BCovX: Blockchain-based COVID Diagnosis Scheme using Chest X-Ray for Isolated Location

Arpit Shukla*, Urvashi Ramdasani†, Gunjan Vinzuda‡, Mohammad S. Obaidat, *Fellow of IEEE and Fellow of SCS*§, Sudeep Tanwar¶, Neeraj Kumar‖
*†‡¶Department of Computer Science and Engineering, Institute of Technology, Nirma University, Ahmedabad, India
§College of Computing and Informatics, University of Sharjah, UAE, King Abdullah II School of IT, University of Jordan, Jordan, and University of Science and Technology Beijing, China
‖Thapar Institute of Engineering & Technology (Deemed to be University), Patiala (Punjab), India
Email: *18bce370@nirmauni.ac.in, †18bce247@nirmauni.ac.in, ‡18bce257@nirmauni.ac.in, §m.s.obaidat@ieee.org, ¶sudeep.tanwar@nirmauni.ac.in, ‖neeraj.kumar@thapar.edu

ICC 2021 - IEEE International Conference on Communications | 978-1-7281-7122-7/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICC42927.2021.9500499

***Abstract*—The COVID-19 pandemic has adversely affected the lives of millions of people worldwide. With an alarming increase in COVID-19 cases, it is important to detect and diagnose COVID-19 in its early stages to prevent its spread. To diagnose remote patients, the Internet can be useful for accessing data of that patient. But, the Internet has also had issues related to data security, reliability, and privacy. Motivated by these challenges, in this paper, we propose a Blockchain (BC) based COVID-19 detection scheme (*BCovX*) for fast and reliable diagnosis of COVID-19 using chest X-Ray (CXR) images. For fast and accurate detection of COVID-19 using CXR, *BCovX* consists of a Convolutional Neural Network (CNN) model, using which a patient can be diagnosed for COVID-19 remotely. CNNs have performed successfully in medical imaging classification. *BCovX* provides reliable and secure data access and exchange using BC and smart contracts (SC). To solve issues related to data storage and its associated cost, the InterPlanetary File System (IPFS) protocol is used to store medical data. We also present a real-time SC developed in Solidity to govern the transaction between the patient and the doctor. The SC has been compiled and deployed on Remix Integrated Development Environment (IDE). Finally, we have evaluated the performance of *BCovX* with traditional schemes in terms of storage cost, bandwidth requirements, and accuracy of the CNN model.**

Index Terms*—COVID-19, X-Ray images, Blockchain, Convolution Neural Network, Smart Contract, Security

I. Introduction

The advancements of technology have affected immensely in the lives of human beings by bringing automation, ease, and precision over regular day-to-day problems. As the cities become smarter through the Internet-of-Things, more innovative aspects are introduced of various technologies into our daily lives. However, due to the outbreak of the COVID-19 pandemic, there is an urgent need to introduce different solutions using these cutting-edge technologies to fight against it. All the technologies are born out of purpose, so we have innovations happening in the existing technologies to make resources more accessible and communication faster everywhere. For example, the use of drones for contact-less deliveries, development of automated systems for sanitation and waste disposal, design of specialized instruments and kits, telesurgery in remote locations, etc.

Despite so many transformations, there is still a lack in terms of innovation and when it comes to COVID-19 detection. First of all, people in remote areas in many countries don't possess adequate supply and availability of medical kits or experienced doctors for the checkup and supervision. Also, restrictions on the movement of people have been imposed in many countries, making it difficult for people to travel across areas and access these resources in their emergency and times of need. Moreover, these resources are not efficient enough and lack in terms of accuracy and expertise. The cases of false negatives, that is predicting negative for a COVID-19 infected person, are of greater concern, either due to insufficient or inefficient medical expertise or faulty equipment. When that person contacts other people, it makes this pandemic even more severe.

The Artificial Intelligence (AI) is a proficient technology that can deal with the current crisis of the world. Some of the major applications that have come up in the pandemic include the development of drugs and vaccines to cure COVID-19, reduction of workload of healthcare workers, screening, tracking, and predicting current and future patients, early detection, and diagnosis of the infection [1]. AI can also be used for COVID-19 detection using predictive modeling. Although AI may not be fully accurate in making predictions, it still can be used to deal with the issue of scarcity of resources. In such remote places, people can get their X-Rays, and then using the AI predictive model of AI the X-Ray can be checked for detection of COVID-19 as an initial measure rather than ignoring any symptoms due to lack of doctors there [2]. However, an AI-based COVID-19 Detection System deployed on a traditional network will be centralized and prone to problems of data privacy and security. In the year 2019, the crucial healthcare data of 41.4 million patients were breached along with a 49 percent increase in hacking [3]. In such applications, the data privacy of reports generated regarding COVID-19 detection is a highly sensitive subject of healthcare which must be addressed appropriately when designing and developing systems.

Hence, the above-discussed issues of centralization can be solved using the revolutionary BC Technology. BC is an emerging distributed ledger technology that can efficiently address the subject of data privacy while maintaining decentralization. It offers data encryption, validation, and independent data verification, making it more reliable than traditional networks. BC also supports partial anonymity, which means transactions are done anonymously while maintaining the trust within the system through achieving a consensus [4]. Each transaction on a BC network is authenticated and linked together with a timestamp in the chain. This cuts out the need for third-party processors and provides transparency as the data is stored permanently and can be referred back at any time needed. Using these properties along with immutability and decentralization, an AI-based COVID-19 detection framework can be deployed to ensure data privacy. Hence, BC becomes a natural candidate over the traditional networks to deploy applications.

Many schemes, that describe the AI system for COVID-19 detection, are given by many authors [5]. For example, the authors in [6] have proposed deep learning-based COVID-19 diagnosis prediction considering four different CNN based models. Then the authors in [7], [8] and [9] have proposed different recommendation systems for COVID-19 detection using X-Ray images. However, they have not considered the problems of data security and privacy of patients. Christodoulou *et al.* [10] have described a health information exchange system using BC technology. The authors have focused on increasing demand for sharing patient information efficiently and securely. But they have not explored the role of AI in detecting COVID-19 among patients. Kalla *et al.* in [11] have explored different use-cases of BC during COVID-19 but the authors are unable to implement it.

With over 28 million COVID-19 cases globally, an automated system that can perform early diagnosis accurately is required, which ensures data security and reliability. There is a potential risk when a case goes unnoticed among such a large volume of patients, which may further transmit the virus to other people. We wish to address these issues, in this paper, through a BC-based framework that can detect COVID-19 among patients competently. Hence, there is a need to embed cutting edge technologies of AI and BC to provide a solution for COVID-19 diagnosis in isolated locations with inadequate doctors and other test equipment. Thus, motivated by the above facts, we present the Ethereum public BC-based scheme where the patients and doctors can be registered through authorization done by third party e-KYC. Then, *BCovX* can be used for remote diagnosis of COVID-19. Also, we present a CNN-based prediction model to detect COVID through chest X-Ray images to automate the initial diagnosis of patients in *BCovX*. In this way, we introduce decentralization and data management using BC and deep learning-based prediction models for COVID-19 diagnosis.

A. Research contributions

The research contributions of this paper are as follows.

- A framework that allows users to test their X-Ray reports of COVID-19 in the case of no or sparse medical expertise using deep learning models of AI.
- Further, BC technology is embedded for decentralization. It also provides end-to-end integrity in communication and data exchange, traceability and transparency in the system comprising of the medical practitioners and the users against unauthorized access.
- Finally, we evaluate the performance of the proposed approach through parameters such as bandwidth, data storage cost, accuracy, precision, f1-score, and recall.

B. Organization

The rest of the paper is organized as follows. Section II presents the system model and problem formulation. Section III describes the proposed approach of *BCovX*. Section IV shows the performance evaluation of *BCovX*, and finally, the research work is concluded in Section V.

II. BCovX: System model

The system model of *BCovX* is a BC-based secure remote COVID-19 detection system using deep learning. The system consists of entities like a patient (E_p), Doctor (E_d), Hospital (E_h), and Authority (E_a), IPFS Protocol and wallets of the patient (W_p) and the doctor (W_d). The entities are checked for their authenticity through e-KYC and are associated with each other through SC deployed over public BC. The health records of patients are stored on IPFS in return of which $IPFS_{key}$ is received and mapped with wallet address in SC. Doctors can request for patients' data using it's $IPFS_{key}$. The doctor will analyze data and update the records with a conclusion on IPFS. Every transaction between entities is stored immutably in the form of a block on Ethereum BC, which can then never be altered.

The BCovX system is made of an entity set E of four entities: patient entity E_p, doctor entity E_d, hospital entity E_h, and authority entity E_a such as $\{E_p, E_h, E_a, E_d\} \in$ E. Patient entity E_p consists of n patients $\{P_1, P_2,..., P_n\}$. Doctor entity E_d is made of m doctors $\{D_1, D_2,..., D_m\}$ associated with k hospitals $\{H_1, H_2,..., H_k\}$. The patient P_i can be associated with at most one doctor D_j. Doctor D_j can be associated with at most one Hospital H_l. Thus the mappings can be mapped as follows.

$$\{n, m, k\} > 0,$$
$$f_1 : D \rightarrow H(m:1), f_2 : D \rightarrow P(1:n), f_3 : P \rightarrow H(k:1) \quad (1)$$

The medical records of the patients $R = \{R_1, R_2,..., R_n\}$ are stored in IPFS with their hash keys $HK = \{HK_1, HK_2,..., HK_n\}$. The record R_i of patient P_i is stored on IPFS as follows

$$R_i = \{HK_i, ID(E_p^i), ID(E_d^j), ID(E_h^m), PD_i, T_{key}\} \quad (2)$$

where PD_i is the health data of the patient. *ID(E)* represents the id of respective entity. The doctor can access the data of the patient using its IPFS hash key HK_i. T_{key} is the hash value of the transaction between patient and doctor. The transaction can be done using e-wallets $W = \{W_p, W_d\}$ of patient and doctor. The transaction consists of the following details:

$$T = \{W_s, W_r, T_{stamp}, GAS_{value}\} \quad (3)$$

where W_s is the wallet address of the sender and W_r is the wallet address of the receiver. Transactions can be validated using SCs. To write any transaction on the Ethereum BC some amount need to be paid in GAS, that is denoted by GAS_{value}. T_{stamp} is the time the transaction happened. The sender side T_{key} is encrypted with first senders' private key and then the receivers' public key. At the receiver side, it will be decrypted first using the senders' public key and then the receivers' private key. The payment can be done through cryptocurrency on BC.

III. BCovX: the proposed approach

This section describes the working of *BCovX*, a BC-based COVID-19 Detection Scheme. *BcovX* is a secure way to connect doctors with patients as it cuts off the need for a third party in controlling the system and prevents their physical interaction. The BC-based COVID-19 Detection Scheme

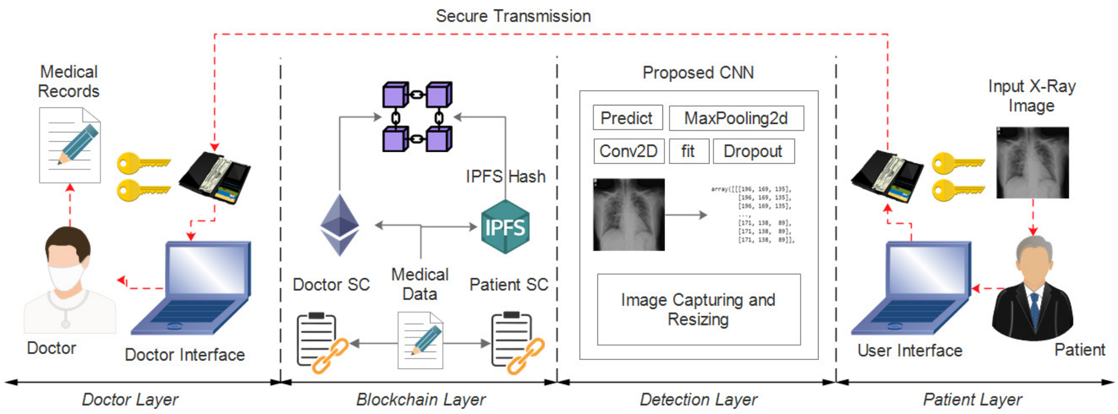


Fig. 1: BCovX Architecture

BCovX is shown in Fig. 1. The 4-layered architecture along with its various components of *BCovX* is described as follows.

A. Patient Layer

In the patient layer, the patient can interact with the *BCovX* interface. The interface provides two options to the patient, Test Remotely and Contact Doctor. In the Test Remotely option, the patient $P_i \in E_p$ can be tested for COVID-19 by our proposed CNN model for COVID-19 detection using the CXR image. The detailed working of the proposed CNN model is described in the Prediction Layer. The results of the prediction serve as a basis for the health conditions of the patient. In the Contact Doctor option, the patient can directly contact a remote doctor of their choice for COVID-19 detection tests through the BC Layer. The patient $P_i \in E_p$ uploads their details with CXR image on BC using its IPFS hash key HK_i. This request is forwarded to hospital $H_k \in E_h$. The patient will be recommended with the available doctors $D_j \in E_d$ related to the hospital H_k. After choosing a consultant, a digital SC is established between the hospital and the patient to verify and validate the data. The CXR image can be exchanged between the patient and the doctor using the BC network. The data exchange is secure, as discussed in the BC Layer.

B. Detection Layer

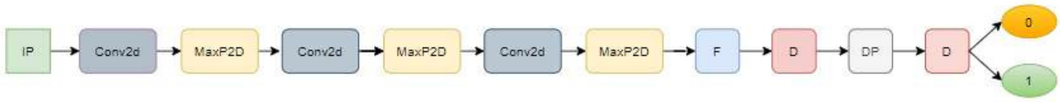
This layer consists of the proposed Convolutional Neural Network or CNN model that will predict the class of the CXR image uploaded by a patient. First, the patient uploads the CXR$_i$ image that the patient P_i wants to classify. The model will perform preprocessing on that image as shown in Algorithm 1. To predict the image, first it will be converted into a matrix of 64×64 pixels. After preprocessing, the model will classify whether the image belongs to class 0 or class 1 (0 for COVID-19 positive and 1 for Normal). This prediction will be redirected to the patient.

Convolutional Neural Network: Convolutional Neural Networks are one of the most popular deep neural networks. In deep learning, CNN models are widely used for imaginary classification. CNN models are made of several layers and each layer consists of several neurons that are connected to the neurons of the next layer. The output of neurons is called activation. The activation of the previous layer's neurons is forwarded to the next connected layer. The activation of the last layer's neurons determines the prediction.

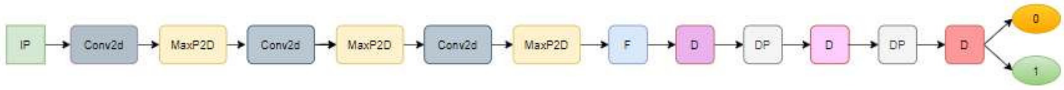
Algorithm 1 BCovX Algorithm

Input: $E, D_j, P_i, HK_i, R_i, H_m, CXR_i, D$
$where D_j \in E_d, P_i \in E_p, HK_i \in HK$, D=$\{I_1, I_2, ..., I_n\}$
Output: Get prediction whether the patient is COVID-19 affected or not
procedure PREPARE_MODEL(D)
  $D \leftarrow$ resize_images($D, size = 64 \times 64$)
  Split dataset into train and test data
  $D\_train, D\_test \leftarrow$ split_dataset(D)
  Train model using train dataset
  $M \leftarrow$ train_model(D_train)
  Test model using test dataset
  $M \leftarrow$ test_model(M, D_test)
end procedure
procedure PATIENT_REQUEST(E, HK_i)
  while (True) **do**
    if ($E \in E_p$) **then**
      Selected entity is patient
      Patient P_i uploads image of CXR and predict it using proposed model
      $E_p \rightarrow$ upload_image(CXR_i)
      predict_image(M, CXR_i)
      $E_p \leftarrow$ get_result
      Patient P_i uploads records on IPFS using hash key
      $E_p \rightarrow$upload_records(HK_i, R_i)
      $E_p \rightarrow$ request(HK_i, H_m)
      $E_p \rightarrow$ grant_data_access($D_1, D_2, .., D_m$)
      $E_p \leftarrow$ NOTIFY("Result")
    else
      $E_p \leftarrow$ NOTIFY("Denial to upload data")
    end if
  end while
end procedure
procedure DOCTOR_REQUEST(E, HK_i)
  while (True) **do**
    if ($E \in E_d$) **then**
      Selected entity is doctor
      if (data_access(HK_i)==True) **then**
        If data access granted doctor D_i fetch records of patient P_i
        $E_d \leftarrow$ fetch_records(HK_i)
        Doctor D_j analyze data and update records with result
        $E_d \rightarrow$ update_records(HK_i)
        $E_d \rightarrow$ NOTIFY(P_i, "Result")
      else
        $E_d \leftarrow$ NOTIFY("Denial to access data")
      end if
    end if
  end while
end procedure

The proposed model: Fig. 2 shows the architecture of the proposed model and the pre-trained model of CNN. The pre-trained model contains three sets of Conv2D and MaxPooling2D layers. In the first set, the Conv2D layer is implemented with 32 filters. The remaining two sets contain Conv2D layers with 64 filters. MaxPooling2D is implemented



(a) Pre-trained model



(b) Proposed model

Fig. 2: Architectures of CNN models (IP: Input; Conv2D: Convolution; MaxP2D: Maxpooling; F: Flatten; D: Dense; DP: Dropout; 0: COVID-19; 1: Normal)

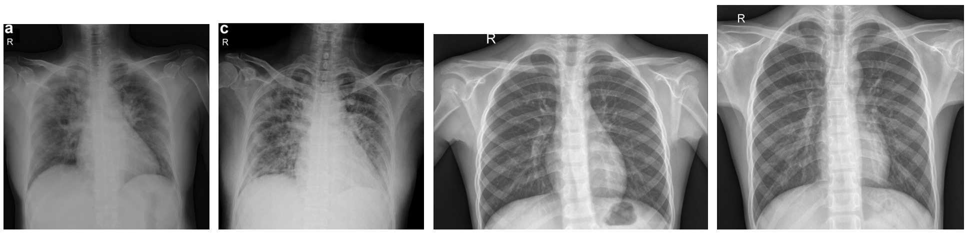


Fig. 3: Some COVID-19 CXR images from dataset

with a 2×2 pooling size. After that the pre-trained model contains these layers : (a) Flatten (b) Dense layer of sigmoid activation function with 128 units (c) Dropout with 0.5 dropout ratio (d) Dense layer of softmax activation function with 2 units output. The two more layers are added in the proposed model after the Dropout layer: (a) Dense layer of the Relu activation function with 64 units (b) One more Dropout layer with a dropout ratio of 0.5. Instead of the Dense layer of sigmoid activation function with 128 units, the proposed model contains a Dense layer of the Relu activation function with 128 units. There is also one change in the last Dense layer of the pre-trained model. Because the model used here is for binary classification, the last layer of the proposed model is the Dense layer of sigmoid activation function with 1 unit instead of the softmax activation function. This change from softmax to sigmoid function has improved the performance of the model. Adam optimizer was used to optimize the performance of the model. In the CNN model, each layer predicts particular features of the image. The output of a layer is forwarded to the next layer. The first three layers of the model are a set of convolutional and pooling filters. The last layer of the model is sigmoid activation which predicts the class of the CXR image. The model was implemented using Python programming language with Keras library.

C. BC Layer

It is a network layer that connects patients and doctors. The major components of this layer are SC developed over Ethereum BC and IPFS protocol.

Smart Contract: The SC is a set of programs that are written in programming languages like Python, Go, Kotlin and Solidity. They provide self-defined and self-validate transactions in BC. After testing the CXR image on our model, the patient can further choose to share their records with the doctor through our network.

IPFS Protocol: IPFS is a peer-to-peer protocol for storing and sharing data. IPFS generates the hash of the file that the patient wants to upload, known as IPFS hash. This hash becomes available on the IPFS network. Now, when the patient wants to share this file, they can simply share this generated hash to the doctor. The doctor can call the hash from the IPFS and can get a copy of the file uploaded by the patient. The reason behind choosing IPFS for storing the data is that it provides less cost for storing data than direct storing on BC. It also provides less latency for writing and retrieving of data thereby making transmission faster.

The patient P_i uploads the records R_i to IPFS and sends request to hospital H_m with its IPFS hash key HK_i. SCs connect the patient layer and the doctor layer. At the hospital side, a Doctor D_j accesses the records of the patient using the IPFS hash key HK_i. Every transaction between the doctor and the patient is stored on BC. These transactions are first validated using SCs.

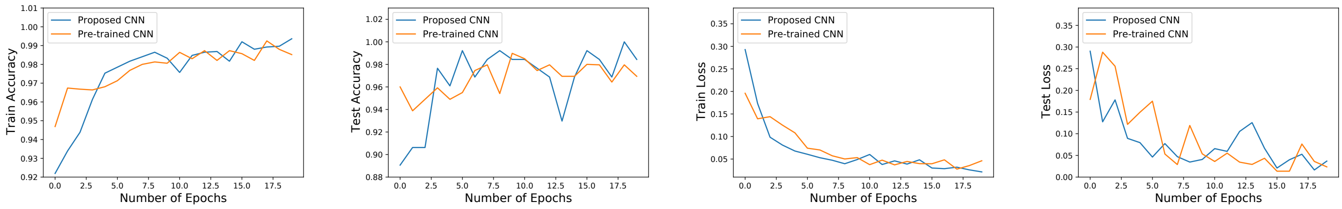
D. Doctor

In the doctor layer, the doctors can access the *BCovX* interface and interact with the remote patients using. The doctor $D_j \in E_d$ can access and examine the medical records of the patient $P_i \in E_p$ using its IPFS hash key. The doctor will upload their conclusion on patients' health about their COVID-19 test on IPFS requesting the patients' IPFS hash key. In this way, the doctors don't need to come in physical contact with the suspected patients. On the other hand, patients can be easily tested for COVID-19 and seek medical care.

IV. Performance Evaluation

A. Dataset Description

The model was trained on two datasets from GitHub and Kaggle with a total of 2875 CXR images of which 211 CXR images were of COVID-19 patients as shown in Fig. 3. The Kaggle dataset also contains images of pneumonia and bacterial infection as well as the CT scans. Dataset was recreated by removing all CT scans and other bacterial, virus, and pneumonia X-Ray images. Dataset is trained for 20 epochs



(a) Training accuracy (b) Testing accuracy (c) Training loss (d) Testing loss

Fig. 4: Epoch wise accuracy and loss comparison of *BCovX* model with pre-trained CNN model

in a batch size of 64 and every batch contains 40 images for prediction. The images were resized to 64 $\times$ 64 pixels resolution and then fed into the model.

B. Classification Performance Parameters

Accuracy: Accuracy is a parameter from which we can predict how well our model is doing. It is a fraction that shows how many images are predicted correctly by the model. Fig. 4 shows a comparison of the accuracy between the proposed model and the pre-trained model. The X-axis represents the number of epochs or iterations and the Y-axis represents accuracy for every epoch. Fig. 4a shows a comparison for the training accuracy and Fig. 4b shows a comparison for the test accuracy between the proposed CNN and the pre-trained CNN. Overall, the proposed model has a higher train and test accuracy than the pre-trained model.

Loss: Loss is a parameter by which the bad performance of a model is estimated. It is a fraction that shows how many images are predicted incorrectly by the model. Fig. 4 shows a comparison of loss between the proposed model and the pre-trained model. The X-axis represents the number of epochs or iterations and the Y-axis represents the loss for every epoch. Fig. 4c shows a comparison for the training loss and Fig. 4d shows a comparison for the test loss between the proposed CNN and the pre-trained CNN. The proposed model has higher accuracy than the pre-trained model, which implies that the proposed model has a lower loss than the pre-trained model.

C. Confusion Matrix

The confusion matrix is another parameter to determine the performance of a model. It is useful to see how our model performed and where it went wrong at the same time. Every value defines the value predicted for that class. The X-axis of the matrix depicts the predicted values of the model and the Y-axis depicts the actual values of the class as depicted in Fig. 6. It is a table with a combination of four values: True Positive (TP), False Positive (FP), True Negative (TN), and False Negative (FN). These four values are used to predict other parameters like Recall, Precision, F1-score, and ROC Curve.

- TP: When the model predicts positive values correctly from actual positive values i.e., the actual value was positive and the model predicts positive. In our analogy, the model has predicted an image as COVID-19 and the image was actually of a COVID-19 patient.
- TN: When the model predicts negative values correctly from actual negative values i.e., the actual value was negative and the model predicts negative. In our analogy, the model has predicted an image as normal and the image was actually of a normal patient.
- FP: When the model predicts positive values incorrectly i.e., the actual value was negative and the model predicts positive. It is an error also known as Type 1 Error. In our analogy, the model has predicted an image as COVID-19 but the image was actually of a normal patient.
- FN: When the model predicts negative values incorrectly i.e., the actual value was positive and the model predicts negative. It is an error also known as Type 2 Error. In our analogy model has predicted an image as normal but the image was actually of a COVID-19 patient.
- Recall: It is the value that defines out of true classes, how many examples have the model predicted correctly. It is the ratio of all the correct positive predictions model has predicted and the total correct predictions. This value should be high for a good model.
- Precision: It is the value that defines out of all positive classes, how much the model has predicted true positives i.e., it is the ratio of all true positives the model has predicted correctly and total positive classes. This value also should be high for a good model.
- F1-score: This score represents the actual comparison between the two models. When a model has high precision value and low recall or vice versa, it is difficult to compare them. F1-score makes this easy by taking the harmonic mean of recall and precision. Hence, it uses both values. Using this score, we can compare the two models more precisely. The model with a high F1-score is a good one.

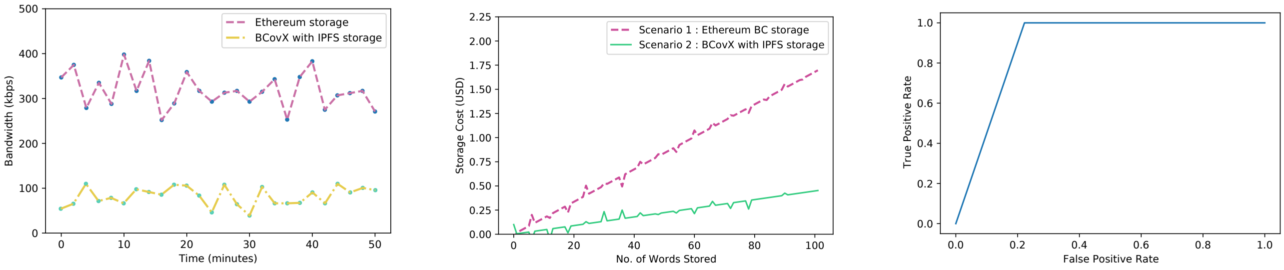
  All the parameters discussed above are shown in Table I in which the proposed CNN outperforms the pre-trained CNN model.

TABLE I: Prediction performance comparison

Model	Recall	Precision	F1 Score	Accuracy (%)	Loss (%)
Proposed CNN	1	0.78	0.87	98.44	0.03
Pre-trained CNN	1	0.76	0.86	96	0.038

D. ROC Curve

ROC Curve stands for the Receiver Operating Characteristic Curve, is used to determine the performance of any classification model in Machine Learning. It represents the capability of a model that how much it can differentiate between true positives and true negatives. AUC (Area under the ROC Curve) is used to determine the performance of a model. Its value ranges from 0 to 1. AUC with 0 represents that model prediction is 100% wrong and 1 represents the model prediction is 100% right. The figure shows the ROC



(a) Bandwidth comparison (b) Data storage cost comparison (c) ROC curve of *BCovX*

Fig. 5: IPFS comparison with traditional BC and ROC curve of proposed prediction model

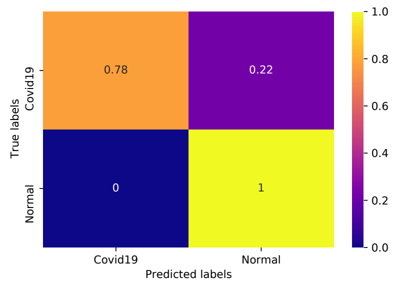


Fig. 6: Confusion matrix of *BCovX*

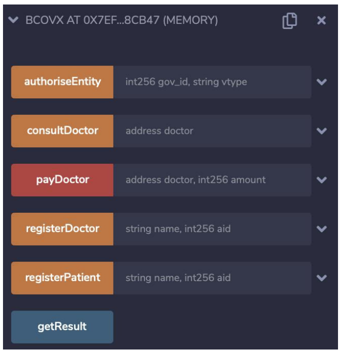


Fig. 7: BC simulation Remix IDE interface for SC testing

of the proposed model. The X-axis of the curve is the False Positive Rate (FPR) and the Y-axis is the True Positive Rate (TPR) as shown in Fig. 5c. The curve at 1 shows that the model is able to successfully distinguish between the classes COVID-19 and normal.

E. BC Simulation and IPFS Off-chain Storage Comparison

Fig. 5a shows the bandwidth requirements of Ethereum BC and *BCovX* at different times. Ethereum BC requires a fairly high bandwidth compared to *BCovX*. Fig. 5b shows the cost comparison between *BCovX* and Ethereum BC for storing words. *BCovX* stores words using distributed *IPFS* protocol, whereas Ethereum stores words on the BC directly. Hence, *BCovX* generates lesser costs for storing patient data. This shows that *BCovX* is efficient both in terms of data storage and bandwidth requirements.

Fig. 7 shows the IDE interface for *BCovX*, which shows the functions and the input parameters of the functions. The SC of *BCovX* has been compiled and deployed on Remix Integrated Development Environment.

V. Conclusion

In this paper, we propose *BCovX*, a BC-based COVID-19 Detection Scheme using CXR images. We briefly describe the detection and transmission mechanism of our system. We also ensure data privacy and security during these processes. We define the SC between the patient and the doctor for transmission. We compile and deploy the SC on Remix IDE. Then, we evaluate the performance of *BCovX* by comparing its storage costs and bandwidth requirements with traditional Ethereum BC. We also evaluate the performance of the CNN model using measures such as accuracy, loss, confusion matrix, and ROC curve. In the future, we will embed Ethereum 2.0 to verify scalability issues of public BC over testbed.

References

[1] D. Dong, Z. Tang, S. Wang, H. Hui, L. Gong, Y. Lu, Z. Xue, H. Liao, F. Chen, F. Yang, R. Jin, K. Wang, Z. Liu, J. Wei, W. Mu, H. Zhang, J. Jiang, J. Tian, and H. Li, "The role of imaging in the detection and management of covid-19: a review," *IEEE Reviews in Biomedical Engineering*, pp. 1–1, 2020.

[2] R. Gupta, A. Kumari, S. Tanwar, and N. Kumar, "Blockchain-envisioned softwarized multi-swarming uavs to tackle covid-19 situations," *IEEE Network*, pp. 1–8, 2020.

[3] J. Davis, "The 10 biggest healthcare data breaches of 2020." https://healthitsecurity.com/news/the-10-biggest-healthcare-data-breaches-of-2020-so-far. Online; Accessed: 2020.

[4] R. Gupta, A. Shukla, and S. Tanwar, "Bats: A blockchain and ai-empowered drone-assisted telesurgery system towards 6g," *IEEE Transactions on Network Science and Engineering*, pp. 1–1, 2020.

[5] M. Abdel-Basset, R. Mohamed, M. Elhoseny, R. K. Chakrabortty, and M. Ryan, "A hybrid covid-19 detection model using an improved marine predators algorithm and a ranking-based diversity reduction strategy," *IEEE Access*, vol. 8, pp. 79521–79540, 2020.

[6] R. Sethi, M. Mehrotra, and D. Sethi, "Deep learning based diagnosis recommendation for covid-19 using chest x-rays images," in *2020 Second International Conference on Inventive Research in Computing Applications (ICIRCA)*, pp. 1–4, 2020.

[7] S. Rajaraman, J. Siegelman, P. O. Alderson, L. S. Folio, L. R. Folio, and S. K. Antani, "Iteratively pruned deep learning ensembles for covid-19 detection in chest x-rays," 2020.

[8] Y. Oh, S. Park, and J. C. Ye, "Deep learning covid-19 features on cxr using limited training data sets," *IEEE Transactions on Medical Imaging*, vol. 39, no. 8, pp. 2688–2700, 2020.

[9] J. Bridge, Y. Meng, Y. Zhao, Y. Du, M. Zhao, R. Sun, and Y. Zheng, "Introducing the gev activation function for highly unbalanced data to develop covid-19 diagnostic models," *IEEE Journal of Biomedical and Health Informatics*, pp. 1–1, 2020.

[10] K. Christodoulou, P. Christodoulou, Z. Zinonos, E. G. Carayannis, and S. A. Chatzichristofis, "Health information exchange with blockchain amid covid-19-like pandemics," in *2020 16th International Conference on Distributed Computing in Sensor Systems (DCOSS)*, pp. 412–417, 2020.

[11] A. Kalla, T. Hewa, R. A. Mishra, M. Ylianttila, and M. Liyanage, "The role of blockchain to fight against covid-19," *IEEE Engineering Management Review*, vol. 48, no. 3, pp. 85–96, 2020.