WJCMS

Wasit Journal of Computer and Mathematics Science

Journal Homepage: https://wjcm.uowasit.edu.iq/index.php/WJCM
e-ISSN: 2788-5879 p-ISSN: 2788-5887

Computational intelligence in the identification of Covid-19 patients by using KNN-SVM Classifier

Shaymaa Adnan Abdulrahman [1,*]

[1]University of Information Technology & Communications, College of Business Informatics Technology, Business Information

*Corresponding Author : Shaymaa Adnan Abdulrahman

DOI: https://doi.org/10.31185/wjcms.306
Received 14 October 2024; Accepted 20 November 2024; Available online 30 December 2024

ABSTRACT: Initiatives to mitigate the persistent coronavirus disease 2019 (COVID-19) crisis shown that quick, sensitive, and extensive screening is essential for managing the present epidemic and future pandemics. This virus seeks to infect the lungs by generating white, patchy opacities inside them. This research presents an advanced methodology employing deep learning techniques for the analysis of medical pictures pertaining to respiratory disorders. This experiment included two data sets, the initial one including normal lungs sourced from the Kaggle data pool. We acquired the anomalous lungs from https://github.com/muhammedtalo/COVID-19. We applied Principal Component Analysis (PCA) and Histogram of Gradients (HOG) as extract features. while we conducted a classification process using K nearest neighbors (KNN) and Support Vector Machine (SVM) algorithms . Results showed that the classification accuracy with SVM for Covid-19 identification is 88.54% while with KNN is 82.31%

Keywords: Covid-19, Chest X-rays, KNN, SVM, Classifier

1. INTRODUCTION

Pneumonia symptoms and chest X-ray examinations are commonly used to recognize Covid-19 [1]. The image-based examination of the chest is the primary imaging modality needed for the identification of that pandemic. A negative example on a standard chest X-ray, a positive case of Covid-19, and a positive instance of severe acute respiratory syndrome are displayed in Figure 1. Prior research has utilized several conventional machine learning techniques to autonomously categorize digital chest pictures [2]. In reference [3], a Support Vector Machine (SVM) classifier employed three statistical variables derived from lung texture to differentiate between malignant and benign lung nodules. The Backpropagation Network [4] employed a gray-level co-occurrence matrix technique to categorize pictures as ordinary or malignant. Given a sufficient quantity of annotated photos, deep learning methodologies have proven to surpass traditional machine learning techniques. The CNN architecture is a prominent deep learning methodology which demonstrated exceptional performance regarding medical imaging. The principal success of CNNs stems from their capacity to autonomously learn features from domain-specific pictures, in contrast to traditional machine learning techniques. The prevalent approach for training CNN architecture is transferring acquired information from a pre-trained network which effectively performed one job to a new one [5]. This approach is more efficient and simpler to implement, requiring no extensive annotated dataset for training; hence, several researchers prefer this strategy, particularly in the realm of medical imaging. There are three main ways to implement transfer learning: a) "shallow tuning," which involves changing only the last classification layer's parameters for the new task while the rest of the network's parameters remain unchanged; b) "deep tuning," which aims to retrain the entire network's parameters in an end-to-end fashion; and c) "fine-tuning," which involves training more layers by adjusting the learning parameters until a significant performance improvement is achieved. The transfer of information through a fine-tuning method shown exceptional efficacy in the categorization of chest X-ray images . Proposals for class decomposition seek to improve low-variance classifiers and provide greater flexibility in their decision bounds. It seeks to streamline the local configuration of a dataset to address any anomalies in the data distribution. Numerous automated learning workbooks have previously employed class decomposition as a pre-processing technique to enhance the efficacy of various classification models.

2. LITERATURE REVIEW

The following are brief accounts of relevant peer-reviewed articles to the study. A research employed Bayes-SqueezeNet to uncover Corona [6]. The suggested network uses Bayesian optimization for model training and offline dataset augmentation. It categorized these pictures as either ordinary, viral pneumonia, or Covid-19 depending on Bayes-SqueezeNet. The network said that it could address the issue of uneven data derived from public sources through data augmentation. A study created an additional convolutional neural network, CoroNet [7], in order to recognize Corona infections based on exam pictures. That methodology has been constructed from pre-trained convolutional neural network (CNN) known as Xception [8]. CoroNet was built upon the Xception model, which featured a dropout stratum and two entirely linked strata. CoroNet contains 33,969,964 trainable parameters overall from 33,969,964 and 54,528 non-training parameters. The network was implemented in a study for a triple categorization (Covid-19, pneumonia, and ordinary) and a four-class categorization (Covid-19, bacterial pneumonia, viral pneumonia, and ordinary). CovidGAN [9] is developed as an auxiliary classifier generative adversarial grid for the purpose of identifying Covid-19. It was derived from GAN (generative adversarial network) [10]. A study that employed the pre-trained VGG-16 [11] and integrated it with four custom layers at the conclusion devised the architecture of CovidGan. These strata involved universal pooling stratum, a 64-unit dense stratum, a dropout stratum with a 0.5 probability. The effectiveness of classification is enhanced by GAN methodology to create X-ray pictures. An additional convolutional neural network model, DarkCovidNet [12], was constructed on the basis of the DarkNet architecture [13] to detect Covid-19 using thoracic X-rays. The DarkCovidNet has progressively higher filters and fewer layers than the original DarkNet. We assessed that procedure against binary categorization (Covid-19 and zero results) and a ternary categorization (Covid-19, zero results, and pneumonia). The investigation in [14] employed pre-trained convolutional neural networks, including VGG1-19, MobileNet-v2 [13], Inception [14], Xception [15][16], and Inception ResNet-v2 [16], to identify Covid-19 in medical pictures. We employed these pre-trained CNNs for binary and ternary classification tasks, employing two datasets comprising pictures of the pandemic, bacterial pneumonia, viral pneumonia, and safe states.

3.MATERIALS AND METHODS

Figure 1 depicts the suggested research technique for this project. Preprocessing, feature extraction, dimensionality reduction, and classification are all included in this four-step procedure.

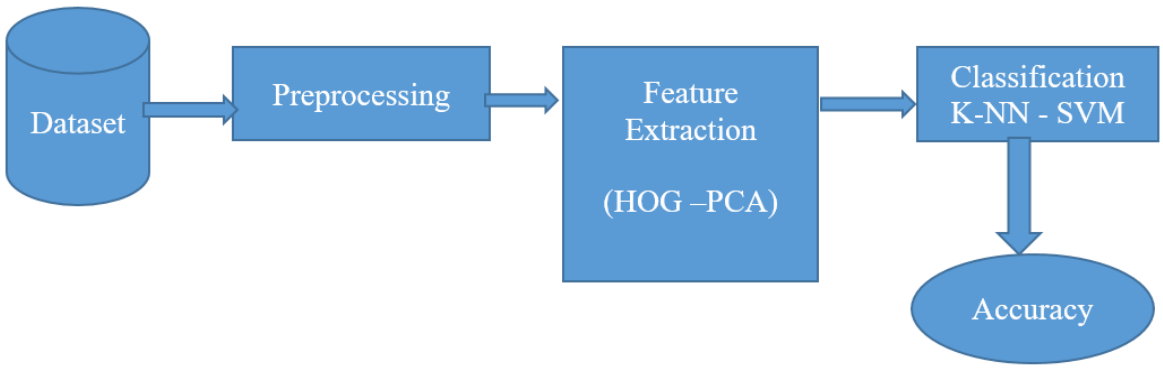

FIGURE 1: The suggested work

Step 1: The preprocessing of the x-ray images that were collected and diagnosed with COVID-19, as well as the removal of noise following the application of segmentation technique.

Step 2: Extraction of features: In the field of medical image processing, it is impossible to extract features from an x-ray image due to the interaction between the features and their companions.

Step 3: The procedure of reducing dimensionality. We will employ the PCA to reduce the dimension of the feature after extracting it from HOG.

Step 4: Classification Process: Utilize categorization tools, such as KNN and SVM, to categorize Covid-related pictures and ordinary and healthy individuals without lung disease.

3.1 COLLECTION OF DATA

In this study, data related to Covid-19(-) are based on the "Kaggle " Repository. They were obtained at open-I repository (http://openi.nlm.nih.gov). [Retrieved on May 15, 2024][17][18]. The specifics of this collection are illustrated in Table 1.

Table 1: Groups of data

Group	No	Source
Covid-19(.+)without ,MERS, SARS, and .ARDS	30	Github1 (D. Joseph Caohen)
Covid-19 (.-)	30	Kaggle- (pneumonia pictures

3.2 FEATURE EXTRACTION

Actually, features extraction is just the process of identifying the most significant information in data by extracting interesting features from it. In technical terms, raw data is transformed into a features vector so that a machine learning algorithm can use it to learn about the data. Histogram of Gradients (HOG) and Principal Component Analysis (PCA) algorithms were employed in the feature extraction stage in our study

3.2.1 HISTOGRAM OF GRADIENTS (HOG)

Image brightness produces a feature vector referred to as a histogram of directed gradients. HOG illustrates object contours, generates local domain histogram from the gradients of neighboring pixels, crucial for modifying lighting and shadows. The brightness gradient is computed by determining the gradient intensity (m) and gradient direction (θ) for these brightness pixels.

$$m = (x,y) = \sqrt{f_{x1}(x.,y)^2 + f_{y1}(x.,y)^2} \quad \ldots\ldots\ldots (1)$$

$$\theta(xi,yi) = tan^{-1}\left(\frac{f_y 2(x,y)^2}{f_x 2(x,y)}\right) \quad \ldots\ldots\ldots (2)$$

$$f_{x1,y1}(x,y) = L(xi+1,yi) - l(xi-1,yi) \quad \ldots\ldots\ldots (3)$$

While Making to Cell: We make histograms from a 5x5-pixel region (See Figure. 2). The histogram is made from the summation of gradient strength that has the same directional-bins. We divide gradient directions into 9bins (Figure. 3)

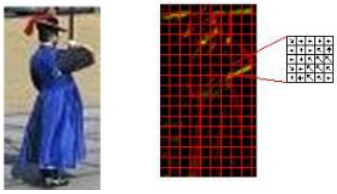

FIGURE. 2: (a) source image, (b) cell image

FIGURE. 3: Sample of a histogram of oriented gradients

In Making to Block: A block is defined with each 3x3 cells. In regularization, the cell having maximum gradient strength represents the block. To determine a magnitude weight, HOG steps are assigned to each pixel's half of the descriptor's width, or sigma (in MATHLAB Programming)

Implementation

Step 1: input X-ray medical image

step2 :Normalize the medical pictures that are the square root of x-ray image intensity depending on their sorts

step 3: Measure the orientation and magnitude of the gradient. Equation (4) represents magnitude, while equation (5) represents direction [5]

$$GM1 \parallel \nabla f \parallel = \sqrt{f_x} + f_x^2 \quad \text{.....(4)}$$

$$(GD)\theta = tan^{-1}\frac{f_x}{f_y.} \quad \text{.....(5)}$$

where f_x is the derivative regarding (wx.ry.t) x gradient in the x- direction, while f_y is the derivative concerning (wx.ry.t) x gradient in the y – direction

Step 4: Make a window, then separate it to cells. Every single cell is a pixel. Create a histogram of the orientation gradient.

Step 5: Gather the cells by sorting them in big collection one. Normalize that collection.

Step 6: As data are taken from HGO, use either machine learning algorithm or classification.

3.2.2 PRINCIPAL COMPONENT ANALYSIS (PCA)

The procedure which is conducted in numerous scientific disciplines. PCA is a technique that turns a set of possibly interconnected observations into linearly non-interconnected variables by applying an equilateral transformation. These variables are referred to as principle components. We employ this modification to guarantee that the initial principle component demonstrates the greatest variance, while each subsequent principal component maintains the highest variance feasible while remaining orthogonal to the preceding components. An uncorrelated orthogonal basis set is represented by the vectors produced by PCA. The PCA is fundamentally reliant on Eigenvector-based multivariate analysis. Principal component analysis may be performed utilizing a single value decomposition, eigenvalue decomposition, or correlation matrix [12]. Component scores, known as factor scores, are the converted rates of variables associated with a single data point. Loadings, which are the weights applied to each normalized original variable to obtain the component score, are the typical outputs of a principal component analysis. Show algorithm 1

Algorithm 1 PCA for Reduction
Input: Image features, the number of features.
Output: The most principal components

1: Standardize data by Z scored to transform all variables to be in the same scale

$$Z = \frac{X-\mu}{\sigma} \quad \text{..... (6)}$$

2: Compute the covariance matrix (A) of the standardized data for each two variables X and Y;

$$\mathrm{COV(X,Y)} = \frac{\sum_{i=1}^{n}(X_i - x)(Y_i - y)}{N-1} \quad \text{..... (7)}$$

3: Calculate the eigenvectors (Nv) and eigenvalues (_) of the covariance matrix and store the eigenvalues in a descending order;

$$\gamma(A - \Lambda I) = 0 \quad \text{..... (8)}$$ where I is the identity matrix.

4: Sort the eigenvectors according to their decreasing order of eigenvalues;
5: Choose k eigenvectors with the largest eigenvalues;
6: Transform the data from the original dimensions to the reduced ones (k) represented by the principal components;
7: return the most k principal components.

3.3 CLASSIFICATION PROCESS

A comparison of various machine learning classifiers was made in order to create a strong predictive model. Two classifiers were employed: such as:- Support Vector Machine (SVM) and K Nearest Neighbors (KNN.).

3.3.1 K-NEAREST NEIGHBOR

Considered the best data mining way, KNN classification aims to spot imbalanced data [26,36,37,39]. Different primary research trajectories exist. An appropriate K value must be established. Another approach involves utilizing the distance function to ascertain the K nearest neighbors. In probability theory, cross-validation is a frequently employed

technique for determining the K value. In the context of a training dataset, the K value assumes significance. Nonetheless, the training sample space displays a distribution of varying densities across the samples. This presents a novel and intricate issue: the prediction of classes requires the use of unique K values due to the presence of varied samples. In a recent work, Cheng et al. [2] calculated parameter K for KNN classification, figuring out a suitable value for every new dataset. Shaymaa et al. [5] devised a KNN method that involves the data-driven calculation of the K parameter. Zahraa A. Jaaz et al. [3] devised a method for accurately determining K for KNN classification. Euclidean distance, clarified below, is employed by the majority of KNN classification methods, despite the existence of numerous distance functions:

$$\text{Dist}(X,Y)=\sqrt{\sum_{i-1}^{p}(Xi-Yi)^2} \quad \ldots\ldots\ldots(9)$$

Xi and Yi (i = 1 … N) are attributes of two samples/instances X and Y, KNN should be categorized for testing and training

Training [1]

1. Pick medical pictures.
2. After that training, these pictures will be readable.
3. Re-shape each picture.
4. Choose pre-edited medical pictures to single out characteristics based on HGO, forming a vector of local features specific to the image.
5. The feature vector constructs the X-ray pictures in matrix-based rows using the local feature.
6. Repeat the image for the test.
7. Prepare the KNN method for testing.

Testing [1]

1. Study pictures to be tested.
2. Training data is applied after the KNN is applied.
3. The image is tagged and it exits if all K neighbors carry similar titles. In every other case, build the distance matrix by calculating the pairwise distance between the K=neighbors.

3.3.2 SUPPORT VECTOR MACHINE (SVM)

A popular guided machine way for classification problems has been Support Vector Machine (SVM). SVM generates an ideal boundary, typically referred to as a hyper-plane. This hyper-plane functions to partition a specified number of dimensions into several groups. Consequently, we may classify a new data point into the best suitable category anytime assessment is required. The Support Vector Machine (SVM) identifies points and vectors that denote the boundaries for the creation of hyper-planes. The technique termed SVM [3] designates these extremes as support vectors, as indicated in Figure. 4.

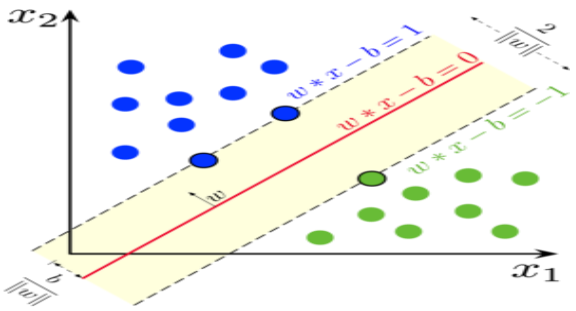

FIGURE 4: support vector machine approach

4. RESULTS AND ANALYSIS

Two data sets were produced by this research. Thirty X-ray medical images which tested positive for Covid-19 and thirty images that tested negative for the virus made up the initial data set. The University of Montreal's postdoctoral researcher Joseph Cohen disseminated pandemic pictures from the relevant source [%] [16] (www.kaggle.com). [Retrieved May 15, 2024]. The said source compiled Corona-related pictures of pneumonia [17]. Covid-19 (+) does not include MERS, SARS, or ARDS. Covid acquired 133 pictures from the above-mentioned repository for the secondary dataset. The suggested models analyzed the two datasets independently. We utilized two datasets for feature extraction employing HGO and PCA. We used the MATLAB 2015b machine learning tools to conduct experimental trials and SVM to distinguish between sick and non-infected lungs. In order to assess the suggested model, this study employed a 10-fold random cross-validation and evaluated the effectiveness of two machine learning algorithms. Equation (10)may be used to see the accuracy formula. Table (2) displays the accuracy of the two classifiers using a 10-fold validation.

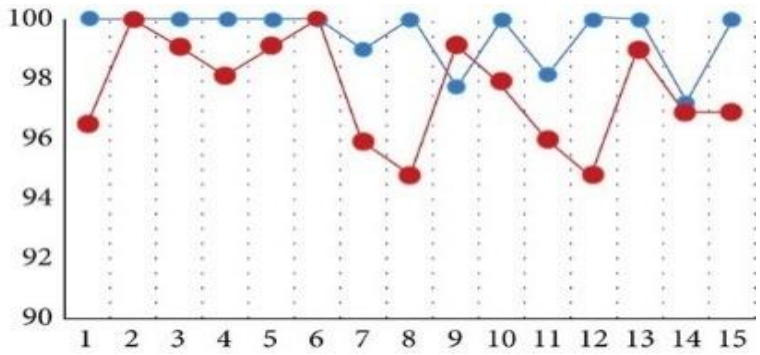

FIGURE 5: Classification of our proposed work

In addition to discussing the experimental results of the suggested system, this part provides a thorough comparison with other models. Accuracy, precision, recall, and F1_score are the four efficiency indicators that form the basis of the evaluation technique. Equations (11) and (12), respectively, measure several characteristics, including False Positive (FP) and False Negative (FN), as well as measuring the performance of our suggested technique. Table- 3- displays the FP and FN outcomes. While The Table -4- illustrates the compression results of previous works.

$$\text{Precision} = \frac{TP+TN}{TP+TN+FP+FN} \quad \ldots\ldots\ldots(10)$$

$$\text{False Positive} = \frac{TP}{TP+TN+FP+FN} \quad \ldots\ldots(11)$$

$$\text{False Negative} = \frac{FN}{TP+TN+FP+FN} \quad \ldots\ldots\ldots(12)$$

$$\text{Recall} = \frac{TN}{TP+FN} \quad \ldots\ldots\ldots(13)$$

$$\text{F1-Score} = 2*\frac{Precision*Recall}{precision+Recall} \quad \ldots\ldots\ldots(14)$$

Table 2: Precise prediction of pandemic mortality using a cross-validation of 10 folds validation

Classifier	Cross Validation	Precision	Recall	Accuracy
SVM	10-fold	83.33	75.76	88.54%
KNN	10-fold	74.29	78.79	82.31%

Table 3: the analysis of the effectiveness of the suggested work following FP and FN

Classifier	False Positive	False negative
SVM	5	7
KNN	6	8

Table -4-Comparison of the suggested study and previous research in terms of accuracy using various methods

No	Authors	Preprocessing &Feature Extraction	Machine Learning Technique	Accuracy	years
1	Ucar .F [6]	CNN	Bayes-SqueezeNet	0.983	2020
2	Asif Iqbal Khan. et.al [7]	CNN	Deep convolutional neural network	89.6%	2020
3	CholletF. Xception [8]	convolutional neural networks	convolutional neural network	0.945	2017
4	ABDUL WAHEED et .al [9]	-	CNN	95%	2020
5	Ioannis D. Apostolopoulos and · Tzani A. Mpesiana [12]	(VGG19 and MobileNet v2),	CNN	94.72	2020
6	Anant Agrawal .et.al [19]	-	SVM	78.6%	2018
7	**Proposed work**	**HOG and PCA**	**SVM**	**88.54%**	-------
			KNN	**82.31%**	

5. CONCLUSION

Efforts to eliminate Covid-19 and related epidemic infections might benefit from the use of artificial intelligence (AI). Nonetheless, AI is still in an initial phase regarding the resolution of COVID-19 issues. To employ AI in this study, it is essential to furnish a database including medical pictures for image analysis and illness detection. We employed two categories of data for this aim. The first category pertains to individuals infected with COVID-19, whilst the second category pertains to uninfected individuals. Both techniques are used in this study. In order to assess the proposed work model, we calculated the FP and FN values. The outcomes are encouraging; SVM achieves an 88.54% detection accuracy for COVID-19, while KNN achieves an 82.31% accuracy. According to this research, SVM is a more reliable method than KNN and K-Nearest Neighbors (KNN).

Funding

None

ACKNOWLEDGEMENT

None

CONFLICTS OF INTEREST

The author declares no conflict of interest.

REFERENCES

[1] S. A. Abdulrahman, W. Khalifa, M. Roushdy, and A. B. M. Salem, "Comparative study for 8 computational intelligence algorithms for human identification," vol. 36, 2020.

[2] A. Narin, C. Kaya, and Z. Pamuk, "Automatic detection of coronavirus disease (COVID-19) using X-ray images and deep convolutional neural networks," arXiv preprint, 2020.

[3] A Abdalrada ,et. al “Predicting Diabetes Disease Occurrence Using Logistic Regression: An Early Detection Approach” Iraqi Journal For Computer Science and Mathematics, vol 5 , pp160-167, 2024
[4] Z. A. Jaaz, S. A. Abdulrahman, and H. M. Mushgil, "A dynamic task scheduling model for mobile cloud computing," ICEE Proc., 2022.
[5] L. Wang and A. Wong, "COVID-Net: Tailored deep convolution neural network design for detection of COVID-19 cases from chest images," 2020.
[6] S. A. Abdulrahman, E. Q. Ahmed, Z. A. Jaaz, and A. R. Ali, "Intrusion detection in wireless body area network using attentive with graphical bidirectional long-short term memory," vol. 19, 2023.
[7] AS Abdalrada, et.al “Relationship between angiotensin converting enzyme gene and cardiac autonomic neuropathy among Australian population” Springer International Publishing, pp135-146 ,2018
[8] F. Ucar and D. Korkmaz, "COVIDiagnosis-Net: Deep Bayes-SqueezeNet based diagnosis of the coronavirus disease (COVID-19) from X-ray images," vol. 140, 2020.
[9] A. I. Khan, J. L. Shah, and M. M. Bhat, "CoroNet: A deep neural network for detection and diagnosis of COVID-19 from chest X-ray images," vol. 19, 2020.
[10] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," arXiv preprint, 2017.
[11] A. Waheed, M. Goyal, D. Gupta, A. Khanna, F. Al-Turjman, and P. R. Pinheiro, "CovidGAN: Data augmentation using auxiliary classifier GAN for improved COVID-19 detection," IEEE Softw., vol. 8, no. 19, pp. 16–23, 2020.
[12] R. A. Jaafar and S. A. Abdulrahman, "Detection and classification of alcoholics using electroencephalogram signal and support vector machine," vol. 2, no. 1, pp. 14–21, 2020.
[13] Ahmad Shaker Abdalrada and , Naseer Ali Husieen ,” A Comparative Performance Evaluation of Hive and Map Reduce for Big-Data” IISTE, vol 5, pp 1-16 ,2015
[14] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," arXiv preprint, 2016.
[15] A.-B. M. Salem and S. A. Abdulrahman, "An efficient deep belief network for detection of coronavirus disease COVID-19," vol. 2, no. 1, pp. 5–13, 2020.
[16] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, et al., "MobileNets: Efficient convolutional neural networks for mobile vision applications," arXiv preprint 1704.04861, 2017.
[17] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," pp. 4278–4284, 2018.
[18] S. A. Abdulrahman and B. Alhayani, "A comprehensive survey on the biometric systems based on physiological and behavioural characteristics," Mater. Today Proc., vol. 80, pp. 2642–2646, 2021.
[19] Kaggle, "[Online]. Available: www.kaggle.com. [Accessed: May 15, 2024].
[20] GitHub, "[Online]. Available: www.github.com. [Accessed: May 15, 2024].
[21] OpenI, "[Online]. Available: http://openi.nlm.nih.gov. [Accessed: May 15, 2024].
[22] A. Agrawal, H. Agrawal, S. Mittal, and M. Sharma, "Disease prediction using machine learning," 2018.
[23] AS Abdalrada, et.al “Tahsien Al-Quraishi, and Herbert F. Jelinek." Relationship between angiotensin converting enzyme gene and cardiac autonomic neuropathy among Australian population”. pp135-146 , 2018
[24] A Abdalrada and IR Khan : Assessing Institutional Performance Using Machine Learning Algorithms” Wasit Journal of Computer and Mathematics Science, vol 3 ,pp11-21 2024