Automatic Detection of Biomarker Genes through Deep Learning Techniques: A Research Perspective

Rajangam ATHILAKSHMI[1]*, Shomona Gracia JACOB[2], Ramadoss RAJAVEL[3]

[1] Department of Computational Intelligence, SRM Institute of Science and Technology,
Faculty of Engineering and Technology, 603203 Kattankulathur, Chengalpattu District, Tamil Nadu, India
athilakr@srmist.edu.in (**Corresponding author*)

[2] Department of Computer Engineering, University of Technology and Applied Sciences,
Nizwa, Sultanate of Oman
graciarun@gmail.com

[3] Department of ECE, SSN College of Engineering, Chennai, India
RajavelR@ssn.edu.in

Abstract: The challenging problems associated with the analysis of microarray datasets are high dimensional feature, small sample size, class imbalance, noisy data, and high variance feature values. This has led to problems such as the curse of dimensionality, a decline in classification accuracy, and overfitting. Deep learning technology has gained massive popularity in biomedical research, and its algorithms are widely used to build models that solve complex classification problems. This study utilizes a deep neural network (DNN) for building classification models for microarray brain cancer data and ADPD (Alzheimer's disease Parkinson's disease) data from the KEGG (Kyoto Encyclopedia of Genes and Genomes) database. The small gene samples high dimensional feature in the given microarray data are addressed by considering a dimensionality reduction technique namely Correlated Feature Selection (CFS). The selected features of CFS were fed into the DNN for classification. For better training of the DNN model, the learning rates of various optimization algorithms were compared. The final optimal subset selected by the CFS-DNN model on brain cancer includes 112 features with an average classification accuracy of 95.83% and on ADPD data includes 40 features with an average classification accuracy of 87.5%. The performance of the proposed model is validated using 10-fold cross-validation. The proposed approach is also evaluated using precision, recall, F1-score, and Receiver Operating Characteristic curve. A comparative analysis of the proposed model with the state-of-the-art method in literature is carried out and the proposed method exhibits better performance than the one of the existing works and conventional machine learning models.

Keywords: Correlated feature selection, Deep Neural Network, Alzheimer's disease, Parkinson's disease.

1. Introduction

The hereditary information of all human beings is generally stored in genes. The change in the DNA sequence of a gene results in genetic disorders. Some genetic changes or mutations develop cancer or other genetic diseases in humans. The genetic changes can be identified by analyzing the data generated from microarray experiments. Microarray technology offered the researchers the possibility to study the expression of thousands of genes from a single sample. The considerable difference between the availability of genes (features) and the number of patients (samples) leads to the curse-of-dimensionality problem. The processing of all the gene features in the original data is not necessary, only some of them are relevant to the analysis. Thus, reducing the genes that show less interaction with class improves the accuracy and performance of the model (Alanni et al., 2019).

In recent years, the availability of a large volume of microarray gene expression data has led to detailed analytics in the field of computational biology. The data originating from microarray experiments has motivated researchers in the field of machine learning to develop and evaluate new algorithms. Qu et al. (2020) proposed a hybrid method called an ensemble multi-population adaptive genetic algorithm for selecting and classifying genes from cancer datasets. In the first phase, they used F-score methods for removing the noisy and redundant genes from the high-dimensional cancer datasets. Next, they applied an adaptive genetic algorithm with a support vector machine for classifying the reduced genes from the first phase. Alirezaei et al. (2019) introduced a bi-objective hybrid optimization algorithm to reduce noise and data dimensions in the PIMA Indian Type-2 diabetes dataset. First, the method identifies and removes the outliers. Following that they applied four bi-objective meta-heuristic algorithms on the reduced data from the first phase, for selecting significant genes with high classification accuracy.

Lu et al. (2017) introduced a hybrid feature selection algorithm called MIMAGA that combined Mutual Information Maximization (MIM) and the Adaptive Genetic Algorithm (AGA). The method removes the redundancies

https://doi.org/10.24846/v32i2y202305

and reduces the gene expression feature data for classification. Salem et al. (2017) proposed a method called IG/SGA (Information Gain/ Standard Genetic Algorithm) in which IG was applied for feature reduction, and SGA was employed to select the optimal number of genes in microarray cancer datasets. The research work of Ramani and Jacob (2013a) focused on differentiating several lung cancers such as Small Cell Lung Cancer (SCLC), and Non-Small Cell Lung Cancer (NSCLC), and on grouping them into common classes. Based on the structural and physicochemical properties of the protein, the lung cancer classification achieved an accuracy rate of 84%.

In this work, a deep learning approach was utilized for the classification of microarray brain cancer and ADPD neurodegenerative brain disorder data. This approach differs from the conventional models by the fact that it focuses on layer-wise feature learning and makes intelligent decisions on its own.

The objectives of this research work are listed as follows:

a. To select optimal features that are highly correlated with class using CFS.

b. To design and implement an efficient DNN model for classification by training the model with learning rates of different optimization algorithms.

c. To evaluate and compare the performance of the proposed CFS-DNN model with those of the existing conventional algorithms.

The rest of this paper is organized as follows: section 2 describes the proposed framework while section 3 elaborates on the experimental setup and discussion of obtained results. Section 4 concludes the paper with possible scope for further investigations.

2. The Proposed CFS-DNN Model

The proposed CFS-DNN model for the classification of brain cancer data and Common Gene Alzheimer-Parkinson (ADPD) neuro-degenerative brain disorder data is illustrated in Figure 1.

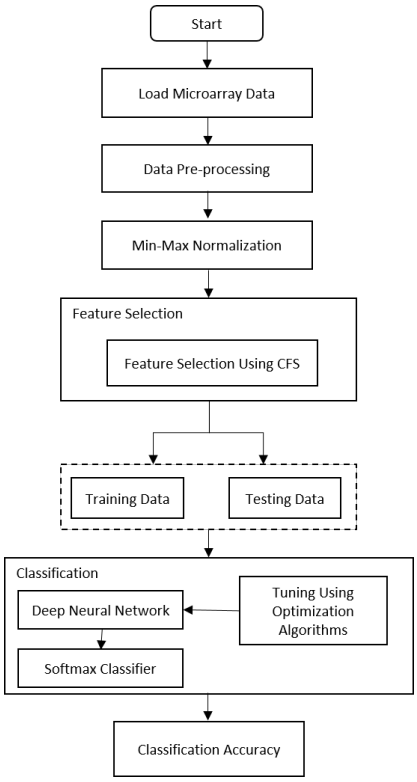


Figure 1. Proposed CFS-DNN Methodology

Firstly, the data is pre-processed using normalization techniques and is made ready for analysis. Secondly, the feature selection methods are applied to select important and optimal features needed for the modeling of an algorithm. In this work, Correlated Feature Selection was applied to select features with a high correlation to the class and a low correlation between each other. Next, the DNN classifier is constructed and trained using reduced data from CFS. Finally, the classification of brain cancer and ADPD data is performed using the Softmax layer in DNN. A detailed description of the proposed approach is explained in the upcoming section.

Algorithm 1: CFS–DNN model

Input: Brain cancer and ADPD data with 'm' samples and 'n' genes.

Output: Optimal and relevant feature set with high classification accuracy.

Step 1: Remove the duplicate, irrelevant and outliers from the input dataset.

Step 2: Apply the Min-Max scaling technique and normalize the input data.

Step 3: Apply the CFS technique for the selection

of optimal subsets based on high feature-class and low feature-feature correlations.

Step 4: Generate training and testing set for evaluation of data.

Step 5: Build the deep neural network model with the training feature set from step 4.

Step 6: Train and tune the DNN model using the hyperparameters given in Table 1.

Step 7: Improve the DNN model learning constructed in steps 5 and 6 using different optimizers.

Step 8: Measure the accuracy and loss of the proposed CFS-DNN classifier model using cross-validation on both training and testing data.

Step 9: Return optimal and relevant feature set with high classification accuracy.

Table 1. Hyperparameters used for the optimization of DNN

Hyperparameters	Value
Number of input nodes (brain cancer)	112
Number of input nodes (ADPD data)	40
Number of hidden layers	3
Number of nodes in Hidden layer 1	100
Number of nodes in Hidden layer 2	50
Number of nodes in Hidden layer 3	20
Number of output nodes (brain cancer)	5
Number of output nodes (ADPD data)	3
Activation Function of Hidden layers	ReLU
Activation Function of Output layers	Softmax
Dropout	0.2
Epochs	200
Batch Size	32

2.1 Normalization

Data normalization is a scaling technique applied to transform input feature values to fix on the common scale. The chosen dataset has features with different ranges of values and hence it is necessary to fit the present calculations in the same range of scale, which helps to train the model best. In this work, Min-Max scaling was applied to make the data fixed within the range of 0 to 1. Min-Max normalization preserves the linear relationship between the gene features and provides flexibility while training the model (Han et al., 2012). The Min-Max scaling of data is defined as follows:

$$X_{norm} = \frac{X - \min val_f}{\max val_f - \min val_f} \quad (1)$$

where $\min val_f$ and $\max val_f$ are the minimum and maximum values of a feature 'f'.

2.2 Feature Selection

Feature selection is a process of identifying the most contributing and relevant features needed for modeling the present data. The main reason for applying a feature selection method is to separate the relevant features from the irrelevant ones and to build an efficient model based only on those relevant features. Having irrelevant features in the dataset decreases the performance of the model, leads to overfitting, and increases training time (Isabelle & Andre, 2003). Thus, CFS technique was employed to overcome the above challenges and build the model with efficient, relevant, and contributing features. CFS is a filter model for finding correlated features with a high correlation to the class and a low correlation between each other. The CFS algorithm will be detailed as follows.

2.2.1 Correlated Feature Selection

The CFS algorithm (Hall,1999) evaluates the worth of a subset of features based on two measures, feature-class correlation and feature-feature correlation. The following equation gives the merit score of a feature subset S consisting of 'p' features:

$$Meritscore, S_p = \frac{p\overline{r_{xy}}}{p + p(p-1)\overline{r_{x_i x_j}}} \quad (2)$$

where r_{xy} is the mean value of all feature-class correlations and $r_{x_i x_j}$ is the mean value of all feature-feature correlations. The CFS criterion is defined as follows:

$$CFS = S_p \left[\frac{r_{x_1 y} + r_{x_2 y} + \ldots + r_{x_p y}}{P + 2(r_{x_1 x_2} + \ldots + r_{x_i x_j} + \ldots + r_{x_p x_{p-1}}} \right] \quad (3)$$

2.3 Classification

Deep Neural Network (DNN) is utilized for the classification of brain cancer and ADPD data features selected by the CFS technique. Deep Neural Network has recently become the standard tool for solving a variety of

bioinformatics problems. The ability to process high-dimensional features makes deep learning very powerful when dealing with complex machine learning problems.

2.3.1 DNN Architecture

The proposed Deep Neural Network consists of an input layer, a hidden layer, and an output layer. The input layer contains input neurons that feed the reduced data from the CFS model into the DNN. Thus, the input layer with 'n' inputs is expressed as:

$$x = \{x_1, x_{2,} x_{3,} ... x_n\} \tag{4}$$

The next is the hidden layer that maps the input layer neuron 'x' to hidden layer neurons and adds random weights 'w' and bias 'b'. Thus, it is expressed as:

$$h(x) = \sum_{i=1}^{n} w_1 x_1 + w_2 x_2 + ... + w_n x_n + b \tag{5}$$

Each hidden layer in the proposed DNN model is associated with the 'RELU' activation function to learn the nonlinearity present in the data (Glorot et al., 2011). It returns true for positive value and false for negative value and it can be written as:

$$f[h(x)] = f(\sum_{i=1}^{n} w_i x_i + b) \tag{6}$$

$$f(x) = \max(0, x) \tag{7}$$

The final output layer processes the hidden layer inputs using the Softmax activation function. The Softmax function converts the vector of 'k' real input values to normalized values within the range 0 and 1 (Goodfellow et al., 2016). The results of the output layer are expressed as:

$$\sigma(\overrightarrow{y_i}) = \frac{e^{y_i}}{\sum_{j=1}^{k} e^{y_j}} \tag{8}$$

where e^{y_i} value is the result of the standard exponential function applied to the input vector values, $\sum_{j=1}^{k} e^{y_j}$ is the result of normalization of all the output values within the range 0 to 1 and 'k' is the total number of classes that exist in the given data. The proposed DNN model is trained and tuned using the hyperparameters given in Table 1. The process of choosing suitable hyperparameters for training a DNN model is important because it directly controls the behavior of the training algorithm and thereby improves the performance of the model. The optimization hyperparameters such as learning rate, batch size, and the number of epochs help to train and learn the network faster and better. Thus, hyperparameter optimization results in an optimal model which shows an increase in model accuracy and decrease in loss function of the given data.

2.4 Evaluation Metrics

The following metrics are used to evaluate the classification performance of the proposed model.

Accuracy measures the ratio of correctly predicted feature samples to the total number of feature samples classified by the model.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{9}$$

Precision measures a ratio of correctly predicted positive feature samples to all positive feature samples that have been returned by the model.

$$\Pr ecision = \frac{TP}{TP + FP} \tag{10}$$

Recall is the proportion of relevant positive samples that are successfully retrieved by the model. It is also known as True Positive Rate (TPR).

$$\mathrm{Re}\, call = \frac{TP}{TP + FN} \tag{11}$$

F1-score measures the harmonic mean of precision and recall.

$$F1 - score = \frac{\Pr ecision.\mathrm{Re}\, call}{\Pr ecision + \mathrm{Re}\, call} \tag{12}$$

False Positive Rate (FPR) is the proportion of false positive feature samples to the total number of ground true negative feature samples predicted by the model.

$$FPR = \frac{FP}{FP + TN} \tag{13}$$

where TP = true positive, TN = true negative, FP = false positive, FN = false negative.

3. Experimental Results

3.1 Environmental Setup

This section discusses details of the hardware and software used for modeling the CFS-DNN algorithm. All experiments have been performed utilizing the KERAS library with Tensor flow as background in the Anaconda 3 environment using the Python 3.6 (Anaconda Software Distribution, 2020). The experiments were set up on Intel(R) Core (TM) AMD A10 CPU @ 3.5 GHz, 8.00 GB RAM, 64-bit Windows 10 OS.

3.2 CFS-DNN Results

This subsection illustrates the results of the proposed CFS-DNN model on two datasets (i) Microarray Brain Cancer (ii) ADPD. A brain cancer is a microarray cancer gene expression dataset downloaded from the Biolabs Data Set Repository which stores both experimental values and the gene names (Mramor et al., 2007). Next, the ADPD dataset contains the gene sets of Alzheimer's and Parkinson's disease based on structural and physicochemical properties of the protein, accessed from the Kyoto Encyclopedia for Genes and Genomes (KEGG) database (Kanehisa et al., 2000). The details of feature extraction and feature nomenclature of ADPD data are described by Jacob and Athilakshmi (2016) and Tejeswinee et. al (2017). Table 2 outlines the characteristics of the experimental datasets.

The proposed CFS–DNN classifier model is tested on the above two datasets. Firstly, the CFS method selects relevant features that strongly interact with the class. Next, the selected features are given to the DNN model for classification. The DNN model built in this work contains an input layer, hidden layers, and an output layer. In the input layer, the given selected features of the CFS algorithm are 112 in the case of brain cancer and 40 in the case of ADPD data. In the hidden layer, three DNN layers have been used where the first DNN layer contains 100 neurons while the second and third DNN layers contain 50 and 20 neurons, respectively. The output DNN layer contains 5 neurons for the brain cancer dataset and 3 neurons for the ADPD dataset. To learn the interactions between the features and nonlinearities present in the data, Rectified Linear Unit (ReLU) was used in the first three hidden layers. A Softmax activation function is implemented in the output layer for classification which converts a vector of numbers into a vector of probabilities. A dropout parameter is added for the first two hidden layers to prevent the model from overfitting. The proposed DNN model utilized 80% of the input data for training and 20% for testing the data. The proposed DNN model was trained several times for hyperparameter optimization.

One of the challenges lying when designing the DNN is choosing the right optimization algorithm with a good learning rate. During CFS-DNN training, various optimization algorithms were tried to improve the learning rate of the model and to generalize the performance. This work tries ADAM, RMSprop (RootMeanSquare propagation), Stochastic Gradient Descent Algorithm (SGD), and Adaptive Gradient (AdaGrad) optimization algorithms for tuning the learning rate and gradient parameters of the CFS-DNN model. The parameter learning rate plays an important role in achieving the model convergence and it must be discovered via trial-and-error method (Brownlee, 2019). For sparse datasets, the adaptive learning rate works better than other optimization techniques like SGD, RMSprop, and momentum methods. ADAM optimizer combines the best properties of the AdaGrad and RMSprop algorithms and can handle sparse gradients more effectively. It is well-suited for problems that are relatively large in terms of data and/or parameters (Walia, 2021). The classification accuracy at different epochs of CFS-DNN model training using different

Table 2. Dataset Description

Datasets	No. of samples	No. of genes	No. of classes
Brain Cancer	40	7130	5
ADPD	199	1437	3

optimizers on brain cancer and ADPD datasets are recorded in Table 3 and Table 4.

Table 3. Validation accuracy of CFS-DNN model on brain cancer

Iterations	Accuracy of optimization algorithms			
	Adam	SGD	RMSprop	AdaGrad
1-50	0.44	0.16	0.20	0.30
51-100	1.00	0.32	0.51	1.00
101-150	1.00	0.40	0.72	1.00
151-200	1.00	0.84	0.71	1.00
Validation accuracy	1.00	0.85	0.62	1.00

Table 4. Validation accuracy of CFS-DNN model on ADPD

Iterations	Accuracy of Optimization algorithms			
	Adam	SGD	RMSprop	AdaGrad
1-50	0.71	0.44	0.30	0.78
51-100	0.80	0.47	0.80	0.78
101-150	0.90	0.54	0.90	0.75
151-200	0.95	0.77	0.97	0.78
Validation accuracy	0.875	0.82	0.85	0.75

For the brain cancer dataset, all the optimization algorithms reached above 80% of validation accuracy whereas RMSprop showed 62% of accuracy. Both ADAM and AdaGrad optimizers showed good validation accuracy of 100% on the brain cancer dataset. For ADPD data, AdaGrad showed a validation accuracy of 75%. Next to AdaGrad, RMSprop and SGD optimizers reached a rate of accuracy above 80%. Finally, the Adam optimizer showed 87.5% of validation accuracy on the ADPD dataset which is comparatively higher than the ones of the other optimizers.

The next metric that is used for evaluating a candidate solution of the proposed model is the 'loss function' or 'cost function'. To calculate the error of the model during the optimization process, a loss function is used. The loss function used for training CFS-DNN models is 'sparse categorical cross-entropy'. The loss values at different epochs of CFS-DNN model training using different optimizers on brain cancer and ADPD datasets are recorded in Table 5 and Table 6.

Table 5. Validation loss of CFS-DNN model on brain cancer dataset

Iterations	Loss value of different algorithms			
	ADAM	SGD	RMSprop	AdaGrad
1-50	1.61	1.61	1.58	1.44
51-100	0.008	1.53	2.10	0.02
101-150	0.008	1.40	2.57	0.01
151-200	0.006	0.97	3.02	0.008
Validation loss	0.005	0.83	2.85	0.007

Table 6. Validation loss of CFS-DNN model on ADPD dataset

Iterations	Loss value of different algorithms			
	ADAM	SGD	RMSprop	AdaGrad
1-50	0.85	0.97	0.96	0.87
51-100	0.08	0.54	0.20	1.43
101-150	0.07	0.44	0.10	1.63
151-200	0.04	0.10	0.08	2.06
Validation loss	0.08	0.20	0.12	2.01

The general form of the sparse categorical cross-entropy loss for a Softmax classifier with 'n' classes is:

$$L = \frac{1}{|F|} \sum_{f \in F} \log(\sigma_{y_j}) \tag{14}$$

where $y_j = 1$ for class 'j' and $y_{i \neq j} = 0$ for all other classes, 'n' is the total number of classes, and 'f' is the number of features selected from the feature set 'F'.

For the brain cancer dataset, the loss value predicted by the ADAM and AdaGrad optimization algorithms is decreasing at iterations. At the end of the 200th epoch, both ADAM and AdaGrad optimizers help the model to reach the validation loss of 0.005-0.007 on the brain cancer dataset. The loss value showed by SGD and RMSprop on the brain cancer is 0.83 and 2.85, respectively, which is comparatively higher than the ones of the other two optimizers ADAM and AdaGrad. For ADPD data, AdaGrad showed a validation loss of 2.01. Next to AdaGrad, SGD and RMSprop showed validation loss of 0.20 and 0.12, respectively. ADAM optimizer showed a good decrease in the loss at iterations and reached a validation loss of 0.08 which is comparatively lower than the one of other optimization algorithms on the ADPD dataset.

While compared to other optimization algorithms, ADAM optimizer converges very fast and effectively improves the learning rate of the model for both datasets. The combination of the proposed CFS-DNN model with ADAM optimizer shows good validation accuracy with low loss for the brain cancer dataset. For the better training of the proposed DNN model, the ADAM optimizer is tuned with various learning rates such as 1.0, 0.1, 0.01, 0.001. Figure 2, 3, 4 and 5 show the comparison of accuracy and loss of different optimizers on brain cancer data.

Figure 2 shows that the CFS-DNN model achieved maximum accuracy of 95.83% on the brain cancer dataset when it was tuned to the learning rate of 0.1 in ADAM optimizer.

Also, in Figure 3, the model shows a stable decrease in loss value when it was trained for the learning rate 0.1 and the trend for both training data and testing data showed a joint decrease in loss value for brain cancer data.

From Figure 4, it can be observed that the CFS-DNN model reached the maximum accuracy of 87.5% on the ADPD dataset when it was tuned to the learning rate of 0.01 in ADAM optimizer, with the trending curve moving in an upward direction for both training and testing data. For other optimizers the model did not show a comparable increase in training accuracy and testing accuracy.

Figure 5 illustrates the stable decrease in loss value when it was tuned to the learning rate of 0.01 in ADAM optimizer. There is no stable decrease in training and testing loss of ADPD data when it was tuned to the learning rates of the other optimizers. Hence it can be observed that the proposed CFS-DNN with ADAM optimizer with a learning rate 0.1 for the brain cancer and of 0.01 for ADPD dataset results in a good rise in classification accuracy with a low loss on the training and testing data.

3.3 Performance Evaluation

The proposed model is validated using 10-fold cross-validation for the classification of brain cancer and ADPD neurodegenerative brain disorder data. The performance of the proposed CFS-DNN classifier model is further compared with the existing five conventional learning algorithms: logistic regression (LR), support

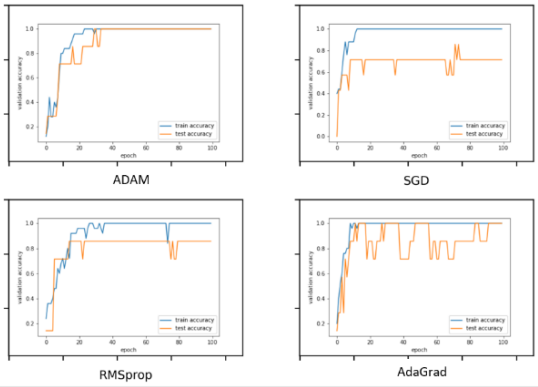


Figure 2. Brain cancer dataset - Comparison of accuracy

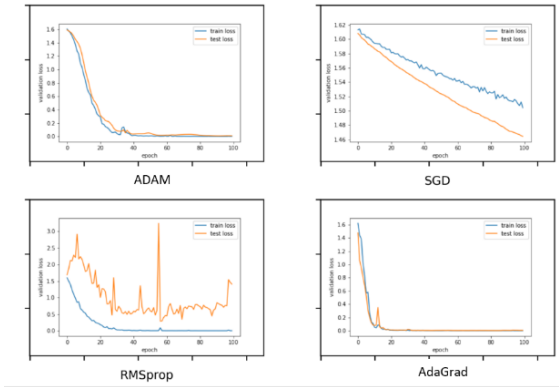


Figure 3. Brain Cancer dataset - Comparison of loss

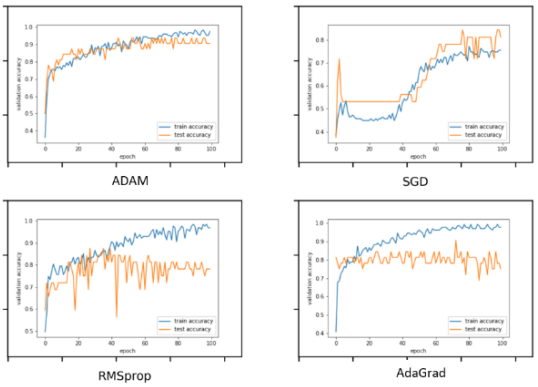


Figure 4. ADPD dataset - Comparison of accuracy

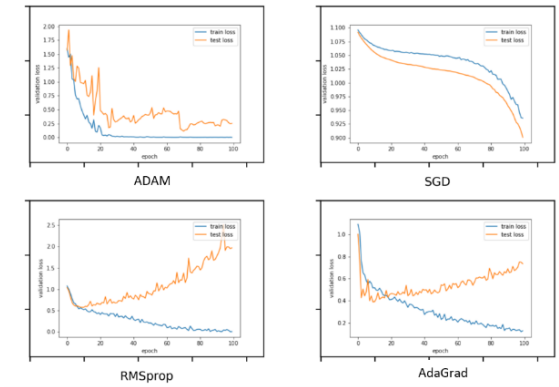


Figure 5. ADPD dataset - Comparison of loss

vector machine (SVM), Naïve Bayes (NB), Random Forest (RF), and Decision Tree (DT) applied on the two datasets brain cancer and ADPD brain disorder data.

The comparative analysis between the performance of the proposed model and those of the existing classifiers on brain cancer data is shown in Table 7. Out of the six models, the proposed CFS-DNN model achieved the average classification accuracy of 95.83%, precision of 0.80, recall of 1.00, and F1-score of 0.98 for the classification of selected features on brain cancer data. Next to the proposed model, Random Forest and Support Vector Machine provided better performance on brain cancer data. The classification accuracy rate and other performance parameters of the three classifiers, namely Logistic Regression, Decision Tree and Naïve Bayes, had a value below 80.

The comparative analysis between the proposed model and the other existing models based on different performance parameters regarding the ADPD dataset is shown in Table 8. For ADPD data, the proposed model reached the average classification accuracy of 87.5%, precision of 0.86, recall of 0.88, and F1-score of 0,88. Next to it, LR and SVM classifiers showed better performance in classifying the ADPD neurodegenerative brain disorder data. Both LR and SVM classifiers provided better accuracy of nearly 83% and 0.8 for precision, recall, and F1-score of nearly 0.8 for the ADPD data. The classification accuracy rate and other performance parameters of the other three classifiers, namely, RF, DT and NB, were lower when compared to those of the other classifiers. The above results demonstrate that the CFS-DNN model performed better on both datasets, when compared to the existing machine learning classifiers.

3.4 Comparative Analysis

This subsection discusses the comparative analysis between the proposed model and different classifiers based on the receiver operating characteristic (ROC) curve. For a better evaluation of the proposed work, results were plotted based on the ROC curve. The class-wise results of the proposed model and ROC analysis of other existing classifiers on two datasets are shown in Figures 6 and 7. The experiments demonstrated that the proposed CFS-DNN significantly outperformed the existing models in the classification of selected features of brain cancer and ADPD data.

In Figure 6(a), class 0 represents the medulloblastoma type, class 1 represents the glioma type, class 2 represents the rhabdoid type, class 3 represents the normal type and class 4 represents PNET (primitive neuroectodermal) type, respectively. The proposed model illustrated a ROC of 0.96 for medulloblastoma class, of 0.91 for glioma class, of 1.0 for rhabdoid class, of 1.0 for normal class, and 0.95 for (PNET) class.

Figure 6(b) shows the ROC curve for the existing machine learning models such as Logistic Regression, Support Vector Machine, Random Forest, Decision Tree, and Naïve Bayes on brain

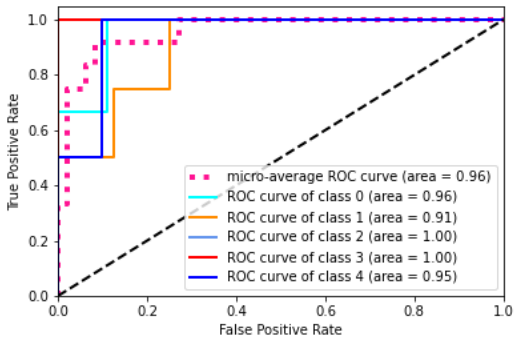
Table 7. Comparative analysis between the proposed model and the existing models on the brain cancer data

Classifier	Precision	Recall	F1-Score	Accuracy (%)
Logistic Regression	0.78	0.75	0.75	75
Support Vector Machine	0.79	0.88	0.82	89
Random Forest	1.00	0.88	0.93	87.5
Decision Tree	0.75	0.50	0.56	60
Naïve Bayes	0.67	0.62	0.62	62.5
CFS-DNN	**0.80**	**1.00**	**0.98**	**95.83**

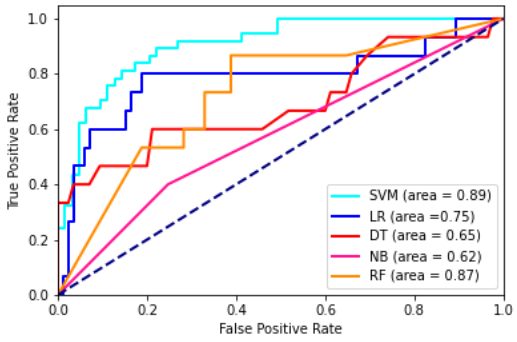
Table 8. Comparative analysis between the proposed model and the existing models on the ADPD data

Classifier	Precision	Recall	F1-Score	Accuracy (%)
Logistic Regression	0.83	0.79	0.80	83
Support Vector Machine	0.82	0.81	0.81	82
Random Forest	0.76	0.78	0.75	77.5
Decision Tree	0.67	0.68	0.67	67
Naïve Bayes	0.68	0.69	0.68	68
CFS-DNN	**0.86**	**0.88**	**0.88**	**87.5**

cancer data. The ROC results of the proposed model improved from ~ 7% to ~ 20% for brain cancer data when compared with the ROC results of the existing machine learning models. Thus, the proposed CFS-DNN model shows better ROC results of 0.96 for brain cancer data when compared to those of existing machine learning models.



(a)

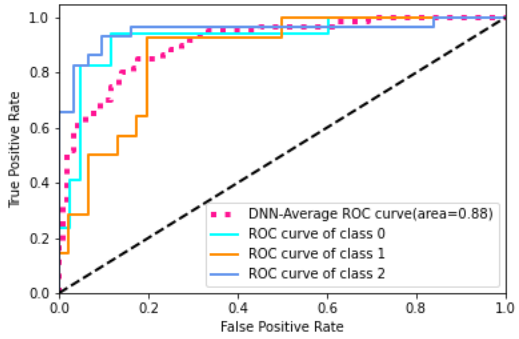


(b)

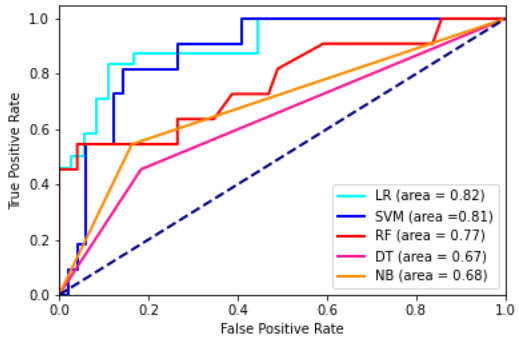
Figure 6. ROC curve for Brain cancer dataset
(a) Class-wise results of the proposed model
(b) Results of other existing classifiers

In Figure 7(a), class 0 represents the Alzheimer's disease, class 1 represents the Parkinson's disease, and class 2 represents the common class, respectively. Figure 7(a) shows the ROC curve values for the proposed CFS–DNN model on ADPD data with an average ROC value of 0.88. Also, the CFS-DNN model reached a ROC value of 0.91 for Alzheimer's class, of 0.89 for Parkinson's class and of 0.92 for common classes, on ADPD neurodegenerative brain disorder data.

Figure 7(b) shows the ROC curve values for the existing machine learning models. Next to the CFS-DNN model, LR and SVM models showed better ROC values of 0.8 on ADPD data, whereas the ROC results of the other three classifiers, namely RF, NB, and DT, were lower when compared to the values of the above-mentioned models. The ROC results of the proposed model improved from~ 6% to ~ 15% for ADPD data when compared with existing machine learning models. Thus, the proposed CFS-DNN model shows better ROC results of 0.88 when compared to existing machine learning models on ADPD data.



(a)



(b)

Figure 7. ROC curve of ADPD dataset
(a) Classwise results of the proposed model
(b) Results of other existing classifiers

Table 9. Comparison between the proposed approach with the existing work on brain cancer and ADPD dataset

Authors	Approach	No. of optimal features	Brain cancer	ADPD
Almars et al. (2021)	HFS	200	95	---
Ramani & Jacob (2013a)	RWFS	3	77.5	---
Patel et al. (2020)	ANN	100	94	---
Venkataramana et al. (2020)	CDSS-DFS	55	---	79.7
Proposed	CFS-DNN	112	**95.83**	---
Proposed	CFS-DNN	40	---	**87.5**

The findings of the research proposed in this paper are in line with the results obtained in similar with similar works from past research on the above two datasets. A comparison between the result of the proposed model and those of earlier works implemented on the above two datasets is shown in Table 9. Most of the earlier works on the brain cancer dataset were tried for '2'classes, and only a few works were based on '5' classes. The Hybrid feature selection (HFS) approach proposed by Almars et al. (2021) selected a number of 200 features and showed an accuracy rate of 95% on the brain cancer dataset. On the same dataset, the least number of features was selected by Rank Weight Feature Selection (RWFS) but it reached an accuracy rate of 77.5% (Ramani & Jacob, 2013b).

The new method based on Artificial Neural Network and used by Patel et al. (2020) reached a good accuracy rate of 94% on the brain cancer dataset. In line with this study, the proposed CFS-DNN model selected 112 features and achieved an average classification accuracy of 95.83% on the brain cancer dataset with five classes, as shown in Table 9, which represents a result better than the existing state-of-the-art results. The first result on the ADPD dataset with three classes was reported by Venkataramana et al. (2020) with a classification accuracy of 79.7%, precision of 0.78, recall of 0.76, and an F1-score of 0.76 by using a Clinical Decision Support System (CDSS) based on Decremental Feature Selection (DFS) approach. Compared with similar previous research works on the above two datasets, the proposed CFS-DNN model achieved an average accuracy of 87.5%, as shown in Table 9, which represents a result better than the existing state-of-the-art results.

4. Conclusion

In recent years there has been a rising interest in the application of machine learning techniques in the medical research field. Medical Research based on computational techniques not only evaluates conceptual models and guides experimental approaches but also acts as a tool to lower the cost and time required for wet-lab experiments. The optimal subset selected by the CFS-DNN model includes 112 features for brain cancer and 40 features for ADPD dataset, with an average classification accuracy of 95.83% and of 87.5%, respectively. The results of the proposed model were compared with the existing state-of-the-art results from the literature and the comparative analysis showed that the model proposed in this paper builds an advanced computational method, by providing a feature selection and classification approach for categorizing brain cancer data and ADPD brain disorders, with good generalization performance. In the future, the proposed work can be extended to detect and categorize the features of other brain disorders that could be targeted for brain disorder therapy.

Acknowledgments

This research reported in this paper is part of the work project titled "Investigation on the effect of Gene and Protein Mutants in the onset of Neuro-Degenerative Brain Disorders (Alzheimer's and Parkinson's disease): A Computational Study", with reference no. SERB – YSS/2015/000737/ ES, funded by the Science and Engineering Research Board (SERB), Department of Science and Technology (DST), under Young Scientist Scheme – Early Start-up Research Grant.

REFERENCES

Alanni, R., Hou, J., Azzawi, H. & Xiang, Y. (2019) Deep gene selection method to select genes from microarray datasets for cancer classification. *BMC Bioinformatics*. 20, 608. doi: 10.1186/s12859-019-3161-2.

Alirezaei, M., Taghi, S., Niaki, A., Armin, S. & Niaki, A. (2019) A bi-objective hybrid optimization algorithm to reduce noise and data dimension in diabetes diagnosis using support vector machines. *Expert Systems and Applications*. 127, 47–57. doi: 10.1016/j.eswa.2019.02.037.

Almars, A. M, Alwateer, M., Qaraad, M., Amjad, S., Fathi, H., Kelany, A. K., Hussein, N. K. & Elhosseini, M. (2021) Brain Cancer Prediction Based on Novel Interpretable Ensemble Gene Selection Algorithm and Classifier. *Diagnostics (Basel)*. 11(10), 1936. doi: 10.3390/diagnostics11101936.

Anaconda Software Distribution. (2020) *Computer software version 3.8*. https://www.anaconda.com [Accessed 2nd February 2023].

Brownlee, J. (23 January 2019) How to Configure the Learning Rate When Training Deep Learning Neural Networks. *Machine Learning Mastery*. https://machinelearningmastery.com/learning-rate-for-deep-learning-neural-networks/ [Accessed 1st February 2023].

Glorot, X., Bordes, A. & Bengio, Y. (2011) Deep sparse rectifier neural networks. In: *Proceedings of International Conference Artificial Intelligence and Statistics AISTATS 2011, 11-13 April 2011, Fort Lauderdale, Florida, USA*. USA, JMLR. pp 315-323.

Goodfellow, I, Bengio, Y. & Courville, A. (2016) Optimization for training deep models. In: Goodfellow, I, Bengio, Y. and Courville, A. (eds.) *Deep Learning*. The MIT Press, pp. 271-325.

Hall, M. (1999) *Correlation-based Feature Selection for Machine Learning*, Ph.D. thesis. University of Waikato.

Han, J., Kamber, M. & Pei, J. (2012) *Data mining: Concepts and techniques*. 3rd ed. USA, Morgan Kaufmann Publishers.

Isabelle, G. & André, E. (2003) An introduction to variable and feature selection. *Journal of Machine Learning Research*. 3(7-8), 1157-1182. doi: 10.1162/153244303322753616.

Jacob, S. G. & Athilakshmi, R. (2016) Extraction of Protein Sequence features for Prediction of Neuro-degenerative Brain Disorders: Pioneering the CGAP database. In: *Proceedings of International Conference Informatics and Analytics ICIA-16*, *25-26 August 2016*, *Pondicherry, India*. New York, NY, United States, Association for Computing Machinery. pp. 1-4.

Kanehisa, M. & Goto, S. (2000) KEGG: Kyoto Encyclopedia of Genes and Genomes. *Nucleic Acids Research*. 28(1), 27-30. doi: 10.1093/nar/28.1.27.

Lu, H., Chen, J., Yan, K., Jin, Q., Xue, Y. & Gao, Z. (2017) A hybrid feature selection algorithm for gene expression data classification. *Neurocomputing*. 256, 56-62. doi: 10.1016/j.neucom.2016.07.080.

Mramor, M., Leban, G., Demsar, J. & Zupan, B. (2007) Visualization-based cancer microarray data classification analysis. *Bioinformatics*. 23(16), 2147-2154. doi: 10.1093/bioinformatics/btm312.

Patel, P., Pasi, K. & Jain, C. K. (2020) Prediction of cancer microarray and DNA methylation data using non-negative matrix factorization. *Computer Science and Information Technology*. 2020, 65-81. doi: 10.5121/csit.2020.100906.

Qu, Y., Li, R., Deng, A., Shang C. & Shen, Q. (2020) Non-unique decision differential entropy-based feature selection. *Neurocomputing*. 393, 187-193. doi: 10.1016/j.neucom.2018.10.112.

Ramani, R. G. & Jacob, S. G. (2013a) Benchmarking Classification Models for Cancer Prediction from Gene Expression Data: A Novel Approach and New Findings. *Studies in Informatics and Control*. 22(2), 133-142. doi: 10.24846/v22i2y201303.

Ramani, R. G., Jacob, S. G. (2013b) Improved classification of lung cancer tumors based on structural and physicochemical properties of proteins using data mining models. *PLoS One*. 8(3), e58772. doi: 10.1371/journal.pone.0058772.

Salem, H., Attiya, G. & Fishawy, N. E. (2017) Classification of human cancer diseases by gene expression profiles. *Applied Soft Comput*ing. 50 (C), 124-134. doi: 10.1016/j.asoc.2016.11.026.

Tejeswinee, K., Jacob, S. G. & Athilakshmi, R. (2017) Feature Selection Techniques for Prediction of Neuro-Degenerative Disorders: A Case-Study with Alzheimer's and Parkinson's Disease. *Procedia Computer Science*. 115, 188-194. doi: 10.1016/j.procs.2017.09.125.

Venkataramana, L., Jacob, S. G., Saraswathi, S. & Athilakshmi, R. (2020) Clinical decision support system for neuro-degenerative disorders: an optimal feature selective classifier and identification of predictor markers. In: Abraham, A., Cherukuri, A., Melin, P. & Gandhi, N. (eds.) *Advances in Intelligent Systems and Computing*. Springer, Cham, pp. 10-20.

Walia, A. S. (10 March 2021) Types of optimization algorithms used in neural networks and ways to optimize gradient descent. *Nerd For Tech*. https://medium.com/nerd-for-tech/types-of-optimization-algorithms-used-in-neural-networks-and-ways-to-optimize-gradient-descent-1e32cdcbcf6c [Accessed 1st February 2023].