Polycystic Ovary Syndrome (PCOS) Detection: Using Deep Learning Approaches for Images and Clinical Features

Thesis submitted in partial fulfillment of the requirements for the award of degree of

Master of Engineering
in
Computer Science and Engineering

Submitted By
Arushi Madaan
802332088

Under the supervision of:

Dr. Anu Bajaj
Assistant Professor

Dr. Sunita Garhwal
Associate Professor

THAPAR INSTITUTE
OF ENGINEERING & TECHNOLOGY
(Deemed to be University)

COMPUTER SCIENCE AND ENGINEERING DEPARTMENT
THAPAR INSTITUTE OF ENGINEERING AND TECHNOLOGY
PATIALA – 147004

June 2025

CERTIFICATE

I hereby certify that the work which is being presented in the thesis entitled, **"***Polycystic Ovary Syndrome (PCOS) Detection: Using Deep Learning Approaches for Images and Clinical Features.***"**, in partial fulfillment of the requirements for the award of degree of Master of Engineering in Computer Science and Engineering submitted in Computer Science and Engineering Department of Thapar Institute of Engineering and Technology, Patiala, is an authentic record of my own work carried out under the supervision of *Dr. Anu Bajaj and Dr. Sunita Garhwal* and refers other researcher's work which are duly listed in the reference section.

The matter presented in the thesis has not been submitted for award of any other degree of this or any other University.

Arushi Madaan

Signature:

Arushi Madaan

This is to certify that the above statement made by the candidate is correct and true to the best of my knowledge.

Sunita
2/7/2025

Anu Bajaj

Dr. Sunita Garhwal — **Dr. Anu Bajaj**

Associate Professor — **Assistant Professor**

Acknowledgement

This project would not have been possible without the help of many people, and it is my pleasure to convey my gratitude to them for their various forms of support, encouragement, and help during the project's duration.

I want to thank my supervisors first and foremost, *Dr. Sunita Garhwal* and *Dr. Anu Bajaj*, for allowing me to work under their capable guidance and for their unwavering support. This endeavor would not have been possible without their support and advice. They were the ones who first taught me topics like machine learning, and deep learning as well as providing me with extremely useful insights through regular discussions during this project. Continuous talks have aided me in acquiring a good understanding of this thesis.

Additionally, I want to express my gratitude to Dr. Shalini Batra, Head of the Department, and to the entire faculty members of the department for their valuable feedback and suggestions.

Arushi Madaan

Arushi Madaan

Abstract

The hormone disorder known as polycystic ovarian syndrome (PCOS) is most prevalent in women nowadays. The ovaries of women are directly impacted by this metabolic disorder, where numerous small sacs of fluid develop around the edges of the ovary, called cysts or follicles. Despite advancements in technology, the exact cause of PCOS still remains unknown. It takes a lot of work for the doctors to manually diagnose it from the ultrasound (US) images. For PCOS identification, a range of machine learning (ML), deep learning (DL) and image processing techniques were employed, that allowed for the identification of follicle counts and follicle size analysis. The current research on PCOS detection for image segmentation and classification, as well as clinical characteristics, was reviewed in this work. For this study, we utilised two different kinds of dataset: image data and clinical feature dataset. We proposed a transfer learning-based DL approach for classifying women with PCOS by using ultrasound images. InceptionV3 and ResNet50 models were used, which achieved accuracy rates of 99.68% and 97.5%, respectively.

Numerous DL algorithms were applied on the clinical dataset using various feature selection techniques. The flower pollination algorithm (FPA), cuckoo search algorithm (CSA), and genetic algorithm (GA) are the nature-inspired feature selection algorithms that have been utilised in the study. The results of the proposed GA-based feature selection using an ensemble model were statistically tested with the existing algorithms using Analysis of variance (ANOVA) and the Tukey test, with a 94.58% accuracy, 97.48% precision, 83.76% recall, 90.08% F1-Score, and 99.09% specificity. The interpretability of the results is further ascertained by explainable artificial intelligence (XAI) techniques, i.e., LIME and SHAP. Overall, by using reliable and accurate prediction models, this work advances predictive analytics in PCOS diagnosis with the goal of assisting early diagnosis and well-informed decision-making processes.

Table of Contents

List of Figures

List of Tables

Chapter

1. Introduction

This chapter covers the fundamental details required for Polycystic Ovarian Disease (PCOD) detection by ML and DL methods, an overview of PCOS Detection and the motivation for starting this research.

1.1 Polycystic Ovary Disease

The term PCOS/PCOD is widely used and considered interchangeable in the field of medicinal healthcare. It is the most prevalent condition in women's life nowadays and typically occurs during the reproductive period [1]. The male hormone that is androgen, when it gets produced highly in a female's body, causes the reason for PCOS. It happens to most women between the ages of 18 and 44. This condition disrupts ovary functioning in that it forms various small sacs of fluid around the corners of the ovary, which is known as a cyst. It is currently the leading cause of infertility among women. There are numerous additional conditions linked to this issue, such as [2] obesity, diabetes, irregular cycles, weight gain, acne, and facial hair growth. Technology has advanced, but the exact root cause of PCOS is still unknown. Therefore, it is crucial to avoid this by making an early diagnosis [3]. To determine whether there are follicles in the ovary, US images were analyzed. Doctors found it difficult to determine the number of follicles visible in the ovary's US images because traditional procedures still depend on subjective assessments and clinical symptoms, which can cause a delay in diagnosis. This was a tedious process where human errors may occur from time to time. [4]. Modern technology has made significant strides in the medical field, allowing the easy and automatic diagnosis of follicles seen in US images using artificial intelligence (AI). The applications of AI across disciplines and related technologies are rapidly evolving and transforming various areas. As shown in Figure 1.1, the future directions in PCOS diagnosis highlight the four main key search areas that are mainly focused on the impact of PCOS detection.

Figure 1.1 Key Research Areas for Advancing PCOS Imaging.

Considering advancements in technology, precisely the underlying cause of PCOS continues unknown. Providing an early diagnosis thus is essential to prevent this [5]. Making modifications to one's lifestyle like losing weight and improving physical activity is one of the main approaches. For women who wants to manage menstrual cycles and reduce testosterone levels, doctors can recommend additional medications. Fertility treatments may be essential for women who want to eventually get pregnant. Surgery may be considered in particular circumstances to eliminate the cysts in the ovary.

PCOS detection therefore proves crucial. The successful functioning of families and the evolution of society are closely associated with the well-being of women. Advances in lifestyle, consumption patterns, surroundings, and social customs have all had major effects on women's declining ovarian wellness in recent decades [6]. Examining medical history, getting an examination of the body, conducting blood tests, undergoing an ultrasound scan, and ruling out other possible disorders are all common steps in the time and energy-consuming PCOS diagnostic approach. Fertility clinics are essential for the early discovery of PCOS because it directly affects ovulation. In addition, human errors can be added to these testing processes, which might result in a false diagnosis. It is crucial to guaranteeing the accuracy of PCOS diagnosis and, more broadly, the identification of any medical issue. The backbone of timely and successful treatments is reliable detection, allowing doctors to design personalized treatment plans [7]. Traditional evaluation is difficult and time-consuming because every facet of womanhood is intricate that can contribute to PCOS and the challenge of interpreting a sonogram cystic scanning.

1.2 Motivation

The substantial impact this condition has on women's overall wellness serves as the driving force behind this investigation. Among women of reproductive age, it is one of the most prominent hormonal illnesses. Effective PCOS treatment and management depend on a prompt and accurate diagnosis; yet, traditional screening methods sometimes entail intrusive procedures and subjective evaluations, which causes delays in diagnosis. The selection of reliable ML and DL models can support early problem diagnosis.

This research is completely in line with my goal of using ML/DL and image processing to solve critical healthcare problems. The capacity of both ML and DL models, such as convolutional neural networks (CNN) and transfer learning architectures, to process large volumes of data accurately is what motivates me to use them for automating PCOS identification. PCOS is among the most significant issues currently having an impact on women who are fertile.

This problem also leads to higher levels of stress and worry. To develop an ML model that can not only detect PCOS but also relate it to a woman's psychological well-being, a paradigm for early identification and prediction has been put forth. To balance a woman's stress and mental wellness, there is a high demand for improved processes and guidelines developed utilizing ML.

The use of transfer learning can significantly enhance classification performance by leveraging pre-trained models, saving time and computational resources. By evaluating key metrics such as Precision, Recall, and Accuracy, the model’s effectiveness can be validated and improved. My aim is to develop interpretable models for PCOS diagnosis, ensuring that predictions are transparent and understandable for both medical professionals and patients. This will help build trust in AI-assisted healthcare solutions and facilitate their integration into clinical practice.

1.3 Background

The information provided in this section is crucial for comprehending the general overview of PCOS detection. It lays the groundwork for using computational tools to improve PCOD diagnosis by highlighting important ideas in image processing, DL, and ML, including classification, segmentation, and feature extraction.

1.3.1 PCOS Detection

A doctor can identify PCOS (see Figure 1.2) by conducting a physical test, which requires laboratory tests that mention the clinical features of the patient, and undertaking the US. In the modern era of technology, PCOS US images are computed in 3D as well as 2D, where 3D tends to give its benefits of feature extraction and exploration and 2D its own. Different computation procedures were used to identify PCOS via these ovarian US images [8].

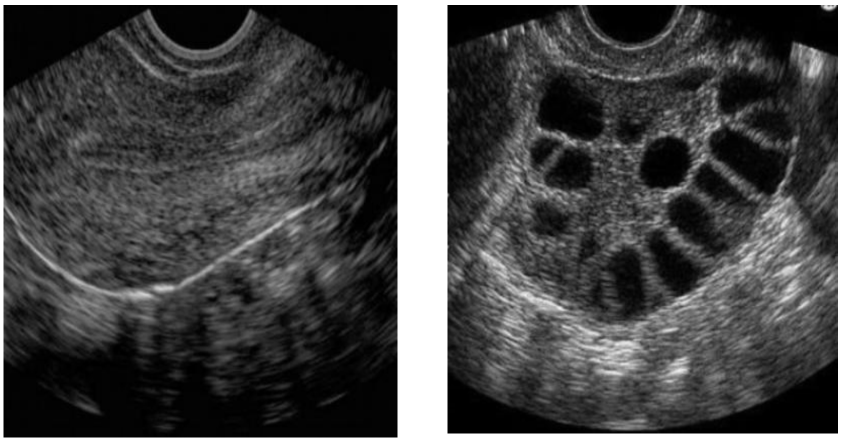


Figure 1.2 Ultrasound image of Normal Ovary and PCOS Ovary [8].

1.3.2 Machine Learning

ML is a branch of AI that encompasses the creation of models and algorithms that enable systems to acquire knowledge from and make selections based on facts. The discipline of ML has grown rapidly in recent years, with people adopting it at an extremely rapid pace. Many ML algorithms are present in today's time and have a great impact on the data they're applied to. From Figure 1.3, it can be seen that ML is divided

[9] into mainly three parts: 1) Supervised Learning: In this labeled training data is used where it is further categorized into classification and regression. In situations where the output value is categorical, classification algorithms are employed, such as identifying spam mail or not, whether disease is there or not, etc. Classification models consist of Random Forest (RF), Support Vector Machine (SVM), and K Nearest Neighbour (KNN) [10], and many more. Similarly, regression is used when the output variable is continuous or real, such as house price prediction or weight prediction. Regression algorithms consist of gradient boosting (GB), decision trees (DT), and linear regression (LR). 2) Unsupervised Learning: In this, the data is not labeled. We can make or search patterns and extract some useful insights from them. Unsupervised learning models include K-means clustering and many more. These algorithms are used in the healthcare industry and have a significant impact on it. 3) Reinforcement Learning: It is a technique that trains software to make selections resulting in optimal outcomes. It is based on the trial-and-error method that we people use to accomplish our goals.

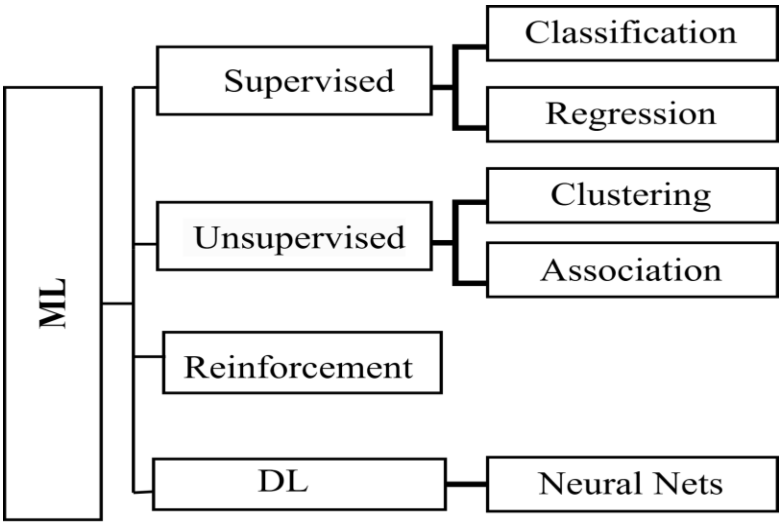


Figure 1.3 Types of Machine Learning.

1.3.3 Deep Learning

A branch of ML called DL [9] uses multilayered neural networks to learn data representations with several levels of abstraction. Here, the neural network is considered to be the heart of DL. This DL approach was based on the term neuron taken from the human brain, which strived to act and behave like humans. A brief of the various algorithms has been presented in the study that included DL such as CNN, VGG-Net, Rest-Net, Mobile-Net, and U-Net [9]. A deep neural network (DNN) consists of an input layer, several hidden layers, along with an output layer. As data moves across these layers, increasingly abstract elements are captured by each layer. The network learns to execute tasks with high precision by adjusting its internal weights through backpropagation, which iterates millions of times.

PCOS is a problem of binary classification (see Figure 1.4); diagnosing whether women have PCOS-causing symptoms or not is the main focus. Many ML algorithms, including LR, SVM, KNN, Classification and Regression Trees (CART), and Naïve Bayes (NB), were used to differentiate between US images that have PCOS and those that do not. DL algorithms such as CNN, DNN, Artificial Neural Networks (ANN), VGGNet16, U-Net, Gated Recurrent Unit (GRU), and many more were used for the detection of PCOS.

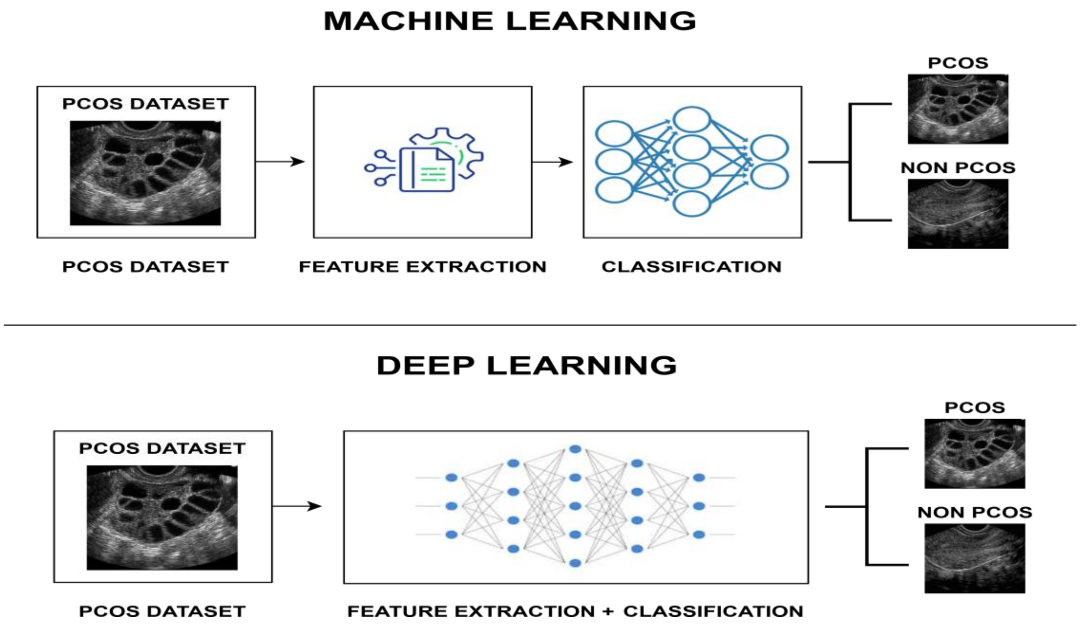


Figure 1.4 Architecture of ML and DL Models.

1.3.4 Image Segmentation

It is one of the types of image processing techniques that is widely used in the healthcare domain for segmenting images and performing object detection. It is the process of taking more information out of the image; the process of segmenting the image into smaller parts allows us to visualize and simplify the image. One of the most crucial processing phase processes is segmentation, which aids in identifying the follicles—the region of interest (ROI)—in the US image. Preprocessing techniques such image enhancement, background filtering, extracting features [10], the process of segmentation and classification are included in the image processing phases. The ROI from the US image can be found by applying particular feature extraction methods, such as Gray-level co-occurrence matrix (GLCM), and segmentation methods, such as detecting edges and threshold. Mainly, image segmentation is of two types, which are divided into subcategories, which Figure 1.5 illustrates.

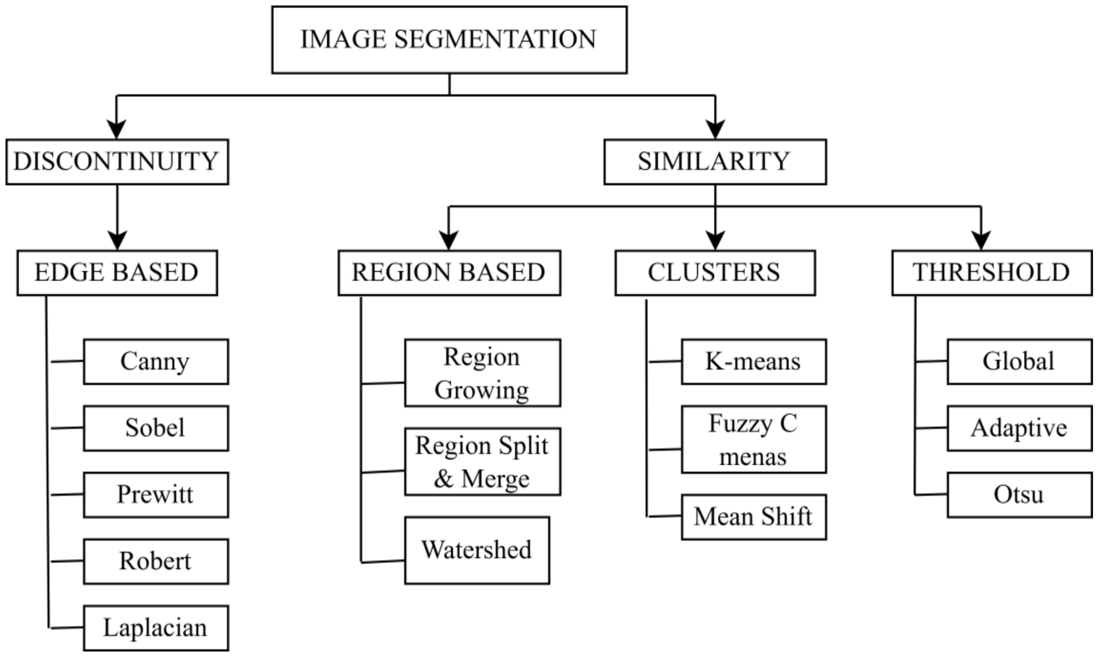


Figure 1.5 Types of Image Segmentation Techniques.

It is crucial for detecting PCOS, especially when examining US images to find ovarian follicles, which are important markers of the disorder. The number, size, and distribution of follicles can be precisely determined by separating and segmenting these areas, which can help with diagnosis. For ovarian structure segmentation, techniques including U-Net, thresholding, region expanding, and deep learning-based approaches like CNNs and Mask R-CNNs are frequently employed. Segmentation is an essential stage in automated PCOS identification because of these techniques, which improve diagnostic accuracy, lower human error, and allow for more thorough and effective analysis.

1.4 Structure of Thesis

The format of this thesis is as follows:

Chapter 2: Literature Review - This chapter reviews existing work on ML and DL methods used in PCOS detection in imaging and clinical feature datasets.

Chapter 3: Problem Statement: This chapter has covered the problem statement as well as the research goals that serve as the foundation for our thesis.

Chapter 4: PCOS Detection using Image Dataset - In this chapter, the methodology for detecting PCOS using US ovarian images as the dataset is discussed. The study's classification method was transfer-learning using the DL model.

Chapter 5: DL-Based Diagnosis of PCOS Using Clinical Data- This chapter discusses the process for identifying PCOS using clinical features as the dataset. For classification, the study employed a variety of DL models with different feature selection strategies.

Chapter 6: Nature-Inspired Algorithms for Clinical Feature-Based PCOS Classification- The method for diagnosing PCOS using clinical features as the dataset is covered in this chapter. Three different types of nature-inspired algorithms—GA, CSA, and FPA- were used in the study for classification.

Chapter 7: Conclusion and Future Scope: This chapter wraps up this thesis and offers a brief proposal for a type of future research that could be done.

Chapter

2. Literature Review

This chapter thoroughly examines and evaluates previous studies in the field of PCOS detection by using clinical feature data and image data for image segmentation and classification. The talk highlights the advantages, disadvantages, and applicability of several approaches, methods, and strategies investigated in recent research. At last, Gap analysis, which involves reviewing prior research to identify what has not been fully or adequately studied in the literature, was then covered in the study. This analysis serves as a basis for finding gaps. A summary of the body of research on PCOS identification is provided below.

Soni and Vashisht [11] presented a method for PCOS detection carrying out segmentation that used fundamental threshold techniques using Otsu thresholding and global thresholding using KNN for classification. Also, CNN was used for feature extraction for the follicle count. The adaptive K means approach was proposed by Sheikdavood and Bala [12], which uses the metaheuristic reptile search algorithm (RSA) to segment the cyst location, size, and measurements. CNN and DNN were utilized for classification. Gotte et al. [13] proposed the processing techniques is crucial for achieving accurate segmentation results. They stated a hybrid approach to denoising US image data, combining denoising CNN with the optimal Bayesian nonlocal means channel. Different filter methods, such as OBNLM, Gaussian, and DnCNN, were compared, along with a comparison of Sobel and Prewitt segmentation. The method for classifying ovarian cysts from a database of US images of women's ovaries was introduced by Narmatha et al. [14]. The Weiner filter was used for preprocessing, and CNN was used for extraction, where deep reinforcement learning and the Harris Hawk optimization (HHO) technique were applied. Using the sequential model, Ravishankar et al. [15] presented a unique approach to PCOS identification. The application of fuzzy convolutional neural networks (FCNN) was used for this approach, which helped in managing uncertainties and variations. Raja and Suresh [16] proposed a methodology using a 2D CNN. For image resizing, they used the BRISQUE methodology. They succeeded in getting 99.37% accuracy, and by maximizing the number of epochs, they were additionally able to achieve 100% accuracy.

Shao et al. [17] aimed to find the number of follicles in US photos, for which the use of Otsu with multilevel threshold segmentation was made. The differential search algorithm was used for optimization. The edges and corners of the cysts present in the US images were detected using the Gabor function, and the author employed bilateral filtering for noise reduction. When comparing the method to other existing techniques, the proposed approach outperformed them. The technique named Enhancer based on binary transition was proposed, which primarily concentrates on how well the US image segmentation was performed. The focus was to maintain the quality of the US images during preprocessing. The cyst segmentation process used the adaptive thresholding method, which divided the cyst into sub-blocks based on which the pixels were measured. The segmentation results were performed based on the following metrics: accuracy, dice coefficient, Jacobsen similarity index, and Matthew's correlation coefficient [18].

A new PCOS detection technique based on the Otsu Method of classification was presented by Rachana et al. [19]. Good results were obtained from feature extraction by using entropy, contrast, energy, and homogeneity as their parameters. A KNN classifier was used for the classification that gave 97% accuracy. Additionally, DT, NB, and SVM were used for comparative analysis, with KNN generating the best results. The identification of follicles through morphology and extraction techniques was suggested by Zulkarnain et al. [20]. Basic preprocessing techniques like image enhancement and segmentation and thresholding were used. Furthermore, techniques like erosion, dilation, and opening were used to extract the images' characteristics, and the US image's follicle count was calculated for each image based on several parameters such as entropy, standard deviation, homogeneity, and mean. The hybrid approach of the combination of classical Otsu Thresholding and Chan-Vese methods without using the Edge Method was presented for a count of follicles in US images. The binary image of Otsu thresholding was used as a mask to quantity the follicles using the Chan-Vese method. The primary evaluation metric utilized for this study was pixel-based [21].

Yilmaz and Ozmen [22] described the various processing techniques that can be used to improve the count of follicles in US images. They suggested two approaches, In the first, various preprocessing techniques were used, including the Wiener, Gaussian, average, and median filters, and histogram thresholding to filter out contrast inequalities. In the second technique, a wavelet transform was employed to smooth noisy images using the Lab* color space transform. K means clustering was used to categorize the PCOS detection. The Canny edge detection approach was used to segment the US pictures in order to count the follicles. Also, the Wiener and Gaussian filters gave better results for noise reduction. Maheswari et al. [23] discussed the significance of the pixel image representation for improved performance. They employed the Furious Fly Approach (FFO) for this, which involved adaptive histogram filtering, identifying the region ROI, and using a modified Furious Fly Approach for follicle identification to identify features. The classification was done using NB and ANN.

Suganya et al. [24] provided a method for categorizing US images of ovarian cysts into five groups. A hybrid approach for classifying image cysts using SVM and deep learning neural networks (DLNN) was proposed. When DLNNSVM and conventional DLNN were compared, the suggested method performed better when certain evaluation parameters such as F1-Score, recall, accuracy, and precision were taken into account. The authors have presented an intelligent approach to PCOS detection, taking into consideration and using both diagnostic and demographic features. A hybrid approach of classifiers, including ANN, SVM, and Latent Dirichlet Allocation (LDA) has been used for texture-based segmentation and feature extraction [25].

Nilofer et al. [26] discussed a successful approach that achieved 97.5% accuracy in classifying whether the US image is infected by PCOS or not by using an ANN. The authors used GLCM, a statistical method for feature extraction that considers the spatial relationship between the pixels, to examine the texture. Additionally, the meta-heuristic approach was used for optimization called improved fruit fly. Additionally, in 2021, [27] proposed a novel methodology using exception maximization optimization (EMO), which is used to segment the follicles from women's ovaries US images, where the Weiner filter was used to remove noise from salt and pepper. Using a variety of factors, including maximal probability, energy, contrast, homogeneity, and entropy for the statistical technique and the median, variance, skewness, and standard deviation for the GLCM feature extraction, features were extracted. Consequently, a hybrid intelligent water drop (IWD) via an extreme learning adaptive neuro-fuzzy inference system was employed for categorisation. To address the local minima issue for follicle detection classification, they implemented modified glowworm swarm optimization (GSO).

The study presented the CR-UNet hybrid approach, which combines a plain U-Net with a spatial recurrent neural network (RNN). Additionally, a different version of CR-UNet was provided by altering the minority base. The augmentation of the image was done for data processing. Also, the proposed method by Li et al. [28] was compared with numerous current approaches such as Deeplabv3+, CR-UNet, and U-Net.

Chen et al. [29] suggested a technique for follicular segmentation that makes use of DNN and edge information. They discussed how difficult it was to detect follicles from US images because the edges of the follicles in the US image were not clear. They used edge information and an attention-gate mechanism to develop the U-Net. For edge detection purposes, they used a canny edge detection operator using an attention gate model. A customized attention U-Net was presented using semantic segmentation. The basic idea behind this model was to label every pixel in an image and then use the segmented conclusions for further processing. Noise reduction was done by using bicubic interpolation. Modern CNN models are used to classify them [30].

Sarkar et al. [31] provided various approaches used today to identify ovarian cyst follicles in US images. For segmentation, they developed DC-UNet. Two contracting paths were added to the conventional U-Net to achieve high accuracy. The model produced the best results and outperformed all other U-Net-based existing models such as UNet, Residual U-Net, along with the Attention U-Net model. The study proposed a novel method for segmenting women's ovarian US images using harmonic attention-based U nets to quantify the size and count of follicles. Additionally, a brief discussion of the methodologies used to calculate the boundaries and shape of the cysts was explained. Three categories were used to categorize the image segmentation: ovary segmentation, follicle segmentation, and follicle count [32].

Choubey et al. created a two-phase image denoising technique for PCOS identification [33] using discrete wavelet transform. The primary goal was to produce noise-free images, which was accomplished by evaluating different noise parameters such as NLM and PNLM. The issue of overfitting was presented for which the various forms of data augmentation for preprocessing were used, such as rotating, shifting, crop, and flipping. VGG-16 was utilized for transfer learning. Furthermore, a comparison was conducted on a range of parameters, demonstrating that combining data augmentation and transfer learning yields good outcomes [34]. The use of AI in healthcare has been demonstrated by author [35].

In this study, the segmentation and edge detection methodologies were combined with the thresholding technique for segmentation. Darknet-19 produced the best classification model results. A hybrid approach for PCOS identification in US images was proposed by Bedi et al. [36]. It used adaptive bi-lateral filtering and attention residual U-net that guarantees its adaptability across both 2D and multimodal images. AresU-Net was utilized for improved segmentation outcomes, and SoftMax probability-based similarity was employed for classification. In the PCOS detection method, the cyst was identified using an adaptive CNN after noise was reduced using a guided trilateral filter. The Pyramidal Dilated Convolutional (PDC) method was utilized for categorization of the type of cyst, whether it is benign or malignant. Furthermore, optimization was done using Wild Horse Optimization [37].

The study aimed to provide an automated and effective method of determining the follicles using the methodology of size and count-based object detection (SCBOD). They suggested using an improved watershed algorithm by local minima for object selection and separation because, as we know, accuracy in determining the precise ROI is crucial. Various geometrical parameters were calculated, and then SCBOD was used for feature extraction and object detection along with KNN as a classifier [38]. The joint fusion method for PCOS classification was proposed, where images and images combined with clinical reports were compared and analyzed. Data augmentation, normalization of the pixel intensities, and adaptive histogram thresholding were used as preprocessing methods. Both a CNN fusion technique and DL model were used in their classification strategies. To classify data for the joint fusion methodology, they also employed transfer learning [39]. Patil et al. [40] performed the study of classifying the ovarian masses into benign, malignant, and PCOD. To complete the feature extraction process, Tamura, GLCM, and Gabor-like techniques were used. Furthermore, using RF for binary segmentation produced better results than KNN classifiers. Nakhua et al. [41] provided an ensemble learning approach to implement a novel PCOS classification technique. Several models, including RF, LR, gradient boosting, XGBoost, KNN, and DT, were combined to increase accuracy. Noise reduction, pipelining, and geometric transformation were all part of the preprocessing. The combination of XGBoost, KNN, and SVM was utilized for this approach. A comparative analysis was conducted using Table 1 derived from previous research articles using the

Image dataset.

Table 2.1 Summary of Existing Research on PCOS Detection using Classification and Segmentation for Image Dataset.

Ref	**Authors**	**Pre-Processing Steps**	**Feature Extracted**	**Algorithms Used**	**Accuracy**
[11]	Soni and Vashisht, 2019	Histogram Equalization, Image Enhancement, Global Basic and Otsu Threshold, Binarization of Image, morphology	CNN	KNN, Region Based, Watershed	-
[12]	Sheikdavood et al., 2023	Gray Scale Conversion, object carving method, Brightness, Contrast adjustment, Frost Filter	CNN, Kernel size, Area, Perimeter, Aspect ratio	Adaptive K-means Using Reptile Search Algo, DNN	99.1%
[13]	Gotte et al., 2020	OBNLM Filter, DnCNN Filter, Histogram Equalization, Canny Edge, Otsu thresholding	-	Hybrid filter using OBLNM and DnCNN	-
[14]	Narmatha et al., 2021	Weiner Filter, Harris Hawk Optimization	CNN	DQ Network	97%
[15]	Ravishankar et al., 2023	Image reshaping	CNN	Fuzzy CNN	98.37%
[16]	Raja and Suresh, 2023	Image Resizing – BRISQUE	CNN	2D CNN	99.37%

[17]	Shao et al., 2019	Bilateral Filter, Multilevel Threshold	Gabor Wavelet	Otsu And Differential Search Algorithm	-
[18]	Sheela et al., 2020	Adaptive Thresholding Segmented Image, Binary Transition	-	Enhancer Based on Binary Transition	-
[19]	Rachana et al., 2021	Gaussian filter, image binarization, enhancement, histogram equalization	Entropy, Contrast, Energy, Homogeneity	KNN and Otsu	97%
[20]	Zulkarnai n et al., 2022	Histogram Equalization, Average Filter	Morphological Operations – Erosion, Open and Dilation	-	-
[21]	Nazarudin et al., 2022	Grayscale Conversion, Median Filter, Histogram Equalization, Brightness, Contrast	-	Classical Otsu Thresholding and ChanVese Method Without Using Edge Method	89%
[22]	Yimaz et al., 2020	1st Method – Gaussian Filter, Histogram Threshold, Binarization, Morphology; 2nd Method – Wavelet Transform; Canny Edge	-	K Means Clustering	-
[23]	Maheswar i et al., 2020	adaptive histogram filtering	furious fly for follicle identification	ANN and NB	98.63%

[24]	Suganya et al., 2022	Image augmentation, image resizing	-	DL With SVM	97%
[25]	V et al., 2023	Discrete Wavelet Transform, l*a*b* Colour Space Transformation	Auto Corelation, Sum Average, Sum Variance, Intensity	SVM, Sequential Minimal Optimization	98%
[35]	Sumathi et al., 2023	Histogram Equalization Canny, Sobel, Prewitt, Robert edge-based detection.	GLCM Properties	Darknet-19, Region Threshold.	99%
[36]	Bedi et al., 2024	Adaptive Bilateral Filtering	CNN Such as Residual U-Net or ResU-net	SoftMax Probability Based Similarity, Attention Residual Unit Model	2D- 97.7%, Multimodal - 98.1%
[37]	Sha, 2024	Guided Trilateral Filter, Wild Horse Optimizer	AdaResUNet and PDC Network Via Dilated CNN	PDC Network, Adaptive Convolutional Neural Network	98.87%
[38]	Jeevitha et al., 2022	Image pixel conversion, Image adjustment, Image enhancement	size, count, mean, standard deviation	Watershed Algorithm using SCBOD method	-
[39]	Alamoudi et al., 2023	Pixel Normalization, Adaptive Histogram Equalization, Data Augmentation	Inception V3	Joint1- MobileNet, Joint2- VGG-19	84.81%
[40]	Patil et al., 2024	Block Matching3D Filter, 3D Discrete Wavelet and Cosine	GLCM, Tamura, Gabor, and	Binary Segmentation with RF	86%

		Transform	Edge Feature		

In a similar vein, a thorough literature analysis on PCOS detection with clinical feature-based datasets was carried out, emphasizing ML and DL techniques for prediction and classification. This section provides an organized summary of study results that used feature selection and ML approaches to classify various clinical PCOS datasets.

A web-based ML approach was proposed by Rahman et al. [42] using Django framework as an interface. Mutual information criteria were used for feature selection. AdaBoost and RF gave the best accuracy, 94\%. Cicek et al. [43] presented another ML approach for PCOS classification. For model interpretability and explanations, the authors used the LIME technique. The RF model gave the highest accuracy of 86.03\% and sensitivity of 86.32\%. Faris et al. [44] in their study selected features using the feature correlation method. SVM, NB, DT, and KNN were the models employed for classification; the KNN classifier got the highest accuracy of 89.51\%.
Khanna et al. [45] presented a study in which they employed three distinct feature engineering methodologies. They used the mutual information algorithm, the Salp Warm Optimisation (SWO) algorithm, and the Harris Hawk Optimisation (HHO) to choose features based on their efficiency of the top 15 features. For classification, a total of 12 ML models were used, including LR, SVM, NB, DT, KNN, RF, XGBoost, and AdaBoost. Three multistack models were developed where STACK-1 is made of 7 classifiers LR, KNN, NB, and SVM (linear, polynomial and sigmoid kernels); STACK-2 is made of ensemble models consisting of XGBoost, AdaBoost, DT, RF, and extra trees and STACK-3 were developed using a multi-level stack of STACK-1 and STACK-2. The final multi-stack ML model gave the highest accuracy of 98\%.
Sreejith et al. [46] developed a clinical decision support system to help doctors monitor PCOS. The Red Deer Algorithm (RDA) was used to optimize the best features after using a wrapper-based selection method for feature selection. The RF classifier was used for classification that achieved an accuracy of 89.81\%. A DL-based approach was introduced for automated PCOS prediction. The study [47] used the Pearson correlation coefficient (PCC) for feature selection and SMOTE (Synthetic Minority Oversampling Technique) for

class balancing. Three models named CNN, long-short-term memory (LSTM), and a combination of CNN and LSTM models were used to train on the dataset. The SMOTE method with the CNN model gave the best convergence compared to others by achieving the highest accuracy of 96.59\%.

Vairachilai et al. [48] presented a hybrid ML approach for PCOS detection using PCC as feature selection. For data balancing, SMOTE was utilized. Many ML and ensemble models, such as LR, SVM, KNN, DT, RF, NB, XGBoost, AdaBoost, and CatBoost, were used for classification. LR gave the best results achieving the highest accuracy of 91\%. Also, among ensemble models, the blending algorithm achieved 91\% accuracy. A comprehensive ML framework was proposed by Kumar et al. [49]. They used Particle Swarm Optimization (PSO) for feature selection. For PSO hyperparameter change, grid search cross-validation (CV) was used to achieve the best results. LR, SVM, NB, KNN, DT, RF, XGBoost, and AdaBoost were among the machine learning algorithms utilized in the study. The PSO algorithm approach was used for optimization, and LR provided the best accuracy of 96.30\%.

Elmannai et al. [50] presented another approach using optimized feature selection along with integration of XAI. For feature selection, the study used mutual information, tree-based, and recursive feature elimination (RFE) methods. The hyperparameter tuning was done using the Bayesian Optimization (BO) algorithm. For class unbalancing, SMOTE and Edited Nearest Neighbor (ENN) hybridization were employed. They used XGBoost, AdaBoost, RF, and NB classifiers to create a stacking ML model utilizing the RFE approach. Additionally, they merged the RF, SVM, AdaBoost, and XGBoost models for the tree-based approach. When compared to other models, the findings indicated that, with a 100\% accuracy rate, the Stacked ML using RFE selection of features model was the most accurate.

Moral et al. [51] proposed the PODBoost named XAI model for classification. To choose the most important characteristics from the dataset, the authors employed a metaheuristic feature selection method based on the GWO algorithm. For balancing dataset, SMOTE-Tomek Links (SMTL) technique was used. The explainable model enhances the efficiency by combining the advantages of bagging and boosting techniques. The PODBoost model achieved an accuracy of 97.42\%. Additionally, the LIME technique for model explainability was applied.

By developing the SmartScanPCOS model, a feature-driven approach was proposed by Mahesswari et al. [52]. The study used three different feature selection methods to achieve the optimal results, i.e., Threshold-driven Optimized Principal Component Analysis (TOPCA), SSO, and Threshold-driven Optimized Mutual Information Method (TOMIM). ML models used for classification include RF, SVM, NB, KNN, LR, DT, AdaBoost, and extra trees classifiers. The two-level RF classifier, which used the top 17 features determined by TOMIM, achieved the highest accuracy of 99.31\%. At last, the Shapely and the QLattice technique was used for model explainability and interpretations.

Mehr et al. [53] used three distinct feature selection techniques to classify data using four ML models. Approaches such as filtering, wrapping, and embedding were used. The study employed AdaBoost, MLP, RF, and extra trees as classification models. The RF classifier gave the highest results by achieving 98.89\% accuracy along with 100\% sensitivity using the embedded feature selection method. The proposed model performed better than existing models, including extra trees, AdaBoost, and MLP.

Table 2.2 Summary of Existing Research on PCOS Detection using Classification for Clinical dataset.

Ref	**Authors**	**Pre-Processing Steps**	**Feature Selection**	**Algorithms Used**	**Accuracy (%)**
[42]	Rahman et al.	Duplicate removal, filling null values	Mutual information	AdaBoost and RF	94%
[43]	Cicek et al.	-	Chi-Square test	RF	86.03%
[44]	Faris and Miften	Missing values removal	feature correlation method	KNN	89.51%
[45]	Khanna et al.	Duplicate removal, filling null values and SMOTE	SWO, HHO and Mutual information	Multi-stack of ML model	98%

[46]	Sreejith et al.	Irrelevant features removal	RDA using wrapper method	RF	89.81%
[47]	Ahmad et al.	SMOTE, Pearson correlation coefficient, Outlier detection	Pearson Correlation coefficient	CNN with SMOTE	96.59%
[48]	S. et al	SMOTE, correlation feature selection and filling null values	Pearson Correlation coefficient	LR	91%
[49]	Kumar et al.	Exploratory data analysis (EDA)	PSO	LR	96.30%
[50]	Elmannai et al.	SMOTE and ENN, Filling Missing Values	mutual information, tree based and RFE	Stacked ML using REF	100%
[51]	Moral et al.	SMOTL, Removing Missing Values and Standardization	GWO	Using bagging and boosting	97.42%
[52]	Umaa et al.	Removing Missing Values	TOPCA, SSO and TOMIM	Two-Level RF	99.31%
[53]	Homay et al.	Removing Missing Values	Filter, wrapper, and embedding methods	RF	98.89%

2.1 Gap Analysis

A summary of both the DL and ML models applied to PCOS is presented in Tables 2.1 and 2.2 using imaging and clinical features. It is observed from these tables that while extensive work has been done using ML and DL models, the aspect of model interpretability remains limited. This highlights the need for developing more reliable and explainable models, which is especially crucial in sensitive domains like healthcare. Therefore, this work aims to apply transfer learning and develop interpretable models for PCOS detection that can support practical clinical use and ultimately enhance patient care and quality of life.

Chapter

3. Research Question or Problem Statement

This chapter clearly defines the problem that motivates the study, laying the groundwork for an orderly analysis of its multiple components. The process begins with the formulation of the Research Question, which outlines the specific inquiry that guides our investigation. The objectives of solving the problem are discussed at last, with an emphasis on the theoretical understanding and useful outcomes that will result from dealing with this problem.

3.1 Problem Statement

PCOS is a condition of hormonal imbalance which is becoming increasingly prevalent in today's women. A large number of small fluid-filled sacs called cysts, which are also known as follicles, grow around the edge of the ovary in women's ovaries, which are the main cause of this medical condition. The exact underlying cause of PCOS remains unknown despite technological developments. US scans to identify numerous follicles are an efficient way to diagnose PCOS early and schedule treatment. Also, doctors conduct physical tests of the patients which generate a report of the clinical features based on which again the PCOS can be diagnosed. The interdisciplinary uses of AI and related technologies are rapidly evolving and transforming various areas. The selection of reliable ML and DL models can support early problem diagnosis. PCOS is a problem of binary classification; diagnosing whether women have PCOS-causing symptoms or not is the main focus.

3.2 Objectives of the study

The current study's main objective is to develop a precise and trustworthy methodology for identifying PCOS by employing ultrasound images and the clinical features dataset by leveraging ML and DL algorithms. The objective of the study is to:

3.2.1 Existing Literature Survey on PCOS Detection

- To carry out an extensive review of the existing works on PCOS detection, with an emphasis on image classification and segmentation methods.

- To examine several techniques and algorithms implemented in the area to determine the best methods for precise PCOS identification.

3.2.2 PCOS Detection using Image Dataset

- To propose a transfer learning-based DL approach for classifying women with PCOS
- To evaluate the performance of the algorithms using metrics accuracy, precision, recall, F1-score.

3.2.3 PCOS Detection using Clinical Features Dataset

- To utilize nature-inspired computational algorithms for feature selection such as GA, FPA, and CSA using soft voting for classification purposes.
- To apply XAI techniques to the model which possesses the highest accuracy.

Chapter

4. PCOS Detection using Image Dataset

This chapter presents an overview of the method used to develop a solution to the research problem of designing an ML/DL approach for PCOS detection using Image data. The section presents the conceptual basis, algorithmic structure, dataset information, and model implementation procedures.

4.1 Overview

To increase diagnostic accuracy and efficiency, we proposed a DL method based on transfer learning for PCOS classification using US dataset. The primary objective was to determine whether or not a woman has PCOS without the supervision from a physician.

4.2 Methodology

The manual process of analysing US images takes a lot of time and is quite prone to human error. To automate the detection process, we propose transfer learning-based classifiers for PCOS identification using US images. In this work, InceptionV3 and ResNet50, the two pre-trained models, were used. For this study, the dataset from Kaggle was used. The Kaggle dataset was further customized by us for better results. The objective is to determine whether the US images of the ovary show PCOS infection or not. The framework of our study method is shown graphically in the block diagram that follows in Figure 4.1. Below is a brief description of the stages this study involved:

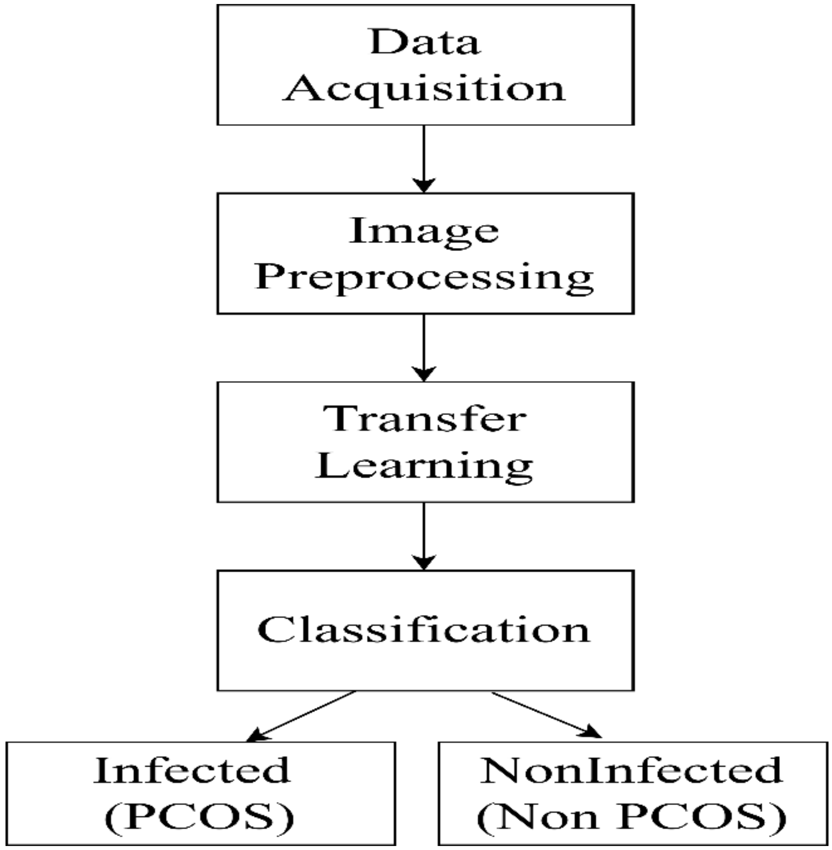


Figure 4.1 . Flowchart of the proposed work.

4.3 Dataset Description

The dataset that was used in the form of the US ovarian images was taken directly from the Kaggle repository [54]. The dataset was separated into two folders: the train folder and the test folder. There were 1,924 frames in the train set and 1,932 frames in the test set. However, due to their striking similarities, the test set was removed and simply the one for training was used. From the train set folder 781 images were the cystic US images labeled as ‘infected’ and 1,143 US images of healthy ovaries labeled as ‘not infected’. Additionally, to the given dataset 610 images were added from [55]. Also, the dataset was manually separated into folders for testing, validation, and training. PCOS-afflected US photos were marked as "infected," while non-PCOS photos were marked as "not infected.". The train set included 1,911 images, the validation set included 302 images and the test set consisted of 320 images. Figure 4.2 displays a sample of the dataset's US images.

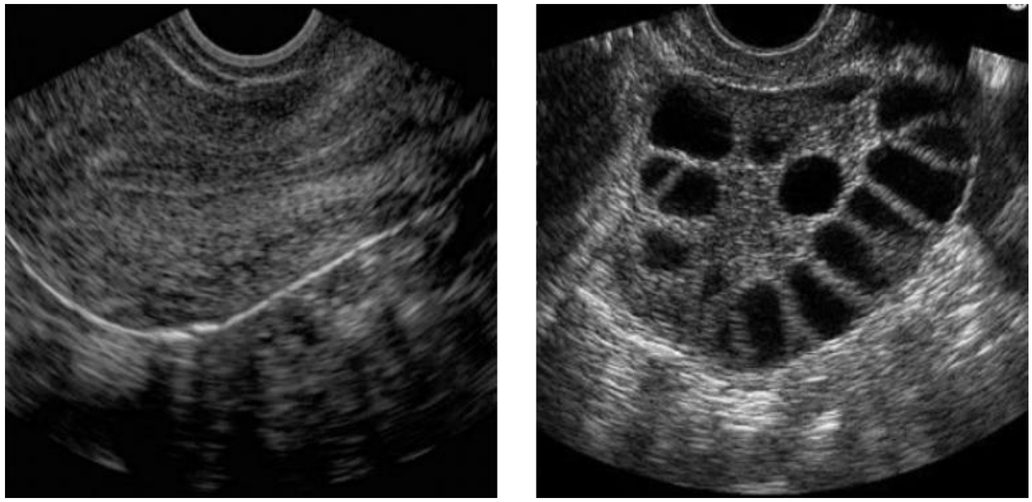


Figure 4.2 Normal Ovary and PCOS Ovary US Image.

4.4 Data Preprocessing

A crucial step in enhancing data before it enters the model training stage is preprocessing. This contributes to the improved quality of the data. For pre-processing, we first resized all the images into 224×224 pixels to provide the model with uniform dimensions. To normalize our image data, we used Kera's ImageDataGenerator for this purpose where the range [0, 1] was used to scale the pixel values for ensuring consistency with model training data. Each of the three training, test, and validation sets contained 16 batches, and the class mode was set to binary. In order to expose the model to a wider range of variability found in US images, data augmentation was used to artificially expand the dataset. To achieve this, data augmentation techniques included a horizontal flip, a 0.1 zoom range, and a 0.1 shear range.

4.5 Transfer Learning

Reusing prior models to address present problems is known as transfer learning, which falls under ML. It is a method used to train algorithms together. It is not necessary to create the solution from scratch because the fundamentals of preexisting training data are used to improve the performance of the current challenge [56]. It enables us to apply the knowledge we've gained from one area of expertise to another. This study looks at the possible uses of transfer learning using the InceptionV3 and ResNet50 model to identify PCOS in US images.

4.6 InceptionV3 Model

The very first version of InceptionV3 was inspired by the 2012 ILSVRC classification concurrence [57]. Google developed this model, which was made available in 2015. It is made up of a stack of layers that interpret incoming images differently, and it is based on the Inception architecture. As it is made up of a stack of layers that interpret incoming images in multiple ways, allowing it to discern properties at various scales and under varied conditions. The model includes a pre-trained network version from the ImageNet database that was trained on over a million pictures. It has 42 layers based on Inception V3 standards. The classification test is passed by the model's final layer after a series of convolutional, dropout, and pooling layers (see Figure 4.3). The network has been trained on over a million photographs and is accessible as a pre-trained version in the ImageNet database. A trained network is liable for categorizing images into 1,000 different categories of objects, including several animals, mouse pads, keyboards, and pencils. This enables it to distinguish between features in various contexts and at various scales. The network has accordingly developed deep representations of features for many different kinds of images. These layers are designed to represent several features of the input photos, such as structure, edges, framework, and texture. The model incorporates the results of these layers to identify the classification or grouping of the image under analysis. The size of the image input for the neural network is 299×299 pixels. Since the binary classification was the aim, the output layer used the sigmoid activation function and had dimensions of 1x1x1000 due to the ILSVRC dataset's 1000 image classes. The original image size was 224×224 pixels. Using the transfer learning technique, all of the layers in the previously trained base model were removed and rendered non-trainable. In other words, their weights won't be changed as they train. Three dense layers were used in the approach using relu as an activation function. Given that our classification problem was binary, our output layer only had one neuron. Model weights were updated efficiently by applying the Adam optimizer, which avoided making major adjustments that would cause training to become unbalanced with 'binary cross entropy' as the loss function.

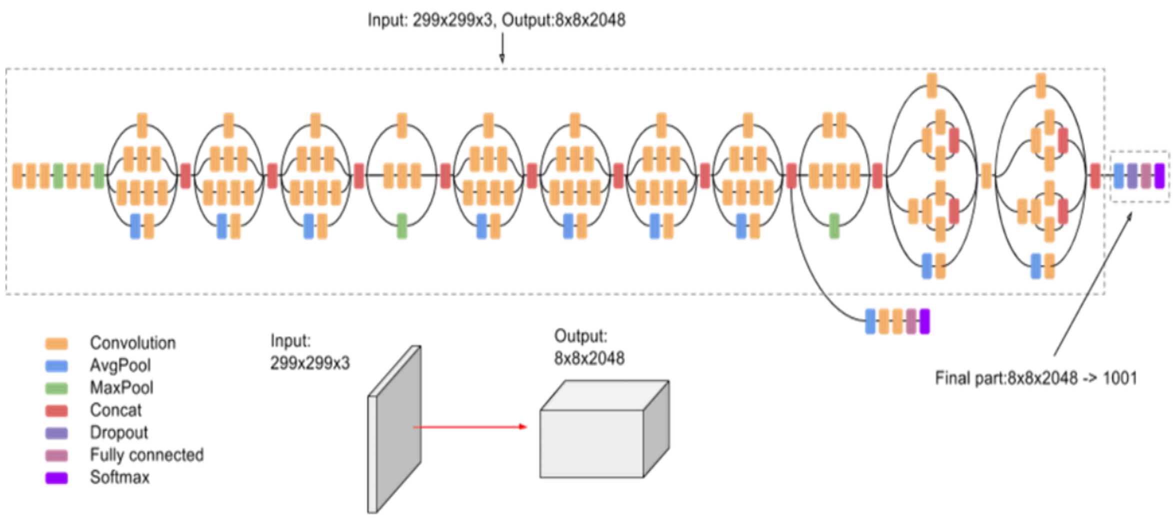


Figure 4.3 InceptionV3 Architecture [57].

4.7 ResNet50 Model

Microsoft Research introduced the CNN model referred to as ResNet50 [58] in 2015. It is an improved version of the widely used" Residual Network," or ResNet, architecture. Many different versions of ResNet function on the same principle but differ in the number of pooling layers. There are fifty layers in the network's structure, and the number "50" in the name signifies the number of layers (see Figure 4.4). The ImageNet dataset, which comprises more than fourteen million photos and 1000 classes, was used for learning it. Diverse representations of attributes for a variety of images have therefore been trained by the neural network. The input image size for a neural network is 224×224. The use of residual connections, which enables the network's components to gain knowledge of a variety of residual functions that correspond to the input to the expected output, is one of its main advances. Without having to deal with the challenge of vanishing gradients, a neural network may develop far more profound designs because of these residual connections. The convolutional layers extract features from the input image, analyzed and modified by the identity block and convolutional block, and ultimately classified by the fully connected layers. ResNet50 is an efficient model for various kinds of imaging classification tasks. The degrading problem associated with deep neural networks was the primary problem that ResNet resolved. The accuracy of connections becomes saturated and then gradually decreases as they get deeper. Overfitting is not the reason for this decline; actually, it is an issue of training process optimization. Similarly, all the top layers in the pre-trained base model of the ResNet50 model were removed, they were kept non-trainable. The input image size used was 224×224 pixels. Two dense layers were used having 256 neurons along with relu as an activation function. The activation function of the output layer used a sigmoid with a dropout of 0.5 to prevent overfitting. Early pausing was also used to prevent overfitting of the training model.

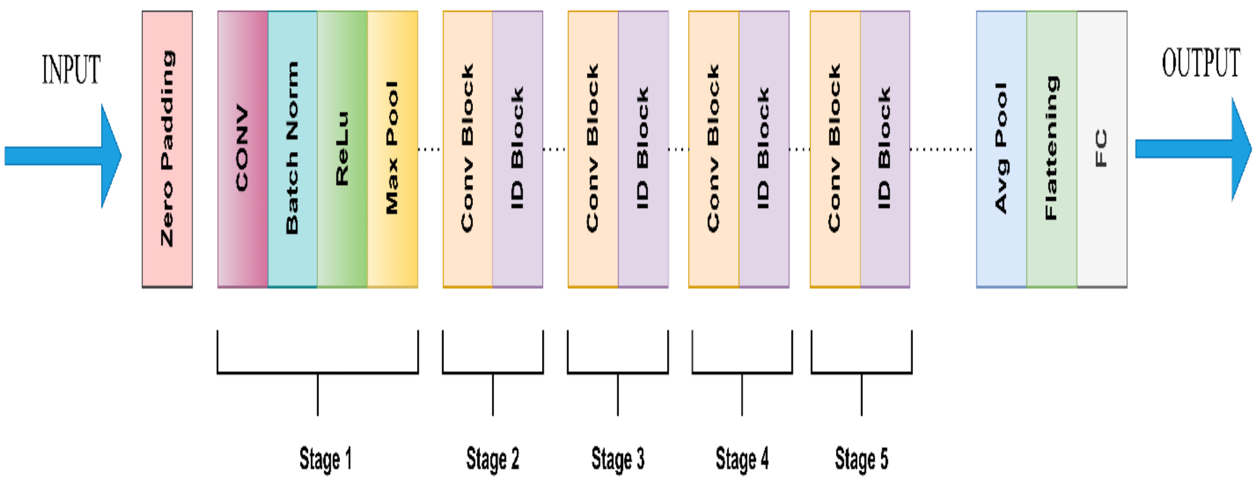


Figure 4.4 ResNet50 Architecture [58].

4.8 Experimental Results

Evaluation metrics are used in ML to evaluate how well the model works. Many metrics, including as the confusion matrix, F1 score, recall, accuracy, and precision, were used for this study. False positives (FP), false negatives (FN), true positives (TP), and true negatives (TN) are the conventional metrics used to evaluate the effectiveness of an algorithm. TP demonstrates the accuracy of the expected and actual classes. TN demonstrates that the actual class and the expected class are both wrong. According to FP, the actual class is incorrect, even though the intended class is correct. FN indicates that the actual class is right while the projected class is wrong. The result of all the work is the accuracy plot, demonstrating the model's accuracy over several iterations. The overall percentage of accurate predictions across all data values is represented by accuracy.

Precision is the number of positive outcomes that were right. Recall is the percentage of true positive data that was anticipated correctly. The F1 Score is determined as the harmonic mean of precision and recall.

Table 4.1 Performance Metrics of Transfer Learning Models.

Metrics	InceptionV3	ResNet50
Accuracy (%)	99.68	97.5
Precision (%)	100	95.23
Recall (%)	99.37	100
F1 Score (%)	99.68	97.56

The quantitative performance of both models was evaluated using precision, F1 Score, accuracy, and recall. The labels 'infected' for PCOS images and 'not infected' for normal ovarian images were applied to these images. The ResNet50 model obtained 97.5% accuracy on the test set, while the InceptionV3 model achieved the highest at 99.68% (see Table 4.1). On the basis of this test set, the confusion matrix over both models was plotted as seen in Figure 4.5, and Figure 4.6 displays the plotting of both training and validation sets' accuracy for both models.

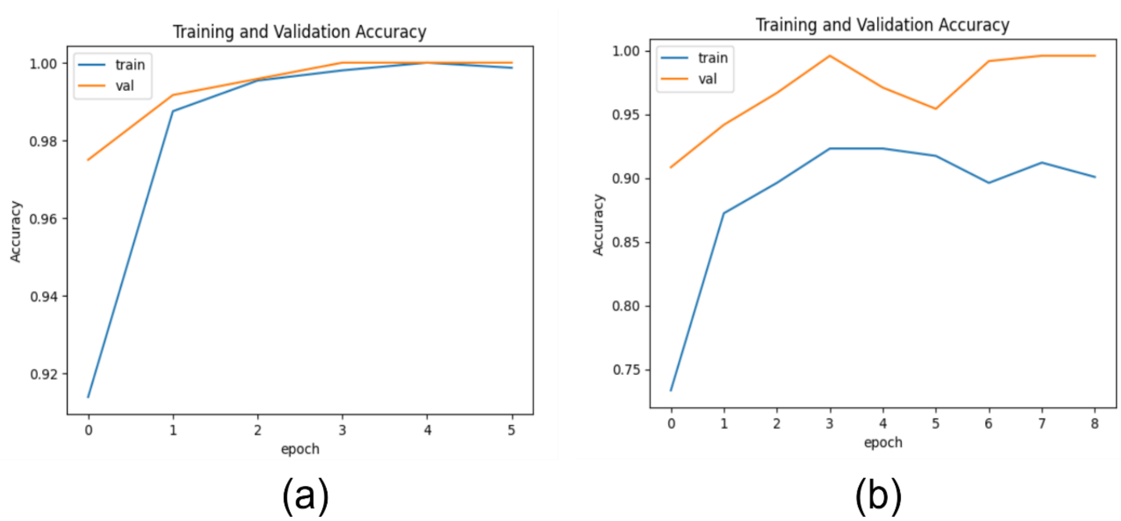
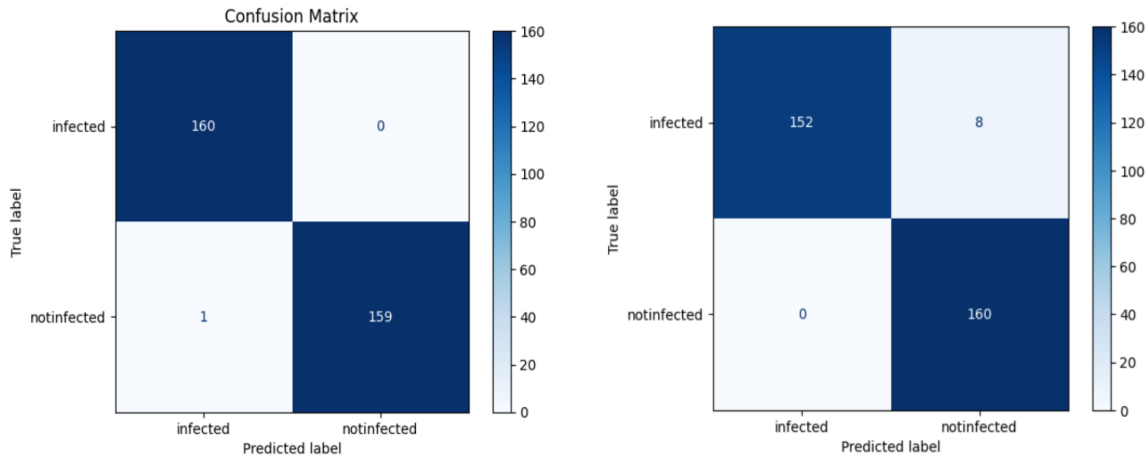


Figure 4.5 Training and Validation accuracy Graph of (a) InceptionV3 and (b) ResNet50.



(a) (b)

Figure 4.6 Confusion Matrix of (a) InceptionV3 and (b) ResNet50 Model.

4.9 Discussion

In summary, the study presents a comprehensive method for detecting PCOS using a transfer learning approach. For this, two pre-trained models were used, i.e. InceptionV3 and ResNet50 model out of which InceptionV3 gave the best outcome resulting accuracy of 99. 68%, F1 score of 99. 68%, recall of 99. 37%, and with 100% precision. This finding is comparable with earlier research that detected PCOS using transfer-based deep learning. The top layers of the models were frozen, making it non-trainable, and using three dense layers the model was fine-tuned, also using kernel regularizer. It can be used to identify affected ovary US images in medical establishments, which helps with early PCOS findings, which is crucial for reducing the syndrome's severe signs.

Chapter
5. DL-Based Detection of PCOS Using Clinical Data

This chapter presents an overview of the method used to develop a solution to the research problem of designing an ML/DL approach for PCOS detection using a Clinical Dataset. The section presents the conceptual basis, algorithmic structure, dataset information, and model implementation procedures.

5.1 Overview

In order to classify PCOS from tabular clinical data, we used a multifaceted approach in this chapter that combines various feature selection techniques with ML algorithms and DL algorithms. We then used these updated feature sets to train various ML-based algorithms for deep learning for tabular data. Our strategy seeks to optimize diagnostic accuracy and robustness by fusing feature selection with data augmentation techniques, proving the value of fusing data-driven selection with cutting-edge classification methods.

5.2 Dataset Description

We used the Kaggle dataset [59] for PCOS, which includes 541 cases and 45 characteristics. The dataset contains information on 541 women in total, 177 of whom have PCOS and 364 of whom do not. Ten hospitals spread around the Indian state of Kerala were employed to carefully gather the PCOS-focused data. This useful database, which comprises a wide range of clinical and physical characteristics, became available to the public in 2020, to be published. The dataset consists of two files, after which PCOS data with and without infertility were integrated, and any extraneous columns were eliminated. The results are shown in the "PCOS (Y/N)" column, where a value of 0 denotes a negative result and a value of 1 indicates that the person has PCOS. Details of the database characteristics and their types are displayed in Table 5.1.

Table 5.1 Overview of the dataset.

Sr. No.	Feature	Dtype	Sr. No.	Feature	Dtype
1	Sl. No	int64	23	Waist(inch)	int64
2	Patient File No.	int64	24	Waist: Hip Ratio	float64
3	PCOS (Y/N)	int64	25	TSH (mIU/L)	float64
4	Age (yrs)	int64	26	AMH (ng/mL)	float64
5	Weight (Kg)	float64	27	PRL (ng/mL)	float64
6	Height (Cm)	float64	28	Vit D3 (ng/mL)	float64
7	BMI	float64	29	PRG (ng/mL)	float64
8	Blood Group	int64	30	RBS (mg/dl)	float64
9	Pulse rate(bpm)	int64	31	Weight gain(Y/N)	int64
10	RR (breaths/min)	int64	32	hair growth(Y/N)	int64
11	Hb(g/dl)	float64	33	Skin darkening (Y/N)	int64
12	Cycle(R/I)	int64	34	Hair loss (Y/N)	int64
13	Cycle length(days)	int64	35	Pimples (Y/N)	int64
14	Marriage Status (Yrs)	float64	36	Fast food (Y/N)	float64
15	Pregnant(Y/N)	int64	37	Reg.Exercise (Y/N)	int64
16	No. of abortions	int64	38	BP _Systolic (mmHg)	int64
17	I beta-HCG (mIU/mL)	float64	39	BP _Diastolic (mmHg)	int64
18	II beta-HCG (mIU/mL)	float64	40	Follicle No. (L)	int64
19	FSH (mIU/mL)	float64	41	Follicle No. (R)	int64
20	LH (mIU/mL)	float64	42	Avg. F size (L) (mm)	float64
21	FSH/LH	float64	43	Avg. F size (R) (mm)	float64

22	Hip(inch)	int64	44	Endometrium (mm)	float64

This dataset contains a wide range of medical information. It includes vital medical indicators such as blood group, respiration, pulse, blood levels, and body mass index (BMI) of the patient. It also contains individually identifiable information about an individual, such as height, weight, and age. Additionally, it includes information about sexuality, including period traits, time since marriage, past pregnancy history, and hormone levels of beta-HCG, FSH, and therefore LH. It also contains heart health information such as diastolic and systolic blood pressure; lifestyle factors such as changes in weight, hair growth, skin problems, thinning hair, and breakouts, as well as dietary patterns and daily physical activity, are tracked. Their physical characteristics, such as the thyroid's health (TSH), the ratio of the stomach to the hips, and measurements of the pelvis and abdomen, are also considered. Among the other factors covered in the report are the levels of AMH, prolactin, vitamin D, progestin stages, ranging, and random blood sugar (RBS) results. Lastly, data on the size of the endometrium and ovarian follicles reveal details about the reproductive health of women.

5.3 Data Preprocessing

To provide the best possible input for the model, it is crucial that the data be preprocessed [60] before the suggested algorithm is used. Preprocessing enhances performance and increases the prediction model's dependability by eliminating noise, correcting inconsistencies, and standardizing. In this work, the main preprocessing processes were identifying and removing outliers and then normalizing the data to put all the features on a similar scale. The detailed discussion of these steps is as follows.

NumPy assisted in identifying missing values in the characteristics "Marriage Status (Yrs)," "AMH (ng/mL)," and "II beta-HCG (mIU/mL)." The median has been used to fill in the missing numerical attributes of the data. The statistical mode of the classification variable "fast food (Y/N)" was used for data imputation. We eliminated the attributes "SI. No." and "Patient File No." since they were not statistically significant. A min-max normalization technique was applied to adjust for dataset scaling in order to preserve model stability and ensure acceptable feature contributions. Outliers have been visualized with the use of Python's Matplotlib module of software. Outlier detection and management can increase model and statistical analysis accuracy. All 541 samples were taken into consideration, and there were no significant outliers. Due to the relatively small dataset size (N = 541) and a slightly high number of features, the models were more prone to overfit, so the stratified 5-fold cross-validation was performed using the StratifiedKFold technique. It ensures constant performance estimates and class balance across folds by enhancing model generalization and by reducing uncertainty caused by sparse data. Each preprocessing step was applied inside each fold using a pipeline to prevent data leaks. The training set comprises 80\% of the dataset, and the testing set comprises 20\%. A random state of 42 was applied to each of the two halves of the dataset. Testing sets have been used to assess the models after they have been trained and optimized using training sets. For training and testing, the complete set is ultimately divided into four distinct subsets: x_test, x_train, y_train, and y_test.

5.4 Data augmentation Techniques

In order to address the problem of class imbalance in the dataset through data augmentation, a number of resampling approaches were employed in this work. In order to produce a balanced dataset and enhance model performance and generalization, these tactics mainly aim to either reduce instances within the majority class or produce newly synthetic samples within the minority class [61]. The following methods were applied:

- *SMOTE* - SMOTE interpolates between pre-existing minority samples to create artificial examples for the minority class. This reduces the bias towards the majority class and provides the classifier with more representative training data.
- *Random UnderSampler*- By deleting samples from the dominant class at random, this technique balances the dataset. Despite its simplicity, it may result in the loss of potentially valuable information from the dataset.
- *Random OverSampler*- Instances in the minority class are randomly duplicated using this method. Even though it can improve performance for unbalanced datasets, it might cause the dataset to lose information that could be useful.

- *ADASYN*- In order to improve the classifier's decision bounds, ADASYN expands on SMOTE by concentrating more on creating synthetic data targeting minority class samples that are more challenging to learn.
- *SMOTE-ENN*- After using SMOTE for over-sampling, this hybrid approach uses the ENN method to clean the dataset by removing ambiguous cases. It helps to reduce noise and streamline the boundaries between classes.
- *SMOTE-Tomek*- Additionally, this method integrates Tomek connections—the closest pairs of samples from different classes—with SMOTE. Following the SMOTE application, these pairs are eliminated to reduce overlapping and tidy the class boundary.

5.5 Feature Selection

This subsection gives an in-depth explanation of all the optimization algorithms (nature-inspired algorithms) used for feature selection. By identifying the most pertinent features, feature selection greatly enhances model performance while lowering computational complexity.

The process of selecting the most crucial features from the initial set in order to create models that are more precise and effective is known as feature selection. In data preparation, feature selection is often utilized to find relevant attributes that were not previously identified and thus eliminates the redundant or irrelevant features that are not essential to the purpose of classification. Enhancing classification accuracy is the goal of feature selection

5.5.1 Filter method

This method gives scores to all the features and is independent of the learning model [62]. Scoring methods include:

1. *Chi-Squared Test*: It calculates the independence between input features and target variables.

2. *Fisher Score*: It evaluates the discriminative power of features.

3. *Mutual Information*: It calculates the information that the goal and input features have in common.

4. *Correlation-Based Feature Selection (CFS):* It selects features on the basis of their correlation with the target variable.

5.5.2 Wrapper method

It is used to evaluate traits with a specific search technique and learning model. The computation is more expensive [63]. Examples include:

1. *Sequential Forward Selection (SFS):* Depending on performance gains, it provides features one at a time.

2. *Recursive Feature Elimination (RFE):* It removes the least important features iteratively.

3. *Sequential Backward Selection (SBS):* It removes features one at a time based on performance deterioration.

5.5.3 Embedded Approaches

This method integrates the feature choice into the classifier's training procedure. To ascertain the feature significance, it makes use of the model's internal algorithms [63]. Examples include Lasso regression, where L1 regularization shrinks less important features.

5.6 Machine Learning Algorithms

This subsection provides an explanation of all the ML models [64] employed in this thesis for PCOS Detection. These algorithms are crucial for deciphering detailed patterns in large datasets that aid in the diagnosis of PCOS.

5.6.1 Support Vector Machine

SVM is one type of supervised ML algorithm used in regression, categorisation, and recognising outlier's applications. Introduced in 1990, SVM works very well on high-dimensional data. It works by finding the hyperplane (decision boundary) between different class target variables (see Figure 5.1). The equation of hyperplane in SVM is:

$$y = w\,Tx + b \tag{5.1}$$

Where:

- x represents input feature
- y represents decision value
- w represents the vector weight
- B is a phrase that is biassed.

Two main types of SVM are:

1. Hard-margin SVM– If the hyperplane separates data points of different classes without any error (misclassification), it is called hard-margin SVM.
2. Soft-margin SVM – If the data points are not perfectly separable or are misclassified, then soft-margin SVM is used.

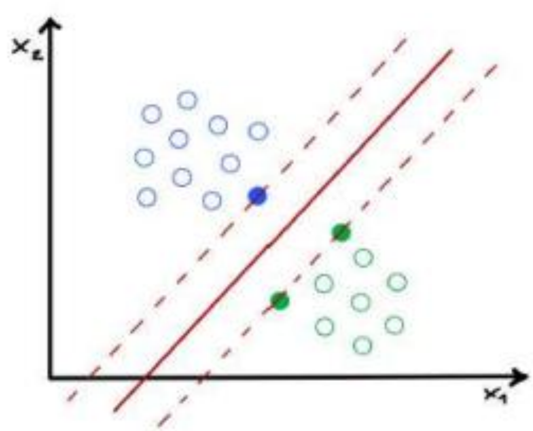


Figure 5.1 SVM for classification [64].

5.6.2 Random Forest

RF is a type of ensemble tree learning technique that has two applications: regression and classification. The model employs numerous decisions trees and averages their output to improve performance. A Key concept used in Random Forest is Bootstrap aggregation or Bagging.

Bagging- Instead of feeding the whole data into a decision tree, bootstrapped data is used. Bootstrap data refers to subsets of data selected randomly with replacement from the original dataset. This reduces the chances of overfitting as this randomness in data causes variations among individual trees. The basic steps in the random forest are:

1. *Bootstrapping*: Multiple bootstrap samples are created with replacement. Some data points may not appear at all in any of the bootstrapped data.
2. *Building decision tree*: With a random selection of features, a decision tree is built for every bootstrap sample.
3. *Aggregating Predictions*: The mean value for regression and the majority vote for categorisation is combined to create the final forecast.
4. *Evaluation*: Performance indicators including precision, accuracy, and precision are used to assess the model.
5. *Hyperparameter Tuning*: Optimise variables such as maximal depth, the number of trees, and a minimum sample for leaf nodes using grid-based cross-validation to find the optimal split.
6. *Final Training*: After optimizing the parameters, the model is trained again and evaluation metrics are noted.

5.6.3 CatBoost

"Categorical Boosting" is what the acronym CatBoost stands for, and it is designed to be effective in both classification and regression problems. The ability of CatBoost to handle categorical data without requiring manual encoding is one of its primary advantages. To address the challenges posed by categorical traits such as big cardinality, it uses a technique called Ordered Boosting. The following are CatBoost's salient features:

1. *Gradient boosting*: It is a powerful ensemble learning technique that combines weak prediction models, usually decision trees, to create a strong predictive model. It works by continuously adding novel algorithms to the ensemble, every single trained to correct the errors made by the previous models.
2. *Categorical Features*: Qualitative data is reflected in categorical parameters, such as color or type. CatBoost is a useful tool for real-world datasets because it can handle categorical characteristics efficiently without requiring one-shot encoding or extensive preparation.
3. *Learning rate*: The model's learning step size during the boosting phase is determined by the learning rate. For balancing the reliability of the model and learning speed, CatBoost automatically calculates the ideal learning rate depending on the dataset's properties.
4. *L2 regularization*: Adding an extra term called penalty to the loss function, also known as ridge regularisation, prevents overfitting and improves the model's generalisability. It is a crucial component of CatBoost that aids in managing the boosted trees' complexity.

5.6.4 Adaboost

Adaboosting is a boosting technique that assigns the same weights to every training sample at first, then iteratively adjusts these weights by highlighting data points that were incorrectly identified for the next model. Although it may be highly susceptible to noise data and outliers, it efficiently lowers bias and variance, making it valuable for classification tasks. Its simplicity of use and adaptive reweighting technique are its main advantages. It delivers robust accuracy and resists overfitting better than many simpler models with only two parameters: the number of learners (T) and the base learner type. This is especially true when using decision stumps or shallow trees.

5.6.5 Multilayer Perceptron

An ANN with several layers of the neurones connected by weighted connections is called a Multi-Layer Perceptron (MLP). Because multilayer neural networks contain one or more hidden layers, they become very powerful in representing complicated relationships within the data. In principle, MLPs have an input layer where every neuron shows a characteristic input that may be attached to the dataset in question. There are then multiple hidden layers, each carrying a vast number of neurones.

More specifically, hidden layer neurons process an input-weighted sum using an activation function—most often sigmoid, tanh, or ReLU. It is the activation functions in the hidden layer that introduce nonlinearities into the process, therefore enabling one to learn and simulate nonlinear correlations in data. In the output layer, which is the final layer of any MLP, the number of neurones is equal to the total number of regression targets or output classes. In order to help the network make predictions, this outputs layer usually provides all of the probabilities in classes for a classification job using a SoftMax activation function. Typically, a linear activation is used in the output layer design to provide continuous data for a regression issue

5.6.6 Linear Regression

A supervised ML algorithm for binary classification is called LR. LR employs a sigmoid or logistic function, which gives a probabilistic value between 0 and 1 based on predictor variables. This sigmoid function makes an S-shape curve with the threshold value. Values above threshold are noted as 1 while below the threshold are noted as 0.

5.6.7 Light Gradient Boosting Machine (LightGBM)

Microsoft created the open-source, high-performance framework known as LightGBM. The gradient boosting strategy, which is used in this ensemble learning architecture, builds a strong learner by progressively adding weak learners in a gradient descent fashion. It is made to be incredibly precise, scalable, and efficient—especially when dealing with big datasets. It makes use of decision trees that minimize memory utilization and maximize training time in order to evolve effectively.

5.6.8 XGBoost

The XGBoost algorithm, a method for ensemble learning that was first introduced in 2014, is well-liked because to its remarkable performance in tasks such as regression and classification, usually on large-scale datasets. "Extreme Gradient Boosting," or XGBoost, is renowned for its precision and scalability. It combines multiple weak learners iteratively to build a stronger learner.

XGBoost leverages the power of decision trees and enhances the outcome using a boosting approach. Boosting refers to the combination of weak learners to form a robust model. The steps involved in XGBoost are:

1. *Initialization:* The process begins with a simple prediction like the mean of the target feature.

2. *Iteration*: A fresh decision tree that forecasts the error will be added to the ensembles with each iteration.

3. *Weights*: A weight is given to each input feature according to its significance.

4. *Tree Construction*: Information gain (IG) is used for the effective splitting of the nodes. IG measures impurity in target feature.

5. *Update:* New predictions are added to previous predictions of the ensemble model, thus increasing accuracy. The process terminates after a certain number of iterations.

XGBoost uses parameters including learning rate, tree depth, and number of trees, and regularization parameters (e.g., lambda) to reduce overfitting. These hyperparameters can be optimized automatically using the Parzen estimation strategy. Therefore, XGBoost is a powerful model for high-dimensional datasets.

5.6.9 K Nearest Neighbor

For both regression and classification tasks, KNN is an instance-based learning model. KNN states that comparable data points are located near one another. The main advantage of KNN is that it is a nonparametric method, i.e., it doesn’t need any assumption about data distribution. For a given data point, KNN identifies k nearest examples closest to it and assigns majority class labels. Because it might affect the model's performance, choosing the "k" value correctly is extremely important. The steps involved in KNN are:

1. Select the appropriate value of a number of neighbors (k). The optimal value for "k" can be determined using the cross-validation approach.

2. To determine the distance between data points, choose the distance metrics. Minkowski distance is the most often used distance metric where n represents a total number of data points.

$$d\,(x, y) = \sqrt{}\sum |xi - y\ i\ |\ n\ p\ i=1\ \boldsymbol{p} \tag{5.2}$$

The aforementioned formula converts to Manhattan distance if p=1 and Euclidean distance if p=2.

3. Determine the separation between the provided data point and every other training example point.

4. Determine which "k" points are closest to the specified point.

5. Out of those ‘k’ points, find the majority class and assign it to the test data point.

6. For each test data, apply the above steps to find class labels.

5.6.10 Naïve Bayes

For classification challenges, the NB probabilistic ML algorithm works best. Under the assumption that, given a class label, features are conditionally independent, it becomes easier to compute and work with high-dimensional data. The hypothesis probability is determined using Baye's theorem in the manner described below.

$$p\ (A, B|A) = p(A) * p(B|A) \tag{5.3}$$

- Two class events are A and B, and $P(B) \neq 0$.
- The likelihood that is posterior of A given B is denoted by P(A/B).
- P(B/A) is the probability of B given A.
- The previous likelihood of class A is represented by P(A).
- The likelihood of class B is represented by P(B).

It computes prior probabilities for each unique value of target feature Y and the likelihood of each unique value for each input feature.

5.6.11 Decision Tree

DT Regression is a method for forecasting continuous values, like scores or prices, using a tree-like structure. It works by breaking up the data into smaller chunks utilising fundamental ideas that are taken from the input features. Prediction errors are reduced by these divisions. At the leaf node, or the end of each branch, the model forecasts the average value of each group.

After breaking a dataset up into progressively smaller subgroups, it progressively builds an associated decision tree. The result is a tree comprising decision and leaf nodes. Two or more branches (such as Sunny, Overcast, and Rainy) supply values for the characteristic under test on a decision node (such as Outlook). On the numerical target, a leaf node (like Hours Played) denotes a choice. The decision node that reflects the best predictor at the highest level of a tree is called the root node. Decision trees can handle both numerical and category data.

5.6.12 Gradient Boost

Gradient boosting is one ensemble learning method for tackling regression and classification issues. It is a boosting strategy that builds a strong prediction model by combining several weak learners. It attempts to fix the mistakes of the preceding model by training models one after the other.

Gradient boosting minimizes the loss function, such as the prior model's mean squared error or cross-entropy, by training each new model using gradient descent. The method trains a new weak model to minimise the upward slope of the loss function, which is determined by predictions that are made in each iteration. The process continues until a stopping requirement is met after the predictions from the new model are added to the ensemble.

5.7 Deep Learning Algorithms

In this section, deep learning models employed in this thesis for PCOS detection are described [65]. These models include ANNs, CNNs, LSTMs, GRUs and RNNs. The functions of these models are described in each of the subsections.

5.7.1 Artificial Neural Network

An ANN is a computational model that mimics the biological neural networks seen in the human brain, and is made for the primary purpose of carrying out basic computations by linked layers of nodes, or neurons. The common constituents of a network are:

Input layer: Data input is received by the input layer, which then sends it to the layers below. Each node indicates a characteristic from the input data set.

Hidden layers: Implements computations and applies a set of transformations to make queries to retrieve attributes.

Output layer: It gives the prediction or classification. Information is the content of the neurons' memories that receive input, multiplication of weights to inputs, summing, and application of activation functions to introduce non-linearities that process it through neural connections in the network. ANNs employ optimisation methods, such as gradient descent, to iteratively modify the weights during training in order to decrease the difference between the intended and projected outputs.

5.7.2 Convolutional Neural Network

CNNs are a better class of DL models that perform well in tasks requiring both temporal and spatial input, as PCOS detection and seizure prediction. After splitting that data into 80% training data and 20% testing data, the preprocess goes through several layers of CNN. Each of the several layers that make them has a separate mission or goal to achieve as described below:

Convolutional Layers: The input layer typically receives the pre-processed PCOS data. Through the filter-like, element-wise operations, convolutional layers combine these inputs by applying learnable filters in order to extract low-level to high-level features.

Activation Layers: An activation function similar to ReLU follows each convolutional layer, adding non-linearity and assisting the network in learning increasingly intricate patterns.

Pooling layers: While preserving their translation invariance and cutting down on computations, pooling layers such as the max pooling layer decrease the dimensionality of the feature maps.

Fully connected Layers: These layers receive flattened data after going through multiple convolutional and pooling layers. These layers make use of what has been retrieved from the features and make the decision making.

Dropout Layers: These layers randomly leave some of the input units off while training to prevent overfitting.

Output layer: The sigmoid layer, which produces a probability distribution across the number of classes—which may include PCOS or non-PCOS—is the last layer utilised in the majority of binary classification tasks.

This combination of layers is particularly good at knowing when PCOS is going to be diagnosed since CNNs are extremely proficient at identifying complex patterns and computations in PCOS prediction (see Figure 5.2).

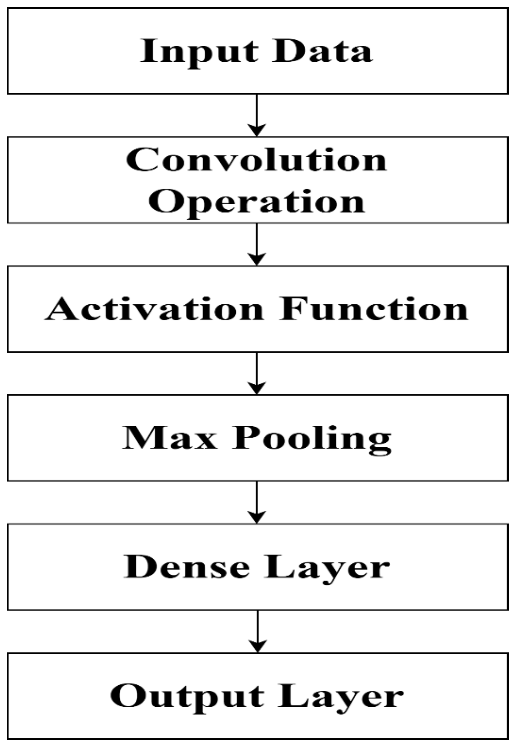


Figure 5.2 Architecture of CNN.

5.7.3 Long Short-Term Memory

The continuous acquisition of dependencies is possible with LSTM, a type of recurrent neural network, they are particularly useful for simulating epileptic convulsions from EEG data. Initial noise is removed and all datasets are normalized for comparability during the data pre-processing stage [66]. The architecture of LSTM is described below:

Layers

Input Layer: It feeds sequential input data to LSTM layers once received.

LSTM layers: Store temporal relations and maintain information for long times.

Components

Forget Gate: It decides what data from the cell state to forget.

Input Gate: It chooses what fresh information should be kept in the cell state.

Output Gate: This controls the amount of the output depending on the condition of the cell. The model includes dropout layers to lessen overfitting. After going through two LSTM layers that give the learnt features additional density, the output is sent via an output layer where a sigmoid function is used for binary classification (see Figure 5.3). The model is built using the binary cross-entropy loss function and optimised using the Adam optimiser. Early stopping in training is employed to stop learning prematurely to avoid overfitting in a model.

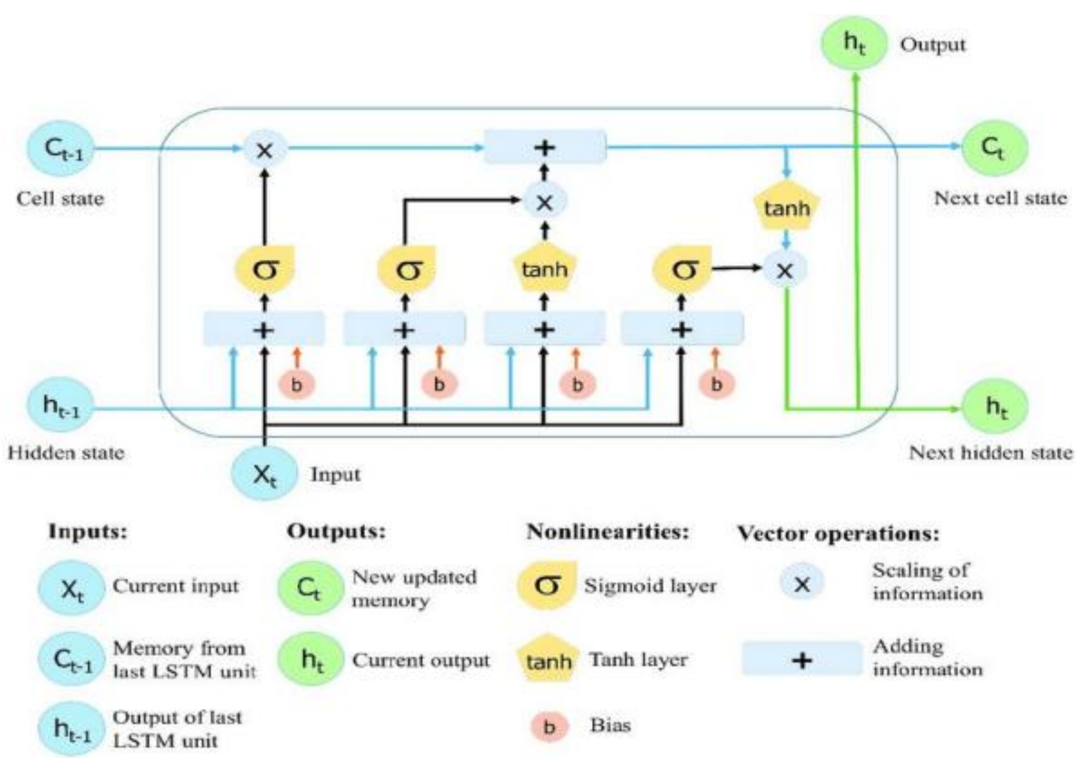


Figure 5.3 Working of LSTM [66].

5.7.4 Gated Recurrent Unit

GRUs, an alternative to LSTMs, use fewer gates as compared to LSTM. GRU does not have a distinct cell state, unlike LSTMs. Rather, it just has input and concealed state (see Figure 5.4).

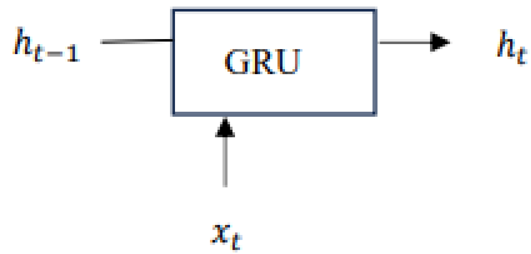


Figure 5.4 GRU Architecture [67].

GRUs are a variety of recurrent neural networks that handle some of the challenges involved in the identification of long-term dependencies in tabular database. Due to its gating mechanisms, GRUs reduce the vanishing gradient problem and let the network retain relevant information over time much better than regular RNNs.

GRU goes through a hidden state that holds the network's representation of the sequential data and gets updated selectively by reset and update gate, powered by sigmoid activation functions. Meanwhile a reset gate decides how much of the prior state to forget, an update gate is responsible for deciding on the amount of the new candidate state that should replace the old state in every GRU unit. On applications in natural language processing and time series, or any domain where modeling sequential dependencies is very important, GRUs turn out to be an integral tool by virtue of the effectiveness and simplicity of training with respect to some complex architectures like LSTM (see Figure 5.5).

GRU accepts an input and the previous concealed state and produces an entirely new hidden state. GRU's architecture has two gates: reset and update.

1. *Reset Gate* – It determines the amount of previously hidden state that must be ignored.
2. *Update Gate* - It communicates about the portion of what was previously hidden that needs to be performed forward.
3. *New Memory Content* – This step creates new memory content.
4. *Final Memory* – It gives the final memory at the current time step.

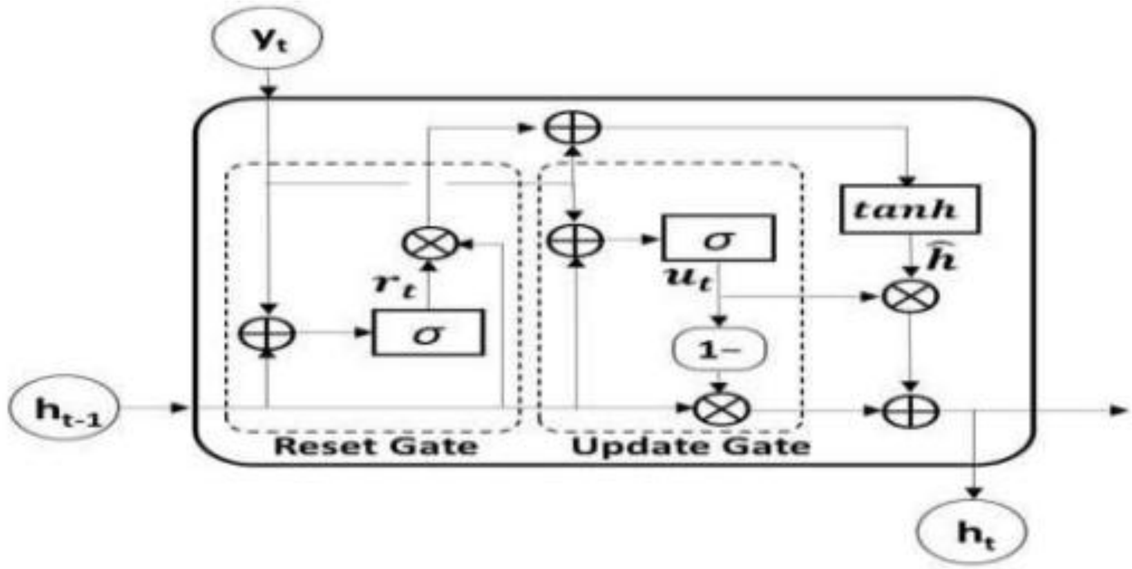


Figure 5.5 Architecture of GRU [66].

5.7.5 Recurrent Neural Network

RNNs are a form of deep learning model that was created primarily to handle sequential input; thus, they are appropriate for tasks like predicting PCOS and binary classification. RNNs are particularly effective in modeling temporal dependencies and dynamics in the data due to their recurrent 21 design where the output system at one-time step depends on the entire previous time steps. This architecture helps RNNs (see Figure 5.6) to have a hidden state which will help in passing information through sequences where the algorithm can learn from patterns in the clinical aspects of the PCOS dataset.

RNNs are a type of neural network designed for sequence modelling. They perform best for problems where sequence of input is crucial, like, speech recognition, and time series prediction. RNN's hidden state is dynamic and changes based on prior and current input [67]. Theoretically, RNNs can easily capture long-term dependencies, but in practice, they face difficulty due to expanding and vanishing gradient problems.

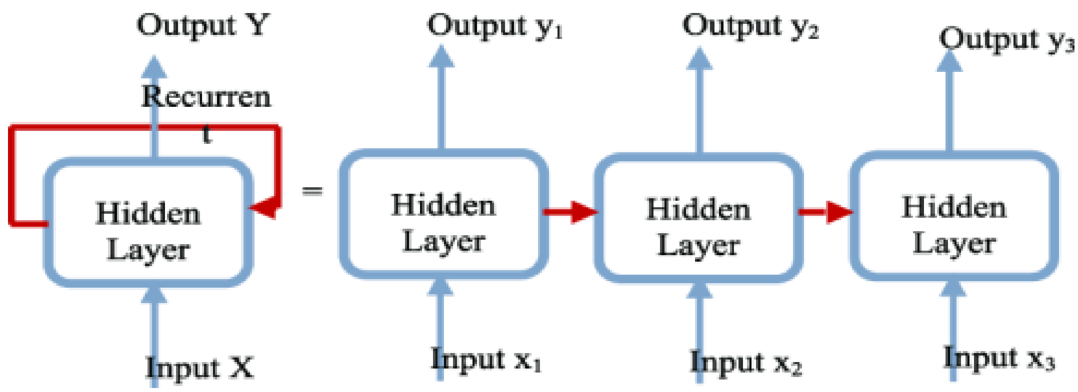


Figure 5.6 Architecture of RNN [67].

5.8 Experimental Results

This section presents the results of models created using ML and DL paired with various feature selection strategies using various data augmentation techniques. For this number of combinations were formed using the models with them respectively.

5.8.1 Machine Learning Models

For ML classification models we used various models such as NB, KNN, RF, GB, DT, LR, SVM, XGBoost, CatBoost, LightGBM, AdaBoost and MLP. But out of which only the classification results with the highest accuracy are mentioned below. We implemented different combinations of ML models along with different filtration techniques using different augmentation methods and came out with seven distinct models as follows:

M1- NB

The data first go through the preprocessing steps. For data augmentation, the SMOTE technique was used to create the artificial images. ANOVA F-statistic was used for feature selection purposes. It comes under the category of statistical approach of filter methods. The NB model classified data with 92.66% accuracy, 87.50% precision, 87.50% recall, 94.81% specificity, and 87.50% F1-score.

M2- LR

It uses random under-sampling to create synthetic samples from the minority class. For feature selection, mutual information about the features was employed. The LR model classified data with 90.83% accuracy, 84.38% precision, 84.38% recall, 93.51% specificity, and a F1 rating of 84.38%.

M3- CatBoost

Data is pre-processed and balanced using Smote-Tomek. The backward SFS technique, which is a wrapper method, is used to pick essential characteristics. Features are deleted from an existing set of attributes one by one based on their performance. The CatBoost model was used for classification, and it achieved 89.91% accuracy, 86.21% precision, 78.12% recall, 94.81% specificity, and an F1-score of 81.97%.

M4- LightGBM

After data is balanced using the Smote-ENN technique, an extra tree classifier is employed to identify the most important characteristics. The LightGBM model classified data with 90.83% accuracy, 82.35% precision, 87.50% recall, 92.21% specificity, and 84.85% F1-score.

M5- RF

The study used the SMOTE-ENN augmentation technique for creating the artificial images. For feature selection, we used a forward-based sequential feature selector. The RF model was employed for classification, and the results were 93.58% accuracy, 85.71% precision, 93.75% recall, 93.51% specificity, and an F1-score of 89.55%.

M6- XGBoost

This method employs the SMOTE-ENN augmentation technique along with mutual information extraction for feature selection. The XGBoost model achieved 91.74% accuracy, 81.08% precision, 93.75% recall, 90.91% specificity, and a F1 value of 86.96%.

M7- AdaBoost

The study used the SMOTE-ENN technique for the augmentation after the preprocessing technique. The L1, LR model technique was used for the most relevant feature extraction. The AdaBoost algorithm was used for classification, and it achieved 93.58% accuracy, 87.88% precision, and 90.62% recall, as well as 94.81% specificity and an F1-score of 89.23%.

Table 5.2 Performance Comparison of Different Machine Learning Algorithms.

Model	Accuracy	Precision	Recall	Specificity	F1 Score
M1	92.66%	87.50%	87.50%	**94.81%**	87.50%.
M2	90.83%	84.38%	84.38%	93.51%	84.38%
M3	89.91%	86.21%	78.12%	**94.81%**	81.97%
M4	90.83%	82.35%	87.50%	92.21%	84.85%
M5	**93.58%**	85.71%	**93.75%**	93.51%	**89.55%**
M6	91.74%	81.08%	**93.75%**	90.91%	86.96%.
M7	**93.58%**	**87.88%**	90.62%	**94.81%**	89.23%

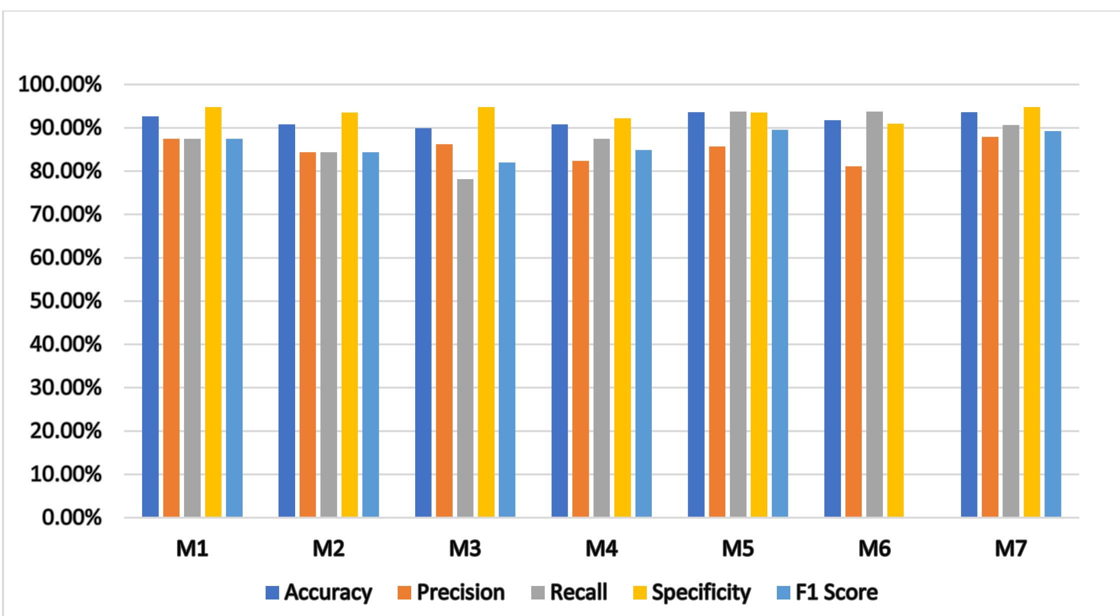


Figure 5.7 Graphical representation of the evaluation metrics for ML models.

- Table 5.7 and Figure 5.2 demonstrate that M-5 and M7 have the highest accuracy (93.58%), afterwards being M-1, M-6, M-2, M4, and M-3.
- M-7 has the greatest precision rating (87.88%), followed closely by M-1 (87.50%). M3, M-5, and M-2; M-4, and M-6.
- M-5 and M-6 had the highest recall rate of 93.75%, indicating their effectiveness in identifying positive instances. M-3 has the lowest possible recall rate, at 78.12%.
- M-1, M-3, and M-7 had the highest specificity (94.81%), followed by M-2 (93.51%), M-5 (93.51%), M-4 (92.21%), and M-6 (90.91% each).
- M-5 had the best F1 score (89.55%), followed by M-7, M-1, M-6, M-2, M-4, and M-5.
- M-7 is the best model all around.

5.8.2 Deep Learning Models

This section explains how different DL algorithms work with different sampling and filtering strategies.

M8- Feedforward

This method employs random under-sampling to balance the data and ANN for final prediction. As ANN is used for solving complex problems which were not computationally possible using traditional ML models. The feedforward or ANN algorithm achieved accuracy of 88.07%, precision of 85.18%, recall of 71.87%, and F1 score of 77.96%.

M9- CNN

CNN helps to capture local features, which has a great aspect for adapting the features of the data hierarchy. We used SMOTE for augmentation purposes. It helps in identifying the patterns in the data very efficiently. The CNN algorithm achieved accuracy of 89.90%, precision of 88.88%, recall of 75%, and F1 score of 81.35.

M10- LSTM

This method employs SMOTE to balance the data and LSTM for final prediction. LSTM stores and retrieves data via memory cells. The algorithm uses three gates: input, forget, and output. Throughout training, the model discovers how to use and modify its memory cells. The model has an accuracy of 85.32%, a precision of 90%, a recall of 56.25%, and an F1 Score of 69.23%.

M11- GRU

GRU is used to find transient patterns in clinical feature data. Similar to CRNN, a dense layer with sigmoid activation function is used for final classification. This solution utilizes the SMOTE balancing technique to address class imbalance. The model achieved accuracy of 86.23%, precision of 81.48, recall of 68.75%, and F1 score of 74.57%.

M12- CRNN

Recurrent and convolutional layers are combined to form CRNN. While recurrent layers like LSTM helps in understanding dependencies among various characteristics, convolutional layers help capture clinical elements within PCOS features. The last layer for the prediction of binary PCOS diagnosis is a dense classification layer. After employment, the initial knowledge is employed to train the model. The model obtained accuracy of 89.90%, precision of 83.87%, recall of 81.25%, and F1 score of 82.53%.

Table 5.3 Comparative analysis of various deep learning algorithms.

Model	Accuracy	Precision	Recall	F1 Score
M8	88.07%	85.18%	71.87%	77.96%
M9	**89.90%**	88.88%	75%	81.35%
M10	85.32%	**90%**	56.25%	69.23%
M11	86.23%	81.48%	68.75%	74.57%
M12	**89.90%**	83.87%	**81.25%**	**82.53%**

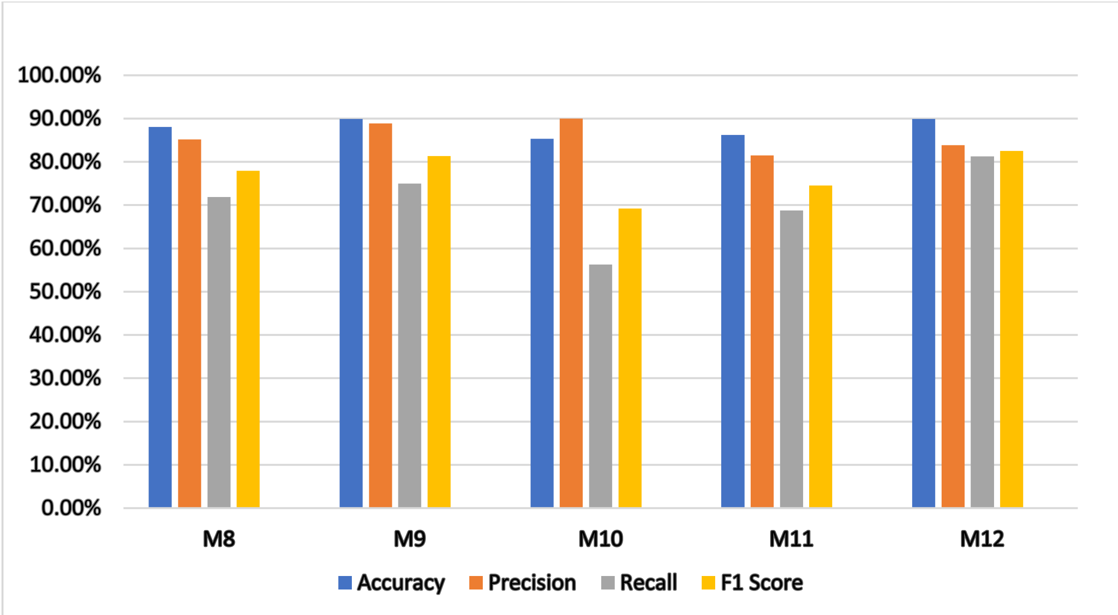


Figure 5.8 Graphical representation of the evaluation metrics for DL models.

- Table 5.3 and Figure 5.8 show that M-9 and M-12 have the best accuracy score (89.90%) among five DL algorithms (M-8 to M-12). M8 (88.07%), M11 (86.23%), and M10 (85.32%).
- M-10 has the highest precision rate (90%), followed by M-9 (88.88%), M-8 (85.18%), M-12 (83.87%), and M-11 (81.48%). The precision of M-11 is the lowest, at 81.48%.
- With a recall value of 81.25%, M-12 has the highest recall, indicating its capacity to identify positive instances. At 75%, M-9 has the lowest recall. Recall percentages for M-8, M-11 and M-10 are 71.87%, 68.75% and 56.25%, respectively.
- M12 is considered an overall good model.
- M-8 has the highest specificity score (99.29%), followed by M-10 with 99.01%. M-6 has a very low specificity (93.28%). The specificities of M-7, M9, and M-11 range from 98 to 99%.
- M-6 is the most reliable model because of its higher recall.

5.9 Discussion

Although encouraging results were obtained, there is still a great deal of room to improve classification accuracy and model robustness, according to the implementation of several DL and ML models using a combination of filter and wrapper-based feature selection methods and diverse data augmentation techniques. The following chapter examines the use of nature-inspired feature selection algorithms to overcome these obstacles and improve performance even more. This provides a more flexible and perceptive method of maximizing the model's efficacy.

Chapter

6. Nature Inspired Algorithm Based Detection of PCOS Using Clinical Data

This chapter presents an overview of the method used to develop a solution to the research problem of designing an ML/DL approach for PCOS detection using a Clinical Dataset. The section presents the conceptual basis, algorithmic structure, dataset information, and model implementation procedures using nature-inspired computational algorithms.

6.1 Overview

In order to classify PCOS from tabular clinical data, we used a multifaceted approach in this chapter that combines optimization techniques with ML models. We have used sophisticated optimization approaches to choose the most informative variables, improving model focus and lowering noise, starting with raw clinical features. A variety of ML classifiers designed for tabular data are subsequently trained using these revised feature sets. Our strategy seeks to optimize diagnostic accuracy and robustness by fusing optimization-based feature selection with advanced predictive models, proving the value of fusing data-driven selection with cutting-edge classification methods. Tukey's Honest Significant Difference (HSD) test was used after Analysis of Variance (ANOVA) to statistically validate the variations in model performance. The Lastly, we use XAI techniques in this high-performing model to guarantee transparency and interpretability, providing both local and global insights into its decision-making process.

6.2 Dataset Description and Data Preprocessing

As discussed in the previous Chapter 5, the same dataset and preprocessing steps are used in this chapter.

6.3 Feature Selection

This subsection gives an in-depth explanation of all the optimization algorithms (nature-inspired algorithms) used for feature selection [68]. By identifying the most pertinent features, feature selection greatly enhances model performance while lowering computational complexity.

The process of selecting the most crucial features from the initial set in order to create models that are more precise and effective is known as feature selection. In data preparation, feature selection is often utilized to find relevant attributes that were not previously identified and thus eliminates the redundant or irrelevant features that are not essential to the purpose of classification. Enhancing classification accuracy is the goal of feature selection.

6.3.1 Nature-Inspired Algorithms

Nature-inspired algorithms are inspired by biological processes to deal with complex problems. This section focusses on three nature-inspired algorithms: Genetic Algorithm, Flower Pollination Algorithm, and Cuckoo Search Algorithm.

Genetic Algorithm

GA offers an effective replacement for conventional heuristic techniques. It constitutes a generic adaptive optimization research methodology that is remarkably similar to Darwinian selection theory [69] and genetics in systems of biology. GA employs a population, which is a collection of possibilities. Using the Darwinian concept of "survival of the fittest," the algorithm runs several multiple computations before reaching the most efficient solution. GA generates successive sets of potential solutions—that is, alternatives to the problem—that are represented by chromosomes until desired results are achieved. The fundamental objective is to iteratively refine feature subsets in order to arrive at a high-quality solution that is approximately optimal without being aware of the search space. The core concept behind this algorithm is "survival of the fittest."

Because GA can handle immense search spaces efficiently, it has a lower probability than other algorithms to identify a local optimal result. This is related to the characteristics of exploitation and experimental search [70]. The important elements of GA consist of (see Figure 6.1):

- *Initialization:* Generates the initial population of individuals.
- Fitness Evaluation: Assess each individual's fitness using a predefined fitness function.
- *Selection:* Individuals are chosen based on their reproductive fitness.
- *Crossover:* Combines selected parents' genetic material to make offspring.
- *Mutation:* Introduces arbitrary modifications to individuals in order to preserve genetic diversity.
- *Replacement:* Integrates the newly generated offspring into the population.
- *Termination:* The algorithm ends when a stopping requirement is reached, such as maximum generations or adequate fitness.

Cuckoo Search Algorithm

CSA is a nature-inspired method widely applied in domains like speech recognition, cloud computing, and software testing [71]. The concept is based on how certain cuckoo species lay offspring in the colonies of other birds, known as hosts. Host birds might decide to destroy the egg or create a new nest. Each nest represents one possible solution to the optimisation problem. Fitness for each nest is calculated during each iteration and the quality of a solution is assessed. Combining Lévy flying randomness with cuckoo bird inspiration, it strikes a balance between exploration and exploitation, maintains few parameters, and performs well in a variety of problem scenarios (see Figure 6.2). It is still the preferred metaheuristic in both research and industry because of its proven effectiveness during engineering design, data analysis, forecasting, and other areas. It excels in global optimization over a range of fields, although careful parameter tuning is required for the best results. Because of its adaptability and successful modifications, it can be used to solve both continuous and combinatorial problems.

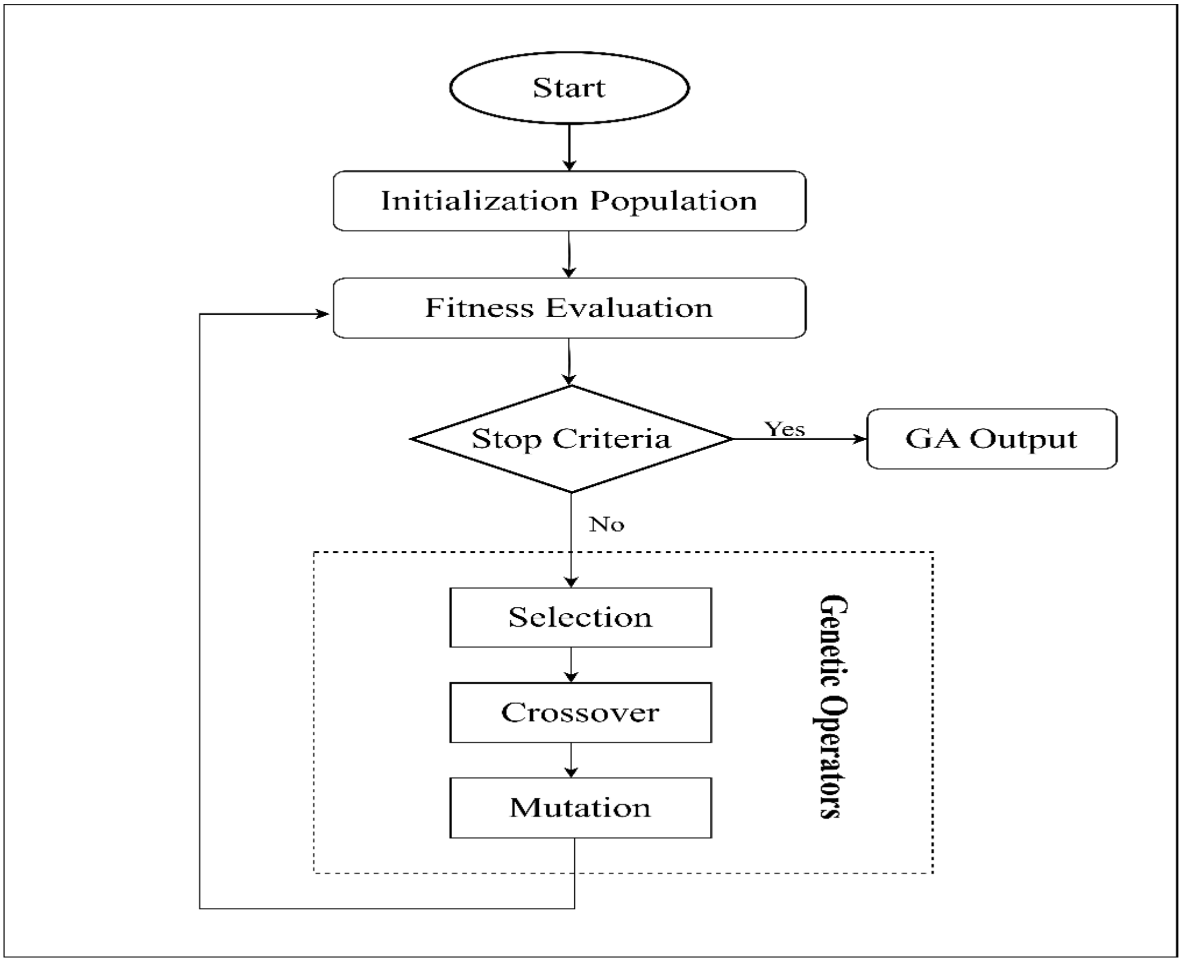


Figure 6.1 Flowchart of GA.

Levy flight, a key mechanism in CS, refers to a random walk where step lengths follow Levy distribution with a heavy tail. This enables the algorithm to make long jumps across the search space thus exploring diverse regions. The steps involved in the CSA optimization algorithm are:

- *Initialization*: Initialize a fixed number of nests (n) within the solution space [72].
- *Evaluating Fitness*: Fitness is computed for each nest to identify the best solutions. The fitness of a newly laid cuckoo egg, representing a potential solution, is compared with that of the host egg within the nest.
- *Levy flight*: Nests are replaced with new ones based on the levy flight mechanism, fostering diversification.

- *Egg laying*: If the fitness cuckoo's egg is higher than existing eggs, the nest is replaced with new solutions.
- *Termination:* The process terminates after reaching a predefined stopping criterion.

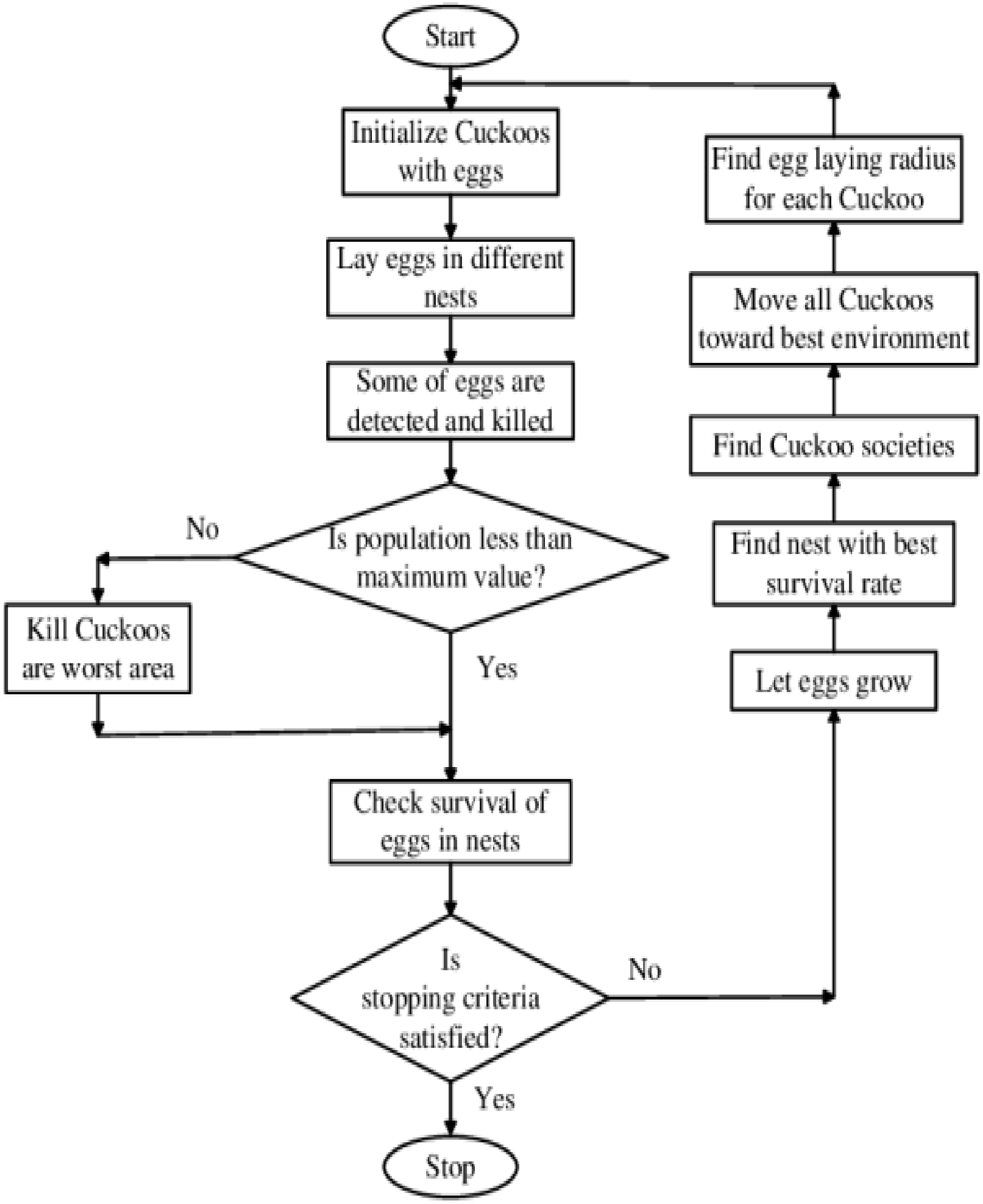


Figure 6.2 Flowchart of the Cuckoo search algorithm [73].

Flower Pollination Algorithm

FPA, a metaheuristic optimisation technique based on nature, replicates pollination in flowering plants. The method mimics two natural pollination processes: global pollination, which involves pollinators like insects or birds flying long distances (biological cross-pollination), and local pollination, which occurs through neighboring flowers via abiotic factors like wind [74]. Pollination is biotic in about 90% of cases and abiotic in 10%. With each iteration, the switching probability p balances the two modes of exploring (global) and exploiting (local) the solution space.

The FPA optimisation algorithm consists of the following steps:

1. *Initialization*: Start by randomly allocating a certain number of flowers/pollen solutions (n) around the solution space. Determine the optimal current solution, g, based on preliminary fitness assessments.
2. *Evaluating Fitness*: Determine each flower's fitness value. The selection procedure is guided by these fitness scores, which also assist in monitoring the current best option over iterations.
3. *Global Pollination*: Perform global pollination with probability p: based on long-distance pollination, create a new solution via a Lévy flight around the optimal solution $g*$.
4. *Local Pollination*: Put local pollination into action with a chance of 1−p: create a potential solution based on the two randomly selected solutions, xj and xk.
5. Best Solution Update: Update $g*$ to reflect the best solution discovered thus far after each generation (iteration).
6. *Termination*: When a stopping criterion is met, such as reaching a maximum number of future generations or achieving an acceptable degree of wellness, the algorithm terminates. The ideal solution, which is the current $g*$, is then returned (see Figure 6.3).

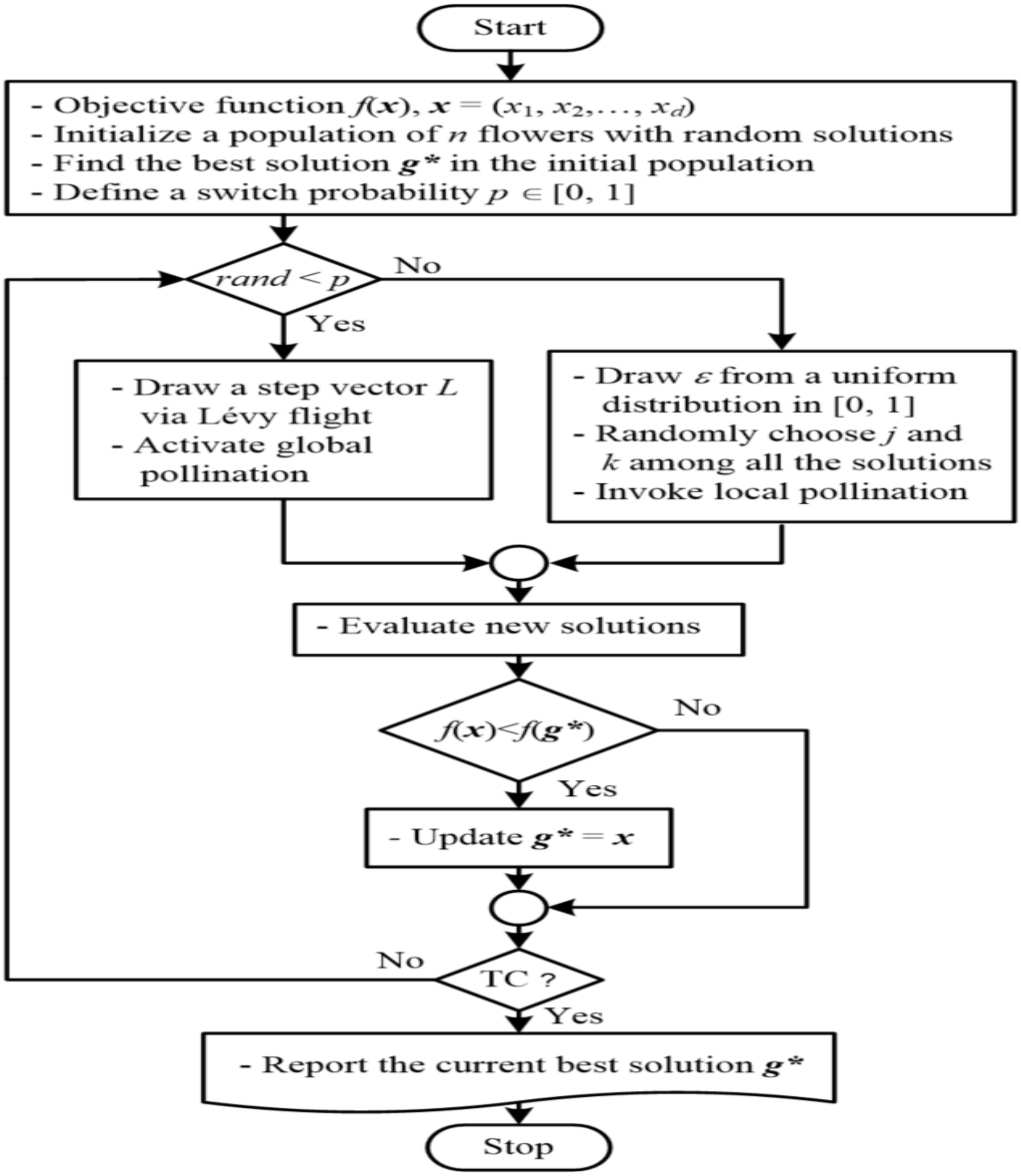


Figure 6.3 Flowchart of the Flower Pollination Algorithm [74].

6.4 Proposed Methodology

We employed various ML algorithms to predict PCOS, including NB, LR, XGBoost, KNN, RF, DT, and AdaBoost. GA was used to select features. Using soft voting, we suggested an ensemble model of DT, NB, KNN, RF, and GB model. To achieve optimal

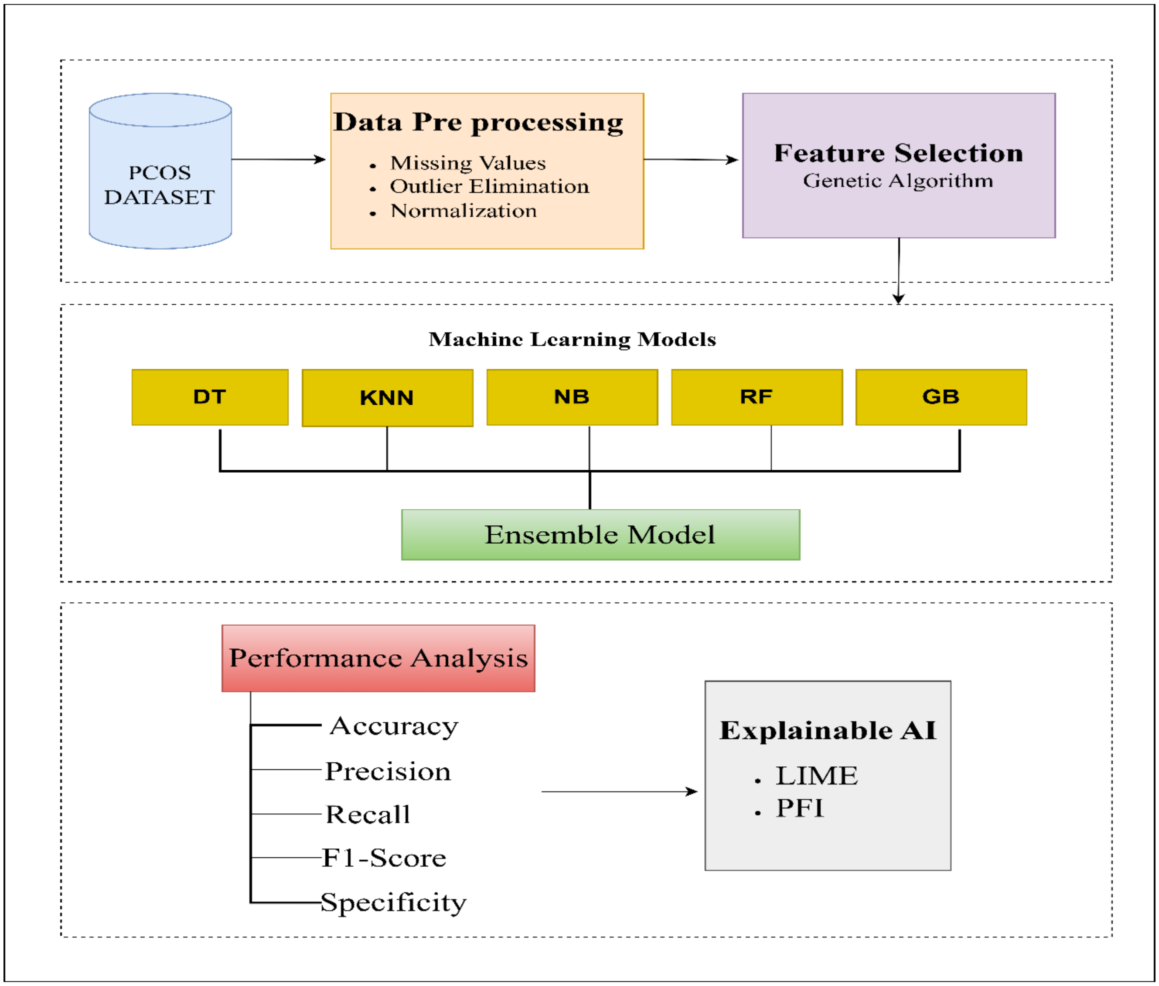


Figure 6.4 Flowchart of the Proposed Methodology.

results, grid search and GA optimisation were employed to select features. Figure 6.4 depicts a block diagram for the suggested approach.

Apart from the GA, we also tried out two other nature-inspired optimization methods for feature selection: the FPA and the CSA. The effectiveness of these techniques in enhancing classification performance was investigated through trial-and-error implementation. However following a comparative analysis, it was found that GA continuously achieved better outcomes in terms of F1 score, accuracy, and precision. Because GA performs better and is more dependable, it was chosen as the study's final feature selection technique.

The original dataset was divided into two datasets: training and testing. Training uses 80% of the dataset, while testing uses the remaining 20% to define parameters. Grid search is one of the most prominent quick methods to perform hyperparameter adjusting, and it has been implemented in the present study. To further enhance the performance of the ML models, particular emphasis was made to hyperparameter optimization. It works by specifying a grid of hyperparameter values and then executing an in-depth search through the grid to identify the optimal set of values.

Grid search [75] involves setting a grid of the hyperparameter values and doing a thorough search to determine the ideal set of values. It was also observed that many models gave better accuracy and performance results without using the hyper parameter tuning.

In order to transform weak models into more robust ones, the proposed ensemble model combines the training data to train numerous algorithms with different setting parameters.

6.5 Experimental Results

This section presents experimental data from ML models and a proposed technique utilising algorithms that are inspired by nature like GA, CSA, and FPA. The Google Compute Engine handled the entire PCOD prediction approach that was conducted using Python 3 on Google Colab along with Jupyter Notebook.

The following hyperparameter tuning parameters were observed using nature-inspired algorithms such as GA, FPA, and CSA optimization. Tables 6.2, 6.3, and Table 6.4 depict the parameter value which suits best for yielding the highest accuracy and efficacy. It was also observed that many models gave better accuracy and performance results without using the hyperparameter tuning. The following depicts the Hyperparameters for the given optimization algorithms.

Table 6.1 GA Parameters.

GA Parameters	Values
Population size	50
Number of generations	100
Crossover probability	0.7
Mutation probability	0.01

Table 6.2 FPA Parameters.

FPA Parameters	Values
Population size	100
Number of generations	50
Alpha	0.05
Beta	1.5

Table 6.3 CSA Parameters.

FPA Parameters	Values
Population size	100
Discovery probability	0.50
Alpha	2
Beta	1.5

After the feature selection using nature-inspired algorithms, the models were then further trained. ML models (RF, SVM, LR, and MLP) have been employed for classification. Also, ensemble models such as CatBoost, and AdaBoost were also employed for classification purposes (see Table 6.6 and Table 6.7).

We also developed a STACK model using multiple models including SVM, LR, RF, and GB. SVM, RF, and GB models were the base learners for the stack and LR was used as the meta learner in this case.

Two ensemble models were also developed named Ensemble-1 and Ensemble-2. The ensemble-1 model was made of the SVM, DT, NB, AdaBoost, GB, and Learned Greedy Method (LGM) model.

The ensemble-2 model which is the proposed model of the study was developed using soft voting. The Ensemble2 model demonstrated the highest accuracy (94.58%), precision (97.48%), recall (83.76%), F1-Score (90.08%), and specificity (99.09%). (See Table 6.5).

Table 6.4 Evaluation of GA-Tuned Models for PCOS Detection.

Model	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F1 Score (%)**	**Specificity (%)**
SVM	91.74	89.65	81.25	85.24	96.10
RF	91.92	94.42	77.19	84.85	98.05
CatBoost	88.99	85.71	75	80.00	94.80
AdaBoost	88.99	85.71	75	80.00	94.80
MLP	89.26	84.21	78.12	81.01	93.76
LR	91.74	92.59	78.12	84.74	97.40
STACK	93.02	96.20	79.37	86.97	98.70
Ensemble-1	90.63	90.11	76.87	82.95	96.49
Ensemble-2	**94.58**	**97.48**	**83.76**	**90.08**	**99.09**

Table 6.5 Evaluation of FPA-Tuned Models for PCOS Detection

Model	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F1 Score (%)**	**Specificity (%)**
SVM	88.99	85.71	75	80	94.80
RF	90.82	86.66	81.25	83.87	94.80
CatBoost	88.07	85.18	71.87	77.96	94.80
AdaBoost	88.07	80.64	78.12	79.36	92.20
MLP	88.99	83.33	78.12	80.64	93.50
LR	89.90	88.88	75	81.35	96.10
STACK	88.99	85.71	75	80	94.80
Ensemble-1	88.99	85.71	75	80	94.80
Ensemble-2	**92.66**	**92.85**	**81.25**	**86.66**	**97.40**

Table 6.6 Evaluation of CSA-Tuned Models for PCOS Detection.

Model	**Accuracy (%)**	**Precision (%)**	**Recall (%)**	**F1 Score (%)**	**Specificity (%)**
SVM	89.90	86.20	78.12	81.96	94.80
RF	92.66	92.85	81.25	86.66	97.40
CatBoost	89.90	88.88	75	81.35	96.10
AdaBoost	87.15	80.45	75	77.41	92.20
MLP	88.07	80.64	78.12	79.36	92.20
LR	88.99	83.33	78.12	80.64	93.50
STACK	89.90	86.20	78.12	81.96	94.80
Ensemble-1	90.82	86.66	81.25	83.87	94.80
Ensembel-2	**94.49**	**93.33**	**87.5**	**90.32**	**97.40**

6.6 Statistical Analysis

In order to support the assessment of the classification models and ensure the reliability of the findings, extensive statistical studies were carried out. Descriptive statistics, ANOVA, and Tukey's Honest Significant Difference (HSD) test are all part of the statistical study.

To assure the accuracy and consistency of the results, each model was run and assessed ten times in a row. The average performance during these runs was then taken into account using a statistical method.

ANOVA was performed to determine whether the means of the models' performance metrics differed significantly, while descriptive statistics provide a summary of the central tendency and distribution of the data. Finally, Tukey's test was employed to identify which model pairs showed statistically significant differences. The next subsections present the results and interpretations of these analyses.

6.6.1 Descriptive Analysis

As previously stated, 10 executions are performed on each model in order to get the mean and optimal fitness values for each execution. Table 6.8 displays the descriptive statistics of the results of the proposed algorithms. We have recorded the mean, standard deviation, variance, and (95%) confidence interval values of the best-obtained fitness following ten iterations of each algorithm.

Table 6.8 presents the descriptive data for each algorithm for classification across the five primary evaluation metrics: F1 score, recall, accuracy, precision, and specificity. For each model and metric, the mean, standard deviation, variance, and 95% CI are displayed. The mean values show the central tendency that characterises each model's performance, while the variance and standard deviation demonstrate the reliability and distribution of findings over numerous iterations. Models with smaller standard deviations and narrower confidence intervals show better stability and reliability.

The Ensemble-2 model outperformed individual models such as STACK, SVM, RF, and LR in terms of predictability and resilience, with higher mean scores and smaller variation across key measures. These descriptive findings provide a solid basis for further inductive analyses, such as ANOVA and Tukey's post-hoc tests, which will evaluate whether observed differences between models are statistically significant.

Table 6.7 Descriptive statistics of case studies.

Evaluation Metric	Algorithm	Mean	Std. Dev.	Variance	95% CI
Accuracy	SVM	91.74	0	0	(91.40, 92.08)
	RF	91.924	0.725706	0.526649	(91.582, 92.266)
	LR	91.74	0	0	(91.40, 92.08)
	Ensemble1	90.638	1.04095	1.08357	(90.296, 90.980)
	Ensemble2	94.582	0.522235	0.272729	(94.240, 94.924)
	CatBoost	88.99	0	0	(88.65, 89.33)
	AdaBoost	88.99	0	0	(88.65, 89.33)
	MLP	89.2627	0.749715	0.562073s	(88.920, 89.605)
	STACK	93.024	0.469922	0.220827	(92.682, 93.366)
Precision	SVM	89.65	0	0	(88.75, 90.55)
	RF	94.425	2.95872	8.75403	(93.525, 95.325)
	LR	92.59	0	0	(91.69, 93.49)
	Ensemble1	90.111	1.73061	2.99501	(89.211, 91.011)
	Ensemble2	97.481	1.73874	3.02321	(96.581, 98.381)
	CatBoost	85.71	0	0	(84.81, 86.61)
	AdaBoost	85.71	0	0	(84.81, 86.61)
	MLP	84.218	1.90618	3.63351	(83.318, 85.118)
	STACK	96.206	0.0722957	0.0052267	(95.3061, 97.1059)
Recall	SVM	81.25	0	0	(80.17, 82.33)
	RF	77.186	3.31105	10.9630	(76.10, 78.27)
	LR	78.12	0	0	(77.04, 79.20)
	Ensemble1	76.871	2.18407	4.77014	(75.790, 77.952)
	Ensemble2	83.746	1.31551	1.73056	(82.665, 84.827)
	CatBoost	75	0	0	(73.92, 76.08)
	AdaBoost	75	0	0	(73.92, 76.08)
	MLP	78.123	2.55155	6.51042	(77.042, 79.204)
	STACK	79.372	1.61633	2.61251	(78.291, 80.453)

F1 Score	SVM	85.24	0	0	(84.59, 85.89)
	RF	84.851	1.54477	2.38632	(84.202, 85.500)
	LR	84.74	0	0	(84.09, 85.39)
	Ensemble1	82.959	1.77229	3.14101	(82.310, 83.608)
	Ensemble2	90.082	0.952946	0.908107	(89.433, 90.731)
	CatBoost	80	0	0	(79.35, 80.65)
	AdaBoost	80	0	0	(79.35, 80.65)
	MLP	81.018	1.46125	2.13524	(80.369, 81.667)
	STACK	86.972	0.996648	0.993307	(86.323, 87.621)
Specificity	SVM	96.1	0	0	(95.76, 96.44)
	RF	98.05	1.10479	1.22056	(97.706, 98.394)
	LR	97.4	0	0	(97.06, 97.74)
	Ensemble1	96.49	0.627960	0.394333	(96.146, 96.834)
	Ensemble2	99.09	0.627960	0.394333	(98.746, 99.434)
	CatBoost	94.8	0	0	(94.46, 95.14)
	AdaBoost	94.8	0	0	(94.46, 95.14)
	MLP	93.76	0.822192	0.676	(93.416, 94.104)
	STACK	98.7	0	0	(98.36, 99.04)

6.6.2 One-Way ANOVA Test

Inferences from descriptive statistics are insufficient. As a result, we used the ANOVA test to conduct a comprehensive statistical study. It uses its ten trial values and the following assumptions to determine whether algorithms deliver remarkable performance.

Null hypothesis: The means are equivalent ($\mu a = \mu b$).

Alternate hypothesis: Not all means are equivalent ($\mu b \neq \mu a$).

The null hypothesis is accepted or rejected at a significance level (p-value) of 0.05. Table 6.9 presents the findings of the one-way ANOVA analysis to determine whether the mean of each of the five metrics varies statistically significantly amongst the ML models under analysis.

For every case study, it displays the ANOVA findings between groups. The Factor row shows the variation generated by model differences, whereas, within-group variation is represented by the error row. The ratio of the mean squares of "between groups" to "within groups" is used to calculate the F-value. The p-value is used to evaluate the significance of the F-value, which shows the degree of confidence in rejecting the null hypothesis. When the p-value is less than 0.05, it is evident that the null hypothesis is strongly rejected. As seen by the fact that all P-Values are 0.000, which puts them below the conventional significance level of 0.05, the means of all performance measures are significantly different among the models. ANOVA, however, doesn't identify which particular groups vary. Comparisons between the groups must be made after the fact in order to ascertain their differences.

This suggests that selecting the type of model has a significant impact on the results of every statistic, supporting the need for extensive comparative analysis that includes post-hoc testing (such Tukey HSD) to determine whether models differ from one another.

Table 6.8 ANOVA test results for evaluation metrics across ML models.

Evaluation Metric	**Source**	**Degree of freedom**	**Sum of Squares**	**Mean Squares**	**F- Value**	**P- Value**
Accuracy	**Factor**	8	297.09	37.1357	125.37	0.000
	Error	81	23.99	0.2962		
	Total	89	321.08			
Precision	**Factor**	8	1870.1	233.759	114.27	0.000
	Error	81	165.7	2.046		
	Total	89	2035.8			
Recall	**Factor**	8	646.4	80.797	27.35	0.000
	Error	81	239.3	2.954		
	Total	89	885.7			
F1-Score	**Factor**	8	906.07	113.258	106.58	0.000
	Error	81	86.08	1.063		
	Total	89	992.14			

Specificity	**Factor**	8	281.55	35.1943	117.96	0.000
	Error	81	24.17	0.2984		
	Total	89	305.72			

Table 6.10 displays the grouping information derived from the Tukey HSD method with a 95% confidence level along with its Tukey ranking. The Tukey HSD uses mean fitness to rank the algorithms as A, B, C, D, E, and F. Grades of A and F are given to the algorithms with the highest and lowest fitness values, respectively. Stated differently, it reflects their preferred order or suggests that the algorithm with the highest rating be chosen first.

If the rankings of the two algorithms differ, the null hypothesis is rejected. Algorithms are considered statistically different if their mean fitness scores differ by <0.05 ($p < 0.05$). The homogeneous grouping of models with the same TR indicates that, at the 0.05 level, there is no observable variation in their mean results. Ensemble-2 consistently generates a unique group "A," proving its superiority over comparable models for all metrics. Following the proposed Ensemble-2 approach, STACK provides the second-best outcomes in terms of most evaluation metrics, as demonstrated by their grouping. Other models, such as MLP, CatBoost, and AdaBoost, have lower TR values, indicating significantly lower mean efficiency.

The Tukey rankings (TR) on accuracy (Acc), precision (Prec), recall (Rec), F1-Score (F1), and specificity (Spec) are shown in Table 6.10.

Table 6.9 Grouping Information Using the Tukey method and 95% confidence.

Models	**Acc**	**TR**	**Pre**	**TR**	**Rec**	**TR**	**F**	**TR**	**Spec**	**TR**
Ensemble2	94.58	A	97.48	A	83.74	A	90.08	A	99.09	A
STACK	93.02	B	96.20	AB	81.25	B	86.97	B	98.70	AB
RF	91.92	C	94.42	BC	79.37	BC	84.85	C	98.05	BC
LR	91.74	C	92.59	C	78.12	CD	84.74	C	97.40	C
SVM	91.74	C	89.65	D	78.12	CD	85.24	C	96.10	D
Ensemble1	90.63	D	90.11	D	77.19	CDE	82.95	D	96.49	D

AdaBoost	88.99	E	85.71	E	76.87	DE	80.00	E	94.80	E
CatBoost	88.99	E	85.71	E	75.00	E	80.00	E	94.80	E
MLP	89.26	E	84.21	E	75.00	E	81.01	E	93.76	F

6.7 XAI

XAI [76] enhances PCOS classification effectiveness, flexibility, and trustworthiness. The use of AI models to detect PCOS is an emerging technological development. The study used methods like LIME [77] and PFI [78] to visualize particular events that helped with model prediction. We employ the XAI techniques LIME and PFI to investigate the ensemble classifier's findings.

As the use of XAI techniques is frequently overlooked in previous studies, this makes this component of our proposed methodology particularly important [79]. By providing clear, human-understandable explanations for predictions, XAI helps clinicians and patients trust and adopt AI-based tools. Incorporating explainability ensures that the models are not just accurate but also transparent and accountable, which is vital for real-world clinical use.

6.7.1 LIME

Using the LIME model, we extended our suggested ensemble model for this study. By using it, we provide lucid insights into our model's decision-making process, improving the interpretability and reliability of our classification results. The LIME approach is used to build an LR around a likely inference point. In the approximations, variables with a high affirmative weight work in favor of the prediction decision, whereas variables with negative weights are opposed to it. The original LIME model has a consistency challenge, indicating that it can result in various interpretations when used continuously under the same circumstance [80].

Figure 6.5 illustrates the interpretability of the proposed model using the LIME. Interpretability refers to an individual's ability to comprehend the logic behind a "black box" model's output. This interpretable model provides both text-based and visual interpretations and shows how the parameters affect the outcome both positively and negatively. Figure's 6.5 left side displays the predicted probability of the suggested model for the "Not Infected," which displays 100%, and "Infected," which displays 0%, for the dataset's fifth observable. This suggests that there is considerable trust that the patient is not infected. The model's prediction certainty is highlighted in this bar chart.

The central part displays the features and their corresponding weights that affect the likelihood of the prediction. This waterfall-style illustration shows how several features contribute to the classification choice. Moving towards "Not Infected" is represented by the left (blue), while pushing towards "Infected" is represented by the right (orange). Features such as weight increase, acne, hair growth, follicle number, skin darkening, and fast-food consumption are used to define the "Not Infected" outlook. While characteristics like hip size (>0.62 inches) and \(\beta\)-hCG (beta-Human Chorionic Gonadotropin) lean somewhat in the direction of "Infected," their impact is minimal. The blue bars' prominent prevalence indicates that their final verdict is "Not Infected”.

LIME's outputs are not quantitatively evaluated for integrity, which could undermine their trustworthiness even though they provide logical local explanations. Another important stability problem is that LIME uses random sampling to produce perturbations, which results in variations in explanations over multiple runs.

To assess the robustness of LIME explanations, stability test was conducted. We determined the Coefficients Stability Index (CSI = 0.82) and Variables Stability Index (VSI = 0.85) by running LIME explanations ten times on the same instance. A VSI of 0.85 suggests strong feature selection a stability, meaning that the most valuable features identified by LIME are consistently replicated about 85% of the time across 10 iterations. A coefficient estimation consistency between runs and minimal feature weight fluctuation is indicated by a CSI of 0.82. These positive outcomes demonstrate that LIME provides dependable and consistent feature selection and weights, so addressing significant concerns surrounding explanation variability.

The true values of the features for the current situation are accordingly represented in Figure 6.5 on the far right. Characteristics like pimples, skin darkening, lack of hair growth, and poor follicle count significantly support this conclusion. Hip size is given as 0.69, PRL (Prolactin) of 0.30, and Vit D3 as 0.61, which is highlighted in orange, tend to show key factors towards "Infected" but their contribution is not enough to change the model’s final prediction.

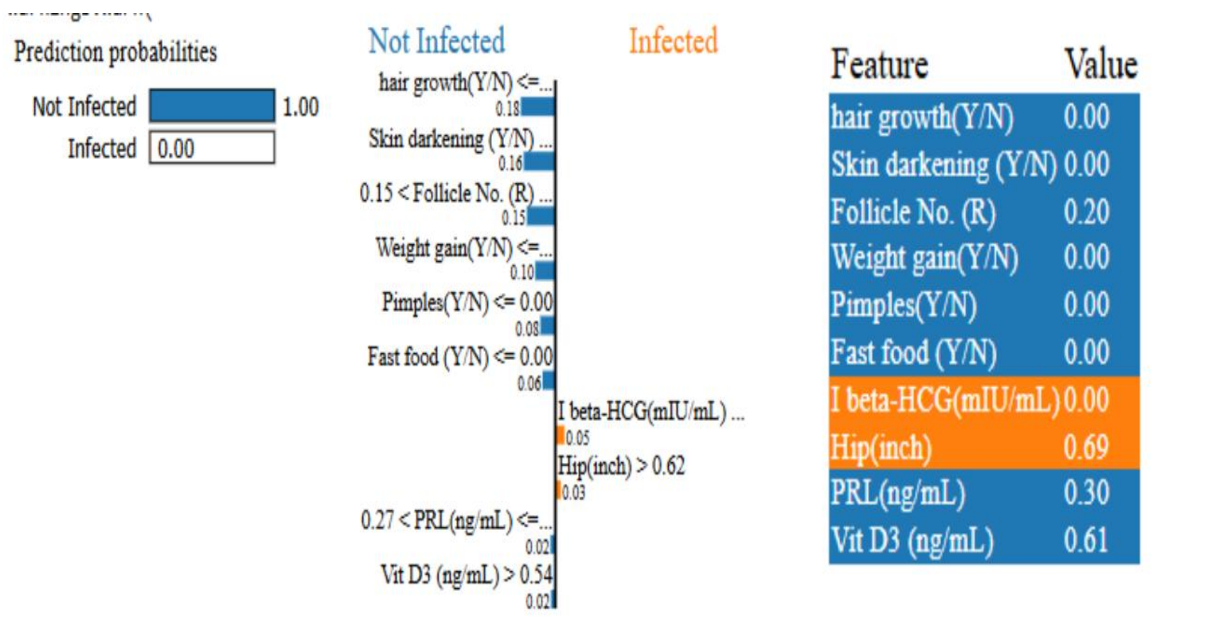


Figure 6.5 A description based on the suggested classifier’s estimation of a person’s likelihood of having PCOS (Yes) or PCOS (No).

6.7.2 PFI

Permutation feature importance is a model inspection technique that measures the relative contribution of each feature to the statistical performance of a fitted model on a given tabular dataset. This method is especially helpful for non-linear or ambiguous estimating methods, like randomly changing the values of a single feature and tracking the ensuing decline in the model's score. By examining the relationship between the feature and the item under study, we determine the rate at which the model depends on a certain property [81].

One of the main advantages of the permutation feature is that its significance is model-agnostic, meaning it applies to any fitted estimator. It can be tested more than once using a substitute feature combination and also gives an estimate of the variance in the predicted feature value for each particular trained model. PFI is comprehensible since it shows the contributions of features in a simple manner. It also helps with feature selection by helping to remove features that aren't necessary [82].

Figure 6.6 shows the permutation feature significance of the proposed ensemble model. Positive and Negative Importance Scores are the foundation upon which the PFI was built. The characteristics placed on the Y-axis are displayed in accordance with their significance in the model, while the feature importance score on the X-axis shows that higher scores correlate to more relevant features. PFI assesses how much a feature affects model performance by randomly rearranging its values and tracking the change in the model's reliability [83].

The Follicle Number (R), which has the greatest value at roughly 0.175, has a considerable impact on the classification's result. Other important factors include weight gain (Y/N), hair growth (Y/N), and skin darkening (Y/N). On the other hand, characteristics like BP Systolic/Diastolic, Waist (inch), Hip (inch), and (beta)-hCG (II) are less significant and contribute less to the model. The model prediction is hardly affected by several traits, such exercise (Y/N) and pimples (Y/N).

Additionally, Figure 6.6 has been found to exhibit the negative significance scores as well. When combined, several features reduce model accuracy somewhat because of noise or redundant functions. The following features had negative significance scores: hair loss (Y / N), waist (inch), hip (inch), PRL (ng / mL), Vit D3 (ng / mL), and average F size (R) (mm).

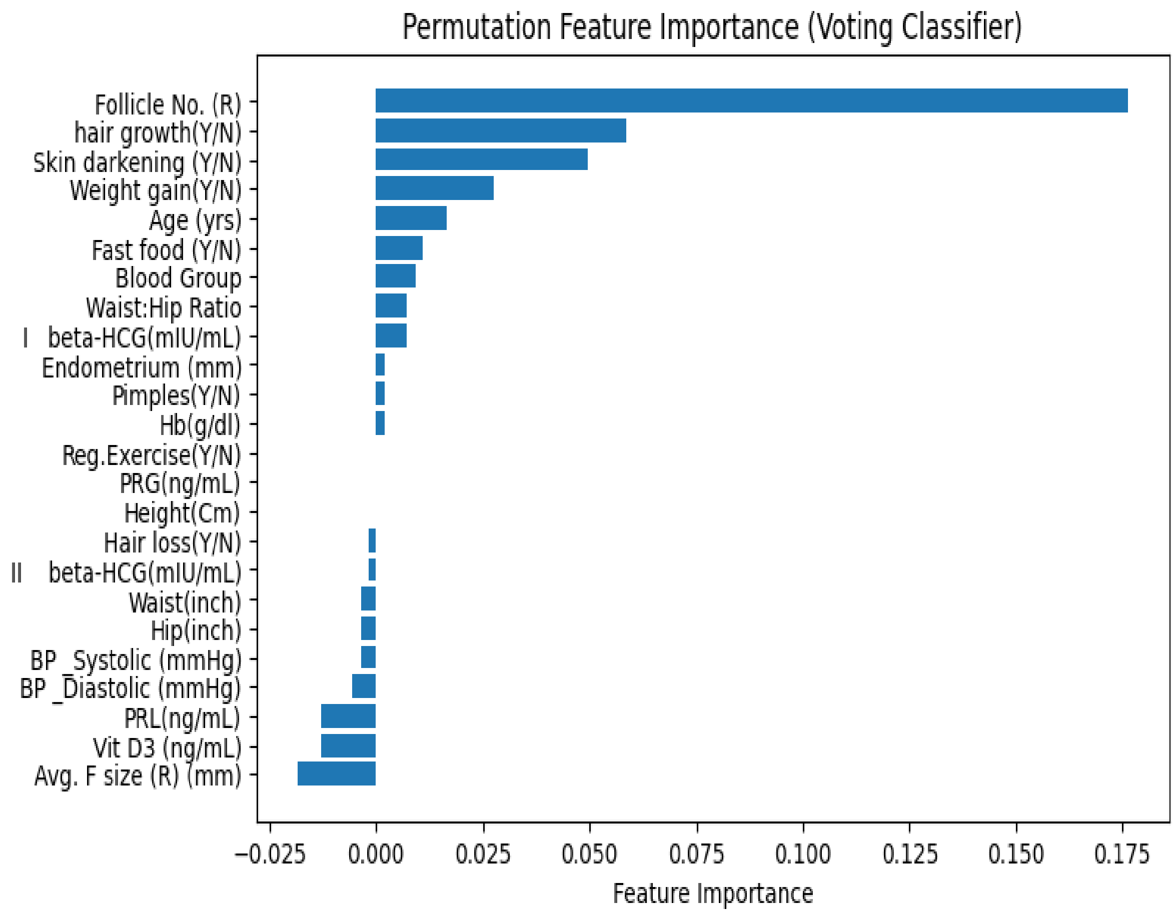


Figure 6.6 Permutation Feature Importance Analysis for PCOS Prediction.

6.8 Discussion

We proposed an ensemble model consisting of DT, NB, KNN, RF, and GB models using soft voting. To obtain the best results, a grid search was carried out along with GA optimization for feature selection. The performance of the algorithm is measured using the metrics recall, accuracy, precision, ROC-AUC score, accuracy, and F1-score. This study also offers model interpretations utilizing XAI, to guarantee model efficacy, performance, and trust. The model interpretations and explanations were examined using the LIME and PFI models. The experiment was also statistically analyzed using the ANOVA and the Tukey test.

Chapter

7. Conclusion and Future Scope

This study analyzes the identification of PCOS using advanced machine learning and deep learning approaches, employing both imaging and clinical feature datasets. The study shows how using clinical criteria and US images, automated systems can increase the accuracy and efficiency of PCOS diagnosis. InceptionV3 and ResNet50, deep learning models based on transfer learning, achieved 99.68% and 97.5% accuracy in classification of US images, respectively. To improve model performance and narrow the feature space for the clinical dataset, a variety of feature selection techniques were used, including nature-inspired algorithms like the FPA, CSA, and GA. The best results were obtained with 94.58% accuracy, 97.48% precision, 83.76% recall, 90.08% F1-Score, and 99.09% specificity when the proposed ensemble model was used in conjunction with GA-based feature selection. In addition, XAI methods were used to enhance the diagnostic system's interpretability and reliability along with statistical and descriptive analysis. Overall, this study shows how advanced computational methods can be used to produce a prompt and precise PCOS diagnosis, reinforcing healthcare professionals' judgment and establishing the way for further developments in women's healthcare.

In future, the PCOS detection can be improved in following ways:

- *Multimodality Datasets:* Most studies concentrate on just one type of dataset. To enhance diagnostic techniques, more multi-modality datasets should be used, providing more insights and variability. A variety of datasets could improve the algorithm's ability to predict PCOS.
- *Integration of Quantum Computing*: The speed and effectiveness of analysis in PCOS prediction can be significantly increased by using quantum computing. By utilizing quantum parallelism, it makes it possible to handle high-dimensional data, like clinical features, high-resolution US images more quickly. The precision and scalability of future models that use quantum-enhanced learning may surpass those of traditional

techniques. In PCOS detection and classification, this field is still mostly unexplored despite its potential.

- *Annotated Datasets*: It has been seen that there doesn't exist any publicly accessible databases with ground truth images present by which the segmentation can be performed. Almost all the datasets that are used in the research paper are kept private and not available publicly.
- *Enhanced Large Datasets:* Previous research has primarily focused on smaller datasets. To ensure the applicability of models in real-world clinical settings, larger datasets covering diverse populations are needed for both training and validation.
- *Software Systems*: The development of practical software systems has been underexplored in current research. Investigating mobile apps and real-time analysis for PCOS detection could enhance accessibility and early diagnosis, ultimately improving patient outcomes.
- *Wearable and IoT-Based Monitoring*: Monitoring physical and hormonal health signs with wearable technologies and artificial intelligence to spot problems early.

List of Publications

Journal Articles (SCIE-indexed)

1. Polycystic Ovary Syndrome (PCOS) Image Detection Using Segmentation and Classification: A Systematic Review (**Communicated)**
2. Polycystic Ovary Syndrome Classification Using Genetic Algorithm-Based Feature Selection and Ensemble Learning (**Communicated)**

Conference Publications (Scopus Indexed)

3. PCOS Detection in Ultrasound Images Using Transfer Learning with InceptionV3 and ResNet50. In 10th International Conference on ICT for Sustainable Development (ICT4SD Goa, India). (**Accepted)**

References

[1] S. Singh, N. Pal, S. Shubham, D. K. Sarma, V. Verma, F. Marotta, M. Kumar, Polycystic ovary syndrome: Etiology, current management, and future therapeutics, Journal of Clinical Medicine 12 (2023).

[2] N. Ajmal, S. Z. Khan, R. Shaikh, Polycystic ovary syndrome (pcos) and genetic predisposition: A review article, European Journal of Obstetrics Gynecology and Reproductive Biology: X 3 (2019) 100060.

[3] R. Azziz, E. Carmina, Z. Chen, A. Dunaif, J. S. Laven, R. S. Legro, D. Lizneva, B. Natterson-Horowtiz, H. J. Teede, B. O. Yildiz, Polycystic ovary syndrome, Nature reviews Disease primers 2 (2016) 1–18.

[4] J. Bjeki´c-Macut, T. Vukaˇsin, Z. Velija-Aˇsimi, A. Burekovi´c, M. Zdravkovi´c, Z. Andri´c, M. Brankovi´c, S. Crevar-Marinovi´c, T. Madi´c, O. Stanojlovi´c, et al., Polycystic ovary syndrome: a contemporary clinical approach, Current Pharmaceutical Design 27 (2021) 3812–3820.

[5] R. J. Rodgers, J. C. Avery, V. M. Moore, M. J. Davies, R. Azziz, E. Stener-Victorin, L. J. Moran, S. A. Robertson, N. K. Stepto, R. J. Norman, et al., Complex diseases and co-morbidities: polycystic ovary syndrome and type 2 diabetes mellitus, Endocrine Connections 8 (3) (2019) R71–R75.

[6] M. Z. Md Muslim, A. Mohammed Jelani, N. Shafii, N. M. Yaacob, N. A. A. Che Soh, H. A. Ibrahim, Correlation between anti-mullerian hormone with insulin resistance in polycystic ovarian syndrome: a systematic review and meta-analysis, Journal of Ovarian Research 17 (1) (2024) 106.

[7] A. Das, D. Choudhury, A. Sen, A collaborative empirical analysis on machine learning based disease prediction in health care system, International Journal of Information Technology 16 (1) (2024) 261– 270.

[8] N. S. Narayan, S. Sivanandan, S. Kudavelly, K. A. Patwardhan, G. A. Ramaraju, Automated detection and segmentation of follicles in 3D ultrasound for assisted reproduction, in: N. Petrick, K. Mori (Eds.), Medical Imaging 2018: ComputerAided Diagnosis, volume 10575, International Society for Optics and Photonics, SPIE, 2018, p. 105751W.

[9] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, nature 521 (2015) 436–444.

[10] J. M. Shivaram, Segmentation of ovarian cyst using improved u-net and hybrid deep learning model, Multimedia Tools and Applications 83 (2024) 42645–42679.

[11] S. V. Palvi Soni, Image segmentation for detecting polycystic ovarian disease using deep neural networks, International Journal of Computer Sciences and Engineering 7 (2019) 534–537.

[12] K. Sheikdavood, M. P. Bala, Polycystic ovary cyst segmentation using adaptive k-means with reptile search algorithm, Information Technology and Control 52 (2023) 85–99.

[13] V. Gotte, J. Singaraju, Speckle reduction and segmentation for the detection of follicles in the ultrasound images of ovaries (2020).

[14] C. Narmatha, P. Manimegalai, J. Krishnadass, P. Valsalan, S. Manimurugan, M. Mustafa, Ovarian cysts classification using novel deep reinforcement learning with harris hawks optimization method, The Journal of Supercomputing 79 (2023) 1374–1397.

[15] T. N. Ravishankar, H. Makarand Jadhav, N. Satheesh Kumar, S. Ambala, M. Pillai N, A deep learning approach for ovarian cysts detection and classification (ocd-fcnn) using fuzzy convolutional neural network, Measurement: Sensors 27 (2023) 100797.

[16] P. Raja, P. Suresh, Variety of ovarian cysts detection and classification using 2d convolutional neural network, Multimedia Tools and Applications 83 (2024) 49473–49491.

[17] D. Shao, C. Xu, Y. Xiang, P. Gui, X. Zhu, C. Zhang, Z. Yu, Ultrasound image segmentation with multilevel threshold based on differential search algorithm, IET Image Processing 13 (2019) 998–1005.

[18] S. Sheela, M. Sumathi, Enhancer for ovarian cyst segmentation using adaptive thresholding technique, Indian Journal of Science and Technology 13 (2020) 4142–4150.

[19] B. Rachana, T. Priyanka, K. N. Sahana, T. R. Supritha, B. D. Parameshachari, R. Sunitha, Detection of polycystic ovarian syndrome using follicle recognition technique, Global Transitions Proceedings 2 (2021) 304–308. International Conference on Computing System and its Applications (ICCSA- 2021).

[20] N. Zulkarnain, A. A. Nazarudin, S. S. Mokri, A. Hussain, I. N. A. M. Nordin, et al., Ultrasound image segmentation for detecting follicle in ovaries using morphological operation and extraction methods, Journal of Pharmaceutical Negative Results 13 (2022) 659–665.

[21] A. A. Nazarudin, N. Zulkarnain, S. S. Mokri, W. M. D. W. Zaki, A. Hussain, M. F. Ahmad, I. N. A. M. Nordin, Performance analysis of a novel hybrid segmentation method for polycystic ovarian syndrome monitoring, Diagnostics 13 (2023).

[22] P. G. Yılmaz, G. Ozmen, Follicle detection for polycystic ovary syndrome by using image processing methods, International Journal of Applied Mathematics Electronics and Computers 8 (2020) 203–208.

[23] K. Maheswari, T. Baranidharan, S. Karthik, T. Sumathi, Modelling of f3i based feature selection approach for pcos classification and prediction, Journal of Ambient Intelligence and Humanized Computing 12 (2021) 1349–1362.

[24] S. A. Suha, M. N. Islam, An extended machine learning technique for polycystic ovary syndrome detection using ovary ultrasound image, Scientific Reports 12(2022) 17123.

[25] V. Kiruthika, S. Sathiya, M. Ramya, K. S. Sankaran, An intelligent machine learning approach for ovarian detection and classification system using ultrasonogram images, Engineered Science 23 (2023) 879.

[26] N. N. (2021). Follicles Classification To Detect Polycystic Ovary Syndrome Using Glcm And Novel Hybrid Machine Learning. *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, *12*(7), 1062–1073.

[27] N.S., Nilofer & R., Dr. (2021). A Novel Hybrid IWD Algorithm with Extreme Learning Adaptive Neuro-fuzzy Inference System based Follicle Identification for Polycystic Ovary Syndrome in Ultrasound Images. Webology. 18. 594-616. 10.14704/WEB/V18SI05/WEB18249.

[28] Li, H., Fang, J., Liu, S., Liang, X., Yang, X., Mai, Z., Van, M. T., Wang, T., Chen, Z., & Ni, D. (2020). CR-Unet: A Composite Network for Ovary and Follicle Segmentation in Ultrasound Images. *IEEE journal of biomedical and health informatics*, *24*(4), 974–983.

[29] Chen, Z., Zhang, C., Li, Z., Yang, J. and Deng, H., 2022. Automatic segmentation of ovarian follicles using deep neural network combined with edge information. *Frontiers*

in Reproductive Health, *4*, p.877216.

[30] Kodipalli, A., Devi, S., & Dasar, S. (2024). Semantic segmentation and classification of polycystic ovarian disease using attention UNet, Pyspark, and ensemble learning model. *Expert Systems*, 41(3), e13498.

[31] Sarkar, Manas, Ardhendu Mandal, and Anil Tudu. "DC-UNet: Looking for follicles in the ovarian ultrasound images." *Franklin Open* 8 (2024): 100149.

[32] Singh, V.K.; Yousef Kalafi, E.; Cheah, E.; Wang, S.; Wang, J.; Ozturk, A.; Li, Q.; Eldar, Y.C.; Samir, A.E.; Kumar, V. HaTU-Net: Harmonic Attention Network for Automated Ovarian Ultrasound Quantification in Assisted Pregnancy. *Diagnostics* **2022**, *12*, 3213.

[33] Choubey, S.B., Choubey, A., Nandan, D., Mahajan, A. (2021). Polycystic ovarian syndrome detection by using two-stage image denoising. Traitement du Signal, Vol. 38, No. 4, pp. 1217-1227.

[34] Vikas, B., Y. Radhika, and K. Vineesha. "Detection of polycystic ovarian syndrome using convolutional neural networks." *Int J Curr Res Rev* 13, no. 6 (2021): 156-60.

[35] M. Sumathi, P. Chitra, S. Sheela, C. Ishwarya, Study and implementation of automated system for detection of pcos from ultrasound scan images using artificial intelligence, The Imaging Science Journal 72 (2024) 828–839

[36] P. Bedi, S. Goyal, A. S. Rajawat, M. Kumar, An integrated adaptive bilateral filter-based framework and attention residual u-net for detecting polycystic ovary syndrome, Decision Analytics Journal 10 (2024) 100366.

[37] M. Sha, Segmentation of ovarian cyst in ultrasound images using adaresu-net with optimization algorithm and deep learning model, Scientific Reports 14 (2024) 18868.

[38] N. Priya, et al., Identifying and classifying an ovarian cyst using scbod (size and count-based ovarian detection) algorithm in ultrasound image, international journal of electrical and computer engineering systems 13 (2022) 799–806.

[39] Alamoudi, Abrar, Khan, Irfan Ullah, Aslam, Nida, Alqahtani, Nourah, Alsaif, Hind S., Al Dandan, Omran, Al Gadeeb, Mohammed, Al Bahrani, Ridha, A Deep Learning Fusion Approach to Diagnosis the Polycystic Ovary Syndrome (PCOS), *Applied Computational Intelligence and Soft Computing*, 2023, 9686697, 15 pages, 2023.

[40] D. Patil, Smital, Pramod J. Deore, and Vaishali Bhagwat Patil. 2024. “An Intelligent Computer Aided Diagnosis System for Classification of Ovarian Masses Using

Machine Learning Approach”. *International Research Journal of Multidisciplinary Technovation* 6 (3):45-57.

[41] GALAGAN, Roman, et al. "Automation of polycystic ovary syndrome diagnostics through machine learning algorithms in ultrasound imaging." *Applied Computer Science* 20.2 (2024): 194-204.

[42] M. M. Rahman, A. Islam, F. Islam, M. Zaman, M. R. Islam, M. S. A. Sakib, H. M. H. Babu, Empowering early detection: a web-based machine learning approach for pcos prediction, Informatics in Medicine Unlocked 47 (2024) 101500.

[43] İ. B. Çiçek, Z. Küçükakçalı, F. H. Yağın, Detection of risk factors of pcos patients with local interpretable model-agnostic explanations (lime) method that an explainable artificial intelligence model, The Journal of Cognitive Systems 6 (2) (2021) 59–63.

[44] N. N. Faris, F. S. Miften, Proposed model for detection of pcos using machine learning methods and feature selection, Journal of Education for Pure Science 13 (1) (2023) 85–93.

[45] V. V. Khanna, K. Chadaga, N. Sampathila, S. Prabhu, V. Bhandage, G. K. Hegde, A distinctive explainable machine learning framework for detection of polycystic ovary syndrome, Applied System Innovation 6 (2) (2023) 32.

[46] S. Sreejith, H. K. Nehemiah, A. Kannan, A clinical decision support system for polycystic ovarian syndrome using red deer algorithm and random forest classifier, Healthcare Analytics 2 (2022) 100102.

[47] R. Ahmad, L. A. Maghrabi, I. A. Khaja, L. A. Maghrabi, M. Ahmad, Smote-based automated pcos prediction using lightweight deep learning models, Diagnostics 14 (19) (2024) 2225.

[48] S. Vairachilai, D. Anuhya, A. Tirkey, S. Raja, Slb-smote logistic blending hybrid machine learning model for chronic polycystic ovary syndrome prediction with correlated feature selection, Informatics for Health and Social Care 49 (3-4) (2024) 190–211.

[49] A. Kumar, J. Singh, A. A. Khan, A comprehensive machine learning framework with particle swarm optimization for improved polycystic ovary syndrome (pcos) diagnosis, Engineering Research Express 6 (3) (2024) 035233.

[50] H. Elmannai, N. El-Rashidy, I. Mashal, M. A. Alohali, S. Farag, S. ElSappagh, H. Saleh, Polycystic ovary syndrome detection machine learning model based on optimized feature selection and explainable artificial intelligence, Diagnostics 13 (8) (2023) 1506.

[51] P. Moral, D. Mustafi, S. K. Sahana, Podboost: an explainable ai model for polycystic ovarian syndrome detection using grey wolfbased feature selection approach, Neural Computing and Applications 36 (30) (2024) 18627–18644.

[52] G. U. Mahesswari, P. U. Maheswari, Smartscanpcos: A feature-driven approach to cutting-edge prediction of polycystic ovary syndrome using machine learning and explainable artificial intelligence (2024).

[53] H. Danaei Mehr, H. Polat, Diagnosis of polycystic ovary syndrome through different machine learning and feature selection techniques, Health and Technology 12 (1) (2022) 137–150.

[54] Kaggle Dataset- https://www.kaggle.com/datasets/anaghachoudhari/pcos-detection-using-ultrasound-images

[55] Handa, P., Saini, A., Dutta, S., Pathak, H., Choudhary, N., Goel, N., Dhanao, J.K.: Pcosgen-train dataset (Dec 2023). https://doi.org/10.5281/zenodo.10430727

[56] A. Hosna, E. Merry, J. Gyalmo, Z. Alom, Z. Aung, and M. A. Azim, 'Transfer learning: a friendly introduction', Journal of Big Data, vol. 9, no. 1, p. 102, 2022.

[57] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, 'Rethinking the Inception Architecture for Computer Vision', in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 2818–2826.

[58] K. He, X. Zhang, S. Ren, and J. Sun, 'Deep Residual Learning for Image Recognition', in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770–778.

[59] P. Kottarathil, Polycystic ovary syndrome (pcos) dataset, https://www.kaggle.com/datasets/prasoonkottarathil/ polycystic-ovary-syndrome-pcos, accessed: 2025-05-02 (2021)

[60] P. Misra, A. S. Yadav, Impact of preprocessing methods on healthcare predictions, in: Proceedings of 2nd international conference on advanced computing and software engineering (ICACSE), 2019

[61] C. Khosla and B. S. Saini, "Enhancing Performance of Deep Learning Models with different Data Augmentation Techniques: A Survey," *2020 International Conference on Intelligent Engineering and Management (ICIEM)*, London, UK, 2020, pp. 79-85, doi: 10.1109/ICIEM48762.2020.9160048.

[62] Kapila, R., Ragunathan, T., Saleti, S., Lakshmi, T.J. and Ahmad, M.W., 2023. Heart Disease Prediction using Novel Quine McCluskey Binary Classifier (QMBC). IEEE Access, 11, pp.64324 – 64347.

[63] Venkatesh, C., Prasad, B.V.V.S., Khan, M., Babu, J.C. and Dasu, M.V., 2024. An automatic diagnostic model for the detection and classification of cardiovascular diseases based on swarm intelligence technique. Heliyon.

[64] Mahesh B. Machine learning algorithms-a review. International Journal of Science and Research (IJSR).[Internet]. 2020 Jan 5;9(1):381-6.

[65] Janiesch C, Zschech P, Heinrich K. Machine learning and deep learning. Electronic markets. 2021 Sep;31(3):685-95.

[66] Shiri FM, Perumal T, Mustapha N, Mohamed R. A comprehensive overview and comparative analysis on deep learning models: CNN, RNN, LSTM, GRU. arXiv preprint arXiv:2305.17473. 2023 May 27.

[67] Kaur M, Mohta A. A review of deep learning with recurrent neural network. In2019 International Conference on Smart Systems and Inventive Technology (ICSSIT) 2019 Nov 27 (pp. 460-465). IEEE.

[68] Fister Jr I, Yang XS, Fister I, Brest J, Fister D. A brief review of nature-inspired algorithms for optimization. arXiv preprint arXiv:1307.4186. 2013 Jul 16.

[69] C. Yuanlin and Li Maojun, "Optimal Configuration of Capacitors in Distribution Network Based on Niche Genetic Algorithm," *2007 Chinese Control Conference*, Zhangjiajie, China, 2007, pp. 727-730, doi: 10.1109/CHICC.2006.4346963.

[70] Zhao Wei and Lan Ying, "Adaptive particle swarm optimization neural network genetic algorithm in nonlinear function optimization extreme," *World Automation Congress 2012*, Puerto Vallarta, Mexico, 2012, pp. 1-4.

[71] C. Zefan and Y. Xiaodong, "Cuckoo search algorithm with deep search," *2017 3rd IEEE International Conference on Computer and Communications (ICCC)*, Chengdu, China, 2017, pp. 2241-2246, doi: 10.1109/CompComm.2017.8322934.

[72] M. Ali, I. Garip and I. Colak, "Improved Cuckoo Search Algorithm for Wind System Optimization," *2022 10th International Conference on Smart Grid (icSmartGrid)*, Istanbul, Turkey, 2022, pp. 431-435, doi: 10.1109/icSmartGrid55722.2022.9848713.

[73] Reddy, S.S., 2017. Optimal reactive power scheduling using cuckoo search algorithm. International Journal of Electrical and Computer Engineering, 7(5), pp.2349-2356.

[74] A. H. Elawdi, M. Kulkarni, C. H. Basha, S. Rafikiran and M. Manohar, "Improved Flower Pollination Algorithm Based Optimum Design of Reinforced Concrete Retaining Walls," *2024 International Conference on Integrated Intelligence and Communication Systems (ICIICS)*, Kalaburagi, India, 2024, pp. 1-5, doi: 10.1109/ICIICS63763.2024.10859574.

[75] K. E. Hoque, H. Aljamaan, Impact of hyperparameter tuning on machine learning models in stock price forecasting, IEEE Access 9 (2021) 163815–163830.

[76] R. Dwivedi, D. Dave, H. Naik, S. Singhal, R. Omer, P. Patel, B. Qian, Z. Wen, T. Shah, G. Morgan, et al., Explainable ai (xai): Core ideas, techniques, and solutions, ACM Computing Surveys 55 (9) (2023) 1–33.

[77] A. Adadi, M. Berrada, Peeking inside the black-box: a survey on explainable artificial intelligence (xai), IEEE access 6 (2018) 52138– 52160.

[78] J. Dieber, S. Kirrane, Why model why? assessing the strengths and limitations of lime, arXiv preprint arXiv:2012.00093 (2020).

[79] T. Katayama, K. Takahashi, Y. Shimomura, H. Takizawa, Xai-based feature importance analysis on loop optimization, in: 2024 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW), IEEE, 2024, pp. 782–791.

[80] D. Gunning, Explainable artificial intelligence (xai), Defense advanced research projects agency (DARPA), nd Web 2 (2) (2017) 1.

[81] F. Gabbay, S. Bar-Lev, O. Montano, N. Hadad, A lime-based explainable machine learning model for predicting the severity level of covid19 diagnosed patients, Applied Sciences 11 (21) (2021) 10417.

[82] M. T. Ribeiro, S. Singh, C. Guestrin, " why should i trust you?" explaining the predictions of any classifier, in: Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining, 2016, pp. 1135–1144.

[83] G. El-Khawaga, M. Abu-Elkheir, M. Reichert, Xai in the context of predictive process monitoring: An empirical analysis framework, Algorithms 15 (6) (2022) 199.