IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 22 January 2024, accepted 18 February 2024, date of publication 20 February 2024, date of current version 27 February 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3368070

RESEARCH ARTICLE

Applying the Deep Learning Techniques to Solve Classification Tasks Using Gene Expression Data

SERGII BABICHEV[1,2], IGOR LIAKH[3], AND IRINA KALININA[4]

[1]Department of Physics, Kherson State University, 73008 Kherson, Ukraine
[2]Department of Informatics, Jan Evangelista Purkyně University in Ústí nad Labem, 40096 Ústí nad Labem, Czech Republic
[3]Department of Information Science and Physics and Mathematics Disciplines, Uzhhorod National University, 88016 Uzhhorod, Ukraine
[4]Department of Intelligent Information Systems, Petro Mohyla Black Sea National University, 54000 Mykolaiv, Ukraine

Corresponding author: Sergii Babichev (sergii.babichev@ujep.cz)

This work was supported in part by the Faculty of Science, Jan Evangelista Purkyně University in Ústí nad Labem, Czech Republic.

ABSTRACT This manuscript explores the application of deep learning (DL) techniques for classifying gene expression data. A key aspect of our research is the comparative analysis of various DL neural network architectures, including Convolution Neural Networks (CNN), Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) Recurrent Neural Networks (RNN), as well as hybrid models that combine these networks. We applied the Bayesian optimization algorithm using 5-fold cross-validation for optimal hyperparameter tuning, which is crucial for DL algorithm performance. Significantly, we have advanced the methods for applying RNNs in processing gene expression data, particularly focusing on LSTM and GRU types. Our study introduces also a novel hybrid quality criterion for data classification, calculated as a weighted sum of partial quality criteria, incorporating an integrated F1-score derived through the Harrington desirability method. Furthermore, we investigate hybrid models that leverage various DL methods, enhancing decision-making objectivity in sample identification. This model uses a step-by-step information processing procedure, initially applying different DL models to gene expression data and subsequently processing these through a CART-based classifier for final decision-making. Our experiments, performed on gene expression data from patients with eight cancer types and one subset with normal samples (without cancer), demonstrated that GRU-RNN-based models, particularly a two-layer GRU-RNN, achieved the highest classification efficacy, with an accuracy of 97.8% on the test dataset. The performance of this model exceeded that of other models, whose accuracy varied between 96.6% and 97.3%. Comparative analysis with other studies in this field suggests that the proposed techniques demonstrate higher efficacy compared to similar research regarding the application of DL models for cancer-type diagnosis.

INDEX TERMS Convolution neural network, LSTM recurrent neural network, GRU recurrent neural network, gene expression data, classification, hybrid model, classification quality criteria, cancer disease.

I. INTRODUCTION

Modern bioinformatics is increasingly focused on processing gene expression data to develop diagnostic systems for complex diseases. The appeal of deep learning (DL) methods for this task stems from their ability to handle the intricate structure and vast volume of experimental data, which often comprises thousands of objects and over ten thousand attributes. DL algorithms stand out for their capacity to process intricate and unstructured data. These methods can find patterns in hierarchically presented data and craft functions that enable precise identification of the objects being studied. Not only do models based on DL offer high accuracy and performance, but they also excel at extracting meaningful patterns directly from raw data. This ability unveils hidden regularities and intricate data relationships that traditional methods might miss.

Another notable feature of deep learning-based models is scalability. They can be efficiently adapted to manage vast data volumes and benefit from parallel or distributed

The associate editor coordinating the review of this manuscript and approving it for publication was Marco Giannelli.

computing architectures. This scalability speeds up both the learning process and result generation. Specifically, for gene expression data, applying DL methods correctly can enhance the efficiency of diagnostic systems for complex diseases such as cancer, Alzheimer's disease, Parkinson's disease, etc. This is due to the superior accuracy in identifying subjects and the parallel processing capability. This fact bolsters increasing the object state determination's objectivity.

In summary, the growing importance and potential benefits of using DL for gene expression data highlight the significance and timeliness of current research in this domain.

Currently, several DL methods exist that can be applied to gene expression data processing, revealing hidden patterns and facilitating predictions about the state of the corresponding object [1]. Figure 1 presents a block chart of the most prevalent DL methods tailored for gene expression data processing and genomic sequence analysis, along with areas of their potential applications.

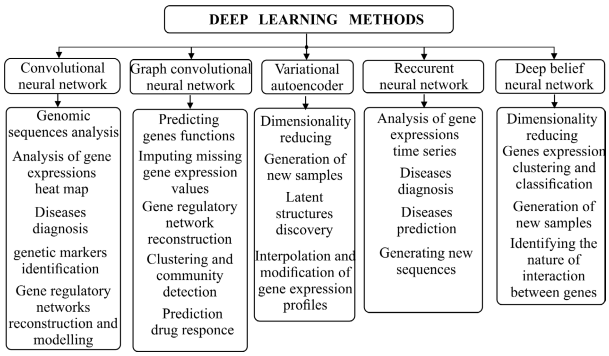


FIGURE 1. Block chart of DL methods and their applications for gene expression data and genomic sequence analysis.

Each DL method is suited for different tasks, with the choice of method being influenced by the nature of the experimental data, research objectives, and constraints. The following factors determine the primary distinctions between existing DL methods:

1. *Network Architecture*:
   - Convolutional Neural Networks (CNNs) is adept at handling both two-dimensional (e.g., images) and one-dimensional data.
   - Recurrent neural networks (RNNs) excel at processing sequential data like text or time series.
   - Variational autoencoders primarily generate new samples from learned latent representations.
   - Graph Convolutional Neural Networks (GCNN) process data represented as graphs. This necessitates prior gene network reconstruction, adding complexity to data processing.

   While each method offers unique advantages and drawbacks, the combination (or hybridization) of these DL models can potentially enhance gene expression data processing.
2. *Input Data Type and Size*:
   - CNNs typically need many input samples for optimal accuracy. However, increasing training epochs can lead to the risk of retraining.
   - RNNs, in contrast to CNNs, can operate effectively on smaller datasets.
   - GNNs necessitate input data in graph format, demanding further research into optimizing the graph structure.
3. *Application Tasks*:
   - Each of the above DL methods can be applied to different tasks, such as classification, clustering, generation sequences, reconstruction, etc.
   - The choice of method is determined by the specific task of gene expression data analysis, such as identification of biomarkers, prediction of health status or identification of the type of disease, detection of the nature of gene interaction, and reconstruction of gene regulatory networks.

Our research aims to improve the efficiency of cancer diagnosis systems using gene expression data by focusing on object classification via gene expression profiles, evaluating various CNN and RNN architectures, including hybrid models, optimizing their structures and hyperparameters through Bayesian optimization and k-fold cross-validation, proposing a hybrid quality criterion and an integrated F1-score for data classification, and investigating hybrid deep learning ensembles for enhanced decision-making accuracy.

The main contributions of this research are:

- The methods of applying RNN for gene expression data processing were further developed in our study. We explored two types of RNN: Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU). A particular enhancement here is our proposed algorithm for optimizing the architecture and hyperparameters of the RNN. This enhancement includes a comparative analysis of optimization methods, such as ordered grid search and the Bayesian optimization algorithm, providing a more systematic and efficient approach to RNN optimization.
- We proposed a hybrid quality criterion for data classification, which is computed as a weighted sum of partial quality criteria assessed during the simulation process. Another significant enhancement is our proposed integral criterion of the F1 score, employing the Harrington desirability method. This method is instrumental in determining the partial values of the F1 score for individual classes, thus offering a more nuanced and detailed assessment of model performance.
- Our research introduces a novel hybrid model that combines various machine learning methods, aimed at increasing the objectivity in identifying the samples under investigation. A significant methodological enhancement in our study is the presentation of this model as a block diagram illustrating a step-by-step information processing procedure. Initially, various

deep learning models are applied to a set of gene expression data, leading to the formation of intermediate solutions. These solutions are then structured as a data table for further processing by a classifier at the second hierarchical level. As a particular addition to our model, a decision tree algorithm (Classification and Regression Trees - CART) is employed as a classifier in the final step, which aids in forming the definitive decision about the state of the object.

The manuscript is structured as follows:

- Section II reviews current research in this domain, highlighting unresolved aspects of the overarching problem.
- Section III presents the theoretical parts of the research, including the flowchart of the stepwise procedure for processing gene expression data, steps of the experimental dataset formation, brief information about the Bayesian optimization algorithm with 5-fold cross-validation application, used DL-based models and quality criteria used for the models' effectiveness evaluation.
- Section IV contains the experimental parts of the research, simulation, obtained results, and discussion.
- Section V contains the conclusions of the research.

II. RELATED WORKS

The burgeoning role of machine learning (ML), especially DL, in analysing and interpreting experimental data in diverse bioinformatics domains is highlighted in numerous scientific studies and reviews. The review [2] extensively examines DL methods used for detecting DNA/RNA motifs and identifying transcription factor binding sites, which are key in human gene regulation. This study discusses 33 unique DL models designed for DNA/RNA motif detection, focusing on their distinct design approaches and implementation styles. The authors also propose methodologies to assess the efficiency of these DL models, considering aspects like model size, automatic calibration, tool selection, and training datasets.

In [3], the authors explore the application of DL for analyzing tumour heterogeneity through single-cell and spatial transcriptomic sequencing data. The research underscores how deep learning facilitates high-resolution insights, crucial for advancing precision oncology, including early cancer detection, diagnosis, patient survival rate assessment, and cancer treatment planning. Similarly, [4] addresses the diagnostic challenges of lung cancer using single histological slides. Employing recent advancements in digital pathology, this study illustrates the potential of DL in classifying lung cancer subtypes, predicting outcomes, deciphering mutational patterns, and estimating expression from histological and cytological images.

The [5] focuses on the prevalence of oral cancer and the transformative impact of artificial intelligence (AI) for its early detection and treatment. This comprehensive review, adhering to the PRISMA-ScR guidelines, compares various machine and deep learning models in identifying early-stage oral cancer lesions, highlighting the potential benefits and limitations of AI in oncological research.

Overall, these studies emphasize the vital role of machine learning and deep learning techniques in processing and analyzing gene expression data. The following subsections will delve deeper into recent advancements and applications of CNNs, RNNs, and Bayesian optimization algorithms in this field.

A. CONVOLUTIONAL NEURAL NETWORK

Much research is dedicated to using CNNs for gene expression data processing. For instance, [6] explores using CNNs for processing microarray gene expression data from the Lung Harvard 2 Dataset (LH2) for medical diagnosis. The study adopts a two-tiered approach: initially applying the Short-Term Fourier Transform (STFT) for feature extraction and then using Particle Swarm Optimization (PSO) and Harmonic Search (HS) for feature selection is carried out. This is followed by employing various classifiers, including Support Vector Machine (SVM) and CNN. Notably, the combination of STFT, PSO-selected features, and SVM (RBF kernel) classifier achieved the highest accuracy, reaching 94.47%.

In [7], researchers investigate different CNN types and architectures for predicting cancer types using gene expression profiles. They introduce three models: 1D-CNN, 2D-Vanilla-CNN, and 2D-Hybrid-CNN, all trained on a dataset from The Cancer Genome Atlas (TCGA), encompassing 10,340 samples across 33 cancer types. These models demonstrated high accuracy rates (93.9-95.0%), with the 1D-CNN model identifying an average of 108 cancer markers per class, including known markers for specific cancers. Their methodology, effectively minimizing the influence of tissue-of-origin, shows great promise for future cancer diagnostics.

In [8], the authors detail using CNNs to predict and classify various cancer types using 2D images derived from gene expression profiles and Protein-Protein Interaction (PPI) networks. Applied to a dataset of 6,136 human samples across 11 cancer types, this approach achieved an accuracy of 97.4% for distinguishing between normal and tumor samples and 95.4% for classifying the 11 cancer types. This innovative technique of generating cancer networks for CNN applications could significantly enhance cancer diagnosis and biomarker identification.

The study [9] examines using GCNN to classify tumor and non-tumor samples into 33 cancer types or as normal, based on unstructured gene expressions. Using TCGA dataset, the GCNN-based models reached prediction accuracies up to 94.7% across 34 classes, identifying 428 specific marker genes. This indicates that GCNN models are highly effective in cancer classification, leveraging cancer-specific marker genes.

However, these studies, along with others, often depend on graph search algorithms for optimal hyperparameter selection, which can be time and resource-intensive. Also,

when handling multi-class problems, classification accuracies varied between 94% and 95%. Our previous research ([10], [11]) involved exploring various CNN architectures for both binary and multi-class problems using gene expression data from different cancer types. We achieved around 97% accuracy for multi-class tasks by de-parallelizing the data processing flow and employing a method of alternative voting for the final decision. This paper extends our investigation into applying CNNs for multi-class problems using gene expression data.

B. RECURRENT NEURAL NETWORK

As was noted hereinbefore, the RNN is widely recognized for its effectiveness in processing sequential data like text, time series, and speech. In [12], the use of LSTM-RNN architecture was explored for optimizing codon usage in protein sequences. The novel tool was developed to learn codon usage bias from a genomic dataset comprising over 7,000 Escherichia coli genes. By capturing the sequential context of codon usage, this method aligns synthetic gene codon selection more closely with the host genome than traditional methods, potentially enhancing recombinant protein expression more effectively.

Another study, [13], applies RNNs to modern investment challenges in the high-frequency trading era, focusing on the Markowitz model. The research presents an in-depth analysis of the model's convergence and portfolio optimization, showing that this new approach, when tested with Dow Jones Industrial Average data, yielded higher returns with reduced risks, outperforming the DJIA index.

The [14] investigates hybrid RNN models for protein secondary structure prediction, a key task in determining protein configurations from amino acid sequences. This study introduced a Hybrid Recurrent Neural Networks (HRNN) approach, incorporating GRU, LSTM, and their bidirectional variants, BGRU and BLSTM, within a two-dimensional RNN (2D-RNN) framework. By integrating protein sequence features with the Position-Specific Scoring Matrix (PSSM), the HRNN models demonstrated a notable enhancement in prediction accuracy, with BiGRU and BiLSTM techniques achieving up to 93% accuracy.

Despite these advancements, a gap remains in the application of RNNs for gene expression data processing, a topic less explored than CNNs. Notably, RNNs typically have fewer hyperparameters than CNNs, potentially simplifying the model-tuning process during hyperparameter optimization. These insights underscore the need for further research in this specific area of RNN application.

C. BAYESIAN OPTIMIZATION METHOD

The Bayesian Optimization Method (BOM) is an optimization technique based on Bayesian networks and probability commonly employed in hyperparameter tuning for machine learning models. In [15], the authors leveraged BOM to improve machine learning models in predicting Medial Tibial Stress Syndrome (MTSS) using 25 anatomic and anthropometric predictors. This study, involving 180 participants, validated various machine learning models, including Ensemble, SVM, and Naive Bayes. Notably, the Naive Bayes classifier achieved an accuracy of 88.89% and an AUC of 0.8571 in a non-resampling experiment, demonstrating the potential of these optimized models in clinical MTSS risk assessment.

In [16], the authors explored using BOM in a novel machine learning algorithm, combining a multilayer perceptron and random forest (MLP-RF), for forecasting daily lake surface water temperatures using air temperature data from eight Polish lakes. This model effectively predicted temperatures, which is vital for lake ecosystem studies and maintained impressive performance even over extended forecast horizons.

In [17], the authors employed BOM alongside a Vision Transformer architecture to develop a computer-aided diagnosis (CAD) system for lung nodule detection from CT scans. Validated with 888 CT images from the LUNA16 dataset, the system achieved a detection sensitivity of 98.39% and a CPM score of 0.909, highlighting the synergy of Bayesian Optimization with Vision Transformer in medical imaging.

In [18], the authors introduced an algorithm named adaptive successive halving automated hyperparameter optimization (ASH-HPO), integrating successive halving, Bayesian optimization, and progressive sampling. This method was used to tune hyperparameters for RNN models, specifically for transient simulations of high-speed channels. Tested on CNNs, LSTMs, and CNN-LSTM networks, ASH-HPO demonstrated its efficiency in applications like PCIe Gen 2 and 5 channels and a PAM4 differential channel. Compared to benchmark HPO methods like standalone Bayesian optimization, successive halving, and hyperband, ASH-HPO showed faster convergence in transient simulation issues.

These studies collectively underscore the superiority of Bayesian optimization methods in optimizing hyperparameters, particularly in comparison to techniques like grid search. In our research, we extend the exploration of BOM in DL models, focusing on gene expression data processing.

III. MATERIAL AND METHODS

Figure 2 depicts the flowchart of the stepwise procedure for processing gene expression data, based on the joint application of DL models and the Bayesian optimization algorithm, as implemented in the framework of our research. The implementation of this procedure involves the following steps:

1. Formation and pre-processing of gene expression data: the dataset should be organized as a data frame, with rows representing examined samples and columns representing genes.
2. Dataset splitting: The dataset is divided into training and testing subsets in a 0.7/0.3 ratio. The training subset is further split into training and validation subsets in a 0.8/0.2 ratio. These subsets are used to operate the

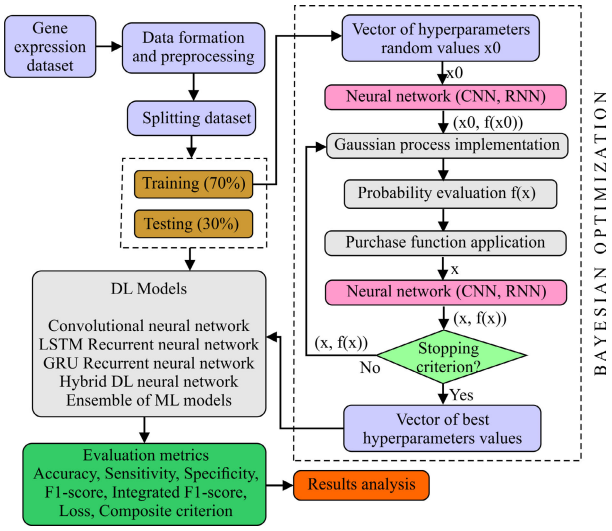


FIGURE 2. Flowchart of stepwise procedure for processing gene expression data, based on the joint application of DL models and the Bayesian optimization algorithm.

Bayesian optimization algorithm and train the Deep Learning (DL) model with optimal hyperparameters.

3. Applying the Bayesian optimization method: This approach is utilized with each DL model to ascertain the vector of optimal hyperparameters. Implementing 5-fold cross-validation at each epoch of the Bayesian algorithm operation is essential to this process.
4. Evaluation of model quality: Forming the classification quality criteria to assess the model's performance.
5. Training and testing of DL models: This step includes calculating the classification quality criteria for each model.
6. Analysis of the obtained results.

A. EXPERIMENTAL DATASET FORMATION AND PREPROCESSING

The simulation was performed using gene expression data from patients examined for various cancer types, freely accessible through The Cancer Genome Atlas (TCGA) [19]. The gene expression data, acquired via the Illumina platform [20] through RNA molecule genomic sequencing, initially encompassed 3269 samples and 19947 genes. Table 1 details the experimental data classification, categorizing both disease type and corresponding sample numbers and including counts of samples from healthy, non-cancerous patients. The gene expression value, in this case, reflects its activity level, which indicates the intensity of the protein synthesis process correlated with that gene type and is proportional to the quantity of akin genes.

In alignment with the methodology elucidated in [10] and [11], initially, the absolute gene counts were transformed into a more facilitative range for subsequent processing

TABLE 1. Classification of experimental gene expression data.

No	Type of cancer	Number of samples
1	Adrenocortical carcinoma - ACC	79
2	Glioblastoma multiforme - GBM	169
3	Sarcoma - SARC	263
4	Lung squamous cell carcinoma - LUSC	502
5	Lung adenocarcinoma - LUAD	541
6	Stomach adenocarcinoma - STAD	415
7	Kidney renal clear cell carcinoma - KIRC	542
8	Brain Lower Grade Glioma - LGG	534
9	Normal	224

(Count Per Million - CPM) utilizing the following formula:

$$CPM_{ij} = \frac{count_{ij}}{\sum_{j=1}^{m} count_{ij}} \cdot 10^6 \quad (1)$$

Here: $count_{ij}$ represents the number of the jth type of gene associated with the ith sample; m signifies the total count of distinct gene types investigated during the experiment.

The implementation of this step significantly reduced the variation range of the absolute values defining the expression (activity level) of the respective genes. In the second phase, data normalization was conducted by applying the function $log_2(CPM)$ to all values. In the third phase, non-expressed genes were removed according to the condition $log_2(CPM) \leq 0$ for all samples under investigation, reducing the gene count by 682 and shaping the gene expression experimental data matrix as: $E = (3269 \times 19265)$. In the final phase, negative gene expression values were replaced with zeros, corresponding to non-expressed genes for some samples, and to ensure accurate initialization of CNN filters, the number of gene expression profiles was increased to 19300 by supplementing with profiles having zero expression. As demonstrated in [21], CNNs possess a high level of resilience to the noise component, meaning that when using 19265 gene expression values as attributes, increasing their number by 35 (gene profiles with zero expression) will not affect sample identification results.

B. BAYESIAN OPTIMIZATION METHOD

Grid search, a traditional method for hyperparameter selection in DL-based neural networks, is known for its intensive computational and time demands. The ordered grid search offers some optimization to enhance efficiency, yet it falls short of being fully optimal. The selection of hyperparameters is crucial, as it significantly influences the model's performance, highlighting the importance of an effective optimization process. In this context, we explore using the Bayesian optimization algorithm to automate the determination of optimal hyperparameters.

Bayesian optimization operates on the principle of an informed search, utilizing previous findings to guide the optimization path [22], [23]. It involves two main elements:

- *Surrogate Model*: Typically a Gaussian process, this statistical model approximates the objective function. It accounts for non-linear dependencies and quantifies the uncertainty in predictions.

- *Acquisition Function*: This function guides the selection of new evaluation points within the surrogate model, striking a balance between exploring new areas ('exploration') and utilizing known effective points ('exploitation').

The optimization follows an iterative pattern: it evaluates new points, updates the surrogate model, and uses the acquisition function to pick the next point, continuing until a preset stopping criterion, like a maximum number of iterations, is met.

5-fold cross-validation is applied at each Bayesian optimization epoch to train the model correctly (without overfitting). This step involves dividing the data into five parts. The model is trained and evaluated five times, using a different fold as the validation set each time. This method enhances the accuracy and generalizability of the model by ensuring hyperparameters are fine-tuned based on varied subsets of data, thus improving the reliability of the optimization process.

C. DL-BASED MODELS

Within the framework of our research, we utilized the following DL-based models: 1D one-layer and two-layer CNNs, one-layer and two-layer LSTM RNNs, and GRU RNNs. In applying CNNs, the simulation process involved optimizing several hyperparameters: the number of filters in the convolutional layers, the kernel size, maximal pooling, and the kernel size of the dense layer (dense kernel). Considering the results of previous studies [10], [11], [21], the activation functions applied were the sigmoid function (sigmoid) for convolutional layers, the SELU (Scaled Exponential Linear Unit) function for the dense layer, and the softmax function for the output layer of neurons. The range of values for the relevant hyperparameters was as follows: $num_filters = [8, 64]$, $kernel_size = [3, 10]$, $max_pooling = [2, 4]$, and $dense_kernel = [16, 256]$. The initial number of points in the hyperparameter feature space was set at 10, and the number of subsequent iterations to search for the optimal hyperparameter combination was 50 when applying a one-layer CNN and 70 for a two-layer CNN. The Dropout rate, representing the proportion of neurons being zeroed at each step during the network training process, was set at 20%.

In the case of RNN model utilized (LSTM and GRU), the number of neurons in layers varied within the range from 20 to 100. We also have investigated sequential and parallel hybrid models based on the integrated application of CNN and RNN. In each case, to determine the optimal hyperparameters and control overfitting, we also applied the Bayesian optimization algorithm and k-fold cross-validation method. Figure 3 depicts the block diagram of a hybrid classification model for one-dimensional gene expression data based on the sequential application of two-layer convolutional and recurrent neural networks, where the recurrent network can be implemented using either the LSTM or GRU algorithm. The value of the hyperparameters can be changed during the simulation procedure implementation.

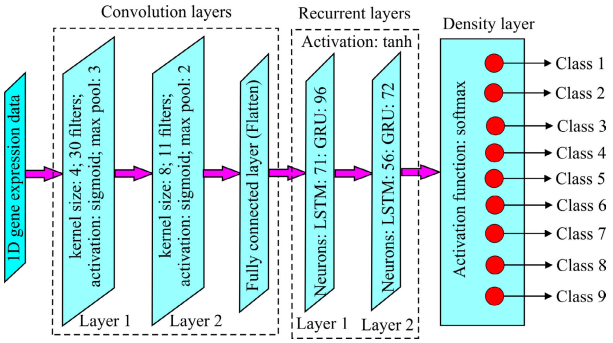


FIGURE 3. The block diagram of the hybrid model for classifying one-dimensional gene expression data, based on the sequential application of two-layer convolutional and recurrent neural networks.

The application of a CNN at the initial stage of model implementation is justified by its ability to detect complex dependencies between genes in the respective gene expression profile. CNNs can identify local dependencies in the gene expression profile, such as local structures, motifs, or patterns indicative of specific functions or pathological processes. Convolutional layers can provide translation invariance of the data. This means that CNNs can recognize the same dependencies in various positions of the gene expression data, regardless of their exact location. Furthermore, CNNs can automatically select useful features from gene expression data during training, which aids in enhancing the quality of forming a fully connected layer for its subsequent use as input data for the recurrent layer. Applying a pooling layer (maximal pooling) at the output of each convolutional layer helps reduce the dimensionality of the data while preserving important features.

Recurrent layers at the model's output allow it to consider the sequence of data, that is, the order of genes in vectors, which can significantly impact the results of gene expression data classification. Applying recurrent layers at the output of the convolutional layer also allows for reducing the number of model parameters compared to using recurrent layers on a fully sequential input, which can decrease the risk of model overfitting. The absence of overfitting was monitored in all cases through the convergence of the model classification accuracy character changes and the loss function value, calculated on the training and validation data during the model training process.

The second hybrid model explored in our study employs a parallel approach using various top-performing DL models for classifying gene expression data. This model makes intermediate decisions which are then aggregated to form the final decision. A key step in this process involves applying a classifier to these intermediate decisions. In this context, we utilized the CART (Classification and Regression Trees) machine learning method. CART is an algorithm for building decision trees, chosen for its ability to recursively split a dataset into subgroups. This splitting is based on the values of a specific feature (the most significant intermediate decision),

resulting in the construction of a decision tree. Each leaf of this tree corresponds to a distinct class, categorizing the objects within that subset of data. A notable advantage of the CART algorithm is its interpretability; the sections and conditions of the decision tree can be easily understood and explained.

The block diagram of the hybrid model for classifying gene expression data based on an ensemble of machine-learning methods is illustrated in Figure 4.

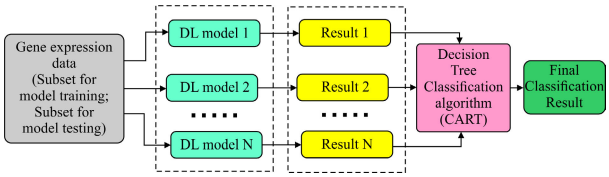


FIGURE 4. The block diagram of the hybrid model for classifying gene expression data, which is based on an ensemble of DL and ML methods.

The following DL models were investigated during the implementation of the simulation process:

- One- and two-layer CNNs.
- One- and two-layer LSTM and GRU RNNs.
- Hybrid CNN-LSTM model.
- Hybrid CNN-GRU model.
- Five ensembles of DL models:
  - Ensemble 1: Two-layer CNN, two-layer GRU RNN, CNN-LSTM, CNN-GRU.
  - Ensemble 2: Two-layer CNN, two-layer GRU RNN, CNN-GRU.
  - Ensemble 3: Two-layer CNN, two-layer GRU RNN, CNN-LSTM.
  - Ensemble 4: Two-layer GRU RNN, CNN-GRU.
  - Ensemble 5: Two-layer CNN, two-layer GRU RNN.

D. QUALITY CRITERIA

In the current research, the classification of objects was performed based on gene expression data, utilizing metrics based on the evaluation of Type I and Type II errors [24]:

- Classification Accuracy - represents the aggregate number of samples correctly identified:

$$ACC = \frac{TP + TN}{TP + FP + TN + FN} \tag{2}$$

- The F1-score is a measure that evaluates the accuracy of a classifier when the samples are distributed in separate classes, by considering both the precision (PR) and recall (RC) of the prediction. It is computed as the harmonic mean of PR and RC.

$$F1 = \frac{2 \cdot PR \cdot RC}{PR + RC} \tag{3}$$

Here, PR is the ratio of true positive predictions to the total positive predictions made, and RC is the ratio of true positive predictions to the actual positive instances in the dataset:

$$PR = \frac{TP}{TP + FP}; \quad RC = \frac{TP}{TP + FN} \tag{4}$$

The F1-score ranges from 0 to 1, where 1 indicates perfect precision and recall, and 0 indicates the worst. In the aforementioned formulas, TP (True Positive) and TN (True Negative) represent the number of objects correctly categorized into their respective classes, whereas FP (False Positive) and FN (False Negative) denote the number of objects inaccurately assigned.

It is noteworthy that in addressing a multiclass problem, criterion (2) evaluates the overall accuracy of sample distribution among classes, while criterion (3) assesses the accuracy of sample distribution within each class independently.

- Cross-entropy Loss function (L), calculated during the model validation procedure implementation. This criterion measures the dissimilarity between the predicted probability distribution and the actual distribution. In the context of a multi-class classification task with C classes:

$$L(y, \hat{y}) = -\frac{1}{N} \sum_{i=1}^{N} \sum_{j=1}^{C} y_{ij} log(\hat{y}_{ij}) \tag{5}$$

Here: y_{ij} is a binary indicator of whether class j is the correct classification for observation i; $\hat{y}_{ij}$ is the predicted probability that observation i is of class j; N is the total number of observations.

Considering that, when dealing with many classes, analyzing F1-score values corresponding to the classes to select the optimal alternative from a list of hyperparameters can be challenging, an integrated F1-score value was calculated. Implementation of this procedure is based on the values obtained in the previous step, applying Harrington's desirability method, which is one of the effective methods for solving multicriteria problems. The algorithm for implementing this procedure involves the following steps:

1) **Initialization**:
   - Present the $F1$ score values in a matrix format, where rows represent classes, and columns - the hyperparameter values being explored in this phase.
2) **Calculation of private desirabilities**:
   - Determine the minimum and maximum values of the $F1$-score during the relevant phase of the DL model operation (using the respective hyperparameter combination).
   - Transform the scale of $F1$-score values into a linear scale of the dimensionless parameter Y, considering the boundary values of the $F1$-score defined in the previous step (the value of parameter Y, according to the desirability method, varies from $Y_{min} = -2$ to $Y_{max} = 5$). Here, coefficients of the linear equation are calculated in the first step:

$$\begin{aligned} Y_{min} &= a + b \cdot F1_{min} \\ Y_{max} &= a + b \cdot F1_{max} \end{aligned} \tag{6}$$

In the second step, a direct transformation of $F1$-score values into Y values occurs:

$$Y = a + b \cdot F1 \tag{7}$$

- Calculate private desirabilities for each $F1$-score value:

$$d = \exp(-\exp(-Y)) \tag{8}$$

3) **Calculation of the integrated $F1$-score value**:
   - For each column of the matrix obtained in step 2, calculate the integrated $F1$-score value as the geometric mean of all private desirabilities:

$$F1_{int}^{j} = \left(\prod_{i=1}^{9} d_{ij}\right)^{\frac{1}{9}} \tag{9}$$

   where j represents the corresponding column of the matrix of private desirabilities.
4) **Analysis of the obtained results**: Create a diagram showing the dependency of the integrated $F1$-score values on the respective combination of the hyperparameter values. Select the optimal combination of the hyperparameter values that corresponds to the maximum of the integrated F1-score.

E. CALCULATION OF THE COMPOSITE CLASSIFICATION QUALITY CRITERION

It should be noted that in most cases, determining the optimal combination of neural network hyperparameters based on a combination of classification quality criteria is challenging. The values of the criteria can contradict each other. Furthermore, a small difference in values can, to some extent, complicate the process of selecting a list of optimal DL hyperparameters. In this case, it is advisable to calculate a composite quality criterion based on computed individual criteria such as classification accuracy of samples, loss function value, and integrated F1-score value, wherein higher values of accuracy and F1-score and a lower value of the loss function correspond to a higher-quality model. The calculation of the composite quality criterion was carried out using the weighted average method:

$$QC_{weighted} = \sum_{i=1}^{n} w_i QC_i \tag{10}$$

where: w_i is the weight of the corresponding i-th QC criterion (Quality Criterion).

The algorithm for calculating criterion (10) within the framework of the current research involves the following steps:

1) Inverting the loss function values into a vector of values that increase with the enhancement of the model's attractiveness:

$$loss_i' = max(loss) - loss_i \tag{11}$$

2) Normalization of all criterion values within the range [0, 1]:

$$QC_i^{norm} = \frac{QC_i - min(QC)}{max(QC) - min(QC)} \tag{12}$$

3) Initialization of the weight vector for the used criteria. When calculating the composite classification quality criterion, it was assumed that the weight of the loss function value, calculated on the data for model validation, is half as significant as the weights of the accuracy criterion and the integrated $F1$ score value, calculated on the test data subset. Therefore, the weight vector for the criteria vector $QC = (ACC, F1_int, loss')$ was initialized as follows: $w = (0.4, 0.4, 0.2)$.
4) Calculation of the composite criterion value using formula (10):

$$QC_i^{comp} = w[1] \cdot ACC_i^{norm} + w[2] \cdot F1_i^{norm} + w[3] \cdot loss_i^{norm} \tag{13}$$

A higher value of the criterion (13) corresponds to a better alternative.

IV. SIMULATION, RESULTS AND DISCUSSION

Figure 5 presents charts depicting the Accuracy and Loss metrics for both the training and validation datasets across epochs, specifically during the training of a one-layer CNN model. Similar charts were generated for other models. Analysis of these charts reveals no signs of overfitting. This is evidenced by the consistent changes in accuracy and loss values for both the training and validation datasets throughout the training and validation phases of the model.

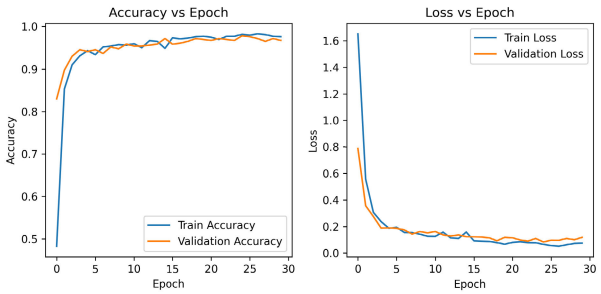


FIGURE 5. Charts depicting the accuracy and loss metrics for both the training and validation datasets across epochs, specifically during the training of a one-layer CNN model.

Table 2 and 3 displays the simulation results regarding applying the Bayesian optimization algorithm for one-layer and two-layer CNNs, LSTM and GRU RNNs to determine the optimal combination of hyperparameters.

Tables 4 and 5 show the classification results of test subset data samples (981) using one-layer (Table 4) and two-layer (Table 5) CNNs, the optimal hyperparameters of which were determined using the Bayesian optimization algorithm.

Tables 6 - 7 and Tables 8 - 9 present the simulation results regarding the application of LSTM and GRU RNN

TABLE 2. Simulation results regarding the application of the Bayesian optimization algorithm for determining the optimal combination of hyperparameters for one-layer and two-layer CNNs.

One-level CNN				
Accuracy	Number of filters	Kernel size	Max pooling	Dence kernel
0.972	44	5	3	48
Two-level CNN				
Accuracy	Number of filters 1	Kernel size 1	Max pooling 1	Dence kernel
0.966	53	8	4	38
	Number of filters 2	Kernel size 2	Max pooling 2	
	27	14	3	

TABLE 3. Simulation results regarding the application of the Bayesian optimization algorithm for determining the optimal combination of hyperparameters for one-layer and two-layer RNNs.

One-level LSTM RNN	Two-level LSTM RNN		One-level GRU RNN	Two-level GRU RNN	
Number of neurons	Number of neurons 1	Number of neurons 2	Number of neurons	Number of neurons 1	Number of neurons 2
43	75	42	74	84	67

TABLE 4. Simulation results regarding the application of a one-layer CNN for the classification of various types of cancer diseases.

Class	PR	RC	F1	F1-int	ACC	LOSS	Comp QC
ACC	0.815	1.000	0.898				
GBM	1.000	0.980	0.990				
KIRC	0.994	0.994	0.994				
LGG	0.971	1.000	0.985				
LUAD	0.951	0.917	0.934	0.351	0.966	0.122	0.176
LUSC	0.935	0.941	0.938				
SARC	0.983	0.881	0.929				
STAD	0.986	1.000	0.993				
NORM	0.979	1.000	0.989				

TABLE 5. Simulation results regarding the application of a two-layer CNN for the classification of various types of cancer diseases.

Class	PR	RC	F1	F1-int	ACC	LOSS	Comp QC
ACC	0.880	1.000	0.936				
GBM	0.981	1.000	0.990				
KIRC	0.994	0.994	0.994				
LGG	0.978	0.985	0.981				
LUAD	0.933	0.982	0.957	0.819	0.973	0.127	0.738
LUSC	0.986	0.922	0.953				
SARC	0.983	0.881	0.929				
STAD	1.000	1.000	1.000				
NORM	0.979	1.000	0.989				

TABLE 6. Simulation results regarding the application of a one-layer LSTM-RNN for the classification of various types of cancer diseases.

Class	PR	RC	F1	F1-int	ACC	LOSS	Comp QC
ACC	0.880	1.000	0.936				
GBM	0.981	1.000	0.990				
KIRC	0.994	0.983	0.989				
LGG	0.992	0.963	0.977				
LUAD	0.952	0.929	0.940	0.666	0.966	0.141	0.305
LUSC	0.947	0.941	0.944				
SARC	0.867	0.970	0.915				
STAD	0.986	1.000	0.993				
NORM	1.000	0.985	0.993				

with optimal number of neurons in the recurrent layers, respectively.

TABLE 7. Simulation results regarding the application of a two-layer LSTM-RNN for the classification of various types of cancer diseases.

Class	PR	RC	F1	F1-int	ACC	LOSS	Comp QC
ACC	0.917	1.000	0.957				
GBM	1.000	1.000	1.000				
KIRC	0.989	0.983	0.986				
LGG	0.978	0.993	0.985				
LUAD	0.926	0.959	0.942	0.872	0.969	0.139	0.574
LUSC	0.966	0.915	0.940				
SARC	0.955	0.940	0.947				
STAD	0.973	0.985	0.986				
NORM	1.000	0.978	0.993				

TABLE 8. Simulation results regarding the application of a one-layer GRU-RNN for the classification of various types of cancer diseases.

Class	PR	RC	F1	F1-int	ACC	LOSS	Comp QC
ACC	0.880	1.000	0.936				
GBM	1.000	1.000	1.000				
KIRC	0.994	0.983	0.989				
LGG	0.978	0.993	0.985				
LUAD	0.942	0.959	0.950	0.847	0.971	0.127	0.693
LUSC	0.953	0.928	0.940				
SARC	0.955	0.940	0.947				
STAD	0.986	1.000	0.993				
NORM	1.000	0.985	0.993				

TABLE 9. Simulation results regarding the application of a two-layer GRU-RNN for the classification of various types of cancer diseases.

Class	PR	RC	F1	F1-int	ACC	LOSS	Comp QC
ACC	1.000	1.000	1.000				
GBM	1.000	1.000	1.000				
KIRC	0.989	0.989	0.989				
LGG	0.985	0.985	0.985				
LUAD	0.942	0.959	0.950	0.876	0.978	0.152	0.800
LUSC	0.966	0.928	0.947				
SARC	0.926	0.940	0.933				
STAD	0.986	1.000	0.993				
NORM	1.000	1.000	1.000				

TABLE 10. Simulation results regarding the application of a hybrid model CNN-LSTM-RNN for the classification of various types of cancer diseases.

Class	PR	RC	F1	F1-int	ACC	LOSS	Comp QC
ACC	0.957	1.000	0.978				
GBM	1.000	1.000	0.981				
KIRC	0.994	0.989	0.991				
LGG	0.978	0.993	0.985				
LUAD	0.902	0.976	0.938	0.806	0.967	0.126	0.533
LUSC	0.979	0.895	0.935				
SARC	0.953	0.910	0.931				
STAD	0.973	0.985	0.989				
NORM	0.993	0.971	0.985				

The classification results of the gene expression data test subset, obtained by applying hybrid CNN-LSTM and CNN-GRU models, are presented in Tables 10 and 11 respectively.

Figure 6 depicts the results of a comparative analysis of all types of DL neural networks and their combinations used during the simulation process.

The analysis of the obtained results indicates that the classification accuracy of the samples is consistently high across all scenarios. Specifically, accuracy ranges from 96.6% with a single-layer CNN and LSTM RNN, to 97.8% when employing a two-layer GRU RNN. Notably, the two-layer GRU RNN showed superior performance in gene expression

TABLE 11. Simulation results regarding the application of a hybrid model CNN-GRU-RNN for the classification of various types of cancer diseases.

Class	PR	RC	F1	F1-int	ACC	LOSS	Comp QC
ACC	0.846	1.000	0.917				
GBM	1.000	1.000	1.000				
KIRC	0.994	0.983	0.989				
LGG	0.978	0.993	0.985				
LUAD	0.941	0.947	0.944	0.675	0.969	0.118	0.547
LUSC	0.947	0.935	0.941				
SARC	0.953	0.910	0.931				
STAD	0.986	1.000	0.993				
NORM	0.993	0.985	0.989				

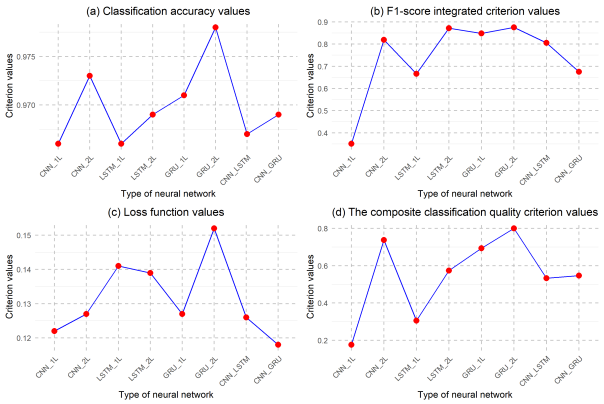


FIGURE 6. Results of the comparative analysis of different types of DL neural networks: a) classification accuracy; b) $F1$-score integrated criterion; c) loss function values; d) composite quality criterion for data classification.

data analysis, excelling in overall classification accuracy (Accuracy) as well as in accuracy for individual classes (Precision, Recall, F1-score). This is further corroborated by the distribution patterns of the composite criterion values (Comp QC = 0.8).

It is important to note that the values of Precision, Recall, and $F1$-score vary with different DL neural network types and structures. This variation underlines the importance of using the $F1$-score as an integrated criterion, calculated for relevant classes as outlined in section III-D. For more accurate comparability, the range of $F1$-score values across all models was normalized (the minimal and maximal $F1$-score criterion values were determined using the matrix of all $F1$-score evaluated for all models), ensuring a consistent scale for different $F1$-score vectors. The coefficients 'a' and 'b' in equation (7) remain constant across all models.

Further analysis of the integrated $F1$-score values for different models reaffirms the superiority of the two-layer GRU RNN model. In this instance, the integrated $F1$-score is 0.876, which is higher than that of other models evaluated.

It's important to highlight that the lowest classification accuracy was observed in samples pertaining to the first class. This outcome can largely be attributed to the limited number of samples available - only 79 in total, with 22 allocated for the test subset. Such a small sample size poses challenges for effective network training. Notably, augmenting the number

TABLE 12. Comparison of various models for multiclass problem-solving using different DL models for cancer identification.

Reference	Number of cancer types	Methodology	Accuracy,%
Gupta at al. (2022) [25]	5	DL with CNN	92
Karthika at al. (2023) [6]	2	DL with CNN	94.56
Mostavi at al. (2020) [7]	33	DL with CNN	93.9 - 95.0
Chuang at al. (2021) [8]	11	DL with CNN	95.4 - 97.4
Ramirez at al. (2020) [9]	33	DL with GCNN	89.9 - 94.7
Srikantamurthy at al. (2023) [26]	8	DL with CNN-LSTM	92.5

of samples has been shown to considerably decrease the variability in F1-score values.

The next phase in the simulation process involves utilizing an ensemble of machine learning methods. As noted in section III-C, this step involves the parallelizing data processing and implementing a consensus decision-making approach based on the interim outcomes from the previous stage. Such a strategy is expected to significantly improve objectivity in finalizing decisions about the object's state.

The simulation results on applying the hybrid model based on the ensemble of DL and ML methods are shown in Figure 7.

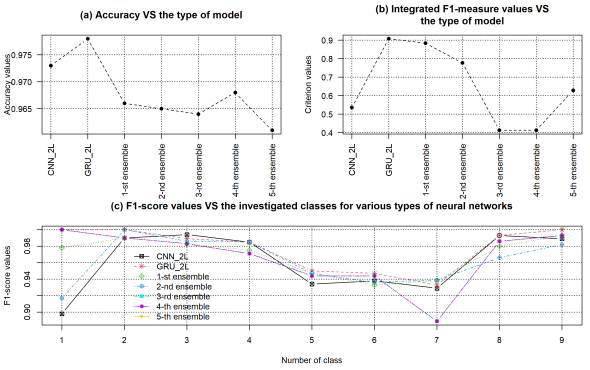


FIGURE 7. Results of the simulation regarding a comparative analysis of DL and ML method ensembles effectiveness.

From the analysis of the results, it becomes clear that using a DL-based models ensemble for classifying a single gene expression dataset does not necessarily offer an advantage in terms of classification accuracy. The quality of sample identification is diminished when compared to the use of optimally tuned two-layer CNN and GRU RNN models.

However, it's noteworthy that the first ensemble of DL and ML models shows high accuracy in categorizing objects into individual classes (F1-score integrated value). Furthermore, when compared to similar multiclass problem-solving using different DL models for cancer identification as presented in Table 12, the classification accuracy is higher in all instances when using the investigated DL models. This underscores the significance of selecting optimal model hyperparameters tailored to the specific data being analyzed.

Considering the research outlined in [27], [28], and [29], we can highlight the key performances of our proposed

technique. In these prior studies, the authors developed effective methods for selecting informative attributes (genes) and applied both machine learning and deep learning techniques to identify various types of cancer. While these studies yielded interesting results, their focus was primarily on feature selection followed by the application of suitable classifiers for sample identification, utilizing 10-fold cross-validation during model training.

In contrast, our research explored a range of deep learning models, including hybrid models, for the classification of various cancer types based on a comprehensive set of genes (19,947). The main objective of our study was to optimize the hyperparameters of the models. This was achieved through the combined use of the Bayesian optimization algorithm and k-fold cross-validation in each epoch of the algorithm's application. Additionally, we enhanced the classification quality criteria by introducing an integrated quality criterion, allowing for a more meticulous evaluation of the classification results. This approach represents the principal distinction between our methodology and the existing ones, offering a more comprehensive and refined analysis in the field of cancer classification.

A minor decrease in sample classification accuracy with ensemble-based DL models could be offset by the increased objectivity in making final decisions about the object's state. In multiclass problems addressed by models ensemble. Models can show for individual samples different identification results. This can lead to a slight drop in classification accuracy, as observed in our results. Nevertheless, higher objectivity is attained through the consistent identification of sample states across various methods. Improving the accuracy of the samples identification, in this instance, could be achieved by a more detailed pre-processing of gene expression data, employing gene ontology analysis, cluster, and bicluster analyses. Exploring these methods further will be the focus of our subsequent research.

V. CONCLUSION

This research has performed a comprehensive comparative analysis of various DL models for gene expression data processing, highlighting different techniques' strengths, weaknesses, and potential applications. The research encompassed different types and architectures of CNNs and RNNs, including hybrid models that combine both models. By employing a Bayesian optimization algorithm with 5-fold cross-validation during the appropriate model training, optimal hyperparameters for each model were determined. The study found that two-layer GRU RNN was most effective for classifying gene expression data, achieving a classification accuracy of 97.8%. Further, we proposed a hybrid model utilizing various DL techniques for gene expression data classification. This model represented as a step-by-step information processing flowchart, applies deep learning models in parallel at the first stage to form intermediate solutions, which are then processed by a decision tree-based classifier (CART) at the second hierarchical level. Simulations using various combinations and quantities of DL models at the first level indicated that GRU-based recurrent networks were more effective regarding the classification quality criteria, suggesting that adding complexity through more neural networks does not necessarily yield better classification accuracy results. However, a minor decrease in sample classification accuracy with ensemble-based DL models could be offset by the increased objectivity in making final decisions about the object's state.

The prospects of our research are focused on developing and evaluating various hybrid models for gene expression data processing, which will integrate gene ontology analysis, different clustering and biclustering techniques for grouping co-expressed gene expression profiles, and employ deep/machine learning methods for forming intermediate and final solutions regarding the state of the investigated objects.

ACKNOWLEDGMENT

The authors would like to express their gratitude to the research team from the Center for Cancer Genomics at the National Cancer Institute, National Institutes of Health, and The Cancer Genome Atlas (TCGA) for providing the opportunity to download and utilize the gene expression datasets from patients investigated for various types of cancer diseases.

REFERENCES

[1] E.-M. Nikolados and D. A. Oyarzún, "Deep learning for optimization of protein expression," *Current Opinion Biotechnol.*, vol. 81, Jun. 2023, Art. no. 102941, doi: 10.1016/j.copbio.2023.102941.

[2] R. Chaurasia and U. Ghose, "Human DNA/RNA motif mining using deep-learning methods: A scoping review," *Netw. Model. Anal. Health Informat. Bioinf.*, vol. 12, no. 1, p. 20, Apr. 2023, doi: 10.1007/s13721-023-00414-5.

[3] R. Halawani, M. Buchert, and Y.-P.-P. Chen, "Deep learning exploration of single-cell and spatially resolved cancer transcriptomics to unravel tumour heterogeneity," *Comput. Biol. Med.*, vol. 164, Sep. 2023, Art. no. 107274, doi: 10.1016/j.compbiomed.2023.107274.

[4] A. Davri, E. Birbas, T. Kanavos, G. Ntritsos, N. Giannakeas, A. T. Tzallas, and A. Batistatou, "Deep learning for lung cancer diagnosis, prognosis and prediction using histological and cytological images: A systematic review," *Cancers*, vol. 15, no. 15, p. 3981, Aug. 2023, doi: 10.3390/cancers15153981.

[5] S. Dixit, A. Kumar, and K. Srinivasan, "A current review of machine learning and deep learning models in oral cancer diagnosis: Recent technologies, open challenges, and future research directions," *Diagnostics*, vol. 13, no. 7, p. 1353, Apr. 2023, doi: 10.3390/diagnostics13071353.

[6] M. S. Karthika, H. Rajaguru, and A. R. Nair, "Evaluation and exploration of machine learning and convolutional neural network classifiers in detection of lung cancer from microarray gene—A paradigm shift," *Bioengineering*, vol. 10, no. 8, p. 933, Aug. 2023, doi: 10.3390/bioengineering10080933.

[7] M. Mostavi, Y.-C. Chiu, Y. Huang, and Y. Chen, "Convolutional neural network models for cancer type prediction based on gene expression," *BMC Med. Genomics*, vol. 13, no. 5, p. 44, Apr. 2020, doi: 10.1186/s12920-020-0677-2.

[8] Y.-H. Chuang, S.-H. Huang, T.-M. Hung, X.-Y. Lin, J.-Y. Lee, W.-S. Lai, and J.-M. Yang, "Convolutional neural network for human cancer types prediction by integrating protein interaction networks and omics data," *Sci. Rep.*, vol. 11, no. 1, Oct. 2021, Art. no. 20691, doi: 10.1038/s41598-021-98814-y.

[9] R. Ramirez, Y.-C. Chiu, A. Hererra, M. Mostavi, J. Ramirez, Y. Chen, Y. Huang, and Y.-F. Jin, "Classification of cancer types using graph convolutional neural networks," *Frontiers Phys.*, vol. 8, Jun. 2020, Art. no. 203, doi: 10.3389/fphy.2020.00203.

[10] S. Babichev, L. Yasinska-Damri, I. Liakh, and J. Škvor, "Hybrid inductive model of differentially and co-expressed gene expression profile extraction based on the joint use of clustering technique and convolutional neural network," *Appl. Sci.*, vol. 12, no. 22, p. 11795, Nov. 2022, doi: 10.3390/app122211795.

[11] S. Babichev, L. Yasinska-Damri, and I. Liakh, "A hybrid model of cancer diseases diagnosis based on gene expression data with joint use of data mining methods and machine learning techniques," *Appl. Sci.*, vol. 13, no. 10, p. 6022, May 2023, doi: 10.3390/app13106022.

[12] R. Jain, A. Jain, E. Mauro, K. LeShane, and D. Densmore, "ICOR: Improving codon optimization with recurrent neural networks," *BMC Bioinf.*, vol. 24, no. 1, Apr. 2023, Art. no. 132, doi: 10.1186/s12859-023-05246-8.

[13] X. Cao, A. Francis, X. Pu, Z. Zhang, V. Katsikis, P. Stanimirovic, I. Brajevic, and S. Li, "A novel recurrent neural network based online portfolio analysis for high frequency trading," *Expert Syst. Appl.*, vol. 233, Dec. 2023, Art. no. 120934, doi: 10.1016/j.eswa.2023.120934.

[14] R. R. Ema, A. Khatun, M. A. Hossain, M. R. Akhond, N. Hossain, and M. Y. Arafat, "Protein secondary structure prediction using hybrid recurrent neural networks," *J. Comput. Sci.*, vol. 18, no. 7, pp. 599–611, Jul. 2022, doi: 10.3844/jcssp.2022.599.611.

[15] V. Sobhani, A. Asgari, M. Arabfard, Z. Ebrahimpour, and A. Shakibaee, "Comparison of optimized machine learning approach to the understanding of medial tibial stress syndrome in male military personnel," *BMC Res. Notes*, vol. 16, no. 1, p. 126, Jun. 2023, doi: 10.1186/s13104-023-06404-0.

[16] F. Di Nunno, S. Zhu, M. Ptak, M. Sojka, and F. Granata, "A stacked machine learning model for multi-step ahead prediction of lake surface water temperature," *Sci. Total Environ.*, vol. 890, Sep. 2023, Art. no. 164323, doi: 10.1016/j.scitotenv.2023.164323.

[17] H. Mkindu, L. Wu, and Y. Zhao, "Lung nodule detection in chest CT images based on vision transformer network with Bayesian optimization," *Biomed. Signal Process. Control*, vol. 85, Aug. 2023, Art. no. 104866, doi: 10.1016/j.bspc.2023.104866.

[18] C. H. Goay, N. S. Ahmad, and P. Goh, "Transient simulations of high-speed channels using CNN-LSTM with an adaptive successive halving algorithm for automated hyperparameter optimizations," *IEEE Access*, vol. 9, pp. 127644–127663, 2021, doi: 10.1109/ACCESS.2021.3112134.

[19] *The Cancer Genome Atlas Program (TCGA)*. [Online]. Available: https://www.cancer.gov/ccg/research/genome-sequencing/tcga

[20] Illumina. *Sequencing*. [Online]. Available: https://www.illumina.com/

[21] L. Yasinska-Damri, S. Babichev, B. Durnyak, and T. Goncharenko, "Application of convolutional neural network for gene expression data classification," in *Proc. Int. Sci. Conf. Intellectual Syst. Decision Making Problem Comput. Intell.*, in Lecture Notes on Data Engineering and Communications Technologies, vol. 149, Sep. 2022, pp. 3–24, doi: 10.1007/978-3-031-16203-9_1.

[22] D. Kim, K. Kwon, K. Pham, J.-Y. Oh, and H. Choi, "Surface settlement prediction for urban tunneling using machine learning algorithms with Bayesian optimization," *Autom. Construct.*, vol. 140, Aug. 2022, Art. no. 104331, doi: 10.1016/j.autcon.2022.104331.

[23] J. Isabona, A. L. Imoize, and Y. Kim, "Machine learning-based boosted regression ensemble combined with hyperparameter tuning for optimal adaptive learning," *Sensors*, vol. 22, no. 10, p. 3776, May 2022, doi: 10.3390/s22103776.

[24] S. Vural, X. Wang, and C. Guda, "Classification of breast cancer patients using somatic mutation profiles and machine learning approaches," *BMC Syst. Biol.*, vol. 10, no. 3, pp. 264–276, Aug. 2016, doi: 10.1186/s12918-016-0306-z.

[25] S. Gupta, M. K. Gupta, M. Shabaz, and A. Sharma, "Deep learning techniques for cancer classification using microarray gene expression data," *Frontiers Physiol.*, vol. 13, Sep. 2022, Art. no. 952709, doi: 10.3389/fphys.2022.952709.

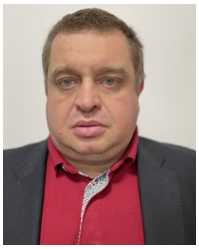
[26] M. M. Srikantamurthy, V. P. S. Rallabandi, D. B. Dudekula, S. Natarajan, and J. Park, "Classification of benign and malignant subtypes of breast cancer histopathology imaging using hybrid CNN-LSTM based transfer learning," *BMC Med. Imag.*, vol. 23, no. 1, p. 19, Jan. 2023, doi: 10.1186/s12880-023-00964-0.

[27] S. Afreen, A. K. Bhurjee, and R. M. Aziz, "Gene selection with game Shapley Harris hawks optimizer for cancer classification," *Chemometric Intell. Lab. Syst.*, vol. 242, Nov. 2023, Art. no. 104989, doi: 10.1016/j.chemolab.2023.104989.

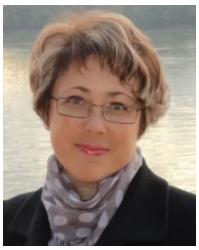
[28] A. A. Joshi and R. M. Aziz, "Deep learning approach for brain tumor classification using metaheuristic optimization with gene expression data," *Int. J. Imag. Syst. Technol.*, Dec. 2023, doi: 10.1002/ima.23007.

[29] R. Mahto, S. U. Ahmed, R. U. Rahman, R. M. Aziz, P. Roy, S. Mallik, A. Li, and M. A. Shah, "A novel and innovative cancer classification framework through a consecutive utilization of hybrid feature selection," *BMC Bioinf.*, vol. 24, no. 1, p. 479, Dec. 2023, doi: 10.1186/s12859-023-05605-5.

SERGII BABICHEV currently holds professorial roles with both the Department of Physics, Kherson State University, Ukraine, and the Department of Informatics, Jan Evangelista Purkyně University in Ústí nad Labem, Czech Republic. He has garnered a wealth of experience and scholarly distinction in technical diagnostics and bioinformatics, with a career that traverses various pivotal roles across Ukraine and the Czech Republic. His dive into IT and computational biology has yielded over 150 scientific works, including notable contributions to the Scopus and WoS databases. His entry into the scientific community was marked by the C.Sc. thesis on "Automated System of Metal Strength Properties Technical Diagnostic Based on Hybrid Neural Networks" and was followed by an intensive exploration into bioinformatics, peaking with the D.Sc. thesis titled "Methods, Models, and Information Technology of Gene Expression Profiles Processing for the Purpose of Gene Regulatory Networks Reconstruction." His current research interests include developing and scrutinizing various types of hybrid models, focusing on the amalgamated application of data mining and machine/deep learning techniques for gene expression data processing. His work aims to enhance disease diagnosis systems, pinpoint subsets of co-expressed genes, and reconstruct gene regulatory networks.



IGOR LIAKH is currently an Associate Professor with the Faculty of Information Technologies, Uzhhorod National University, Ukraine. He has garnered extensive experience and has navigated through various key positions throughout his career in Ukraine. His immersion into the IT field has yielded over 60 scientific works, including significant contributions to the Scopus and WoS databases. His entrance into the scientific community was marked by the defense of his candidate's dissertation titled "Information Technologies of Data Protection Systems in Mass Media Electronic Means," followed by intensive research into bioinformatics, which smoothly transitioned into writing the Ph.D. dissertation, upon which he is now diligently working. His current research interests include the development and exploration of various types of hybrid models, concentrating on the combined application of data mining and machine/deep learning methods for gene expression data processing. His work is aimed at improving disease diagnostic systems, identifying subsets of co-expressed genes, and reconstructing gene regulatory networks.



IRINA KALININA is currently an Associate Professor with the Faculty of Computer Science, Petro Mohyla Black Sea National University, Ukraine. Her expertise and scholarly contributions lie in the domain of Bayesian analysis and data mining technologies. She has authored over 100 scientific articles, a notable portion of which are indexed in the Scopus database. The inception of her scientific career was highlighted by the candidate's thesis, titled "Geometric Modeling of Elements of Flow Parts of Radial-Axial and Axial Turbomachines." Subsequent research pivoted toward data analysis technologies, exploring neural network methods, and employing technologies for modeling complex systems via colored Petri nets. Her current research interests include big data analysis systems utilizing Bayesian analysis methods and applying probabilistic statistical analysis methods in machine learning challenges.