International Journal of Computational Intelligence Systems (2024) 17:3
https://doi.org/10.1007/s44196-023-00388-2

RESEARCH ARTICLE

Detecting Thyroid Disease Using Optimized Machine Learning Model Based on Differential Evolution

Punit Gupta[1] · Furqan Rustam[1] · Khadija Kanwal[2] · Wajdi Aljedaani[3] · Sultan Alfarhood[4] · Mejdl Safran[4] · Imran Ashraf[5]

Received: 24 August 2023 / Accepted: 5 December 2023

Abstract

Thyroid disease has been on the rise during the past few years. Owing to its importance in metabolism, early detection of thyroid disease is a task of critical importance. Despite several existing works on thyroid disease detection, the problem of class imbalance is not investigated very well. In addition, existing studies predominantly focus on the binary-class problem. This study aims to solve these issues by the proposed approach where ten types of thyroid diseases are considered. The proposed approach uses a differential evolution (DE)-based optimization algorithm to fine-tune the parameters of machine learning models. Moreover, conditional generative adversarial networks are used for data augmentation. Several sets of experiments are carried out to analyze the performance of the proposed approach with and without model optimization. Results suggest that a 0.998 accuracy score can be obtained using AdaBoost with DE optimization which is better than existing state-of-the-art models.

Keywords Thyroid detection · Model optimization · Differential evolution · Machine learning · Deep learning

1 Introduction

The thyroid is a small, but very important gland in the neck that allows the human body to maintain digestion and heart rate [1]. The thyroid organ releases the hormones that control metabolisms such as body temperature and heart rate. It produces two important hormones, T4 and T3. For several metabolic activities, these hormones are responsible such as heart rate and body weight. The thyroid gland produces thyroid hormones that travel in the blood to help control several organs. When the function of the thyroid gland is affected, it leads to inappropriate production of the thyroid hormone. The symptoms of thyroid disease may involve high cholesterol, an unusual pulse rate, and high blood pressure. There are five common types of thyroid disease including hypothyroidism, structural abnormalities, hyperthyroidism, tumors, and subclinical hyperthyroidism or subclinical hypothyroidism. To diagnose hypothyroidism,

Punit Gupta and Furqan Rustam have contributed equally to this work.

✉ Sultan Alfarhood
sultanf@ksu.edu.sa

✉ Imran Ashraf
imranashraf@ynu.ac.kr

Punit Gupta
punitg07@gmail.com

Furqan Rustam
furqan.rustam1@gmail.com

Khadija Kanwal
khadijakanwal.6022@wum.edu.pk

Wajdi Aljedaani
wajdi.j1@gmail.com

Mejdl Safran
mejdl@ksu.edu.sa

[1] School of Computer Science, University College Dublin, Dublin D04 V1W8, Ireland

[2] Institute of Computer Science and Information Technology, The Women University Multan, Multan, Pakistan

[3] Department of Computer Science and Engineering, University of North Texas, Denton, TX, USA

[4] Department of Computer Science, College of Computer and Information Sciences, King Saud University, P. O. Box 51178, Riyadh 11543, Saudi Arabia

[5] Department of Information and Communication Engineering, Yeungnam University, Gyeongsan 38541, Republic of Korea

Published online: 03 January 2024

the blood sample is tested in a laboratory where medical specialists are needed to analyze the test reports for the hormones and other parameters to diagnose thyroid disease. In hypothyroidism, the thyroid gland does not produce sufficient thyroid hormone. Hypothyroidism may involve several symptoms such as the feeling of tiredness, poor capability to tolerate colds, constipation, depression, slow heart rate, and weight gain. Doctors adjust the medicine dose according to the patient's conditions to normalize TSH levels and thyroxine. Excessive thyroid hormones are produced in hyperthyroidism disease. Symptoms of hyperthyroidism include troubled sleeping, irritability, nervousness, fine brittle hair, increased perspiration, hand tremors, heart racing, anxiety, skin thinning, and muscular weakness. Hyperthyroidism is very common after the age of 60 years. There are three major treatment methods for hyperthyroidism disease such as medications, radioiodine therapy, and thyroid surgery.

The thyroid gland creates hormones to carry out several important functions in the body, and lack of thyroid balance (creating too much or too little) leads to thyroid disease. There are various diseases related to the thyroid which lead to imbalance and malfunctioning of various other organs [2]. If not properly treated, it can lead to complications such as goiter, heart disease, pregnancy problems, and more dangerous myxoedema coma [3]. According to [4], thyroid disease affects 200 million people worldwide and an estimated 40% of people are at risk of iodine deficiency which helps to produce thyroid hormone. Several kinds of thyroid diseases may occur and each has its own impact on human health, for example, hypothyroidism and hyperthyroidism and the two main types of thyroid disease that cause thyroid imbalance. To avoid such complications, early prediction of the correct thyroid disease type is important so that treatment can be done according to thyroid type.

In the past, various tests have been carried out that record different symptoms for thyroid diseases. These symptoms can be used to diagnose a specific thyroid disease [5]. For example, these tests and symptoms can be used to predict and diagnose thyroiditis/Graves', disease/Hashimoto's disease/goiter/thyroid, nodule/thyroid cancer, etc. Thyroid disease can be categorized based on various symptoms and features. Existing studies provide many features relevant to thyroid diseases. Table 1 provides the names and descriptions of a few features that can be used to predict thyroid disease; for example, lithium, goiter, hypopituitary, Psych, TSH, T3, TT4, T4U, FTI, and TBG can be used in thyroid detection.

Previously, several studies investigated thyroid diseases and their symptoms [6–8]. Some approaches focus on data analytics, while others carry out the recording of tests and symptoms. Based on the technique used for disease detection like statistical analysis, machine learning model, or deep learning model, the accuracy and robustness of such approaches vary. Predominantly, existing approaches make use of publicly available datasets and suffer from model overfitting. Most of the available datasets suffer from an imbalanced class problem where the number of samples for the positive (disease) class is substantially small. When such a dataset is used with a machine or deep learning model, the model overfits the majority class and produces false predictions for the minority class. Another limitation of existing studies is that only a few thyroid diseases are used for classification; for the most part, existing studies focus on the binary-class problem which makes those approaches unsuitable for real-world disease detection. This study focuses on resolving this issue by increasing the number of samples of the minority class using synthetic data samples.

Table 1 Thyroid test features

Feature	Description
TSH Thyroid-stimulating hormone	This is produced by pituitary glands to manage the thyroid hormones (TSH level in the blood from laboratory work)
T3 Triiodothyronine tests	100–200 ng/dL
Free T3 (FT3)	2.3–4.1 pg/mL
Test T4 (TT4)	
Thyroxine tests (11>T4>5) (T4)	Low T4 refers to hypothyroidism; high refers to hyperthyroidism
Free T4 or free thyroxine (FTI)	0.9–1.7 ng/dL

This study aims at mitigating the influence of class imbalance and increasing disease detection performance. In summary, this study makes the following contributions:

- An efficient machine learning approach is designed to predict thyroid disease with high accuracy. The performance of the models is optimized using a differential evolution (DE)-based optimization algorithm.
- DE interacts with hyperparameters of various machine learning models to identify the best hyperparameters. The DE algorithm is used to find the optimal parameters for machine learning models to obtain significant improvement in the accuracy of models for thyroid disease detection.
- This study specifically deals with the class imbalance problem. The impact of class imbalance is mitigated using conditional generative adversarial networks (CTGAN) to perform data augmentation.
- A higher number of classes are considered compared to existing studies. Thyroid disease detection is performed using ten classes with several machine learning models including random forest (RF), gradient boosting (GB), AdaBoost, logistic regression (LR), and support vector machine (SVM). In addition, long short-term

memory (LSTM), convolutional neural network (CNN), and recurrent neural network (RNN) are also employed.
- Extensive experiments are performed regarding the performance of models with and without DE optimization. The performance is further validated using k-fold cross-validation.

The rest of this paper is organized as follows: Sect. 2 discusses the related works. The study methodology is given in Sect. 3 and Sect. 4 presents study results and discussion. In the end, Sect. 5 concludes this study.

2 Related Work

For thyroid disease prediction, deep learning and machine learning methods have been applied in various existing research works. Prediction of thyroid disease at its early stages and categorization into cancer, hyperthyroidism, or hypothyroidism is very helpful for treating and recovering the maximum number of patients. To identify the recent research studies in the presented work, various thyroid disease classification and detection methods have been described here.

In [9], the authors present a machine learning approach for B-Raf proto-oncogene, serine/threonine kinase (BRAF) mutation presence in cancer thyroid nodules. The study also presented ultrasonic images of 96 thyroid nodules. Machine learning models such as RF, LR, and SVM are used for detecting the presence of BRAF mutation. Using these models, a classification accuracy of higher than 60% is reported. In another similar study, fine-needle aspiration (FNA) and ultrasonic features were used to reduce the false-negative rate for thyroid cancer. The RF model reported better results than other methods such as gradient descent and decision tree (DT). The authors applied LR and the least absolute shrinkage and selection operator (LASSO) models in [10] to choose the malignant thyroid nodule associated with the ultrasonic features. To classify the malignant thyroid nodules, the RF model is used along with a scoring system. The logistic lasso regression (LLR) with RF attained higher than 80% accuracy. In [6], the authors analyzed the data by applying different machine learning algorithms. The results are compared with ten different classifiers. An 84% accuracy was achieved by using an extra tree classifier.

The study [11] used SVM for detecting thyroid disease. The reported accuracy is 83.37%. Additionally, the model correctly distinguishes between four thyroid states. In [8], the authors performed experimentation to predict and classify thyroid disease using the DT model. In addition, researchers introduced a machine learning-based tool, a machine learning tool for thyroid disease diagnosis (MLTDD) to predict intelligently thyroid diseases. MLTDD shows an accuracy of 98.7%. Machine learning algorithms, including LR, RF, SVM, GBM, and DNN, are used to predict the highest probable molecules that initiate the homeostasis thyroid hormone in [12]. The authors investigated feature engineering using deep learning and machine learning methods in [13]. Backward feature elimination, forward feature selection, machine learning-based feature selection, and bidirectional feature elimination with an extra tree classifier were adopted. The proposed approach can predict non-thyroidal syndrome, Hashimoto's thyroiditis, autoimmune thyroiditis, and binding protein. Results indicate an improved accuracy of 99% using extra tree classifier-based selected features and with the RF classifier.

A multi-kernel SVM is presented in [14] to predict cancer and thyroid diseases. The gray-wolf optimization is applied for feature selection and improves performance. The study reports a 97.49% accuracy using the multi-kernel SVM. In [15], the authors applied image processing methods and feature selection techniques to select the important features from the database and obtain improved performance for thyroid diseases. Using machine learning and selective features techniques, [15] performed multi-class hypothyroidism. Hypothyroidism is categorized into four groups. Experimental results demonstrate that RF achieved 99.81% accuracy compared with KNN, DT, and SVM algorithms. The study [16] investigated three feature selection approaches using DT, SVM, RF, naive Bayes (NB), and LR for the prediction of hypothyroidism. Principal component analysis, univariate feature selection, and recursive feature selection were used for feature selection. Recursive feature selection combined with machine learning algorithms reported improved performance than other techniques. DT, SVM, RF, naive Bayes (NB), and LR algorithms are combined with RFE and achieve 99.35% accuracy.

The study [17] introduced a multiple multi-layer perceptron (MMLP) model for thyroid disease classification. The MMLP is reported to achieve 99% accuracy for large-scale datasets. Another study [18] presented the XGBoost technique to predict and classify thyroid disease. The XGBoost algorithm's efficiency is compared with DT, LR, and KNN approaches. The XGBoost algorithm improves the accuracy by 0.02% than the KNN algorithm. In [19], a comparative analysis for machine learning-based techniques RF, DT, artificial neural network (ANN), and KNN is presented. Experiments are carried out on a large-scale dataset. Moreover, both original and sampled data are considered for experiments. RF attained improved performance with 94.8% accuracy.

Besides using machine learning models, some studies specifically focus on using deep learning models for detecting thyroid disease. For example, a deep neural network (DNN) is used to predict and classify thyroid disease in [20]. DNN is reported to obtain an accuracy of 99.95%. The authors compared several machine learning algorithms such as extra

trees, CatBoost, LightGBM, ANN, KNN, SVM, RF, DT, XGBoost, and GaussianNB, to improve the thyroid prediction accuracy in [21]. The accuracy, recall, precision, and F1 score are examined to evaluate the performance. The research reported 96% accuracy using an ANN classifier. The study [22] utilized a CNN-based ResNet architecture to detect thyroid from an image dataset. The research focused on five types of thyroid conditions and achieved a 94% accuracy rate using ResNet models with the stochastic gradient descent (SGD) optimizer. The study [23], introduces a novel transfer learning approach, the distant domain high-level feature fusion (DHFF) model. DHFF aims to narrow the distribution gap between the source and target domains while preserving their unique characteristics. This approach prevents the overblending of features while facilitating a more effective transfer of knowledge acquisition. The proposed approach achieved 88.92% accuracy when applied to thyroid ultrasound auxiliary source domains.

The above-discussed studies investigate various machine learning and deep learning approaches for thyroid disease detection and report different results regarding the accuracy, precision, F1 score, etc. However, these studies have several limitations; in particular, we identified four major gaps concerning thyroid disease prediction. First, several studies perform experiments with smaller datasets and their results cannot be generalized. Secondly, the majority of the studies use thyroid detection, and the type of disease is not investigated. Thirdly, predominantly, the existing studies utilized imbalanced datasets. Although high accuracy is reported in these studies, the models lack generalizability. The models may experience bias and overfitting, leading to wrong predictions for the minority classes. The model overfitting can cause a higher number of false positives for the minor class. Lastly, for the most part, the optimization is carried out concerning the feature engineering and there is a lack of emphasis on model tuning. The model's fine-tuning holds a prominent significance, especially with respect to dataset characteristics. Traditional tuning methods have proven to be inefficient. Therefore, this study focuses on hyperparameter optimization using the DE algorithm. The comparative analysis for state-of-the-art methods is given in Table 2.

3 Materials and Methods

This section describes the dataset used for experiments, the use of CTGAN for data balancing, the working of the DE algorithm in the context of hyperparameter optimization, and a brief overview of machine learning models used in this study.

This study designs a machine learning approach for thyroid disease detection. Figure 1 shows the architecture of the proposed methodology. First, we acquire the dataset from the Kaggle [25]. The dataset contains 25 target classes, of which the top 10 target classes are selected for experiments. These classes are selected based on the high number of samples. The rest of the classes have very few samples, so they are not included in this study. The selected targeted dataset is imbalanced, so to make the dataset balanced we used the CTGAN augmentation technique. This technique generates samples for the minority class. Data splitting is done to divide the dataset into training and testing sets with a 0.8 to 0.2 ratio, where 80% is used for training and 20% is used for testing. Machine learning models work on numeric data, so we use a Label-encoder to convert data into numeric form before passing it to machine learning models. We train machine learning models with a training set and perform hyperparameter optimization using DE optimizer which helps to select the best hyperparameter setting for models. In the end, we evaluate models in terms of accuracy, precision, recall, F1 score, and confusion matrix.

3.1 Thyroid Disease Dataset

The datasets used in this study are taken from the Kaggle repository. The thyroid disease dataset comprises 9172 samples and every sample has 31 features. The dataset consists of various records for different thyroid diseases and the target classes. The target classes include health condition state and diagnosis classes. The importance of features should be evaluated to select the optimal number for features of thyroid disease classification. In addition, these features belong to different types including float, boolean, string, and int. The proposed approach works with ten classes, namely '-', 'K', 'G', 'I', 'F', 'R', 'A', 'L', 'M', and 'N'. The detailed description of each class is given in Table 3.

The class counts indicate that the dataset is imbalanced. For example, many samples in the dataset are not used in any particular class. However, data pre-processing is performed to attain the standard dataset to evaluate the performance. The classification count is used to classify as "no condition". Additionally, categorization is not performed on data samples for any other classes such as hypothyroid, hyperthyroid, general health, binding proteins, anti-thyroid treatment, replacement therapy, and many more. The patients who do not have thyroid disease are considered in "no condition". Therefore, concurrent non-thyroidal disease is generally seen in severely ill patients with chronic disease, and serum thyroid levels are changed due to chronic disease [35].

The thyroid disease dataset contains 9172 records of which 6771 samples belong to normal people, while others suffer from different types of thyroid diseases. For example, there are 647 hyperthyroid patients, 733 primary hypothyroid, 836 concurrent non-thyroidal disease patients, 859 compensated hypothyroid patients, etc. Detailed records for all ten classes used in this study are shown in Table 4.

Table 2 Summary of the systematic analysis studies in related work

Study	Year	Dataset	Sample size	Methods	Classes	Evaluation metrics	Results
[12]	2020	ToxCasts	N/A	RF, LR, XGB, SVM, ANN	2	F1 score	(TR) RF-81% and (TPO) XGB-83%
[14]	2020	UCI	7547,30 features	multi-kernel SVM	3	Accuracy, specificity, sensitivity	Sensitivity (99.05%), accuracy (97.49%), specificity (94.5%)
[15]	2021	UCI	3771,30 attributes	DT, RF, KNN, SVM	4	Accuracy	KNN−98.3%, DT−99.5%, SVM−96.1%, RF 99.81%
[16]	2021	Diagnostic center Dhaka, Bangladesh	519 samples	SVM, DT, RF, LR, and NB.RFE, UFS, PCA	4	Accuracy	RFE, LR, DT, SVM, RF Accuracy−99.35%
[17]	2021	UCI	7200,21 features	Multiple MLP	3	Accuracy	Multiple MLP 99%
[13]	2022	UCI	9172,31 features	RF, ADA, GBM, SVM, CNN–LSTM	4	Recall, precision, F1 score	RF-99% accuracy, CNN-94% precision
				RF, ADA, GBM, SVM, CNN–LSTM	4	Cross-validation, accuracy	CNN-92% recall, CNN-93% F1 score
[19]	2022	UCI	3162	DT, RF, KNN, ANN	2	Accuracy	Accuracy RF−94.8%
[18]	2022	UCI	215 With 5 features	KNN, XGB,LR, DT	3	Accuracy	KNN 81.25%, XGBoost 87.5%, LR 96.875%, DT 98.59%
[20]	2022	UCI	3152,23 features	DNN	2	Accuracy	Accuracy 99.95%
[24]	2022	UCI	7247,21 attributes	GWO, IGWO, HFBO	3	Accuracy, specificity, sensitivity	Sensitivity(99.2%), accuracy (99.28%), specificity (98%)
[21]	2022	UCI	3162	Extra-Trees, CatBoost, LightGBM, ANN, KNN, SVC, RF, DT, XGBoost, GaussianNB	4	Accuracy	Accuracy 95.7%
[22]	2023	Image dataset	6356	CNN-based ResNet and SGD	5	Accuracy	94%

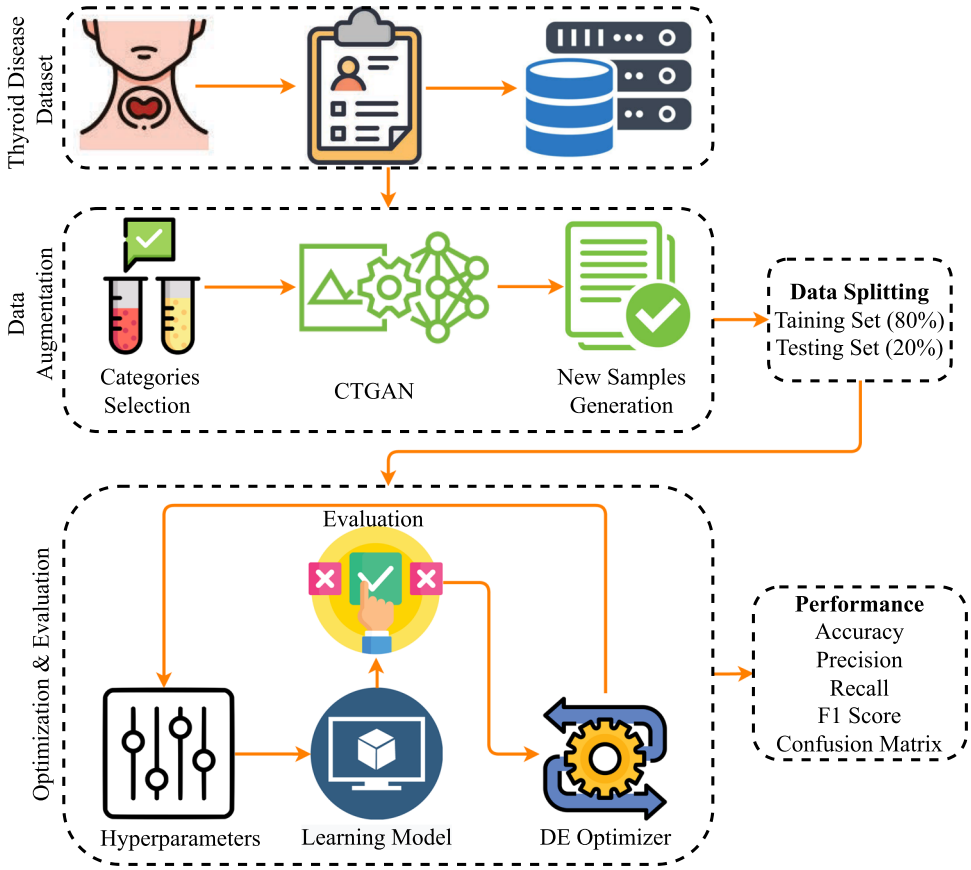

Fig. 1 Architecture of the proposed methodology

A total of 800 samples were randomly selected from the normal class. The primary hypothyroid increased binding protein, hypothyroid, hyperthyroid, consistent with replacement therapy, over-replaced, discordant assay results, and under-replaced and concurrent non-thyroid disease counts were not changed.

3.2 Data Balancing Using CTGAN

In this study, data balancing is carried out using CTGAN to generate samples. In the dataset used in this study, the normal class has the highest number of samples, while some other classes have a very small number of samples. To balance the dataset, 400–500 samples are generated using CTGAN. Table 5 shows the number of samples before and after data augmentation using CTGAN.

The purpose of dataset balancing is to avoid model overfitting which happens when a model is trained on a highly unbalanced dataset. For the unbalanced dataset, the feature distribution is skewed concerning the majority class as shown in Fig. 2b. Dataset balancing helps to normalize the feature distribution and reduces the probability of model overfitting. Feature distribution of the balanced dataset is shown in Fig. 2b.

3.3 Differential Evolution (DE)

This study uses a DE-based optimization algorithm to find the set of optimal hyperparameters for machine learning models such as RF, LR, SVM, AdaBoost, GB, etc . [36]. The hyperparameter optimization improves the performance of the models. The DE optimizer is divided into five phases: (1) Initialization, (2) fitness evaluation, (3) mutation, (4) crossover, and (5) stopping condition.

1. **Initialization:** In this phase, n^{th} random solutions are generated for a given problem. For this study, a random combination of hyperparameters (max_depth and a number of iterations (n_ite)) is generated. Each randomly generated solution is treated as a chromosome. This phase keeps a record of initializing all the basic parameters listed below:

   - **Number of the population (n):** 50.
   - **Number of iteration (I):** 100.
   - **Weighting factor(wf):** 0.9 (0 <wf<50).
   - **Crossover probability (CP):** 0.5 (0<CP<1),

   where $50 \leq Max\_Depth \leq 300$, $50 \leq n\_ite \leq 300$.
2. **Fitness Evaluation:** This phase evaluates the fitness of each chromosome/solution. For our problem, fitness

Table 3 Description of classes in the dataset

Class	Description	Detail
K	Concurrent non-thyroid illness	Non-thyroidal disease is normally used to define the changes in hormones related to the thyroid that may arise in serum or chronic disease that is not produced with an intrinsic irregularity in the thyroid function [26]
G	Compensated hypothyroid	Compensated hypothyroidism is also known as subclinical hypothyroidism. It is a condition that is associated with a high serum concentration for TSH, but a normal serum-free thyroxine (FT4) [27]
I	Increased binding protein	A binding protein is some protein that acts as an agent to combine two or many molecules [28]
F	Primary hypothyroid	Primary hypothyroidism is described as low levels of blood thyroid hormone because it damages the thyroid gland. This type of destruction is typically due to auto-immunity, including surgery, radiation, and radio-iodine [29]
R	Discordant assay results	Assay interfering may be a reason for abnormal thyroidal function tests. Recognition at an early stage prevents inappropriate patient management [30]
N	Over-replaced	In [31], the use of a high free T4 along with TSH to describe the over-replaced group, which is combined by use of a normal free T4 with low TSH to define the group not likely to be over-replaced, will reduce the possibility of error in allocating patients to the affected and control groups
A	Hyperthyroid	Excessive thyroid hormones are produced in hyperthyroidism disease. Symptoms of hyperthyroidism include trouble sleeping, irritability, nervousness, fine, brittle hair, increased perspiration, hand tremors, heart racing, anxiety, thinning of the skin, and muscular weakness [32]
L	Consistent with replacement therapy	Thyroid hormone therapy is usually prescribed when the patient's thyroid is not producing sufficient thyroid hormones naturally. This condition is called hypothyroidism. Another reason to use thyroid hormone therapy can rarely be: its use to control the growth of thyroid goiter [33]
M	Under-replaced	Hypothyroidism is generally treated by taking hormone replacement tablets daily, known as levothyroxine. It replaces thyroxine hormone, which is not enough in the patient's body [34]. It can rarely comprise: its use to control the growth of thyroid goiter [33]
-	No condition	No thyroid disease in patient / Normal report

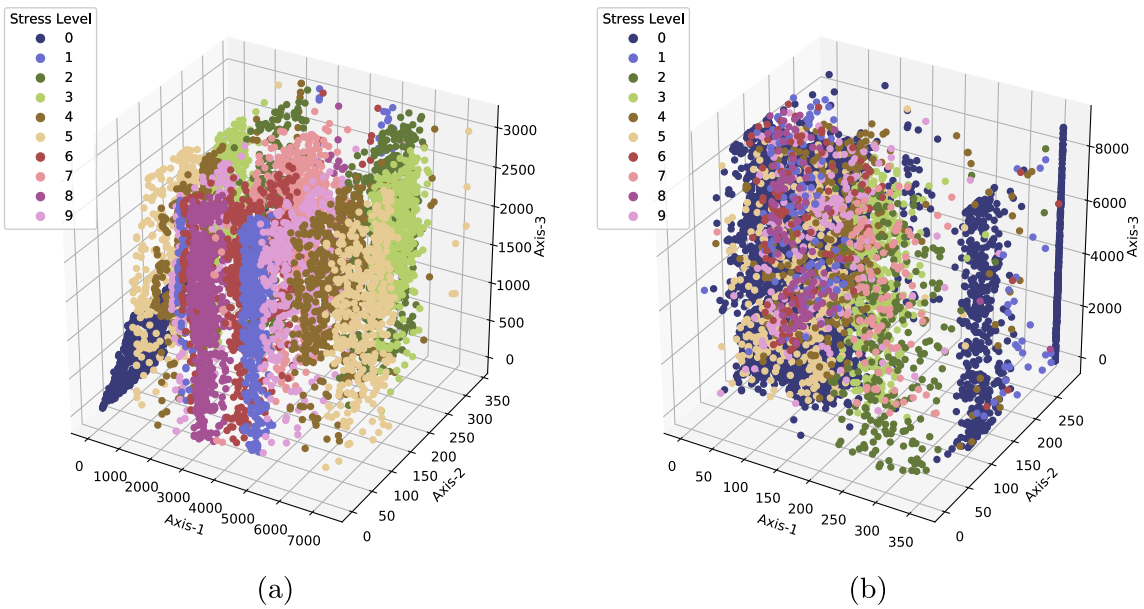

Fig. 2 Feature space visualization in three dimensions. For this, we used the principal component analysis (PCA) technique which converts high-dimensional data into three dimensions and then we visualize this 3D data using a scatter plot. **a** Feature space after CTGAN. **b** Feature space using original dataset

Table 4 Number of records for each class

Class	# of samples
Normal	800
Over-replaced	436
Concurrent non-thyroidal illness	359
Compensated hypothyroid	346
Increased binding protein	233
Primary hypothyroid	196
Discordant assay results	147
Hyperthyroid	115
Consistent with replacement therapy	111
Under-replaced	110

Table 5 Details of the dataset before and after data augmentation

Class	Count	New sample count
Normal	800	800
Over-replaced	436	836
Concurrent non-thyroidal illness	359	859
Compensated hypothyroid	346	846
Increased binding protein	233	733
Primary hypothyroid	196	596
Discordant assay results	147	647
Hyperthyroid	115	615
Consistent with replacement therapy	111	611
Under-replaced	110	610

refers to the accuracy score of the model using the given set of hyperparameters.

3. **Mutation:** In this phase, a new offspring is generated from the existing solution in search of a new better solution. This involves the random selection of two chromosomes. Then using the weighting factor, one target chromosome is selected with maximum fitness. Figure 3 shows the mutation process. A new chromosome can be defined as

$$New_{chromosome} = target_{vector} + wf * (random\_selected_1 - random\_selected_1). \quad (1)$$

4. **Crossover:** In this process, some of the values of the new solution are interchanged with the existing solution. The crossover probability defines the probability of swapping the values. For this study, it is 0.5 which means 50% of the chromosomes are updated. Figure 4 shows the process carried out in the crossover.
5. **Stopping Condition:** This step defines when to stop the process of searching for new solutions. The stopping condition is the number of iterations to find the solution. In the proposed approach, if the condition for the number of iterations is met, the process stops, and it returns the best-explored solution along with the best accuracy score achieved using that global solution. Figure 5 illustrates the flowchart of the DE and the interaction between DE and ML as proposed in this study.

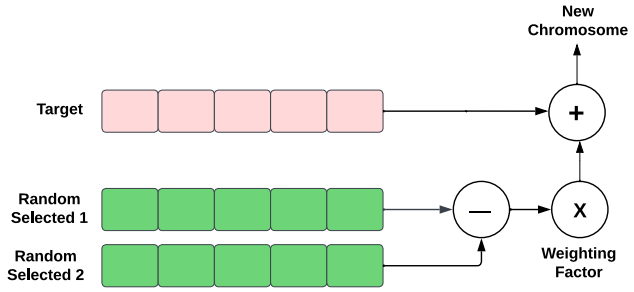

Fig. 3 Process of mutation

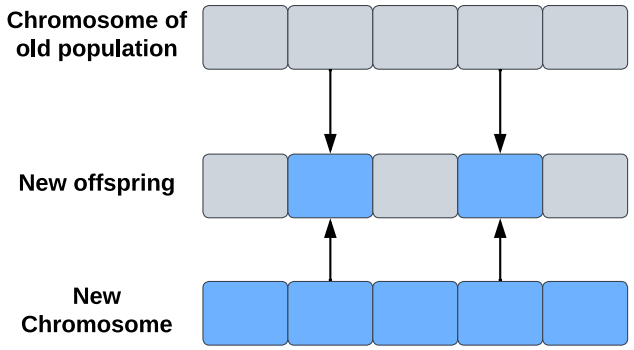

Fig. 4 Process of crossover

3.4 Machine Learning Models

The presented approach employs various machine learning models to detect thyroid disease. RF, SVM, LR, AdaBoost, GBM, CNN, RNN, and LSTM are used in the presented study. These classifiers are tuned for performance enhancement using DE optimizer and a list of optimized parameter values is given in Table 6.

3.4.1 Random Forest

RF is employed for regression and classification problems. RF is an ensemble classifier that uses a tree-based classification technique. Additionally, RF is applied to reduce the overfitting problems using a bootstrap approach for sampling. It defines the best prediction by the voting process. Additionally, it detects the significant elements within a dataset and reports a simple indicator for feature significance. Feature selection is applied in classification research to reconstruct the data and also improve the accuracy. Many

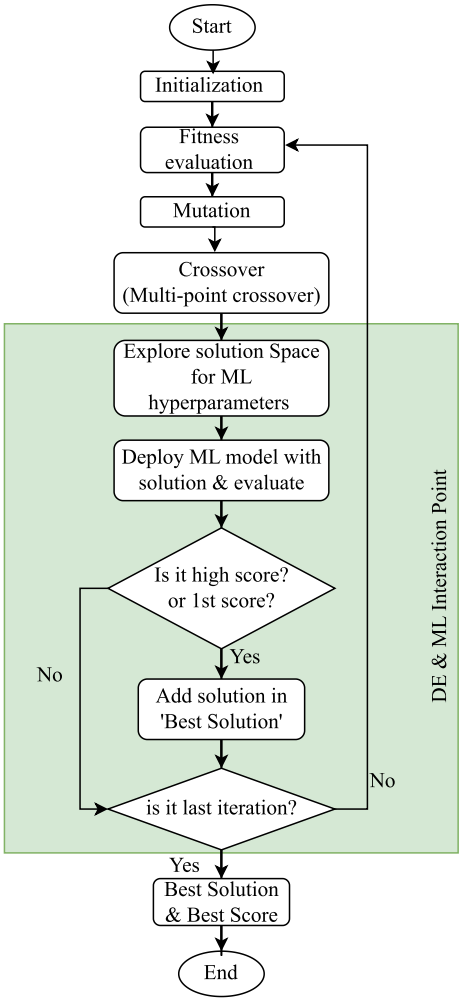

Fig. 5 DE optimizer flow diagram

models are trained on boot-strapped samples that are used for classification in the bagging approach. RF can produce more consistent ensemble forecasts than a DT. The test statistic of a single function is computed using the feature selection method in Eq. 2

$$norm\ t_j = \frac{t_j}{\sum_{jk \epsilon all features}^{t}}. \tag{2}$$

In Eq. 2, $norm\ t\_j$ is used to normalize the importance of feature j.

In Eq. 3, the total number of trees is divided by the assigned value to every node feature importance.

$$RFt_j = \frac{\sum_{ji \epsilon alltrees} normt_j}{R}. \tag{3}$$

RFt_j is used for feature importance, and j is computed from all trees in the RF classifier. RFt_j denotes the normalized feature importance for j in tree k, and R is used for the total number of trees.

3.4.2 Logistic Regression

LR is a statistical method that is used to analyze the data and comprises one or more variables to predict outcomes. LR is employed to evaluate the probability of class members to confirm the target variable. The logistic function is applied to estimate the probabilities of behavior among independent and dependent variables [37]. The 'solver' variable is set as 'linear' because of linearly separable data. Furthermore, the 'multi-class' variable is used with the 'multi-nomial' value because of multi-class classification. The 'C' parameter is set to 4, which represents the inverse of regularization strength and decreases gradually the overfitting probability [38].

3.4.3 Support Vector Machine

A support vector machine is a linear classifier that is used for classification and regression. SVM is used to divide the sample data into various classes using a hyperplane or set of hyperplanes in n-dimensional space [39, 40]. SVM achieves classification by selecting the 'best-fit' hyperplane that can differentiate the classes. This study uses a 'linear' kernel of SVM which is often used when the dataset has several features. The training with the linear kernel is very fast due to the need for W regularization variable optimization. The value of C is set to 5.0.

Table 6 Hyperparameter setting and optimization range for machine learning models

Model	Hyperparameters	Hyperparameters range
RF	n_estimators=282,max_depth=15	n_estimators = {2 to 300}, max_depth = {2 to 300}
GB	n_estimators = 148, max_depth = 107, learning_rate = 0.5	n_estimators = {2 to 300}, max_depth = {2 to 300}, learning_rate = {0.1 to 0.9}
AdaBoost	n_estimators = 286, learning_rate = 0.6	n_estimators = 2 to 300, learning_rate = {0.1 to 0.9}
LR	Solver = 'saga', multi_class = 'multinomial', C = 4	Solver = {'newton-cg', 'lbfgs', 'sag', 'saga' }, multi_class = 'multinomial', C = { 1 to 10}
SVM	kernel= 'linear', C=5.0	kernel = {'linear', 'poly', 'rbf', 'sigmoid', 'precomputed'}, C = { 1.0 to 10.0}

3.4.4 Gradient Boosting Machine

GBM is the best technique due to its prediction accuracy and speed, particularly for complex and large-scale datasets. GBM is used to minimize the bias error. GBM is similar to AdaBoost; the major difference is that GBM has a fixed base estimator, i.e., DT, whereas in AdaBoost the base estimator can be changed according to requirements. In GBM, the base model is built to predict the observations in the training dataset as defined here: [41]

$$H(x) = arg_{\alpha} mini \sum_{k=1}^{m} l(x_k, \alpha), \tag{4}$$

where l is used for the loss function, α predicts the value and $argmini$ represents the predicted α, where the loss function is minimum. The target column is a continuous loss function that will be

$$l = \frac{1}{m} \sum_{k=0}^{m} (x_k - \alpha_k)^2, \tag{5}$$

where x_k is used for observed values and α is for the predicted value in Eq. 5. We simply calculate the average of all numbers in a leaf.

$$\alpha_n = arg_{\alpha} mini \sum_{k=1}^{m} l(x_k, f_{n-1}(y_k) + \alpha H_n(y_k)), \tag{6}$$

where $H_n(y_k)$ is the decision tree made on residuals and n is the number of the decision tree.

3.4.5 Long Short-Term Memory

LSTM is an RNN that contains an efficient memory cell to help LSTM forget or remember things. In LSTM there are four interacting layers including forget gate, update gate, input gate, and output gate. Using the forget gate, the decision is made whether the information is thrown away from the cell state as shown in Eq. 7.

$$r_s = \delta(Y_r.[g_s - 1, z_s] + \alpha_r), \tag{7}$$

where g_s is used for the weight matrix, a_r is the bias vector, and r_s is a number between 0 and 1, where 0 denotes the forget value and 1 is the keeping value.

The input gate is used with a *tanh* layer and a sigmoid layer to decide which values will be modified as shown in Eqs. 8 and 9.

$$i_s = \delta(Y_i.[g_s - 1, z_s] + \alpha_s), \tag{8}$$

$$\tilde{f_s} = tanh(Y_f.[g_s - 1, z_s] + \alpha_f). \tag{9}$$

In Eqs. 8 and 9, Y_i and Y_f are used for weight matrices. Here, a_s and a_f are for bias vectors. i_s, f_s is for outputs.

In Eq. 10, the update gate updates the old cell state by value from the input gate.

$$f_s = r_s * f_s - 1 + i_s * \tilde{f_s}, \tag{10}$$

where r_s is used to decide which information is to be forgotten. $i_s * \tilde{f_s}$ selects the total number of values to be modified in the cell.

Lastly, the output gate in Eqs. 11 and 12 is used to decide which value is the output from the layer [42].

$$i_d = \delta(Y_d.[g_s - 1, z_s] + \alpha_d), \tag{11}$$

$$e_s = d_s * tanh(f_s). \tag{12}$$

In Eqs. 11 and 12, the value of i_d is used to decide the output state. The new cell state f_s is multiplied by d_s. The $tanh$ function is selected to achieve e_s in Eq. 12 which is the output of i_d.

3.4.6 Convolutional Neural Network

The CNN is mainly presented to deal with the variability for two-dimensional shapes. CNN contains two layers: pooling layer and a convolutional layer. The convolution layer performs the convolution for the previous layer output along with a sliding filter bank to generate the output feature map. Sigma is the sigmoid function that is used as a function for network activation. Both W_{qm}^{ae} and D_q^{ae} are the filters that create the training parameters for convolution layers in Eq. 13 [43].

$$F_b^{ae} = sigma\left(\sum_{m=1}^{n} j_m^{(ae-1)} * W_{qm}^{ae} + D_q^{ae}\right). \tag{13}$$

The pooling layer is used to minimize feature map resolution and the sensitivity of the output. Max pooling is generally used for pooling in convolutional neural networks.

3.4.7 Recurrent Neural Network

An RNN is designed to handle sequential or time series data, accept the current input, and receive previous inputs. RNN simulates a discrete-time dynamic system that has a_d for the input layer, b_d for the hidden layer, and c_d for the output layer. The d is used to represent the time. The dynamical model is described in Eqs. 14 and 15 [44].

$$b_d = F_b(\alpha_d, b_{d-1}), \tag{14}$$

$$c_d = F_0(b_d), \tag{15}$$

Table 7 RF performance comparison with and without DE optimizer

Without optimization				With optimization			
Class	Precision	Recall	F1 score	Models	Precision	Recall	F1 score
-	1.00	1.00	1.00	**–**	0.99	1.00	1.00
A	0.96	0.99	0.97	**A**	1.00	1.00	1.00
F	1.00	1.00	1.00	**F**	0.99	0.99	0.99
G	0.98	1.00	0.99	**G**	0.99	1.00	1.00
I	0.98	0.98	0.98	**I**	0.99	0.99	0.99
K	0.99	1.00	1.00	**K**	1.00	0.99	1.00
L	0.99	0.97	0.98	**L**	1.00	1.00	1.00
M	1.00	0.99	1.00	**M**	0.98	1.00	0.99
N	1.00	1.00	1.00	**N**	1.00	1.00	1.00
R	0.98	0.94	0.96	**R**	1.00	0.97	0.98
Average	0.99	0.99	0.99	**Average**	1.00	0.99	1.00

where F_b and F_0 are functions for the state transition and output, respectively, in the equation. Every function is parameterized using a set of parameters, ϕb and $\phi 0$.

Suppose there is a set of M training sequences $A = ((\alpha_1^{(m)}, c_1^{(m)}), ..., ((\alpha_{Dm}^{(m)}, c_{Dm}^{(m)})_{m=1}^{M}$. RNN's parameters are estimated to minimize cost function as described in Eq. 16 [44].

$$V(\phi) = \frac{1}{M} \sum_{m=1}^{M} \sum_{d=1}^{Dm} j(c_d^{(m)}, F_o(b_d^{(m)})), \tag{16}$$

where $b_d^{(m)} = F_b(\alpha) =_d^{(m)}, c_{d-1}^{(m)})$ and $b_d^{(m)} = 0$, $j(x, y)$ is a predefined divergence value between x and y, as cross-entropy or Euclidean distance.

4 Results and Discussions

This section contains the results of machine learning models to analyze the performance of models with and without the DE optimizer. In addition, k-fol cross-validation and performance comparison with existing state-of-the-art models is also carried out.

4.1 Experimental Setup

The machine learning models are investigated in terms of accuracy, precision, recall, and F1 score. Moreover, each model is evaluated using confusion matrix values. This study performs experiments using an Intel Core i7 11th generation system with 16 GB RAM, 1TB SSD, and Windows 10.0 operating system. We used Jupyter Notebook and Python language to implement the proposed approach. The proposed approach uses the sci-kit learn library and TensorFlow framework.

4.2 Results of Machine Learning Models

Table 7 shows the results of RF with and without optimization. RF shows very good performance in terms of all evaluation parameters using the DE optimizer with 1.00, 0.99, and 1.00 scores, for precision, recall, and F1 score, respectively, while without an optimizer, RF achieved 0.99, 0.99, and 0.99 scores for precision, recall, and F1 scores, respectively. The class-wise results indicate that some classes have lower precision and F1 scores when the model is not optimized. However, when using DE optimization, we observed improvements in the results for individual classes. For instance, prior to optimization, class A had precision and F1 scores of 0.96 and 0.97, respectively. After optimization, we achieved significant scores of 1.00 for both metrics. Similarly, we observed the significance of DE for classes I and R results.

Table 8 shows the confusion matrix for the RF model. With the DE optimizer, RF gives 1464 correct predictions out of 1471 and only 7 predictions are wrong. Similarly, without DE optimizer, RF gives 1454 correct predictions and 17 wrong predictions. These results show that the RF achieved significantly better results with the DE optimizer.

Table 9 contains the results of GBM using DE optimization, as well as results without the optimization. Results show that GBM has significant performance, similar to the RF model. GBM achieves a 1.00 score each for precision, recall, and F1 score. RF recall score was 0.99, while GBM achieved a 1.00 F1 score. GBM achieved a 0.99 score in terms of all evaluations without the DE optimizer. Similarly, the performance of GBM for the individual class is also better when it is optimized using DE optimization.

Overall, GBM is better as compared to RF, as GBM predicts only six wrong predictions which is one prediction less as compared to RF, as shown in Table 10. GBM gives 1455 correct predictions and 16 wrong predictions without the DE

Table 8 RF confusion matrix with and without DE optimizer

Without optimizer									
205	0	0	0	0	0	0	0	0	0
0	135	0	0	0	0	0	0	0	1
0	0	157	0	0	0	0	0	0	0
0	0	0	178	0	0	0	0	0	0
0	0	0	2	165	0	1	0	0	1
0	0	0	0	0	149	0	0	0	0
0	1	0	0	3	0	112	0	0	0
0	0	0	0	0	1	0	117	0	0
0	0	0	0	0	0	0	0	121	0
0	5	0	1	1	0	0	0	0	115
Optimizer									
198	0	0	0	0	0	0	0	0	0
0	115	0	0	0	0	0	0	0	0
0	0	134	0	0	0	0	1	0	0
0	0	0	188	0	0	0	0	0	0
0	0	0	1	176	0	0	0	0	0
1	0	0	0	0	168	0	0	0	0
0	0	0	0	0	0	132	0	0	0
0	0	0	0	0	0	0	111	0	0
0	0	0	0	0	0	0	0	121	0
0	0	1	0	2	0	0	1	0	121

Table 10 GBM confusion metrics with and without the DE optimizer

Without optimizer									
205	0	0	0	0	0	0	0	0	0
3	133	0	0	0	0	0	0	0	0
2	0	155	0	0	0	0	0	0	0
0	0	0	178	0	0	0	0	0	0
2	0	0	2	165	0	0	0	0	0
0	0	0	0	0	149	0	0	0	0
1	0	0	0	2	0	113	0	0	0
3	0	0	0	0	0	0	115	0	0
0	0	0	0	0	0	0	0	121	0
0	0	0	0	1	0	0	0	0	121
Optimizer									
198	0	0	0	0	0	0	0	0	0
0	115	0	0	0	0	0	0	0	0
0	0	135	0	0	0	0	0	0	0
0	0	0	188	0	0	0	0	0	0
0	0	0	0	177	0	0	0	0	0
1	0	0	0	0	168	0	0	0	0
0	0	0	1	0	0	131	0	0	0
0	0	0	0	0	0	0	110	1	0
0	0	0	0	0	0	0	0	121	0
0	1	1	0	1	0	0	0	0	122

optimizer. Although GBM shows better performance due to its boosting operations, DE optimization further improves its performance.

Table 11 shows the performance of the AdaBoost model. AdaBoost is similar to GBM, but it identifies the shortcomings of the existing weak learners by high-weight data points, while GBM uses gradients. Results show that Adaboost outperforms in all evaluation parameters, as it achieved a 1.00 score each for precision, recall, and F1 score when used with the DE optimizer. However, using Adaboost without the DE optimization does not produce good results, as it achieved scores of 0.95, 0.90, and 0.87 in terms of precision, recall, and F1 scores, respectively.

Confusion metrics for AdaBoost are shown in Table 12. According to the results, AdaBoost gives only 3 wrong predictions and 1468 correct predictions using DE optimizer which is the highest ratio as compared to other models used in this study. On the other hand, AdaBoost gives 1337 cor-

Table 9 GBM performance comparison with and without the DE optimizer

Without optimization				With optimization			
Models	Precision	Recall	F1 score	Models	Precision	Recall	F1 score
–	0.95	1.00	0.97	**–**	0.99	1.00	1.00
A	1.00	0.98	0.99	**A**	0.99	1.00	1.00
F	1.00	0.99	0.99	**F**	0.99	1.00	1.00
G	0.99	1.00	0.99	**G**	0.99	1.00	1.00
I	0.98	0.98	0.98	**I**	0.99	1.00	1.00
K	1.00	1.00	1.00	**K**	1.00	0.99	1.00
L	1.00	0.97	0.99	**L**	1.00	0.99	1.00
M	1.00	0.97	0.99	**M**	1.00	0.99	1.00
N	1.00	1.00	1.00	**N**	0.99	1.00	1.00
R	1.00	0.99	1.00	**R**	1.00	0.98	0.99
Average	0.99	0.99	0.99	**Average**	1.00	1.00	1.00

Table 11 Adaboost performance comparison with and without the DE optimizer

Without optimization				With optimization			
Models	Precision	Recall	F1 score	Models	Precision	Recall	F1 score
–	1.00	1.00	1.00	-	0.99	1.00	1.00
A	1.00	0.99	1.00	**A**	0.99	1.00	1.00
F	1.00	0.99	1.00	**F**	1.00	1.00	1.00
G	1.00	0.99	1.00	**G**	1.00	0.99	1.00
I	0.99	1.00	1.00	**I**	1.00	1.00	1.00
K	1.00	1.00	1.00	**K**	0.99	0.99	0.99
L	0.51	0.99	0.67	**L**	1.00	0.99	1.00
M	0.99	1.00	1.00	**M**	1.00	1.00	1.00
N	1.00	0.05	0.10	**N**	1.00	1.00	1.00
R	0.98	1.00	0.99	**R**	1.00	1.00	1.00
Average	0.95	0.90	0.87	**Average**	1.00	1.00	1.00

Table 12 ADA confusion metrics with and without the DE optimizer

Without optimizer									
223	0	0	0	0	0	0	0	0	0
0	131	0	0	0	0	0	0	0	1
0	0	137	0	0	0	0	0	0	1
0	0	0	181	1	0	0	0	0	0
0	0	0	0	148	0	0	0	0	0
0	0	0	0	0	151	0	0	0	0
0	0	0	0	0	0	136	1	0	0
0	0	0	0	0	0	0	114	0	0
0	0	0	0	0	0	130	0	7	0
0	0	0	0	0	0	0	0	0	109
Optimizer									
198	0	0	0	0	0	0	0	0	0
0	115	0	0	0	0	0	0	0	0
0	0	135	0	0	0	0	0	0	0
0	0	0	187	0	1	0	0	0	0
0	0	0	0	177	0	0	0	0	0
1	0	0	0	0	168	0	0	0	0
0	1	0	0	0	0	131	0	0	0
0	0	0	0	0	0	0	111	0	0
0	0	0	0	0	0	0	0	121	0
0	0	0	0	0	0	0	0	0	125

rect predictions and 134 wrong predictions without a DE optimizer.

In this study, we also used linear models such as LR and SVM which can be good on linearly separable data. LR does not show good results in this study because LR performs better when the feature set is larger compared to the number of features. However, in this study, the feature set is small compared to the number of samples. Table 13 shows the results of LR in terms of precision, recall, and F1 score. LR shows poor results without and with DE optimization. LR gives 0.62, 0.61, and 0.60 scores in terms of precision, recall, and F1 scores, respectively, without the DE optimizer. LR achieved 0.62, 0.62, and 0.61 scores for precision, recall, and F1 scores, respectively, using the DE optimizer which is marginally better than results without optimization.

For the number of correct and wrong predictions, LR gives 946 correct predictions out of 1471 predictions, and 525 predictions are wrong which is the highest number of wrong prediction ratios as compared to all other models, as shown in Table 14.

SVM is good as compared to LR, but not good when compared with the results from RF, GBM, and AdaBoost, as shown in Table 15. SVM can be good on multi-class data because of kernel property and it can be good for small feature sets as well as on large feature sets. We used it with optimized hyperparameters and achieved good results. It achieved 0.96, 0.96, and 0.96 scores for precision, recall, and F1 scores, respectively without the DE optimizer. SVM shows poor results without DE optimization as it gives 0.89, 0.89, and 0.89 scores for precision, recall, and F1 scores, respectively.

The confusion matrix given in Table 16 indicates that it gives 49 wrong predictions and 1422 correct predictions using the DE optimizer. Although this performance is better than LR, however, GBM, RF, and AdbaBoost show better results than SVM.

Performance comparison of machine learning models in terms of accuracy score is given in Table 17. AdaBoost shows the highest accuracy score, as it gives 0.998 accuracy. It is followed by GBM, which has an accuracy score of 0.996. Both these models use boosting algorithms that train weak learners sequentially. Each preceding model is trained on the output of the conceding model which helps to achieve better results. RF is also good with a 0.995 accuracy score, but LR shows the worst performance as it gives only a 0.643 accuracy score. These results show that tree-based ensemble models

Table 13 LR performance comparison with and without the DE optimizer

Without optimization				With optimization			
Models	Precision	Recall	F1 score	Models	Precision	Recall	F1 score
–	0.92	0.92	0.92	**–**	0.89	0.95	0.92
A	0.51	0.65	0.57	**A**	0.49	0.68	0.57
F	0.90	0.94	0.92	**F**	0.90	0.90	0.90
G	0.52	0.66	0.58	**G**	0.51	0.59	0.55
I	0.77	0.77	0.77	**I**	0.81	0.77	0.79
K	0.63	0.71	0.67	**K**	0.62	0.66	0.64
L	0.72	0.31	0.43	**L**	0.66	0.42	0.52
M	0.17	0.11	0.14	**M**	0.28	0.23	0.25
N	0.55	0.69	0.61	**N**	0.55	0.67	0.61
R	0.51	0.36	0.42	**R**	0.50	0.27	0.35
Average	0.62	0.61	0.60	**Average**	0.62	0.62	0.61

Table 14 LR confusion metrics with and without the DE optimizer

Without Optimizer									
206	0	0	0	2	0	0	1	13	1
0	86	0	2	3	0	6	8	26	1
2	0	130	5	0	1	0	0	0	0
5	1	0	121	7	14	0	34	0	0
0	3	3	9	114	5	0	6	3	5
10	0	2	10	5	107	7 1	7	6	3
0	32	0	0	6	15	42	2	15	25
0	0	9	79	6	5	0	13	0	2
0	30	0	0	1	0	5	6	95	0
0	17	1	6	5	22	4	1	14	39
Optimizer									
189	0	2	0	1	3	0	0	3	0
0	78	0	4	3	0	6	1	23	0
1	0	122	7	1	0	0	3	0	1
4	2	2	111	4	17	0	47	0	1
1	3	3	8	137	10	3	3	6	3
17	0	3	12	5	112	0	4	11	5
0	28	0	0	3	11	56	4	10	20
0	1	3	65	6	7	0	26	0	3
0	27	0	0	0	0	7	5	81	1
0	21	1	12	10	21	13	1	12	34

are significantly better for thyroid disease prediction, while linear models show poor results.

4.3 Results of Deep Learning Models

In addition to machine learning models, this study deployed several deep learning models for comparison. DE optimization is not performed for deep learning models; these models are deployed on the data after applying CTGAN. We used four deep learning models including LSTM, CNN, RNN, and CNN–LSTM. We used all models with an embedding layer with a 5000 vocabulary size and 200 output size. After the embedding layer, the layer of each model is used. The ending layer of all models consists of ten neurons and a softmax function. We used all models with categorical_crossentropy loss function, the Adam optimizer, and 100 epochs. Architectural details of all deep learning models are provided in Table 18.

Table 19 shows the results of deep learning models for thyroid disease detection. Results suggest that the performance of deep learning models is not good. The LSTM achieved a 0.90 accuracy score and CNN achieved 0.86 accuracy. LSTM is a recurrent architecture that has feedback connections, as it is capable of processing the entire sequence of data, apart from single data points, while CNN requires a large feature set to make correct predictions. We also used a combination of CNN and LSTM in which CNN extracts the features for LSTM. CNN does not show good performance. Since the feature set from CNN is not good, LSTM could not perform well. We used RNN which also has recurrent architecture and can perform well on a small feature set.

4.4 K-Fold Cross-Validation Results

We also performed K-fold cross-validation to analyze the models' performance using DE optimization. We evaluate models in terms of mean accuracy and standard deviation (SD). We used tenfold cross-validation in this study and results are shown in Table 20. All models perform well; however, AdaBoost shows significantly better performance with 0.99 means accuracy and ±0.11 SD. Similarly, RF is also good with a 0.98 accuracy and ±0.03 SD. LR and SVM show a similar performance: 0.61 accuracy and ±0.09 SD each.

Table 15 Results using SVM model

Without optimization				With optimization			
Models	Precision	Recall	F1 score	Models	Precision	Recall	F1 score
–	0.98	1.00	0.99	**–**	1.00	1.00	1.00
A	0.84	0.92	0.88	**A**	0.99	0.99	0.99
F	0.93	0.91	0.92	**F**	0.99	0.99	0.99
G	0.89	0.95	0.92	**G**	0.99	0.99	0.99
I	0.89	0.84	0.87	**I**	0.99	0.99	0.99
K	0.88	0.89	0.88	**K**	0.99	0.99	0.99
L	0.82	0.84	0.83	**L**	0.99	0.99	0.99
M	0.96	0.94	0.95	**M**	0.91	0.98	0.94
N	0.92	0.85	0.88	**N**	0.83	0.95	0.89
R	0.80	0.72	0.75	**R**	0.96	0.98	0.97
Average	0.89	0.89	0.89	**Average**	0.96	0.96	0.96

Table 16 Confusion matrix of SVM for thyroid disease detection

Without optimizer									
223	0	0	0	0	0	0	0	0	0
0	122	0	1	3	0	1	0	0	5
0	2	126	7	0	0	0	3	0	0
0	2	1	173	3	2	0	0	0	1
0	3	3	7	125	4	1	0	1	4
5	0	1	3	3	134	0	0	0	5
0	5	0	0	3	0	115	0	9	5
0	0	2	0	0	2	3	107	0	0
0	0	1	0	1	0	18	1	116	0
0	11	1	3	2	11	2	1	0	78
Optimizer									
198	0	0	0	0	0	0	0	0	0
0	109	0	0	0	0	1	1	0	4
0	0	134	0	0	0	0	0	0	1
0	0	0	186	1	1	0	0	0	0
0	0	1	0	176	0	0	0	0	0
0	0	0	1	0	168	0	0	0	0
0	1	0	0	0	0	105	4	22	0
0	0	0	0	0	0	1	109	1	0
0	0	0	0	0	0	0	6	115	0
0	3	0	0	0	0	0	0	0	122

4.5 Comparison With Existing Models

In this section, we have compared the performance of the current approach with existing studies. For comparison, we select recent studies which utilized similar datasets. To perform a fair comparison, we implemented the models from the selected studies and utilized them with the dataset used in this study. The study [45] worked on heart disease prediction using a transfer learning approach. The study used CNN

Table 17 Machine learning models in terms of accuracy score

Without optimization		With optimization	
Models	Accuracy	Models	Accuracy
RF	0.988	RF	0.995
GBM	0.989	GBM	0.996
AdaBoost	0.910	**AdaBoost**	0.998
LR	0.643	LR	0.643
SVM	0.899	SVM	0.966

Table 18 Architecture of deep learning models

Model	Architecture
RNN	Embedding(50,000,200,)
	Dropout(0.5)
	SimpleRNN(128)
	Dense(10, activation='softmax')
CNN	Embedding(50,000,200,)
	Conv1D(128, 3, activation='relu')
	MaxPooling1D(pool_size=3)
	Dropout(0.5)
	Flatten()
	Dense(10, activation='softmax')
LSTM	Embedding(50,000,200,)
	Dropout(0.5))
	LSTM(100)
	Dense(10, activation='softmax')
CNN–LSTM	Embedding(50,000,200,)
	Conv1D(128, 3, activation='relu')
	MaxPooling1D(pool_size=3)
	LSTM(100)
	Dense(10, activation='softmax'
loss='category_crossentropy', optimizer='adam', epochs =100	

Table 19 Deep learning model performance for thyroid disease prediction

LSTM				CNN			
Models	Precision	Recall	F1 score	Models	Precision	Recall	F1 score
–	0.80	0.93	0.86	–	0.65	0.97	0.78
A	0.77	0.99	0.86	**A**	0.97	0.91	0.94
F	1.00	0.82	0.90	**F**	0.83	0.99	0.90
G	0.98	0.90	0.94	**G**	0.99	0.83	0.90
I	0.90	0.93	0.92	**I**	0.99	0.74	0.85
K	0.96	0.70	0.81	**K**	0.97	0.78	0.86
L	0.96	0.97	0.97	**L**	0.77	0.85	0.80
M	0.82	0.96	0.88	**M**	0.97	0.99	0.98
N	0.95	1.00	0.97	**N**	0.84	0.99	0.91
R	0.95	0.88	0.91	**R**	0.82	0.64	0.72
Average	0.91	0.91	0.90	**Average**	0.88	0.87	0.86
Accuracy	0.90			**Accuracy**	0.86		
CNN–LSTM Models	**Precision**	**Recall**	**F1 score**	**RNN Models**	**Precision**	**Recall**	**F1 score**
–	0.53	0.97	0.69	–	0.51	0.92	0.65
A	0.83	0.94	0.88	**A**	0.99	0.98	0.99
F	0.94	0.92	0.93	**F**	0.95	0.92	0.93
G	0.93	0.94	0.93	**G**	0.97	0.81	0.88
I	0.84	0.53	0.65	**I**	0.99	0.79	0.88
K	0.90	0.63	0.74	**K**	0.94	0.62	0.75
L	0.52	0.82	0.64	**L**	1.00	1.00	1.00
M	0.94	0.90	0.92	**M**	1.00	1.00	1.00
N	0.96	0.44	0.61	**N**	0.97	1.00	0.99
R	0.62	0.41	0.49	**R**	1.00	0.90	0.95
Average	0.80	0.75	0.75	**Average**	0.93	0.90	0.90
Accuracy	0.76			**Accuracy**	0.87		

for feature selection and an ensemble model is used for prediction. The study [20] worked on thyroid disease using deep neural networks. We deployed the deep neural network as per the architecture given in the study and performed experiments on the current dataset. Similarly, the study [13] investigated various feature extraction and machine learning techniques for thyroid disease detection. Performance comparison is carried out with these studies and results are shown in Table 21.

4.6 Results of T-test

We conducted a statistical analysis to compare the performance of this study with that of previous studies. We employed a statistical T-test and examined the results of all approaches. The T-test involves the formulation of two hypotheses.

- Null hypothesis (H_0): there is no significant difference in accuracy between this study and previous studies.
- Alternative hypothesis (H_a): there is a significant difference in accuracy between this study and previous studies.

T-test results are presented in Table 22. With a significance level (alpha) set at 0.05, the critical value is 2.35. When the absolute T-score exceeds the critical value, the T-test typically leads to the rejection of the H_0. In this case, the T-scores significantly exceed the critical value, indicating strong evidence of significant differences in all three comparisons. Notably, in the comparison between this study and [13], while the T-Score is lower compared to the other two comparisons, it is still sufficiently high to suggest a significant difference in the performance metrics.

5 Conclusion

The thyroid gland is an important organ of the human body that controls the metabolic operations of the body, and inappropriate production of thyroid hormone can lead to many

Table 20 Results of tenfold cross-validation

Models	Accuracy	SD
RF	0.98	±0.03
GBM	0.92	±0.13
AdaBoost	0.99	±0.11
LR	0.61	± 0.09
SVM	0.61	± 0.09

Table 21 Comparison with other studies

Ref.	Year	Accuracy	Precision	Recall	F1 score
[45]	2022	0.860	0.86	0.86	0.86
[20]	2022	0.890	0.89	0.89	0.89
[13]	2022	0.990	0.99	0.99	0.99
This study	2022	0.998	1.00	1.00	1.00

Table 22 Statistical T-test scores

Comparison	T-score	H_0
Proposed vs [45]	279.00	Rejected
Proposed vs [20]	219.00	Rejected
Proposed vs [13]	19.00	Rejected

complications. Early detection of thyroid disorders is critical to avoid such complications. This study employs a differential evolution-based optimization algorithm to find optimal parameters for machine learning models to obtain higher performance for thyroid disease detection. It is further aided by data augmentation using the CTGAN model. Experimental results suggest that an accuracy of 0.998 can be obtained using the optimized AdaBoost model by differential evolution. These results are further validated by k-fold cross-validation and performance appraisal with state-of-the-art approaches. Results indicate that contrary to linear models, ensemble models tend to show better performance. Machine learning models show better results using augmented datasets than deep learning models. This study provides two major contributions to enhancing thyroid detection. Using a differential evolution algorithm for hyperparameter optimization provides improved performance by the machine learning models compared to existing studies where conventional hyperparameter optimization is carried out. Secondly, CTGAN helps to balance the number of samples of each class which mitigates the probability of model bias and overfitting. Therefore, the models show robust performance and are generalizable compared to existing models. We intend to increase the dataset size to further analyze the performance of deep learning models in the future.

Acknowledgements The authors extend their gratitude to the researchers with supporting project number (RSPD2024R890), King Saud University, Riyadh, Saudi Arabia.

Author Contributions PG conceived the idea, performed data analysis, and wrote the original draft. FR conceived the idea, performed data curation, and wrote the original draft. KK performed data curation and formal analysis, and designed the methodology. WA did project administration, dealt with software, and performed visualization. SA acquired the funding for research and performed visualization and initial investigation. MS dealt with software, carried out project administration, and performed validation. IA supervised the study, performed validation, and reviewed and edited the manuscript. All authors read and approved the final manuscript.

Funding This research is funded by the Researchers Supporting Project Number (RSPD2024R890), King Saud University, Riyadh, Saudi Arabia.

Availability of Data and Materials Not applicable.

Declarations

Conflict of Interest The authors declare no conflict of interests.

Ethics Approval Not applicable.

Consent to Participate Not applicable.

Consent for Publication Not applicable.

Code Availability Not applicable.

References

1. Sharma, K.A., Arya, R., Mehta, R., Sharma, R., Sharma, K.A.: Hypothyroidism and cardiovascular disease: factors, mechanism and future perspectives. Curr. Med. Chem. **20**(35), 4411–4418 (2013)
2. healthline: The 6 Common Thyroid Problems & Diseases. https://www.healthline.com/health/common-thyroid-disorders (2018)
3. UK, N.H.S.: Underactive thyroid (hypothyroidism). https://www.nhs.uk/conditions/underactive-thyroid-hypothyroidism/ (2021)
4. Diabetes, T.L., Endocrinology: The untapped potential of the thyroid axis. https://www.thelancet.com/journals/landia/article/PIIS2213-8587(13)70166-9/fulltext#articleInformation (2013)
5. Zhang, B., Tian, J., Pei, S., Chen, Y., He, X., Dong, Y., Zhang, L., Mo, X., Huang, W., Cong, S., et al.: Machine learning-assisted system for thyroid nodule diagnosis. Thyroid **29**(6), 858–867 (2019)

6. Idarraga, A.J., Luong, G., Hsiao, V., Schneider, D.F.: False negative rates in benign thyroid nodule diagnosis: Machine learning for detecting malignancy. J. Surg. Res. **268**, 562–569 (2021)
7. Razia, S., Rao, M.N.: Machine learning techniques for thyroid disease diagnosis-a review. Indian J. Sci. Technol. **9**(28), 1–9 (2016)
8. Aversano, L., Bernardi, M.L., Cimitile, M., Iammarino, M., Macchia, P.E., Nettore, I.C., Verdone, C.: Thyroid disease treatment prediction with machine learning approaches. Procedia Computer Science **192**, 1031–1040 (2021)
9. Kwon, M.-R., Shin, J., Park, H., Cho, H., Hahn, S., Park, K.: Radiomics study of thyroid ultrasound for predicting braf mutation in papillary thyroid carcinoma: preliminary results. Am. J. Neuroradiol. **41**(4), 700–705 (2020)
10. Chen, D., Hu, J., Zhu, M., Tang, N., Yang, Y., Feng, Y.: Diagnosis of thyroid nodules for ultrasonographic characteristics indicative of malignancy using random forest. BioData mining **13**(1), 1–21 (2020)
11. Razia, S., Siva Kumar, P., Rao, A.S.: Machine learning techniques for thyroid disease diagnosis: a systematic review. Modern Approaches in Machine Learning and Cognitive Science: A Walkthrough, 203–212 (2020)
12. Lomana, M., Weber, A.G., Birk, B., Landsiedel, R., Achenbach, J., Schleifer, K.-J., Mathea, M., Kirchmair, J.: In silico models to predict the perturbation of molecular initiating events related to thyroid hormone homeostasis. Chem. Res. Toxicol. **34**(2), 396–411 (2020)
13. Chaganti, R., Rustam, F., De La Torre Díez, I., Mazón, J.L.V., Rodríguez, C.L., Ashraf, I.: Thyroid disease prediction using selective features and machine learning techniques. Cancers **14**(16), 3914 (2022)
14. Shankar, K., Lakshmanaprabu, S., Gupta, D., Maseleno, A., De Albuquerque, V.H.C.: Optimal feature-based multi-kernel svm approach for thyroid disease classification. J. Supercomput. **76**(2), 1128–1143 (2020)
15. Das, R., Saraswat, S., Chandel, D., Karan, S., Kirar, J.S.: An ai driven approach for multiclass hypothyroidism classification. In: International Conference on Advanced Network Technologies and Intelligent Computing, pp. 319–327 (2021). Springer
16. Riajuliislam, M., Rahim, K.Z., Mahmud, A.: Prediction of thyroid disease (hypothyroid) in early stage using feature selection and classification techniques. In: 2021 International Conference on Information and Communication Technology for Sustainable Development (ICICT4SD), pp. 60–64 (2021). IEEE
17. Hosseinzadeh, M., Ahmed, O.H., Ghafour, M.Y., Safara, F., Ali, S., Vo, B., Chiang, H.-S., et al.: A multiple multilayer perceptron neural network with an adaptive learning algorithm for thyroid disease diagnosis in the internet of medical things. J. Supercomput. **77**(4), 3616–3637 (2021)
18. Sankar, S., Potti, A., Chandrika, G.N., Ramasubbareddy, S.: Thyroid disease prediction using xgboost algorithms. Journal of Mobile Multimedia **18**(3), 1–18 (2022)
19. Alyas, T., Hamid, M., Alissa, K., Faiz, T., Tabassum, N., Ahmad, A.: Empirical method for thyroid disease classification using a machine learning approach. BioMed Research International **2022** (2022)
20. Jha, R., Bhattacharjee, V., Mustafi, A.: Increasing the prediction accuracy for thyroid disease: A step towards better health for society. Wireless Pers. Commun. **122**(2), 1921–1938 (2022)
21. Islam, S.S., Haque, M.S., Miah, M.S.U., Sarwar, T.B., Nugraha, R.: Application of machine learning algorithms to predict the thyroid disease risk: an experimental comparative study. PeerJ Computer Science **8**, 898 (2022)
22. Prathibha, S., Dahiya, D., Robin, C., Nishkala, C.V., Swedha, S.: A novel technique for detecting various thyroid diseases using deep learning. Intell. Autom. Soft Comput **35**(1), 199–214 (2023)
23. Tang, F., Ding, J., Wang, L., Ning, C.: A novel distant domain transfer learning framework for thyroid image classification. Neural Process. Lett. **55**(3), 2175–2191 (2023)
24. Sureshkumar, V., Balasubramaniam, S., Ravi, V., Arunachalam, A.: A hybrid optimization algorithm-based feature selection for thyroid disease classifier with rough type-2 fuzzy support vector machine. Expert. Syst. **39**(1), 12811 (2022)
25. Kaggle, E.F.W.: Thyroid diseases dataset. https://www.kaggle.com/datasets/emmanuelfwerr/thyroid-disease-data (2022)
26. Yasar, Z., Kirakli, C., Cimen, P., Ucar, Z.Z., Talay, F., Tibet, G.: Is non-thyroidal illness syndrome a predictor for prolonged weaning in intubated chronic obstructive pulmonary disease patients? Int. J. Clin. Exp. Med. **8**(6), 10114 (2015)
27. Salerno, M., Improda, N., Capalbo, D.: Management of endocrine disease subclinical hypothyroidism in children. Eur. J. Endocrinol. **183**(2), 13–28 (2020)
28. Bielli, P., Busà, R., Paronetto, M.P., Sette, C.: The rna-binding protein sam68 is a multifunctional player in human cancer. Endocr. Relat. Cancer **18**(4), 91–102 (2011)
29. Thyroiditis, C.A.: Chronic autoimmune thyroiditis. The Thyroid and Its Diseases: A Comprehensive Guide for the Clinician, 379 (2019)
30. Srichomkwun, P., Scherberg, N.H., Jakšić, J., Refetoff, S.: Diagnostic dilemma in discordant thyroid function tests due to thyroid hormone autoantibodies. AACE clinical case reports **3**(1), 22–25 (2017)
31. Livingston, M., Birch, K., Guy, M., Kane, J., Heald, A.: No role for tri-iodothyronine (t3) testing in the assessment of levothyroxine (t4) over-replacement in hypothyroid patients. Br. J. Biomed. Sci. **72**(4), 160–163 (2015)
32. Sharma, A.: Thyroid: An updated study on the diagnosis and treatment of hyper and hypo thyroidism (2017)
33. Biondi, B., Cooper, D.S.: Thyroid hormone therapy for hypothyroidism. Endocrine **66**(1), 18–26 (2019)
34. Jonklaas, J., Bianco, A.C., Bauer, A.J., Burman, K.D., Cappola, A.R., Celi, F.S., Cooper, D.S., Kim, B.W., Peeters, R.P., Rosenthal, M.S., *et al.*: Guidelines for the treatment of hypothyroidism: prepared by the american thyroid association task force on thyroid hormone replacement. thyroid **24**(12), 1670–1751 (2014)
35. Wajner, S.M., Maia, A.L.: New insights toward the acute non-thyroidal illness syndrome. Front. Endocrinol. **3**, 8 (2012)
36. Schmidt, M., Safarani, S., Gastinger, J., Jacobs, T., Nicolas, S., Schülke, A.: On the performance of differential evolution for hyperparameter tuning. In: 2019 International Joint Conference on Neural Networks (IJCNN), pp. 1–8 (2019). IEEE
37. Rustam, F., Mehmood, A., Ahmad, M., Ullah, S., Khan, D.M., Choi, G.S.: Classification of shopify app user reviews using novel multi text features. IEEE Access **8**, 30234–30244 (2020)
38. Grégoire, G.: 4-complements and applications. In: Statistics for Astrophysics, pp. 145–180. EDP Sciences (2022)
39. Zainuddin, N., Selamat, A.: Sentiment analysis using support vector machine. In: 2014 International Conference on Computer, Communications, and Control Technology (I4CT), pp. 333–337 (2014). IEEE
40. Zheng, W., Ye, Q.: Sentiment classification of chinese traveler reviews by support vector machine algorithm. In: 2009 Third International Symposium on Intelligent Information Technology Application, vol. 3, pp. 335–338 (2009). IEEE
41. Ivatt, P.D., Evans, M.J.: Improving the prediction of an atmospheric chemistry transport model using gradient-boosted regression trees. Atmos. Chem. Phys. **20**(13), 8063–8082 (2020)

42. Hong, X., Lin, R., Yang, C., Zeng, N., Cai, C., Gou, J., Yang, J.: Predicting alzheimer's disease using lstm. Ieee Access **7**, 80893–80901 (2019)
43. Namdeo, R.B., Janardan, G.V.: Thyroid disorder diagnosis by optimal convolutional neuron based cnn architecture. Journal of Experimental & Theoretical Artificial Intelligence **34**(5), 871–890 (2022)
44. Pascanu, R., Gulcehre, C., Cho, K., Bengio, Y.: How to construct deep recurrent neural networks. arXiv preprint arXiv:1312.6026 (2013)
45. Rustam, F., Ishaq, A., Munir, K., Almutairi, M., Aslam, N., Ashraf, I.: Incorporating cnn features for optimizing performance of ensemble classifier for cardiovascular disease prediction. Diagnostics **12**(6), 1474 (2022)