Annals of Computing (2025), 1(4):26-35
https: doi.org /10.5281/zenodo.15235961

Research Article

# Deep Learning Study of Pandemic Contact Tracing

**C. O. Uzoigwe[1], J. Akaabo[2], N. V. Blamah[3], G.I.O. Aimufua[4], M. Olalere[5] and C. I. Osuji[6]**

[1,3,4,5]Department of Computer Science, Nasarawa State University Keffi, Nigeria.
[2]Department of Mathematics and Computer Science Benue State University, Makurdi, Nigeria.
[6]Department of Computer Science, Federal University, Wukari, Nigeria.

Received 23 March 2025; Acceptance 15 April 2025; Published 17 April 2025.

## Abstract

The outbreaks of infectious diseases exemplified by the COVID-19 pandemic, have left the global health community with the continuous search for novel methods that can help in emergencies. This is because traditional pandemic contact tracing methods have failed in enhancing accuracy, efficiency, and adaptability within pandemic scenarios. In this study, deep learning study of pandemic contact tracing was investigated. Deep learning models were developed using neural networks, namely, Convolutional Neural Networks (CNNs). Networks dataset of images which capture individuals and their interactions were used as the CNN input data and to analyze the transformed data. Results reveal an accuracy of 97% from the CNN. Finding further reveal that the CNN model in the present study compares favourably with the results of other traditional pandemic contact tracing model such as MobileNetV2, VGG16, and InceptionV3. Based on these results, it was therefore recommended among others, that the CNN model holds significant promise for the future of pandemic contact tracing due its scalability and adaptability in emergency situations. Overall, the adoption of the CNN by health workers, could lead to superior accuracy and efficiency in identifying potential contacts and exposures, a critical aspect in curbing the spread of diseases during pandemics.

**Keywords**: Deep Learning, CNN model, Pandemic, Contact tracing.

## Introduction

As the global health community continues to seek innovative ways of detecting diseases, deep learning methods are gaining attention among health research community [1-2]. This is because traditional contact tracing (TCT) methods have proven to be resource-intensive, time-consuming, error- prone, and difficult to scale; they also present privacy and ethical challenges and are difficult to integrate with existing public health systems [3-4]. Moreover, deep neural networks can process large datasets, such as pandemic cases and contact information, and learn intricate patterns, enhancing the speed and accuracy of contact tracing efforts [5]. This can lead to quicker identification and notification of potential contacts, ultimately aiding in

Correspondence to: Chris Okoro Uzoigwe, e-mail: uzoigwe@gmail.com

breaking the chain of transmission and controlling the spread of the virus more effectively. However, it is essential to consider potential biases in deep learning algorithms [6] and the challenge of making deep learning models smaller, faster, and better [7] for practical implementation in resource-constrained scenarios. This makes it necessary to explore methods for efficient deep learning to optimize the models used for pandemic contact tracing. Furthermore, understanding the strengths and limitations of deep learning models compared to other machine learning techniques, as highlighted in studies on radiology text report classification and disease prediction [2], can aid researchers in making informed choices when designing the contact tracing models. Generally, the application of deep learning in the study can lead to valuable insights and innovations in pandemic contact tracing. The utilization of state-of-the-art deep learning algorithms can enhance the effectiveness of the contact tracing process, facilitate decision-making, and contribute to the ongoing efforts to combat the pandemic [8].

Several research papers have explored the fundamentals of deep learning algorithms and architectures [9]. Some papers surveyed the different techniques and applications of deep learning [1], some investigated deep learning potential in disease prediction [2], soil water erosion susceptibility assessment [10], and risk prediction for pancreatic cancer [11]. The author in [12] observed that deep learning techniques, particularly Convolutional Neural Networks (CNNs), have emerged as promising solutions [12]. This unique ability of CNNs to detect intricate patterns within complex datasets aligns well with the subtleties of contact tracing data, improving accuracy and potentially addressing privacy concerns through enhanced data anonymization [13]. To this end, deep learning study of pandemic contact tracing is presented.

## Research Method

### Dataset Preparation Data Source

The dataset used in this study consists of a total of 49,069 instances, each representing a specific observation or data point [14]. Overall, 10 features were provided in the dataset with each representing different characteristic of pandemic cases.

### Data Pre-processing

In the data pre-processing step, the collected data undergoes several transformations to prepare it for training and evaluation. One crucial aspect of this pre-processing is handling missing or incomplete data, removing duplicates, and resolving inconsistencies. This step also allows for the combination of different parts of the dataset by stacking the rows of a dataset on top of another to give a single comma- separated values (CSV) file. This gives a suitable format for training and testing with a split ratio of 80:20. Specifically,

80% allocated for training set and 20% for testing set. This enables the model to have sufficient data to learn from in the training process and at the same time, giving room for independent validation of the testing process. Overall, the standardized data allows for uniform scaling across various features and helps model convergence. Data cleaning techniques are applied to enhance data integrity and prepare it for deep learning analysis.

**Deep Learning Model**

Deep learning models were developed using neural networks, namely, Convolutional Neural Networks (CNNs) to analyze the transformed data. The model is trained using labeled data, where known pandemic cases and their contacts are used to learn patterns and correlations. The CNN classifier has different layers as detailed thus:

**Convolutional Layer**

The convolutional layer is the basic layer of the CNN. This is accountable for determining the design characteristics. The input picture is passed through a filter in this layer. The function map is obtained from the output of the same filters by convolution operation. The multiplication of sets of weights with the input is performed by a convolution operation. A filter consists of a two-dimensional collection of weights multiplied by an array of input data. A dot product is a type of multiplication that is applied between a filter-sized patch of the input and the filter, which results in a single value. This product is applied between the filter-sized patch of the input and the filter. The filter is smaller than the input, and the same filter is used to multiply the input from different points. The filter is designed as a special technique to identify specific types of features as it systematically covers the entire image.

Assume that the NN input is V ∈ RAxB, where A denotes the number of features that indicate an input frequency band and B denotes the total number of input frequency bands. The size of the filter bank function vector is represented by B in the case of filter bank features. Assume that v = [v1 v2 . . . vB], where vB denotes the function vector for band b. The activations of the convolution layer can be calculated as

$$h_{j,k} = \theta\left(\sum_{b=1}^{s} w_{b,j}^{T} v_{b+k-1} + a_j\right), \quad (1)$$

where hj, k is the jth feature map's convolution layer output of the convolution layer band of kth, s indicates the filter scale, wb,j indicates the weight vector for the jth filter's b th band, aj is the jth feature map's bias, and (x) represents the activation function.

**Pooling Layer**

The pooling layer summarizes the presence of features by facilitating the down sampling features. It is normally applied after a convolution layer and has some spatial in variance. Two popular pooling methods, average pooling and max pooling, summarize the average presence of a function, and the most activated presence of a function. In fact, the pooling layer deletes the unnecessary features from the images and makes the image literate. In average pooling, the layer averages the value of its current view every time. When using max-pooling, the layer selects the maximum value from the filter's current view each time. By using the matrix size specified in each feature map, the max-pooling technique selects only the maximum value, resulting in reduced output neurons. Thus, the size of the image becomes very small, but the scenario remains the same. A pooling layer is important for reducing the number of feature maps and network parameters, and a dropout layer is used to prevent overfitting. The activation of max-pooling can be calculated as follows:

$$P_{j,m} = max_{k=1}^{r}\left(h_{j,(m-1)(n+k)}\right), \tag{2}$$

The *j*th function map's pooling layer and the *m*th pooling layer band's performance are denoted by pj,m. Where *n* is the subsampling factor and *r* is the pooling scale, these three variables determine how many bands will be combined.

**Flattened Layer**

The flattened layer is used to convert data from the matrix into a one-dimensional array for use in the fully connected layer and to create a single one-dimensional feature that is both long and narrow. Flattening vectors are an option. Finally, it connects the single vector to the final classification model, which is also known as a fully connected layer. All pixel data are given in one and connect with fully connected layers. Flattening and fully connected layers are the last few steps of the CNN. It is prepared for the next fully linked layer of picture categorization by converting it into a one- dimensional array.

**Fully Connected Layer**

CNNs rely mostly on fully connected layers, which have proven to be quite useful in computer vision image

recognition and classification. Convolution and pooling are the starting levels of the CNN process, which breaks down the image into attributes and analyses them separately. In a fully connected layer, each input is connected to all neurons, and the inputs are flattened. The ReLU activation function is commonly used as a fully connected layer. The SoftMax activation function was used to predict the output images in the last layer of the fully connected layer. The convolutional neural network architecture uses a fully connected layer.

These are the last few layers and important layers of the convolutional neural network.

**Pretrained Models**

The scarcity of medical data or datasets is one of the greatest challenges for researchers in medical-related research, and data are one of the most crucial components of deep learning approaches. Data analysis and labelling are both costly and time consuming. Transfer learning provides the advantage of avoiding the requirement for large datasets. The calculations become lower and less costly. Transfer learning is a method in which the pretrained model, which is trained on a large dataset, is transferred to the new model that needs to be trained, including new data that are relatively smaller than required. For a certain task, this process initialized the training of the CNN with a small dataset, including a large- scale dataset that was already trained in the pretrained models.

Tree CNN-based pretrained models were used to classify CXR images in this investigation. These applied models are MobileNet_V2, VGG16, and InceptionV3. The CXR images were of two classes. One is normal, and the other is a SARS- CoV-2-infected patient. This study also used a transfer learning method, which can perform with inadequate data by using ImageNet data, and it is also efficient in training time.

MobileNetV2 improves the cutting-edge performance of versatile models on numerous assignments and seat stamps across a range of model sizes. In every line of MobileNetV2, it works as a sequence of n repeated layers. Depth wise separable is used in MobileNet, which factorizes the normal form into depth wise convolution. This implies a depth of 1 x1, which is also known as a pointwise convolution.

InceptionV3 is another pretrained model that was used. It normally has a maximum number of pooling layers. VGG16 is also quite helpful because it can extract features at low levels with the help of a small kernel. For CXR images, a small-sized kernel can efficiently extract the features.

Because of the insufficient dataset, this study used VGG16 with appropriate layer addition for the final result.

**Algorithm for the Formulated Model**

The systems plotted confusion matrix, with columns representing real values and rows rep- resenting the predicted values. In a classification model, the summary of the prediction results is known as the confusion matrix. In the confusion matrix, correct and incorrect predictions are summed and split down by class, and for n x n matrices FP, FN, TP, and TN are calculated using the following equations:

$$TP_i = a_{ii} \tag{3}$$

$$FP_i = \sum_{j=1, j \neq i}^{n} a_{ji}, \tag{4}$$

$$FN_i = \sum_{j=1, j \neq i}^{n} a_{ij}, \tag{5}$$

$$TN_i = \sum_{j=1, j \neq i}^{n} \sum_{k=1, k \neq 1}^{n} a_{jk}, \tag{6}$$

Three terms are important in error analysis. These are predictions, data, and features. Prediction-based error analysis can be performed using a confusion matrix, where it can be visualized by the percentage of true positives, true negatives, false positives, and false negatives. Data size and nature are also important for the error analysis. Splitting the data accordingly for making trains and tests is also considerable for error analysis because the training and test sets may affect the result on a large scale. Features play a vital role in error analysis. Feature engineering and regularization were also performed to reduce errors.

**Performance Evaluation of the Model Using Standard Metric**

The performance analysis of the models is evaluated based on accuracy, precision, recall, and F1-score. The performance of the proposed model was assessed using the terms true positive (TP), false positive (FP), true negative (TN), and false negative (FN). The rate of properly detecting the affected photographs from all images is referred to as recall, also called sensitivity. Precision is the opposite of recall. The F1-score is a combined measure of precision and recall, which shows how often the predicted value is accurate.

It is also known as the harmonic mean of p and r in mathematics. These equations are given below.

Matrixes can be used to evaluate a system's performance, and after the development of the model, its performance.

Accuracy is a measure of how well a model or system works (i.e., the number of times the model correctly predicted the actual outcome) and should be calculated. The mathematical formulas for determining the accuracy are expressed in the following equations:

$$accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{7}$$

$$accuracy = \frac{correct\ predictions}{total\ number\ of\ examples} \tag{8}$$

The rate of successfully detecting the real value from a set of all values is recognized as recall, also called sensitivity.

Recall can be determined using the following expression:

$$recall = \frac{TP}{TP+FN} \tag{9}$$

The number of correct identifications is referred to as precision. The number of times the model's positive forecast was right can be calculated, and this is more related to the with the model's positive identification, using the following mathematical formula:

$$precision = \frac{TP}{TP+FP} \tag{10}$$

For both recall and precision, a single matrix can be used to summarize the classifier's performance, and the F1-score is a single matrix that characterizes precision and recall. It is also known as a harmonic means of precision and recall in other mathematical words. The F1-score is calculated using the following equation:

$$F_1 - score = \frac{2pr}{p+r} \tag{11}$$

In equations (3) and (4), TP stands for true positive, FP for false positive, and FN for false negative. The letters p and r in equation (11) represent precision and recall, respectively.

## Results and Discussion

The detailed results of the CNN approach are presented.

### Accuracy (ACC)

The accuracy of the CNN model is 97% on the test dataset, underscoring its capability to correctly classify samples.

### Precision (PREC)

With a precision of 100%, the CNN model demonstrates its proficiency in minimizing false positives, crucial for ensuring the reliability of pandemic contact tracing.

### Recall (REC)

The hybrid model exhibits a recall of 94%, indicating its effectiveness in capturing true positives, an important part in pandemic contact tracing.

**F1 Score**

The CNN model achieves an F1 score of 97%, striking a balance between precision and recall. This offers a robust performance across different aspects of the evaluation.

**Table 1. Comparison of the present CNN model with other the accuracy and loss of other models**

| Model | Accuracy (%) | Validation accuracy (%) | Loss (%) | Validation loss (%) |
|---|---|---|---|---|
| Custom CNN | 97 | 97 | 6 | 8 |
| Modified MobileNetV2 | 98 | 97 | 5 | 6 |
| VGG16 | 98 | 98 | 4 | 6 |
| ResNet50 | 88 | 91 | 29 | 21 |
| InceptionV3 | 98 | 98 | 5 | 5 |

**Table 2. Model Evaluation**

| Model | State | Precision | Recall | F1 score |
|---|---|---|---|---|
| Custom CNN | COVID-19 | 1.000 | 0.94 | 0.97 |
| Custom CNN | Normal | 0.95 | 1.00 | 0.97 |
| MobileNetV2 | COVID-19 | 0.99 | 0.96 | 0.98 |
| MobileNetV2 | Normal | 0.97 | 0.99 | 0.98 |
| VGG16 | COVID-19 | 1.00 | 0.95 | 0.97 |
| VGG16 | Normal | 0.96 | 1.00 | 0.98 |
| InceptionV3 | COVID-19 | 1.00 | 0.98 | 0.99 |
| InceptionV3 | Normal | 0.98 | 1.00 | 0.99 |

Tables 1 and 2 give the model evaluation of precision, recall, and F1-score of the present CNN model, and other pandemic contact tracing models, namely MobileNetV2, VGG16, and InceptionV3 as shown in Table 2. From these results, it can be inferred that InceptionV3 and MobileNetV2 have higher precision, recall, and F1-score values than the other models. InceptionV3 performs exceptionally well among other models, and its accuracy also yielded higher results, which are shown in Table 1. Nonetheless, robustness of the CNN model and the accuracy of 97 % makes it a suitable tool for emergency cases like pandemic contact tracing. The CNN model is generally preferred over traditional deep learning methods due to its unparalleled ability in recognizing intricate patterns within contact tracing data. This unique selection enhances the system's precision in identifying potential contacts and exposures.

## Conclusion

The study concludes that CNN- based contact tracing system significantly improved the accuracy and efficiency of identifying and managing pandemic contacts in the urban area, demonstrating its potential for widespread application in the future pandemic scenarios. In comparison with traditional pandemic contact tracing models, results reveal that the CNN model in the present study exhibits an accuracy of 97%. This makes the present model a suitable alternative to the traditional model, especially when robust and timely evaluation is needed. Based on these results, it was therefore recommended among others that the adaptable nature of CNN ensures its applicability across different scenarios, preparing organizations for diverse outbreak patterns. Additionally, by incorporating this advanced technology, the CNN model offers superior accuracy and efficiency in identifying potential contacts and exposures, a critical aspect in curbing the spread of diseases during pandemics [15].

## References

1. Pouyanfar, S., Sadiq, S., Yan, Y., Tian, H., Tao, Y., Reyes, M.P., ... & Iyengar, S.S. (2018). A survey on deep learning: Algorithms, techniques, and applications. ACM Computing Surveys (CSUR), 51(5), 1-36.
2. Yu, Z., Wang, K., Wan, Z., Xie, S., & Lv, Z. (2023). Popular deep learning algorithms for disease prediction: a review. Cluster Computing, 26(2), 1231- 1251.
3. Yi, F., Xie, Y. & Jamieson, K. (2022). Cellular-assisted, deep learning based PANDEMICcontact tracing. Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies, 6(3), 1-27.
4. Tang, J. Vinayavekhin, S., Weeramongkolkul, M., Suksanon, C., Pattarapremcharoen, K., Thiwathittayanuphap, S. & Leelawat, N. (2022). Agent- based simulation and modeling of PANDEMICpandemic: a Bibliometric analysis. Journal of Disaster Research, 17(1), 93-102.
5. Yan, W.Q. (2023). Computational Methods for Deep Learning: Theory, Algorithms, and Implementations (2nd).
6. Vardi, G. (2023). On the implicit bias in deep-learning algorithms. Communications of the ACM, 66(6), 86- 93.
7. Menghani. G. (2023). Efficient deep learning: A survey on making deep learning models smaller, faster, and better. ACM Computing Surveys, 55(12), 1-37.

8. Hang, C. N., Tsai, Y. Z., Yu, P. D., Chen, J., & Tan, C. W. (2023). Privacy-Enhancing Digital Contact Tracing with Machine Learning for Pandemic Response: A Comprehensive Review. Big Data and Cognitive Computing, 7(2), 108.
9. Shrestha, A., & Mahmood, A. (2019). Review of deep learning algorithms and architectures. IEEE access, 7, 53040-53065.
10. Khosravi, K., Rezaie, F., Cooper, J. R., Kalantari, Z., Abolfathi, S., & Hatamiafkoueieh, J. (2023). Soil water erosion susceptibility assessment using deep learning algorithms. Journal of Hydrology, 618, 129229.
11. Placido, D., Yuan, B., Hjaltelin, J. X., Zheng, C., Haue, A. D., Chmura, P. J., ... & Sander, C. (2023). A deep learning algorithm to predict risk of pancreatic cancer from disease trajectories. Nature Medicine, 1- 10.
12. Adetunji, C. O., Olaniyan, O. T., Adeyomoye, O., Dare, A., Adeniyi, M. J., Alex, E., ... & Shariati, M. A. (2022). Machine learning approaches for PANDEMICpandemic. Assessing PANDEMIC and Other Pandemics and Epidemics using Computational Modelling and Data Analysis, Sensors vol 1,133-143.
13. Siddiq, M. (2023). Exploring the Role of Machine Learning in Contact Tracing for Public Health: Benefits, Challenges, and Ethical Considerations. American Journal of Economic and Management Business (AJEMB), 2(3), 99-110.
14. Kaggle (2022). (https://www.kaggle.com/datasets/imdevskp/corona-virus report?select=covid_19_clean_complete.csv)
15. Singh, P. K., Nandi, S., Ghafoor, K. Z., Ghosh, U., & Rawat, D. B. (2020). Preventing PANDEMIC spread using information and communication technology. IEEE Consumer Electronics Magazine, 10(4), 18-27