Received: November 5, 2024; **Accepted:** February 28, 2025; **Published:** March 3, 2025.
DOI. 10.30473/coam.2025.72710.1269
Winter-Spring (2025) Vol. 10, No. 1, (73-89)
Research Article

Open Access

Control and Optimization in Applied Mathematics - COAM

Hybrid of Convolutional Neural Network and Support Vector Machine for Cancer Type Prediction

Soghra Mikaeyl Nejad ✉

Department of Computer Engineering and Information Technology, Payame Noor University, Tehran, Iran.

✉ **Correspondence:**
Soghra Mikaeyl Nejad
E-mail:
smikaeylnejad@pnu.ac.ir

How to Cite
Mikaeyl Nejad, S. (2025). "Hybrid of convolutional neural network and support vector machine for cancer type prediction", Control and Optimization in Applied Mathematics, 10(1): 73-89, doi: 10.30473/coam.2025.72710.1269

Abstract. Gene expression signatures reflect the response of cell tissues to diseases, genetic disorders, and drug treatments, containing hidden patterns that can provide valuable insights for biological research and cancer diagnostics. This studyproposes a hybrid deep learning approach combining convolutional neural networks (CNNs) and support vector machines (SVMs) to classify cancer types using unstructured gene expression data. We applied three hybrid CNN-SVM models to a dataset of 10,340 samples spanning 33 cancer types from the Cancer Genome Atlas. The CNN component extracted latent features from the gene expression data, while the SVM replaced the softmax layer to enhance classification robustness. Among the proposed models, the Hybrid-CNN-SVM model achieved superior performance, demonstrating excellent prediction accuracy and outperforming other models. This study highlights the potential of hybrid deep learning frameworks for cancer type prediction and underscores their applicability to high-dimensional genomic datasets.

Keywords. Deep learning, Convolutional neural networks, Support vector machine, Gene expression, The cancer genome atlas, Cancer type prediction.

MSC. 92C40; 62H30.

1 Introduction

Cancer remains one of the most significant global health challenges of the 21st century, ranking as the second leading cause of death worldwide after cardiovascular diseases. According to the world health organization, over 10 million people are diagnosed with cancer annually, with this number projected to rise to 15 million in the near future [13]. Despite advancements in medicine, the late detection of cancer continues to be a critical issue, as more than 70% of patients are diagnosed at advanced stages, rendering them ineligible for surgical interventions (for more details refer to [19],[23]). Early detection and classification of cancer types, therefore, play a pivotal role in improving survival rates and treatment outcomes. Research has shown that identifying cancer at its early stages significantly enhances prognosis, underscoring the importance of developing accurate and efficient diagnostic tools.

In recent years, the advent of high-throughput sequencing technologies in cancer genomics has generated vast amounts of complex data, presenting both challenges and opportunities for cancer research. Deep learning (DL), a subset of machine learning, has emerged as a powerful tool for analyzing large-scale, high-dimensional datasets. Unlike traditional machine learning methods, such as regression, deep learning models leverage neural network architectures to automatically learn patterns and structures hidden within raw data. This capability allows deep learning models to scale effectively with the increasing volume and complexity of data, making them particularly well-suited for cancer detection and classification tasks. By harnessing the potential of deep learning, researchers aim to develop robust models for early cancer screening and tumor type classification, ultimately contributing to improved patient outcomes and survival rates (see e.g., [12] and [21]).

2 Literature Reviews

Gene expression analysis and RNA-Seq data evaluation have been extensively studied using various computational and machine learning approaches. Clustering-based methods, such as hierarchical clustering [12], k-means clustering [20], Pearson similarity clustering [8], term frequency-inverse document frequency (TF-IDF) clustering [16], and self-organizing maps (SOM) [10], have been widely applied to analyze gene expression data and identify gene groups. More recently, clustering techniques have been combined with deep neural networks to enhance the analysis of gene expression patterns [18]. These methods have provided valuable insights into gene expression data, but their reliance on unsupervised learning limits their classification accuracy, particularly for cancer type prediction.

Machine learning and deep learning techniques have emerged as powerful tools for gene expression analysis, particularly in cancer research. Xiao et al. [22] proposed a semi-supervised stacked sparse autoencoder (SSAE) for cancer prediction using RNA-Seq data, achieving promising results. Jiang et al. [7] introduced a generative adversarial network (GAN) with a denoising autoencoder (DAE) as the generator and a multilayer perceptron (MLP) as the discriminator for disease gene prediction. Huang et al. [6] applied deep learning models, including Cox-nnet, DeepSurv, and AECOX, to predict cancer survival prognosis from RNA-Seq data. Other studies have focused on feature extraction from high-dimensional data, such as Arowolo et al. [2], who used PCA and traditional classifiers like KNN and decision trees, and Barman and Kwon [3], who developed a genetic algorithm-based Boolean network

inference (GABNI) for time-series gene expression data. Spiking neural networks (SNNs) [17], dynamic Bayesian networks [1], and dual attention recurrent neural networks (RNNs) [14] have also been employed to model and analyze gene expression data, demonstrating the versatility of machine learning methods in this domain.

In the context of cancer type classification, several studies have explored the application of deep learning and hybrid models. Li et al. [9] used genetic algorithms and k-nearest neighbors (KNN) to achieve over 90% prediction accuracy across 31 cancer types using the cancer genome atlas (TCGA) dataset. Lyu and Haque [11] converted gene expression data into image samples for training convolutional neural networks (CNNs), achieving over 95% accuracy across 33 TCGA cancer types. Mostavi et al. [15] applied CNN architectures to RNA-Seq data, achieving 95% classification accuracy while considering tissue origins. Ramirez et al. [19] developed a Graph CNNs (GCNNs) that incorporated prior knowledge from protein-protein interaction and gene co-expression networks for cancer type prediction. These studies highlight the potential of deep learning and hybrid models in addressing the challenges of high-dimensional gene expression data.

Building on these advancements, this study proposes a Hybrid-CNN-SVM model to classify cancer marker genes using RNA-Seq data from 10,340 samples across 33 cancer types. The CNN component extracts latent features from unstructured gene expression inputs, while the SVM replaces the softmax layer for robust classification. This hybrid approach aims to leverage the strengths of both CNNs and SVMs to improve classification accuracy, generalization, and robustness in cancer type prediction. The proposed method is evaluated using comprehensive experimental setups, including cross-validation, to ensure reliable and reproducible results.

3 Materials and Method

In this section, we describe the CNN and the SVM. Then, we will introduce the hybrid method of CNN-SVM and explain its implementation flowchart. In the following, we will introduce both used databases and finally, we will evaluate the evaluation criteria to compare the performance of the methods presented in this article.

3.1 Convolution Neural Networks (CNNs)

Convolutional neural networks are a one of the most important deep learning models in which multiple layers are taught in a powerful way. CNNs are highly efficient in learning hierarchical representations, making them one most widely used methods in fields like computer vision, natural language processing, and bioinformatics. In this study, CNNs are employed to extract high-level features from the biological dataset for subsequent classification. A CNN typically consists of three main layers, each performing a specific function in the feature extraction process:

- **Convolution layer:** The convolutional layer is the core building block of CNNs. It applies a set of filters (kernels) to the input data to create feature maps. These filters are initialized with random

values and are updated during training to capture meaningful patterns in the data, such as edges in images or specific biological signatures in gene expression data. The convolution operation involves multiplying the input data with the filter weights and summing the results, which allows the network to detect local patterns in the data.

- **Pooling layer:** The pooling layer is usually placed after the convolution layer and can be used to reduce the size of feature maps and network parameters. Like convolutional layers, pooling layers are unchanged from displacement due to the side features in their calculations. The two most common pooling methods are:

  Max Pooling: Selects the maximum value from each pooling window.

  Average Pooling:Computes the average value within each pooling window.

- **Fully connected (FC) layer:** These layers convert the 2D feature maps from the pooling step into a one-dimensional feature vector. Fully connected layers act like their counterparts in traditional artificial neural networks and often comprise the majority of the network's parameters. However, the high number of parameters in FC layers can lead to significant computational costs, especially when training on large datasets.

The training of a CNN involves two main stages:

- **Feed-Forward Stage:** The input data is passed through the network, layer by layer, to produce an output. Each layer performs operations such as convolution, activation, and pooling to extract features and transform the data into a form suitable for classification.
- **Back-Propagation Stage:** The network output is compared with the ground truth using an error function, and the error rate is calculated. The back-propagation algorithm adjusts the network parameters by computing the gradient of the error with respect to each parameter using the chain rule. These gradients are used to update the weights and biases of the network. The feed-forward and back-propagation steps are repeated iteratively until the network converges to an optimal solution.

While CNNs are highly effective at processing complex data, they face several challenges:

- **Slow Training:** Training CNNs can be computationally expensive, particularly due to the large number of parameters in fully connected layers.
- **Sensitivity to Noise:** Large datasets often contain noise, which can negatively impact the accuracy and increase training and testing time, especially in the FC layers.

To address these limitations, we propose a hybrid approach combining CNNs and SVMs. CNNs are utilized to extract high-level features from the input data, while SVMs are employed for robust classification. This integration takes advantage of the strengths of both methods:

- **CNNs:** Automatically extract complex, hierarchical features from raw data.
- **SVMs:** Efficiently classify the extracted features, even in the presence of noise, by finding an optimal decision boundary.

In this study, the CNN-SVM hybrid method is applied to classify the dataset, achieving high accuracy in distinguishing cancer types and normal tissues.

3.2 Support Vector Machine (SVM)

Support vector machines are supervised machine learning algorithms designed for classification and regression tasks. SVMs are particularly effective in scenarios where the data is high-dimensional or not linearly separable. In this study, SVMs are used to classify the features extracted by the CNN. The primary goal of SVMs is to find the optimal hyperplane that separates data points of different classes with the maximum margin. The key concepts of SVMs include:

- **Hyperplane with Maximum Margin:** SVMs construct a hyperplane that maximizes the distance between the two closest data points from each class (support vectors). This margin maximization ensures better generalization and reduces the risk of overfitting.
- **Kernel Functions:** When the data is not linearly separable, SVMs use kernel functions to map the input data into a higher-dimensional space where it becomes linearly separable.

SVMs construct a hyperplane that maximizes the distance between the two closest data points from each class (support vectors). This margin maximization ensures better generalization and reduces the risk of overfitting. The SVM classifier form is as follow equation:

$$g(x) = \sum_{i=1}^{L_s} \alpha_i d_i K(x_i, x) + \alpha_0, \tag{1}$$

which K is the kernel function and x_i represents the support vector which is obtained from the training data. L_s represents the number of support vectors and d_i represents the corresponding class number x_i, and finally α_i are fixed numbers that are obtained in the training step. By definition, support vectors are elements of educational data that are located exactly on or within the boundaries of classification decision making. In other words, these vectors include samples that are more difficult to classify than other samples [11]. In Figure 1, the schematic of a SVM is provided.

Since the dataset involves multiple classes (33 cancer types and 23 normal tissues), the SVM is extended to handle multi-class classification using the One-Against-One (OAO) method. In OAO, the SVM constructs binary classifiers for each pair of classes, resulting in $(n \times (n-1))/2$ classifiers for n classes. The final classification is determined by majority voting across all classifiers. SVMs offer several advantages, including:

- **Robustness to Noise:** SVMs are less sensitive to noisy data due to their reliance on support vectors.
- **Efficient Classification:** SVMs provide high accuracy in separating classes, even in high-dimensional feature spaces.

3.3 Hybrid-CNN-SVM Approach for Cancer Classification

The Hybrid-CNN-SVM approach combines the feature extraction capabilities of CNNs with the classification power of SVMs. In this study, CNNs are used to extract high-level features from the dataset of 33 cancer types and 23 normal tissues and SVMs classify the extracted features into cancerous or normal

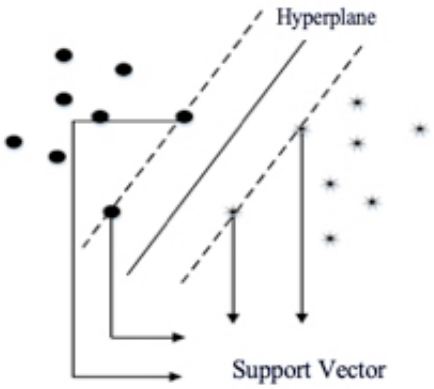

Figure 1: The schematic of a binary SVM.

tissues with high accuracy. This hybrid approach addresses the limitations of CNNs, such as sensitivity to noise and slow training, while leveraging SVMs' ability to handle noisy data and efficiently classify complex patterns. By combining these methods, the study achieves improved accuracy and robustness in analyzing the dataset.

- **1D-CNN-SVM**

  In the first model, the input layer of the CNN processes gene expression data represented as a vector, applying one-dimensional kernels to the input. High-dimensional gene expression data, comprising 7109 genes, is embedded into a vector format by appending nine zeros, resulting in a vector of length 7100. The convolutional layer consists of 100 tiles with 32 filters, each having a kernel length of 71. To identify distinguishing features from the input, the output of the convolutional layer is passed through a max-pooling layer and subsequently through a fully connected layer. The extracted features are then fed into the final layer of the CNN, which is replaced with an SVM classifier. Using these features, the SVM trains on the data to generate its own model for feature computation. The structure of this model is illustrated in Figure 2.

- **Vanilla-CNN-SVM**

  The second model employs a straightforward CNN architecture combined with a SVM classifier. The input layer processes data in the form of a 100×71 matrix. A convolutional layer with a 10×10 kernel is applied, followed by batch normalization and ReLU activation to enhance feature extraction and model stability. Max pooling is then utilized to reduce the dimensionality of the features. Subsequently, a FC layer is introduced, and the softmax layer is replaced with an SVM classifier. This architecture is illustrated in Figure 3.

- **Hybrid-CNN-SVM**

  The third model, illustrated in Figure 4, employs two parallel convolutional layers on the 2D input data (comprising 100×71 genes). One layer operates vertically, utilizing 32 one-dimensional kernels of size (1×71), while the other operates horizontally with 32 one-dimensional kernels of size (100×1). The outputs of these parallel layers are processed through max-pooling , and the

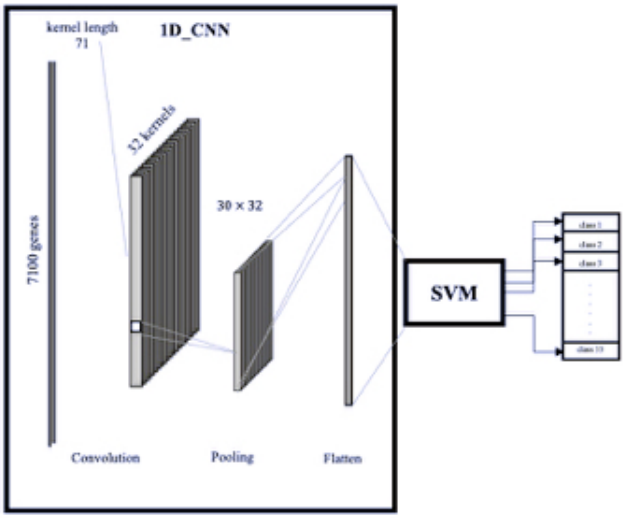

Figure 2: 1D-CNN-SVM.

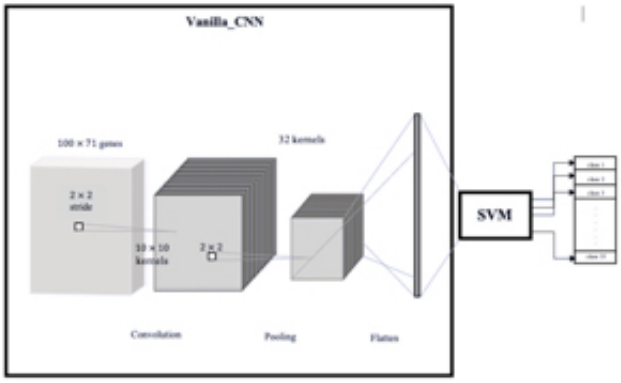

Figure 3: Vanilla-CNN-SVM.

resulting outputs are subsequently flattened and concatenated. Similar to the previous models, the CNN classification layer is replaced by an SVM classifier.

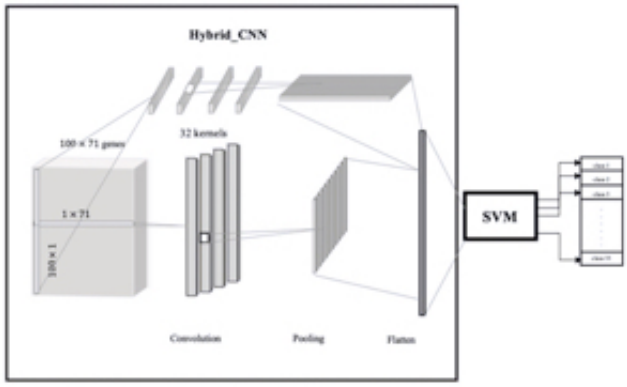

Figure 4: Hybrid-CNN-SVM.

3.4 Dataset

In the initial step of this study, RNA-Seq data were obtained from The Cancer Genome Atlas (TCGA) as referenced in [4, 5]. The dataset included 10,340 samples spanning 33 distinct cancer types, along with 731 matched normal tissue samples from 23 tissue types. Gene expression levels in the dataset were transformed using the log_2(FPKM +1) formula, where FPKM (Fragments Per Kilobase per Million mapped reads) represents the unit of gene expression measurement. Following the preprocessing strategy outlined in [4], genes with a mean expression level of < 0.5 and a standard deviation of < 0.8 across all samples, irrespective of cancer type, were excluded. This preprocessing step was implemented to minimize noise and remove genes with limited discriminative value in the dataset.

4 Results

4.1 Model Construction, Hyperparameters and Training

All three models were implemented using Keras 2.3 with TensorFlow 2.0 as the backend. The code was executed on Google Colab utilizing GPU acceleration. As previously mentioned, CNNs serve as the primary algorithm in this study. The optimal architecture parameters, including the number and size of kernels, kernel stride, and the number of nodes in the FC layer, are summarized in Table 1.

Table 1: Key Hyperparameters

Dense Layer Size	Filter	Kernel
128	(1,71)	32

The 1D-CNN-SVM model processes a one-dimensional vector of gene expressions arranged alphabetically by gene symbols. In contrast, the Vanilla-CNN-SVM and Hybrid-CNN-SVM models utilize a two-dimensional input matrix of dimensions 100×71, representing reshaped gene expression data. These architectural differences allow for a comparative analysis of performance across varying input formats.

The 1D-CNN-SVM model replaces the conventional softmax activation and cross-entropy loss function with a SVM-based hinge loss function, which directly optimizes the margin between classes. Similarly, the Vanilla-CNN-SVM and Hybrid-CNN-SVM models incorporate the SVM hinge loss function for classification. The hybrid models combine the strengths of CNN feature extraction with SVM classification for enhanced performance.

The network parameters for all models were optimized using the Adam optimizer. Hyperparameter tuning was performed using a grid search method to optimize key parameters, such as the number and size of kernels, kernel stride, and the number of the FC layer nodes. The training process employed a batch size of 47 for the 1D-CNN-SVM model and 128 for the Vanilla-CNN-SVM and Hybrid-CNN-SVM models, with a fixed number of 50 epochs. Early stopping with a patience of 4 epochs was implemented for the latter models to prevent overfitting. The activation function (AF) used in all layers was ReLU, while the final layer used softmax for prediction in the Vanilla-CNN-SVM and Hybrid-CNN-SVM models.

For training and evaluation, 80% of the dataset was allocated for training and 20% for validation. A total of 10,340 tumor samples were used for training all three models. To ensure robustness and mitigate stochastic dependencies inherent in neural networks, a 5-fold cross-validation procedure was repeated six times. The mean and standard deviation of classification accuracy were reported for all models.

The 1D-CNN-SVM model achieved an average classification accuracy of $95.11 \pm 0.45\%$, surpassing the $94.66 \pm 0.43\%$ accuracy of the standard 1D-CNN model. The Hybrid-CNN-SVM model demonstrated the highest accuracy among the tested architectures, achieving 96.00% average accuracy, compared to 95.00% for the standard Hybrid-CNN model. Similarly, the Vanilla-CNN-SVM model exhibited improved performance compared to its non-SVM counterpart.

4.2 Comparative Analysis

In this study, we evaluated the proposed Hybrid-CNN-SVM model against the standard CNN and standard SVM in terms of multiple performance metrics, including Accuracy, Generalization, Robustness to Noise, Interpretability, and Use Case Suitability. The results indicate that the hybrid model offers distinct advantages over the standalone CNN and SVM approaches, particularly in balancing accuracy and generalization while addressing limitations in computational efficiency and robustness.

The Hybrid-CNN-SVM model demonstrated a modest improvement in classification accuracy compared to the standard CNN, with a 1% increase. While the improvement may seem incremental, it reflects the complementary strengths of the two models. The CNN component excels in extracting hierarchical features from input data, while the SVM provides a robust decision boundary for classification. In contrast, the standard CNN relies on the softmax classifier, which is prone to overfitting, and the standard SVM struggles with high-dimensional feature spaces when used alone. Thus, the hybrid approach

achieves a balance between feature extraction and robust classification, outperforming both standalone models in terms of accuracy.

Generalization, or the ability of a model to perform well on unseen data, is a critical metric in machine learning. The Hybrid-CNN-SVM model exhibited superior generalization compared to the standard CNN due to the optimization criteria used by its SVM component. The CNN employs empirical risk minimization (ERM), which focuses on minimizing training error, often at the expense of generalization. This can lead to overfitting, especially when the training data is noisy or limited. Conversely, the SVM in the hybrid model adopts structural risk minimization (SRM), which explicitly balances training error and model complexity, reducing overfitting and improving generalization. The standard SVM, while also employing SRM, lacks the powerful feature extraction capabilities of CNN, limiting its generalization ability in complex, high-dimensional tasks. Robustness to noise is another area where the Hybrid-CNN-SVM model outperforms the standard CNN. The SVM component's optimization strategy ensures that the decision boundary is less sensitive to noisy or outlier data points, which can significantly degrade the performance of the softmax classifier used in the standard CNN. The standard SVM also exhibits strong robustness to noise due to its margin-maximization principle. However, the lack of feature extraction capabilities in the standard SVM limits its applicability to complex datasets. By combining CNN's ability to extract meaningful features with SVM's robustness to noise, the hybrid model achieves a higher level of resilience to noisy data compared to either standalone approach.

The suitability of each model for specific use cases depends on the nature of the task, the dataset, and the performance requirements. The Hybrid-CNN-SVM model is particularly well-suited for applications where both high accuracy and robust generalization are critical, such as medical image analysis, fault detection, or biometric recognition. Its ability to handle high-dimensional data with noise makes it a versatile choice for complex classification tasks. The standard CNN is more appropriate for applications where computational efficiency and scalability are paramount, such as real-time image processing or embedded systems. The standard SVM, while effective for smaller datasets or low-dimensional problems, struggles with scalability and feature extraction, limiting its use in high-dimensional or large-scale tasks. The summary of the comparative analysis is presented in Table 2.

Table 2: Summary of comparative analysis

Metric	Hybrid-CNN-SVM	Standard CNN	Standard SVM
Accuracy	High	Moderate	Moderate
Generalization	High	Moderate	High
Robustness to Noise	High	Moderate	High
Use Case Suitability	Versatile	High	Limited

The Hybrid-CNN-SVM model effectively combines the strengths of CNN and SVM, achieving a balance between accuracy, generalization and robustness. The hybrid approach is particularly well-suited for complex classification tasks requiring high reliability, making it a valuable alternative to standalone CNN or SVM models. Future research could explore further optimization of the hybrid architecture to enhance computational efficiency and broadening its applicability to a wider range of use cases.

4.3 Classification Report

Classification report used to measure the quality of predictions from a classification algorithm. This report includes Precision, Recall, F1-score and Support.The Precision is the fraction of True Positive elements divided by the total number of positively predicted units for each class. In particular, True Positive are the elements that have been labelled as positive by the model and they are actually positive, while False Positive are the elements that have been labelled as positive by the model, but they are actually negative.

$$\text{Precision}_k = \frac{\text{TP}_k}{\text{TP}_k + \text{FP}_k}.$$

Recall is the ability of a classifier to find all positive instances. For each class it is defined as the ratio of true positives to the sum of true positives and false negatives.

$$\text{Recall}_k = \frac{\text{TP}_k}{\text{TP}_k + \text{FN}_k}.$$

F1-score is a weighted harmonic mean of Precision and Recall normalized between 0 and 1. F-score of 1 indicates a perfect balance as Precision and the Recall are inversely related. A high F1-score is useful where both high Recall and Precision is important. In order to obtain Macro F1-score, Macro-Precision and Macro-Recall should be computed before. They are respectively calculated by taking the average Precision for each predicted class and the average Recall for each actual class.

$$\text{Macro Average Precision} = \frac{\sum_{k=1}^{K} \text{Precision}_k}{K},$$
$$\text{Macro Average Recall} = \frac{\sum_{k=1}^{K} \text{Recall}_k}{K}.$$

Eventually, Macro F1-Score is the harmonic mean of Macro-Precision and Macro-Recall:

$$\text{Macro F1-Score} = 2 \cdot \frac{\text{Macro Average Precision} \cdot \text{Macro Average Recall}}{\text{Macro Average Precision}^{-1} + \text{Macro Average Recall}^{-1}}.$$

Support is the number of actual occurrences of the class in the test data set. Imbalanced support in the training data may indicate the need for stratified sampling or rebalancing.

To evaluate the performance of the proposed method, all three CNN-SVM models were trained with all 10340 tumor samples in two different approaches. In the first approach, a simple convolution network CNN is used. In the second approach, the proposed hybrid approach CNN-SVM is used. Table 3 and Table 4 represent the classification reports of methods for databases, respectively.

The classification results demonstrate that the proposed CNN-SVM models effectively enhance the performance of cancer type prediction compared to standalone CNN models. The 1D-CNN-SVM and Hybrid-CNN-SVM models achieved superior or comparable accuracy, Precision, Recall, and F1-scores across most cancer types, as evidenced by the macro and weighted averages. For instance, the Hybrid-CNN-SVM model outperformed the Hybrid-CNN model, achieving a higher weighted average F1-score of 0.96 compared to 0.93. Similarly, the 1D-CNN-SVM model achieved a weighted average F1-score of 0.94, slightly outperforming the 1D-CNN model. These improvements highlight the effectiveness of integrating SVM as a classifier, leveraging the feature extraction capability of CNN while enhancing classification accuracy. However, the Vanilla-CNN-SVM model did not exhibit significant improvements

over the Vanilla-CNN model, indicating the potential influence of architectural design on performance. Overall, the proposed CNN-SVM approach demonstrates robust classification capabilities, particularly in complex datasets with high-dimensional gene expression data.

Table 3: Comparison of classification report

Cancer type	Classifications report of ID-CNN-model				Classifications report of ID-CNN-SVM-model			
	Precision	Recall	F1-score	support	Precision	Recall	F1-score	support
ACC	0.83	1.00	0.91	15	1.00	0.94	0.97	16
BLCA	0.96	0.92	0.94	83	0.94	0.87	0.90	83
BRCA	1.00	0.99	1.00	222	0.94	1.00	0.96	221
CESC	0.76	0.97	0.85	61	0.83	0.90	0.87	61
CHOL	0.80	0.57	0.67	7	0.88	0.88	0.88	8
COAD	0.77	0.96	0.86	96	0.90	0.75	0.82	95
DLBC	1.00	1.00	1.00	9	1.00	1.00	1.00	10
ESCA	0.96	0.70	0.81	33	0.89	0.78	0.83	32
GBM	0.63	1.00	0.78	33	0.94	1.00	0.97	34
HNSC	0.90	0.94	0.92	101	0.99	0.91	0.95	100
KICH	1.00	0.85	0.92	13	0.91	0.77	0.83	13
KIRC	0.97	0.92	0.94	108	1.00	0.94	0.97	108
KIRP	0.82	0.96	0.89	57	0.86	0.98	0.92	58
LAML	1.00	1.00	1.00	31	0.94	1.00	0.97	30
LGG	1.00	0.79	0.88	106	1.00	0.99	1.00	105
LIHC	0.99	0.97	0.98	75	0.99	0.97	0.98	75
LUAD	0.93	0.93	0.93	107	0.97	0.91	0.94	107
LUSC	0.98	0.84	0.90	100	0.87	0.93	0.90	101
MESO	1.00	0.82	0.90	17	0.94	1.00	0.97	17
OV	1.00	0.97	0.99	76	0.99	1.00	0.99	76
PAAD	0.97	0.94	0.96	36	0.97	1.00	0.99	35
PCPG	0.92	0.97	0.95	36	0.93	1.00	0.96	37
PRAD	1.00	0.99	0.99	100	1.00	1.00	1.00	100
READ	0.70	0.21	0.33	33	0.48	0.70	0.57	33
SARC	0.88	0.94	0.91	53	0.96	0.90	0.93	52
SKCM	0.99	0.96	0.97	94	1.00	0.97	0.98	95
STAD	0.88	0.99	0.93	75	0.91	0.95	0.93	74
TGCT	1.00	0.93	0.96	28	0.93	0.97	0.95	29
THCA	1.00	1.00	1.00	102	0.99	1.00	1.00	101
THYM	1.00	1.00	1.00	24	1.00	0.92	0.96	24
UCEC	0.93	0.99	0.96	110	0.95	0.95	0.95	111
UCS	0.88	0.64	0.74	11	0.86	0.55	0.67	11
UVM	0.94	1.00	0.97	16	1.00	1.00	1.00	16
Accuracy			0.93	2068			0.94	2068
Marco avg.	0.92	0.90	0.90	2068	0.93	0.92	0.92	2068
Weighted avg.	0.94	0.93	0.93	2068	0.94	0.94	0.94	2068

Table 4: Comparison of classification report

Cancer type	Classifications report of Hybrid-CNN-model				Classifications report of Hybrid-CNN-SVM-model			
	Precision	Recall	F1-score	support	Precision	Recall	F1-score	support
ACC	1.00	1.00	1.00	16	0.94	1.00	0.97	16
BLCA	0.96	0.93	0.94	83	0.96	0.98	0.97	83
BRCA	0.99	0.99	0.99	222	1.00	0.99	0.99	222
CESC	0.98	0.97	0.98	61	1.00	0.93	0.97	61
CHOL	0.67	0.29	0.40	7	1.00	0.43	0.60	7
COAD	0.89	0.82	0.85	96	0.86	0.91	0.88	95
DLBC	0.90	1.00	0.95	9	0.90	1.00	0.95	9
ESCA	0.78	0.88	0.83	33	0.91	0.94	0.93	33
GBM	1.00	0.94	0.97	33	1.00	0.97	0.98	33
HNSC	0.97	0.94	0.95	101	0.97	0.98	0.98	101
KICH	0.92	0.85	0.88	13	0.92	0.92	0.92	13
KIRC	0.95	0.96	0.96	107	0.99	0.95	0.97	107
KIRP	0.95	0.91	0.93	58	0.92	0.95	0.93	58
LAML	1.00	1.00	1.00	30	1.00	1.00	1.00	30
LGG	1.00	1.00	1.00	106	1.00	1.00	1.00	106
LIHC	0.97	0.97	0.97	75	0.96	0.99	0.97	75
LUAD	0.95	0.95	0.95	107	0.95	0.93	0.94	107
LUSC	0.90	0.91	0.91	100	0.91	0.93	0.92	100
MESO	1.00	0.94	0.97	17	1.00	1.00	1.00	17
OV	1.00	1.00	1.00	76	0.99	1.00	0.99	76
PAAD	0.97	0.94	0.96	36	0.92	1.00	0.96	36
PCPG	1.00	0.97	0.99	36	1.00	1.00	1.00	36
PRAD	1.00	1.00	1.00	100	1.00	1.00	1.00	100
READ	0.54	0.67	0.59	33	0.64	0.55	0.59	33
SARC	0.80	0.92	0.86	53	0.91	0.98	0.95	53
SKCM	0.99	0.97	0.98	94	1.00	0.99	0.99	94
STAD	0.95	0.53	0.68	75	0.96	0.95	0.95	75
TGCT	0.36	0.96	0.52	28	0.96	0.93	0.95	28
THCA	0.99	0.95	0.97	102	0.99	1.00	1.00	102
THYM	1.00	1.00	1.00	24	1.00	1.00	1.00	24
UCEC	0.99	0.97	0.98	110	0.96	0.99	0.98	110
UCS	1.00	0.73	0.84	11	1.00	0.73	0.84	11
UVM	1.00	1.00	1.00	16	0.94	1.00	0.97	16
Accuracy			0.93	2068			0.96	2068
Marco avg.	0.92	0.91	0.90	2068	0.95	0.94	0.94	2068
Weighted avg.	0.95	0.93	0.93	2068	0.96	0.96	0.96	2068

5 Discussion

This study introduced three CNN-SVM hybrid models for classifying and predicting cancer types using high-dimensional gene expression data. The integration of CNNs and SVMs aimed to combine the feature extraction capabilities of CNNs with the robust classification properties of SVMs, offering an effective solution to the challenges of high-dimensional data analysis. The results demonstrated that

the hybrid approach significantly improved classification accuracy and generalization compared to standalone CNN models.

The 1D-CNN-SVM model processed gene expression data as a one-dimensional vector, leveraging SVM's hinge loss function to optimize classification margins. In contrast, the Vanilla-CNN-SVM and Hybrid-CNN-SVM models utilized two-dimensional input matrices, allowing for a more complex representation of gene expression data. These architectural differences facilitated a comparative analysis of the models' performance across varying input formats. The Hybrid-CNN-SVM model consistently outperformed its counterparts, achieving the highest classification accuracy of 96.00%, compared to 95.00% for the standard Hybrid-CNN model. This improvement underscores the complementary strengths of CNNs and SVMs, where CNNs extract hierarchical features, and SVMs provide robust decision boundaries.

The performance of the models was further enhanced through rigorous optimization of network parameters, including kernel size, stride, and the number of the FC layer nodes. The use of the Adam optimizer, along with grid search-based hyperparameter tuning, ensured efficient training. Additionally, early stopping was employed to mitigate overfitting, particularly for the Vanilla-CNN-SVM and Hybrid-CNN-SVM models. The robustness of the results was validated through 5-fold cross-validation, repeated six times, ensuring reliable performance metrics.

A key strength of the hybrid approach lies in its ability to generalize well to unseen data. The SVM component's use of Structural Risk Minimization (SRM) helped balance training error and model complexity, reducing overfitting—a common limitation of CNNs trained with Empirical Risk Minimization (ERM). Furthermore, the hybrid models demonstrated superior robustness to noise, as the SVM's margin-maximization principle ensured that the decision boundary was less sensitive to outliers. This robustness is particularly critical in gene expression datasets, which are often noisy and high-dimensional. Despite these advantages, the Vanilla-CNN-SVM model did not outperform its standalone CNN counterpart. This suggests that the simplicity of the Vanilla-CNN architecture may not have extracted sufficiently robust features for the SVM classifier to utilize effectively. This finding highlights the importance of feature extraction in hybrid models and suggests that more complex CNN architectures, such as those in the Hybrid-CNN-SVM model, are better suited for integration with SVMs.

The comparative analysis also revealed that the Hybrid-CNN-SVM model strikes an optimal balance between accuracy, generalization, and robustness, making it particularly suitable for complex classification tasks, such as cancer type prediction. However, the computational cost of training hybrid models, especially on large datasets, remains a limitation that warrants further exploration. Future research could focus on optimizing the computational efficiency of these models and extending their applicability to other high-dimensional datasets in genomics and bioinformatics.

6 Conclusion

This study highlights the potential of CNN-SVM hybrid models for classifying cancer types using high-dimensional gene expression data. By combining the feature extraction capabilities of convolutional neural networks (CNNs) alongside the robust classification power of support vector machines (SVMs), our proposed models outperformed both standalone CNN and SVM approaches. The 1D-CNN-SVM and

Hybrid-CNN-SVM models demonstrated significant improvements in accuracy, generalization, and robustness, with the Hybrid-CNN-SVM model achieving the highest classification accuracy of 96.00%. These results emphasize the effectiveness of integrating deep learning and traditional machine learning techniques to address the challenges associated with high-dimensional data analysis. Furthermore, the findings also highlight the critical role of feature extraction in hybrid architectures, as evidenced by the suboptimal performance of the Vanilla-CNN-SVM model. This indicates that careful design and optimization of CNN structures are essential for maximizing the benefits of the CNN-SVM integration. The robustness of the hybrid models to noise and their strong generalization to unseen data make them particularly suitable for complex classification tasks in genomics and related fields. Overall, the proposed CNN-SVM hybrid models represent a promising approach to cancer type prediction and other high-dimensional classification problems. Future research could explore further optimization of the hybrid architecture, including joint optimization of CNN and SVM components, and integrating biologically informed features to enhance both interpretability and performance. By addressing these challenges, the CNN-SVM framework could be further refined to advance Precision medicine and improve the accuracy of cancer diagnostics.

Declarations

Availability of Supporting Data

All data generated or analyzed during this study are included in this published paper.

Funding

The author conducted this research without any funding, grants, or support.

Competing Interests

The author declare that they have no competing interests relevant to the content of this paper.

References

[1] Ajmal, H.B., and Madden, M.G. (2022). “Dynamic Bayesian network learning to infer sparse models from time series Gene expression data”, *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 19(5), 2794-2805, doi:10.1109/TCBB.2021.3092879.

[2] Arowolo, M.O., Adebiyi, M., Adebiyi, A., and Okesola, O. (2020). “PCA model for RNA-seq malaria vector data classification using KNN and decision tree algorithm”, *International Conference in Mathematics, Computer Engineering and Computer Science (ICMCECS)*, 1-8, doi:10.1109/ICMCECS47690.2020.240881.

[3] Barman, S., and Kwon, Y.-K. (2018). "A Boolean network inference from time-series gene expression data using a genetic algorithm", *Bioinformatics*, 34(17), i927-i933, doi:10.1093/bioinformatics/bty584.

[4] Ciriello, G., Gatza, M.L., Beck, A.H., Wilkerson, M.D., Rhie, S.K., Pastore, A., Zhang, H., McLellan, M., Yau, C., Kandoth, C., Bowlby, R., Shen, H., Hayat, S., Fieldhouse, R., Lester, S.C., Tse, G.M., Factor, R.E., Collins, L.C., Allison, K.H., Chen, Y.Y., Jensen, K., Johnson, N.B., Oesterreich, S., Mills, G.B., Cherniack, A.D., Robertson, G., Benz, C., Sander, C., Laird, P.W., Hoadley, K.A., and King, T.A. (2015). TCGA Research Network; "Comprehensive molecular portraits of invasive lobular breast cancer", Perou CM. Cell. 163(2):506-519. PMID: 26451490; PMCID: PMC4603750, doi:10.1016/j.cell.2015.09.033.

[5] Colaprico, A., Silva, T.C., Olsen, C., Garofano, L., Cava, C., Garolini, D., Sabedot, T.S., Malta, T.M., Pagnotta, S.M., Castiglioni, I., Ceccarelli, M., Bontempi, G., and Noushmehr, H. (2016). "TCGAbiolinks: An R/Bioconductor package for integrative analysis of TCGA data". Nucleic Acids Research, 44(8): e71. PMID: 26704973; PMCID: PMC4856967, doi:10.1093/nar/gkv1507.

[6] Huang, Z., Johnson, T.S., Han, Z., Helm, B., Cao, S., Zhang, C., Salama, P., Rizkalla, M., Yu, C.Y., Cheng, J., Xiang, S., Zhan, X., Zhang, J., and Huang, K. (2020). "Deep learning-based cancer survival prognosis from RNA-seq data: Approaches and evaluations", *BMC Medical Genomics*, 13(S5), doi:10.1186/s12920-020-0686-1.

[7] Jiang, X., Zhao, J., Qian, W., Song, W., and Lin, G.N. (2020). "A generative adversarial network model for disease gene prediction with RNA-Seq data", *IEEE Access*, 8, 37352-37360, doi:10.1109/ACCESS.2020.2975585.

[8] Kim, T., Chen, I.R., Lin, Y., Wang, A.Y.-Y., Yang, J. Y., and Yang, P. (2018). "Impact of similarity metrics on single-cell RNA-seq data clustering", *Briefings in Bioinformatics*, 20(6), 2316-2326, doi:10.1093/bib/bby076.

[9] Li, Y., Kang, K., Krahn, J.M., Croutwater, N., Lee, K., Umbach, D.M., and Li, L. (2017). "A comprehensive genomic pan-cancer classification using the cancer genome atlas gene expression data", *BMC Genomics*, 18(1), doi:10.1186/s12864-017-3906-0.

[10] Locati, L.D., Serafini, M.S., Iannò, M.F., Carenzo, A., Orlandi, E., Resteghin, C., Cavalieri, S., Bossi, P., Canevari, S., Licitra, L., and De Cecco, L. (2019). "Mining of self-organizing map gene-expression portraits reveals prognostic stratification of HPV-positive head and neck squamous cell carcinoma", *Cancers*, 11(8), 1057, doi:10.3390/cancers11081057.

[11] Lyu, B., and Haque, A. (2018). "Deep learning based tumor type classification using gene expression data", Proceedings of the 2018 ACM International Conference on Bioinformatics, Computational Biology, and Health Informatics, 89-96, doi:10.1155/2022/4715998.

[12] Ma, R., Yang, G., Xu, R., Liu, X., Zhang, Y., Ma, Y., and Wang, Q. (2019). "Pattern analysis of conditional essentiality (pace)-based heuristic identification of an in vivo colonization determinant as a novel target for the construction of a live attenuated vaccine against Edwardsiella piscicida", *Fish & Shellfish Immunology*, 90, 65-72, doi:10.1016/j.fsi.2019.03.079.

[13] Melana, S.M., Nepomnaschy, I., Hasa, J., Djougarian, A., Djougarian, A., Holland, J.F., and Pogo, B.G. (2010). "Detection of human mammary tumor virus proteins in human breast cancer cells", *Journal of Virological Methods*, 163(1), 157-161, doi:10.1016/j.jviromet.2009.09.015.

[14] Monti, M., Fiorentino, J., Milanetti, E., Gosti, G., and Tartaglia, G.G. (2022). "Prediction of time series gene expression and structural analysis of gene regulatory networks using recurrent neural networks", *Entropy (Basel, Switzerland)*, 24(2), 141, doi:10.3390/e24020141.

[15] Mostavi, M., Chiu, Y.-C., Huang, Y., and Chen, Y. (2020). "Convolutional neural network models for cancer type prediction based on gene expression", *BMC Medical Genomics*, 13(S5), doi:10.1186/s12920-020-0677-2.

[16] Moussa, M., and Măndoiu, I.I. (2018). "Single cell RNA-seq data clustering using TF-IDF based methods", *BMC Genomics*, 19(S6), doi:10.1186/s12864-018-4922-4.

[17] Nandini, D., Capecci, E., Koefoed, L., Laña, I., Shahi, G.K., and Kasabov, N. (2018). "Modelling and analysis of temporal gene expression data using spiking neural networks", *Lecture Notes in Computer Science*, 571-581, doi:10.1007/978-3-030-04167-0_52.

[18] Peng, J., Wang, X., and Shang, X. (2019). "Combining gene ontology with deep neural networks to enhance the clustering of single cell RNA-Seq Data", *BMC Bioinformatics*, 20(S8), doi:10.1186/s12859-019-2769-6.

[19] Ramirez, R., Chiu, Y.-C., Hererra, A., Mostavi, M., Ramirez, J., Chen, Y., Huang, Y., and Jin, Y.-F. (2020). "Classification of cancer types using graph convolutional neural networks", *Frontiers in Physics*, 8, doi:10.3389/fphy.2020.00203.

[20] Stickels, R.R., Murray, E., Kumar, P., Li, J., Marshall, J.L., Di Bella, D., Arlotta, P., Macosko, E.Z., and Chen, F. (2020). "Sensitive spatial genome wide expression profiling at cellular resolution", bioRxiv, 989806, doi:10.1101/2020.03.12.989806.

[21] Tran, K.A., and Kondrashova, O., Bradley, A., Williams, E.D., Pearson, J.V., and Waddell, N. (2021). "Deep learning in cancer diagnosis, prognosis and treatment selection", *Genome Medicine*, 13(1), doi:10.1186/s13073-021-00968-x.

[22] Xiao, Y., Wu, J., Lin, Z., and Zhao, X. (2018). "A semi-supervised deep learning method based on stacked sparse auto-encoder for cancer prediction using RNA-Seq data", *Computer Methods and Programs in Biomedicine*, 166, 99-105, doi:10.1016/j.cmpb.2018.10.004.

[23] Xie, Y., Meng, W.-Y., Li, R.-Z., Wang, Y.-W., Qian, X., Chan, C., Yu, Z.-F., Fan, X.-X., Pan, H.-D., Xie, C., Wu, Q.-B., Yan, P.-Y., Liu, L., Tang, Y.-J., Yao, X.-J., Wang, M.-F., and Leung, E.L.-H. (2021). "Early lung cancer diagnostic biomarker discovery by machine learning methods", *Translational Oncology*, 14(1), 100907, doi:10.1016/j.tranon.2020.100907.