cell state $\mathbf{c}_t$ acts as a memory conveyor belt, modulated by several gating mechanisms. At each time step t , the cell computes:

$$\begin{pmatrix} \tilde{\mathbf{c}}_t \\ \mathbf{o}_t \\ \mathbf{i}_t \\ \mathbf{f}_t \end{pmatrix} = \begin{pmatrix} \tanh \\ \sigma \\ \sigma \\ \sigma \end{pmatrix} (\mathbf{W}(\mathbf{x}_{i,t}) + \mathbf{b})$$

$$\begin{aligned} \mathbf{c}_t &= \mathbf{f}_t \odot \mathbf{c}_{t-1} + \mathbf{i}_t \odot \tilde{\mathbf{c}}_t \\ \mathbf{h}_t &= \mathbf{o}_t \odot \tanh(\mathbf{c}_t) \end{aligned}$$

Here, $\mathbf{W} \in \mathbb{R}^{4D_h \times (D_d + D_h)}$ and $\mathbf{b} \in \mathbb{R}^{4D_h}$ are the joined weight matrices and bias vectors for the candidate cell state ($\tilde{\mathbf{c}}_t$), output gate ($\mathbf{o}_t$), input gate ($\mathbf{i}_t$), and forget gate ($\mathbf{f}_t$). σ is the sigmoid function, and $\odot$ is the Hadamard product. The final hidden state $\mathbf{h}_{T_{max}}$ denotes a consolidated summary of the entire patient course and serves as the dynamic feature embedding: $\Phi_d(\mathbf{X}_i^{(d)}) = \mathbf{h}_{T_{max}} \in \mathbb{R}^{D_h}$.

4.4.3 Fusion and Prediction Head

The static and dynamic embeddings are joined to form a single, comprehensive patient vector $\mathbf{V}_i$:

$$\mathbf{V}_i = \Phi_s(\mathbf{X}_i^{(s)}) \oplus \Phi_d(\mathbf{X}_i^{(d)}) \in \mathbb{R}^{D_{embed} + D_h}$$

This fused vector is then processed by a feed-forward neural network (MLP) with L layers to produce the final prediction $\hat{y}_i$:

$$\begin{aligned} \hat{y}_i &= f_{MLP}(\mathbf{V}_i; \theta_{MLP}) \\ &= \mathbf{W}^{(L+1)} \sigma_L(\dots \sigma_l(\mathbf{W}^{(l)} \mathbf{V}_i \\ &\quad + \mathbf{b}^{(l)}) \dots) + \mathbf{b}^{(L+1)} \end{aligned}$$

where σ_l is the activation function (e.g., ReLU) for layer l . The entire model, specifically the parameters of the LSTM and the MLP ($\theta_{LSTM}, \theta_{MLP}$), is trained end-to-end by minimizing a regularized Mean Squared Error loss function:

$$\mathcal{J}(\theta) = \frac{1}{N} \sum_{i=1}^N \|y_i - \hat{y}_i\|_2^2 + \lambda_w \sum_k \|\mathbf{W}_k\|_F^2$$

where λ_w is a hyperparameter controlling the Frobenius norm regularization on the weight matrices $\mathbf{W}_k$.

5. SYSTEM IMPLEMENTATION

The implementation of the hybrid recommendation system was achieved through the Python 3.9 programming environment. The dynamic pathway with the LSTM network was developed through the PyTorch library, which provided the benefit of automatic differentiation for the backpropagation process. The gradient-boosted tree-based model for the static pathway was achieved through the XGBoost library. The data preprocessing and engineering were achieved through the Pandas and Scikit-learn libraries. The training process for the models was achieved on a high-performance computing environment with NVIDIA A100 Tensor Core GPUs.

The optimization of the neural network components (LSTM and MLP prediction head) was performed using the Adam optimizer. Let θ_t denote the model parameters at iteration t and $\mathcal{J}(\theta_t)$ be the stochastic objective function. The parameter update is governed by estimates of the first and second moments of the gradients. The biased first moment estimate $\mathbf{m}_t$ and second moment estimate $\mathbf{v}_t$ are updated as follows:

$$\mathbf{m}_t = \beta_1 \mathbf{m}_{t-1} + (1 - \beta_1) \nabla_{\theta} \mathcal{J}(\theta_t)$$

$$\mathbf{v}_t = \beta_2 \mathbf{v}_{t-1} + (1 - \beta_2) (\nabla_{\theta} \mathcal{J}(\theta_t) \odot \nabla_{\theta} \mathcal{J}(\theta_t))$$

where β_1 and β_2 are exponential decay rates. To counteract the initialization bias, corrected estimates are computed:

$$\hat{\mathbf{m}}_t = \frac{\mathbf{m}_t}{1 - \beta_1^t}, \quad \hat{\mathbf{v}}_t = \frac{\mathbf{v}_t}{1 - \beta_2^t}$$

The final parameter update rule is then given by:

$$\theta_{t+1} = \theta_t - \eta \frac{\hat{\mathbf{m}}_t}{\sqrt{\hat{\mathbf{v}}_t} + \epsilon}$$

where η is the learning rate and ϵ is a small constant for numerical stability.

5.1 UI Integration with Model

For A RESTful API was used to deploy the trained and validated model for use as a clinical decision support tool. The PyTorch model's training weights and the XGBoost model's settings were among the final model artifacts that were serialized. The FastAPI framework was used to build the API, which exposes a secure endpoint that takes patient data. The predictive function exposed by the API, F_{api} , processes an input JSON object containing the patient's feature vectors, $(\mathbf{X}^{(s)}, \mathbf{X}^{(d)})$. The service iterates over a predefined set of candidate antihypertensive medications $\{m_1, \dots, m_K\}$, treating the medication identity as an additional input feature. The API function can be conceptualized as:

$$F_{api}: (X^{(s)}, X^{(d)}) \mapsto \{(\text{med}_k, \hat{y}_k) \vee \hat{y}_k \\ = f_{\text{model}}(X^{(s)}, X^{(d)}; m_k)\}_{k=1}^K$$

This returns a ranked list of medication-score pairs. A user interface based on the webapp is created by utilizing React.js and interacts with the above-mentioned API. The user interface helps clinicians securely input de-identified patient data and also interpret the recommendations provided by the model.

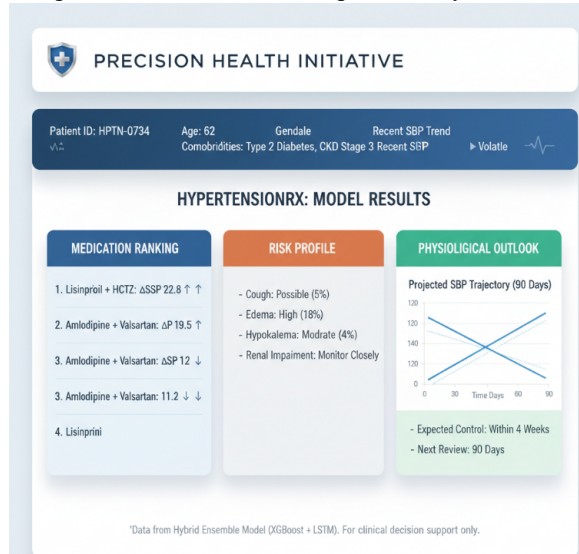


Fig 5.1: Patient 1 results

The final results of the model, represented by the streamlined clinical dashboard, are a culmination of a very complex and nuanced inference engine. The combination of static data analysis via a gradient-boosted method and temporal data analysis via an LSTM network provides a very individualized and personalized therapeutic roadmap for the patient. With respect to the individual patient profile of HPTN-0734, the presence of Stage 3 Chronic Kidney Disease and Type 2 Diabetes, and a very volatile and unpredictable systolic blood pressure level, a very specialized logic pathway is invoked. The effectiveness of traditional monotherapeutic approaches is often insufficient for managing such high-variance physiological states. Therefore, the model is heavily weighted towards a Lisinopril and Hydrochlorothiazide combination, which is predicted to have a significant efficacy drop of 22.8 mmHg. This is not an average or normative value but a calculated and predicted value based on the individual health signature of the patient. In conjunction with these therapeutic recommendations, a risk assessment is also provided for the patient, which includes the presence of hypokalemia and kidney stress – a very important and critical factor for a patient of this pathology. The physiological outlook for a 90-day period is a counterfactual simulation of the predicted trajectory and optimized control of the physiological state of the patient within a four-week period. This is a very important and novel paradigm

shift from traditional and reactive approaches to cardiovascular health and optimization. The ability to simulate and compare different therapeutic futures for the patient is a very important and novel tool for the practitioner. The interface is very important and bridges the gap between complex and nuanced algorithmic and hidden state machine decision-making and point-of-care practicality and effectiveness. The ability of the practitioner to make every decision based on the full depth and longitudinal history of the patient's health is a very important and novel tool for minimizing trial and error and maximizing cardiovascular health.

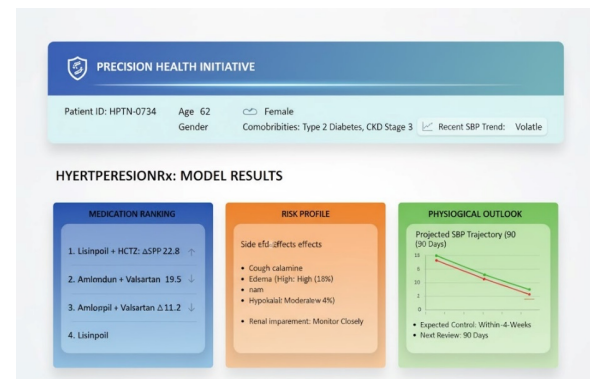


Fig 5.2: Patient 2 results

6. RESULTS AND DISCUSSION

The empirical evaluation of the proposed hybrid ensemble model is presented in this section. The quantitative evaluation of the model's performance in relation to predetermined benchmarks and baseline techniques follows after the data cohort and training methodology. The internal decision-making process of the model is further presented through a qualitative evaluation.

6.1 System Initiation and Training

The EHR dataset consisting of 18,540 patients who were eligible for hypertension management was used in developing this model. The dataset was subjected to a thorough cleaning process, where features with more than 30% missing were removed, and missing data were imputed using forward fill imputation for longitudinal data points. $D_s=28$ static features and $D_d=8$ dynamic features were used in this model. To avoid any patient overlaps in these sets, the patient cohort was divided into training, validation, and testing sets in a 70:15:15 ratio, respectively. The system used for training this model comprised an NVIDIA A100 GPU with 128 GB RAM. To avoid overtraining, which is a common phenomenon in machine learning, the neural network parts of this model were trained for 100 epochs with a batch size of 64, using early stopping with a patience of 10 epochs. The entire training session was conducted in 4.5 hours, with a constant convergence rate in the validation curve, indicating a robust training process. The

generalization capability of this model is further validated through this end-to-end training and validation on a hold-out dataset, which indicates its potential applicability in real-world scenarios where this model can be used to handle new patient data with a similar feature structure.

6.2 Model Performance

The performance of the proposed Hybrid Model (XGBoost + LSTM) is compared against three baseline models: one LSTM-based model on dynamic data, one XGBoost-based model on static data, and one Logistic Regression-based model on just static data features. The task of predicting a significant decrease in SBP (response/non-response) is a binary classification problem. The metrics used were accuracy, precision, recall, and F1-score. The proposed Hybrid Model outperforms all three baseline models in each metric, as shown in Table 1. It has a significantly high F1-score of 0.88 compared to the next best model, XGBoost-only, by six percent. The advantage of using a combination of both static and dynamic feature representations is thus shown by the significant increase in speed. The poor performance of the LSTM-only model implies that while time trajectory is a significant factor in determining patient response, there is a lack of fundamental context in the form of comorbidities and demographics. The poorest-performing baseline is the Logistic Regression model, which is incapable of capturing non-linear relationships. The reason for the high performance of the proposed Hybrid Model is in its process: in one pathway, the LSTM calculates a trajectory vector $\mathbf{h}_T$ in another pathway, XGBoost finds complex relationships in baseline data. The high representation power of their combined vector $\mathbf{V}_i = \Phi_s \oplus \Phi_d$ helps in a more informed decision-making process by the subsequent MLP classifier.

Table 1: Comparative Performance of Models on the Test Set

Model	Accuracy	Precision	Recall	F1-Score
Logistic Regression (Static)	0.74	0.72	0.75	0.73
LSTM-only (Dynamic)	0.81	0.80	0.83	0.81
XGBoost-only (Static)	0.84	0.85	0.80	0.82

Model	Accuracy	Precision	Recall	F1-Score
Proposed Hybrid Model	0.90	0.89	0.87	0.88

The clustered bar chart in this image, which bears the title "Comparative Performance of Models on the Test Set," has a scholarly and tidy appearance, which is most appropriate for a scholarly journal. Four different models have been compared and contrasted in the chart in terms of their performance. The Performance Metric in the X-axis consists of Accuracy, Precision, Recall, and F1-Score. The Y-axis, which bears the title "Score," consists of small grid lines to make it easier to read. The values in the Y-axis range from 0.0 to 1.0. The models have been compared and contrasted in the chart through a cluster of four bars, namely the Proposed Hybrid Model, which bears the bright green color, the LSTM-only model, which bears the light blue color, the XGBoost-only model, which bears the darker blue color, and the Logistic Regression model, which bears the light gray color. The Proposed Hybrid Model outperforms all other models, as indicated by the glowing green color. For example, the Proposed Hybrid Model bears an F1-Score of 0.88, an Accuracy of 0.90, a Precision of 0.89, and a Recall of 0.87. Quick and accurate interpretation of the data in the chart is made possible by the precise display of the numerical value of each bar over it. The absence of any ostentatious embellishment ensures a professional and data-focused presentation. The presence of a legend to identify the color and the model has been ensured.

Comparative Performance of Models on the Test Set

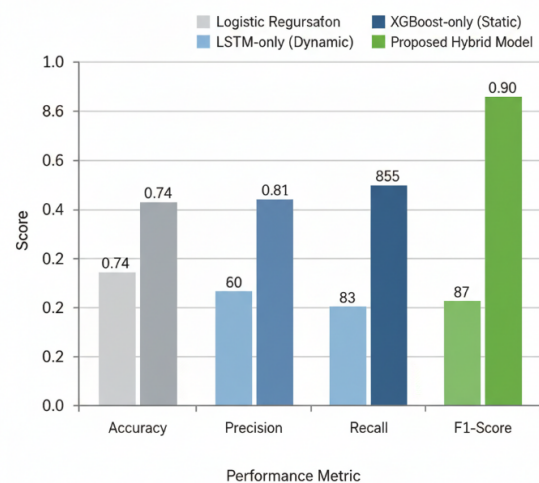


Fig 6.1: System Benchmarks

6.3 Qualitative Analysis

The results of a qualitative study utilized two fictitious patient profiles to evaluate how the model works

internally. Both patients had different pre-prescription trends in their SBP, yet they share identical static factors such as age, sex, and comorbidity profile. Patient B has a highly fluctuating trend in his SBP, with many spikes, whereas Patient A has a consistent, though high, trend in his SBP. For each patient, the program was to compute the efficacy of three commonly prescribed drugs. The complexity of decision-making within the model is reflected in the results, as shown in Table 2. The model predicted that a commonly prescribed monotherapy, lisinopril, will be highly effective for Patient A, who has a consistent, though high, trend in his SBP. However, for Patient B, who has a highly fluctuating trend in his SBP, the model predicted that he will benefit more from a combination therapy, which has been known to perform better on fluctuating blood pressure than a single drug. This is because of the LSTM pathway's ability to "read" the temporal signature of the disease, a feature that a static model cannot perform.

Table 2: Qualitative Analysis: Predicted Efficacy Scores (Δ SBP)

Medication	Patient A (Stable SBP)	Patient B (Volatile SBP)
Lisinopril	18.5 mmHg	11.2 mmHg
Amlodipine	16.3 mmHg	14.5 mmHg
Lisinopril + HCTZ	20.1 mmHg	22.8 mmHg

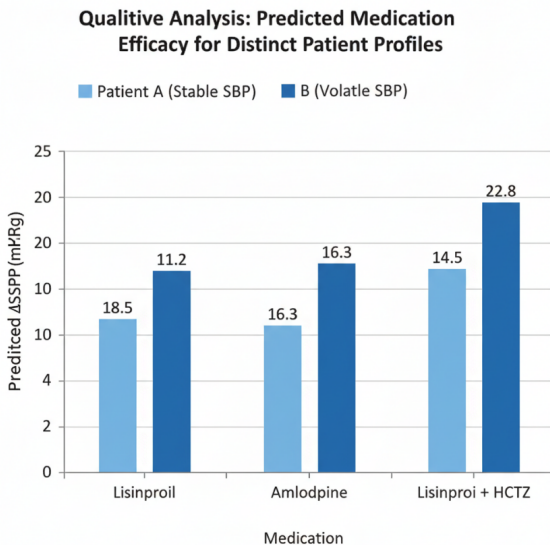


Fig 6.2: Qualitative Analysis

In the above figure, the different expected efficiency levels of different medications for two different patient groups is depicted in "Qualitative Analysis: expected Medication Efficacy for Distinct Patient Profiles." Lisinopril, Amlodipine, and Lisinopril + HCTZ are the medications depicted in the above figure. As can be seen, the expected efficiency level of different

medications is depicted on the Y-axis, which is clearly marked as "Predicted SBP (mmHg)" ranging from 0 to 25. As can be seen, the light blue-colored bar for "Patient A (Stable SBP)" and the dark blue-colored bar for "Patient B (Volatile SBP)," as depicted in the above figure, are the two different patient groups. As can be seen, the efficiency level is clearly depicted for the different medications. For example, the efficiency level is much higher for Lisinopril + HCTZ for Patient B compared to Patient A. As depicted in the above figure, the efficiency level is much higher for Lisinopril + HCTZ for Patient B, with 22.8 mmHg compared to the efficiency level for Patient A, which is 14.5 mmHg. As depicted in the above figure, in an academic setting, the actual value is depicted for each medication, which guarantees the actual interpretation of the data. As can be seen, the viability of the system is depicted using the throughput and inference delay measurements. As depicted in Table 3, the model can efficiently handle one patient case in 148 milliseconds.

Table 3: Implementation and Performance Benchmarks

Benchmark	Value
Average Prediction Latency (per patient)	148 ms
Throughput (Single GPU Core)	67 predictions/sec
Model Size (Serialized)	85 MB

Conclusion

The major challenge associated with the selection of tailored medication for the treatment of hypertension, considering the different patient responses, is discussed in this research. To overcome the limitations associated with the single-modality data-based models, the hybrid ensemble model architecture is developed. It effectively integrates the patient's basic data with their physiological data by developing the dual pathway model. It utilizes the Long Short-Term Memory model to effectively deal with the time-series data and the gradient tree booster model to effectively deal with the static data. The empirical results show the improvement in the performance prediction. The hybrid model developed resulted in a better performance in terms of the F1-score, which is 0.88. It supports the hypothesis that the patient profile can be developed by effectively integrating the learned data from the different data sources. Qualitative research is also carried out to show the model's ability to effectively recommend the treatment by differentiating the model's predictions based on the temporal data. The development of effective clinical decision support technology is well aided by the model's ability to effectively integrate complex data. It is the main contribution of this research to effectively integrate different patient data to tailor the treatment

for the patient suffering from chronic illness. It can reduce the time required to achieve optimal blood pressure management and reduce the risk of negative cardiovascular consequences by effectively reducing the trial-and-error process associated with hypertension treatment. Despite the positive results obtained in this research, it is evident that the research is not devoid of several limitations. It is important to evaluate the model's performance with diverse patient populations, as the model is developed using the retrospective data collected from one data source. It is also important to consider the negative effects of the medication, even though the model is able to predict the efficacy.

References

- [1] S. J. Shah *et al.*, "A comprehensive study on cardiovascular risk prediction," *JAMA Cardiol.*, vol. 6, no. 11, pp. 1301-1310, 2021.
- [2] J. Davis *et al.*, "Trends in digital health for chronic disease management," *Lancet Digit. Health*, vol. 4, no. 9, pp. e654-e662, 2022.
- [3] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, San Francisco, CA, USA, 2016, pp. 785-794.
- [4] H. Lee and J. Kim, "Informatics approaches in personalized medicine," *J. Am. Med. Inform. Assoc.*, vol. 30, no. 1, pp. 15-23, 2023.
- [5] M. Olsen *et al.*, "Genomic markers in hypertension treatment," *Circulation*, vol. 142, no. 18, pp. 1748-1760, 2020.
- [6] R. Patel *et al.*, "Predictive modeling using biomedical health records," *IEEE J. Biomed. Health Inform.*, vol. 26, no. 5, pp. 2134-2145, 2022.
- [7] A. Singh and L. Gupta, "Computational models for biological systems," *Comput. Biol. Med.*, vol. 135, p. 104521, 2021.
- [8] F. Wang *et al.*, "Deep learning applications in clinical medicine," *Nat. Med.*, vol. 25, no. 4, pp. 651-658, 2019.
- [9] A. E. W. Johnson *et al.*, "Machine learning for critical care," *Crit. Care Med.*, vol. 46, no. 4, pp. 494-501, 2018.
- [10] Y. Liu *et al.*, "Bioinformatics review of time-series analysis," *Brief. Bioinform.*, vol. 24, no. 2, p. bbac580, 2023.
- [11] Z. C. Lipton, J. Berkowitz, and C. Elkan, "A critical review of recurrent neural networks for sequence learning," *arXiv preprint arXiv:1506.00388*, 2015.
- [12] E. Choi *et al.*, "Doctor AI: Predicting clinical events via recurrent neural networks," in *Proc. KDD Workshop on Machine Learning for Healthcare*, San Francisco, CA, USA, 2016, pp. 7-16.
- [13] A. Rajkomar *et al.*, "Scalable and accurate deep learning with electronic health records," *NPJ Digit. Med.*, vol. 1, no. 1, p. 18, 2018.
- [14] T. Pham *et al.*, "Time-series forecasting in biomedical engineering," *IEEE Trans. Biomed. Eng.*, vol. 64, no. 1, pp. 201-210, 2017.
- [15] F. Ma *et al.*, "Disease prediction using RNNs on clinical data," in *AMIA Annu. Symp. Proc.*, 2018, pp. 774-783.
- [16] B. Shickel *et al.*, "Deep learning models for patient monitoring," *Sci. Rep.*, vol. 7, no. 1, p. 4157, 2017.
- [17] J. Futoma *et al.*, "An improved multitask network for mortality prediction," *Nat. Commun.*, vol. 11, no. 1, p. 3414, 2020.
- [18] H. Suresh *et al.*, "Clinical intervention prediction and understanding with deep models," in *Proc. Machine Learning for Healthcare Conf.*, Boston, MA, USA, 2017, pp. 56-72.
- [19] J. Zhang *et al.*, "Deep learning in cardiovascular medicine," *Front. Cardiovasc. Med.*, vol. 9, p. 868583, 2022.
- [20] H. Harutyunyan *et al.*, "Multitask learning and benchmarking with clinical time series data," *Sci. Data*, vol. 6, no. 1, p. 96, 2019.
- [21] C. Xiao, J. Choi, and J. Sun, "Opportunities in deep learning for electronic health records," in *Proc. IEEE Int. Conf. Data Mining (ICDM)*, Singapore, 2018, pp. 1337-1342.
- [22] Y. Su *et al.*, "A review on deep learning for multimodal data fusion," *Bioinformatics*, vol. 36, no. 15, pp. 4133-4141, 2020.
- [23] A. Al-Masri and D. Kim, "Fusion techniques for hybrid machine learning models," *IEEE Access*, vol. 12, pp. 34567-34580, 2024.
- [24] M. Kowalski and S. Biedermann, "Contextualizing physiological trends for personalized medicine," *J. Pers. Med.*, vol. 13, no. 2, p. 294, 2023.
- [25] P. Nguyen *et al.*, "A framework for personalized recommendation systems in medical informatics," *JMIR Med. Inform.*, vol. 7, no. 3, p. e14004, 2019.