

Multi-Objective Genetic Algorithm Based Deep Learning Model for Automated COVID-19 Detection using Medical Image Data

Shrey Bansal

IIT Delhi: Indian Institute of Technology Delhi

Mukul Singh

IIT Delhi: Indian Institute of Technology Delhi

Rahul Dubey (✉ rahul.dubey2011@ieee.org)

Bosch Engineering GmbH <https://orcid.org/0000-0002-1662-3718>

Bijaya Ketan Panigrahi

IITD: Indian Institute of Technology Delhi

Research Article

Keywords: Coronavirus (COVID-19) , Convolutional Autoencoder , Multi-Objective Genetic Algorithm , Feature subset selectio

Posted Date: April 16th, 2021

DOI: <https://doi.org/10.21203/rs.3.rs-311776/v1>

License:  This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Version of Record: A version of this preprint was published at Journal of Medical and Biological Engineering on September 1st, 2021. See the published version at <https://doi.org/10.1007/s40846-021-00653-9>.

Noname manuscript No.
(will be inserted by the editor)

Multi-Objective Genetic Algorithm Based Deep Learning Model for Automated COVID-19 Detection using Medical Image Data

S. Bansal · M. Singh · R. K. Dubey ·
B. K. Panigrahi

Received: date / Accepted: date

Abstract In early 2020, the world is amid a significant pandemic due to the novel coronavirus disease outbreak, commonly called the COVID-19. Coronavirus is a lung infection disease caused by the Severe Acute Respiratory Syndrome Coronavirus 2 virus (SARS-CoV-2). Because of its high transmission rate, it is crucial to detect cases as soon as possible to effectively control the spread of this pandemic and treat patients in the early stages. RT-PCR-based kits are the current standard kits used for COVID-19 diagnosis, but these tests take much time despite their high precision. A faster automated diagnostic tool is required for the effective screening of COVID-19. In this study, a new semi-supervised feature learning technique is proposed to screen COVID-19 patients using chest CT Scans. The model proposed in this study uses a three-step architecture, consisting of a Convolutional Autoencoder based unsupervised feature extractor, a Multi-Objective Genetic Algorithm based feature selector, and a Bagging Ensemble of Support Vector Machines(SVMs) based classifier. The Autoencoder generates a diverse set of features from the images, and an optimal subset, free of redundant and irrelevant features, is selected by the evolutionary selector. The Ensemble of SVMs then performs the binary classification of the features. The proposed architecture has been designed to provide precise and robust diagnostics for binary classification (COVID vs.nonCOVID). A Dataset of 1252 COVID-19 CT scan images, collected from 60 patients, has been used to train and evaluate the model. The experimen-

Shrey Bansal · Mukul Singh
Computer Science and Engineering Department, Indian Institute of Technology Delhi, New Delhi 110016, India
E-mail: Shrey.Bansal.cs118@cse.iitd.ac.in — Mukul.Singh.cs118@cse.iitd.ac.in

Rahul Kumar Dubey
Robert Bosch Engineering and Business Solutions Private Limited Head Office, 123, Hosur Rd, 7th Block, Koramangala, Bengaluru, Karnataka 560095, India
E-mail: rahul.dubey2011@ieee.org

Bijaya Ketan Panigrahi
Electrical Engineering Department, Indian Institute of Technology Delhi, New Delhi 110016, India
E-mail: bkpanigrahi@ee.iitd.ac.in

tal results prove the superiority of the proposed methodology in comparison to existing methods.

Keywords Coronavirus (COVID-19) · Convolutional Autoencoder · Multi-Objective Genetic Algorithm · Feature subset selection

1 Introduction

A chest infection disease affects the functioning of the lungs[1]. The common lung infections are lung cancer, Chronic Obstructive Pulmonary Disease (COPD), bronchitis, pneumonia and, asthma. Coronavirus disease (COVID-19) is a of lung infection disease caused due to the novel discovered virus known as SARS- CoV-2[2]. COVID-19 began with reports of unknown causes of pneumonia in Wuhan City, China, around December 2019. The worldwide economy was impacted by the unprecedented rise in COVID-19 cases and it has been declared a pandemic by the WHO. [3].

On 18 June 2020, a total of 8,379,081 patients became infected with COVID-19, and 215 countries listed 450,101 deaths [3]. The standard diagnostic test for COVID-19 is the Reverse Transcriptase Polymerase Chain Reaction (RT- PCR) [4]. Due to PCR's high selectivity and sensitivity, it is prevalent. The limitations of the PCR technique are 1) time consuming, 2) expensive, 3) shortage of kits, and 4) long production time [5]. A faster and cheaper testing mechanism is required to tackle the alarming rates of spread of COVID-19. Radiological analysis like Chest CT (Computed Tomography) scans and X-Rays produce high hit-rate in COVID-19 diagnosis. [6] established a high correlation between radiological results and RT-PCR. The above reasons encouraged developing a cheaper and faster COVID-19 screening mechanism using a radiological approach [7].

From the comprehensive analysis of the COVID-19 diagnosis field, it is inferred that the best alternative for COVID-19 detection to the RT-PCR test kits is chest radiography (X-rays and CT scan)[8]. However, CT scan modality seems to be more efficient than chest X-ray for the following reasons: 1) X-rays provide only a 2D perspective whereas CT scan provides a detailed 3D view of the organ, 2) in X-rays, ribs overlap the lungs and heart, whereas, the CT scan does not. A deep-learning-based three-step model is proposed for CT-scan based screening, consisting of a Convolutional Autoencoder based unsupervised feature extractor, an evolutionary algorithm based feature subset selector, and a feature classifier.

A CNN-based dense Autoencoder has been used as the feature extractor because of CNN's high representational power and the generality of unsupervised learning from it. The Autoencoder ensures an accurate and diverse feature set, while the feature selector removes all redundant and irrelevant features improving the performance. After obtaining a reduced representation of raw data as a diverse set of features, the evolutionary algorithm based feature subset selectors is used to select optimal feature subsets. Finally, the Bagging Ensemble of Support Vector Machines (SVM) is trained on the subsets chosen by the various selectors, and their performance is compared.

2 Related Works

Table 1 consists of various state of the art techniques currently available in the literature of COVID-19 diagnosis. Further, a detailed analysis of the review is presented.

Works from [10, 11, 13, 14] have used pre-trained CNN models for COVID-19 diagnosis. Transfer Learning techniques are useful when data is limited, but they often fail to learn intricate features unique to the required dataset. Some authors have performed fine-tuning, but retraining the last few layers might not change the basic features extracted by the CNN.

Authors in [9, 12, 15, 16] have used random forest, peekaboo, and segmentation classification. They have not used explicit feature extractors, and since the classification uses chest CT images, a deep feature extractor architecture like CNN might perform significantly better in this case.

The authors in the literature have obtained quality results by focusing only on feature extractors and classifiers. In our work, we propose to shift the attention from feature extraction to feature selection as it is critical to remove the redundant features in an unsupervised extractor and improve the performance of any standard classifier.

The author in [17] obtained improved results using MO-DE[18] feature selector over Deep CNN models, thus showcasing the importance of proper feature selection technique in medical image classification. We extend their work further and try to analyze and compare various feature reduction and selection techniques ranging from linear dimensionality reduction (PCA) to various Multi-Objective Feature Selectors. We obtained state-of-the-art results, validating their results, and obtaining an improved, robust model for COVID-19 screening.

Further, authors in [19] have found genetic selectors to outperform standard results on the Flavia dataset. [20] uses an NSGA-II based MOGA for feature selection and evaluates its performance on various datasets. [21] shows the use of the GA based feature selector for network intrusion detection. [22] compares GA based feature selectors on medical datasets focusing on diagnostic radiology. Authors of [22] compare GA based feature selectors to other approaches. In the stated studies, optimization of internal parameters of the MOGA has not been explored. Further, there is no comparative analysis among MOGA and other multi-objective evolutionary techniques for feature selection on medical images. Multi-Objective Optimization using Evolutionary Algorithms has not been well explored in its use as a feature selector.

We try to improve upon the previous works by analyzing the effects of optimizing parameters of MOGA. We also studied and compared MOGA with other multi-objective evolutionary techniques for feature selection on COVID-19 CT Scan Image Dataset, not done previously by any works.

3 Theoretical Background

3.1 Autoencoder (AE) based Feature Extractor

Autoencoders [23] are unsupervised learning methods trained to reconstruct their inputs, usually by going through a compressed representation of lower dimensionality [24]. Structurally an AE comprises two parts, namely an Encoder and a Decoder. Figure 1 summarizes the structure of an AE.

The encoder (E) converts the input image (x) to an encoded representation (h), which reflects the features of the image due to the constraint to reduce dimensionality. An encoder deterministically maps its input to a reduced representation generally using an affine map:

$$h = E(W \cdot x + b) \quad (1)$$

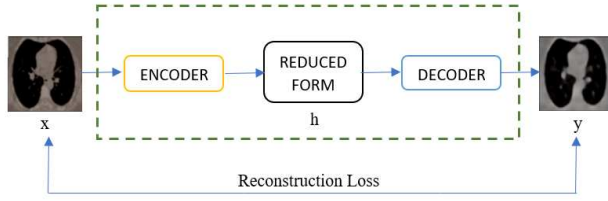


Figure 1: Schema of basic Autoencoder

here W denotes the weights for the encoder part, b represents the bias, and h represents the reduced representation. Similarly, the decoder (D) takes the reduced representation (h) and outputs the reconstructed image (y). An Autoencoder is trained to minimize the reconstruction error of its input. Hence, training of AE can be seen as a minimization of the following cost function:

$$\text{Cost} = \frac{1}{N} \sum_j \mathcal{L}_{\text{oss}}[x_i, y_i] \quad (2)$$

where N represents the number of images, x_i and y_i represent the i^{th} input-output image pair, and \mathcal{L}_{oss} is the reconstruction error between two images. Mean Squared Error has been used as the reconstruction error. CAE combines convolutional operations with the architecture of an AE. The authors of [25] have shown that CAE shows high accuracy in finger vein identification. Since CNN can extract a very detailed set of feature maps from images, convolutional AE has been used as a feature extractor in this study.

3.2 Multi Objective Genetic Algorithm based Feature Selector

3.2.1 Multi Objective Genetic Algorithm (MOGA)

Multi-Objective Optimization is the process of simultaneously optimizing more than one competing objective function. Two Objectives have been considered in this work, namely, classification accuracy and size of feature subset. These are competing objectives, and a single solution optimizing both might not exist. An alternative is to generate a set known as the Pareto Optimal set of solutions. A Pareto set is a set of solutions where no solution is dominated by any other solution in the set. There is always a degradation in some objectives, required to improve any objective in a Pareto set of solutions.

Consider a set of M objectives that have to be minimized. $\mathcal{H} = \{h_1, h_2, \dots, h_M\}$. Consider $x_1, x_2 \in \{\text{Pareto Set}\}$, then x_1 dominates x_2 if:

$$\forall i = 1, 2, \dots, M, \{h_i(x_1) \leq h_i(x_2)\} \text{ and} \\ \exists i = 1, 2, \dots, M, \{h_i(x_1) < h_i(x_2)\} \quad (3)$$

A solution is said to be Pareto Optimal if there exists no solution which dominates it. All such Pareto Optimal solutions together form the Pareto Optimal Set.

There exist various algorithms for multi-objective Genetic Optimizations. NSGA-II [26] is one such elitist principle-based algorithm much superior to classic gradient-based approaches. NSGA-II has been used to carry out the multi-objective feature subset selection in this study. Figure 2 summarizes the implementation of NSGA-II.

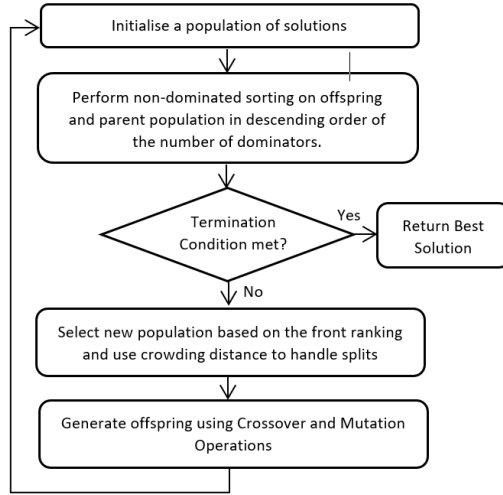


Figure 2: The working of NSGA-II

3.2.2 Initial Population and Encoding

Solutions in the population (a.k.a chromosomes) are represented as binary strings. The i^{th} gene in a chromosome is one if the solution contains the i^{th} feature of the input set. For the initial population, random binary chromosomes have been generated.

3.2.3 Crossover and Mutation

The creation of two new offspring chromosomes using the selected parent pair is known as crossover. Single point crossover has been used in this work, where each gene is randomly selected from one of the parents. Parents are selected using tournament-based selection.

Mutation conserves population diversity. Mutation involves random modifications in the value of the chromosomes. Random bit flip has been used as the mutation operator in this study.

3.2.4 Termination

The MOGA based selector terminates when either the maximum number of generations or the stall generation limit has been reached. After termination, the selector returns the final population with objective scores and front rankings.

3.3 Ensemble SVM based Classification

The SVM ensemble with Bagging is used in classification as SVM is a weak learner[27]. Using many small classifiers can increase robustness and produce low error. Bagging [28], uses randomized training sets for creating different models. A single classifier's training set is randomly generated by drawing N random data points (N is the size of the original training set) from the original training set with replacement. Figure 3 illustrates the structure of the bagging ensemble-based SVM.

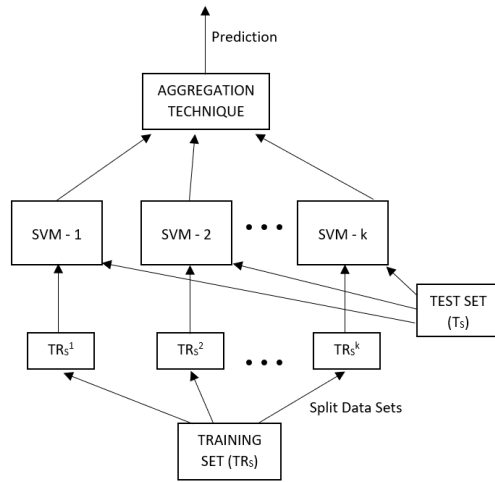


Figure 3: A general architecture of SVM ensemble with an aggregation step

As described above, bootstrap builds K duplicate training datasets from the given training data set (TR) $\{TR_k | k = 1, 2, \dots, K\}$ using random re-sampling with replacement.

After training, the independently trained SVMs are aggregated. Thus, majority voting has been used in the study because it uses upper layer SVM to combine several lower layer SVMs (double layer hierarchical combining).

4 Proposed Method

A 3-step architecture is proposed for the screening of COVID-19 chest CT Scans. The proposed architecture consists of a feature extractor, a feature selector, and a classifier. Flowchart summarizing the proposed architecture is depicted in Figure 5

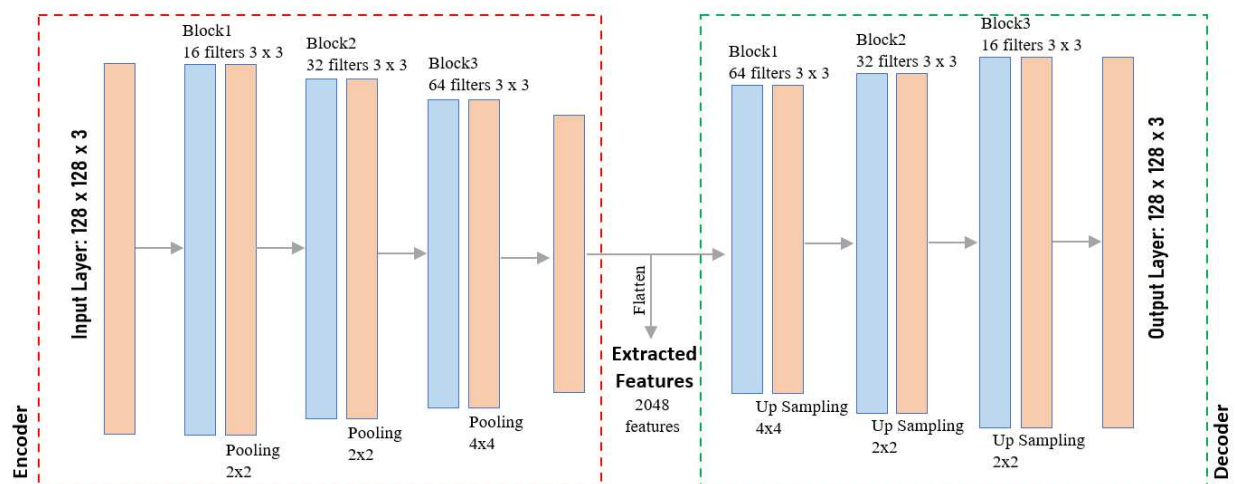


Figure 4: Proposed architecture of Convolution Auto-encoder

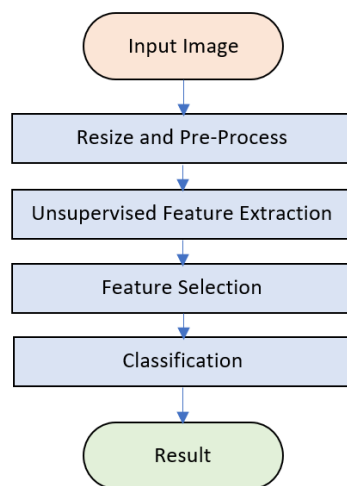


Figure 5: Flowchart summarizing the proposed architecture

An autoencoder based unsupervised learning approach is used to generate features from the CT scan images automatically. This gives us a diverse feature set, essential for this classification.

Though diverse, the features extracted by the Autoencoder have very high dimensionality and suffer from a redundancy of features. To remove the extra features, a MOGA based feature selector is proposed to select an optimal set of features.

Finally, for classification, a bagging based ensemble of support vector machines is used to carry out the binary classification of the feature sets into COVID-19

and non-COVID classes. A brief outline of the various methods is highlighted in the subsequent study.

4.1 Auto Encoder Structure and Training

The input image of size $128 \times 128 \times 3$, is fed into the CNN, which contains convolutional layers (kernel size 3) and max-pooling layers (downscaling factor of 2). ReLu activation is applied after every convolution. The encoder layers have 32, 16, and 8 filters (output channels), respectively. A decoder follows the encoder to reconstruct the image using deconvolution and up-sampling layers. The output of the encoder has the shape $14 \times 14 \times 16$. This is flattened to generate a feature vector of length 2048 per CT Scan image. The CNN architecture has been summarized in Figure 4.

The Auto Encoder is trained using the training set with the validation set for validation, as explained in section 4.1. Adam optimizer has been used for training the AE, with Mean Squared Error (MSE) as the loss function. The AE has been trained for two hundred epochs with a batch size of 10 per epoch. Figure 6 shows a reconstruction of test set images by AE.

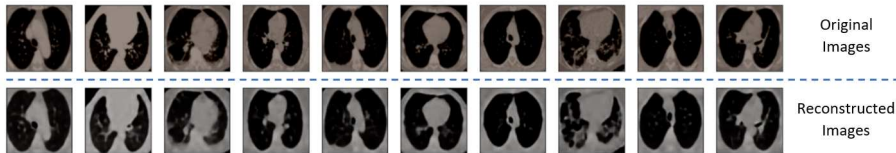


Figure 6: Original and Reconstructed (from Convolution Autoencoder) Chest CT Images from validation set

4.2 Feature Selector

The feature extractor extracts 2048 features from an input image of $128 \times 128 \times 3$. MOGA has been applied for selecting a superlative set from the extracted features using two fitness criteria:

$$\mathcal{C}_1 = \frac{1}{\mathcal{S}} \quad \mathcal{C}_2 = Accuracy(\mathcal{F})$$

where \mathcal{S} is the cardinality of \mathcal{F} and \mathcal{F} is the subset of features selected, and Accuracy is classification accuracy on the test set. Reducing the number of features ensures that there are no redundant or irrelevant features in the dataset. Classification accuracy is measured on the test set using an SVM.

Instead of constant Crossover and mutation rates, linear crossover and mutation rates have been used in this study. This ensures a high initial mutation rate preventing premature convergence and a low mutation rate when MOGA is close to the Pareto front. Similarly, the crossover rate is initially low to maintain diversity and gradually increases. Figure 8 shows the plot of the crossover and mutation rates against generations for the MOGA.

The summary of GA Parameters is given in Table 3. For evaluation, an average of 100 runs has been considered. The run summary of the MOGA based selector

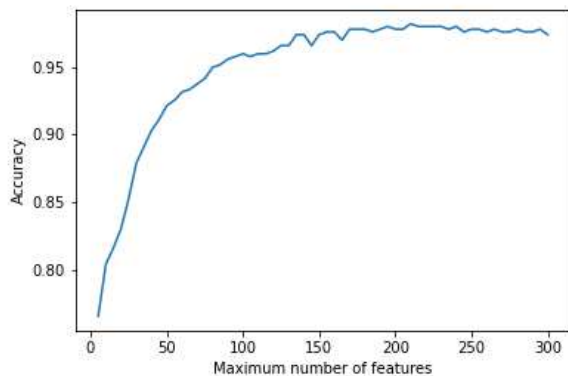


Figure 7: The plot of highest accuracy vs the maximum number of features)

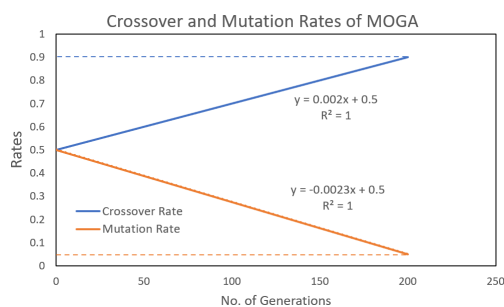


Figure 8: Crossover and Mutation Rates of MOGA vs Generations Plot

showing the min., max., avg., and std. dev. of the number features and highest accuracy for the given generation (using SVM as a classifier) is shown in Table 2. The plot of highest accuracy vs. No of features selected by MOGA is shown in Figure 7

4.3 Feature Classifier

An Ensemble of Support Vector Machines (SVM) is used to classify the selected features. The bagging technique is used to construct the SVM ensemble. For classification, the dataset is randomly divided into ten parts, and the individual SVMs are trained independently (bootstrap techniques). These individual models are then aggregated by the deterministic averaging process to make a joint decision. Each SVM has an RBF kernel with C and Gamma tuned values using the Genetic Algorithm-based Hyperparameter Optimizer. The classifier's performance, evaluated using the test set, and the number of features is stated in Table 8.

5 Data and Validation

5.1 Dataset

The images of CT Scans used in this study are taken from the public database of COVID-19 CT Scans by the name of "SARS-COV-2 Ct-Scan Dataset" published

and maintained by Soares et Al. [29] The dataset consists of 2482 images of chest CT Scans, out of which 1252 are from patients infected with COVID-19. The remaining 1230 images are from patients of other non-COVID pulmonary Diseases. The presence of other non-COVID respiratory diseases allows the model to learn COVID specific features.

The patients considered in the compilation of the dataset mentioned above are from various hospitals in Sao Paulo, Brazil. The COVID-19 CT Scan images are collected from 60 patients (32 males and 28 females). The non-COVID CT Scan images were also collected from 60 patients (30 males and 30 females).

The dataset has been split into three sets, namely training (0.6), validation (0.2), and testing (0.2). The splitting is random, and an average of 5 splits is stated for all evaluations. The summary of the dataset after splitting is stated in Table 4.

5.2 Evaluation Metrics

The screening performance of the model was assessed by Accuracy (ACC), Precision (PRE), Area under ROC Curve (AUC), Recall/Sensitivity (REC), and F1 Score (F1). Precision is the number of true positives over total positive predictions. Recall is defined as the number of true positives over the number of correct classifications. F1 score is simply the harmonic mean of precision and sensitivity of the model. AUC is the total area contained under a ROC Curve, and it shows the usefulness of tests on the model.

5.3 Experimentation

5.3.1 Comparison of AE Depths

Depth of any Neural Network directly affects its performance, and an optimal depth ensures an accurate and robust model. The reconstruction Structural Similarity Index (SSIM) and Mean Squared Error (MSE) has been used to compare various autoencoders. Three different autoencoders have been considered for this with 2, 3, and 4 convolution layers, respectively, in the encoder. The exact structure of the autoencoders is given below:

- **2-Layers:** Two Convolution layers of kernel 3x3 with 32 and 64 filters, respectively. Each layer is followed by a max-pooling layer of 2x2.
- **3-Layers (Proposed) :** Three Convolution layers of kernel 3x3 with 16, 32, and 64 filters, respectively. Each layer is followed by a max-pooling layer of 2x2.
- **4-Layers:** Four Convolution layers of kernel 3x3 with 8, 16, 32, and 64 filters, respectively. Each layer is followed by a max-pooling layer of 2x2.

The analysis is summarized in Table 5. The AE has been trained on the train set and tested on the validation set for this analysis. The size of images used is 128x128, and the pixel values have been scaled to lie between 0 and 1.

5.3.2 Effect of Bagging Estimators on Performance

Bagging Ensemble uses several estimators instead of a single estimator for prediction. This improves performance since a single estimator may have high test

error, but it is overcome by using many small estimators. A different number of estimators are compared based on their accuracy on the validation set, and the box plot of the accuracy vs. the number of estimators is shown in Figure 9. It can be seen that the accuracy improves till 20 estimators, then it saturates.

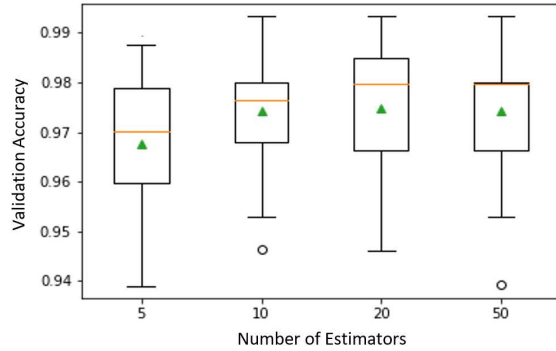


Figure 9: Box Plot of Validation Set Accuracy vs Number of Bagging Estimators

5.3.3 Comparing Different Population Sizes

Optimal population size is obtained by applying the proposed MOGA based selector on the validation set. For obtaining the accuracy values, multiple runs were conducted, and an average of these was recorded. The graphs show the accuracy against the population size of MOGA, which is varied between 50 to 300 in increments of 50. The plot shows that the performance improves up to size 200, after which it stabilizes. Figure 10 shows the plot of the accuracy vs. population size.

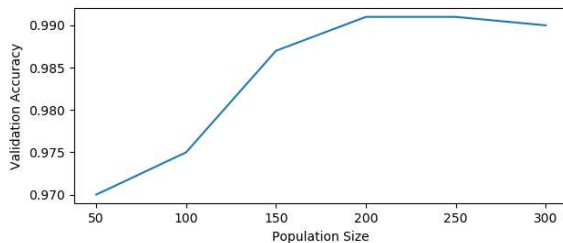


Figure 10: The plot of Population size vs validation accuracy for MOGA Selector

5.3.4 Comparing Generation size of MOGA

Improvement of Pareto fronts with generation is studied in this section. The fronts are plotted using 5 points from each generation, with the parameters for MOGA

being as stated in Table 3. Y-axis represents the selected subset's accuracy on the validation set, while the X-axis represents the inverse of the number of features selected. It can be seen that the fronts improve till 150 generations and then the front stabilizes. This is also observed in the overlap of the fronts in generation 150 and 200. Figure 11 shows the generation wise Pareto fronts.

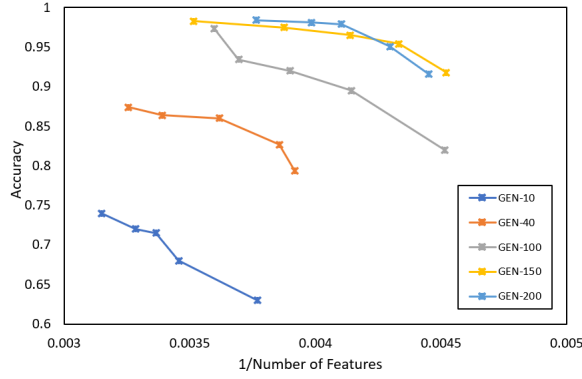


Figure 11: The plot of various Pareto Fronts w.r.t. Generations, for the proposed MOGA. Five points are chosen from each generation for plotting this graph

5.3.5 Comparison with Simple GA, PCA and No-Selector

This section compares the MOGA based selector, PCA, and Simple GA. Accuracy on the validation set is taken as the comparison metric. PCA, a popular dimensionality reduction technique, is applied with a variance set to 0.95. Simple GA tries to find the optimal feature set using validation accuracy as the fitness function. Direct classification with all the extracted features without any feature selection has also been performed. The results obtained are summarized in Table 6. Directly using the features without selection results in poor performance of the model. The proposed model outperforms all the techniques in terms of accuracy. In terms of the number of features, it can be seen that MOGA selects considerably fewer features than simple GA.

5.3.6 Comparison of Crossover and Mutation Rates

Crossover and mutation rates are the parameters that control the convergence of the MOGA selector. A non-constant linear crossover rate has been used in this study to improve the selector's ability to find the optimal front. The proposed selector is compared with constant crossover and mutation rate based MOGA. Accuracy on the validation set averaged over multiple runs is used for this comparison. The result of the analysis is summarized in Table 7.

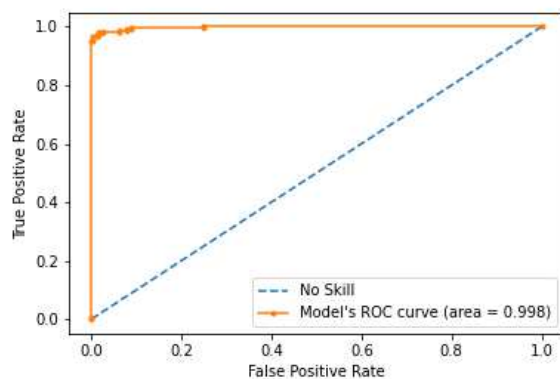


Figure 12: ROC characteristics curve for the proposed methodology (Convolutional Autoencoder + MOGA + Bagging Ensemble with SVM)

5.3.7 Comparison of Feature Selectors

Multiple feature selection techniques, namely Multi-Objective Particle Swarm Optimization (MOPSO) [30], Multi-Objective Differential Evolution (MODE) [18] and MOGA are compared in this study. The standard implementations of these techniques (except the proposed method) are used for this analysis. Table 8 shows the evaluation results on different selectors. For evaluation, features are extracted using the proposed AE architecture, selected using different selectors, and finally classified using SVM Ensemble. For more effective comparison, the test set, which is unseen by the selectors, is used for the evaluation. The details of the train-test split are provided in section 4-A. The results obtained show that the proposed model outperforms other multi-objective feature selection techniques. Figure 13 shows the confusion matrices obtained for different feature selectors on the test set.

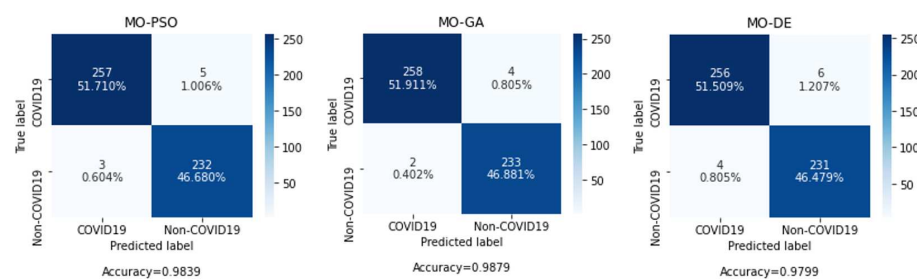


Figure 13: Confusion matrices of the proposed methodology with different multi objective feature selectors on the Test set

6 Results and Analysis

For evaluation, the dataset is split according to Table 4. The proposed method has been evaluated using the test set, composed of 260 COVID-19 chest CT images and 237 non-COVID chest CT Images. The performance is measured based on the evaluation metrics discussed in Section-IV-B. The features are extracted using the AE encoder defined in III-A and selected using MOGA as described in III B. Finally, the features are classified using a Bagging Ensemble of SVM classifiers, described in III C.

The proposed methodology is implemented on python software, running on a CPU. The system architecture uses an Intel Core i7 processor with a 4 GB graphic card, running at 1.80 GHz, a 64-bit operating system, and 16 GB RAM.

The proposed architecture achieves an Accuracy(ACC) of 98.79%, Precision(PRE) of 98.47%, Sensitivity(SEN) of 99.23%, F1 Score(F1) of 98.85%, Specificity(SPE) of 98.31%, Net Positive Rate(NPV) of 99.14% and Area under ROC Curve(AUC) of 99.8%. The prediction time on the system used is 127 ms per image. The proposed model outperforms the current state of the art COVID-19 diagnostic techniques in terms of speed and accuracy.

The Receiver Operator Characteristic Curve on the proposed model's test set is depicted in Figure 12. The area obtained under the ROC Curve (AUC) is 0.998. A high value of AUC shows the robustness of the proposed model. Table 8 summarizes the evaluation results of the proposed architecture. Confusion matrices for different feature selectors on the test set are shown in Figure 13.

The proposed study (MOGA) outperforms other multi-objective feature selectors. A decrease in ACC is expected in GA with an increase in the number of variables. As the number of variables for optimizing the selection are less, MOGA outperforms MODE and MOPSO.

7 Conclusion and Future Works

An unsupervised learning-based approach is proposed for feature generation because of the higher feature diversity obtained from such an approach. Various Evolutionary and non Evolutionary feature selectors are compared in this study, and finally, a MOGA based selector is proposed. An Ensemble of SVMs is used for the final classification. The bagging technique is used in the ensemble as it works well with complex feature maps.

The model achieves better results than state of the art techniques for all performance metrics. With such high-performance results and a little prediction time compared to Physical RT-PCR tests, the proposed model can be an effective and efficient COVID-19 Chest CT Scan screening Technique. The study further finds many insights in feature extraction, feature selection, and classification, which are listed below.

- Unsupervised learning-based feature extractors can provide detailed and accurate feature maps for medical image classification.
- Evolutionary Feature Selectors remove data redundancy better than standard techniques like PCA in terms of accuracy and number of features. Not using a feature selector results in inferior performance
- Optimizing the number of features and accuracy forces the model to learn from a smaller feature set, resulting in a more robust model since only the most productive features are retained.

- MOGA outperforms MOPSA and MODE in medical image classification because of the large number of parameters that need to be optimized for MOPSA and MODE.
- Variable Crossover and Mutation rates for MOGA can significantly improve performance in medical image classification.
- Bagging improves a classifier’s performance, as a large number of classifiers produce a lower test error than a single classifier. This is because diversity compensates for bias.

An efficient and effective system for COVID-19 diagnostic using Chest CT Scan images has been proposed in this work. Future work will be to localize the infection regions, map them in the images, and track the degree of infection.

Compliance with ethical standards

Conflict of interest The authors declare no conflict of interest.

Ethical approval All procedures performed in studies involving human participants were in accordance with the ethical standards of the institutional and/or national research committee and with the 1964 Helsinki declaration and its later amendments or comparable ethical standards.

Informed consent Informed consent was obtained from all individual participants included in the study.

References

1. WebMD, 2020. Last accessed date 18th June 2020.
2. WHO, 2020. Last accessed date 18th June 2020.
3. Worldometer, 2020. Last accessed date 18th June 2020.
4. Alireza Tahamtana and Abdollah Ardebili. Real-time rt-pcr in covid-19 detection: issues affecting the results. *Expert review of molecular diagnostics*, 20:453-454, 2020.
5. Tao Ai, Zhenlu Yang, Hongyan Hou, Chenao Zhan, Chong Chen, and et al. Correlation of chest ct and rt-pcr testing in coronavirus disease 2019 (covid-19) in china: A report of 1014 cases. *Radiology*, 2020.
6. Zheng Ye, Yun Zhang, Yi Wang, Zixiang Huang, and Bin Song. Chest ct manifestations of new coronavirus disease 2019(covid-19): a pictorial review. *European Radiology*, 2020.
7. Jeffrey P Kanne, Brent P Little, Jonathan H Chung, Brett M Elicker, and Loren H Ketai. Essentials for radiologists on covid-19: an update- radiology scientific expert panel. *Radiology*, 2020.
8. Ophir Gozes, Maayan Frid-Adar, Nimrod Sagie, Huangqi Zhang, Wenbin Ji, and Hayit Greenspan. Coronavirus detection and analysis on chest ct with deep learning, 2020.
9. Feng Shi, Liming Xia, Fei Shan, Dijia Wu, Ying Wei, Huan Yuan, Huiting Jiang, Yaozong Gao, He Sui, and Dinggang Shen. Large-scale screening of covid-19 from community acquired pneumonia using infection size-aware classification, 2020.
10. Xiaowei Xu, Xiangao Jiang, Chunlian Ma, Peng Du, Xukun Li, Shuangzhi Lv, Liang Yu, Yanfei Chen, Junwei Su, Guanqing Lang, Yongtao Li, Hong Zhao, Kaijin Xu, Lingxiang Ruan, and Wei Wu. Deep learning system to screen coronavirus disease 2019 pneumonia, 2020.

11. Arpan Mangal, Surya Kalia, Harish Rajgopal, Krithika Rangarajan, Vinay Namboodiri, Subhashis Banerjee, and Chetan Arora. Covidaid: Covid-19 detection using chest x-ray, 2020.
12. Yu-Huan Wu, Shang-Hua Gao, Jie Mei, Jun Xu, Deng-Ping Fan, Chao-Wei Zhao, and Ming-Ming Cheng. Jcs: An explainable covid-19 diagnosis system by joint classification and segmentation, 2020.
13. Sanhita Basu, Sushmita Mitra, and Nilanjan Saha. Deep learning for screening covid-19 using chest x-ray images, 2020.
14. Mohamed Loey, Florentin Smarandache, and Nour Eldeen M. Khalifa. A deep transfer learning model with classical data augmentation and cgan to detect covid-19 from chest ct radiography digital images. *Preprints*, 2020.
15. Ophir Gozes, Maayan Frid-Adar, Hayit Greenspan, Patrick D. Browning, Huangqi Zhang, and et al. Rapid ai development cycle for the coronavirus (covid-19) pandemic: Initial results for automated detection patient monitoring using deep learning ct image analysis. 2020.
16. Aryan Mobiny, Pietro Antonio Cicalese, Samira Zare, Pengyu Yuan, Moham-madsajad Abavisani, Carol C. Wu, Jitesh Ahuja, Patricia M. de Groot, and Hien Van Nguyen. Radiologist-level covid-19 detection using ct scans with detail-oriented capsule networks, 2020.
17. Dilbag Singh, Vijay Kumar, Vaishali, , and Manjit Kaur. Classification of covid-19 patients from chest ct images using multi-objective differential evolution-based convolutional neural networks. *European Journal of Clinical Microbiology Infectious Diseases*, 2020.
18. Bing Xue, Wenlong Fu, and Mengjie Zhang. Multi-objective feature selection in classification: A differential evolution approach. In Grant Dick, Will N. Browne, Peter Whigham, Mengjie Zhang, Lam Thu Bui, Hisao Ishibuchi, Yaochu Jin, Xiaodong Li, Yuhui Shi, Pramod Singh, Kay Chen Tan, and Ke Tang, editors, *Simulated Evolution and Learning*, pages 516–528, Cham, 2014. Springer International Publishing.
19. Oluleye Hezekiah Babatunde, Leisa Armstrong, Jinsong Leng, and Dean Diepeveen. A genetic algorithm-based feature selection. *British Journal of Mathematics Computer Science*, 2014.
20. A. Khan and A.R. Baig. Multi-objective feature subset selection using non-dominated sorting genetic algorithm. *Journal of Applied Research and Technology*, 13(1):145 – 159, 2015.
21. A. Ferriyan, A. H. Thamrin, K. Takeda, and J. Murai. Feature selection using genetic algorithm to improve classification in network intrusion detection system. In *2017 International Electronics Symposium on Knowledge Creation and Intelligent Computing (IES-KCIC)*, pages 46–49, 2017.
22. S. Sindhiya and S. Gunasundari. A survey on genetic algorithm based feature selection for disease diagnosis system. In *Proceedings of IEEE International Conference on Computer Communication and Systems ICCCS14*, pages 164–169, 2014.
23. Dana H. Ballard. Modular learning in neural networks. In *Proceedings of the Sixth National Conference on Artificial Intelligence - Volume 1, AAAI'87*, page 279–284. AAAI Press, 1987.
24. Pierre Baldi. Autoencoders, unsupervised learning and deep architectures. In *Proceedings of the 2011 International Conference on Unsupervised and Transfer Learning Workshop - Volume 27, UTLW'11*, page 37–50. JMLR.org, 2011.

25. B. Hou and R. Yan. Convolutional auto-encoder based deep feature learning for finger-vein verification. In *2018 IEEE International Symposium on Medical Measurements and Applications (MeMeA)*, pages 1–5, 2018.
26. K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, 2002.
27. D. Opitz and R. Maclin. Popular ensemble methods: An empirical study. *Journal of Artificial Intelligence Research*, 11:169–198, Aug 1999.
28. Leo Breiman. Bagging predictors. *Machine Learning*, pages 123–140, 1996.
29. Eduardo Soares, Plamen Angelov, Sarah Biaso, Michele Higa Froes, and Daniel Kanda Abe. Sars-cov-2 ct-scan dataset: A large dataset of real patients ct scans for sars-cov-2 identification. *medRxiv*, 2020.
30. C. A. Coello Coello and M. S. Lechuga. Mopso: a proposal for multiple objective particle swarm optimization. In *Proceedings of the 2002 Congress on Evolutionary Computation. CEC'02 (Cat. No.02TH8600)*, volume 2, pages 1051–1056 vol.2, 2002.

Table 1: Related Work results analysis on COVID-19 screening

Ref	Technique	Key Findings
[9]	infection Size Aware Random Forest method (iSARF)	The accuracy of 87.9%, a sensitivity of 90.7%, and a specificity of 83.3% are achieved on chest CT scan.
[10]	ResNet-18 (CNN model)	The performance parameters are: • specificity: 92.2%, • sensitivity: 98.2% and, • AUC: 0.996.
[11]	Pre-trained CheXNet and DenseNet	An accuracy of 90.5%, a sensitivity of 100% is achieved using 5323 (COVID19-115, normal - 1341, and pneumonia-3867) Chest XRay images.
[12]	Joint Classification and Segmentation (JCS)	Used a dataset of 400 COVID-19 patients (144,167 images) and 350 Non-COVID patients. The model achieves a dice score of 78.3%, sensitivity of 95% and a specificity of 93% for the segmentation task.
[13]	Domain Exten. Transfer Learning (DETL) with Gradient Class Activation Map (Grad-CAM)	The Data A - Binary classes disease (13 diseases) and normal. Data B - Four classes (normal, pneumonia, other diseases, and Covid19). An accuracy of 95.3% using X-ray scans.
[14]	AlexNet, VGG16, VGG19, GoogleNet, and ResNet50	Pre-trained models used to train CNN on 742 chest CT scans for two binary classes (COVID and non-COVID). The highest accuracy of 82.91% is achieved with the ResNet50 pre-trained CNN model.
[15]	3-Dimensional deep learning	The specificity of 92.2%, a sensitivity of 98.2%, and AUC of 0.996 is achieved by the 3-D CNN model.
[16]	Detail-Oriented Capsule Nets + Peekaboo (patch crop and drop strategy)	A recall of 91.5%, accuracy of 87.6%, precision of 84.3%, and AUC of 96.1 is achieved on chest CT scan dataset for classification to binary classes(COVID-19 and Non-COVID).
[17]	Multi-Objective Differential Evolution (MODE) deep learning	The performance parameters of MODE outperforms by 1.927% of Kappa statistics, 1.68% of specificity, 1.82% of sensitivity and, 2.09% of F-measure in comparison to authentic CNN models.

Table 2: Summary of the run for MOGA selector

GENERATIONS	NUMBER OF EVALUATIONS	NUMBER OF FEATURES				ACCURACY
		Min	Max	Avg	Std. Dev.	
10	147	265	308	287.45	7.71	0.65
20	126	263	305	284.3	6.21	0.79
30	142	259	302	279.64	7.34	0.83
40	112	257	299	276.34	5.66	0.86
50	139	253	294	273.3	7.27	0.88
60	154	245	291	268.18	5.83	0.89
70	128	242	292	264.82	4.59	0.91
80	146	239	291	261.74	6.69	0.92
90	135	234	287	258.84	5.52	0.93
100	128	235	285	256.4	6.34	0.94
110	117	231	281	250.24	5.21	0.94
120	115	228	279	248.32	5.91	0.95
130	128	224	274	245.96	6.47	0.97
140	108	227	275	247.32	8.45	0.96
150	113	226	275	246.33	7.93	0.97
160	123	226	271	243.9	5.24	0.98
170	105	224	272	242.14	4.35	0.97
180	101	224	271	241.66	6.01	0.98
190	107	221	269	240.72	6.81	0.98
200	97	224	273	242.35	4.46	0.99

Table 3: Summary of the hyperparameters used for MOGA

Parameter	Value
Population Size	200
No. of Generations	200
Mutation Probability	linear increase from 0.5 to 0.9
Crossover Probability	linear decrease from 0.5 to 0.1
Selection Procedure	Tournament Based
Mating Pool Size	50

Table 4: The brief details of the dataset for CT scans

Set of Images	Training Set	Validation Set	Test Set	Total
COVID-19	756	236	260	1252
non-COVID	732	261	237	1230
Total	1488	497	497	2482

Table 5: Comparison of AE Depth on the basis of reconstruction

Auto-Encoder	SSIM Score	MSE Score
2-Layer	0.76	0.0014
3-Layer	0.91	0.0004
4-Layer	0.89	0.0007

Table 6: Validation accuracies with MOGA, PCA and GA as feature selector

Feature Selector	Accuracy	No. of Features
MOGA	0.991	244
Simple GA	0.961	316
PCA	0.947	207
No Selector	0.893	2048

Table 7: Validation accuracy of MOGA selector using different crossover and mutation rates

Crossover Rate	Mutation Rate	Accuracy
Linear increase from 0.5 to 0.9	Linear decrease from 0.5 to 0.1	0.991
0.9	0.1	0.973
0.8	0.2	0.977

Table 8: Comparative Assessment of various feature selectors on the test set

Feature Selector	Number of features Selected	Classification Summary				Evaluation Metric						
		TP	TN	FP	FN	ACC	PRE	SEN	F1	SPE	NPV	AOC
Multi Objective Differential Evolution	252	256	231	6	4	0.9798	0.9770	0.9846	0.9808	0.9746	0.9829	0.991
Multi Objective Particle Swarm Optimization	227	256	232	5	4	0.9839	0.9809	0.9884	0.9846	0.9789	0.9872	0.994
Multi Objective Genetic Algorithm (Proposed Study)	244	258	233	4	2	0.9879	0.9847	0.9923	0.9885	0.9831	0.9914	0.998

Figures



Figure 1

Schema of basic Autoencoder



Figure 2

The working of NSGA-II



Figure 3

A general architecture of SVM ensemble with an aggregation step



Figure 4

A general architecture of SVM ensemble with an aggregation step



Figure 5

Flowchart summarizing the proposed architecture



Figure 6

Original and Reconstructed (from Convolution Autoencoder) Chest CT Images from validation set



Figure 7

The plot of highest accuracy vs the maximum number of features)



Figure 8

Crossover and Mutation Rates of MOGA vs Generations Plot



Figure 9

Box Plot of Validation Set Accuracy vs Number of Bagging Estimators



Figure 10

The plot of Population size vs validation accuracy for MOGA Selector



Figure 11

The plot of various Pareto Fronts w.r.t. Generations, for the proposed MOGA. Five points are chosen from each generation for plotting this graph



Figure 12

ROC characteristics curve for the proposed methodology (Convolutional Autoencoder + MOGA + Bagging Ensemble with SVM)



Figure 13

Confusion matrices of the proposed methodology with different multi objective feature selectors on the Test set