An approach for cancer-type classification using feature selection techniques with convolutional neural network

Murtada K. Elbashir [1*], mmmm[1], mmmm[1,4], mmmmm[5,6], mmmmmmm [1,2,3]

[1]College of Computer and Information Sciences, Jouf University, Sakaka 72441, Saudi Arabia.

[2]School of Mathematics, Statistics and Computer Science, University of KwaZulu-Natal, Pietermaritzburg, Private Bag X01, Scottsville 3209, South Africa.

* Corresponding Author, email@gmail.com

Abstract

Cancer diagnosis and treatment depend on accurate cancer-type prediction. A prediction model can be used to infer significant cancer features (genes). Gene expression is among the most frequently used features in the detection of cancer. Deep Learning (DL) architectures, which demonstrate cutting-edge performance in many disciplines, are not appropriate for the data of gene expression since it contains a few numbers of samples that have thousands of features. In this study, we present an approach that applies three feature selection techniques (Lasso, Random Forest, and Chi2) on gene expression data obtained from Pan-Cancer Atlas through the TCGA Firehose Data using R statistical software. We calculated the feature importance of each selection method and then calculated the mean of the feature importance to be used in determining the threshold for selecting the most relevant features. We constructed five models with a simple convolutional neural networks (CNNs) architecture, which are trained using the selected features and then selected the winning model. The winning model achieved a precision of 94.11, a recall of 94.26, an f1-score of 94.14, and an accuracy of 96.16 on a test set.

Introduction

Cancer has been one of the main causes of death worldwide. In the US, Cancer mortality reached 163.5 per 100,000 persons. Worldwide, 609,820 cancer-related deaths, and more than 1.9 million new cancer diagnoses are anticipated for the year 2023[1]. Furthermore, according to data from 2013 to 2015, 38.4% of Americans will receive a cancer diagnosis at some point in their lifespan. Cancer detection and treatment methods have been the subject of extensive research to decrease its negative effects on human health. Cancer prediction places a lot of emphasis on cancer

susceptibility, recurrence, and prognosis. A shift toward multi-omics investigations is currently occurring [2, 3], with a strong focus on genomes, transcriptomics, and proteomics. The goal is to give clinicians a more profound understanding of patients' internal states to make accurate clinical decisions. A comprehensive understanding of the intricacies of the patterns involved in the cancer process is being provided by recent improvements made through collaborations between machine learning and gene expression data analysis of cancer [4]. Therefore, gene expression data raises the necessity for cutting-edge machine learning techniques, which increasingly serve as one of the primary motivators for numerous clinical and translational applications.

Huge volumes of cancer data have been created by recently implemented technologies and facilities and have been shared with the cancer research community. Traditional machine-learning techniques have been created over the last ten years due to the availability of publicly accessible cancer data [5-10]. On the other hand, a set neural network models with muli-layers called deep learning (DL) excels at the challenge of being trained with large amounts of data. Like traditional machine learning techniques, DL entails two steps: training, which involves estimating the parameters of the network from a specified dataset known as the training set, and testing, which makes use of testing set to evaluate the learned network performance. The development of deep learning approaches that have innovative interpretability and high accuracy in predicting the types of cancers was made possible by accumulating whole transcriptome profiling of tumor data. One of these profiling data is the Cancer Genome Atlas (TCGA), a well-known database for cancer transcriptome profiling, which contains the 33 most common types of cancer [11]. Many models that are based on DL have been created for the detection and classification of cancer. A research that utilized multiple models based on convolutional neural networks (CNNs) built for various input data types was reported by Milad Mostavi et al. in their publication [12]. The ability of the convolution kernels is rigorously examined by these models. Milad Mostavi et al. assessed the performance of their models in predicting tumor types using the TCGA data, which contains the gene expression of 33 types cancer. Their models achieved prediction accuracies ranging from 93.9% to 95%. Four Graph CNNs models were suggested and trained by Ricardo Ramirez et al. [13] utilizing the whole set of TCGA gene expression data sets to classify 33 different cancer types. Their models had prediction accuracies of (89.9-94.7%). Lyu et al. [14] developed a CNNs model and obtained classification accuracy of more than95% for 33 cancer types retrieved from TCGA . They mapped the gene expression samples into two-dimensional matrices to be used as input.

Zexian Zeng et al. [15] presented a CNNs approach for the classification of seven types of cancer retrieved from TCGA dataset and obtained an overall accuracy of 77.6%. Our group [16] proposed five 1D-CNN based stacking ensemble approach for classifying the most forms of malignancies that affects women. In the developed model, RNASeq data obtained from TCGA is used as an input. The output of these models is integrated using Neural Network (NN), which is then utilized as a meta-model.

Ramroach et al. [17] assessed the application of five different machine learning (random forest, GBM, REFRN, SVM, and KNN) with RNAseq data from 17 different cancer types. They partitioned the data into 75% training set and 25% testing set. Thereafter, they used the full number of feature (genes) on the training set to build their models and the validation of the models' performance was based on the testing set. Their obtained results depicted that the ensemble algorithms performed better the other methods on the entire genes' list, while the clustering and classification models achieved higher performance when features (genes) reduced to 20 genes. Hong et al. [18] created a multitask model based on deep learning for classifying tissue, disease condition, tissue origin, and neoplastic subtype using the full transcriptome (RNA-seq) datasets of peri-neoplastic, neoplastic, and non-neoplastic tissue. Their results indicated that the model achieved 99% accuracy for classifying disease state, an accuracy of 97% for classifying tissue origin, and an accuracy of 92% for subclassification of neoplastic. Osseni et al [19] proposed multi-omics transformer (MOT) using the transformer architecture that integrate different omics data. They used five different omics data types including transcriptomics, epigenomics, copy number variations (CNV), and proteomics. Their model scored F1-score equal to 98.37% and 96.74% on a test set with and without missing omics representation, respectively in classifying the 33 tumor types. Moreover, their model detects the omics types that are required for classifying each phenotype. Khan and Lee [20] proposed gene transformer deep learning based model to detect the significant biomarkers across different cancer subtypes. They used gene expression data of 33 tumor types from the TCGA. Their results indicated that their proposed model outperformed the traditional classification models. Zhang et al. [21] developed an explainable deep learning model called Transformer for Gene Expression Modeling (T-GEM) for predicting the cancer types and identifying the type of immune cell using TCGA and ScRNA-Seq data respectively. Moreover, they used their proposed model to obtain the relevant markers. Their results depicted that their developed model has accuracies of 94.92% and 90.73% for the TCGA and PBMC ScRNA-Seq

datasets, respectively. lkf Cai et al. [22] developed a transformer model based on deep learning called DeePathNet that combines omics data and the pathways information. They used the datasets TCGA, CCLE, and ProCan-DepMapSanger. The performance of their proposed model was assessed on classifying cancer types and subtypes, in addition to prediction of the drug response. Their proposed model outperformed the traditional classifiers by achieving over 95% of recall measure for most of the cancer types.

In this work, we constructed a CNNs model that classifies 33 cancer types in addition to normal samples using RNA-Seq gene expressions data as inputs. The Illumina HiSeq platform and R software are used to obtain the data of gene expression from Pan-Cancer Atlas [23] via the RTCGAToolbox package [24, 25]. We selected the stddata run data of 20160128, which is determined using the getFirehoseRunningDates function. Consequently, the getFirehoseData function is used to download the gene expression data. Then, we processed the downloaded data using a normalization technique to the data to ensure that the expression can be inferred properly from the gene expression data and prevent the occurrence of biased expression measures. The normalized data is processed using filtration through the genefilter package to filter the genes that exhibit low variation across the samples.

Material and Methods

Dataset

The R/Bioconductor package RTCGAToolbox package [25] is used to retrieve the Pan-Cancer Atlas RNASeq gene expression data via the TCGA Firehose Data. The obtained data contains 10456 samples from 33 tumor types with their corresponding normal samples and it has 20501 genes in total. The gene expression data was log2 transformed using the formula $log_2(value + 1)$. Thereafter, dataset undergoes normalization and filtration processes which reduced the number of genes to 15271. Figure **1** presents the number of sample in each cancer type.

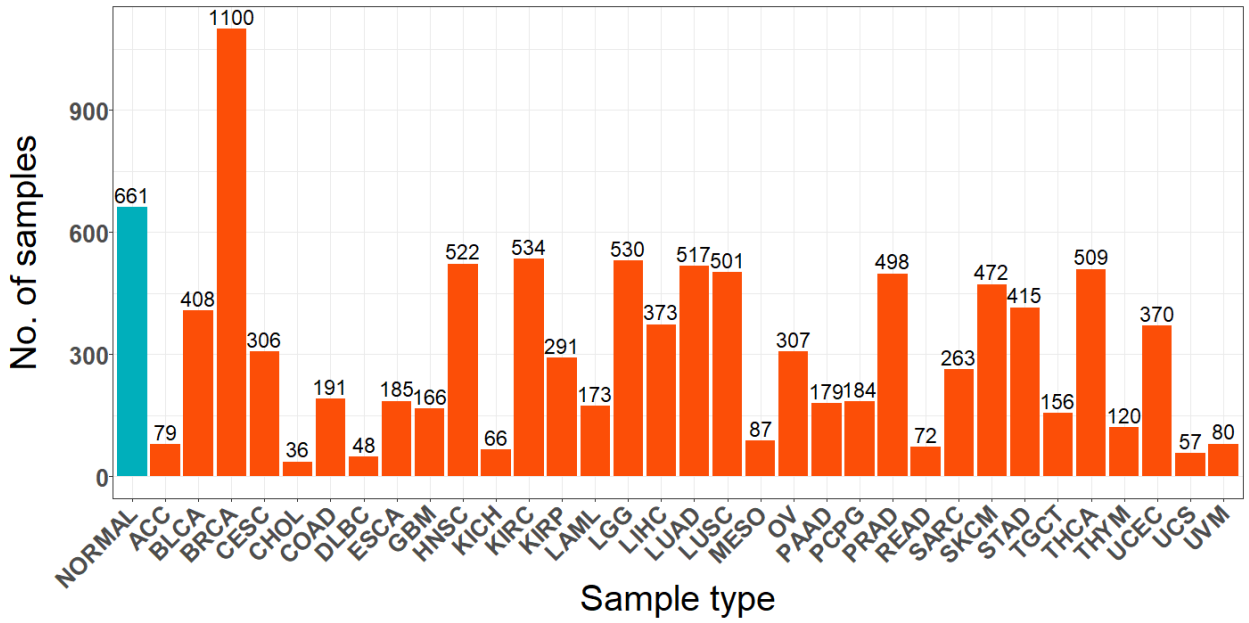

Figure **1**. the number of samples in each cancer type.

The Proposed Approach

Figure **2** depicts the complete framework of our proposed method. First, we split the entire data into training and testing sets before processing it with feature selection algorithms to prevent data leakage and model overfitting. Then, we applied feature selection procedures to the training set. This way, we will ensure that no information is shared between the training and testing sets when applying the features selection algorithm. Suppose feature selection is used to prepare the data, followed by model selection and training on the chosen features. In this case, the model will be given the training set as a whole for making feature selection decisions. This could lead to models that are improved by the chosen features over other models being tested to appear to have better results when they actually have biased results [26]. Three feature selection techniques are used. These techniques are Lasso, Random Forest, and Chi2. These feature selection techniques are essential because they can identify the most important features that strongly impact the target variable and remove the less important features. We calculated the feature importance of each selection method and then calculated the mean of the feature importance to be used in determining the threshold for selecting the most relevant features, which are then reshaped into 2D-image-like data. The thresholds that we used in this research are μ, 0.5 μ, 2 μ, 4 μ, and 8 μ and they are used

to create five classification models. These classification models are trained based on a 10-fold cross-validation approach.

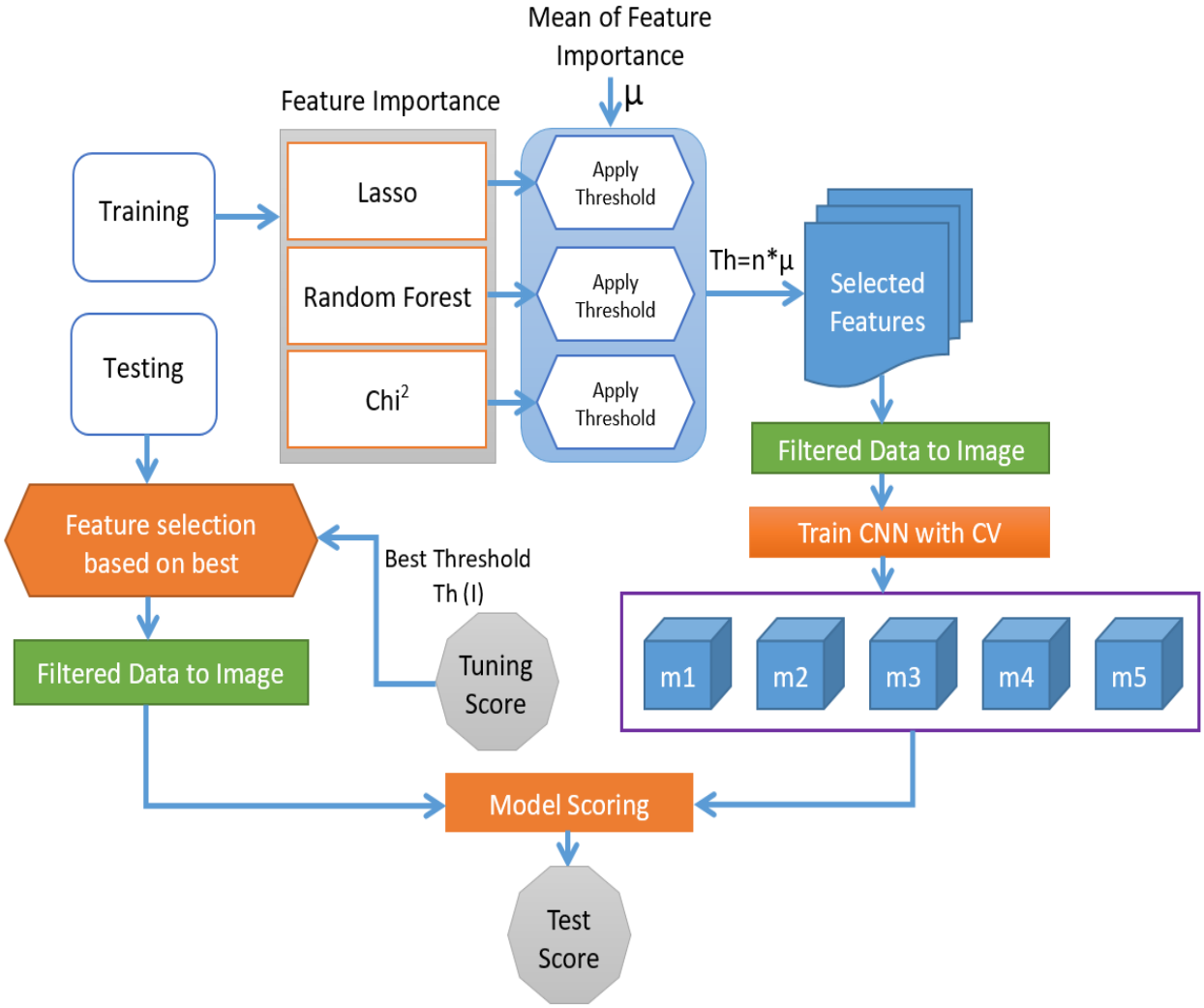

Figure 2. Overall framework of the proposed method.

Lasso is a regularization method that reduces the coefficients of less significant features to zero, which helps simplify the model. Features with non-zero coefficients are regarded as significant. Both classification and regression issues can be solved with Lasso. When there are many features, and the target variable is only affected by a small number of features, Lasso performs well. A penalty factor determines the number of features that are maintained. Choosing the penalty factor using cross-validation increases the likelihood that the model will generalize well to new data sets.

If we consider a multinomial response with more than two levels ($K > 2$), we can suppose that $p_\ell(g_i) = \Pr(C = c_i|g_i)$, where $c_i \in \{1, 2, 3, \dots, K\}$ represents the probability of observing i^{th} response. The multinomial LASSO model's log-likelihood can be expressed as follows [27]

$$\max_{\{\beta_{0\ell}, \beta_\ell\}ss_1^K \in \mathbb{R}^{K(p+1)}} \left[\frac{1}{N}\sum_{i=1}^{N} \log p_{c_i}(g_i) - \lambda \sum_{\ell=1}^{K} P_\alpha(\beta_\ell)\right]. \quad (1)$$

The aforementioned log-likelihood can be optimized through a penalized approach.

The matrix Y represents the target variable where Y has a dimension of NxK, and $y_{i\ell} = I(c_i = \ell)$. Equation **1** can be represented in more detailed as follow

$$\ell(\{\beta_{0\ell}, \beta_\ell\}_1^K) = \frac{1}{N}\sum_{i=1}^{N} \left[\sum_{\ell=1}^{K} y_{i\ell}(\beta_{0\ell} + g_i^T\beta_\ell) - \log\left(\sum_{\ell=1}^{K} e^{\beta_{0\ell} + g_i^T\beta_\ell}\right)\right]. \quad (2)$$

$$P_\alpha(\beta) = (1-\alpha)\frac{1}{2}\|\beta\|_{L_2}^2 + \alpha\|\beta\|_{L_1}, \quad (3)$$

$$= \sum_{j=1}^{p}\left[\frac{1}{2}(1-\alpha)\beta_j^2 + \alpha|\beta_j|\right]. \quad (4)$$

β_ℓ is a vector that represents the regression parameters, the penalty component of the equation above denoted by P_α, the expression level of the gene of sample i is represented by g_i, and the response value $y_{i\ell}$ for sample i. The penalty of LASSO regression can be achieved by setting $\alpha = 1$ in Equation **3**. LASSO is selected because it employs the total absolute values of the model parameters, which are restricted to be lower than a set threshold, as the penalty term.

In statistics, the chi-square test determines if two events are independent. Equation **5** shows the calculation of chi2 statistics where we can obtain the actual count O and the expected value E from a two variables data. Chi-Square determines the discrepancy between the anticipated count E and the actual count O. While choosing features, our goal is to select those that depend heavily on the outcome. If two features are independent then the observed count will be relatively close to the expected count, hence the value of the Chi-Square statistics will be smaller. Normally, a high Chi-Square statistic indicates that the independence hypothesis is not true. As a result, features with higher Chi-Square statistics will be selected for training the model.

$$\chi_c^2 = \sum \frac{(O_i + E_i)^2}{E_i} \quad (5)$$

Random Forests (RF) is a learning method based on an ensemble approach that builds many decision trees during training and returns each tree's mean prediction or mode of the classes. Problems involving classification and regression are both addressed by RandomForest. Based on

the impurity reduction they offer, RandomForest chooses the most crucial features. The most vital characteristics are those that offer the most significant impurity reduction. The RF algorithm procedures are presented in Algorithm **1** [26].

Algorithm 1. Random Forests Pseudocode

Training Phase:

for rf = 1 to C **do**

Draw B_{rf} a size N bootstrap sample from the training data D_{rf}.

Call GrowTree (B_{rf})

End for

GrowTree (B)
If B includes observations of only one class, **then**
return
Else
From the p genes in B, choose g possible splitting genes at random.

Using an impurity measure, choose the best G gene to divide on.

Create f child nodes of B, $B_1, \dots, B_f$, where G has f possible values $G_1, \dots, G_f$.

for rf = 1 to f **do**

Set the contents of B_{rf} to D_{rf}, where D_{rf} is all observations in B that match G_{rf}

Call GrowTree (B_{rf})

End for

End if

Prediction Phase:

To predict a new sample S: let $\hat{C}_{rf}(s)$ the probability assigned by classifier rf^{th} random forest tree. Therefore, the $\hat{C}_{RF}(s) = majority\ vote\ \{\hat{C}_{rf}(s)\}_{rf=1}^{C}$

Performance Measures

In this work, we used four metrics to assess our proposed model. These metrics, namely, F1-measure, precision, recall, and classification accuracy are commonly employed to evaluate the effectiveness of a model on bioinformatics data. The overall performance of the model is measure by F1-score and accuracy. In addition, the sensitivity and recognition rate are measured by recall and precision, respectively. The mathematical expressions for these metrics are provided below.

$$accuracy = \frac{\sum_i m_{ii}}{\sum_{i,j} m_{ij}} \quad (6)$$

$$recall_j = \frac{m_{jj}}{\sum_i m_{ji}} \quad (7)$$

$$precision_j = \frac{m_{jj}}{\sum_j m_{ji}} \quad (8)$$

$$F1\ measure_j = \frac{2 \times recall_j \times precision_j}{recall_j + precision_j} \quad (10)$$

i and j represent the different classes.

Results and Discussion

Keras Library [28] was used for the implementation of our deep-learning approach. The original dataset contains 15271 genes (features). As mentioned in the proposed method section, we initially split the entire dataset into training and testing sets. The training data was then used to create our model. We applied Lasso, Chi2, and Random Forest features selection to select the genes that significantly affect the class by calculating their feature importance. To select the most relevant features, we used six thresholds μ, 0.5 μ, 2 μ, 4 μ, and 8 μ on each feature selection method. We used the features that are selected by these five thresholds to create five classification models. The features are reshaped into 2D-image-like data before we pass them to the model. The CNNs of the models are designed to be very simple, and we limited the number of convolutional layers in their designs to one convolutional layer. That is because Increased CNN model depth does not necessary improve the performance on bioinformatics data [29], despite the fact that deeper models that are based on CNN have shown great performance in computer vision problems [30]. For issues like prediction of cancer types, where the number of samples is very small compared to the number of parameters, shallower models are recommended. Such simple models use fewer training resources and prevent overfitting [31]. Based on these two factors, we built a

CNN model that adheres to the most often used CNN applications in computer vision when the input has a 2-D format, such as an image. The 2D kernels in this CNN are used to extract local features. The grid search method [32] is used to tune the size, number, and stride of kernel parameters in addition to the number of nodes in the fully connected layers.

Since the classes in the training data are imbalanced, we set the class weight parameter to ‘'balanced'’ to automatically adjust the weights based on the class frequencies in the training data. We utilized the cross-validation method with a leave-one-out to assess the correctness of our model. Our training data set was initially divided into ten roughly equal sets. One of these sets was then eliminated to represent the validation set, and the other nine sets were pooled to form the training set. We went through this process ten times, taking out one set each time to stand in for the validation set. In this manner, we will have a distinct validation set each time, allowing us to assess the generalizability of our model. The number of features that resulted from the different feature selection methods (Lasso Random Forest, Chi2) using the six thresholds together with their corresponding model tuning accuracy is shown in Table **1**, Table **2**, and Table **3**.

Table 1. models accuracies and the number of features when using Lasso with the different thresholds.

Model No.	CV Tuning average Accuracy	Tuning std	Threshold (feature coefficient)	No. of Features
1	95.83%	(+/-1.28%)	No	15271
2	96.76%	(+/-0.41%)	½ μ	7535
3	97.03%	(+/-0.37%)	μ	4865
4	96.98%	(+/-0.43%)	2 μ	2356
5	**97.11%**	**(+/-0.42%)**	**4 μ**	**597**
6	95.18%	(+/-0.80%)	8 μ	81

Table 2. models accuracies and the number of features when using Random Forest with the different thresholds.

Model No.	CV Tuning average Accuracy	Tuning std	Threshold (feature coefficient)	No. of Features
1	95.83%	(+/-1.28%)	No	15271
2	96.83%	(+/-0.25%)	½ μ	5166
3	96.93%	(+/-0.08%)	μ	2896
4	**96.94%**	**(+/-0.28%)**	**2 μ**	**1656**
5	96.92%	(+/-0.14%)	4 μ	870
6	96.59%	(+/-0.15%)	8 μ	321

Table 3. models accuracies and the number of features when using Chi2 with the different thresholds.

Model No.	CV Tuning average Accuracy	Tuning std	Threshold (feature coefficient)	No. of Features
1	95.83%	(+/-1.28%)	No	15271
2	**96.69%**	**(+/-0.30%)**	**½ μ**	**7137**
3	96.52%	(+/-0.30%)	μ	4432
4	96.29%	(+/-0.41%)	2 μ	2220
5	95.06%	(+/-0.34%)	4 μ	688
6	85.59%	(+/-1.08%)	8 μ	64

As revealed in Table **1**, model 5 achieved a tuning accuracy of 97.11% and that makes it as the best model. The features for model 5 are selected using Lasso with feature importance threshold set to 4 μ. The selected threshold produces 597 features that will be used to measure the model accuracy on test data. The best model in Table **2 is** model 4 which achieved a tuning accuracy of 96.94%. The features for model4 are selected using Random Forest. 2 μ is used as feature importance threshold and that produces 1656 features that will be used to measure the model accuracy on test data. Table **3** shows that the best model when using Chi2 as a feature selection method is model 2 with a tuning accuracy of 96.69% at a feature importance threshold equal to 0.5 μ which produces 7137 features that will be used to measure the model accuracy on the test data. From the above results, it is clear that the best model is the model that resulted from using the Lasso as a features selection method at a feature importance threshold equal to 4 μ. The accuracy obtained from evaluating the best model on the test set is 96.16% with test std: (+/-0.40%). The classification report of the best model for each cancer type and the normal cases is depicted in Table **4**. Table **4** shows that the f1-score provides the harmonic mean of precision and recall.

In comparison to all other classes, the scores assigned to each class will indicate how accurately the classifier classified the data points inside that class. Table **5** shows that the proposed model has very high identification ability on the classes BRCA, CESC, HNSC, LGG, PCPG, PRAD, SKCM, TGCT, THCA, and UCEC with f1-score ranging between 0.98 and 0.99. Also, the table shows that the proposed model performed very weak in classifying CHOL class**,** where the number of samples of the real response that belong to it is 35.

Table 4. presents the classification report for each cancer type in addition to the normal cases.

Classes	Precision	Recall	F1-score
Normal	0.91	0.93	0.92
ACC	1	0.93	0.96
BLCA	0.94	0.96	0.95
BRCA	0.99	0.98	0.99
CESC	0.99	0.97	0.98
CHOL	0.62	0.57	0.6
COAD	0.86	0.88	0.87
DLBC	1	1	1
ESCA	0.89	0.85	0.87
GBM	1	1	1
HNSC	0.99	0.99	0.99
KICH	0.77	0.88	0.82
KIRC	0.96	0.97	0.97
KIRP	0.98	0.91	0.94
LAML	1	1	1
LGG	0.99	0.99	0.99
LIHC	0.97	0.94	0.95
LUAD	0.94	0.93	0.94
LUSC	0.94	0.95	0.94
MESO	0.93	1	0.97
OV	1	1	1
PAAD	0.93	0.97	0.95
PCPG	1	0.97	0.99
PRAD	0.98	0.99	0.98
READ	0.65	0.61	0.63
SARC	0.95	0.98	0.96
SKCM	0.98	0.99	0.99
STAD	0.94	0.96	0.95
TGCT	1	0.97	0.98
THCA	1	0.99	0.99
THYM	1	1	1
UCEC	0.99	0.99	0.99
UCS	0.91	1	0.95
UVM	1	1	1

Figure **3** shows the proposed model Confusion matrix on 33 tumor types in addition to the normal samples. By examining the confusion matrix carefully, it is clear that the majority of errors are in the classification of Read, KICH, ESCA, and CHOL. For the READ cancer type, 39% of the samples were misclassified as COAD (colon adenocarcinoma), while 12% of the samples of COAD were misclassified as READ. This misclassification between READ and COAD is

observed in the study [13]. Similarly, 29% and 14% of cholangiocarcinoma (CHOL), a type of liver cancer that forms in the bile duct, are misclassified into hepatocellular carcinoma (LIHC) and Lung adenocarcinoma (LUAD), respectively. Figure 2 also shows that the proposed model is able to classify the eight classes (UVM, UCS, THYM, OV, MESO, LAML, GBM, DLBC) into their corresponding.

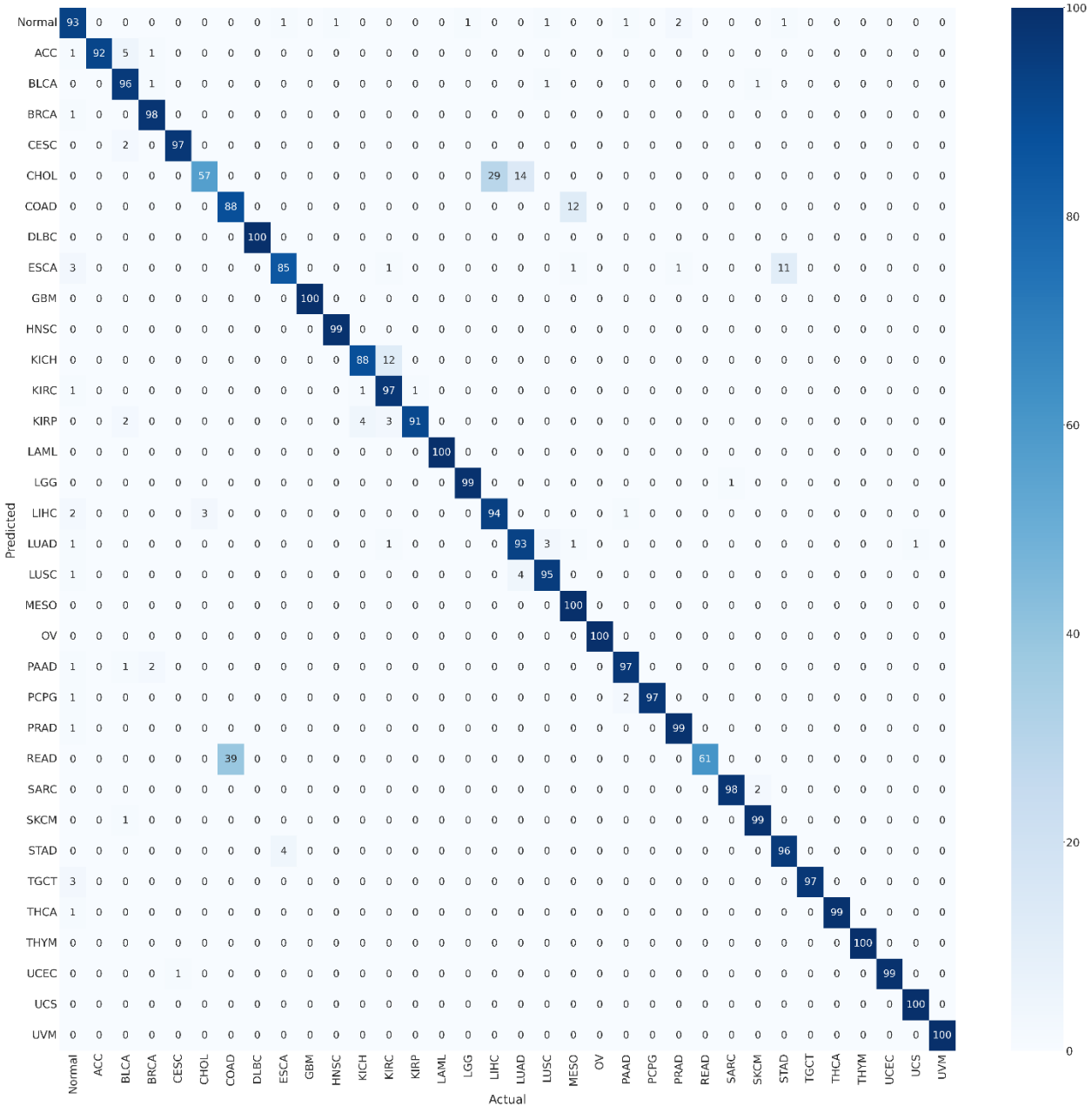

Figure 3. The proposed model Confusion matrix on 33 tumor type in addition to the normal samples.

Table 5 shows the comparison between our proposed method, Mostavi et al. [12], and Ramirez, Ricardo [13]. Mostavi et al. and Ramirez, Ricardo used the same 33 cancer type's RNA-Seq gene expression data that we used which are downloaded from The Cancer Genome Atlas (TCGA) database. Our proposed method achieved an accuracy of 96.16% with test std: (+/-0.40%), precision = 94.11%, recall = 94.26%, and F-measure = 94.14.

Table 5. The comparison of our approach with other cancer types classification methods.

Classification method	Accuracy	Precision	Recall	F1-score
Our Proposed approach (33 cancer types + Normal)	96.16	94.11	94.26	94.14
Mostavi et al. [12]	95.0	92.5		
PPI + singleton GCNN model [13]	94.61			

Conclusion

In this paper, we developed a new deep-learning approach to identify cancer types based on RNA-Seq gene expression data. We employed three feature selection methods to select the best features that can be used for cancer types identification. For each feature selection method, we calculated the feature's importance that rate input features according to how well they are able to predict a given target variable. Based on the importance of the features, we devised different thresholds for extracting the best features and then trained five CNN models based on a ten-fold cross-validation approach. We selected the best model for each feature selection method and then selected the winning model based on the validation accuracy. The winning model shows an accuracy of 96.16, Precision of 94.11, recall of 94.26, and f1-score of 94.14 on a test set.

Authors' contributions

Data curation, M.A.; formal analysis, A.M. and M.E.; investigation, Ab.M, M.A and A.M.; supervision, M.A.; writing—original draft, Ad.M. and A.M.; writing—review and editing, A.M., M.E., M.A. All authors have read and agreed to the published version of the manuscript.

Funding

This work was funded by the Deanship of Scientific Research at Jouf University under Grant No. (DSR-2022-RG-0104).

Competing interests

The authors declare that they have no conflicts of interest to report regarding the present study.

Data availability

The datasets are publicly available on The Cancer Genome Atlas (TCGA) repository.

References

1. Siegel, R.L., et al., *Cancer statistics, 2022.* CA: a cancer journal for clinicians, 2023. **73**(2):.
2. Bersanelli, M., et al., *Methods for the integration of multi-omics data: mathematical aspects.* BMC bioinformatics, 2016. **17**(2): p. 167-177.
3. Kim, M. and I. Tagkopoulos, *Data integration and predictive modeling methods for multi-omics datasets.* Molecular omics, 2018. **14**(1): p. 8-25.
4. de Anda-Jáuregui, G. and E. Hernández-Lemus, *Computational oncology in the multi-omics era: state of the art.* Frontiers in oncology, 2020. **10**: p. 423.
5. Kourou, K., et al., *Machine learning applications in cancer prognosis and prediction.* Computational and structural biotechnology journal, 2015. **13**: p. 8-17.
6. Statnikov, A., L. Wang, and C.F. Aliferis, *A comprehensive comparison of random forests and support vector machines for microarray-based cancer classification.* BMC bioinformatics, 2008. **9**(1): p. 1-10.
7. Cruz, J.A. and D.S. Wishart, *Applications of machine learning in cancer prediction and prognosis.* Cancer informatics, 2006. **2**: p. 117693510600200030.
8. Liu, J.J., et al., *Multiclass cancer classification and biomarker discovery using GA-based algorithms.* Bioinformatics, 2005. **21**(11): p. 2691-2697.
9. Li, Y., et al., *A comprehensive genomic pan-cancer classification using The Cancer Genome Atlas gene expression data.* BMC genomics, 2017. **18**: p. 1-13.
10. Holzinger, A., et al. *Current advances, trends and challenges of machine learning and knowledge extraction: from machine learning to explainable AI*. in *Machine Learning and Knowledge Extraction: Second IFIP TC 5, TC 8/WG 8.4, 8.9, TC 12/WG 12.9 International Cross-Domain Conference, CD-MAKE 2018, Hamburg, Germany, August 27–30, 2018, Proceedings 2*. 2018. Springer.
11. Grossman, R.L., et al., *Toward a shared vision for cancer genomic data.* New England Journal of Medicine, 2016. **375**(12): p. 1109-1112.
12. Mostavi, M., et al., *Convolutional neural network models for cancer type prediction based on gene expression.* BMC medical genomics, 2020. **13**: p. 1-13.

13. Ramirez, R., et al., *Classification of cancer types using graph convolutional neural networks.* Frontiers in physics, 2020. **8**: p. 203.
14. Lyu, B. and A. Haque. *Deep learning based tumor type classification using gene expression data*. in *Proceedings of the 2018 ACM international conference on bioinformatics, computational biology, and health informatics*. 2018.
15. Zeng, Z., et al., *Deep learning for cancer type classification and driver gene identification.* BMC bioinformatics, 2021. **22**(4): p. 1-13.
16. Mohammed, M., et al., *A stacking ensemble deep learning approach to cancer type classification based on TCGA data.* Scientific reports, 2021. **11**(1): p. 1-22.
17. Ramroach, S., M. John, and A. Joshi. *The efficacy of various machine learning models for multi-class classification of rna-seq expression data*. in *Intelligent Computing: Proceedings of the 2019 Computing Conference, Volume 1*. 2019. Springer.
18. Hong, J., L.D. Hachem, and M.G. Fehlings, *A deep learning model to classify neoplastic state and tissue origin from transcriptomic data.* Scientific Reports, 2022. **12**(1): p. 9669.
19. MA, O., et al., *MOT: a Multi-Omics Transformer for multiclass classification tumour types predictions.* 2022.
20. Khan, A. and B. Lee, *Gene transformer: Transformers for the gene expression-based classification of lung cancer subtypes.* arXiv preprint arXiv:2108.11833, 2021.
21. Zhang, T.-H., et al., *Transformer for Gene Expression Modeling (T-GEM): An Interpretable Deep Learning Model for Gene Expression-Based Phenotype Predictions.* Cancers, 2022. **14**(19): p. 4763.
22. Cai, Z., et al., *Transformer-based deep learning integrates multi-omic data with cancer pathways.* bioRxiv, 2022: p. 2022.10. 27.514141.
23. 53, D.C.C.B.R.J.M.A.K.A.P.T.P.D.W.Y. and T.S.S.L.D.A. 68, *The cancer genome atlas pan-cancer analysis project.* Nature genetics, 2013. **45**(10): p. 1113-1120.
24. Colaprico, A., et al., *TCGAbiolinks: an R/Bioconductor package for integrative analysis of TCGA data.* Nucleic acids research, 2016. **44**(8): p. e71-e71.
25. Samur, M.K., *RTCGAToolbox: a new tool for exporting TCGA Firehose data.* PloS one, 2014. **9**(9): p. e106397.
26. Hastie, T., et al., *The elements of statistical learning: data mining, inference, and prediction*. Vol. 2. 2009: Springer.
27. Friedman, J., T. Hastie, and R. Tibshirani, *Regularization paths for generalized linear models via coordinate descent.* Journal of statistical software, 2010. **33**(1): p. 1.
28. Keras, C.F., *GitHub.* 2022.
29. Min, S., B. Lee, and S. Yoon, *Deep learning in bioinformatics.* Briefings in bioinformatics, 2017. **18**(5): p. 851-869.
30. LeCun, Y., Y. Bengio, and G. Hinton, *Deep learning.* nature, 2015. **521**(7553): p. 436-444.
31. Ioffe, S. and C. Szegedy. *Batch normalization: Accelerating deep network training by reducing internal covariate shift*. in *International conference on machine learning*. 2015. pmlr.
32. Pedregosa, F., et al., *Scikit-learn: Machine learning in Python.* the Journal of machine Learning research, 2011. **12**: p. 2825-2830.