DOI: 10.21917/ijsc.2026.0563

EFFECTIVE RNN BASED FEATURE INTERACTION MODEL USING CLOUD DATABASES

R. Sridevi
Department of Computer Science, Tagore Government Arts and Science College, India

Abstract
Identification of Feature interaction is the challenging task particularly for high dimensional database in the research field of feature selection which plays a major role in Machine Learning. "Effective RNN based Feature Interaction Model using Cloud Databases" is a Recurrent Neural Network based approach in the Cloud environment which detects positive feature interaction. The proposed RNN based Feature Interaction model consists of Feature Interaction Identifier and Redundant Feature Interaction Remover. The Feature Interaction Identifiers identifies the Feature Interaction pair in the given dataset which is retrieved from Cloud database. The Redundant Feature Interaction Remover removes the duplicate Feature Interaction Pairs which are produced from the Feature Interaction Identifiers and achieves Positive Feature Interaction. The four Medical Databases obtained from AWS Cloud is used to evaluate the performance of the proposed Feature Interaction Model. The experimental and evaluation results obtained on these Cloud databases show that a proposed RNN based Feature Interaction Model performs feature interaction with significant improvement in classification accuracy better than conventional Feature Selection Models.

Keywords:
Feature Interaction, Cloud database, Recurrent Neural Network

1. INTRODUCTION

The rapid growth in computer advancement tends to the accumulation of massive quantities of high-dimensional data. Machine learning techniques provide computers with the ability to learn these high-dimensional data efficiently and the learning performance is highly increased only with the relevant data. A commonly used technique to retrieve relevant data is Feature selection and this has been a challenging research area in Machine Learning, text categorization, pattern recognition, and data mining. The process of feature selection involves the subset selection of input variables by removing features with less or no predictive capacity. The advantage of feature selection for learning lies in a reduction of feature subset to realize effective learning, improvement of predictive accuracy, and reduction of execution time. Unfortunately, the accuracy of performance predictions may be degraded when considering features only in isolation. The classification accuracy of feature selection algorithm can be highly enhanced when one or more features interact with each other in the feature subset. The features that are visible to be irrelevant or imperceptibly relevant with the class individually, but when it joined with other features, it may highly associate to the target class are called Interacting features. Achieving feature interaction is an exigent task in feature selection.

Recent advancement in Machine learning makes the computers to learn automatically and improve their performance without being explicitly programmed. The development of unsupervised learning using neural networks that can teach themselves is the major focus of Machine learning. Machine learning models are easily overfitted with real world dataset which contains high dimensional input features. A neural network is a computational model that aims to simulate the functional part of biological neural networks. A neural network approach can be followed over traditional programming to find solution to the problems that do not have algorithmic solution or the available solution is too difficult to be found. The neural network demonstrates the behaviour of the Feed Forward network and recurrent network. Most of the machine learning methods has been used Feed Forward Neural Network and Recurrent Neural network for feature interaction.

Cloud computing is an on-demand self-service computing technology that can be metered based on utility and consumption of computing resources. A cloud database is a database that has been created and accessed from cloud environment. Many extensive database services are offered by Database-as a Service model in Cloud Computing. In the proposed model cloud data base offers the services of providing required datasets during feature interaction.

The rest of the paper is organized as follows. Section 2 explores the various existing feature interaction-based feature selection methods and their limitations and also the different areas where the Recurrent Neural Network is used with respect to the feature selection domain. Section 3 proposes "Effective RNN based Feature Interaction Model using Cloud Databases" (RFIM) and its components. Section 4 derives algorithm for the proposed model. Section 5 demonstrates the experimental results which are implemented using MATLAB. Performance Analysis of the proposed model is discussed in Section 6. Finally, Section 7 summarizes the proposed model.

2. RELATED WORK

2.1 FEATURE INTERACTION BASED FEATURE SELECTION METHODS

Feature Selection facilities in machine learning aims to select relevant features and to remove irrelevant ones. A particular feature may be considered irrelevant based on its association with the class but it may become very significant if combined with other features. The importance of Feature Interaction lies in the ability of feature selection methods to identify interacting features that improves the classification accuracy by providing additional information during classification. Usha and Anuradha [1] reviews and compares the performance of various feature selection methods using standard classifier models. Zeinab Noroozi et al. [2] investigate the effect of sixteen feature selection techniques on seven machine learning algorithms for heart disease prediction using Cleveland heart disease dataset.

Achieving feature interaction in the domain of feature selection is one of the most challenging tasks. Some of the existing conventional feature selection methods [3]-[7] support 2-way, 3-way and multi-way interaction among features. By performing feature interaction, classification accuracy is improved. A generalized n-way interaction among the features is achieved by using KIRKWOOD superposition approximation which observes both negative and positive interactions for a supervised learning field [3]. In order to overcome the feature ordering problem, a filter algorithm INTERACT [4] is devised for finding interacting features. It employs symmetrical uncertainty as a heuristic to rank features and it effectively reduces a large number of features. By using sequential forward search technique, an "Interaction Weight based Feature Selection algorithm" (IWFS) [5] identifies and filter out the irrelevant, redundant features. In IWFS, the 3-way interactions were also observed by calculating the interaction weight factor between the features. I.A. Gheyas et al. [6] introduced a multi-way interaction among features using "Simulated Annealing Genetic Algorithm (SAGA)" to select optimal feature subsets efficiently by avoiding being trapped in a local minimum of simulated annealing using strong local search of greedy algorithms. Using a coverRatio metric, "FOIL Rule based Feature Selection (FRFS)" [7] combines the features present in the antecedents of all FOIL rules and forms a candidate feature subset that removes redundant features and preserves interactive ones.

In the research field of feature interaction, by focusing on the high dimensional data, feature selection methods [8]-[10] performs feature interaction using deep learning. The first and second order features interactions [10], [11] higher-order feature interactions [8], [13], [14] are achieved through neural network framework.

The drawback of mRMR feature selection method, which does not remove redundant features from the already selected features is overcome by a two-stage feature selection algorithm [11] which derives mRMR(Maximum Relevance and Minimum Redundancy) for first –order incremental feature selection and combine mRMR with feature selectors. The first and second order features interactions are captured by the "Factorization Machines" (FM) proposed by S. Rendle [10]. FMs capture all interactions between variables using factorized parameters and can effectively compute these interactions even in highly sparse problems. In order to capture higher-order feature interactions using a multilayer perceptron (MLP) in a non-linear method, K. Ding et al. [8] proposes "A feature interactions-aware GNN (Graph Neural Network)" framework for learning node representations on feature sparse graphs. A high-order feature interaction approach [12] for Fine-Grained Visual Classification using shared GNNs improves computational efficiency by constructing both inter- and intra-region graphs and, in this approach, intra-region graphs explore high-dimensional convolutional features to capture finer details within specific regions of an object. W. Song et al. [13] propose a method "AutoInt" based on multi- head self-attentive neural network, which perform low-order interactions as well as high-order feature interactions and can deal large-scale-high-dimensional sparse data.

The positive and negative feature interactions are identified in the methods [9], [14] for an unsupervised learning domain. The Contribution Score of each neuron is computed only in single backward pass and not possible to apply recurrent neural network in "DeepLIFT(Deep Learning Important FeaTures)" method [14] which backpropagating the contributions of all neurons in the network to every feature of the input for attaining the output prediction of a neural network on a particular input. G.Liu et al. [9] propose "DeepResolve", a gradient ascent based method for deep convolution models of genome function which visualize individual feature contribution in decision making and capable of handling complex feature contribution patterns also. This method also discovers important features, negative features and feature interactions in the shared biological mechanism. Based on order of feature interaction, the existing feature selection methods can be categorized as shown in Table 1.

Table.1. Categorization of Feature Interaction Methods

Method	Order of Feature Interaction
FRFS	First-order Induction Learner
SAGA, IWFS	3-way & Multi way interaction
FM	Low Order (1st & 2nd order) Feature Interaction
INTERACT, GNN, AutoInt	High Order Feature Interaction

2.2 EXISTING NEURAL NETWORK APPROACH IN FEATURE SELECTION METHODS

Some of the existing feed-forward neural network-based framework [15]-[18] detect statistical interaction, higher-order interaction between the features and are able to detect biological sequences also. A maximum entropy method [15] provides mathematical tools which interprets deep neural networks and retrieves learned features from input data of biological sequence using feed-forward neural networks. Statistical interaction of any order or form captured by a feed forward neural network by examining its weight matrices is detected in "Neural Interaction Detection" framework [16]. The "Compressed Interaction Network (CIN)" which learns high-order feature interactions explicitly and generates vector- wise level feature interactions implicitly [17]. In order to combine the strength of both wide linear model which memorizes sparse feature interactions using cross-product feature transformation and deep neural network model which generalizes previously unseen feature interactions using low dimensional embedding's, Cheng et al. [18] proposes "Wide & Deep learning" Framework.

Since Hopfield networks are dynamical models of auto associative memory, existing Hopfield Neural network-based model namely a kernel-HNN [19] can be able to handle higher dimensional feature space efficiently. The characteristics of network being able to generate arbitrary shapes without prior knowledge makes the Hopfield neural network applied in the area like Image classification, medical applications [20],[21]. Even in noisy background, a technique is applied for binary object extraction in real time using "quantum bi-directional self-organizing neural network" architecture proposed by Konar et al. [21]. A self-supervised learning algorithm recommended in this architecture, extracted images with great precision but fails to extract binary object in unsupervised learning environment.

In order to improve the accuracy of Hopfield Neural Network, D.J. Hemanth et al. [20] propose a “Modified Hopfield Neural Network (MHNN)” for abnormality classification from human retinal images. H.Yoon et al. [22] presents a Neural Network approach, which is called an Algorithm Learning Based Neural Network (ALBNN), to produce new relevant features and to improve classification accuracy by integrating Feature Selection and classification procedures. Based on type of Neural Network applied, the existing feature selection methods can be categorized as shown in Table.2.

Table.2. Categorization of Feature Selection Methods

Feature Selection Methods	Applied type of Neural Network
AutoInt	Self- Attentive Neural Network
DEEPLIFT, ALBNN	Recurrent Neural Network
Kernel-HNN	Hopfield Neural Network
CIN, Neural Interaction Detection framework	Feed Forward Neural Network

From the literature survey, it is understood that most of the existing feature selection methods [4] [7] supports 2-way and multi-way interaction but fails to perform high-order feature interaction. Since the proposed model aims to perform feature interaction for the datasets retrieved from Cloud database, conventional feature interaction-based feature selection methods [3],[6] cannot be used during classification which degrades the classification accuracy. Though the existing methods [8]-[14] supports high-order feature interaction using neural network framework, fails to perform feature interaction using Recurrent Neural Network which has the advantage of having recursive path for fine tuning in the classification accuracy.

Most of the conventional feed-forward neural network-based framework [19]-[21] fail to handle feature interaction using Neural Network which is based on unsupervised learning. Since the knowledge is encoded in the network during design and not learnt, Hopfield Neural Network outputs settle down to a steady state. The existing Hopfield neural network-based methods [20],[21] efficiently used in Medical and Image Classification area but fails to perform feature interaction.

Thus, a new model “Effective RNN based Feature Interaction Model using Cloud Databases” (RFIM) is proposed that performs feature interaction for the dataset taken from the Cloud databases using Recurrent Neural Network. An Algorithm is also derived for this model which accepts features from the cloud dataset as an input, performs feature interaction and identifies positive feature interactions. The efficiency of the model has been improved by removing the redundant feature pair using relevancy score of the individual feature which in turns enhance the classification accuracy.

3. ARCHITECTURE

From the literature survey, the Feature Selection methods like [6],[7] identifies feature interaction using different measures but does not support high-order feature interaction. Since conversion of features into encoding vectors results in features with millions of dimensions, supervised learning methods [23],[24] are not applicable for high- dimensional features. Most of the machine learning models is easily overfitted with high dimensional input features. High- order feature interactions are crucial for a good performance. Some machine learning algorithms [8],[13] supports high-order feature interaction but fails to identify positive feature interaction in the cloud-based environment.

The existing Feature Selection Methods mostly concentrate on Feature Selection but fail to achieve Feature Interaction in cloud-based environment. On the other hand, some of the existing Feature Selection Methods gives better results for High-order feature interactions but fail to support dynamically changing Cloud Database. Eventually, there is no way of fine tuning on the results obtained in these methods by applying recursive paths. However, in order to improve the classification accuracy using feedback path available in Recurrent Neural Network and to support dynamically growing Cloud Database, an efficient Feature Interaction model based on Recurrent Neural Network should emerge. Most of the existing neural network methods [19],[21],[22] handle higher dimensional feature space effectively but fails to perform feature interaction in the cloud-based environment. To overcome the limitations, present in the conventional feature selection methods, “RNN based Feature Interaction Model” (RFIM) is proposed which performs feature interaction in the given dataset and effectively identifies positive feature interaction by removing redundant feature pair.

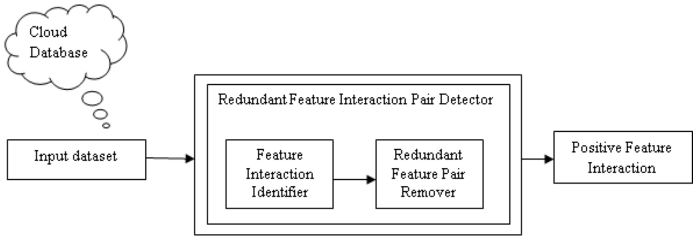

Fig.1. Structure of RFIM

The RFIM is a machine learning model which is implemented by using Recurrent Neural Network. The structure of the proposed model is a Recurrent Neural Network based model which is dynamic in nature. The proposed model also supports unsupervised learning so that datasets with unlabeled classes can also be efficiently performs feature interaction. The Fig.1 demonstrates the structure of RFIM which interacts with the features of dataset from cloud database. It comprises Feature Interaction Identifier (FII) and Redundant Feature pair Remover (RFR). Feature Interaction Identifier detects the Feature Interaction for the dataset which is retrieved from cloud database. The Feature Interaction Identifier uses Recurrent Neural Network to identifies Feature interactions between all pairs of features and detect positive Feature Interaction between these identified feature pairs. A set of feature interaction pair identified using Feature Interaction Identifier is given as an input to Redundant Feature pair Remover. Let the training dataset (TD) from the Cloud Database consists of Schema $T=\{F_1,F_2,\ldots,F_n\}$ where $\{F_1,F_2,\ldots,F_n\}$ are features and $\{X_1,X_2,\ldots,X_n\}$ be the instance of schema T which are used as input vectors for Feature Interaction Identifier. Using Interaction Earned measure, Feature Interaction Identifier detects the Positive Interaction feature pairs from the given dataset which is from cloud database. During the training, Redundant Feature pair Remover detects the redundant feature interaction pairs and removes those redundant pairs using relevancy score of individual feature pair and produces the positive feature interaction.

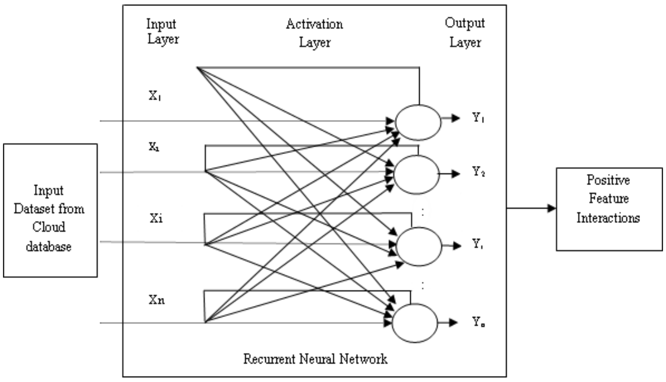

Fig.2. Architecture of RFIM

The proposed RNN based Feature Interaction Model intends to achieve interaction between the features using Recurrent Neural Network. The Fig.2 shows the architecture of RNN based Feature Interaction Model.

The RFIM is a Dynamic Neural Network which consists of input layer, activation layer and output layer. Input vectors are derived from the Cloud Database which is used as features. Let Schema $M = \{R_1, R_2 \ldots R_n\}$ from Cloud Database contain a set of attributes or features $\{R_1, R_2 \ldots R_n\}$. Let $X = \{X_1, X_2 \ldots X_n\}$ be the instance of schema M which are used in the proposed model as input vectors. The set of target vectors in the Output layer are $Y = \{Y_1, Y_2 \ldots Y_n\}$. The proposed model gains additional information once when the positive Feature Interaction occurs during Feature Interaction. Alternatively, negative Feature Interaction in RFIM causes the output with less feature dependency. The input layer fans out the instance of schema $X=\{X_1, X_2 \ldots X_n\}$ to the activation layer and 'α' is the predefined constant which denotes learning rate of RFIM.

Let FIM be the NxN Feature Interaction Matrix of the Recurrent Network and the feature's interaction earned is mapped as neurons in the NxN network. The Feature Interaction Matrix implements the concept of Recurrent Neural Network with feed-back connections. Since the interaction earned from feature 'i' to feature 'j' is same as that of from feature 'j' to feature 'i', the Feature Interaction Matrix is symmetric in nature (i.e.,) $w_{ij} = w_{ji}$. Similarly, the diagonal elements in Feature Interaction Matrix are zero (i.e.,) $w_{ii} = 0$, indicates that there is no interaction gain when the i^{th} feature interact with itself. The activation layer consists of Feature Interaction Matrix (FIM) which holds the interaction earned between every pair of features present in the input vectors.

3.1 FEATURE INTERACTION IDENTIFIER (FII)

The FII identifies the interaction earned between every pair of features present in the input vectors using Feature Interaction Matrix. Let $H(F_i,;T_C)$ and $H(F_j,;T_C)$ be the entropy to measure the uncertainty of Feature F_i and F_j from the input vector F with respect to Target Class TC respectively. Then the Interaction-Earned (IE) for individual feature F_i which is the measure of dependence of the Feature F_i with respect to the Target Class TC and it is calculated using Eq.(1).

$$\mathrm{IE}(F_i,;T_C) = H(F_i,) - H(F_i,;T_C) \quad (1)$$

The Eq.(1) measures the Interaction-Earned (IE) by the individual features present in the training set before the Feature Interaction is performed using FII. (i.e.,) the Interaction Earned (IE) calculates the interaction of individual feature with respect to the target class TC for the given Input Feature Vector before the training process.

Let $H(F_i,F_j;TC)$ denote the Joint entropy of F_i and F_j with respect to Target class TC. The measure of interaction between the Features F_i and F_j is denoted by $IE(F_i,F_j;TC)$ and it is calculated with respect to the target class TC using Eq.(2).

$$IE(F_i,F_j;TC) = H(F_i,;TC) + H(F_j,;TC) - H(F_i,F_j;TC) \quad (2)$$

where $IE = \begin{cases} +1 & \text{indicates positive feature interaction,} \\ -1 & \text{indicates negative feature interaction.} \end{cases}$

The Interaction-Earned (IE) calculates for the Feature F_i and F_j in the Input Feature Vector is used to denote the connection strength in a network and it is represented by the elements of a Feature Interaction Matrix, FIM. The positive value of IE indicates that there is additional information gained when Feature F_i interacts with F_j which leads into positive Feature Interaction. Eventually, negative value of IE implies the negative Feature Interaction which tends to redundant of information.

The weight w_{ij} (from node 'i' to node 'j') of the FIM is the Interaction-Earned (IE) between two features namely 'i' and 'j' across all patterns and it is calculated using Eq.(3).

$$w_{ij} = IE(F_i,F_j;TC) \text{ where } i,j \text{ varies from 1 to } n \text{ \& } i \neq j \quad (3)$$

The output layer $Y = \{Y_1, Y_2 \ldots Y_m\}$ are the set of target vectors where Y_i denotes the i^{th} feature's maximum Interaction- Earned (IE) in 2-tuple format along with feature pair and it is calculated using Eq.(4).

$$(Y_{i,1}, Y_{i,2}) = \left(F_i, F_j, \max(w_{ij})\right) \quad (4)$$

where $i, j \in \{1, 2, \ldots, n\}, i \neq j, \text{and } w_{ij} > 0$

The output Y_i calculated using Eq.(4) contains the highest Interaction-Earned value of the i^{th} feature along with the pair of features among the positive interaction earned with all other remaining features. The weights of the negative interaction earned from the i^{th} feature to j^{th} feature (i.e.,) w_{ij}<0 are ignored during the calculation of Yi.

3.2 REDUNDANT FEATURE PAIR REMOVER (RFR)

In order to remove redundant feature pair which has the same IE values, previously stored Y_i values are compared with the currently calculated Y_i values using Redundant Feature pair Remover and based on the relevancy score of feature pairs Y_i value is updated. The relevancy score of feature pair $<F_i,F_j>$ with respect to the target class TC is calculated using the Eq.(5).

$$Rel\_Score(F_i,F_j;TC) = 1\text{-}(H(F_i;TC) + H(F_j;TC) \quad (5)$$

Once when the relevant score value for currently selected feature pair which is calculated using Eq.(5) is greater than that of previously stored feature pair value then the previously stored feature pair value is updated by the current feature pair by Redundant Feature pair Remover. Thus, the updated output vector Y produced by RFR does not contain any redundant feature pair.

At time $t > 0$, the weights of output neuron 'i' and its adjacent neurons should be corrected using Eq.(6) and learning rate 'α' is increased at each epoch.

$$w_{ij}(new) = w_{ij}(old) + \alpha(t)[IE_{ij}(t) - w_{ij}(old)] \tag{6}$$

In the training process, the given Input dataset from cloud database is partitioned into two sets: a training set and a testing set. During training, the value of w_{ij} is updated to the newly calculated IE value only when it is lower than the new value of IE in the Feature Interaction Matrix for the training set. The newly updated Feature Interaction Matrix is send back to the network again and the output Y_i for the neuron 'i' is calculated for this newly updated Feature Interaction Matrix. The process of updating the Feature Interaction Matrix and calculating value for output Y for the training set is recursively done until the network reaches into the stable state. At each epoch, the Interaction-Earned values for the features in the network are updated synchronously (i.e.,) all the nodes are updated concurrently by freezing the network whereas the next value is computed across the network. The energy function which is used to determine the stable state of the Recurrent Network is defined using Eq.(7).

$$E = \sum_{i=1}^{m} Y_{i,2} \tag{7}$$

The energy function calculated using Eq.(7) will decrease whenever state of neuron changes in a stable network. The energy change ΔE is calculated when neuron 'i' has changed state from Yi(old) to Yi(new) using Eq.(8).

$$\Delta E = E(Y_i(new)) - E(Y_i(old)) \tag{8}$$

The network trained using training set reaches into stable state when ΔE calculated using Eq.(8) becomes minimized.

At the end of training, the output vector Y contains the feature pairs which have the highest Interaction-Earned values without any redundant feature pairs using Redundant Feature pair Remover. A testing set of input dataset is applied on this Recurrent Network which is trained using the training set of given dataset in order to test the convergence of the correct output vector.

3.3 ALGORITHM

In continuation with the architecture demonstrates in the Fig.2, the following algorithm is developed to interact with the input dataset which is obtained from Cloud database offered by Amazon's AWS and produce Output Vector $Y = \{Y_1, Y_2 \ldots Y_m\}$ which has positive Feature Interaction.

Input: D - Dataset from Cloud Database n – Number of Features

Output: Y vector (i.e.) $Y = \{Y_1, Y_2 \ldots Y_m\}$ with positive Feature Interaction.

Method:

Step 1: Let Schema $M = \{R_1, R_2 \ldots R_n\}$ consists of set of features $\{R_1, R_2 \ldots R_n\}$ from D.

Step 2: Let $X=\{X_1,X_2 \ldots X_n\}$ be the input vector which is the instance of M.

Step 3: At time t=0, set the learning rate $\alpha(n)$ to predefined constant

Step 4: For each input vector $\{X_i\}$ in X do

Step 5: Let Feature Interaction Matrix of the network is denoted by FIM.

Step 6: For i = 1 to n

Step 7: For j = 1 to n

Step 8: if ($i \neq j$) then

Step 9: Calculate $IE(F_i,F_j;TC)=H(F_i;TC)+H(F_j;TC)-H(F_i,F_j;TC)$

Step 10: Update w_{ij} with $IE(F_i,F_j;TC)$

Step 11: End if

Step 12: End For

Step 13: End For

Step 14: End For

Step 15: Let the Output Vector is denoted by Y

Step 16: Initialize m to 1

Step 17: For i = 1 to n

Step 18: MAX $IE = w_{i,1}$

Step 19: For j = 2 to n

Step 20: if ($i \neq j$) and ($w_{ij} > 0$) and ($w_{ij} >$ MAXIE) then

Step 21: MAX$IE = w_{ij}$

Step 22: End if

Step 23: End for

Step 24: Set redundant = FALSE

Step 25: For k = 1 to (i-1)

Step 26: if ($Y[k,2]$ = MAXIE) then

Step 27: if (Rel-Score($F_i,F_j;TC$) > Rel-Score($Y[k,1]; TC$) then

Step 28: Update $Y[k,1] = \langle F_i,F_j \rangle$; $Y[k,2]$ = MAXIE

Step 29: Set redundant = TRUE

Step 30: End if

Step 31: End if

Step 32: End For

Step 33: if (not redundant) then

Step 34: $Y[m,1] = \langle F_i,F_j \rangle$; $Y[m,2]$ = MAXIE

Step 35: $m = m + 1$

Step 36: End if

Step 37: End for

Step 38: At time $t > 0$, the weights of output neuron i and its adjacent neurons should be updated using

$$w_{ij}(new) = w_{ij}(old) + \alpha(t)[IE_{ij}(t) - w_{ij}(old)]$$

Step 39: Increase learning rate $\alpha(n)$

Step 40: Calculate $E = \sum_{i=1}^{m} Y_{i,2}$

Step 41: Calculate $\Delta E = E(Y_i(new)) - E(Y_i(old))$

Step 42: Repeat Step 4 to Step 41 until the network trains the given Input vector X and ΔE becomes minimized (i.e.,) the network reaches into stable state.

Step 43: Outputs the updated Vector $Y=\{Y_1,Y_2 \ldots Y_m\}$ which has positive Feature Interaction without any redundant feature pair.

The algorithm RFIM accepts the Input vector $X=\{X_1, X_2 \ldots X_n\}$ as an input and forms the Feature Interaction Matrix FIM based on the Interaction earned between every pair of features calculated using Eq.(2) from Step-4 to Step-14 for given feature vector. The highest values of the Interaction-Earned (IE) between each feature pair <F_i,F_j> in the i^{th} row of Feature Interaction Matrix FIM is identified by FII from Step-16 to Step-23 and assigned into MAXIE.

In order to remove the redundant feature pairs which have the same Interaction-Earned (IE) value, RFR first sets the 'redundant' flag value to 'FALSE'(Step-24) and compares the currently calculated Interaction Earned value with the IE values which are already stored in the output vector produced by the network through Step-25 to Step-32. The currently calculated Interaction Earned (IE) value matched with previously stored values in the output vector indicates that there exists a redundant feature pair and the corresponding 'redundant' flag value is set to 'TRUE' (Step-29). Based on the relevant score of these two redundant feature pair, output vector is updated with the feature pair which has the highest relevant score with respect to the target concept TC calculated using Eq.(5) in Step-27 and Step-28.

The feature pair < F_i,F_j> with MAXIE value is added into the output vector along with its IE value using Eq.(4) if MAXIE is not already present in the created output vector (Step-33 to Step 36). The updation of the neuron weights using Eq.(3) and learning rate is done by Step-38 and Step-39.

The energy function E is calculated using Eq.(7) and the difference between the consecutive energy function ΔE is calculated using Eq.(8) in Step-40 and Step-41. The network is trained until output vector $\{Y_1,Y_2...Y_m\}$ reaches into stable state (i.e.,) the value of ΔE becomes minimized. The value of updated output vector $\{Y_1,Y_2...Y_m\}$ is produced as output which contains positive interaction feature pair without redundant feature pair (Step 43). Thus, RFIM highly increases the classification accuracy by removing the redundant feature pair and identifying positive Feature Interactions.

4. EXPERIMENTS AND EVALUATION

The experiments are conducted on four datasets namely Heart Disease, Cardiac Arrhythmia, Dermatology and Hepatitis from AWS (Amazon Web Services) Cloud Database. These four datasets are collected from four different cloud locations. The detailed experimental results of Heart Disease dataset on RFIM are discussed. For the remaining three datasets namely Cardiac Arrhythmia, Dermatology and Hepatitis experimental results are shown in performance analysis.

4.1 HEART DISEASE DATASET AND RESULT

For experimental purpose, Heart Disease medical dataset from AWS (Amazon Web Services) Cloud Database have been taken. The four AWS cloud medical databases namely Hungarian, long-beach-VA, Cleveland and Switzerland have been tested for simulation. These four medical datasets are collected from four different cloud locations. The proposed model interacts with the features in Heart Disease medical dataset from AWS Cloud Database. Using the AppDesigner available in Matlab, the Recurrent Neural Network based App have been developed for simulation. The Hungarian database contains 800 instances, Long Beach VA contains 600 instances, Cleveland database contains 900 instances and Switzerland database contains 700 instances. There are 76 features in the medical Cloud Database. For experimental purpose, features like sex, age, cp, chol, fbs, thalach, trestbps, ca, rest_ecg, oldpeak, exang, slope,thal and one feature namely 'Num' is the predicted feature as well as Target Class which detect the presence or absence of heart disease has been used. These features are classified as data types like nominal, continuous and integer.

4.2 EVALUATION OF RNN BASED FEATURE INTERACTION MODEL

In order to evaluate the performance of RFIM, Heart Disease dataset is divided into two sets namely training set and testing set. The training set is used to train the RFIM and testing set is used to evaluate the classification accuracy of trained RFIM. The Input Feature Vector (X) from cloud database is taken as training set to the RFIM with the learning rate (α) chosen as 0.1.

According to algorithm for RFIM, Interaction-Earned Value for each attribute pair <F_i,F_j> is calculated for the given input feature vector in the training set using Eq.(2) and forms Feature Interaction Matrix (FIM). The values of Feature Interaction Matrix for each feature versus all other features are calculated. From the Feature Interaction Matrix, the maximum Interaction-Earned value for each feature is picked and forms the output vector Y in 2-tuple format like {<Feature pair>, IE of <Feature pair>} using FII. The feature pair which has the same Interaction-Earned values indicates the redundant feature pairs. In order to achieve better classification accuracy, these redundant feature pairs are identified and removed using Relevant score in RFR. The Energy function is calculated for the output vector values using Eq.(7) and value of Output Vector and Energy function obtained at the end of every 105th epochs are shown in Fig.3.

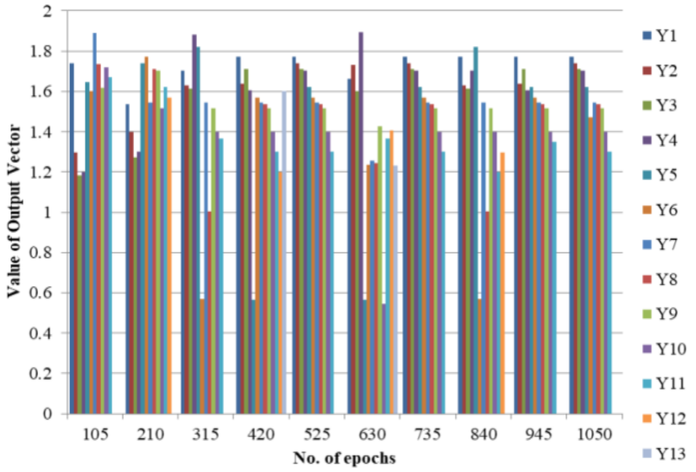

Fig.3. Values of Output Vector in RFIM

The Fig.3 shows the value of output vectors $\{Y_1,Y_2...Y_{13}\}$ which denotes the highest Interaction Earned for the feature pair at regular interval of epochs. The size of the output vectors varies at different interval of epochs depends on the generation and removal of redundant feature pair during Feature Interaction.

The trained network is applied on testing set and the corresponding interaction effect of feature pairs namely <Age, Trestbps> and <Thalach, Oldpeak> are depicted in Fig.4. The results obtained from the RFIM shows that when the feature namely "Thalach" in Heart disease dataset which denotes the maximum heart rate achieved yields the positive Feature

Interaction with the features like CP (Chest Pain Type), Fbs (Fast Blood Sugar), Exang, Slope and RestECG.

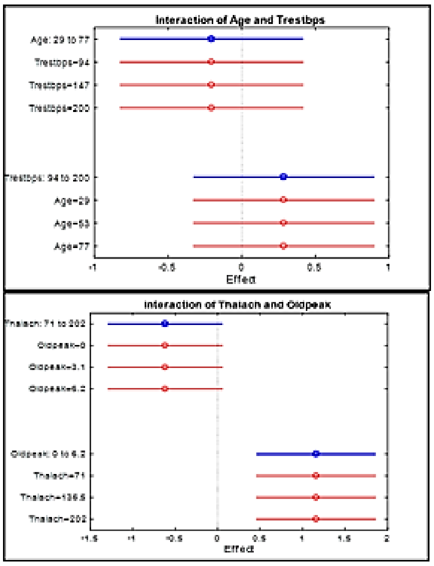

Fig.4. Interaction Effect of Two Feature Pairs

At the end of training, classification accuracy on the testing set at the regular interval of epochs is calculated. The Classification Accuracy (CA) on testing set is calculated using Eq.(9) and performance of RFIM is shown in Fig.5.

$$CA = \left(\frac{FV_{CC}}{FV_T} \right) \times 100 \tag{9}$$

where FV_{cc} - Number of Input Feature Vectors correctly classified and FVT - Total Number of Input Feature Vectors in the Testing set.

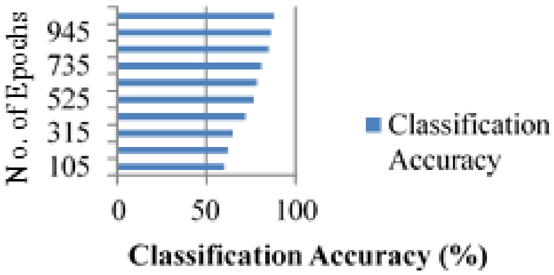

Fig.5. Performance of RFIM for Heart Disease Dataset

4.3 PERFORMANCE ANALYSIS FOR RFIM

The performance of RFIM with traditional Interaction based Feature Selection Methods are carried out on four datasets namely Heart disease, Cardiac Arrhythmia, Dermatology and Hepatitis. Using Neural Network toolbox in Matlab the comparative analysis is made on these four datasets. The conventional interaction-based Feature Selection Methods namely SAGA, INTERACT and FRFS are compared with the performance of RFIM on four datasets namely Heart Disease, Cardiac Arrhythmia, Dermatology, Hepatitis and percentage of accuracy is shown in Fig.6.

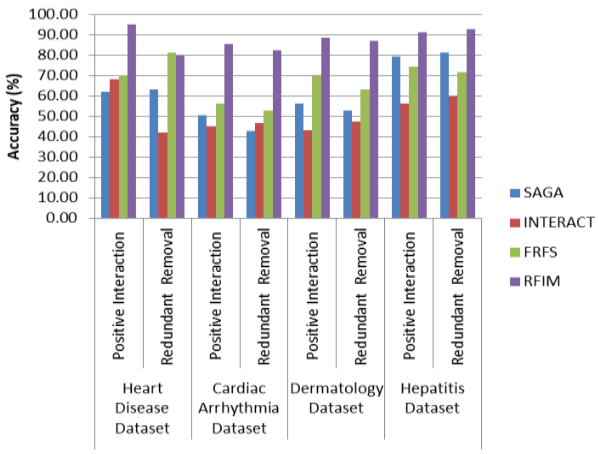

Fig.6. Performance of RFIM

The conventional methods namely INTERACT and FRFS uses Symmetrical Uncertainty and Cover ratio as evaluation measure respectively and comparatively produces less accuracy than the proposed RFIM which uses Interaction-Earned (IE) as evaluation measure.

The Fig.6 shows the performance comparison of RFIM with Interaction based Feature Selection Methods. From the performance comparison analysis, it confirms that existing conventional Feature Selection Methods namely SAGA, INTERACT and FRFS could not show its ability in detecting best positive Feature Interaction. The accuracy of RFIM in detecting positive Feature Interaction which is the Recurrent Neural Network based method is comparatively higher than other Feature Selection Methods.

5. CONCLUSION

The RFIM detects the positive feature interaction in the given dataset retrieved from the Cloud database. The Feature Interaction pairs from FII are sent to RFR for removal of redundant feature pairs. The Feature Interaction achieved by RFIM helps to detect positive and negative Feature Interaction efficiently. For experimental purpose, four data sets have been taken from cloud environment. The RFIM detects positive Feature Interaction as well as removes Redundant Feature pairs in all four datasets effectively when compared to the existing conventional methods. The efficiency of the proposed model is increased by removing the redundant feature pair exists in the output feature vectors which degrades the accuracy during classification. Thus, RFIM achieves Feature Interaction in the given dataset and gains additional information during classification. The information gained by positive Feature Interaction highly increases the accuracy of RFIM.

REFERENCES

[1] P.Usha and M.P. Anuradha, “Feature Selection Techniques in Learning Algorithms to Predict Truthful Data”, *Indian Journal of Science and Technology*, Vol. 16, No. 10, pp. 744-755, 2023.

[2] Z. Noroozi, A. Orooji and L. Erfannia, “Analyzing the Impact of Feature Selection Methods on Machine Learning

Algorithms for Heart Disease Prediction", *Scientific Report*, Vol. 13, pp. 1-15, 2023.
[3] A. Jakulin and I. Bratko, "Testing the Significance of Attribute Interactions", *Proceedings of International Conference on Machine Learning*, pp.409-416, 2004.
[4] Z. Zhao and H. Liu, "Searching for Interacting Features in Subset Selection", *Intelligent Data Analysis*, Vol. 13, No. 2, pp. 207-228, 2009.
[5] Z. Zeng, H. Zhang, R. Zhang and C. Yin, "A Novel Feature Selection Method Considering Feature Interaction", *Pattern Recognition*, Vol. 48, No. 1, pp. 2656-2666, 2015.
[6] I.A. Gheyas and L.S. Smith., "Feature Subset Selection in Large Dimensionality Domains", *Pattern Recognition*, Vol. 43, No. 1, pp. 5-13, 2010.
[7] G. Wang, Q. Song and Y. Zhou, "Selecting Feature Subset for High Dimensional Data via The Propositional FOIL Rules", *Pattern Recognition*, Vol. 46, pp. 199-214, 2013.
[8] K. Ding, Y. Li, J. Li, C. Liu and H. Liu, "Graph Neural Networks with High-Order Feature Interactions", *Proceedings of International Conference on Machine Learning*, pp. 45-56, 2019.
[9] G. Liu, H. Zeng and D.K. Gifford, "Visualizing Complex Feature Interactions and Feature Sharing in Genomic Deep Neural Networks", *BMC Bioinformatics*, Vol. 34, No. 3, pp. 1-15, 2019.
[10] S. Rendle, "Factorization Machines In Data Mining (ICDM)", *Proceedings of International Conference on Data Mining and Technologies*, pp. 995-1000, 2010.
[11] H. Peng, F. Long and C. Ding, "Feature Selection based on Mutual Information Criteria of Max-Dependency, Max-Relevance, and Min-Redundancy", *Proceedings of International Conference on Pattern Analysis and Machine Intelligence*, pp. 1226-1238, 2005.
[12] Arindam Sikdar, Yonghuai Liu, Siddhardha Kedarisetty and Yitian Zhao, "Interweaving Insights: High-Order Feature Interaction for Fine-Grained Visual Recognition", *International Journal of Computer Vision*, Vol. 133, pp. 1755-1779, 2025.
[13] W. Song, C. Shi, Z. Xiao, Z. Duan, Y. Xu, M. Zhang and J. Tang, "AutoInt: Automatic Feature Interaction Learning Via Self-Attentive Neural Networks", *Proceedings of International Conference on Pattern Analysis and Machine Learning*, pp. 1-6, 2019.
[14] A. Shrikumar, P. Greenside and A. Kundaje, "Learning Important Features through propagating Activation Differences", *Proceedings of International Conference on Machine Learning*, pp.40-46, 2019.
[15] A. Finnegan and J.S. Song, "Maximum Entropy Methods for Extracting the Learned Features of Deep Neural Networks", *PLOS Computational Biology*, Vol. 13, No. 10, pp. 1005836-1005845, 2017.
[16] M. Tsang, D. Cheng and Y. Liu, "Detecting Statistical Interactions from Neural Network Weights", *Proceedings of International Conference on Learning Representations*, pp. 1-12, 2018.
[17] J. Lian, X. Zhou, F. Zhang, Z. Chen, X. Xie and G. Sun, "DeepFM: Combining Explicit and Implicit Feature Interactions for Recommender Systems", *Proceedings of International Conference on Knowledge Discovery and Data Mining*, pp. 1-11, 2018.
[18] H.T. Cheng, L. Koc, J Harmsen, T. Shaked, T. Chandra and H. Aradhye, "Wide and Deep Learning for Recommender Systems", *Proceedings of International Conference on Learning Representations*, pp. 1-8, 2016.
[19] A. Garcia and J.A. Moreno, "*The Hopfield Associative Memory Network: Improving Performance with the Kernel "Trick""*, Springer, 2004.
[20] D.J. Hemanth, J. Anitha, L.H. Son and M. Mittal, "*Diabetic Retinopathy Diagnosis from Retinal Images using Modified Hopfield Neural Network*", Springer, 2018.
[21] A. Konar, S. Bhattacharyya, B.K. Panigrahi and K. Nakamatsu, "A Quantum Bi- Directional Self-Organizing Neural Network (QBDSONN) Architecture for Binary Object Extraction from a Noisy Perspective", *Applied Soft Computing*, Vol. 46, pp. 731-752, 2016.
[22] H. Yoon, C.S. Park, J.S. Kim and J.G. Baek, "Algorithm Learning based Neural Network Integrating Feature Selection and Classification", *Expert Systems with Applications*, Vol. 78, pp. 231-241, 2013.
[23] I. Inza, P. Larranaga, R. Etxeberria and B. Sierra, "Feature Subset Selection by Bayesian Network-Based Optimization", *Artificial Intelligence*, Vol. 123, pp.157-184, 2000.
[24] Z. Zeng, X. Jiang and R. Neapolitan, "Discovering Causal Interactions using Bayesian Network Scoring and Information Gain", *BMC Bioinformatics*, Vol. 17, pp. 221-236, 2016.