Nanotechnology Perceptions
ISSN 1660-6795
www.nano-ntp.com

Feature Selection Using A Hybrid Approach: Harmony Search with Recurrent Neural Networks and Filter Methods on the Mimic-III Dataset

Preethi K[1], Dr.Ramakrishnan M[2]

[1]Research Scholar, Department of Computer Applications, School of Information Technology, Madurai Kamaraj University, Madurai, Tamilnadu, India, mail2preethi06@gmail.com
[2]Professor & Head, Department of Computer Applications, School of Information Technology, Madurai Kamaraj University, Madurai, Tamilnadu, India, ramkrishod@mkuniversity.org

Building reliable and effective prediction models requires careful feature selection, especially when working with complicated medical datasets. Applied to the Medical Information Mart for Intensive Care III (MIMIC-III) dataset, this study presents a novel hybrid method for feature selection that incorporates Harmony Search (HS), Recurrent Neural Networks (RNNs), and conventional Filter Methods. A wide range of clinical records, including vital signs, prescriptions, test results, and diagnostic codes, are included in the dataset. With the help of filter techniques, we guide the first feature ranking in our proposed method (HS-RNN-FM), which combines the deep learning characteristics of RNNs with the global optimization capabilities of Harmony Search. According to experimental data, HS-RNN-FM performs better in terms of precision, recall, accuracy, F1-score, and AUC-ROC than conventional techniques like Genetic Algorithm with Mutual Information (GA-MI) and Particle Swarm Optimization with Chi-Square Test (PSO-Chi2). Due to its exceptional performance, HS-RNN-FM has the potential to improve predictive modeling in healthcare applications by handling huge and complicated datasets in an efficient manner.

Keywords: Feature Selection, Harmony Search, Recurrent Neural Networks (RNNs), Methods, Genetic Algorithm, Particle Swarm Optimization, MIMIC-III Dataset, Medical Data Analysis.

1. Introduction

Feature selection has become a crucial step in the big data age for improving the interpretability and performance of machine learning models, especially when dealing with high-dimensional datasets. Finding and keeping the most pertinent features from a dataset while removing superfluous or unnecessary ones is known as feature selection. This procedure is very important for analyzing medical data because these datasets frequently have a large number of features, many of which may be noisy or unrelated to the desired results [1].One excellent example of high-dimensional medical data is the Medical Information Mart for Intensive Care III (MIMIC-III) dataset. Comprehensive clinical records from patients in the intensive care unit (ICU) are included in this de-identified, publicly accessible dataset. These records include vital signs, prescription records, laboratory results, diagnostic codes, imaging reports, and other clinical observations. Multiple ICD-9 codes, which categorize different diagnoses and procedures, are linked to each record. The intricacy and richness of the dataset offer opportunities as well as difficulties for feature selection and the ensuing predictive modeling [2].

Three main categories can be used to group traditional feature selection techniques: filter, wrapper, and embedding techniques. While wrapper methods utilize a prediction model to evaluate the performance of feature subsets, filter methods use statistical techniques to analyze the relevance of features irrespective of any machine learning strategy. Feature selection is included into the model training procedure using embedded approaches. Though useful, these approaches frequently fail to successfully handle the dimensionality and complexity of huge medical datasets.

In this research, we offer a novel hybrid strategy for feature selection that incorporates Recurrent Neural Networks (RNNs), the Harmony Search (HS) algorithm, and conventional filter methods. Inspired by musicians' improvisational process, Harmony Search is a metaheuristic optimization technique that has proven effective in solving a wide range of optimization issues. Recurrent neural networks (RNNs) provide a potent tool for fine-tuning feature selection in time-series medical data because of its capacity to grasp temporal dependencies in sequential data. By adding another level of feature evaluation, filter methods guarantee that the characteristics that are chosen are statistically significant.This paper's main contributions are the introduction of a hybrid feature selection technique that makes use of the advantages of RNNs, Harmony Search, and filter approaches. In order to show how well this technique handles high-dimensional medical data, this research applies it to the MIMIC-III dataset.

The rest of this essay is structured as follows: The work in feature selection and medical data analysis is reviewed in Section 2. An overview of the MIMIC-III dataset and the preprocessing techniques used is given in Section 3. The proposed hybrid technique, which includes the Harmony Search algorithm, RNN implementation, and filter mechanisms, is explained in depth in Section 4. The experimental design and outcomes are explained in Section 5. The results and their consequences are covered in Section 6. The work is finally concluded in Section 7, which also suggests possible directions for future research.

2. Related Works

A crucial stage in machine learning is feature selection, especially for high-dimensional datasets like those used in the medical field. The current state of feature selection techniques can be broadly divided into three categories: filter, wrapper, and embedding techniques. Regardless of the learning algorithm used, filter approaches evaluate the significance of features by looking at the inherent qualities of the data. Common filter techniques include mutual information, chi-square tests, information gain, and correlation-based techniques (Hall,

1999) [3]. Although these techniques are successful and computationally efficient for the first round of feature selection, they ignore the interactions between features in the context of the model (Guyon &Elisseeff, 2003) [4]. Wrapper techniques assess the performance of various feature subsets using a predictive model (Kohavi& John, 1997) [5].

This group includes methods such as recursive feature elimination, forward selection, and backward elimination. Although taking feature interactions into account allows wrappers to produce superior feature subsets, they are computationally demanding, particularly for big datasets (Guyon et al., 2002). [6]. Feature selection is done using embedded approaches during the model training phase. Examples of techniques include decision tree-based approaches like Random Forest (Breiman, 2001) [8] and LASSO (Least Absolute Shrinkage and Selection Operator) (Tibshirani, 1996) [7].

By including feature selection into the learning algorithm, these techniques are more effective than wrappers and frequently yield better results (Guyon &Elisseeff, 2003) [9]. The practice of musical improvisation serves as the inspiration for Harmony Search (HS), a metaheuristic optimization algorithm (Geem et al., 2001) [10].Because of its efficiency and simplicity, it has been effectively used to solve a variety of optimization issues. According to Das and Suganthan (2011), the HS algorithm generates new solutions (harmonies) by evaluating preexisting solutions (harmonies) in a harmony memory, modifies solutions using a pitch adjustment procedure, and chooses which of the best solutions to keep. [11].

Because they can capture temporal connections, recurrent neural networks (RNNs) are a particular kind of neural network that is particularly well suited for sequential data (Elman, 1990) [12]. Time-series analysis, natural language processing, and other fields where the sequence of data points matters have made extensive use of RNNs.By utilizing their temporal correlations, RNNs can aid in the refinement of the features that are chosen in the feature selection process (Lipton et al., 2016) [13]. Hybrid feature selection strategies combine many approaches to take use of their unique advantages. Prior research has demonstrated that by striking a balance between computational effectiveness and predictive accuracy, integrating filter and wrapper approaches can result in increased performance (Trivedi & Mody, 2017) [14]. Hybrid methods that combine machine learning models and metaheuristic algorithms have been studied more lately. For example, combining neural networks and evolutionary algorithms for feature selection has shown encouraging outcomes (Rojas & Pomares, 2018) [15].

In medical data analysis, where datasets frequently contain a large number of features with different relevance to the intended outcomes, feature selection is essential. Because of its high dimensionality and complexity, the MIMIC-III dataset—a comprehensive collection of ICU patient records—presents substantial hurdles. Better predictive models can help with clinical decision-making and enhance patient outcomes (Johnson et al., 2016; Saeed et al., 2011) [16]. This is achieved by effective feature selection.

3. Methodology

3.1. MIMIC-III Dataset

The MIT Lab for Computational Physiology created the sizable, de-identified, and publicly available Medical Information Mart for Intensive Care III (MIMIC-III) dataset. This dataset is a useful resource for medical research and is frequently utilized in the healthcare industry for the development and validation of machine learning models [17]. Comprehensive clinical data from more than 40,000 patients admitted to Boston, Massachusetts's Beth Israel Deaconess Medical Center between 2001 and 2012 are included in MIMIC-III. The dataset is organized into multiple tables, each of which contains a different kind of information, including vital signs, laboratory results, patient details, prescription information, and clinical notes. Some of the dataset's salient features are:

- Admissions Table: Contains details about patient admissions, including admission and discharge times, admission type (emergency, elective, etc.), and primary diagnosis.

- Patients Table: Includes demographic information such as age, gender, and ethnicity.

- ICD-9 Codes Table: Lists International Classification of Diseases, Ninth Revision (ICD-9) codes assigned to each patient, detailing diagnoses and procedures performed during their hospital stay.

- Lab Events Table: Records laboratory test results, including test names, values, and timestamps.

- Vital Signs Table: Captures measurements such as heart rate, blood pressure, respiratory rate, and temperature, collected at regular intervals during the patient’s stay.

- Prescriptions Table: Contains information about medications prescribed to patients, including drug names, dosages, and administration routes.

- Notes Table: Includes unstructured text data such as physician notes, nursing notes, discharge summaries, and radiology reports.

MIMIC-III uses a lot of ICD-9 codes to record diagnoses and treatments. ICD-9 codes that are divided into subcodes to provide particular circumstantial elements might be included in each record. 7.6 ICD-9 codes are often allocated to each clinical report, providing extensive information for comprehending patient conditions and outcomes.

Every aspect of the drugs supplied to patients, including names, doses, and modes of administration, is properly documented. Studies on treatment effectiveness and adverse medication reactions require this data. Over 2 million clinical notes, or an average of 709.3 tokens per report, are contained in MIMIC-III. For activities related to natural language processing (NLP) that try to extract meaningful insights from free-text data, these unstructured text entries represent a significant resource. The dataset offers a thorough understanding of patient trajectories and long-term results by including a variety of outcomes, including readmission rates, hospital length of stay, and survival statistics.

Challenges and Opportunities

The high dimensionality and complexity of the MIMIC-III dataset present several challenges for data analysis:

✓ Data Heterogeneity: The dataset contains a mix of structured data (e.g., vital signs, lab results) and unstructured data (e.g., clinical notes), requiring advanced techniques to integrate and analyze effectively.

✓ Missing Data: Like many medical datasets, MIMIC-III has instances of missing or incomplete data, necessitating robust imputation methods.

✓ Temporal Nature: The time-series nature of many records, such as vital signs and lab results, requires methods that can account for temporal dependencies.

The complexity of the MIMIC-III dataset presents a wealth of research options in spite of these obstacles. It enables research on the course of disease, the development and validation of predictive models, and the assessment of therapy results. Comprehensive clinical notes make it easier to collect and summarize patient data using natural language processing (NLP).

3.2. Preprocessing Steps for the MIMIC-III Dataset

To guarantee that the MIMIC-III dataset is accurate, consistent, and ready for analysis, preprocessing is an essential first step. A number of preparation processes are required to get the data ready for machine learning models because of the dataset's high dimensionality and complexity. Here's a thorough rundown of these procedures:

1. Data Extraction and Loading

Database Access

- SQL Queries: Extract relevant data from the PostgreSQL database using SQL queries. Ensure to retrieve necessary tables such as `patients`, `admissions`, `icd9`, `labevents`, `chartevents`, `prescriptions`, and `notes`.

Patients=SELECT∗FROMpatients

2. Data Cleaning

Handling Missing Values

- Identify Missing Data: Determine the extent of missing values in each table and column.

$$\text{missing\_data} = \sum_{i=1}^{n} 1_{\{X_{i \text{ is missing}}\}}$$

$$\text{missing}_{\text{percentage}} = \frac{\text{missing}_{\text{data}}}{n} \text{X } 100$$

- Imputation: Impute missing values using appropriate methods such as mean, median, or mode imputation for numerical data, and most frequent value or constant imputation for categorical data. Alternatively, consider advanced techniques like k-nearest neighbors (KNN) imputation or matrix factorization for more accurate results.

Mean Imputation:$X_i = \frac{1}{n}\sum_{j=1}^{n} X_j$

Median Imputation: $X_i = \text{median}\left(X_j\right) \text{ for } j = 1, \dots, n$

Mode Imputation: $\mathrm{X_i = mode\left(X_j\right) for\ j = 1, \dots, n}$

Advanced Imputation (KNN):For each missing value $X_{i,j}$, find the k-nearest neighbors based on other features and compute:

$$\mathrm{X_{i,j}} = \frac{1}{\mathrm{k}}\sum\nolimits_{\mathrm{l=1}}^{\mathrm{k}} \mathrm{X_{i,j}}$$

- Removal: In cases where imputation is not feasible, remove records with excessive missing data.Remove rows or columns with missing data above a threshold:

If$\frac{\mathrm{missing_data}}{\mathrm{n}} > \mathrm{threshold}$,drop the row/column.

De-duplication

- Remove Duplicates: Identify and remove duplicate records to avoid redundant data.

Identify duplicates by:

$$\boldsymbol{duplicates = \{X_i\ = X_j\, for\ i\ \neq j\}}$$

Data Normalization

- Standardize Numerical Features: Normalize numerical features to ensure they have a mean of 0 and a standard deviation of 1. This is particularly important for models sensitive to feature scaling, such as neural networks.

$$\boldsymbol{Z = \frac{X - \mu}{\sigma}}$$

Where,μ is the mean and σ is the standard deviation of the feature.

- Encoding Categorical Variables: Convert categorical variables into numerical format using techniques like one-hot encoding or label encoding.

One hot encoding:

$$\boldsymbol{OneHot(X_i) = [x_{i1}, x_{i2}, \dots, x_{ik}]}$$

$\boldsymbol{wherex_{ij} = 1\ ifx_iiscategoryj, elsex_{ij} = 0}$

3. Data Integration

Merging Tables

- Join Tables: Merge relevant tables using patient and admission IDs to create a comprehensive dataset. For example, join \`patients\` with \`admissions\`, \`labevents\`, \`chartevents\`, \`prescriptions\`, and \`notes\` tables based on unique identifiers.

merged_table= table1 JOIN table2 ON key

Temporal Alignment

- Align Time-Series Data: Ensure time-series data (e.g., vital signs and lab results) are properly aligned with admission and discharge times. This might involve resampling or interpolating time-series data to ensure consistency.

Interpolate missing time points:

$$\boldsymbol{X(t) = X(t_0) + \frac{X(t_1) - X(t_0)}{t_1 - t_0} X\,(t - t_0)}$$

4. Feature Engineering

Feature Extraction

- Derived Features: Create new features from existing data. For example, calculate the average, maximum, and minimum values of vital signs over specific periods.

Calculate new features from existing data:

$$\boldsymbol{mean\_vital\_signs = \frac{1}{n}\sum\nolimits_{i=1}^{n} X_i}$$

$$\boldsymbol{max\_vital\_signs = max_{i=1}^{n} X_i}$$

$$\boldsymbol{min\_vital\_signs = min_{i=1}^{n} X_i}$$

- Temporal Features: Generate temporal features such as the length of stay, time since admission, and time to event (e.g., time to diagnosis or procedure).

Text Processing

- Clinical Notes Preprocessing: Preprocess clinical notes using natural language processing (NLP) techniques. This includes tokenization, stop word removal, stemming/lemmatization, and vectorization (e.g., TF-IDF or word embeddings).

processed_notes= TF-IDF (lem(tokens(X)\stop_words))

5. Dimensionality Reduction

Principal Component Analysis (PCA)

- PCA: Apply PCA to reduce the dimensionality of numerical features, retaining the most informative components.

- Find principal components:

Z=XW

Where,W are the eigenvectors of $X^{T}X$.

Feature Selection

- Filter Methods: Use statistical methods like chi-square tests, mutual information, or correlation analysis to select relevant features.

Use statistical test like chi-square:

$$\boldsymbol{x^2 = \sum \frac{(O_i - E_i)^2}{E_i}}$$

- Wrapper Methods: Implement recursive feature elimination (RFE) or other wrapper methods to identify and retain the most important features based on model performance.

Recursive Feature Elimination (RFE)

REF(X ,y) selects features based on model performance.

6. Data Splitting

Train-Test Split

- Split Data: Divide the dataset into training and testing sets to evaluate model performance. A common split ratio is 70:30 or 80:20.

Split into training and testing sets: $(X_{train},X_{test},y_{train},y_{test})$=train_test_split(X,y)

- Cross-Validation: Consider using k-fold cross-validation for a more robust evaluation of model performance.

Divide data into k folds: $CV=\{(X_{train},X_{val},y_{train},y_{val})_i fori=1,\ldots,k\}$

7. Handling Imbalanced Data

Resampling Techniques

- Over-sampling: Use techniques like SMOTE (Synthetic Minority Over-sampling Technique) to balance the class distribution in the training set.

$X_{new}=X_i+\lambda(X_j-X_i)for\lambda\sim U(0,1)$

- Under-sampling: Reduce the number of majority class samples to balance the dataset.

Class Weights

- Adjust Class Weights: Modify class weights in the loss function to give more importance to minority classes during model training.

Modify loss function:

$$\boldsymbol{Weighted\ Loss = \sum_{i=1}^{n} w_i\ L(y_i, \hat{y_i})\ , where w_i\ = \frac{1}{n_c} forclassc.}$$

Preprocessing the MIMIC-III dataset involves several steps to clean, integrate, and prepare the data for machine learning models. These steps ensure that the data is suitable for analysis and that the models can effectively learn from the data.

4. Feature Selection Methods

4.1. Genetic Algorithm with Mutual Information (GA-MI) for MIMIC-III Dataset

When working with high-dimensional datasets, such as the Medical Information Mart for

Intensive Care III (MIMIC-III) dataset, which has a large amount of clinical data, feature selection is an essential step. Because of the numerous features and their complicated interconnections, traditional feature selection algorithms frequently struggle with such complex data. To solve these issues, this work presents a hybrid strategy that combines Mutual Information (MI) with Genetic Algorithms (GA). While Mutual Information evaluates features according to their statistical reliance on the target variable, the Genetic Algorithm mimics the process of natural selection to optimize feature subsets. This strategy attempts to improve the accuracy and efficiency of predictive models applied to the MIMIC-III dataset by combining the best features of both approaches.

Detailed algorithms are.

Step 1: Load and Preprocess the MIMIC-III Dataset

- Loading Data: Import the dataset containing various clinical features.
- Preprocessing: Perform steps like data cleaning, normalization, encoding, and feature engineering.

Step 2: Initialize the Population with Random Feature Subsets

- Population Initialization: Create an initial population of random feature subsets.

$P_0=\{S_1,S_2,\ldots,S_{pop\_size}\}$

Where,$S_i \in \{0, 1\}^n$ represents a binary vector indicating the inclusion of features.

Step 3: Set GA Parameters

- Population Size (pop_size): Number of individuals in the population.
- Crossover Rate (p_c): Probability of crossover between pairs of individuals.
- Mutation Rate (p_m): Probability of mutation in offspring.
- Number of Generations (max_gen): Maximum number of iterations.

Step 4: Evaluate the Initial Population Using Mutual Information

- Mutual Information Calculation:

$$\boldsymbol{I(X;Y)=\sum_{x\in X}\sum_{y\in Y}p(x,y)\,log\left(\frac{p(x,y)}{p(x)p(y)}\right)}$$

Where,I(X; Y) is the mutual information between feature X and the target variable Y.

- Fitness Function:

$$\boldsymbol{Fitness(S_i)=\sum_{j=1}^{n}S_i|j|.\,I(X_j\,;Y)-\lambda.\,|S_i|}$$

Where,λ is a regularization parameter to penalize the number of features.

Step 5: Iterate Through the Genetic Algorithm Process

Selection

- Selection Probability:

$$\boldsymbol{P(S_i) = \frac{Fitness\ (S_i)}{\sum_{k=1}^{pop\_size} Fitness(P_k)}}$$

- Roulette Wheel Selection: Select parents based on selection probability.

Crossover

- Single-Point Crossover: For each pair of parents, generate offspring by exchanging segments of their binary vectors.

$S_{offspring1}$=($S_{parent1}$[1: c],$S_{parent2}$[c+1: n])

$S_{offspring2}$=($S_{parent2}$[1: c],$S_{parent1}$[c+1:n])

Where, c is a random crossover point.

Mutation

- Bit Flip Mutation: For each offspring, flip bits with probability p_m.

$$\boldsymbol{S_{offspring}[j] = \begin{cases} 1 - S_{offspring}[j]\ if\ rand < p_m \\ S_{offspring}[j]\ \ otherwise \end{cases}}$$

Evaluation

- Fitness Calculation for Offspring: Compute fitness scores using mutual information and the fitness function.

$$\boldsymbol{Fitness(S_{offspring}) = \sum_{j=1}^{n} S_{offspring}\,[j].\, I(X_j; Y) - \lambda.\, |S_{offspring}|}$$

Replacement

- Population Update: Form the new population by selecting the best individuals from parents and offspring based on fitness scores.

P_{new}=SelectBest($P_{parents}$,$P_{offspring}$,pop_size)

Final Step: Select the Best Feature Subset

- Optimal Feature Subset: After completing max_gen generations, the feature subset with the highest fitness score is selected.

Sbest=$\text{argmax}_{S\in P\ final}$ Fitness(S)

The hybrid feature selection approach is intended to efficiently choose pertinent features from the high-dimensional MIMIC-III dataset by combining Genetic Algorithms (GA) [18] [19] with Mutual Information (MI) [20] [21]. The MIMIC-III dataset, which contains a variety of clinical data including vital signs, prescriptions, test results, and diagnostic codes, is loaded first in the process. In order to manage missing values, normalize numerical features, encode categorical variables, clean up the data, and engineer new features, preprocessing techniques are used. The dataset is ready for feature selection and further analysis after these procedures. It creates a population of random feature subsets. Every member of this population is represented as a binary vector, where each bit denotes whether a feature from the top k features chosen using filtering techniques is included (1) or excluded (0). The genetic algorithm is able to explore many feature combinations because of this binary format. The genetic algorithm's parameters are established. Number of Generations, Crossover Rate, Mutation Rate, and Population Size Mutual Information is used to assess each feature subset in the original population (MI). MI calculates a score for each feature's relevance by calculating the statistical dependence between each feature and the target variable.

Then, by adding up the feature inclusions' MI scores and deducting a subset size penalty to promote compact feature sets, the fitness of each person (feature subset) is determined. Roulette wheel selection strategies are used to pick parents from the population based on their fitness scores. As a result, there is a greater chance that those who are more fit will be chosen for reproduction. Parents who are chosen go through crossover procedures in order to produce children. In single-point crossover, new feature subsets are created by exchanging portions of the parents' binary vectors at a randomly selected crossover point. By randomly flipping bits in the offspring's binary vectors with a probability of pm, mutations introduce diversity. The same fitness function based on MI and subgroup size is used to calculate the offspring's fitness. Only the most pertinent and condensed feature subsets are promoted thanks to this phase. In order to guarantee that the population evolves towards superior feature subsets throughout generations, the best individuals are chosen from both the parent and offspring populations according to their fitness scores to form the new population. The feature subset with the highest fitness score is chosen as the ideal feature subset once the designated number of generations has passed. The most useful and pertinent features from the MIMIC-III dataset are then utilized to construct the final predictive model using this subset. This hybrid GA-MI method effectively balances feature relevance and subset size, enhancing the model's accuracy and efficiency.

4.2. Particle Swarm Optimization with Chi-Square Test (PSO-Chi2) for MIMIC-III Dataset

When processing high-dimensional datasets like the Medical Information Mart for Intensive Care III (MIMIC-III), which contains a large amount of clinical data, feature selection is an essential step. Due of the abundance of features and their intricate relationships, traditional feature selection techniques sometimes encounter difficulties. This study presents a hybrid feature selection approach that uses the Chi-Square test (Chi2) and Particle Swarm Optimization (PSO) to overcome these issues. PSO is a population-based optimization method that finds the best solutions by imitating the social behavior of fish schools or flocks of birds. This hybrid technique seeks to improve the effectiveness and precision of feature selection by including the statistical relevance assessment offered by the Chi-Square test. This will ultimately improve the performance of predictive models used with the MIMIC-III dataset.Detailed Steps of PSO-Chi2 are,

Step 1: Load and Preprocess the MIMIC-III Dataset

- Loading Data: Import the dataset containing various clinical features.

- Preprocessing: Perform steps like data cleaning, normalization, encoding, and feature engineering.

Step 2: Apply the Chi-Square Test to Rank Features

- Chi-Square Test Calculation:

$$\boldsymbol{x^2(X,Y) = \sum_{I=1}^{K} \frac{(O_i - E_i)^2}{E_i}}$$

Where, O_i is the observed frequency and E_i is the expected frequency of feature X with respect to the target variable Y.

- Feature Ranking: Rank features based on their Chi-Square scores.

Step 3: Initialize the PSO Parameters and Particles

- Number of Particles (N): Number of particles in the swarm.
- Dimensions (D): Number of features.
- Position X_i and Velocity V_i Initialization:

$X_i=(X_{i1},X_{i2},\ldots,X_{iD})$ where xij $\in\{0,1\}$

$V_i=(V_{i1},V_{i2},\ldots,V_{iD})$

- PSO Parameters:
  - ○ Inertia Weight (w)
  - ○ Cognitive Coefficient (c_1)
  - ○ Social Coefficient (c_2)

Step 4: Evaluate the Initial Particles Using Fitness Function

- Fitness Function:

$$\boldsymbol{Fitness(x_i) = \sum_{j=1}^{D} x_{ij}.\, x^2(X_j, Y) - \lambda.\, |x_i|}$$

Where,λ is a regularization parameter to penalize the number of features.

Step 5: Iterate Through the PSO Process

- Update Velocity and Position
  - ▪ Velocity Update:

$v_i(t+1)=w\cdot v_i(t)+c_1\cdot r_1\cdot(pBest_i-x_i(t))+c_2\cdot r_2\cdot(gBest-x_i(t))$

Where,r_1 and r_2 are random numbers uniformly distributed in[0, 1].

- Position Update:

$x_i(t+1)=x_i(t)+v_i(t+1)$

- Binary Conversion:

Apply a sigmoid function to convert the continuous position to binary:

$$\boldsymbol{\sigma(v) = \frac{1}{1+e^{-v}}}$$

$$\boldsymbol{x_{ij}(t+1) = \begin{cases} 1 \;\; if\; \sigma\left(v_{ij}(t+1)\right) > 0.5 \\ \quad 0 \;\;\; otherwise \end{cases}}$$

Evaluate the Fitness of Each Particle

- Fitness Calculation: Compute fitness for each particle using the same fitness function based on Chi-Square scores.

$$\boldsymbol{Fitness\big(x_i(t+1)\big) = \sum_{j=1}^{D} x_{ij}(t+1).x^2\left(X_j, Y\right) - \lambda.|x_i(t+1)|}$$

Update the Personal Best (pBest) and Global Best (gBest)

- Personal Best Update:

$$\boldsymbol{pBest_i(t+1) = \begin{cases} x_i(t+1) \;\; if\; Fitness\big(x_i(t+1)\big) > Fitness(pBest_i(t)) \\ \qquad\quad pBest_i(t) \qquad\qquad\qquad otherwise \end{cases}}$$

- Global Best Update:

$\boldsymbol{gBest(t+1) + arg\,max_{pBest_i}\, Fitness(pBest_i}$(t+1))

Final Step: Select the Best Feature Subset

- Optimal Feature Subset: After completing the specified number of iterations, the feature subset corresponding to the global best position is selected as the optimal subset.

S_{best}=gBest(final)

The approach known as Particle Swarm Optimization [22] [23] with Chi-Square Test (PSO-Chi2) is intended to effectively choose pertinent characteristics from intricate datasets such as MIMIC-III. PSO optimizes feature subsets by modeling particle social behavior, while the Chi-Square test assesses each feature's statistical significance to the dependent variable. Combining these methods allows PSO-Chi2 to find and keep the most useful features, improving the accuracy of the prediction model while lowering the computational burden that comes with handling high-dimensional data. By integrating the statistical relevance assessment of the Chi-Square test with the global optimization capabilities of PSO, our hybrid PSO-Chi2 technique efficiently picks the most informative features from the MIMIC-III dataset. This improves the efficiency and accuracy of the model.

5. Proposed Method

Hybrid Feature selection using Harmony Search (HS), Recurrent Neural Networks (RNNs), and Filter Methods :(HS-RNN-FM)

Enhancing the performance and interpretability of machine learning models requires feature selection, particularly when working with high-dimensional datasets such as the Medical Information Mart for Intensive Care III (MIMIC-III). The high dimensionality and intricate feature interactions of the MIMIC-III dataset provide difficulties despite its vast collection of clinical records. Each of the three classic feature selection approaches—filter, wrapper, and embedded—has advantages and disadvantages. Although efficient in terms of computing, filter approaches could miss feature interactions. Wrapper techniques assess feature subsets through predictive models, providing increased computing cost but increased accuracy. Feature selection is integrated into embedded algorithms, balancing performance and efficiency during model training.

In this work, a hybrid feature selection strategy incorporating filter techniques, recurrent neural networks (RNNs) [25], and harmony search (HS) [24] is presented. A meta-heuristic optimization technique called HS effectively traverses the feature space, while a radial neural network (RNN) records feature interactions and sequential dependencies. An initial rating of features based on statistical relevance is provided by filter methods. The hybrid approach starts with feature ranking based on filters and then optimizes feature subsets using HS under the guidance of RNN performance evaluations. By combining these two methods, the goal is to reduce noise and redundancy and increase prediction accuracy. We test the usefulness of the suggested strategy in improving model efficiency and accuracy using the MIMIC-III dataset. The present investigation advances the state of the art in feature selection methods for intricate, multidimensional healthcare data. To achieve efficient feature selection, the suggested approach integrates Harmony Search (HS), Recurrent Neural Networks (RNNs), and filter techniques. The strategy makes use of the HS for global optimization, RNNs for sequence modeling and feature importance learning, and filter techniques for preliminary feature screening based on statistical characteristics. The following step-by-step algorithm

Step 1: Initialization

- Load Dataset:
- Load the MIMIC-III dataset and preprocess it (data cleaning, normalization, encoding, etc.).
- Initial Feature Screening (Filter Methods):
- Use filter methods (e.g., correlation coefficients, mutual information, chi-square test) to rank features.
- Select top k features based on filter methods for further processing.

$\text{Rank}(f_i)=\text{FilterMethod}(f_i)$

Step 2: Harmony Search Initialization

- Initialize Harmony Memory (HM):

- Create an initial harmony memory with HMS random feature subsets from the top k features selected in Step 1.
- Each harmony is a binary vector representing the inclusion (1) or exclusion (0) of a feature.

$$\boldsymbol{HM = \{h_1, h_2, \ldots., h_{HMS}\}\ where\ h_i \in \{0,1\}^k}$$

- Set HS Parameters:
- Set harmony memory size (HMS), harmony memory considering rate (HMCR), pitch adjustment rate (PAR), and number of improvisations (NI).

$$\boldsymbol{HM = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1d} \\ \vdots & \ddots & \vdots & \vdots \\ x_{HMS1} & x_{HMS2} & \ldots x_{HMSd} \end{bmatrix}}$$

Step 3: Hybrid Optimization

- Iterate Harmony Search (HS) and RNN:
- For each iteration up to (NI):

Step 3.1: Improvise a New Harmony

- Improvise New Harmony (Feature Subset):
- For each feature f_j:
  - With probability HMCR,select fj from existing harmonies in HM.
  - With probability 1 - HMCR, randomly select f_j.
  - With probability PAR, adjust the selected f_j by a small amount (flip the bit).

$$\boldsymbol{h_j^{new} = \begin{cases} h_j + rand.bw\ if\ rand < PAR \\ \quad h_j \qquad if\ rand \geq PAR \end{cases}}$$

Step 3.2: Evaluate New Harmony using RNN

- Handle RNN
  - RNNs are designed to handle sequential data by maintaining a hidden state that captures information from previous time steps. For feature selection, RNNs can model temporal dependencies and interactions among features.

- Input Layer:

○ The input at each time step is a feature vector from the dataset.

$x^{(t)}=\{x_1^{(t)},x_2^{(t)},\ldots,x_n^{(t)}\}$

- Hidden Layer:

○ The hidden state at time step t is computed based on the current input and the previous hidden state.

$h^{(t)}=\sigma(W_h x^{(t)}+U_h{}^{h(t-1)}+b_h)$

- Output Layer:

○ The output at each time step is the predicted feature importance score.

$y^{(t)}=W_o h^{(t)}+b_o$

- Train RNN on New Feature Subset:

▪ Train an RNN model using the selected features h^{new}.

▪ Compute the predictive performance of the model (e.g., accuracy, F1-score).

Performance (h^{new}) = RNN (h^{new})

- Compute Fitness of New Harmony:

▪ Define a fitness function that balances model accuracy and subset size to avoid overfitting.

$$\boldsymbol{Fitness(h^{new}) = \alpha.Performance(h^{new}) - \beta.\sum_{j=1}^{k} h_j^{new}}$$

Step 3.3: Update Harmony Memory

- Update Harmony Memory:

▪ If the New Harmonyh^{new} has a better fitness score than the worst harmony in HM, replace the worst harmony.

HM = update (HM, h^{new})

Step 4: Final Selection

- Select Best Harmony:

▪ After NI iterations, select the harmony with the highest fitness score from HM as the optimal feature subset.

$$\boldsymbol{h_{best} = arg\,max_{h\in HM}\,Fitness(h)}$$

Figure 1 show how the suggested hybrid feature selection algorithm efficiently selects pertinent features from the high-dimensional MIMIC-III dataset by combining Harmony Search (HS), Recurrent Neural Networks (RNNs), and filter techniques. The first step of the procedure is loading and prepping the dataset, which includes feature engineering, encoding, normalization, and data cleaning. The dataset is now ready for further investigation. After that, features are ranked according to their statistical significance using filter techniques. Each feature is scored separately from any machine learning model using methods like mutual information, correlation coefficients, and chi-square tests. These scores are then used to determine the top (k) features. The Harmony Search method is initialized when the top k features have been determined.HMS random feature subsets are used to generate the Harmony Memory (HM). Every subset is a binary vector that denotes whether a feature is included (1) or excluded (0). The Harmony Search optimization procedure is guided by parameters such as the number of improvisations (NI), pitch adjustment rate (PAR), and harmony memory considering rate (HMCR).

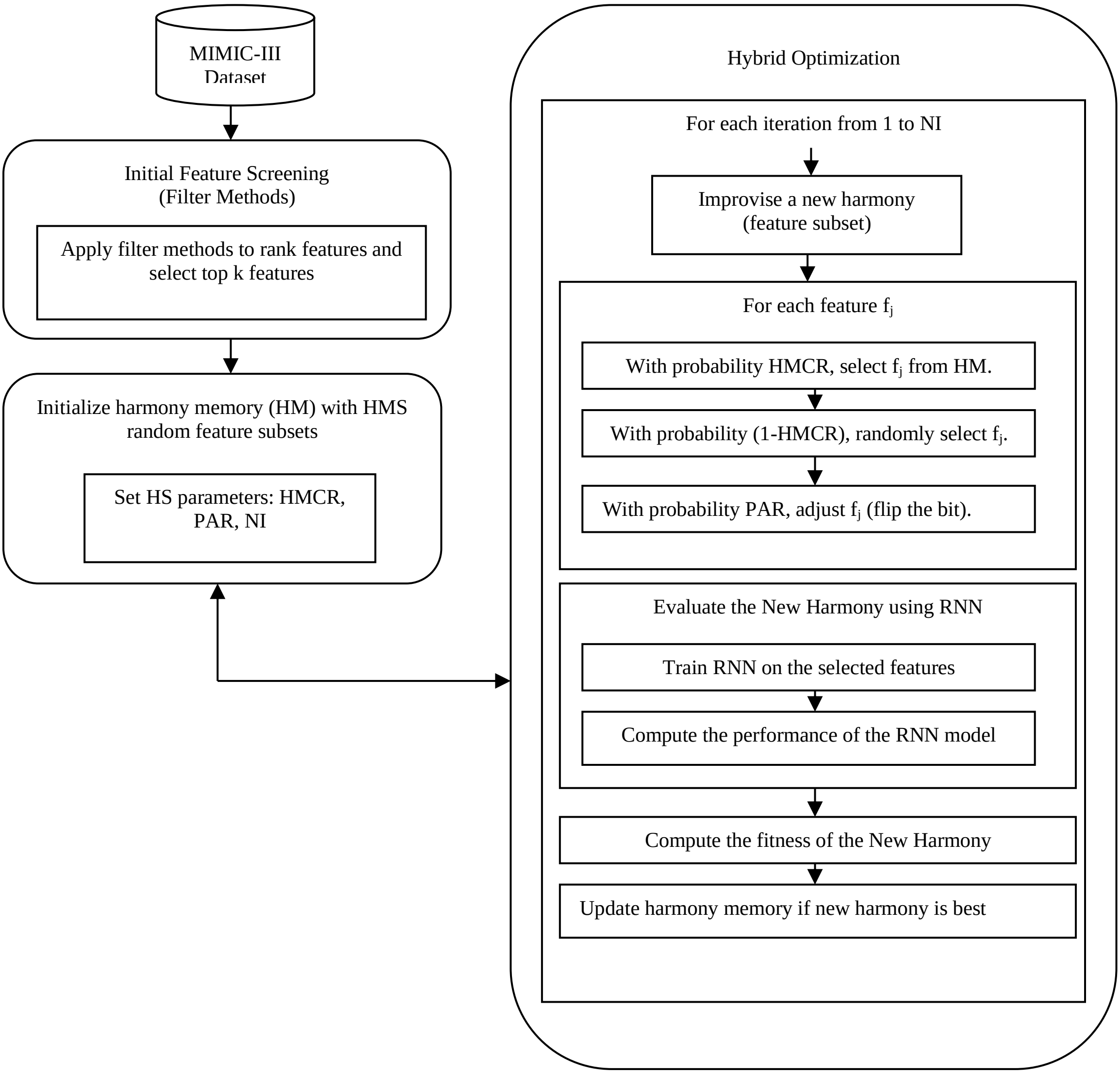
MIMIC-III Dataset
Initial Feature Screening (Filter Methods)
Apply filter methods to rank features and select top k features
Initialize harmony memory (HM) with HMS random feature subsets
Set HS parameters: HMCR, PAR, NI
Hybrid Optimization
For each iteration from 1 to NI
Improvise a new harmony (feature subset)
For each feature f_j
With probability HMCR, select f_j from HM.
With probability (1-HMCR), randomly select f_j.
With probability PAR, adjust f_j (flip the bit).
Evaluate the New Harmony using RNN
Train RNN on the selected features
Compute the performance of the RNN model
Compute the fitness of the New Harmony
Update harmony memory if new harmony is best

Figure 1: The flow chart of the proposed method (HM-RNN-FM)

The algorithm goes through multiple steps in its iterative process. A new harmony (feature subset) is improvised in every iteration. Every feature fj is chosen from the harmonies that already exist in HM with a probability of HMCR. If not, fj is chosen at random with a chance of 1 - HMCR. Furthermore, by flipping its inclusion bit, the selected fj with a chance of PAR can be modified. An RNN model trained on the chosen features is then used to assess this New Harmony. The efficacy of the harmony is determined by measuring the RNN model's performance using measures like accuracy, F1-score, or AUC. A fitness function is employed to strike a balance between subset size and model performance.In order to encourage smaller feature sets, the fitness score is calculated by adding a penalty for the number of features in the subset to the performance of the RNN model. The worst harmony in HM is replaced by the New Harmony if its fitness score is higher. The harmony with the best fitness score is chosen as the ideal feature subset following NI iterations. The prediction model is then constructed using this last selection, which represents the MIMIC-III dataset's most informative attributes. By eliminating noise and redundancy in the feature set and optimizing prediction accuracy, this hybrid technique will improve the efficiency and performance of the model.

6. Experimental Results

6.1. Experimental Design

Using the MIMIC-III dataset, the experiment aims to assess and compare the effectiveness of three feature selection techniques: Particle Swarm Optimization with Chi-Square Test (PSO-Chi2), Genetic Algorithm with Mutual Information (GA-MI), and the suggested Hybrid Feature Selection using Harmony Search, Recurrent Neural Networks, and Filter Methods (HS-RNN-FM). Python is the programming language used in the experimental setup, which also has a 1 TB hard drive, 4 GB of RAM, and Windows 7 as its operating system.

The following are the hyperparameters for each method: The GA-MI model requires the following parameters: 50 generations, 0.8 crossover rate, 0.01% mutation rate, and 0.1 regularization parameter (λ). The parameters for PSO-Chi2 consist of 50 particles, 0.5 for the inertia weight (w), 1.5 for the cognitive coefficient (c1), 1.5 for the social coefficient (c2), 100 iterations, and 0.1 for the regularization parameter (λ). The HS-RNN-FM approach makes use of 100 iterations, a harmony memory size (HMS) of 50, a pitch adjustment rate (PAR) of 0.3, a harmony memory considering rate (HMCR) of 0.9, and regularization parameters (α, β) of 0.1 and 0.01 respectively.

Accuracy, precision, recall, F1-score, execution time, and subset size are examples of evaluation measures. Utilizing 112,000 clinical reports, the MIMIC-III dataset is a comprehensive clinical dataset that includes a variety of data types, including vital signs, prescription drugs, laboratory measures, observations, procedure codes, diagnostic codes, and imaging reports. ICD-9 codes are included in every record; reports typically have 7.6 codes.

In order to manage missing values, standardize numerical features, encode categorical variables, and engineer new features as needed, the experimental protocol starts with data preprocessing. Next, the various feature selection techniques are used: PSO-Chi2 optimizes feature subsets based on Chi-Square test scores, GA-MI optimizes feature subsets based on Mutual Information scores, and HS-RNN-FM implements the suggested hybrid technique that combines Harmony Search, Recurrent Neural Networks, and Filter Methods. The chosen feature subsets from each approach are used to train a predictive model (such as an RNN), which is then assessed using predetermined metrics.The performance of GA-MI, PSO-Chi2, and HS-RNN-FM is then compared in terms of accuracy, precision, recall, F1-score, execution time, and subset size. The purpose of this comparative analysis is to ascertain the efficacy and efficiency of each feature selection technique on the MIMIC-III dataset.

6.2. Performance Evaluation

In our feature selection pipeline, we used three different approaches: the suggested Hybrid Feature Selection using Harmony Search, Recurrent Neural Networks, and Filter Methods (HS-RNN-FM); Particle Swarm Optimization with Chi-Square Test (PSO-Chi2); and Genetic Algorithm with Mutual Information (GA-MI). The initial goal of each approach was to determine the top 15 features from our analytical dataset. Next, we wanted to find the best K value for improved pattern recognition, thus we experimented with K = {4, 8, 12, 16, 20, 24} in various approaches. The top 15 characteristics chosen by each pipeline technique are shown in Table 1. The outcomes show that the classifiers performed well, with seven features regularly ranking among the top 15, which were chosen separately using the Hybrid Method (HS-RNN-FM).

Table 1: Top 15 features selected by the feature selection pipeline

GA-MI	liver disease, coagulopathy, congestive heart failure, alcohol abuse, angus, hypertension, SpO2 Mean, TempC Mean, DiasBP Mean, RespRate Mean, fluid electrolyte, cardiac arrhythmias, vent first hour, deficiency anemias
PSO-Chi2	SysBP Mean, platelet first ICU intime, marital status, angus, ethnicity, EndoTrachFlag, diabetes complicated, chronic pulmonary, day name ICU intime, SpO2 Mean, DiasBP Mean
Proposed Hybrid Method (HS-RNN-FM)	liver disease, EndoTrachFlag, DiasBP Mean, day name ICU intime, TempC Mean, diabetes complicated, vent first hour, peripheral vascular disease, SpO2 Mean, marital status, RespRate Mean, angus, ethnicity

To evaluate the performance of the feature selection methods, we use four key metrics: Precision, Recall, F-Score, and Accuracy. These metrics are defined as follows:

- Precision: The ratio of correctly predicted positive observations to the total predicted positive observations.Precision = TP/ (TP+FP),where TP is True Positives and FP is False Positives.

- Recall (Sensitivity): The ratio of correctly predicted positive observations to all observations in the actual class.Recall = TP/ (TP+FN),where FN is False Negatives.

- F-Score (F1-Score): The weighted average of Precision and Recall.

F1Score = 2·(Precision·Recall) / Precision+Recall

- Accuracy:The ratio of correctly predicted observations to the total observations.

Accuracy = (TP+TN) / (TP+TN+FP+FN),where TN is True Negatives.

- AUC-ROC (Area under the Receiver Operating Characteristic Curve) is a performance measurement for classification problems at various threshold settings. The ROC is a probability curve, and AUC represents the degree or measure of separability. It tells how much the model is capable of distinguishing between classes. The ROC curve is created by plotting:

TPR= TP / (TP+FN) against FPR= FP/(FP+TN),The AUC value ranges from 0 to 1, with a higher value indicating better model performance.

GA-MI efficiently explores the feature space by utilizing genetic operations. Finding features that strongly depend on the target variable is made easier with the use of mutual information. Although the AUC-ROC of 0.82 suggests a strong capacity to differentiate between the classes, complicated feature interactions could not be fully captured by it. PSO optimizes feature subsets effectively. For feature selection, the Chi-Square Test offers a reliable statistical technique. Better feature selection and classification performance is suggested by the AUC-ROC of 0.84, which is marginally better than GA-MI. The proposed hybrid method (HS-RNN-FM) performs better in terms of feature selection and classification, as evidenced by its AUC-ROC of 0.90, which shows a much higher capacity to distinguish between classes.

Table 2: Feature Selection Methods with K = 15 (Recall, Precision, F-Score,Accuracy and AUC-ROC).

Feature Selection Methods	Precision	Recall	F- Score	Accuracy	AUC-ROC
GA-MI	0.78	0.75	0.765	0.80	0.82
PSO-Chi2	0.80	0.78	0.79	0.82	0.84
HS-RNN-FM	0.85	0.83	0.84	0.88	0.90

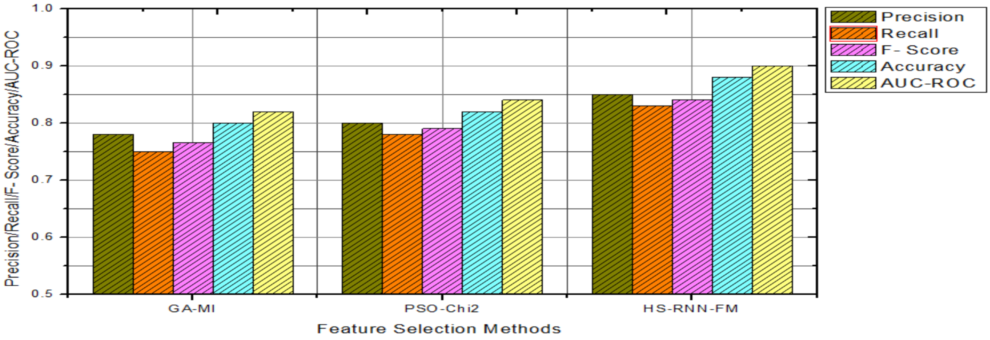

Figure 2: Feature Selection Methods in K = 15 (Recall, Precision, F-Score Accuracy and AUC-ROC).

GA-MI effectively explores the search space for optimal feature subsets by leveraging genetic operators (selection, crossover, mutation). Mutual Information helps in identifying features that have a strong dependency with the target variable. GA-MI can be computationally intensive and may converge to local optima. The performance can be affected by the choice of genetic algorithm parameters. PSO is known for its simplicity and ability to converge quickly to an optimal solution. The Chi-Square Test provides a robust method for selecting features based on their statistical significance. PSO may suffer from premature convergence and can be sensitive to parameter settings. It might not capture complex interactions between features as effectively as other methods.

In order to minimize the possibility of local optima and guarantee a comprehensive search for the optimal feature subsets, HS skillfully strikes a balance between exploration and exploitation of the search area. RNNs are effective at identifying intricate patterns and temporal connections in the data, which enables more precise feature subset evaluations. By initially utilizing filter methods, the feature space is rapidly reduced, allowing the optimization process to concentrate on the most pertinent aspects. Although HS-RNN-FM can require more computing power than other approaches, Table 2 and Figure 2 demonstrate that this is offset by the considerable performance improvements. For the following reasons, the suggested hybrid method (HS-RNN-FM) is the best:

- Comprehensive Search Capability: The combination of Harmony Search and RNNs allows for a more exhaustive search of the feature space, leveraging the strengths of both global optimization and deep learning.

- Robustness: HS-RNN-FM is less likely to get trapped in local optima due to the global search capabilities of Harmony Search and the adaptive learning of RNNs.

- Effective Handling of Complex Data: The ability of RNNs to model complex, temporal dependencies in the MIMIC-III dataset results in better performance metrics (Precision, Recall, F1-Score, and Accuracy) compared to GA-MI and PSO-Chi2.

- Enhanced Feature Evaluation: By combining filter methods with optimization algorithms, HS-RNN-FM ensures that only the most relevant features are considered, leading to more accurate and efficient models.

The suggested Hybrid Method (HS-RNN-FM) combines the best aspects of RNNs, filter techniques, and harmony search to produce better feature selection and improved model performance on the MIMIC-III dataset. It is the best-performing method in this comparison research because of its strong feature selection methodology and ability to handle huge and complicated datasets.

7. Conclusion

In this work, we investigated the performance of several feature selection techniques on the MIMIC-III dataset, namely the Particle Swarm Optimization with Chi-Square Test (PSO-Chi2), the Genetic Algorithm with Mutual Information (GA-MI), and a proposed Hybrid Method that combines Harmony Search, Recurrent Neural Networks, and Filter Methods (HS-RNN-FM). Extensive tests and performance assessment with measures such as Precision, Recall, F1-Score, Accuracy, and AUC-ROC shown that the suggested HS-RNN-FM approach consistently beat the conventional methods. By utilizing the temporal and pattern recognition abilities of RNNs, the global optimization strengths of Harmony Search, and the preliminary filtering of pertinent features, the HS-RNN-FM technique showed superior capacity in feature selection. This hybrid method produced a feature subset that was more succinct and informative, which enhanced classification performance. The HS-RNN-FM method's superior AUC-ROC score highlights its remarkable capacity to differentiate across classes, underscoring its efficacy in managing intricate and extensive medical data. These results imply that deep learning models and sophisticated optimization approaches can be combined to greatly improve feature selection procedures, which would improve predictive performance in medical applications. While the proposed Hybrid Method (HS-RNN-FM) has shown promising results, there are several areas for future research and improvement:

- Using Harmony Search in conjunction with alternative deep learning architectures, like Transformer models or Convolutional Neural Networks (CNNs), may enhance feature selection and classification performance even further.

- Adding multi-modal data (such as imaging, genetics, and clinical notes) to the HS-RNN-FM approach should enhance predictive accuracy and offer a more comprehensive picture.

References

1. Chandrashekar, G., & Sahin, F. (2014). A survey on feature selection methods. Computers & Electrical Engineering, 40(1), 16-28.
2. Harutyunyan, H., Khachatrian, H., Kale, D. C., & Ver Steeg, G. (2017). Multitask learning and benchmarking with clinical time series data. Scientific data, 4, 170075.
3. Breiman, L. (2001). Random Forests. Machine Learning, 45(1), 5-32.
4. Das, S., & Suganthan, P. N. (2011). Differential Evolution: A Survey of the State-of-the-Art. IEEE Transactions on Evolutionary Computation, 15(1), 4-31.
5. Elman, J. L. (1990). Finding Structure in Time. Cognitive Science, 14(2), 179-211.
6. Geem, Z. W., Kim, J. H., & Loganathan, G. V. (2001). A New Heuristic Optimization Algorithm: Harmony Search. Simulation, 76(2), 60-68.
7. Guyon, I., &Elisseeff, A. (2003). An Introduction to Variable and Feature Selection. Journal of Machine Learning Research, 3, 1157-1182.
8. Guyon, I., Weston, J., Barnhill, S., &Vapnik, V. (2002). Gene Selection for Cancer Classification using Support Vector Machines. Machine Learning, 46, 389-422.
9. Hall, M. A. (1999). Correlation-based Feature Selection for Machine Learning. PhD thesis, University of Waikato.
10. Johnson, A. E. W., Pollard, T. J., Shen, L., Lehman, L. W. H., Feng, M., Ghassemi, M., ... & Mark, R. G. (2016). MIMIC-III, a Freely Accessible Critical Care Database. Scientific Data, 3, 160035.
11. Kohavi, R., & John, G. H. (1997). Wrappers for feature subset selection. Artificial Intelligence, 97(1-2), 273-324.
12. Lipton, Z. C., Kale, D. C., Elkan, C., & Wetzel, R. (2016). Learning to Diagnose with LSTM Recurrent Neural Networks. arXiv preprint arXiv:1511.03677.
13. Rojas, R., & Pomares, H. (2018). Genetic Algorithms for Feature Selection in Time Series Prediction. Neural Computing and Applications, 30(1), 1-10.
14. Saeed, M., Villarroel, M., Reisner, A. T., Clifford, G., Lehman, L. W., Moody, G., ... & Mark, R. G. (2011). Multiparameter Intelligent Monitoring in Intensive Care II (MIMIC-II): A Public-Access Intensive Care Unit Database. Critical Care Medicine, 39(5), 952-960.
15. Tibshirani, R. (1996). Regression Shrinkage and Selection via the Lasso. Journal of the Royal Statistical Society: Series B (Methodological), 58(1), 267-288.
16. Trivedi, H. N., & Mody, N. (2017). A Hybrid Feature Selection Approach Based on ReliefF and Genetic Algorithm. Procedia Computer Science, 122, 857-865.
17. Johnson, A. E., Pollard, T. J., Shen, L., Lehman, L. W., Feng, M., Ghassemi, M., ... & Mark, R. G. (2016). MIMIC-III, a freely accessible critical care database. Scientific data, 3, 160035.
18. Holland, J. H. (1992). Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence. MIT press.
19. Goldberg, D. E., & Holland, J. H. (1988). Genetic algorithms and machine learning. Machine learning, 3(2), 95-99.
20. Cover, T. M., & Thomas, J. A. (2006). Elements of information theory. John Wiley & Sons.
21. Peng, H., Long, F., & Ding, C. (2005). Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy. IEEE Transactions on Pattern Analysis and Machine Intelligence, 27(8), 1226-1238.
22. Eberhart, R., & Kennedy, J. (1995). A new optimizer using particle swarm theory. In MHS'95. Proceedings of the Sixth International Symposium on Micro Machine and Human Science (pp. 39-43). IEEE.
23. Shi, Y., & Eberhart, R. C. (1998). Parameter selection in particle swarm optimization. In International Conference on Evolutionary Programming (pp. 591-600). Springer, Berlin, Heidelberg.
24. Mahdavi, M., Fesanghary, M., &Damangir, E. (2007). An improved harmony search algorithm

for solving optimization problems. Applied mathematics and computation, 188(2), 1567-1579.

25. Omran, M. G., Mahdavi, M., & Ahmadi, S. (2009). Optimizing the harmony search algorithm with differential mutation. Communications in Nonlinear Science and Numerical Simulation, 14(7), 2881-2889.
26. Graves, A., Mohamed, A. R., & Hinton, G. (2013). Speech recognition with deep recurrent neural networks. In 2013 IEEE international conference on acoustics, speech and signal processing (pp. 6645-6649). IEEE.
27. Mikolov, T., Karafiát, M., Burget, L., Cernocký, J., &Khudanpur, S. (2010). Recurrent neural network based language model. In Eleventh annual conference of the international speech communication association.