CIT. Journal of Computing and Information Technology, Vol. 32, No. 3, September 2024, 195–215
doi: 10.20532/cit.2024.1005801

Enhancement of Breast Cancer Classification Using Bat Feature Selection with Recurrent Deep Learning

Ali Nafaa Jaafar

Electrical Engineering Technical College, Middle Technical University, Baghdad, Iraq

DNA is a valuable tool for classifying expression of genes in detection of breast cancer. Gene expression data are biological data that extract valuable hidden information from gene datasets. Extracting useful features from datasets is a challenging task. Our gene expression dataset had a small number of samples but many features. This paper compared three types of recurrent deep learning models, including recurrent neural networks (RNN), long short-term memory (LSTM), and gated recurrent unit (GRU), for classification of breast cancer. The goals of the study were to improve the accuracy of classification and to enhance the effectiveness of feature selection; the basic principle was to select the best features from the original datasets. The bat algorithm assists in selecting the best relevant feature when integrated with recurrent deep learning models, which improves breast cancer classification by leveraging training datasets. Data preprocessing involves removing unnecessary columns and filling out missing values with the median value. The result was a comparative study using recurrent deep learning with the bat algorithm to classify breast cancer. The bat algorithm with LSTM achieved higher accuracy than RNN and GRU, where GRU had the lowest accuracy.

ACM CCS (2012) Classification: Computing methodologies → Machine learning → Machine learning approaches → Neural networks

Applied computing → Life and medical sciences → Genomics → Computational genomics

Keywords: RNN, LSTM, GRU, bat algorithm, gene expression, feature selection

1. Introduction

Breast cancer is the most frequent cancer in women worldwide, and it is one of the primary reasons for death in women. There are three different forms of breast cancer: invasive, benign, and in situ. The benign tumors are minor and not harmful, while mammary duct lobule system-specific in situ cancer is curable with early detection. The deadliest kind of carcinoma is invasive because it can spread to other organs [1]. Genomic analysis or histological image analysis are methods to diagnose breast cancer. Genomics is more efficient but less often applied because of its high cost and processing power requirements. The primary distinction between these two approaches is in the information gap that exists between the study of molecular and genetic disease biomarkers; the latter may result in over- or under-treatment because of its impreciseness [2].

Biology research is moving towards the post-genome age with the completion of the Human Genome Project. The precise workings of DNA sequences are still mostly unknown despite biologists having gathered much sequence data. The complexity of genomes is seen even in the most basic organisms [3]. Biology was once a data-poor science, but under increasingly sophisticated methods developed recently, biologists can now turn enormous volumes of valuable biological data into useful data [4]. Numerous techniques have been developed to

comprehend how genes behave. One significant approach for tracking the expression levels of multiple genes at once is microarray technology. Gene expression is defined as the transcription of a gene's DNA sequence into RNA (ribonucleic acid), which acts as a template for the synthesis of proteins. Gene expression level reveals how active a gene is in a given tissue at a particular moment or during a specific experiment. In addition to reflecting the actions of the respective proteins under particular circumstances, the tracked gene expression levels offer a general overview of the genes [5].

Scholars have suggested that nature is an essential wellspring of ideas for the creation of intelligent systems and provides answers to complex issues. The bat algorithm is a recent entry in the nature-inspired metaheuristic optimization algorithms. Bats possess remarkable abilities in echolocation, which has captured the interest of researchers across various disciplines. Echolocation works similarly to sonar in that bats create a loud and transient sound pulse, wait for it to bounce off an object, and then receive the echo in their ears; this enables bats to determine the distance between themselves and an object. Furthermore, their exceptional orientation mechanism allows bats to differentiate between obstacles and prey, allowing them to hunt even in total darkness. The bat algorithm has gained significant popularity and has been successfully employed in diverse applications such as engineering optimization and pattern recognition [6].

The bat algorithm extracts features from the dataset to improve classification accuracy and decrease the number of features; feature selection may be used to choose the most informative features from various tumor datasets. The bat algorithm's primary objective is to improve the effectiveness of the feature selection approach [7].

Technology is advancing quickly in medical diagnosis, and computer-aided diagnosis (CAD) is becoming widely used due to its high speed and accuracy. Breast cancer can be classified as benign or malignant using CAD software. A significant contributor to this advancement is deep learning. The RNN, LSTM, and GRU models are examples of classifiers from recurrent deep learning [8].

The three primary methodologies that underpin breast cancer diagnosis are preprocessing, feature selection and classification. Feature selection is critical in deep learning of cancer diagnosis [9]. In this paper, we implemented the bat algorithm with recurrent deep learning and applied them to training datasets in order to enhance the prediction accuracy of breast cancer classification. In recent years, there has been significant research focused on deep learning. These networks feature complete connections across layers but lack intra-layer connections, making them suitable for processing sequential input with limitations in recalling previous data. RNNs with internal memory were developed to address this, allowing for considering current and previous data. RNNs can lose accuracy over longer spans due to exploding gradients and vanishing issues. The LSTM and GRU networks are an enhanced form of RNNs to overcome these problems [10]. These systems integrate two core mechanisms: states (memory) and gates. The memory cells within can discern when certain information should be forgotten and pinpoint the optimal duration for time lags. Furthermore, gates offer a method of controlling the information flow-through by employing a pointwise multiplication operation after a layer of sigmoid neural networks. LSTM and GRU have demonstrated effectiveness in several applications, including gesture recognition, intrusion detection, handwriting recognition, language translation, speech synthesis, and data analysis. The LSTM stands out for its exceptional classification and feature extraction accuracy. LSTM is used for breast cancer classification by incorporating different types of layers for the classification task [11]. The structure of this paper is set out as follows: Section 2 considers related work, Section 3 concentrates on the specifics of the underlying theory, the dataset is discussed in Section 4, the proposed work is described in Section 5, Section 6 provides the experiment and results, and Section 7 summarizes the conclusions.

2. Related Work

This study [12] investigates the use of artificial intelligence algorithms to classify breast cancer DNA, with a focus on machine learning and deep learning techniques. It involves the ap-

plication of genetic algorithms to identify gene expressions and reduce misclassified cancers. In their research, Omondiagbe *et al.* [13] used the WDBC dataset with naive Bayes, ANN and SVM with radial basis kernel to identify breast cancer with 98.82% accuracy, 98.41% sensitivity, and 99.07% specificity. The objective of study [14] was to develop a method for predicting the recurrence of breast cancer using advanced neural network architectures such as LSTM and GRU, in combination with feature selection methods like logistic regression (LR) and analysis of variance (ANOVA). The models, LR-LSTM and ANOVA-GRU, have demonstrated significant success. In[15], a breast cancer prediction model was developed using an optimized deep learning approach. The model utilized an optimized deep RNN with a Keras tuner. The study [16] used an ANN to detect breast cancer. To evaluate the classifier's performance in various noise levels and ensure its practicality, the authors integrated three loss functions: cross-entropy, hinge, and correntropy. This approach helped determine the most suitable loss function for the ANN-based classifier.

Bhardwaj and Tiwari [17] proposed a technique that combined a wrapper and filter approach to achieve a high classification rate. The method included a preprocessing step to improve the effectiveness of the search for the best features. It proved advantageous in addressing the problem of overfitting and avoiding getting stuck in a locally optimal solution. To minimize the adverse effects of mutation and crossover operators, the researchers introduced a new method for detecting breast cancer, called enhanced ANN-based method. The approach utilized a genetically optimized neural network to distinguish between malignant and benign tumors accurately. Compared to previous methods, the genetically optimized neural network-based approach achieved a significantly improved classification accuracy rate of 13.56%. In a study [18], the Wisconsin dataset was used to achieve a classification accuracy of 99.25% for breast cancer detection. The study tested and confirmed the false alarm detecting rate of the backpropagation technique based on the feed-forward benefits of an ANN. The ANN model was trained without noise to assess the degree of roughness. The proposed mechanism resulted in a 23.4% reduction in the false alarm rate compared to other ANN schemes. Moreover, this focused approach to breast cancer identification based on the false alarm rate of the ANN improved the classification accuracy by 14.3%.

In a research study [19], two commonly used machine learning algorithms, multilayer perceptron (MLP) and convolutional neural network (CNN) were utilized to develop a model for detecting and diagnosing malignancy in breast cells. The study concluded that the CNN algorithm outperformed the MLP algorithm in accuracy. In [20], a study conducted a thorough experiment to determine which factors in the breast cancer dataset were the least significant. The study covered machine learning techniques such as decision trees, random forests, naive Bayes, logistic regression, k-nearest neighbor, neural networks, and SVM. The findings showed that SVM produced a precision score of 0.95 with fifteen features, while naive Bayes and random forest produced a promising accuracy score of 0.94 with thirty features. Verma *et al.* [21] proposed implementing a neural network-based breast cancer management system and decision tree. This system is called the transparent breast cancer management system with P-rules (TBCMS-PR); it used datasets from the UCI library and machine learning algorithms. Dasgupta *et al.* [22] experimented with feature selection using datasets related to breast cancer. The study aimed to develop a model for cancer diagnosis and evaluate its accuracy. For this purpose, they utilized various techniques such as ANN, Bayesian network, random forest, and decision tree. The study compared different methods to determine the most accurate algorithm for predicting cancer type.

A unified deep learning architecture was presented by [23] to learn features from images for breast cancer classification automatically. The system's architecture uses pointwise gated Boltzmann machines (PGBM). Researchers also used CNNs to analyze mammograms to detect breast cancer. A study [24] utilized LSTM and CNN-based semantic features to classify mammography images for breast cancer detection. The performance of the suggested model was evaluated using classification accuracy and loss rate.

3. Theoretical Background

This section describes the recurrent deep learning algorithms and feature selection by the bat algorithm for classification of breast cancer.

3.1. Deep Learning

Deep learning falls under the umbrella of machine learning in artificial intelligence, employing artificial neural networks that independently learn from unstructured data. The learning process can be supervised or unsupervised; Figure 1 shows several algorithms of the machine learning model. A deep learning method can learn from large amounts of unstructured data that would take humans a long time to process and understand. Deep learning levels learn to convert incoming data into more abstract and complicated representations [25]. The progress made in machine learning has created a notable opportunity for applying deep learning models in disease prediction and breast cancer classification. Deep learning has demonstrated its effectiveness in solving problems related to image processing, classification, and pattern recognition. With the abundance of publicly available data from microarray gene expression and RNA-Seq, deep learning is becoming crucial in identifying specific patterns within large gene expression datasets. Classifying cancer cells based on gene expression levels continues to present a substantial challenge; unsupervised recurrent deep-learning techniques are employed to address this issue [26].

3.1.1. Recurrent Neural Network

RNN excels at analyzing gene expression because each neuron has an internal memory that allows it to retain information about prior inputs. RNNs are often more suitable for tasks involving sequential inputs like speech, natural phenomena, and DNA sequences [27]. RNNs are capable of processing sequential data of varying lengths without the need for a predetermined input size, making them effective for analyzing gene expression patterns across diverse experimental conditions. However, it's important to note that RNNs face problems, including the risk of vanishing and exploding gradients during training, which demand careful management during model development [28]. The RNNs are referred to as recurrent because they do the same task for each sequence element with the output dependent on past calculations. The RNN architecture consists of essential components: an input, hidden, and output layer for processing sequential data and revealing hidden dependencies, as shown in Figure 2 [29]

- **Input Layer**: RNN starts by accepting a sequence of data as a vector, this means the RNN considers two types of inputs: the present input $x(t)$ and the input derived from previous computations $h(t)$ [28];
- **Hidden Layer**: This crucial layer processes the data in sequence, continuously updating a hidden state $h(t)$ that incorporates information from both the previous step $h(t-1)$ and the current input $x(t)$, with an activation function adjusting the inputs

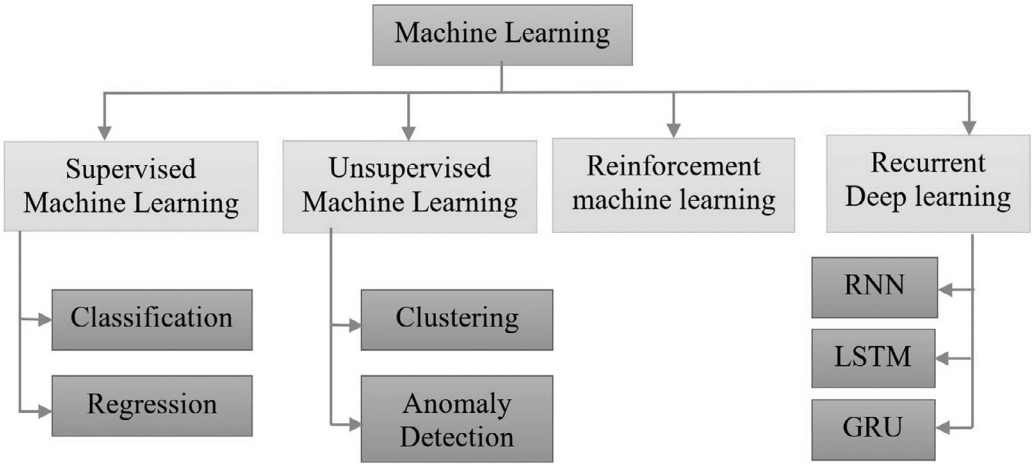

Figure 1. Types of machine learning.

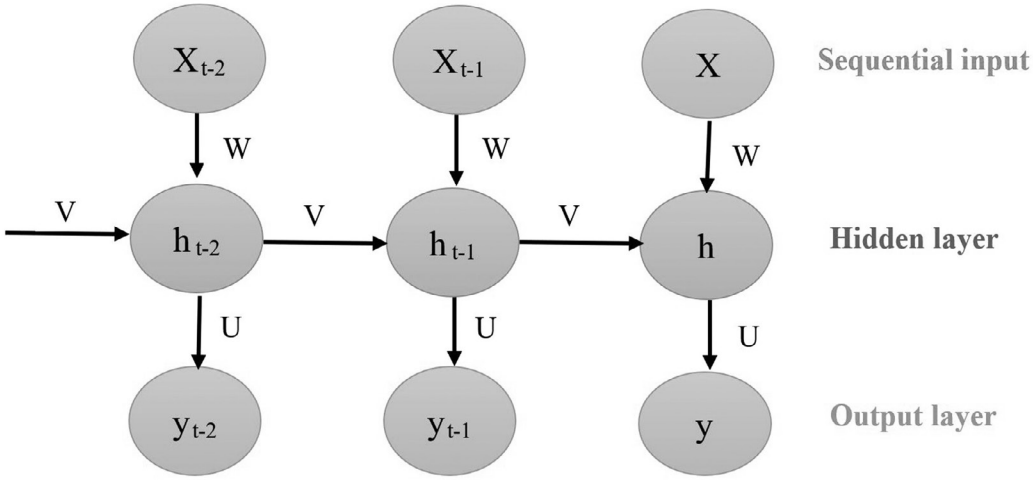

Figure 2. Diagram of RNN architecture.

shown in Equation (1). The function f is considered a non-linear transformation, like the hyperbolic tangent (tanh) function shown in Equation (2) or the Rectified Linear Unit (ReLU) function shown in Equation (3). These weights (W, V, U) are shared with (input-to-hidden, hidden-to-hidden, hidden-to-output).

- **Output Layer**: The processed data from the hidden layer is passed to the output layer, which produces the final output. The network's output represents the $y(t)$ shown in Equation (4). After the output is generated, a loss function calculates the difference between the predicted and actual results. The error is then utilized to adjust the weights in the hidden layer, improving the effectiveness of the model. Consequently, the modeling approach will focus on optimizing two important RNN hyperparameters: the number of hidden layers and the number of neurons [30], [31].

$$h(t) = F(w\,x(t) + v\,h(t-1)) \tag{1}$$

$$\mathrm{Tanh}(x) = (e^{x} - e^{-x})/(e^{x} + e^{-x}) \tag{2}$$

$$\mathrm{ReLU}(x) = \max(0, x) \tag{3}$$

$$y(t) = g(U\,h(t)) \tag{4}$$

3.1.2. LSTM (Long Short-Term Memory)

LSTMs are an advanced version of the RNNs, created to solve the vanishing gradient problem by integrating a dedicated memory unit. Their design features memories and gates that help them effectively learn and maintain long-term dependencies, with gates that selectively include or exclude information from a cell [32]. LSTM comprises three types of gates: input, forget, and output. Each gate includes a sigmoid neural network layer and a pointwise multiplication operation, as shown in Figure 3. The LSTM unit makes decisions based on the current input X_t, previous output h_{t-1}, and stored memory C_{t-1}, generating new output ht and updating its memory C_t, the key elements of the LSTM cell are [33]:

- **In the memory cell and forget gate**: The input gate controls the input activation into the memory cell, the output gate regulates the output flow from cell activation into the network, and the amount of memory passed to the next LSTM unit is determined by the sigmoid layer of this gate [34]. The information no longer needed is removed from the cell state by the sigmoid layer output. It evaluates the function based on the previous state h_{t-1} and the current input x_t. W_f denotes the weight vector, and b_f denotes the bias of the forget gate layer. The output of the forget gate layer is represented as f_t, which is depicted in Equation (5).

$$f_t = \sigma(W_f \cdot [h_{t-1}, X_t] + b_f) \tag{5}$$

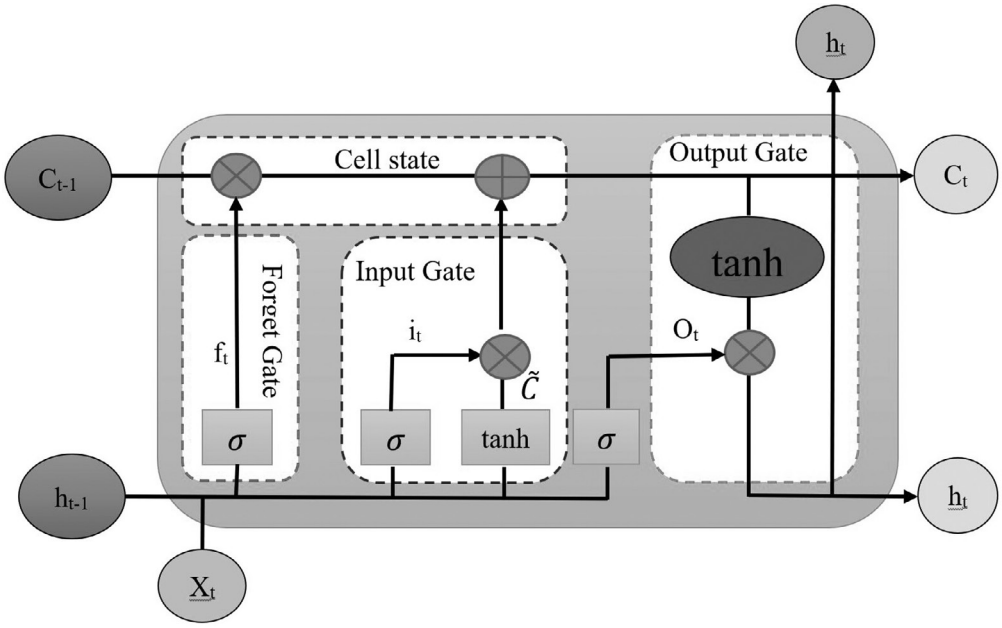

Figure 3. An illustration of the architecture of the LSTM.

In the input gate and candidate gate: The input gate comprises two sections: the sigmoid and the tanh function. The sigmoid layer determines which values to update. The tanh layer generates a vector of fresh candidate values for the state. These elements are combined in the following step to update the state, as detailed in Equations (6) and (7). The previous cell state (C_{t-1}) is replaced with the new cell state (C_t). The old state is multiplied by f_t to forget the unnecessary information, and the candidate $\tilde{C}_t * i_t$ is added as in Equation (8).

$$i_t = \sigma\,(W_i \cdot [X_t, h_{t-1}] + b_i) \tag{6}$$

$$\tilde{C} = \tanh(W_c \cdot [X_t, h_{t-1}] + b_c) \tag{7}$$

$$C_t = f_t * C_{t-1} + \tilde{C}_t * i_t \tag{8}$$

- **In the output gate**: The output is calculated in two steps: first, a sigmoid layer is used to pick the relevant parts of the cell state that will be expressed in the output; second, the cell state is passed on via tanh (to normalize values between −1 and 1) and multiplied by the output of the sigmoid gate. The output gate layer represents the current LSTM blocks output, shown in Equation (9) and (10).

$$O_t = \sigma(W_o[h_{t-1}, X_t] + b_o) \tag{9}$$

$$h_t = O_t * \tanh(C_t) \tag{10}$$

The three gates learn to select which information to retain in memory. By organizing memory cells into blocks that share gates, the system efficiently reduces the total number of adjustable parameters [35–37]. Initial modeling results indicated that the tanh activation function performed significantly better than ReLU and sigmoid in LSTM models [28].

3.1.3. Gated Recurrent Unit

GRU is a simplified version of LSTM and represents the advanced version of the RNN, it is designed to address the vanishing gradient problem. GRUs have demonstrated their effectiveness in various applications involving gene expression analysis, modeling complex non-

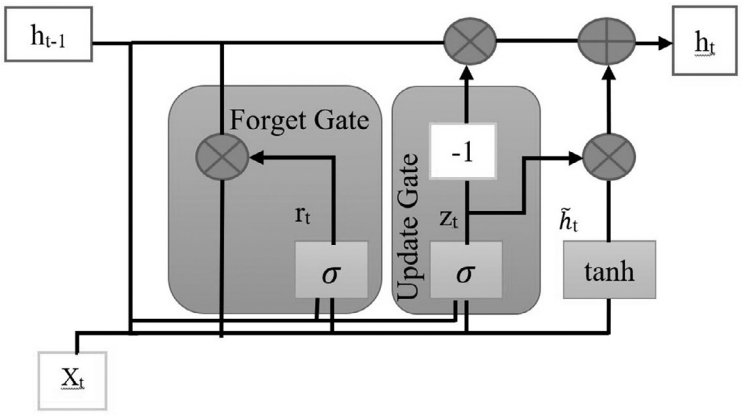

Figure 4. An illustration of the architecture of GRU.

linear relationships and extracting key features from time series. GRU is constructed without a cell state and contains only two gates (update and forget (reset)). These gates are made up of a sigmoid layer and a pointwise multiplication action, depicted by two vectors that generate values within the [0, 1] range, if the reset gate's output equals zero then overlooks the stored memory information. Figure 4 shows the GRU architecture [38], [39].

The forget gate r_t regulates combining the current input (x_t) with the existing memory (h_{t-1}) shown in Equation (11).

$$r_t = \sigma(W_r \cdot [h_{t-1}, x_t]) \tag{11}$$

The update gate employs a sigmoid function to proportionally update the state h_t using a newly computed state $\tilde{h}_t$, shown in Equations (12–14) [40].

$$z_t = \sigma\,(W_z \cdot [h_{t-1}, x_t]) \tag{12}$$

$$\tilde{h}_t = \tanh(W \cdot [r_t * h_{t-1}, x_t]) \tag{13}$$

$$h_t = (1 - z_t) * h_{t-1} + z_t * \tilde{h}_t \tag{14}$$

3.2. Feature Selection

The analysis of Gene expression data involves various stages. Firstly, the data is preprocessed by utilizing feature selection methods, eliminating noisy and redundant features and leaving only the informative ones. Afterwards, a recurrent deep learning algorithm is trained using the bat algorithm to detect cancer subtypes. This algorithm is trained on a generated subset of features, as shown in Figure 5 [41].

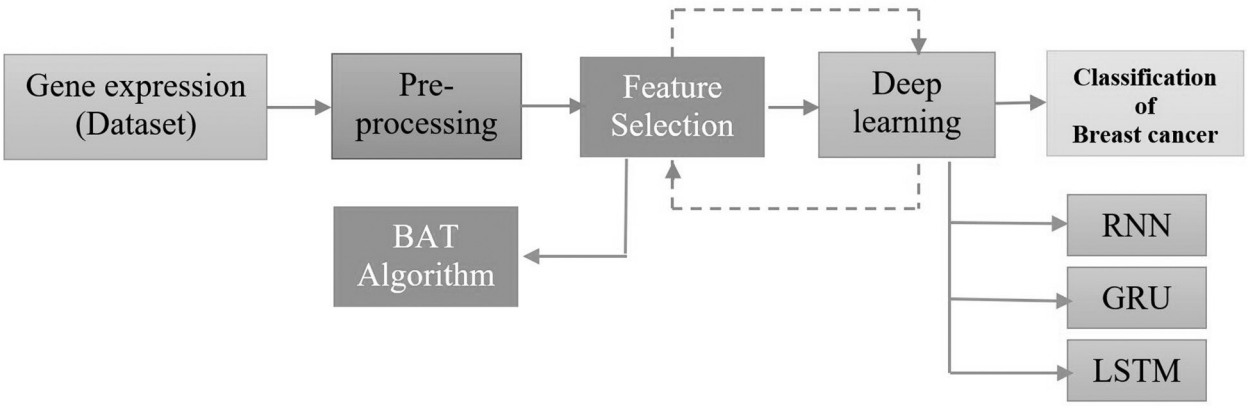

Figure 5. A flowchart of breast cancer classification.

Feature selection involves identifying factors that influence predictions, either automatically or manually:

Reduce overfitting: It is necessary to eliminate noisy and redundant data that hinder the model's ability to generalize effectively from the training data to unseen data.

- Improve accuracy: The accuracy can be improved by training the model with fewer misleading data.
- Reduce training time: Less computation time is required for training with fewer features.
- Provide biologists with an understanding of the relationship between gene signatures and diseases [42].

One of the four feature selection techniques can be used: embedded, filters, hybrid, and wrappers, as shown in Figure 6. However, using more than one feature selection method can increase computational costs. Each technique has advantages and disadvantages to maximize performance, and it is essential to ensure diversity while making the feature selection process more frequent [43]. Table 1 presents the advantages and disadvantages of different feature selection methods [42].

3.3. Bat Algorithm

The bat algorithm draws inspiration from bats' echolocation process for sensing distances. The bats send out short, powerful sound waves and listen for echoes reflecting off barriers or potential prey. Bats have a unique auditory system that allows them to determine the size and location of objects. This echolocation feature of bats was the basis for the bat algorithm, as proposed by Yang [44]. The algorithm of the bat is outlined in Algorithm 1.

Table 1. Advantages and disadvantages of different feature selection methods.

Type	Advantages	Disadvantages
Filter	• Independence from any specific algorithm. • Simple and quick on the computation.	• Does not take into account an interaction with a classifier. • Low performance.
Wrapper	• Always chooses a nearly perfect subset. • The error rate is less than that of alternative methods.	• There is a greater risk of overfitting compared to filter methods. • It is quite computationally demanding in comparison to other techniques. • They are intended for the specific learning machine that has undergone testing.
Embedded	• Less computationally demanding than wrapper techniques. • Comprises the way interaction along with the classification model is used. • Eliminates a need to divide the training data into a training and validation set, making better use of the given data. • Finds a solution faster by reducing the need to train the predictor for each variable subset examined thoroughly.	• Specific to a learning machine. • Higher risk of overfitting than filter methods.
Hybrid	• Bring together the benefits of multiple strategies.	• The complexity of time might increase.

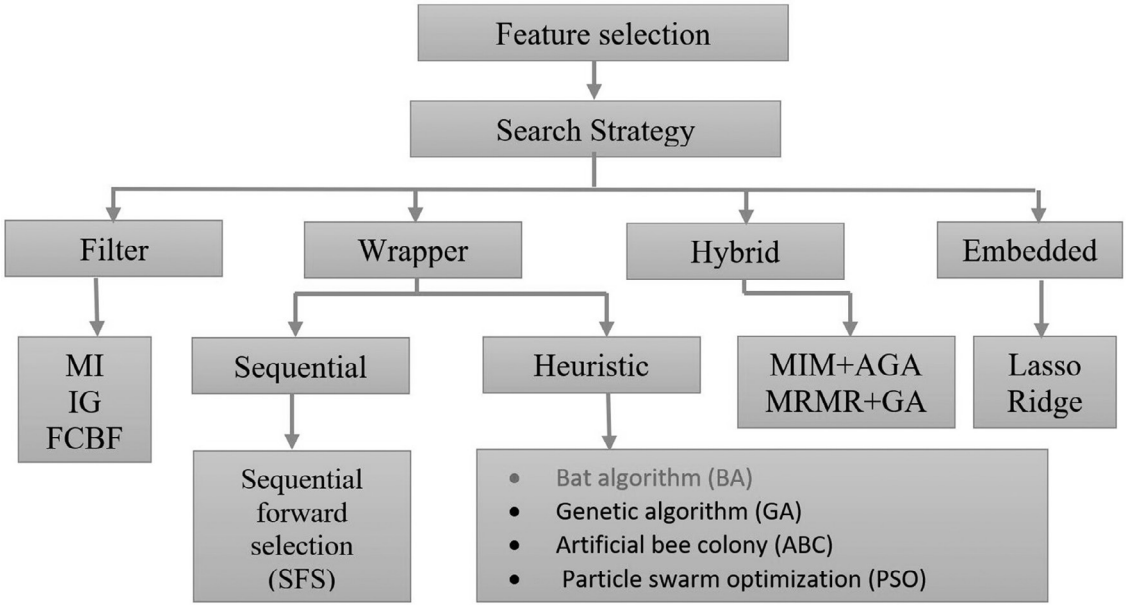

Figure 6. Kinds of feature selection.

Algorithm 1. The bat algorithm.

Step 1: Set up the initial parameters for the bat algorithm, as illustrated in Table 2.

Step 2: Modify the *i*-th bat's global best position X*, along with its pulse frequency, velocity, and position, as detailed below:

$$f_i = f_{\min} + (f_{\max} - f_{\min})\beta, \quad \beta \in [0, 1]$$

$$V_i^{t+1} = V_i^t + (X_i^t + X^*) f_i$$

$$X_i^{t+1} = X_i^t + V_i^t$$

In this formula, V_i^t and X_i^t refer to the velocity and position at the time t, respectively. V_i^{t+1} and X_i^{t+1} indicate the velocity and position at the subsequent time point, $t + 1$. Additionally, β represents a random value selected from a range between 0 and 1.

Step 3: When the random number exceeds r_i, a new solution for the bat is calculated using the equation below:

$$X_{\text{new}} = X_{\text{old}} + \varepsilon A^t$$

In this case, ε symbolizes a random number that falls within the range of [−1, 1], and A^t denotes the mean loudness of all bats at the time t.

Step 4: When the random number is smaller than A_i and $f(X_i)$ is lesser than $f(X^*)$, the new solution gets approved. Subsequently, A_i and r_i are revised in the following way:

$$A_i^{t+1} = \propto A_i^t$$

$$r_i^t = r_i^o\,[1 - e^{-\gamma t}]$$

Here, A_i^{t+1} and A_i^t refer to the loudness levels at time instances t and $t + 1$, consecutively. The terms r_i^o and r_i^t denote the initial pulse rate and the pulse rate at time t, consecutively. α is defined as a constant parameter within the range of [0, 1], and γ is also a constant parameter with a value greater than 0. As t approaches infinity, A_i^t approaches 0, and r_i^t approaches the initial value r_i^o.

Step 5: Organize the bats in order of their fitness levels and determine the current best solution, denoted as X^*.

Step 6: Go back to Step 2 and continue until the maximum iteration count is achieved; then, output the globally best solution [6], [45].

Table 2. Group of parameters utilized in the bat algorithm.

Parameter symbol	Description	Value
M	Represents the total number of bats in the population.	30
I	Identifies an individual bat, with a value from 1 to M.	1 to 30
r_i	The pulse rate of the i-th bat.	randomly
X_i	The position of the i-th bat.	best DNA feature
α	A fixed parameter within the [0, 1] range, used for adjusting A_i (the loudness).	0.9
f_i	The i-th bat pulse frequency is within f_{max} and f_{min} range.	$f_{min} = 0$ and $f_{max} = 2$
X^*	The optimal global location or solution at present.	high accuracy
N	The maximum iterations allowed.	100
A_i	The loudness level of the i-th bat.	initialized to 1
$f(X)$	The fitness function.	initialized to 0
Γ	A stable parameter also in the [0, 1] range, employed for modifying r_i (the pulse rate).	0.9
V_i	The velocity of the i-th bat.	initialized to 0

In this paper, the bat algorithm is used to select the best 10 features out of a total of 70 features. Each bat in the algorithm stands for a potential solution, aligning with a specific feature set. The algorithm modifies each bat's position to represent the selected features by adjusting frequency and velocity parameters, guided by the most effective solutions found during the search. The quality of each feature set is evaluated through a fitness function that checks the classification performance of recurrent deep learning (RNN, LSTM, GRU). The 10 features that achieve the highest classification accuracy are chosen and this iterative process repeats until the maximum accuracy is achieved.

4. Gene Expression Data for Breast Cancer

The analyzed dataset consists of 72 columns. The first column includes the sample_number (id), columns 2–71 represent the input features and column 72 indicates the class label. Table 3 presents gene expression data from a DNA dataset for breast cancer patients. It includes columns for each gene (G5, FGF18, G6, and G7) and a 'Label' column that indicates prognosis (0 for good, 1 for poor), this information is crucial for cancer prognosis. Microarray analysis was used to evaluate the previously generated prognostic profile of 70 genes. This dataset found 295 patients with primary breast cancer who had a gene-expression pattern that was associated with a poor or excellent prognosis. All patients were younger than 53 years old and had stage I or II breast cancer; 151 had lymph node-negative disease, and 144 had lymph node-positive disease. The prognostic profile's predictive ability was assessed using multivariable and univariable statistical techniques. From the group of 295 patients, 82 presented a prognostic signature that indicated a poor prognosis, while 115 received a positive prognosis, with mean (SE) overall 10-year survival rates of 54.6 ± 4.4% and the 94.5 ± 2.6%, respectively. Ten years later, the group with a poor prognosis signature had a 50.6 ± 4.5% likelihood of still being free of distant metastases, while the group with a good prognosis signature had an 85.2 ± 4.3% chance. When comparing the group with a poor prognosis signature to the group with a favorable prognosis signature, the calculated hazard ratio for distant metastases was 5.1 (95% confidence range, 2.9–9.0; $P < 0.001$) when the groups were examined based on the status of their lymph nodes, this ratio continued to be significant. An investigation of multivariable Cox regression revealed that the prognostic profile was a highly reliable independent predictor of the course of the disease [46].

Table 3. A sample from the gene expression data.

Id	Input features (contains only 70 features)						Class Label
	…	G5	FGF18	G6	G7	…	
127	…	–0.026	–0.425	0.204	0.016	…	0
245	…	0.084	–0.303	0.234	–0.459	…	0
247	…	–0.218	–0.148	0.164	0.166	…	0
251	…	–0.124	–0.185	0.192	–0.001	…	0
254	…	–0.132	–0.188	–0.145	0.058	…	0
258	…	0.609	0.146	–0.161	0.202	…	0
260	…	–0.093	–0.072	–0.313	0.006	…	0
345	…	–0.386	–0.256	–0.255	–0.068	…	1

5. Proposed System Architecture

This section will explain the approach of the proposed model. First, we loaded the dataset and cleaned it by removing an unnecessary column and filling in any missing values with the median value. We used the bat algorithm with recurrent deep learning methods such as RNN, LSTM, and GRU to classify the gene expression data. Algorithm 2 illustrates the mechanism of work of these techniques.

This study employs the bat algorithm to optimize the feature selection process, further refining the model's potential to differentiate between various types of breast cancer cells. The bat algorithm is leveraged with GRU, RNN, and LSTM models to enhance the accuracy of breast cancer classification from gene expression data. The block diagram for classification with the best feature is illustrated in Figure 7.

Algorithm 2. The proposed approach.

Input: Breast cancer gene expression data.

Output: Classify the dataset

Step 1: Read the Dataset: Load the gene expression data for breast cancer.

Step 2: Preprocessing

- Clean the dataset by removing unnecessary columns.
- Impute missing values with median values.

Step 3: Dataset Splitting: split the dataset into an 80% training and the 20% testing using cross-validation methods.

Step 4: Classification Using Bat Optimization with GRU, RNN, or LSTM

- Generate an initial population, where each individual represents a potential solution, and each 'cell' in an individual corresponds to a column in the dataset.
- Define the Bat algorithm parameters, including the frequency, velocity, and loudness of bats (individual solutions).
- Compute the fitness of each bat based on the classification accuracy achieved by GRU, RNN, or LSTM models. This accuracy serves as the objective function.
- Identify the best local and global solutions based on their fitness values.
- Iteratively update the solutions using the Bat algorithm:
    - For each bat, update its velocity based on its relation to the best local and global solutions.
    - Normalize the velocity to ensure controlled exploration of the solution space.
    - Update the position of each bat based on its new velocity.
- Compute the fitness of each updated bat.
- Identify the best local solution for the new population.
- Update the global best solution if a better solution is found.
- Repeat the process for a specified number of iterations.

Step 5: Return the best global solution, which represents the optimal set of features for breast cancer classification.

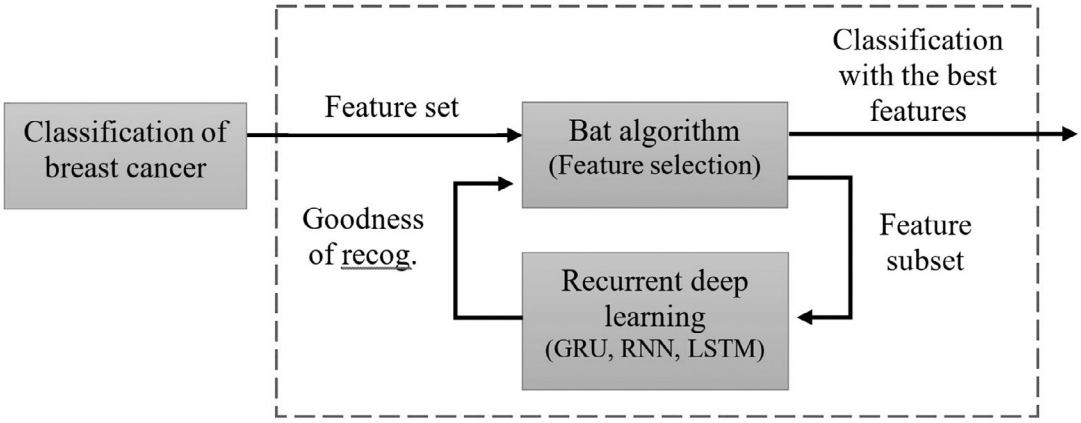

Figure 7. Block diagram of the classification and feature selection process.

6. Results

This section displays the findings of a breast cancer diagnosis utilizing gene expression data with approaches (recurrent deep learning combined with the bat algorithm) implemented in MATLAB® R2021a. The bat algorithm with GRU, RNN, and LSTM classifiers allowed faster generation and comparison of results. The first step involved collecting an entire DNA dataset, and then the cleaning process and filling of the missing values were conducted. We used a variety of recurrent deep learning approaches with the bat algorithm to classify gene expression data and compare methods. The data is divided into two sets: 80% is utilized to train the models (GRU, RNN, and LSTM), with the remaining 20% used for testing. The bat algorithm improves the performance of GRU, RNN, and LSTM models by adjusting hyperparameters like the number of layers and learning rate. It assesses these adjustments based on accuracy to identify the best settings, continuously refining solutions by iteratively updating position and velocity. The dataset dimensions for the proposed RNN model were organized as 20*1*70. The RNN was configured with a learning rate of 0.01 and incorporated techniques such as backpropagation through time for efficient training. The network architecture comprised multiple recurrent layers, each designed to capture different aspects of the sequence data. Regularization with a weight decay of $5 \times 10{-}4$ was applied to prevent overfitting, and a dropout ratio of 0.5 was used to enhance generalization. The LSTM and GRU models utilized similar learning rates and regularization parameters but differed in their internal architectures. LSTMs incorporated cell states and gates (input, output, and forget gates) to control the flow of information, while GRUs used to update and reset gates for a similar purpose but with a simpler structure. Training for these models was conducted over 500 epochs with an initial learning rate of 0.01 and adjusted based on the performance. The training process was stopped when the models reached the optimal state, as depicted in Table 4. GRU, RNN, and LSTM are using sequential processing to improve the accuracy of breast cancer classification from gene expression data. In this context, the bat algorithm's job was to optimize feature selection and boost the model's performance in identifying different breast cancer cells.

After setting up the network architecture as outlined in Table 4, the GRU, RNN, and LSTM model training commenced. In the GRU model at epoch one, the elapsed time was six seconds, the mini-batch accuracy was 51.47%, and the mini-batch loss was 0.6891. The RNN model showed a mini-batch accuracy of 50.89% and a mini-batch loss of 0.6922 at the same epoch with an identical elapsed time of six seconds. The LSTM model started slightly slower with an elapsed time of seven seconds, but with a slightly higher mini-batch accuracy of 52.36% and a mini-batch loss of 0.6875.

Table 4. The parameters of the GRU, RNN, and LSTM models.

Model	Parameter				Value			
GRU	Input layer (dimensions)				20*1*70			
	Number of layers				4			
	Neurons in layer 1	32	Neurons in layer 2	64	Neurons in layer 3	128	Neurons in layer 4	64
	Learning rate				0.005			
	Epochs				150			
	Batch size				32			
	Output layer				cancer or normal			
RNN	Input layer (dimensions)				20*1*70			
	Number of layers				3			
	Neurons in layer 1	32	Neurons in layer2		64	Neurons in layer 3	32	
	Learning rate				0.01			
	Epochs				120			
	Batch size				16			
	Output layer				cancer or normal			
LSTM	Input layer (dimensions)				20*1*70			
	Number of layers				5			
	Neurons in layer 1	32	Neurons in layer 2	64	Neurons in layer 3	128	Neurons in layer 4	64	Neurons in layer 5	32
	Learning rate				0.003			
	Epochs				200			
	Batch size				64			
	Output layer				cancer or normal			

Significant improvements were noticed at epoch 100; the GRU model reached a mini-batch accuracy of 97.85% with a loss of 0.1543 in one minute, the RNN model achieved a mini-batch accuracy of 97.43% with a loss of 0.1589 in the same duration, and the LSTM model attained a mini-batch accuracy of 98.22% with a loss of 0.1466 in one minute and ten seconds. As the training progressed to epoch 500, the GRU model recorded a stellar mini-batch accuracy of 99.89% with a loss of 0.0214 at five minutes; the RNN model followed closely with a mini-batch accuracy of 99.83% and a loss of 0.0247, and the LSTM model topped the accuracy at 99.93% with the lowest loss of 0.0189 in five minutes and fifty seconds, as depicted in Table 5 and Figure 8.

Table 5. Overview of the training phase for the GRU, RNN, and LSTM.

Epoch	Model	Time Elapsed (hh:mm:ss)	Mini-batch Accuracy	Mini-batch Loss	Base Learning Rate
1	GRU	00:00:06	51.47%	0.6891	0.005
1	RNN	00:00:06	50.89%	0.6922	0.01
1	LSTM	00:00:07	52.36%	0.6875	0.003
50	GRU	00:00:30	92.14%	0.3072	0.005
50	RNN	00:00:30	91.78%	0.3120	0.01
50	LSTM	00:00:35	93.07%	0.3011	0.003
100	GRU	00:01:00	97.85%	0.1543	0.005
100	RNN	00:01:00	97.43%	0.1589	0.01
100	LSTM	00:01:10	98.22%	0.1466	0.003
150	GRU	00:01:30	98.96%	0.0954	0.005
150	RNN	00:01:30	98.72%	0.0991	0.01
150	LSTM	00:01:45	99.37%	0.0882	0.003
...	...	...	...	...	...
500	GRU	00:05:00	99.89%	0.0214	0.005
500	RNN	00:05:00	99.83%	0.0247	0.01
500	LSTM	00:05:50	99.93%	0.0189	0.003

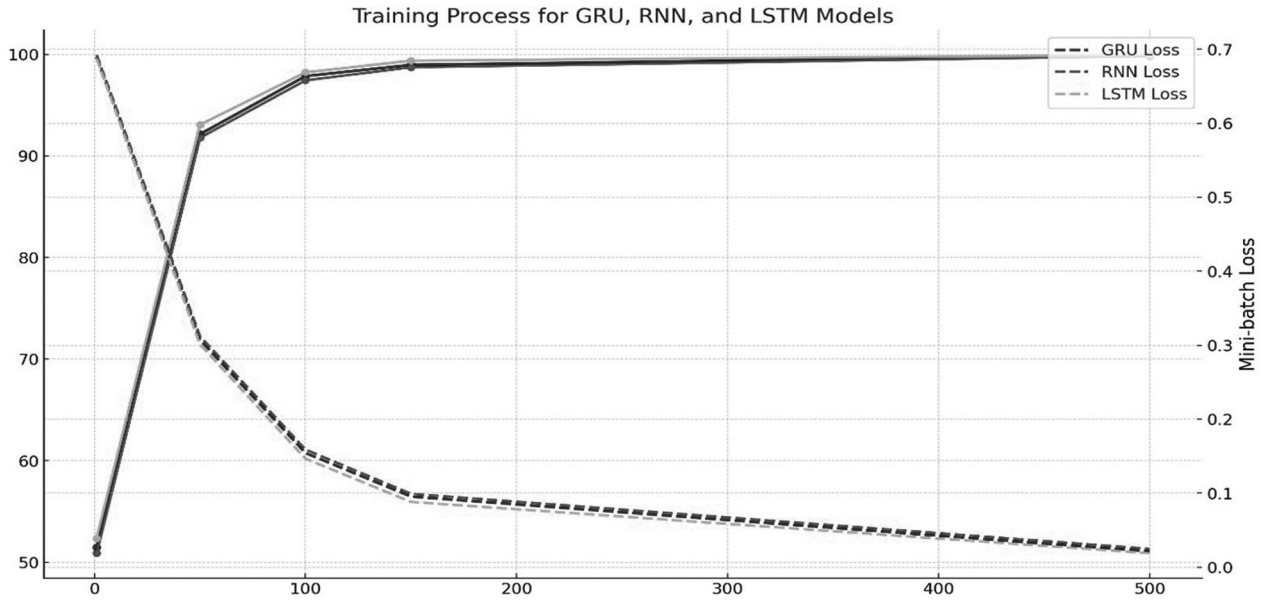

Figure 8. Plots of the training progress of the proposed GRU, RNN, and LSTM.

After the previously described algorithms (GRU, RNN, and LSTM) were applied separately, they were combined with the nat algorithm to achieve better results; we selected the columns in the classification procedure that worked effectively. As was previously mentioned, the gene expression data consisted of 72 columns (one sample number (id), seventy attributes, and one label); the bat algorithm chose the ten best columns. The procedure began by initializing 100 particles, each representing a potential solution with ten attributes from the dataset. The fitness of each particle was assessed based on its effectiveness in the classification task. A crossover event between two randomly chosen particles was performed through a single-point crossover, ensuring diversity among solutions. Mutation was also employed to replace repeating attributes with unique ones, preventing redundancy in the selected features. This iterative crossover and mutation process was repeated for 100 generations, enhancing the particles' feature selection capabilities. The particles' effectiveness was gauged by their accuracy in classifying the dataset after applying the hybrid models: Bat-GRU, Bat-RNN, and Bat-LSTM. The performance results indicated substantial improvements in accuracy in the true positive rate (TPR), and precision with a decrease in false positive rate (FPR) and false negative rate (FNR), underscoring the robustness of the hybrid models in handling the breast cancer classification task. The detailed outcomes of this hybrid approach are summarized in Table 6.

As depicted in Table 6, the hybrid algorithm, combining bat with GRU, RNN, or LSTM achieved better accuracy rates of 0.879, 0.918, and 0.964, respectively, when compared to using only the individual recurrent algorithms. Figure 9 is a visual representation of a confusion matrix that outlines the performance of the proposed method, which employs a hybrid approach combining the bat algorithm with GRU, RNN, or LSTM models for the classification of breast cancer. Each cell in the confusion matrix corresponds to the counts of true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN) for each model, providing a clear depiction of their predictive capabilities.

The performance of the models was evaluated using different data split ratios: 90-10-10, 80-10-10, 70-15-15, 60-20-20, and 50-25-25. The analysis showed that the 80-10-10 split provided the highest accuracy, especially for the hybrid bat-LSTM model, which achieved an accuracy of 0.964. This result highlights the importance of choosing the right data split ratio to optimize model performance, as depicted in Table 7.

Table 6. Confusion matrix for the hybrid algorithm, combining bat with GRU, RNN, or LSTM.

Metric	Hybrid bat-GRU	Hybrid bat-RNN	Hybrid bat-LSTM	GRU	RNN	LSTM
TP (true positives)	163	169	178	135	145	158
TN (true negatives)	108	115	121	99	105	111
FP (false positives)	19	14	6	48	38	26
FN (false negatives)	7	3	1	34	17	10
ACC (accuracy)	0.879	0.918	0.964	0.701	0.806	0.864
TPR (true positive rate)	0.929	0.975	0.994	0.729	0.895	0.940
FPR (false positive rate)	0.149	0.109	0.047	0.327	0.279	0.196
TNR (true negative rate)	0.850	0.890	0.952	0.672	0.720	0.803
FNR (false negative rate)	0.070	0.025	0.0056	0.270	0.104	0.059
Precision	0.895	0.923	0.967	0.737	0.792	0.858

In our paper involving the deep learning algorithms, we have found that, when combining advanced techniques with the custom bat algorithm, the hybrid approach yields superior results compared to traditional machine learning methods that use genetic algorithms, as seen in previous studies [12]. This suggests that integrating innovative algorithms with deep learning can significantly enhance classification performance.

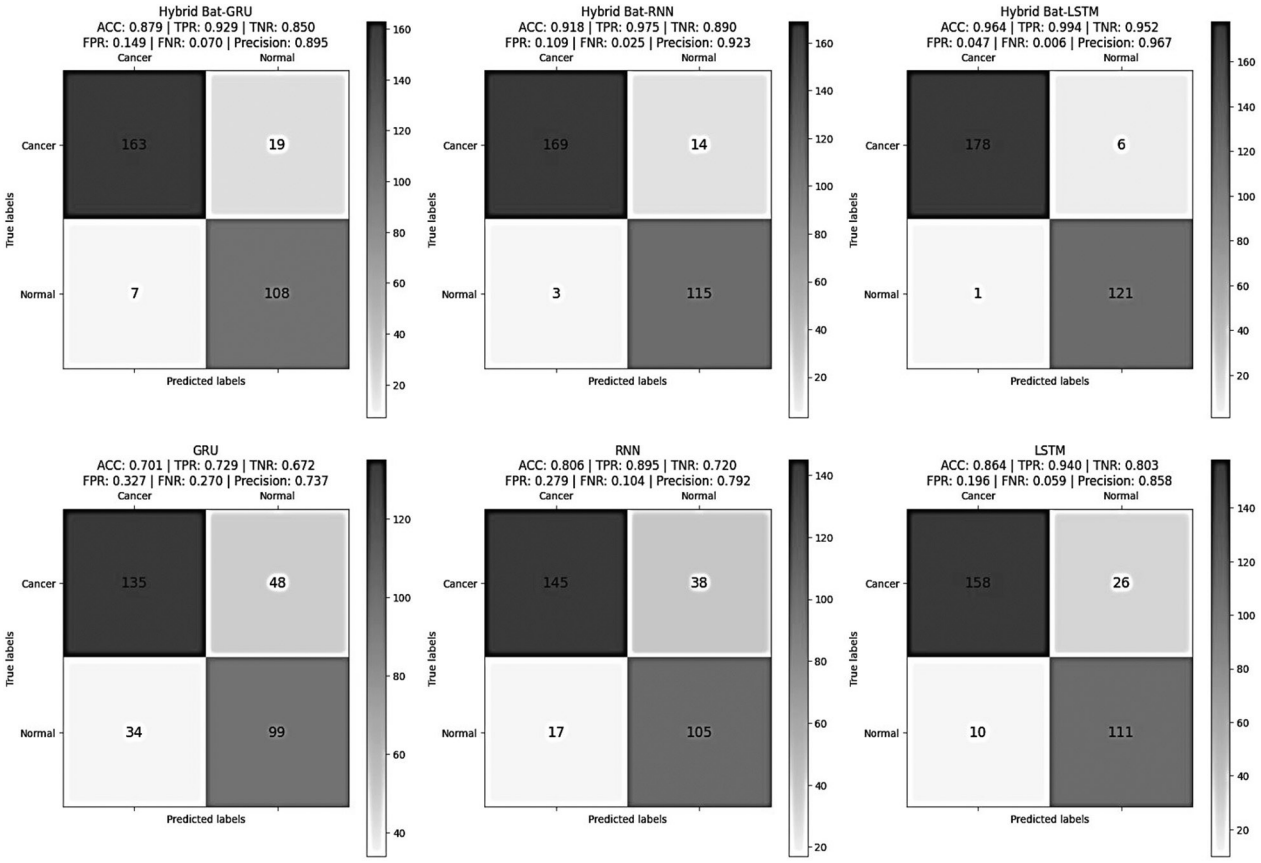

Figure 9. Confusion matrix displaying the proposed approach performance.

Table 7. Comparison of model accuracy with various data split ratios.

Split Ratio			**Accuracy**					
Train ratio	**Validation ratio**	**Test ratio**	**Hybrid bat-GRU**	**Hybrid bat-RNN**	**Hybrid bat-LSTM**	**GRU**	**RNN**	**LSTM**
90%	10%	10%	0.850	0.890	0.950	0.680	0.760	0.830
80%	10%	10%	0.879	0.918	0.964	0.701	0.806	0.864
70%	15%	15%	0.830	0.880	0.940	0.660	0.740	0.800
60%	20%	20%	0.800	0.850	0.920	0.640	0.720	0.780
50%	25%	25%	0.750	0.810	0.890	0.620	0.700	0.750

7. Conclusion

Most papers published possess the ability to identify a variety of characteristics that significantly influence breast cancer diagnosis. The reliance on a single medical professional's view to categorize cancer has grown increasingly problematic. They need deep learning to help medical practitioners diagnose breast cancer, and these procedures are universally acknowledged for determining the likelihood of patient survival. The proposed system is a way of classifying the gene expression of breast cancer genes to achieve better results; the large number of features and limited sample size affected the accuracy of this result. Feature reduction techniques and feature selection approaches were employed to mitigate the decline in accuracy. The missing values of the breast cancer gene expression dataset were filled in to obtain better results.

The bat algorithm combined with recurrent deep learning to extract important features related to breast cancer patients by utilizing the training dataset, thereby improving the classification accuracy of breast cancer. The hybrid bat-LSTM approach has the highest TPR (recall) of 0.994, excelling at identifying true positives, while the regular GRU model has the lowest recall (0.729), suggesting average performance. Hybrid models are often more effective at capturing true positive rates than the traditional RNN and LSTM architectures. These results suggest that integrating feature selection algorithms with deep learning techniques holds great promise for advancing the field of medical diagnostics. The proficiency of the hybrid bat-GRU, bat-RNN, and bat-LSTM models in discerning the nuances of gene expression data has laid the groundwork for future research.

In future research, a suggestion would be to explore a hybrid feature selection technique incorporating an embedded search strategy with deep learning. The idea behind combining multiple individual models for feature selection is that it can lead to better results than using a single feature selection method. However, the improvement is not just because of having multiple models, like classification ensembles, but also because of the variety of feature subsets obtained.

References

[1] X. Zhou *et al.*, "A Comprehensive Review for Breast Histopathology Image Analysis Using Classical and Deep Neural Networks", *IEEE Access*, vol. 8, pp. 90931–90956, 2020. http://dx.doi.org/10.1109/ACCESS.2020.2993788

[2] A. B. Nassif *et al.*, "Breast Cancer Detection Using Artificial Intelligence Techniques: A Systematic Literature Review", *Artif Intell Med*, vol. 127, p. 102276, 2022. https://doi.org/10.1016/j.artmed.2022.102276

[3] M. Lu *et al.*, "Artificial Intelligence in Pharmaceutical Sciences", *Engineering*, 2023. https://doi.org/10.1016/j.eng.2023.01.014

[4] W. W. Bin Goh and L. Wong, "The Birth of Bio-data Science: Trends, Expectations, and Applications", *Genomics Proteomics Bioinformatics*, vol. 18, no. 1, pp. 5–15, 2020. https://doi.org/10.1016/j.gpb.2020.01.002

[5] J. M. Engreitz *et al.*, "Local Regulation of Gene Expression by lncRNA Promoters, Transcription and Splicing", *Nature*, vol. 539, no. 7629, pp. 452–455, 2016. http://dx.doi.org/10.1038/nature20149

[6] M. Chawla and M. Duhan, "Bat Algorithm: A Survey of the State-of-the-Art", *Applied Artificial Intelligence*, vol. 29, no. 6, pp. 617–634, 2015. http://dx.doi.org/10.1080/08839514.2015.1038434

[7] A. H. Gandomi *et al.*, "Bat Algorithm for Constrained Optimization Tasks", *Neural Comput Appl*, vol. 22, no. 6, pp. 1239–1255, 2013. http://dx.doi.org/10.1007/s00521-012-1028-9

[8] H. Aljuaid *et al.*, "Computer-aided Diagnosis for Breast Cancer Classification Using Deep Neural Networks and Transfer Learning", *Comput Methods Programs Biomed*, vol. 223, p. 106951, 2022. https://doi.org/10.1016/j.cmpb.2022.106951

[9] R. Chawla *et al.*, "Brain Tumor Recognition Using an Integrated Bat Algorithm with a Convolutional Neural Network Approach", *Measurement: Sensors*, vol. 24, p. 100426, 2022. https://doi.org/10.1016/j.measen.2022.100426

[10] F. Gers *et al.*, "Learning Precise Timing with LSTM Recurrent Networks", *Journal of Machine Learning Research*, vol. 3, pp. 115–143, 2002. http://dx.doi.org/10.1162/153244303768966139

[11] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory", *Neural Comput*, vol. 9, no. 8, pp. 1735–1780, 1997. http://dx.doi.org/10.1162/neco.1997.9.8.1735

[12] N. A. K. Hussein and B. Al-Sarray, "Deep Learning and Machine Learning via a Genetic Algorithm to Classify Breast Cancer DNA Data", *Iraqi Journal of Science*, vol. 63, no. 7, pp. 3153–3168, 2022. http://dx.doi.org/10.24996/ijs.2022.63.7.36

[13] D. A. Omondiagbe *et al.*, "Machine Learning Classification Techniques for Breast Cancer Diagnosis", *IOP Conf Ser Mater Sci Eng*, vol. 495, no. 1, p. 012033, 2019. http://dx.doi.org/10.1088/1757-899X/495/1/012033

[14] R. I. Arathi Chandran and V. M. A. Bai, "Breast Cancer Recurrence Prediction with Deep Neural Network and Feature Optimization", *Automatika*, vol. 65, no. 1, pp. 343–360, 2024. http://dx.doi.org/10.1080/00051144.2023.2293280

[15] H. Saleh *et al.*, "Predicting Breast Cancer Based on Optimized Deep Learning Approach", *Comput Intell Neurosci*, vol. 2022, p. 1820777, 2022. http://dx.doi.org/10.1155/2022/1820777

[16] H. Hajiabadi *et al.*, "Combination of Loss Functions for Robust Breast Cancer Prediction", *Computers & Electrical Engineering*, vol. 84, p. 106624, 2020. https://doi.org/10.1016/j.compeleceng.2020.106624

[17] A. Bhardwaj and A. Tiwari, "Breast Cancer Diagnosis Using Genetically Optimized Neural Network model", *Expert Syst Appl*, vol. 42, no. 10, pp. 4611–4620, 2015. https://doi.org/10.1016/j.eswa.2015.01.065

[18] L. S. Solanki *et al.*, "An ANN Approach for False Alarm Detection in Microwave Breast Cancer Detection", in *Proceedings of the 2016 IEEE Congress on Evolutionary Computation (CEC)*, 2016, pp. 1370–1374. http://dx.doi.org/10.1109/CEC.2016.7743948

[19] M. Desai and M. Shah, "An Anatomization on Breast Cancer Detection and Diagnosis Employing Multi-layer Perceptron Neural Network (MLP) and Convolutional Neural Network (CNN)", *Clinical eHealth*, vol. 4, pp. 1–11, 2021. https://doi.org/10.1016/j.ceh.2020.11.002

[20] Prateek, "Breast Cancer Prediction: Importance of Feature Selection", in *Advances in Computer Communication and Computational Sciences*, S. K. Bhatia, S. Tiwari, K. K. Mishra, and M. C. Trivedi, Eds., Singapore: Springer Singapore, 2019, pp. 733–742.

[21] A. K. Verma *et al.*, "Breast Cancer Management System Using Decision Tree and Neural Network", *SN Comput Sci*, vol. 2, no. 3, p. 234, 2021. http://dx.doi.org/0.1007/s42979-021-00644-2

[22] S. Dasgupta *et al.*, "Feature Selection for Breast Cancer Detection Using Machine Learning Algorithms", *International Journal of Innovative Technology and Exploring Engineering*, vol. 8, no. 9, pp. 2080–2083, 2019.

[23] Q. Zhang *et al.*, "Multimodal Feature Learning and Fusion on B-mode Ultrasonography and Sonoelastography Using Point-wise Gated Deep Networks for Prostate Cancer Diagnosis", *Biomedical Engineering / Biomedizinische Technik*, vol. 65, 2019. http://dx.doi.org/10.1515/bmt-2018-0136

[24] K. J. Dsouza and Z. A. Ansari, "Histopathology Image Classification Using Hybrid Parallel Structured Deep-CNN Models", *Applied Computer Science*, vol. 18, no. 1, pp. 20–36, 2022. http://dx.doi.org/10.35784/acs-2022-2

[25] F. Gao *et al.*, "DeepCC: A Novel Deep Learning-based Framework for Cancer Molecular Subtype Classification", *Oncogenesis*, vol. 8, no. 9, p. 44, 2019. http://dx.doi.org/10.1038/s41389-019-0157-8

[26] T. Ahn *et al.*, "Deep Learning-based Identification of Cancer or Normal Tissue using Gene Expression Data", in *Proceedings of the 2018 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, 2018, pp. 1748–1752. http://dx.doi.org/10.1109/BIBM.2018.8621108

[27] J. Han *et al.*, "Depression Prediction Based on LassoNet-RNN Model: A Longitudinal Study", *Heliyon*, vol. 9, no. 10, p. e20684, 2023. https://doi.org/10.1016/j.heliyon.2023.e20684

[28] S. Babichev *et al.*, "Applying a Recurrent Neural Network-Based Deep Learning Model for Gene Expression Data Classification", *Applied Sciences (Switzerland)*, vol. 13, no. 21, 2023. http://dx.doi.org/10.3390/app132111823

[29] F. Shan *et al.*, "Effects of Data Smoothing and Recurrent Neural Network (RNN) Algorithms for Real-time Forecasting of Tunnel Boring Machine (TBM) Performance", *Journal of Rock Mechanics and Geotechnical Engineering*, 2023. https://doi.org/10.1016/j.jrmge.2023.06.015

[30] T. Zhang *et al.*, "Spatial–Temporal Recurrent Neural Network for Emotion Recognition", *IEEE Trans Cybern*, vol. 49, no. 3, pp. 839–847, 2019. http://dx.doi.org/10.1109/TCYB.2017.2788081

[31] J. Wu *et al.*, "A Hierarchical Recurrent Neural Network for Symbolic Melody Generation", *IEEE Trans Cybern*, vol. 50, no. 6, pp. 2749–2757, 2020. http://dx.doi.org/10.1109/TCYB.2019.2953194

[32] X. Li, "Construction of Transformer Fault Diagnosis and Prediction Model Based on Deep Learning", *Journal of Computing and Information Technology*, vol. 30, no. 4, pp. 223–238, 2022. http://dx.doi.org/10.20532/cit.2022.1005691

[33] J. Oruh *et al.*, "Long Short-Term Memory Recurrent Neural Network for Automatic Speech Recognition", *IEEE Access*, vol. 10, pp. 30069–30079, 2022. http://dx.doi.org/10.1109/ACCESS.2022.3159339

[34] M. Dua *et al.*, "An Improved RNN-LSTM based Novel Approach for Sheet Music Generation", *Procedia Comput Sci*, vol. 171, pp. 465–474, 2020. https://doi.org/10.1016/j.procs.2020.04.049

[35] M. R. Ibraheem *et al.*, "Diagnosis of Patellofemoral Osteoarthritis Using Enhanced Sequential Deep Learning Techniques", *Egyptian Informatics Journal*, vol. 24, no. 3, p. 100391, 2023. https://doi.org/10.1016/j.eij.2023.100391

[36] P. P. Barman and A. Boruah, "A RNN Based Approach for Next Word Prediction in Assamese Phonetic Transcription", *Procedia Comput Sci*, vol. 143, pp. 117–123, 2018. https://doi.org/10.1016/j.procs.2018.10.359

[37] S. Hu *et al.*, "Refining Short-Term Power Load Forecasting: An Optimized Model with Long Short-Term Memory Network", *Journal of Computing and Information Technology*, vol. 31, no. 3, pp. 151–166, 2023. http://dx.doi.org/10.20532/cit.2023.1005730

[38] M.-C. Chiu *et al.*, "A Hybrid CNN-GRU Based Probabilistic Model for Load Forecasting from Individual Household to Commercial Building", *Energy Reports*, vol. 9, pp. 94–105, 2023. https://doi.org/10.1016/j.egyr.2023.05.090

[39] M. Wang and F. Ying, "Point and Interval Prediction for Significant Wave Height Based on LSTM-GRU and KDE", *Ocean Engineering*, vol. 289, p. 116247, 2023. https://doi.org/10.1016/j.oceaneng.2023.116247

[40] Y. Zheng *et al.*, "State of Health Estimation for Lithium Battery Random Charging Process Based on CNN-GRU Method", *Energy Reports*, vol. 9, pp. 1–10, 2023. https://doi.org/10.1016/j.egyr.2022.12.093

[41] V. Bolón-Canedo *et al.*, "A Review of Microarray Datasets and Applied Feature Selection Methods", *Inf Sci (N Y)*, vol. 282, pp. 111–135, 2014. https://doi.org/10.1016/j.ins.2014.05.042

[42] M. Abd-Elnaby *et al.*, "Classification of Breast Cancer Using Microarray Gene Expression Data: A Survey", *J Biomed Inform*, vol. 117, p. 103764, 2021. https://doi.org/10.1016/j.jbi.2021.103764

[43] V. Bolón-Canedo and A. Alonso-Betanzos, "Ensembles for Feature Selection: A Review and Future Trends", *Information Fusion*, vol. 52, pp. 1–12, 2019. https://doi.org/10.1016/j.inffus.2018.11.008

[44] W. Z. Al-Dyani *et al.*, "Binary Bat Algorithm for Text Feature Selection in News Events Detection Model Using Markov Clustering", *Cogent Eng*, vol. 9, no. 1, p. 2010923, 2022. http://dx.doi.org/10.1080/23311916.2021.2010923

[45] J. Huang and Y. Ma, "Bat Algorithm Based on an Integration Strategy and Gaussian Distribution", *Math Probl Eng*, vol. 2020, pp. 1–22, 2020. http://dx.doi.org/10.1155/2020/9495281

[46] M. J. van de Vijver *et al.*, "A Gene-expression Signature as a Predictor of Survival in Breast Cancer", *N Engl J Med*, vol. 347, no. 25, pp. 1999–2009, 2002. http://dx.doi.org/10.1056/nejmoa021967

Received: March 2024
Revised: August 2024
Accepted: August 2024

Contact addresses:
Ali Nafaa Jaafar
Electrical Engineering Technical College
Middle Technical University
Baghdad
Iraq
e-mail: ali_nafaa@mtu.edu.iq

Ali Nafaa Jaafar received the BSc degree from the College of Science, Baghdad University, Iraq, in 2003. He received the MSc degree in computer science from the College of Science, Al Mustansiriyah University, Iraq, in 2019. He is currently assistant teacher at the Electrical Engineering Technical College, Middle Technical University, Baghdad, Iraq. His interests include artificial intelligence and data security.