RNN-CNN Based Cancer Prediction Model for Gene Expression

Mrs. Uma H R Asst. Professor, Dept of computer science and engineering, BGS Institute of Technology, Adhichunchanagiri university BG Nagara, Karnataka

Sonashree N,4th year ,8th semester Computer Science and Engineering, BGS Institute of Technology, Adhichunchanagiri university BG Nagara, Karnataka

ABSTRACT

One of those sicknesses that is generally destructive to individuals is disease. The best way to forestall any damage to mankind is by its initial disclosure and treatment. Different kinds of tests are directed in the clinical labs for the identification of malignant growth. Disease can likewise be distinguished at the hereditary level. For this unmistakable AI and profound learning techniques as of now exist. This paper proposes a crossover strategy in view of Repetitive Brain Organization (RNN) and Convolution Brain Organization (CNN) to foresee various kinds of disease like Bosom, Lung, Uterine, Kidney, Prostate and colon malignant growth from quality articulation information. The bottleneck highlights are removed utilizing the sandwich stacked strategy in light of VGG16 and VGG19 pre-prepared models. A while later, the proposed half breed classifier in light of RNN-CNN has been utilized to order the information into different classes. The proposed model performs better compared to the next existing strategies, for example, VGG16, VGG19, ResNet50, Commencement V3 and MobileNet classifier as far as different execution measurements like exactness, Mean Square Mistake (MSE), accuracy, review, and F1 score. RNN-CNN classifier gives the most elevated precision of 0.978 among the wide range of various existing techniques for Dataset 1 and the most noteworthy exactness of 0.994 for Dataset 2 at 80% preparation information. Then again, RNN-CNN classifier gives the least MSE of 0.101 among the wide range of various existing strategies for Dataset 1 and the most reduced MSE of 0.006 for Dataset 2 at 80% preparation information.

INDEX TERMS Cancer, CNN, deep learning, gene expression, machine learning, RNN.

I. INTRODUCTION

One of the biggest risks to human health has been well established to be cancer [1], [2]. Cancer has been identified as one of the top causes of death in humans, second only to infectious, cardiovascular, and cerebrovascular disorders, according to epidemiologic data, posing a serious threat to human health [3]. One of the most prevalent aspects in the classification of cancer is thought to be gene expression. The physiological state and gene activity of biological systems can be determined at the transcriptome level through gene expression i.e., the transcriptome reflects the active expressed genes at any given moment [4]. Any Ribonucleic Acid (RNA), including messenger RNA (mRNA), is referred to as the transcriptome. These molecules transmit genetic data to the ribosome from Deoxyribonucleic Acid (DNA) [5]. RNA-Seq analyses the transcription of a particular gene. The gene expression profiles from various tissues are compared making it possible to pinpoint the genes that are crucial for the specification of the phenotype. Comparing healthy and diseased tissues, for example, can shed new light on the genetic factors implicated in pathology. To identify gene regularity targets, diagnose diseases, and create new drugs, researchers can use aspects from gene expression data that can be examined using computational approaches. According to studies, these data can reveal very crucial details on the tumor's features, opening choices for the patient's care, management, and treatment [6], [7], [8], [9].

Gene expression data (GED) provides some additional insights that helps to enhance cancer classification [49]. Using the data, it is thought that a challenge that requires the use of computational approaches is the identification of genes that are abundantly expressed in tumor cells rather than normal ones. Due to the high dimensionality and relatively low sample size of the data, these data presented additional difficulties for the application of computational approaches. For the categorization of cancer using GED, numerous supervised and unsupervised learning techniques have been created [9], [10]. When examining the data for cancer, Deep Learning (DL) technologies are addressing the shortcomings of conventional machine learning (ML) techniques [19].

ML methods have not achieved a good success in terms of dimensionality reduction and cancer classification for GED due to lack of identification of most significant features. But on the other side, DL methods have achieved better performance in these two fields [9]. GED has less samples and more features. Due to this ML models create several redundant learners which always give the same low-level performance in spite of any input given to the machine [63].

A. MACHINE LEARNING

The belief in science fiction stories can cause a war between machine and their developers due to the development in the field of Artificial Intelligence (AI). The machines have come so far from playing games to controlling traffic lights and military drones. We are surrounded by data that makes this era an era of big data. Every organization, either govt or private and even a single person is generating data every moment. This data is easily accessible, and it can be processed further to get meaningful insights. ML is a field that provides various algorithms with the help of which we can extract meaningful information from the given data. ML is a process to teach our machine how to use the data to perform different tasks such as classification, regression, clustering, and so on. Machines are performing extraordinarily in different fields but still these are dependent on humans. As an example, machine can find important features from huge data but for the analysis of these features, humans are still required [22].

Due to the availability and accessibility of the immense data from the internet or through other sources, computer applications have switched from simple data processing to the field of ML. ML is training machines using different algorithms on data which helps in automating human assistance [23]. ML is a sub-field of AI in which machines are trained from past experiences to think like humans. It does not require that much human interference to analyze the derived conclusions, but still human dependency is there. ML has numerous impacts on various fields such as production lines, healthcare, education, transportation and food [24]. There are some other applications of ML such as finding a mail is span or harm, identify genes responsible for a disease, customer segregation for advertisement, weather forecasting, financial losses estimation, suspicious transactions, and prediction of election result, and many more [22].

B. DEEP LEARNING

DL is considered as the subset of ML. DL is an emerging field of the era as it is having many advantages such as DL algorithms provide outstanding results as compared to ML models, required data and resources are approachable and new algorithms are coming into picture [25]. DL models are based on stochastic gradient descent (SGD). In spite of very good results with the ML models, these models are behind in some areas such as speech recognition and object recognition. This act as one of the reasons behind DL field [26]. DL models emerged from artificial neural network (ANN) which is influenced by the biological neurons of the brain. ANN consists of three layers: input, hidden and output layer with multiple nodes. These models learn through an iterative training process known as backpropagation, but generalization is missing in supervised tasks. ANN with more hidden layers becomes a Deep Neural Network (DNN) for providing better generalization. It also helps in better feature extraction and better learning with huge number of parameters. There are various platforms for implementing DL models such as TensorFlow, PyTorch, Theano and Caffe [27].

Deep multilayer perceptron (DMLPs), CNNs and RNNs are three types of supervised learning models in DL [28]. CNN works good for imaging data. It consists of convolution, activation and pooling [29]. RNN works on the information received from the previous layer that makes this model most suitable for sequence modelling [30]. The autoencoder (AE) and restricted Boltzmann machine (RBM) are two types of unsupervised learning models in DL. Unsupervised learning does not require labels of the data for training model. In AE, encoder transforms the latent representations of the input before feeding them into decoder for the reconstruction at the output. RBM is also known as generative model which maps the representations on the input data based on estimations of probability distribution [31].

C. MOTIVATION AND CONTRIBUTION

High throughput sequencing technologies are fast developing which helps in better collection of genomic data e.g., GED to solve various clinical and biological problems. But GED suffers from few problems such as high dimensionality and low features, noisy data and, complexity and variability. It is also used in disease classification [17]. In our work, we are targeting two issues that are related to GED. One is cancer classification and the other is dimensionality reduction. There are few questions that motivate us for this research such as what are the various methods available in the literature which can be used for cancer classification, what are the different ways of reducing the dimensions of the GED, which method can be the most suitable method for feature extraction based on which new method can be proposed and which method can be the most suitable for cancer classification based on which new classifier can be proposed. To support our questions, we proposed two methods: One for feature extraction which is based on pre-trained models, VGG16 and VGG19. And other

for cancer classification which is based on RNN and CNN. The contribution of our work is summarized as follows:

- Summary of various DL methods for cancer classification
- Feature extraction method based on VGG16 and VGG19 pre-trained models.
- RNN-CNN based classifier.
- Evaluating the proposed methods using various performance measures with existing methods.

The remaining part of the paper is partitioned into multiple segments. Section II presents the existing methods in the field of DL for classification. The details of the dataset and methods used for our work are given in Section III. The outcomes of the research work are given in Section IV. Section V gives the concluding remarks of this work.

II. RELATED WORK

To categorize widely expressed genes and rarely expressed genes and to build an ideal RNN classifier, the incremental feature selection approach with a supervised classifier RNN has been used [11]. The lightweight CNN having two layers of convolution is used to classify breast cancer from GED with an accuracy of 98.76% to address the issues with the GED's small size and high dimensionality [12]. Long short-term memory (LSTM) neural network-based L-GEPM, which captures the non-linear characteristics affecting gene expression and leverages learnt features to forecast the target genes, has been employed. It can produce minimal error and excellent fitting results [13].

The classification of various cancer types using tumor RNA sequence (RNA-Seq) GED has been accomplished using a DL strategy grounded on binary particle swarm optimization (PSO) with a decision tree (DT) (BPSO-DT) and CNN with an accuracy of 96.90% [14]. GeneXNet, a CNN architecture has been developed to address the complicated nature of gene expressions [15]. The short length of gene expression time series affects the majority of clustering approaches, or they neglect the temporal structure of the data. These problems are addressed by DeepTrust, a brand-new DL-based approach for grouping gene expression time series. To provide richer data representations, DeepTrust first converts time series data into images. The created images are then subjected to a deep convolutional clustering method [16].

To combine data from the feature and sample spaces, a hierarchical graph convolution network (HiGCN) has been used. HiGCN increased the performance of the classification and survival analysis even when there were few labeled data points, according to experiments on both simulated and actual data. This method also learned more potent representations of the GED [17]. The development of DL techniques and multidimensional data presents chances for a more thorough examination of the molecular features of breast cancer, which can enhance detection, treatment, and prevention. For the prognostic prediction of breast cancer, a multimodal deep neural network by incorporating multi-dimensional data (MDNNMD) has been deployed [18].

Although the magnitude of the microarray data has increased over time, its difficult qualities, such as its limited sample size and high complexity, have not changed. To effectively handle these difficulties, more intelligent algorithms in both gene selection and classification need to be created. To preprocess data and choose informative genes for precise cancer detection, a hybrid algorithm using deep fuzzy neural network has been developed that provides better performance than other conventional classification methods [20]. Combining the LSTM and CNN, a two-level predictor known as DeepDRBP-2L was proposed. It is the first computational technique capable of distinguishing between DBPs (DNA-Binding Proteins), RBPs (RNA-Binding Proteins), and DRBPs (both DNA RNA Binding Proteins). Thorough cross-validations and independent tests demonstrated that DeepDRBP-2L can go beyond the limitations of the current approaches to find DRBPs [21].

AFExNet is used to extract the features from the high dimensional GED used for breast cancer classification. This method consists of two stages: pre-training using unsupervised learning and fine-tuning using supervised learning. AFExNet is a NN that is based on adversarial auto AE (AAE) [32]. Features are also extracted using deep convolutional neural networks from in situ hybridization (ISH) images for GED. Deep CNN-based methods outperformed existing methods in the field of feature extraction [33]. Not only deep CNN, ANNs are also used for cancer classification from GED and help to identify the most relevant genes for classification [34]. Principal component analysis-fisher analysis (PCA-FA) is used to analyze GED using supervised learning. PCA is used to reduce the correlation of gene expression levels, and the most relevant features are extracted using FA [35].

The tumor tissues are categorized into more than two classes by analyzing gene expression profiles using semi-supervised Ellipsoid ARTMAP and PSO. This technique gives very good results on three publicly available datasets, but the method still suffers from a problem of noise and dimensionality [36]. Classes from the GED are also found using a method based on perturbation technique, cluster ensemble approach, and cluster validity index [37]. One of the most prominent cancers is the breast cancer in women. Therefore, it is segmented using multi-channel Markov random fields (MRF). It gives better results for segmentation than single-channel MRF [38]. Locality-sensitive Laplacian score (LSLS) and wrapper method as a combined approach have been used for supervised feature selection. This method is tested on six datasets and is found to be effective [39]. Tumor tissues are also classified using a support vector machine (SVM). But before classification, features are selected using robust PCA (RPCA) and RPCA plus linear discriminant analysis (LDA) [40].

Linear regression (LR) has not worked well in the field of gene expression as compared to DL-based methods namely D-GEX, a multilayer feed-forward NN. DL-based method outperformed LR with an improvement of 15.33% [41].

Supervised learning-based RNN classifiers and incremental feature selection have been used to differentiate between widely expressed and rarely expressed genes [42]. As GED suffers from a problem of small samples and large features, CNN consisting of two convolutional layers making it lightweight CNN, is used to classify breast cancer. This method attains good accuracy of 98.76% as compared to existing models [43]. Features are also selected BPSO-DT from GED, which are then classified using CNN to find the different types of cancer with an accuracy of 96.90 [44]. Genes are selected using nonnegative matrix factorization (NMF) or sparse NMF (SNMF) for the classification of tumors from GED. This method comes out to be efficient and effective for the classification of tumors and normal tissues [45].

To extract the non-linear features from the GED, LSTM based method known as LGEPM is used. These features are then used for the target gene prediction. This method performs better than logistic regression, DT, and K-nearest neighbor (K-NN) [46]. In spite of cancer classification from single tissue type, cancer can also be classified from multi-tissues using multilayer CNN, known as GeneXNet. This method provides an accuracy of 98.9% [47]. Cancer subtype classification is also of utmost importance for the diagnosis and treatment of cancer. To work with DNN is not so easy because of its structure. On the other hand, a Flexible neural tree (FNT) can be used in such cases, but these cannot be used for multi-class classification. So, FNT combined with DNN known as DFNForest has been used to convert multi-class problems into binary classes for each forest. This forest consists of multiple FNTs [48]. Different ML and DL models are compared which shows that DL models provide good results on GED rather than ML models [50], [51].

Oral cancer has been detected using a novel method based on CNN and Deep Belief Network (DBN). The proposed method based on CNN and DBN is efficient and provides an accuracy of 97.35%. Features are extracted using optimized CNN and classification of oral cancer has been done using optimized DBN [69]. CNNs are providing significant improvements in the segmentation of biomedical images for brain tumor segmentation. Unwanted material from the images is extracted using a gray-level co-occurrence matrix (GLCM) and then images are segmented using deep CNN (DCNN) [71].

The brain tumor has been classified using Caps-VGGNet, a model based on CapsNet and VGGNet. This method has the advantage of extracting the features automatically along with no requirement of large datasets as required in most of the DL models [72]. Group CNN (G-CNN) along with special Euclidean (SE2) motion group and Discrete Cosine Transform (DCT) has been used for better classification of brain cancer providing an accuracy of 94.84% [73]. Different types of cancer cells for various cancer such as lung, brain, breast, and cervical are detected using pre-trained models of CNN namely MobileNet, VGGNet and DenseNet.

These models have worked on Learning without Forgetting (LwF) [74].

Deep learning can also be used to determine skin cancer over time. It can help to improve the performance of and accuracy of the cancer prediction. Seven different types of skin cancer are detected by optimized CNN. The proposed method helps to deal with the imbalance class problem [83]. Features are extracted for the successful classification of breast cancer images from various participating environments. Recall has been analyzed with proper care more than accuracy because false negatives are of utmost importance because of the medical data. Features are extracted using a federated learning facility which is based on transfer learning [84]. The multi-omics data integration for gene expression has been done using the method based on Gene Similarity Network (GSN) and CNN. CNN has been used to predict the stage of breast cancer and the levels of Gleason score of prostate cancer patients [85]. The features related to Nottingham Prognosis Index (NPI) class are chosen and are represented using GSN maps and breast cancer is then classified using deep residual neural network [86]. Table 1 shows the comparison of the various related work including dataset used and findings that has been discussed in the given section.

Various methods based on ML and DL are available in the literature for the classification of cancer and a few of these methods applicable for the GED are discussed above. Deep neural networks are providing better results than other methods and it has been found that performance can be improved further with some advancements in the field of NNs. To support this, we proposed two methods which will be explained in further sections.

III. PRPOSED MATERIALS AND METHODS

A. DATASET

Two datasets are used in this work. Both the datasets are publicly available. Dataset-1 used in this work is available at [64]. This dataset consists of 5 different types of cancer namely ''breast invasive carcinoma (BRCA), kidney renal cell carcinoma (KIRC), lung adenocarcinoma (LUAD), lung squamous cell carcinoma (LUCS), and uterine corpus endometrial carcinoma (UCEC)''. There are 2086 samples and 971 genes. Each row consists of a specific record for each patient. Every column comprises of RNA-Sequence values of a gene. The last column of the dataset consists of classes of cancer coded as 1,2,3,4 and 5 for BRCA, KIRC, LUAD, LUSC and UCEC, respectively. There are 878 samples of BRCA, 537 samples of KIRC, 162 samples of LUAD, 240 samples of LUSC and 269 samples of UCEC.

Dataset-2 used for our work is available at [65]. It contains 5 types of cancer namely BRCA, KIRC, LUAD, colon adenocarcinoma (COAD) and prostate adenocarcinoma (PRAD). There are total of 801 samples available in the given dataset. Each sample has 20532 gene sequences. The record of each patient is present row wise, and RNA-Sequence of each gene is present column wise. The classes of cancer as per the dataset are coded as 0,1,2,3 and 4 for PRAD, LUAD, BRCA,

TABLE 1. Comparisons of the related works.

Ref. No.	Objective	Technique/Tool	Dataset	Findings
[11]	Classification of widely and rarely expressed genes with RNN.	Incremental Feature Selection, RNN.	S2 Table in [66].	The difference between rarely and widely expressed genes specifies the gene expression pattern and distribution for cancer and normal tissues. It also specifies distribution of gene expression for tumor tissues.
[12]	Classifies the breast cancer from gene expression data using CNN.	Normalization, PCA, t-Distributed Stochastic Neighbor Entities, R Studio.	Breast Cancer Dataset from Pan-Cancer Atlas.	Used only for single class cancer classification.
[13]	Finds the non-linear features of the data for predicting targeting genes.	K Means Clustering, Normalization, L-GEPM.	GEO Data from LINCS cloud, GTEx Samples and 1000 Genomes Expression data from Illumina RNA Sequence Platform.	L-GEPM act as a solution to the problem of LINCS program by finding the non-linear features of the gene expression data.
[14]	Performs multi-class classification for cancer prediction.	BPSO-DT, CNN, MATLAB.	Cancer Types Dataset available at Mendeley [64].	Less complex method with less training time.
[15]	Performs cancer classification from multi-tissues without feature selection.	Normalization, Deep CNN, Consensus Hierarchical Clustering.	Different datasets from The Cancer Genome Atlas (TCGA).	Deep Learning provides promising results in the field of medical diagnosis.
[16]	Resolves the problem of short length time series data using convolutional deep clustering method.	Convolutional Autoencoder (DeepTrust).	Simulated and Biological dataset.	Convolutional networks along with autoencoder provides good results for the time series data.
[17]	Learns the information of feature and sample space of gene expression data using HiGCN.	Graph Convolution Network (GCN), HiGCN, Sparse GCN, Feature-Weighted GCN.	2 simulated and 2 real datasets.	Works well with few labelled data and performs features selection from the noisy data.
[18]	Integrates multi-dimensional data for breast cancer classification.	mRMR feature selection, Multimodal DNN.	METABRIC dataset.	Gives better results than the methods with single dimensional data.
[20]	Develops a deep fuzzy neural network for handling uncertainties in gene expression data.	Index Based Filtering, Clustering, Graph Ranking, Gene Selection.	Colon, Leukemia, Prostate, Central Nervous System, Ovarian and Breast Cancer Datasets.	Provides an accuracy of 90% with less than 20 genes.
[21]	Identifies DBPs, RBPs, and DRBPs.	CNN, Bidirectional LSTM.	Benchmark Dataset and Independent Dataset [67]	Feasible and effective method than existing methods.
[32]	Extracts features from high dimensional breast cancer data.	Sampling, Normalization, AFExNet (Based on AAE).	BRCA dataset from cBioportal, UCEC dataset from TCGA.	Provides good performance on multiclass classification. Better results than existing feature extractors.
[33]	Extracts the features from ISH images for generic representation.	Deep Convolutional Neural Network, VGG16.	Drosophila ISH images from FlyExpress Database.	Best representation of gene expression has been made by intermediate layers of deep models.
[34]	Classifies cancer from GED based on identification of the most significant genes.	ANN, PCA.	cDNA microarray data.	This serves as the first application of ANN for classification of cancer from cDNA microarray GED.
[35]	Performs feature selection and multi-class classification of cancer.	Semi-supervised Ellipsoid ARTMAP, PSO.	NC160 Data, Acute Leukemia Data, ALL Data.	Provides good results on all the three datasets specifically for NC160 on which existing methods have not given promising results
[36]	Finds the linear relationship between cancer patient and his survival.	PCA, FA, Fuzzy FA.	Samples from the dataset available at [68].	Genes with higher correlation are to be used in local attributes network.
[37]	Finds the number of classes in GED.	Neural Gas Clustering Algorithm, Disagreement Agreement Index (DAI).	3 synthetic and 4 cancer datasets.	The hidden structures from the datasets are found using DAI and it outperforms the existing cluster validity indexes.
[40]	Performs cancer classification based on feature selection.	RPCA, LDA, SVM.	Acute Leukemia, Colon Cancer, Gliomas, Medulloblastoma, Prostate cancer, 11_Tumors data, Brain Tumor data.	Feasible and Effective method
[41]	Interprets the expression of targeted genes from GED.	Multilayer feed forward neural network (D-GEX), Python.	GEO Dataset, GTEx Expression Data, 1000 Genomes Expression Data.	DL provides better results than linear regression on gene expression. Due to hardware restrictions, targeted genes are randomly grouped into different sets.

TABLE 1. *(Continued.)* Comparisons of the related works.

[41]	Interprets the expression of targeted genes from GED.	Multilayer feed forward neural network (D-GEX), Python.	GEO Dataset, GTEx Expression Data, 1000 Genomes Expression Data.	DL provides better results than linear regression on gene expression. Due to hardware restrictions, targeted genes are randomly grouped into different sets.
[42]	Widely expressed and rarely expressed genes classification from normal and cancer tissues.	mRMR, Incremental Feature Selection (IFS), RNN.	GED available at [66].	Normal tissues and tumor tissues are distinguished in a better way. It specifies the distribution of gene expression for tumor tissues.
[45]	Performs the dimensionality reduction and classification of cancer from GED.	NMF, SNMF, SVM.	Colon cancer, Acute Leukemia, Medulloblastoma Dataset.	Efficient and effective method. Analysis of the biological meaning of the genes after gene selection.
[48]	Classifies the sub-types of cancer using DL.	Ensemble FNT (DFNForest), Fisher Ratio, Neighborhood Rough Set.	Breast Invasive Carcinoma, Glioblastoma Multiforme and Lung Cancer from TCGA.	Better performance and higher accuracy than existing methods
[69]	Classifies oral cancer from medical images	CNN, DBN, PSOBER	Oral cancer Images available at [70]	Provides good results as compared to other existing methods. Efficient method
[71]	Segmentation of brain tumor images using D-CNN	GLCM, Vantage Point Tree (VPT), DCNN, U-NET	Brain MRI Dataset	Method achieves accuracy of 99.40%
[72]	Feature extraction and classification of brain tumor using Caps-VGGNet	CapsNet, VGGNet, Leaky ReLU activation function.	Brats 2019, Brats 2020 dataset	0.98 accuracy on Brat 2019 and 0.99 on Brat 2020. High effectiveness and high efficacy.
[73]	Breast cancer classification using G-CNN	GCNN, DCT,	MNIST Dataset, Curated Breast Image Subset of the Digital Database	Achieved an accuracy of 94.84%. High computational performance
[74]	Cancer classification using LwF	CNN, MobileNet, VGGNet, DenseNet, Bayesian Optimization	Different cancer datasets in [75-82]	Pre-trained models with LwF perform better.
[83]	Skin cancer classification using optimized CNN	CNN, Activation function such as ReLU, Tanh and Swish, Grad-CAM, Grad-CAM++	HAM10000 Dataset	Helps in the early detection of cancer. Provides a classification accuracy of 82%
[84]	Breast cancer classification using transfer learning	Synthetic minority oversampling technique (SMOTE), FeAVG+CNN, MobileNet	Digital Database For Screening Mammography (DDSM) Datasets, Curated Breast Imaging Subset of DDSM (CBIS-DDSM)	Good classification performance
[85]	Multi-omics data integration	GSN, CNN	PRCA, BRCA	Provides 99% accuracy and performs better than iSOM-GSN
[86]	Classification of Breast Cancer NPI	ResNet, SOM, GSN, Chi-Square, mRMR, t-SNE	METABRIC Dataset	Provides accuracy of 98.48% and the AUC 0.9999

KIRC and COAD. Out of total 801 samples, 300 samples are of BRCA class has 300 samples, KIRC has 146, LUAD has 141, COAD has 78 and PRAD class has 136 samples.

Data Pre-Processing: Datasets used in this research work are available at [64] and [65]. Dataset 1 is a MAT file which has been stored in dataframe after loading into Jupyter note-book. It contains 2086 rows and 962 columns. The dataset's last column is the class column which has been renamed into Target column for better understanding. Dataset has been split into 2 parts: training and testing where 80% is the training data and 20% is the testing data. The training and testing dataset is further arranged into (1668,961) and (418,961) respectively in order to make 2D array and which are then reshaped into (1668,31,31,1) and (418,31,31,1) for further implementation. Similarly for the dataset 2 some pre-processing has been done. Dataset 2 contains 2 csv files which are later merged into one and stored in a data frame. It consists of 801 rows and 20533 columns. Column number 1 has been removed from the data frame as it was just containing the sample numbers which are not going to play any role in the classification of the data. The last column name is

a class column that contains the names of different cancers. These class names are replaced by numbers from 0-4. The size of the dataset then becomes (801,20531) after storing the class column in a separate variable. Later before further processing, the dataset was divided into training and testing data and 20531 was reshaped into (143,143) and then again into (32,32,1).

B. METHODS

Two hybrid methods are proposed in this work. The first method is sandwich stacked bottleneck (SSB) feature extraction based on VGG-16 and VGG-19. Another is a hybrid classifier based on RNN and CNN. The details of the methods are explained in this section.

CNN: It is a DNN that is used for image and video recognition. However, CNN has its applications in the fields of natural language processing (NLP) and discovery of various drugs [58]. This model comprises of various layers namely convolution layers, sub-sampling layers, fully connected layers, and an activation function [59]. A fully connected layer performs the actual classification. CNN works on 2D images.

In each layer of the model, not every neuron is connected to every other neuron of the next layer. Unlike other neural networks, CNN has a lesser number of connections that help to reduce the training time as well as the problem of overfitting [60].

Convolution layer: There is an image i with filter or kernel K on which convolution operation C has been performed as given in equation (1).

$$C(x, y) = (I * K)(x * y)$$
$$= \sum_i \sum_j I(x - i, y - j) . K(i, j) \quad (1)$$

where,

$C(x, y)$ is the convolution operation which is performed at (x, y) position in the feature map.

$I(x - i, y - j)$ is pixel's input value at $(x - i, y - j)$ position.

$K(i, j)$ is the convolutional kernel's value at (i, j) position.

Activation Function: The activation functions are applied after convolution to add non-linearity. The most used function is ReLU.

$$A(x, y) = \max(0, C(x, y))$$

Pooling: Pooling is used to decrease the size of the sample or for down sampling the feature map. Max pooling is generally used and for which equation (2) is applicable.

$$P(x, y) = \max_{i,j} A(s.x + i, s.y + j) \quad (2)$$

where

$P(x, y)$ is the pooling operation's output.

s is the stride.

Fully Connected Layer: The output of the previous layer is flattened into a vector and is multiplied by the W weights and B bias is added to it as per equation (3):

$$Z = W.Flatten(P) + b \quad (3)$$

Loss Function: The cross-entropy loss function with softmax activation can be defined as per equation (4):

$$Loss = -\sum_i y_{true,i} . \log(y_{pred,i}) \quad (4)$$

where,

$y_{true,i}$ is the true label for class i.

$y_{pred,i}$ is the predicted value for class i

· *VGG-16:* VGG-16, being one of the popular CNN models, is used for classification of images [52]. It can be used for feature extraction as well, as it provides promising results [53]. This comes out to be an expensive model in terms of evaluation, memory, and parameters. There are 138 million parameters available in the model [54]. This model is also known as a pre-trained model as it is already trained on huge dataset, ImageNet. This makes this model good for even small datasets as well. By default, input size for an image is 224, 224, 3 for VGG16 [55]. But for our research work, the input size is 32,32,1.

· *VGG-19:* This is among the popular DL models, developed by Simonyan and Zisserman. It keeps the number of parameters low even while managing the depth of relevant layers. It comprises of 16 convolution layers along with 3 fully connected layers [56]. Like VGG16, it is also the pre-trained model which is trained on ImageNet dataset. 16 convolution layers are serving the purpose of feature extraction, and 3 fully connected layers are used for classification task [57].

RNN: It is also a type of NNs but with a closed loop feedback. RNN learns sequential and time varying patterns [61]. It has computing power of the brain that helps to work extraordinarily. RNN provides the same accuracy as feed forward NN but with a smaller size. There are multiple categories of RNN that are globally RNN, locally RNN, time delayed RNN and simultaneous RNN [62]. The architecture of RNN is of two different types: fully connected and partially connected [61].

Hidden State: The hidden state h_t of RNN is updated which is based on input x_t at time t and the previous hidden state h_{t-1}. It is represented using equation (5):

$$h_t = \sigma(W_{hx}.x_t + W_{hh}.h_{t-1} + b_h) \quad (5)$$

where,

x_t is the input vector at time t.

h_t is the vector of hidden state.

W_{hx} is the matrix of weight for x_t.

W_{hh} is the matrix of weight for h_{t-1}.

b_h is the bias.

σ is the activation function.

Output: The output of RNN is calculated based on hidden state h_t and weight matrix W_{hy} as per equation (6):

$$y_t = W_{hy}.h_t + b_y \quad (6)$$

where,

y_t is the output vector at time t.

W_{hy} is the weight matrix for h_t to y_t.

b_y is the bias.

C. PROPOSED METHODOLOGY

Different datasets have been taken from multiple sources.

As a first step, we imported the desired libraries and then pre-processing was performed. The data processing includes removing NULL values if any, dropping the unwanted columns if any, and labelling of the classes in the dataset. After pre-processing, the next step is to split the data into various training and testing ratios such as 80:20, 70:30, 60:40, etc. Then the proposed feature extractor has been applied to the data. The proposed feature extractor has been compared with existing models such as ''VGG16, VGG19, ResNet50, and Inception V3'' and it comes out to be the best among all. After relevant feature extraction, proposed classifier has been applied on the extracted data to classify the data into given classes. The proposed classifier has been compared with existing classifiers such as ''VGG16, VGG19, ResNet50, Inception V3 and MobileNet''. The proposed classifier comes out to be the best among all.
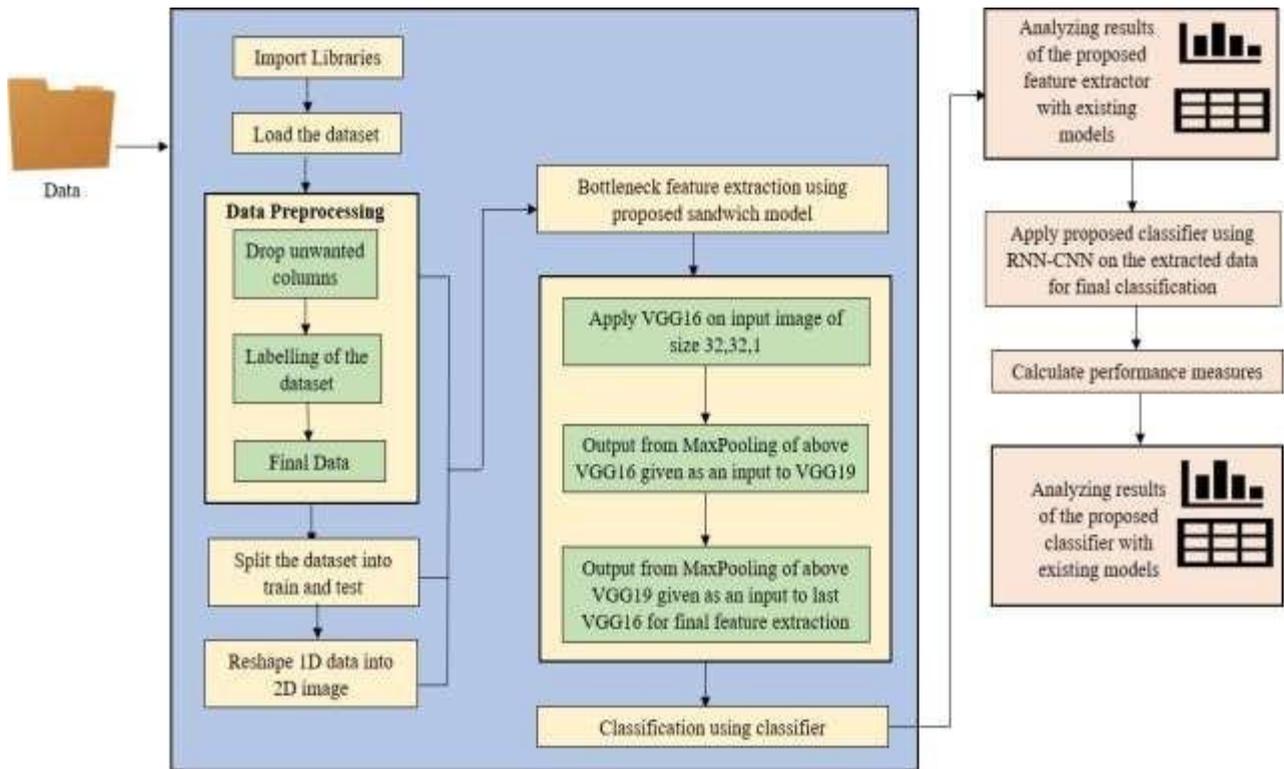
FIGURE 1. An overview architecture of the proposed method.

1) SSB FEATURE EXTRACTOR

This method has been proposed for extracting the bottleneck features from the dataset as shown in Figure 2. The method is known as sandwich stacked because VGG19 is stacked between two VGG16 models, which is like a structure of a sandwich. First VGG16 takes an input of size 32, 32, 1. After processing, the output of the VGG16 has been given to the VGG19 as an input and then further the output of VGG19 has been given to VGG16 as an input. The output of the final VGG16 is known as the bottleneck features as fully connected layers are removed from these pre-trained models VGG16 and VGG19. Figure 1 shows the flowchart of the proposed methodology.

2) RNN-CNN CLASSIFIER

This proposed method has been used to classify the data into multiple classes as shown in Figure 3. The extracted features are given to the CNN model. The presented CNN model contains three 2D convolution layers and each convolution layer relates to activation layer of relu, batch normalization and the dropout layer with 0.25 value. The first convolution layer takes an input of size 32,2,2. After all the convolution blocks, the output has been given to the RNN model. It consists of layers of RNN, LSTM and Gated Recurrent Unit (GRU). After taking output from these layers, it has been given to the flatten layer for converting the output received from the above layers into 1 D and then it has been passed to the dense layer with SoftMax activation for final classification.

The implementation is conducted using various ML and DL packages in python. Python's packages such as sklearn.model selection, numpy, keras, tensorflow, keras.layers and many more have been used. The dataset-1 is a mat file that has been converted into a dataframe consisting of 2086 rows and 962 columns. The training and testing data have been created from the dataset. Features are stored in one variable and labels are stored in another variable. Train and test data with a ratio of 80:20 is further arranged in the form of 2D array with a shape of (1668,961) and (418,961) respectively that has been again reshaped into (1668, 31, 31,1) and (418, 31, 31,1). Then to implement the feature extractor based on VGG16 and VGG19, train and test images are then reshaped into (1668,32,32,1) and (418,32,32,1). The VGG16 consists of convolution, flattened and dense layer. The input shape of (32,32,1) has been given to the convolution layer of the VGG16 without including the top layers of VGG16. The dense layer consists of 1024 neurons. Then features are extracted at the end of this dense layer which are then reshaped into (32,32,1) and then given to VGG19 which again consists of convolution layer with input size of (32,32,1) and without including top layers, and then to flatten layer and dense layer.

with 1024 neurons. At this step, again features are extracted and then again reshaped into (32, 32, 1) and given to the second VGG16. It consists of convolution with input size of 32,32,1 and then flatten and dense layer with 1024 neurons which extracted the features. At this step the, the features
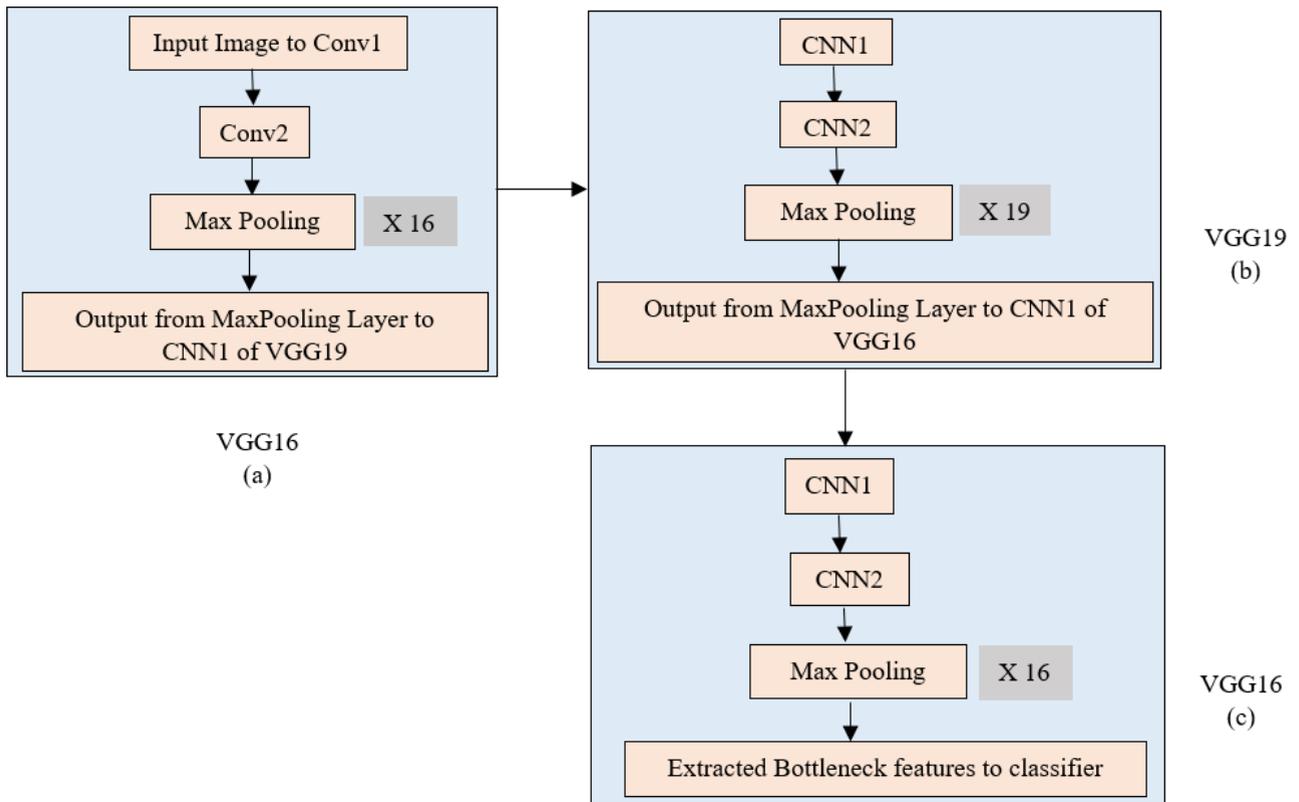
FIGURE 2. An overview architecture of the sandwich stacked bottleneck feature extractor.

received are extracted features from the sandwich stacked model. The extracted features are then reshaped into (32, 32, 1) before applying the final classifier. The final proposed classifier consists of RNN and CNN. CNN model consists of 3 blocks. Each block consists of 2Dconvolution, activation and batch normalization layers. The convolution layer in the first block applies 32 filters of size 2×2 on the given input and then relu activation and batch normalization. The output received from the batch normalization layer of the first block has been given as an input to the convolution 2D layer of the second block which applies 64 filters of size 2×2, the output of which are again passed to the relu activation and batch normalization layer of the second block. The output received from the batch normalization has been given to the convolution 2D layer of the third block which applies 128 filters of size 2×2, which are then passed to the relu activation and batch normalization.

The third block also consists of a dropout layer with a value of 0.25 which helps to prevent overfitting. The output received from the CNN model having a shape of (4,4,128) is again reshaped into (4,512) to apply a layer of simple RNN. This layer consists of 128 units as a dimension of the output provided by it and it returns that output to the next layer of LSTM with same output dimension. The sequences returned from the LSTM layer are passed to the dropout layer with a value of 0.25 and then the output has been passed to the flatten

layer and then again to the dropout layer with a value of 0.25. The output received from the last dropout layer is having a shape of (None,512) which is then passed to a dense layer with 5 neurons as the number of classes as per the dataset 1 and softmax activation. This way the final model has been developed, which has then been fitted on the testing data for the final validation.

The dataset-2 initially consists of two csv files. One consists of features and the other consists of labels. These two files are then combined to form one file and stored in the form of dataframe with 801 rows and 20533 columns. The first column has been dropped from the dataframe which stores the sample number only, which made the number of columns as 20532. Last column in the dataframe stored the labels name, which are then replaced with the values from 0-4. The class column has been dropped from the actual dataframe and has been stored separately. After this the size of the dataframe becomes (801, 20531). After this pre-processing step, the data has been split into various training and testing ratios and 20531 has been reshaped into (143,143), which has been reshaped again into (32,32,1) for applying proposed feature extractor. And then the final classifier has been applied to create a model, which then be fitted on the testing data for the final validation. The activation function, optimizer, and loss function used for proposed as well as for the existing models is softmax, adam and poisson respectively.
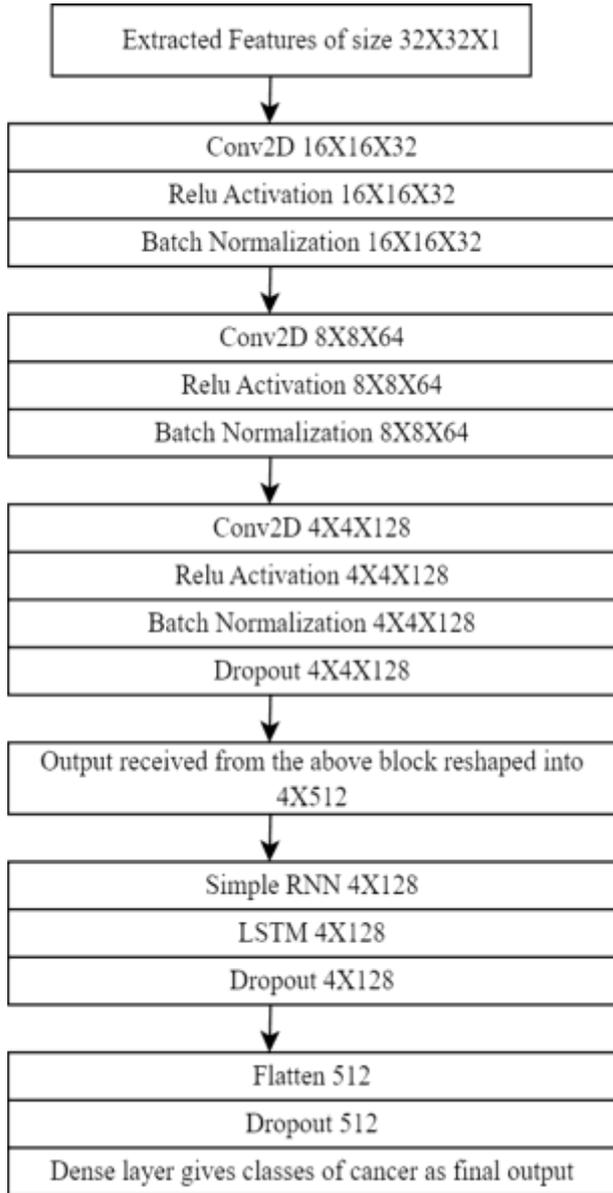
FIGURE 3. Flowchart diagram of the proposed RNN-CNN classifier.

IV. EXPERIMENTAL RESULTS AND DISCUSSION

A. EXPERIMENTAL PLATFORM

The proposed models use high power and high-performance hardware. That is why implementation is done using Graphical Processing Unit (GPU). Table 2 shows the detail of used hardware and software for our research work.

B. EVALUATION METRICS

A variety of performance metrics such as accuracy, precision, recall, F1 score and MSE have been used to evaluate effectiveness of the proposed model. The ''true positive (TP), false positive (FP), true negative (TN) and false negative (FN)'' are used to calculate these metrics.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{7}$$

TABLE 2. Experimental platform.

Parameter	Details
GPU	A100
VRAM	40 GB
RAM	86 GB
OS	Linux
Processor	i7 12 Core
Language	Python
IDE	Jupyter (Google Co-lab)

$$Recall = \frac{TP}{TP + FN} \tag{8}$$

$$Precision = \frac{TP}{TP + FP} \tag{9}$$

$$F1 - Score = \frac{2 * Precision * Recall}{Precision + Recall} \tag{10}$$

$$MSE = (1/n) * \sum (y_i - \hat{y}_i)^2 \tag{11}$$

where

n is the number of samples

y_i is the expected value of ith sample

$\hat{y}_i$ is the predicted value of ith sample

C. RESULTS

This section presents the results of our research work to show the effectiveness of our proposed methods. We have used two different gene expression datasets to validate our methods. Both the datasets are multi-class datasets. After pre-processing on data, it has been partitioned into training and testing data for further processing. Different training and testing ratios have been selected to train and test each sample of our data.

1) FEATURE EXTRACTION ON DATASET 1

Features are extracted from the dataset using proposed sandwich stacked method based on VGG16 and VGG19. The model is evaluated with existing ones such as ResNet50, Inception V3, VGG16 and VGG19 in terms of accuracy and MSE (Mean Square Error). The model has been trained and tested with multiple training and testing ratios. The different training and testing ratios taken are 60:40, 70:30, and 80:20. The proposed feature extractor gives maximum accuracy at 80:20 where 80% is the training data and 20% is the testing data. Figure 4 shows the comparison of the accuracy of the proposed feature extractor with existing model at different training testing ratios. VGG19 is having lowest accuracy of 0.82 and the proposed feature extractor is having highest accuracy of 0.954 at 80% training data. Figure 5 shows the comparison of MSE of the proposed feature extractor with existing models. The proposed model gives the lowest MSE of 0.187 and VGG19 gives the highest MSE of 0.914 at 80% training data. In each training and testing ratio, the proposed model provides the lowest MSE among all the given models.
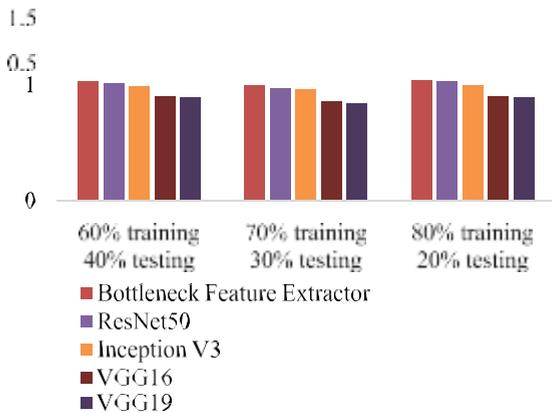
FIGURE 4. Accuracy comparison of the proposed feature extractor with existing models on dataset 1.
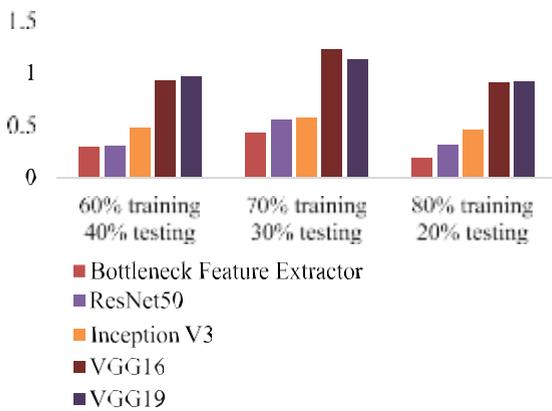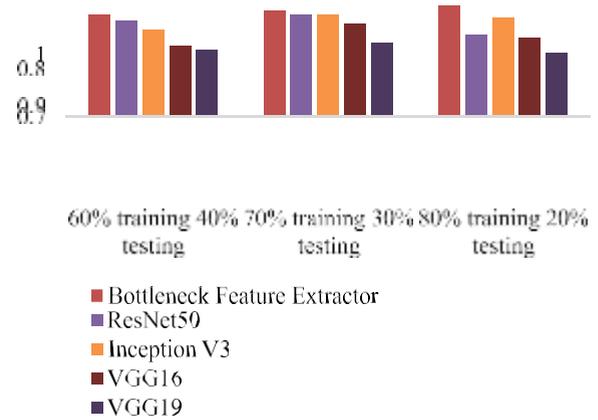

FIGURE 6. Accuracy comparison of the proposed feature extractor with existing models on dataset 2.

FIGURE 5. MSE comparison of the proposed feature extractor with existing models on dataset 1.
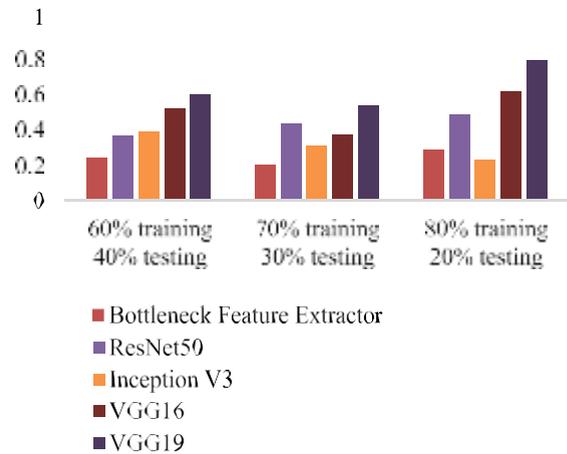
FIGURE 7. MSE comparison of the proposed feature extractor with existing models on dataset 2.

2) FEATURE EXTRACTION ON DATASET 2

Dataset 2 also contains 5 classes just like dataset 1 but the types of classes are little bit different from dataset 1. And the number of samples and features are different. Figure 6 shows the comparison of accuracy of the presented model with existing ones. The proposed model is trained and tested against different ratios of training and testing just like the dataset 1. The proposed model provides the maximum accuracy of 0.931 at 80%. VGG19 performs worst among all the models. But our proposed feature extractor performs best among all the models. Figure 7 shows the MSE comparison of the proposed model with existing ones. The proposed model provides the minimum MSE of 0.286 at 80% training. VGG19 performs worst among all the models with the highest MSE of 0.789.

3) CLASSIFICATION ON DATASET 1

One of the objectives of our research work was to propose a deep learning-based classifier which can classify cancer from gene expression data. We proposed a classifier based on RNN

and CNN deep learning models. The proposed classifier has been trained and tested on the above discussed datasets. The proposed classifier has been compared with other state of art classifiers such asVGG16, VGG19, ResNet50, InceptionV3 and MobileNet. Our proposed classifier outperforms existing methods in terms of various performance measures such as accuracy, precision, recall, F1 score and MSE. Figure 8 shows the comparison of accuracy of proposed classifier with existing classifiers. The proposed classifier provides the maximum accuracy of 0.978 at 80% training data and lowest accuracy of 0.946 at 60% training data.

Proposed classifier performs best among all the classifiers in terms of accuracy. VGG19 performs worst among all the classifiers with minimum accuracy of 0.105 and maximum accuracy of 0.2577. Figure 9 shows the comparison of precision of proposed classifiers with existing classifiers. The proposed classifier provides the maximum precision of 0.977 at 80% training and minimum precision of 0.946 at 60% training data. VGG19 performs worst among all the
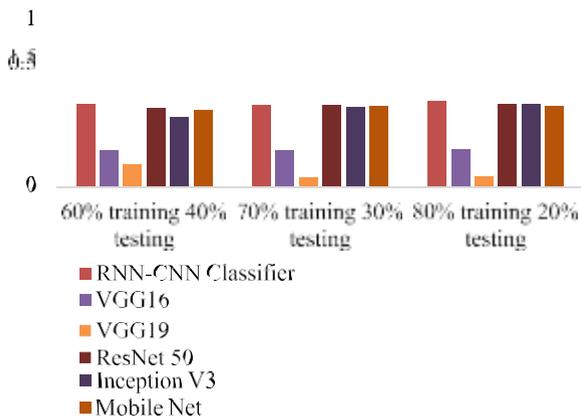
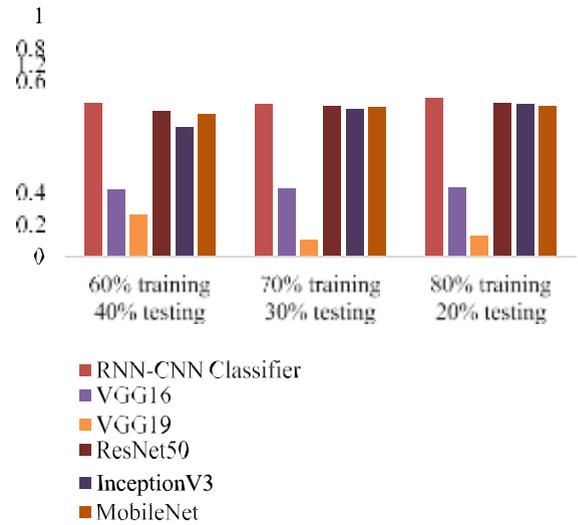**FIGURE 8.** Accuracy comparison of the proposed classifier with existing classifiers on dataset 1.

FIGURE 9. Precision comparison of the proposed classifier with existing classifiers on dataset 1.

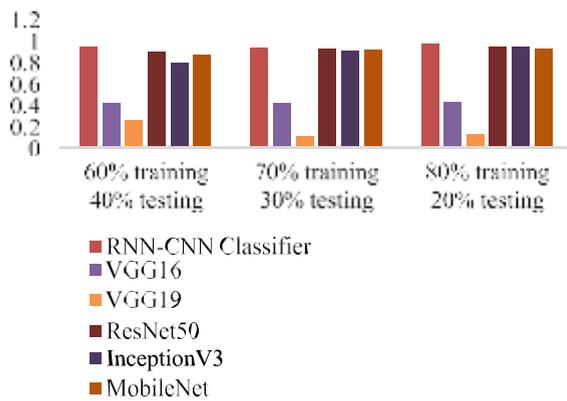**FIGURE 10.** Recall comparison of the proposed classifier with existing classifiers on dataset 1.

FIGURE 11. F1 Score comparison of the proposed classifier with existing classifiers on dataset 1.

FIGURE 12. MSE comparison of the proposed classifier with existing classifiers on dataset 1.

models with maximum precision of 0.2577 and minimum precision of 0.105.

Figure 10 shows the recall of the proposed classifier in comparison with existing classifiers. The proposed classifier provides the maximum recall of 0.977 at 80% training and minimum of 0.946 at 60% training data. VGG19 performs worst among all the models with a maximum recall of 0.257 and minimum recall of 0.105. Figure 11 shows the F1 score of the proposed classifier in comparison with existing classifiers. The proposed classifier performs best among all the models with a maximum F1 score of 0.977 and minimum F1 score of 0.946 at 80% and 60% training data respectively. Figure 12 shows the MSE of the proposed classifier in comparison with existing classifiers. The proposed classifier performs best among all the models in all the cases with a minimum MSE of 0.101 and maximum MSE of 0.208 at 80% and 60% training data respectively. The VGG19 provides the maximum MSE of 5.079 at 80% training data.

Figure 13 and 14 shows the confusion matrix for the proposed classifier for Dataset 1 at 80:20 and 70:30 respectively which shows that proposed classifier is providing good performance in classifying cancer of multiple types.

Figure 15 and 16 shows the learning and loss curve for Dataset 1 on 80% training and 20% testing data. Figure 15

shows that accuracy of both the training and the testing data is increasing with time. As the number of epochs increased,
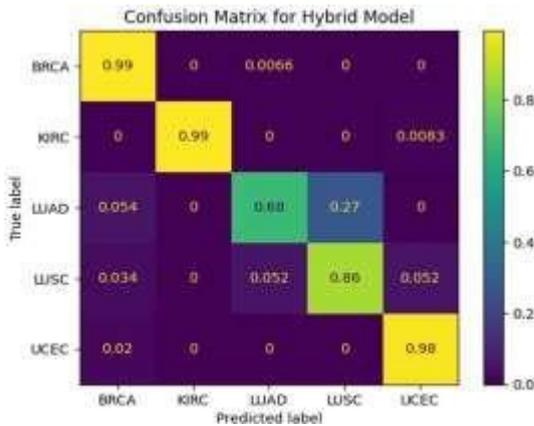
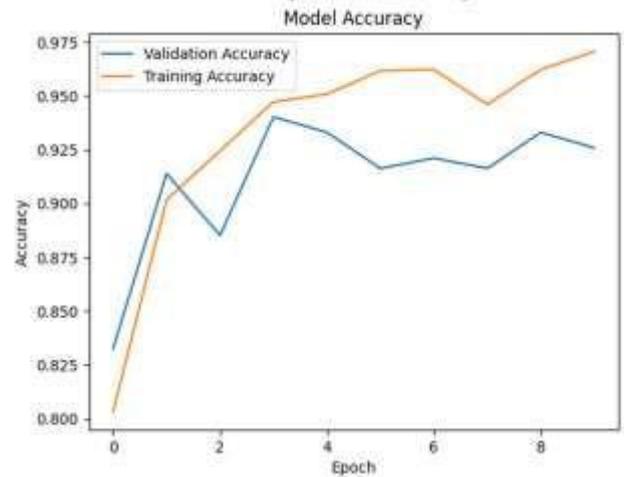**FIGURE 13.** Confusion matrix for proposed classifier on dataset 1.

FIGURE 14. Confusion matrix for proposed classifier on dataset 1.

FIGURE 15. Learning curve for dataset 1 (80:20).
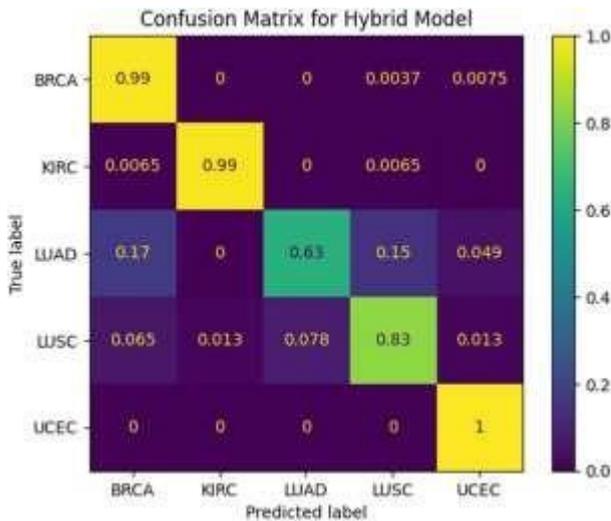
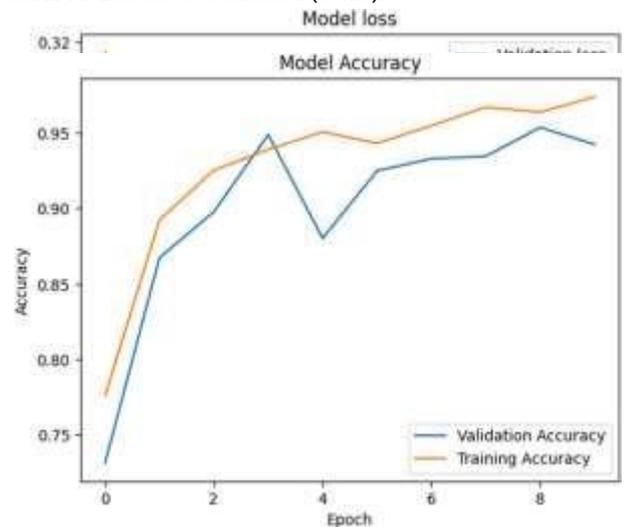**FIGURE 16.** Loss curve for dataset 1 (80:20).

FIGURE 17. Learning curve for dataset 1 (70:30).

training and validation accuracy also increased. Figure 16 shows the loss curve. It shows that training and validation loss decreased as the epochs increased. Both the plots show that our proposed model is the best fit for training and testing data. Figure 17 and 18 show the learning and the loss curve for dataset 1 at 70% training and 30% testing data.

4) ROC CURVE FOR DATASET 1

The performance of classification model is calculated using the receiver operating characteristic curve. ROC is based on true positive rate (TPR) and false positive rate (FPR). Figure 19 to Figure 24 shows the ROC curve for different models along with proposed classifier for dataset 1 at 80:20. These curves show that the accuracy of the proposed model is more than other existing models.

5) CLASSIFICATION ON DATASET 2

Proposed RNN-CNN classifier has been trained and tested on dataset 2 as well for different training and testing ratios

just like dataset 1. Dataset 2 has more features as compared to dataset 1. The proposed classifier has outperformed the
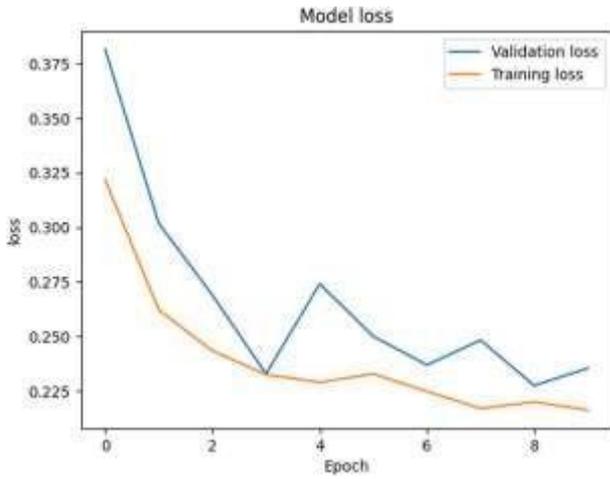
FIGURE 18. Loss curve for dataset 1 (70:30).

FIGURE 19. ROC curve for VGG16 on dataset 1.

FIGURE 20. ROC curve for VGG19 on dataset 1.

existing classifiers in terms of accuracy, precision, recall, F1 score and MSE. Figure 25 shows the accuracy of the
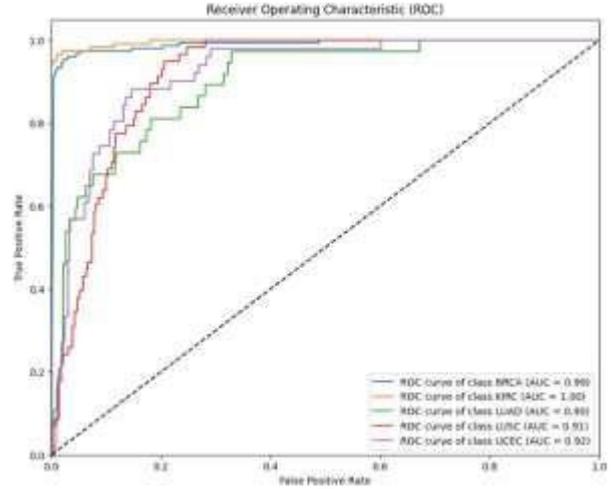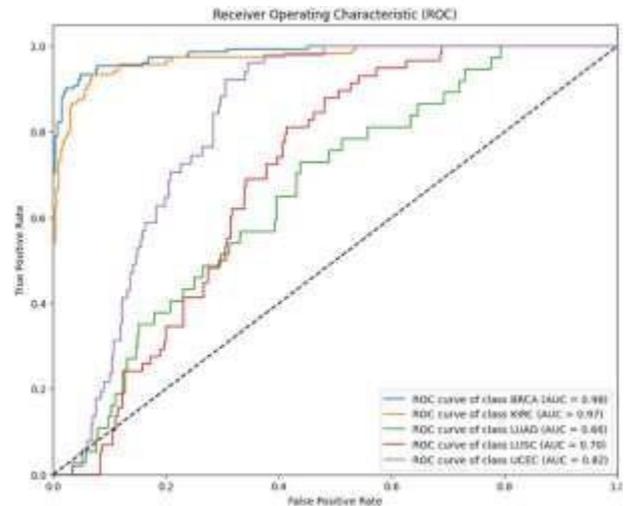

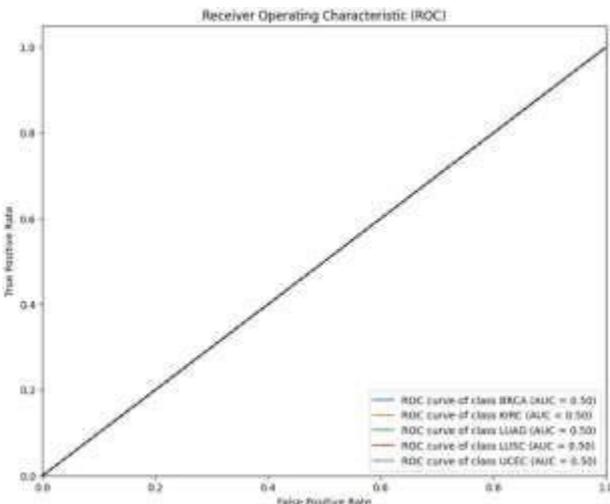
FIGURE 21. ROC curve for ResNet50 on dataset 1.

FIGURE 22. ROC curve for InceptionV3 on dataset 1.

proposed classifier in comparison with existing classifiers such as VGG16, VGG19, ResNet50, InceptionV3 and MobileNet. The proposed classifier performs best among all the classifiers with a maximum accuracy of 0.994 and minimum accuracy of 0.946 at 80% and 60% training respectively. VGG19 performs worst among all the classifiers with a maximum accuracy of 0.361 and minimum accuracy of 0.149. Figure 26 shows the comparison of precision of proposed classifier with existing classifiers. The proposed classifier performs best among all in terms of precision as well with a maximum precision of 0.994 and minimum precision of 0.945 at 80% and 60% training data respectively. VGG19 performs worst among all the classifiers with a minimum precision of 0.149 and maximum precision of 0.361.

Figure 27 shows the recall of the proposed classifier in comparison with existing classifiers. The proposed performs
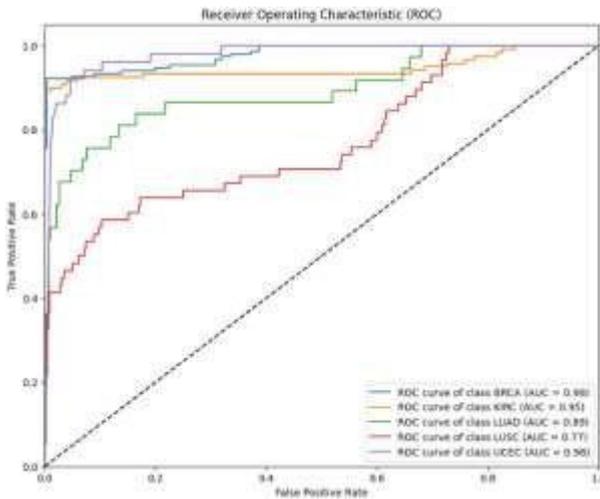
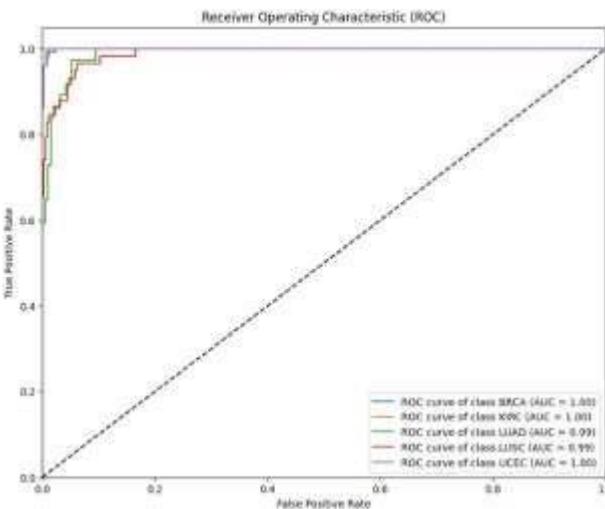**FIGURE 23.** ROC curve for MobileNet on dataset 1.

FIGURE 24. ROC curve for proposed classifier on dataset 1.
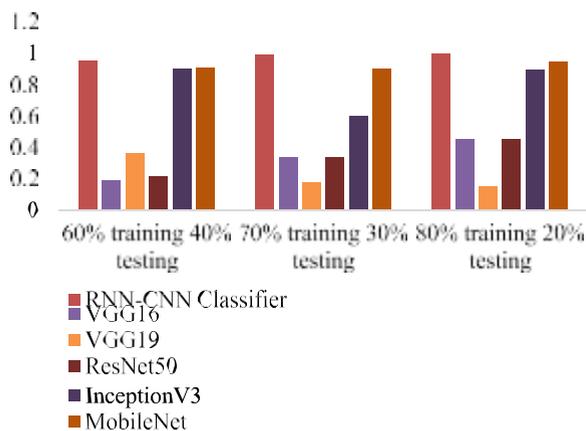
FIGURE 25. Accuracy comparison of the proposed classifier with existing classifiers on dataset 2.
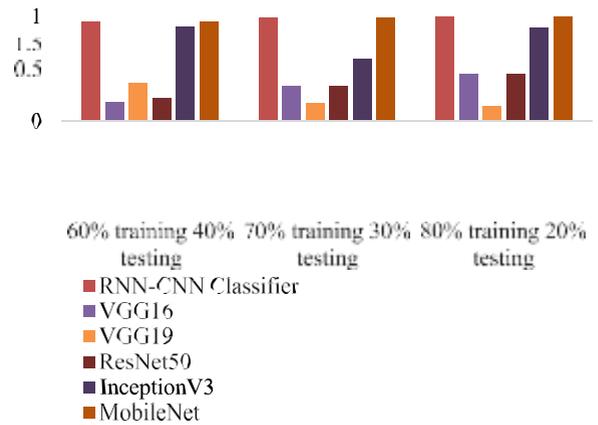

FIGURE 26. Precision comparison of the proposed classifier with existing classifiers on dataset 2.
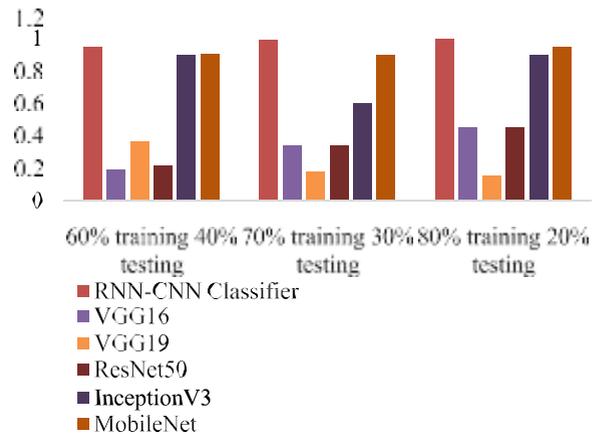

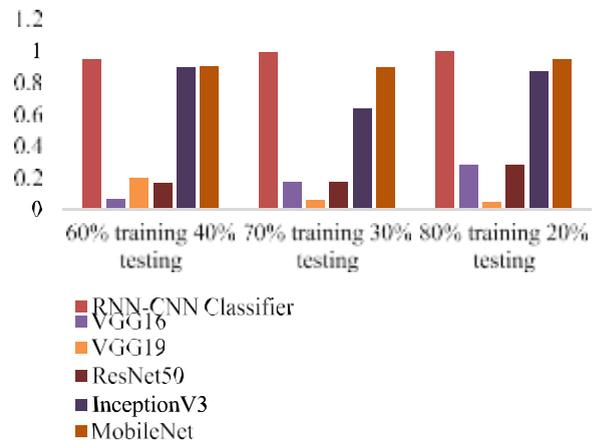**FIGURE 27.** Recall comparison of the proposed classifier with existing classifiers on dataset 2.

FIGURE 28. F1 Score comparison of the proposed classifier with existing classifiers on dataset 2.

best among all the classifiers jin terms of recall. It provides the maximum recall of 0.994 and minimum recall of 0.945 at

80% and 60% training data respectively. VGG19 performs worst among all the classifiers with a minimum precision of 0.149 and maximum precision of 0.361.
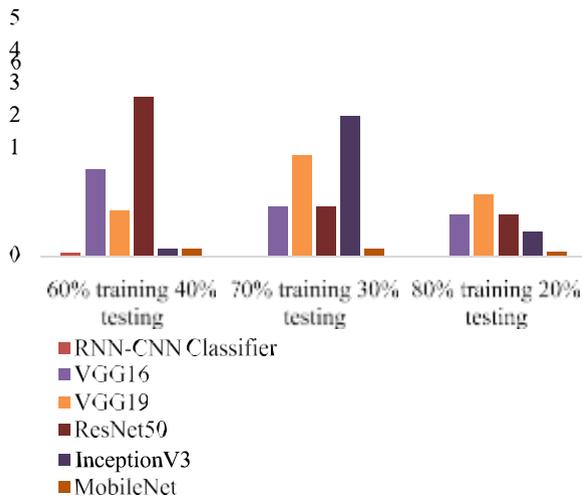
FIGURE 29. MSE comparison of the proposed classifier with existing classifiers on dataset 2.
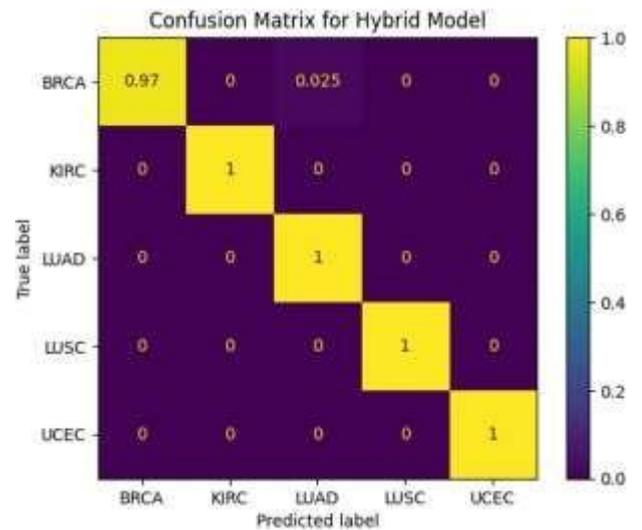

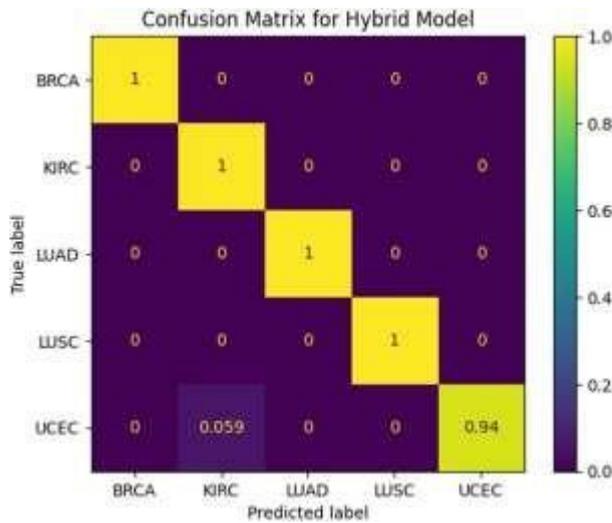**FIGURE 31.** Confusion matrix for proposed classifier on dataset 2.

FIGURE 30. Confusion matrix for proposed classifier on dataset 2.

FIGURE 32. Learning curve for dataset 2 (80:20).

Figure 28 shows the F1 score of the proposed classifier in comparison with existing classifiers. The proposed method performs well among all the classifiers in terms of F1 score with a maximum F1 score of 0.994 at 80% training data. It provides the minimum F1 score of 0.947 at 60% training data. On the other hand, VGG19 performs worst among all the models. It provides a maximum F1 score of 0.192 and minimum F1 score of 0.039. Figure 29 shows the comparison of MSE of the proposed classifier with existing models. The proposed model also performs best among all the models in terms of MSE as well. It provides the minimum MSE of 0.006 at 80% training data. But it also provides the maximum MSE of 0.125 at 60% training data. VGG19 performs worst among all the classifiers with a maximum MSE of 3.116 and minimum MSE of 1.424.

Figure 30 and 31 shows the confusion matrix for dataset 2. Figure 30 presents the confusion matrix for 80% training and 20% testing data. Similarly, Figure 31 presents the confusion
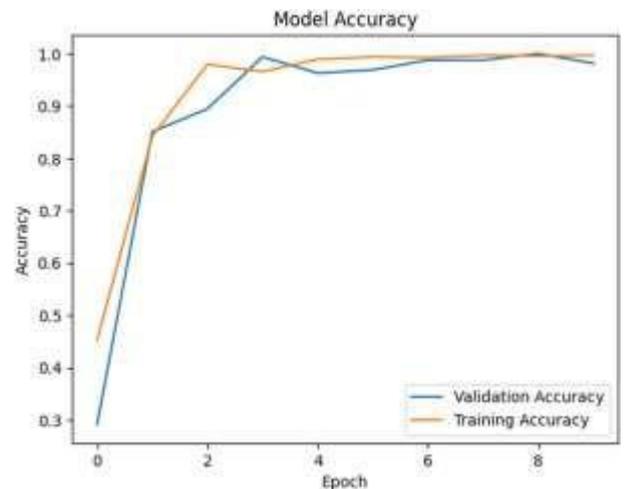
matrix for 70% of the training and 30% of the testing data which shows that proposed model is providing good classification results.

Figure 32, 33 and Figure 34, 35 shows the learning and loss curves for dataset 2 at 80:20 and 70:30 training and testing data respectively. Learning curves show that as the number of epochs was increased, both training and validation accuracy increased. Loss curves show that as the number of epochs was raised, both training and validation loss decreased.

6) ROC CURVE FOR DATASET 2

Figure 36 to 41 shows the ROC curve for the classification task on dataset 2 at 80% training and 20% testing data. The proposed classifier ROC curve is showing best performance as compared to other existing classifiers.

The experimental work shows the effectiveness of our proposed models in terms of various performance measures such
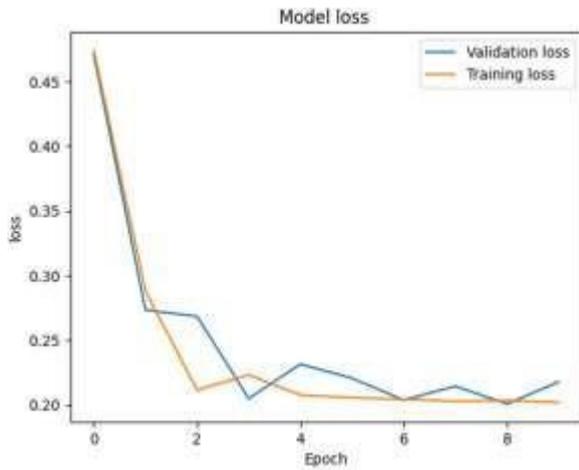
FIGURE 33. Loss curve for dataset 2 (80:20).
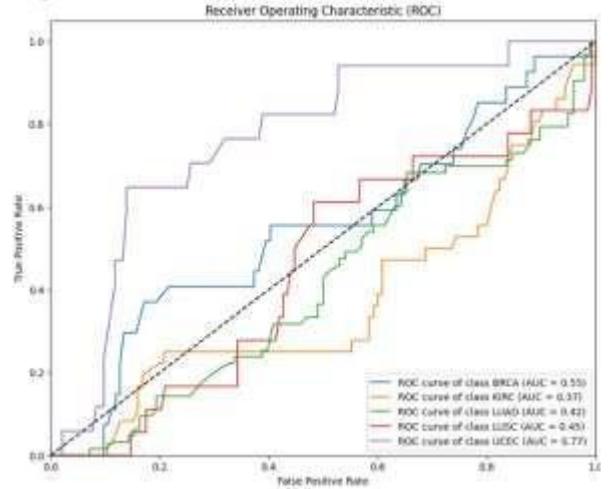
FIGURE 36. ROC curve for VGG16 on dataset 2.
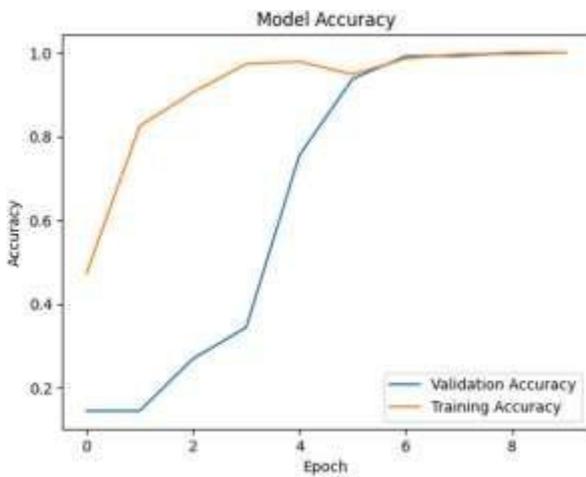
FIGURE 34. Learning curve for dataset 2 (70:30).

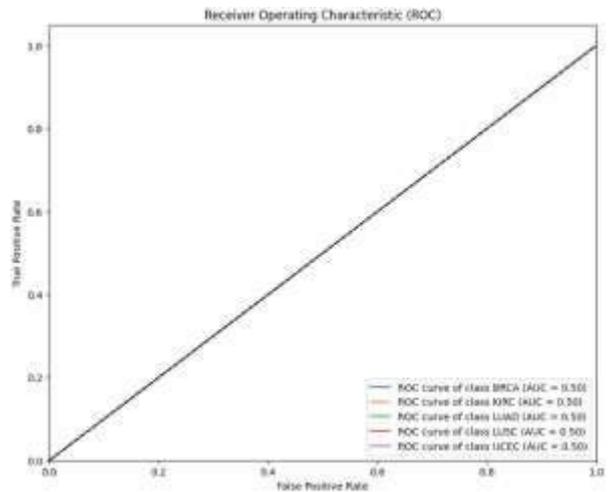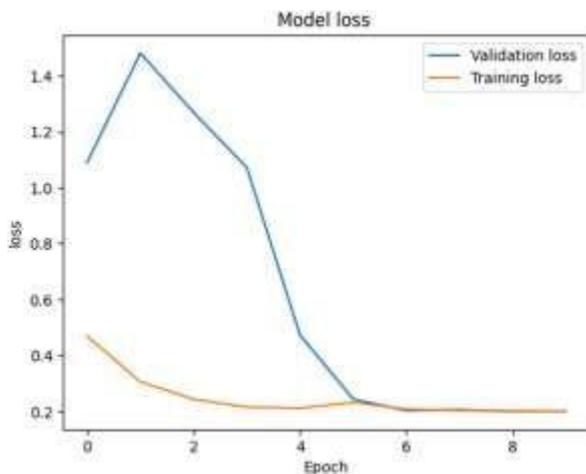**FIGURE 37.** ROC curve for VGG19 on dataset 2.

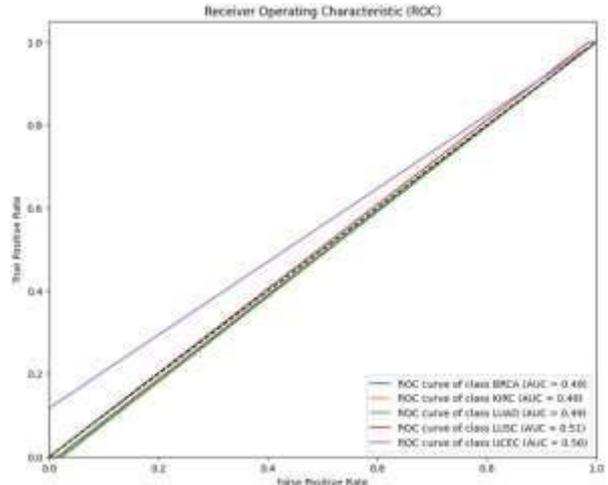**FIGURE 35.** Loss curve for dataset 2 (70:30).

FIGURE 38. ROC curve for ResNet50 on dataset 2.

as accuracy, precision, recall, F1 score, and MSE as shown in Table 3. Our proposed methods perform best among all the

given feature extractors and classifiers. These measures are calculated to show the effectiveness and statistical analysis
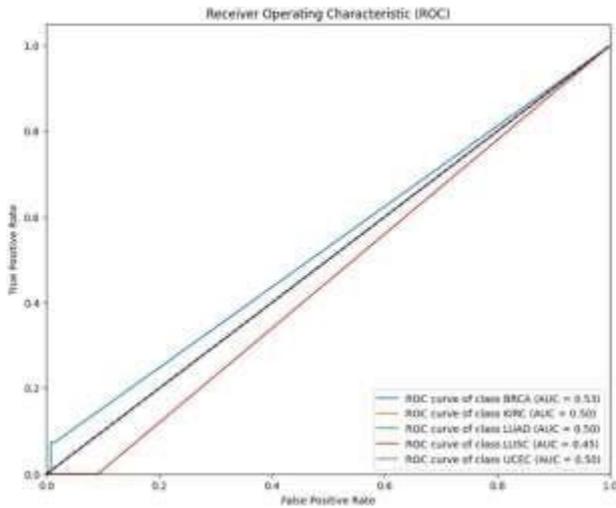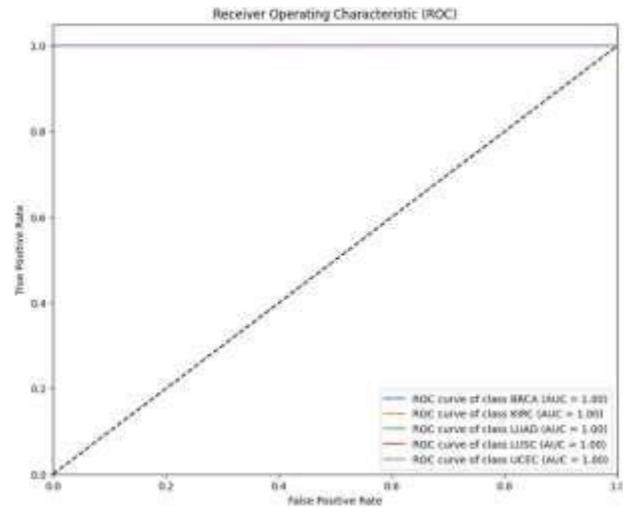
FIGURE 39. ROC curve for InceptionV3 on dataset 2.

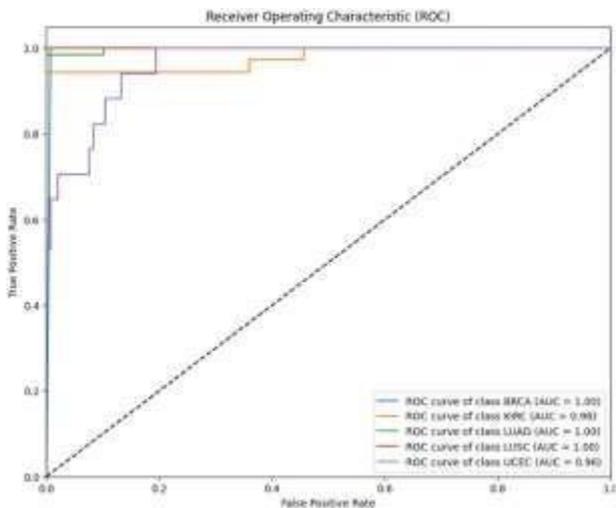**FIGURE 41.** ROC curve for proposed classifier on dataset 2.

FIGURE 40. ROC curve for MobileNet on dataset 2.

TABLE 3. Performance metrics of various models along with proposed classifier.

Data set	Model	F1-Score	Accuracy	Precision	Recall	MSE
Data set-1	VGG16	0.226	0.847	0.841	0.398	0.773
	VGG19	0.203	0.830	0.887	0.459	0.789
	ResNet 50	0.962	0.926	0.926	0.909	0.635
	Inception V3	0.951	0.903	0.903	0.765	0.662
	Mobile Net	0.947	0.951	0.951	0.817	0.437
	CNN-RNN Classifier	0.968	0.971	0.971	0.977	0.286
Data set-2	VGG16	0.216	0.732	0.863	0.411	0.782
	VGG19	0.225	0.643	0.895	0.442	0.785
	ResNet 50	0.943	0.917	0.955	0.928	0.654
	Inception V3	0.932	0.922	0.938	0.831	0.661
	Mobile Net	0.943	0.904	0.940	0.945	0.437
	CNN-RNN Classifier	0.951	0.968	0.963	0.975	0.294

of our proposed model. To show that the proposed model is better and different from existing models, the values of the performance measures of the proposed model should be different from the existing ones. And this is shown in Table 3 that proposed classifier is a superior classifier as compared to the other existing models. Also, the best results are obtained at 80:20 ratio of training and testing data. Even the proposed feature extractor provides significant results when extracting the most prominent features from dataset having multiple classes as shown in the results of dataset 2.

D. DISCUSSION

The research work focused on classifying distinct cancer types from gene expression data using DL methods. Cancer is one of the life-threatening diseases, so it requires a major concern. So, we proposed two models, one for feature extraction and another for classification. Feature extractor extracts

features from the data using VGG16 and VGG19, pretrained models. The extracted features are then given to the proposed classifier which is based on CNN and RNN for further classification. Our proposed model provides promising results as compared to existing models from the literature. Methods in [44] have provided an accuracy of 96.90 on the dataset 1. However, our proposed classifier is providing 97.8% accuracy on dataset 1 and 99% accuracy on dataset 2. This shows that our method is performing well than other state-of-the-art

methods. The presented research work is novel in terms of the number of layers used for CNN and RNN. 3 convolutional blocks along with activation, batch normalization and drop out are used in CNN and in RNN, layers of LSTM and simple RNN have been added.

V. CONCLUSION AND FUTURE SCOPE

Cancer, a highly life-threatening disease, necessitates timely detection and treatment to save patients' lives. In the current era, a range of tests, including CT scans, MRIs, and genetic data analysis, are available for cancer detection. However, the use of microarray technologies for detecting diseases at the genetic level is gaining popularity. One significant challenge with gene expression data is the scarcity of samples and the abundance of features. To tackle this concern, we propose a method in this paper that has two stages. The first stage is the feature extractor, called the SSB feature extractor, which leverages a combination of pre-trained VGG16 and VGG19 models to extract features from gene expression data. The second stage introduces a DL-based classifier, namely the RNN-CNN classifier, designed to classify samples into multiple classes. We evaluate our proposed classifier with the latest methods such as VGG16, VGG19, ResNet50, InceptionV3, and MobileNet, and demonstrate superior performance across various evaluation metrics. The suggested method has been implemented on the secondary datasets. So, to support this, as a work of future, we plan to implement the proposed methodology on the real time data to check the effectiveness of our work. In addition to this, we can implement a methodology to check for the cancer traces if any in the embryo so that necessary actions can be taken on time to avoid any future consequences.

REFERENCES

[1] F. V. Filipp, ''Precision medicine driven by cancer systems biology,'' *Cancer Metastasis Rev.*, vol. 36, no. 1, pp. 91–108, Mar. 2017.

[2] T. C. Archer et al., ''Systems approaches to cancer biology systems approaches to cancer biology,'' *Cancer Res.*, vol. 76, no. 23, pp. 6774–6777, 2016.

[3] T. Vos et al., ''Global, regional, and national incidence, prevalence, and years lived with disability for 310 diseases and injuries, 1990–2015: A systematic analysis for the global burden of disease study 2015,'' *Lancet*, vol. 388, no. 10053, pp. 1545–1602, 2016.

[4] S. Liu, C. Xu, Y. Zhang, J. Liu, B. Yu, X. Liu, and M. Dehmer, ''Feature selection of gene expression data for cancer classification using double RBF-kernels,'' *BMC Bioinf.*, vol. 19, no. 1, pp. 1–14, Dec. 2018.

[5] F. Finotello and B. Di Camillo, ''Measuring differential gene expression with RNA-seq: Challenges and strategies for data analysis,'' *Briefings Funct. Genomics*, vol. 14, no. 2, pp. 130–142, Mar. 2015.

[6] M. Maienschein-Cline, J. Zhou, K. P. White, R. Sciammas, and A. R. Dinner, ''Discovering transcription factor regulatory targets using gene expression and binding data,'' *Bioinformatics*, vol. 28, no. 2, pp. 206–213, Jan. 2012.

[7] E. E. Schadt et al., ''An integrative genomics approach to infer causal associations between gene expression and disease,'' *Nature Genet.*, vol. 37, no. 7, pp. 710–717, Jul. 2005.

[8] K. Shabana, K. A. Nazeer, M. Pradhan, and M. Palakal, ''A computational method for drug repositioning using publicly available gene expression data,'' *BMC Bioinf.*, vol. 16, no. S17, pp. 1–9, Dec. 2015.

[9] P. Danaee, R. Ghaeini, and D. A. Hendrix, ''A deep learning approach for cancer detection and relevant gene identification,'' in *Proc. Biocomputing*, Jan. 2017, pp. 219–229.

[10] A. Alizadeh et al., ''Distinct types of diffuse large B-cell lymphoma identified by gene expression profiling,'' *Nature*, vol. 403, no. 6769, pp. 503–511, 2000.

[11] L. Chen, X. Pan, Y.-H. Zhang, M. Liu, T. Huang, and Y.-D. Cai, ''Classification of widely and rarely expressed genes with recurrent neural network,'' *Comput. Struct. Biotechnol. J.*, vol. 17, pp. 49–60, Jan. 2019.

[12] M. K. Elbashir, M. Ezz, M. Mohammed, and S. S. Saloum, ''Lightweight convolutional neural network for breast cancer classification using RNA-seq gene expression data,'' *IEEE Access*, vol. 7, pp. 185338–185348, 2019.

[13] H. Wang, C. Li, J. Zhang, J. Wang, Y. Ma, and Y. Lian, ''A new LSTM-based gene expression prediction model: L-GEPM,'' *J. Bioinf. Comput. Biol.*, vol. 17, no. 04, Aug. 2019, Art. no. 1950022.

[14] N. E. M. Khalifa, M. H. N. Taha, D. E. Ali, A. Slowik, and A. E. Hassanien, ''Artificial intelligence technique for gene expression by tumor RNA-seq data: A novel optimized deep learning approach,'' *IEEE Access*, vol. 8, pp. 22874–22883, 2020.

[15] T. Khorshed, M. N. Moustafa, and A. Rafea, ''Deep learning for multi-tissue cancer classification of gene expressions (GeneXNet),'' *IEEE Access*, vol. 8, pp. 90615–90629, 2020.

[16] O. F. Özgül, B. Bardak, and M. Tan, ''A convolutional deep clustering framework for gene expression time series,'' *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 18, no. 6, pp. 2198–2207, Nov. 2021.

[17] K. Tan, W. Huang, X. Liu, J. Hu, and S. Dong, ''A hierarchical graph convolution network for representation learning of gene expression data,'' *IEEE J. Biomed. Health Informat.*, vol. 25, no. 8, pp. 3219–3229, Aug. 2021.

[18] D. Sun, M. Wang, and A. Li, ''A multimodal deep neural network for human breast cancer prognosis prediction by integrating multi-dimensional data,'' *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 16, no. 3, pp. 841–850, May 2019.

[19] M. Khalsan, L. R. Machado, E. S. Al-Shamery, S. Ajit, K. Anthony, M. Mu, and M. O. Agyeman, ''A survey of machine learning approaches applied to gene expression analysis for cancer prediction,'' *IEEE Access*, vol. 10, pp. 27522–27534, 2022.

[20] T. K. Bamunu Mudiyanselage, X. Xiao, Y. Zhang, and Y. Pan, ''Deep fuzzy neural networks for biomarker selection for accurate cancer detection,'' *IEEE Trans. Fuzzy Syst.*, vol. 28, no. 12, pp. 3219–3228, Dec. 2020.

[21] J. Zhang, Q. Chen, and B. Liu, ''DeepDRBP-2L: A new genome annotation predictor for identifying DNA-binding proteins and RNA-binding proteins using convolutional neural network and long short-term memory,'' *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 18, no. 4, pp. 1451–1463, Jul. 2021.

[22] B. Lantz, *Machine Learning With R: Expert Techniques for Predictive Modeling*. Birmingham, U.K.: Packt, 2019.

[23] S. V. Mahadevkar, B. Khemani, S. Patil, K. Kotecha, D. R. Vora, A. Abraham, and L. A. Gabralla, ''A review on machine learning styles in computer vision—Techniques and future directions,'' *IEEE Access*, vol. 10, pp. 107293–107329, 2022.

[24] S. Sah, ''Machine learning: A review of learning types,'' *Preprints*, 2020, Art. no. 2020070230, doi: 10.20944/preprints202007.0230.v1.

[25] J. Salas, F. de Barros Vidal, and F. Martinez-Trinidad, ''Deep learning: Current state,'' *IEEE Latin Amer. Trans.*, vol. 17, no. 12, pp. 1925–1945, Dec. 2019.

[26] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.

[27] T. Zhu, K. Li, P. Herrero, and P. Georgiou, ''Deep learning for diabetes: A systematic review,'' *IEEE J. Biomed. Health Informat.*, vol. 25, no. 7, pp. 2744–2757, Jul. 2021.

[28] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.

[29] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, ''Gradient-based learning applied to document recognition,'' *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[30] Y. Bengio, P. Simard, and P. Frasconi, ''Learning long-term dependencies with gradient descent is difficult,'' *IEEE Trans. Neural Netw.*, vol. 5, no. 2, pp. 157–166, Mar. 1994.

[31] G. E. Hinton, S. Osindero, and Y.-W. Teh, ''A fast learning algorithm for deep belief nets,'' *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, Jul. 2006.

[32] R. K. Mondol, N. D. Truong, M. Reza, S. Ippolito, E. Ebrahimie, and O. Kavehei, ''AFExNet: An adversarial autoencoder for differentiating breast cancer sub-types and extracting biologically relevant genes,'' *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 19, no. 4, pp. 2060–2070, Jul. 2022.

[33] W. Zhang, R. Li, T. Zeng, Q. Sun, S. Kumar, J. Ye, and S. Ji, ''Deep model based transfer and multi-task learning for biological image analysis,'' in *Proc. 21st ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2015, pp. 1475–1484.

[34] J. Khan, J. S. Wei, M. Ringnér, L. H. Saal, M. Ladanyi, F. Westermann, F. Berthold, M. Schwab, C. R. Antonescu, C. Peterson, and P. S. Meltzer, ''Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks,'' *Nature Med.*, vol. 7, no. 6, pp. 673–679, 2001.

[35] S. Weng, C. Zhang, and X. Zhang, ''PCA-FA: Applying supervised learning to analyze gene expression data,'' *Tsinghua Sci. Technol.*, vol. 9, no. 4, pp. 428–434, Aug. 2004.

[36] R. Xu, G. Anagnostopoulos, and D. Wunsch, ''Multiclass cancer classification using semisupervised ellipsoid ARTMAP and particle swarm optimization with gene expression data,'' *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 4, no. 1, pp. 65–77, Jan. 2007.

[37] Z. Yu and H.-S. Wong, ''Class discovery from gene expression data based on perturbation and cluster ensemble,'' *IEEE Trans. Nanobiosci.*, vol. 8, no. 2, pp. 147–160, Jun. 2009.

[38] A. B. Ashraf, S. C. Gavenonis, D. Daye, C. Mies, M. A. Rosen, and D. Kontos, ''A multichannel Markov random field framework for tumor segmentation with an application to classification of gene expression-based breast cancer recurrence risk,'' *IEEE Trans. Med. Imag.*, vol. 32, no. 4, pp. 637–648, Apr. 2013.

[39] B. Liao, Y. Jiang, W. Liang, W. Zhu, L. Cai, and Z. Cao, ''Gene selection using locality sensitive Laplacian score,'' *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 11, no. 6, pp. 1146–1156, Nov. 2014.

[40] J.-X. Liu, Y. Xu, C.-H. Zheng, H. Kong, and Z.-H. Lai, ''RPCA-based tumor classification using gene expression data,'' *IEEE/ACM Trans. Comput. Biol. Bioinf.*, vol. 12, no. 4, pp. 964–970, Jul. 2015.

[41] Y. Chen, Y. Li, R. Narayan, A. Subramanian, and X. Xie, ''Gene expression inference with deep learning,'' *Bioinformatics*, vol. 32, no. 12, pp. 1832–1839, Jun. 2016.

[42] L. Chen, X. Pan, Y.-H. Zhang, M. Liu, T. Huang, and Y.-D. Cai, ''Classification of widely and rarely expressed genes with recurrent neural network,'' *Comput. Struct. Biotechnol. J.*, vol. 17, pp. 49–60, Jan. 2019.

[43] M. K. Elbashir, M. Ezz, M. Mohammed, and S. S. Saloum, ''Lightweight convolutional neural network for breast cancer classification using RNA-seq gene expression data,'' *IEEE Access*, vol. 7, pp. 185338–185348, 2019.

[44] N. E. M. Khalifa, M. H. N. Taha, D. Ezzat Ali, A. Slowik, and A. E. Hassanien, ''Artificial intelligence technique for gene expression by tumor RNA-seq data: A novel optimized deep learning approach,'' *IEEE Access*, vol. 8, pp. 22874–22883, 2020.

[45] C.-H. Zheng, T.-Y. Ng, L. Zhang, C.-K. Shiu, and H.-Q. Wang, ''Tumor classification based on non-negative matrix factorization using gene expression data,'' *IEEE Trans. Nanobiosci.*, vol. 10, no. 2, pp. 86–93, Jun. 2011.

[46] H. Wang, C. Li, J. Zhang, J. Wang, Y. Ma, and Y. Lian, ''A new LSTM-based gene expression prediction model: L-GEPM,'' *J. Bioinf. Comput. Biol.*, vol. 17, no. 4, Aug. 2019, Art. no. 1950022.

[47] T. Khorshed, M. N. Moustafa, and A. Rafea, ''Deep learning for multi-tissue cancer classification of gene expressions (GeneXNet),'' *IEEE Access*, vol. 8, pp. 90615–90629, 2020.

[48] J. Xu, P. Wu, Y. Chen, Q. Meng, H. Dawood, and M. M. Khan, ''A novel deep flexible neural forest model for classification of cancer subtypes based on gene expression data,'' *IEEE Access*, vol. 7, pp. 22086–22095, 2019.

[49] T. Thakur, I. Batra, M. Luthra, S. Vimal, G. Dhiman, A. Malik, and M. Shabaz, ''Gene expression-assisted cancer prediction techniques,'' *J. Healthcare Eng.*, vol. 2021, pp. 1–9, Aug. 2021.

[50] T. Thakur, I. Batra, and A. Malik, ''A comparative analysis of various machine learning and deep learning models for gene expression,'' in *Proc. Int. Conf. Comput. Sci. (ICCS)*, Dec. 2021, pp. 139–142.

[51] T. Thakur, I. Batra, and A. Malik, ''Performance evaluation of various machine learning and deep learning models for gene expression,'' *J. Phys., Conf. Ser.*, vol. 2327, no. 1, Aug. 2022, Art. no. 012034.

[52] W. W. Lo, X. Yang, and Y. Wang, ''An xception convolutional neural network for malware classification with transfer learning,'' in *Proc. 10th IFIP Int. Conf. New Technol., Mobility Secur. (NTMS)*, Jun. 2019, pp. 1–5.

[53] S. Khaki, H. Pham, Y. Han, A. Kuhl, W. Kent, and L. Wang, ''DeepCorn: A semi-supervised deep learning method for high-throughput image-based corn kernel counting and yield estimation,'' *Knowl.-Based Syst.*, vol. 218, Apr. 2021, Art. no. 106874.

[54] E. Rezende, G. Ruppert, T. Carvalho, A. Theophilo, F. Ramos, and P. de Geus, ''Malicious software classification using VGG16 deep neural network's bottleneck features,'' in *Information Technology-New Generations*. Cham, Switzerland: Springer, 2018, pp. 51–59.

[55] D. Theckedath and R. R. Sedamkar, ''Detecting affect states using VGG16, ResNet50 and SE-ResNet50 networks,'' *Social Netw. Comput. Sci.*, vol. 1, no. 2, pp. 1–7, Mar. 2020.

[56] Z. N. K. Swati, Q. Zhao, M. Kabir, F. Ali, Z. Ali, S. Ahmed, and J. Lu, ''Content-based brain tumor retrieval for MR images using transfer learning,'' *IEEE Access*, vol. 7, pp. 17809–17822, 2019.

[57] M. Bansal, M. Kumar, M. Sachdeva, and A. Mittal, ''Transfer learning for image classification using VGG19: Caltech-101 image data set,'' *J. Ambient Intell. Humanized Comput.*, vol. 14, no. 4, pp. 3609–3620, Apr. 2023.

[58] A. Shrestha and A. Mahmood, ''Review of deep learning algorithms and architectures,'' *IEEE Access*, vol. 7, pp. 53040–53065, 2019.

[59] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, ''Gradient-based learning applied to document recognition,'' *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[60] Y. LeCun and Y. Bengio, ''Convolutional networks for images, speech, and time series,'' in *The Handbook of Brain Theory and Neural Networks*, vol. 3361, no. 10. Cambridge, MA, USA: MIT Press, 1995, p. 1995.

[61] L. Medsker and L. C. Jain, Eds., *Recurrent Neural Networks: Design and Applications*. Boca Raton, FL, USA: CRC Press, 1999.

[62] K. L. Du and M. N. Swamy, *Neural Networks and Statistical Learning*. London, U.K.: Springer, 2013.

[63] J. A. Cruz and D. S. Wishart, ''Applications of machine learning in cancer prediction and prognosis,'' *Cancer Informat.*, vol. 2, Jan. 2006.

[64] *Cancer Types: RNA Sequencing Values From Tumor Samples/Tissues*. Accessed: Jul. 2020. [Online]. Available: https://data.mendeley.com/datasets/sf5n64hydt/1

[65] *GSE45827*. Accessed: Jan. 2022. [Online]. Available: https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE45827

[66] M. Uhlen et al., ''A pathology atlas of the human cancer transcriptome,'' *Science*, vol. 357, no. 6352, p. eaan2507, 2017.

[67] *DeepDRBP-2L*. Accessed: Apr. 2023. [Online]. Available: http://bliulab.net/DeepDRBP-2L/

[68] A. Bhattacharjee, W. G. Richards, J. Staunton, C. Li, S. Monti, P. Vasa, C. Ladd, J. Beheshti, R. Bueno, M. Gillette, M. Loda, G. Weber, E. J. Mark, E. S. Lander, W. Wong, B. E. Johnson, T. R. Golub, D. J. Sugarbaker, and M. Meyerson, ''Classification of human lung carcinomas by mRNA expression profiling reveals distinct adenocarcinoma subclasses,'' *Proc. Nat. Acad. Sci. USA*, vol. 98, no. 24, pp. 13790–13795, Nov. 2001.

[69] H. Myriam, A. A. Abdelhamid, E. M. El-Kenawy, A. Ibrahim, M. M. Eid, M. M. Jamjoom, and D. S. Khafaga, ''Advanced meta-heuristic algorithm based on particle swarm and Al-biruni earth radius optimization methods for oral cancer detection,'' *IEEE Access*, vol. 11, pp. 23681–23700, 2023.

[70] S. Baro. (2020). *Oral Cancer (Lips and Tongue) Images Dataset*. Accessed: Oct. 4, 2022. [Online]. Available: https://www.kaggle.com/code/shivam17299/oral-cancer-lips-and-tongue-images-dataset/data

[71] S. Rajendran, S. K. Rajagopal, T. Thanarajan, K. Shankar, S. Kumar, N. M. Alsubaie, M. K. Ishak, and S. M. Mostafa, ''Automated segmentation of brain tumor MRI images using deep learning,'' *IEEE Access*, vol. 11, pp. 64758–64768, 2023.

[72] A. Jabbar, S. Naseem, T. Mahmood, T. Saba, F. S. Alamri, and A. Rehman, ''Brain tumor detection and multi-grade segmentation through hybrid caps-VGGNet model,'' *IEEE Access*, vol. 11, pp. 72518–72536, 2023.

[73] Z. Sani, R. Prasad, and E. K. M. Hashim, ''Breast cancer classification using equivariance transition in group convolutional neural networks,'' *IEEE Access*, vol. 11, pp. 28454–28465, 2023.

[74] M. Subramanian, J. Cho, V. E. Sathishkumar, and O. S. Naren, ''Multiple types of cancer classification using CT/MRI images based on learning without forgetting powered deep learning models,'' *IEEE Access*, vol. 11, pp. 10336–10354, 2023.

[75] *Acute Lymphoblastic Leukemia (ALL) Image Dataset*, Kaggle, San Francisco, CA, USA, 2021.

[76] F. Spanhol. *Breast Cancer Histopathological Database*. Accessed: Nov. 30, 2022. [Online]. Available: https://www.kaggle.com/datasets/anaselmasry/breast-cancer-dataset

[77] SIPaKMeD. *A New Dataset for Feature and Image Based Classification of Normal and Pathological Cervical Cells in Pap Smear Images*. Accessed: Oct. 2018. [Online]. Available: https://www.kaggle.com/datasets/prahladmehandiratta/cervical-cancer-largest-dataset-sipakmed

[78] *CT Kidney Dataset: Normal-Cyst-Tumor and Stone*. Accessed: Dec. 2021. [Online]. Available: https://www.kaggle.com/ datasets/nazmul0087/ctkidneydataset-normal-cyst-tumor-and-stone

[79] A. A. Borkowski, L. B. Thomas, C. P. Wilson, L. A. DeLand, and S. M. Mastorides. *Lung and Colon Cancer Histopathological Image Dataset (LC25000)*. Accessed: Dec. 2019. [Online]. Available: https://www.kaggle.com/datasets/andrewmvd/lung-and-colon-cancer-histopathological-images

[80] N. V. Orlov, W. W. Chen, D. M. Eckley, T. J. Macura, L. Shamir, E. S. Jaffe, and I. G. Goldberg, ''Automatic classification of lymphoma images with transform-based global features,'' *IEEE Trans. Inf. Technol. Biomed.*, vol. 14, no. 4, pp. 1003–1013, Jul. 2010.

[81] *Histopathologic Oral Cancer Detection Using CNNs*. Accessed: Dec. 2020. [Online]. Available: https://www.kaggle.com/ datasets/ashenafifasilkebede/dataset

[82] J. Cheng, ''Brain tumor dataset,'' figshare, Dataset, 2017, doi: 10.6084/m9.figshare.1512427.v5.

[83] K. Mridha, Md. M. Uddin, J. Shin, S. Khadka, and M. F. Mridha, ''An interpretable skin cancer classification using optimized convolutional neural network for a smart healthcare system,'' *IEEE Access*, vol. 11, pp. 41003–41018, 2023.

[84] Y. N. Tan, V. P. Tinh, P. D. Lam, N. H. Nam, and T. A. Khoa, ''A transfer learning approach to breast cancer classification in a federated learning framework,'' *IEEE Access*, vol. 11, pp. 27462–27476, 2023.

[85] B. ElKarami, A. Alkhateeb, H. Qattous, L. Alshomali, and B. Shahrrava, ''Multi-omics data integration model based on UMAP embedding and convolutional neural network,'' *Cancer Informat.*, vol. 21, pp. 1176–9351, Jan. 2022.

[86] L. Zhou, M. Rueda, and A. Alkhateeb, ''Classification of breast cancer Nottingham prognostic index using high-dimensional embedding and residual neural network,'' *Cancers*, vol. 14, no. 4, p. 934, Feb. 2022.