Comparative Analysis of Federated Learning and Centralized Approach for detecting different lung diseases

Access to a large dataset is necessary to improve disease detection with excellent accuracy. However, due to data confidentiality and privacy restrictions, collecting data from hospitals or other organizations is a significant challenge in the healthcare sector. Due to this, Federated Learning (FL), which adopts a decentralized approach, is developed to replace the conventional machine learning methodology in the development of improved screening methods. Since there is no requirement for data to be centralized in federated learning, patient data privacy is ensured. In this paper, we compared the sequential model and the ensemble model for both federated learning and centralized approach, two different types of models. For each approach, these models were applied on separate X-ray images for the detection of two different lung diseases: lung cancer and tuberculosis. In this paper, we also showed the analysis of their accuracy and demonstrated how FL can be the most effective strategy through comparison.

CCS Concepts: • **Federated Learning**; • **Ensemble Method**; • **Sequential Method**;

Additional Key Words and Phrases: Lung disease, FL, CNN

ACM Reference Format:
. . Comparative Analysis of Federated Learning and Centralized Approach for detecting different lung diseases. . ACM, New York, NY, USA, 11 pages.

1 INTRODUCTION

Disease detection from medical imaging has long been accomplished using machine learning (ML) [1]. Numerous studies have been conducted in order to diagnose and screen for a variety of diseases[2]. DL-based classification models have continued to advance and have been shown to be the most effective ones [3, 4]. But the main problem is right here: in order for these models to work successfully, they need access to a large dataset. Prior studies in the healthcare sector have concentrated on centralized algorithms, which assume a single data repository (database) that can store and process the data from every participant [5, 6]. However, a single point of failure introduced by such an architecture poses concerns about the integrity and privacy of the data.

In the medical field, there is a lack of publicly available data which is even spread among numerous hospitals, organizations, and clinics. Because of this, data gathering became a major issue, which makes it difficult for researchers to proceed properly. The fundamental causes for introducing decentralized method FL in the healthcare sector include data scarcity, data dispersed across multiple organizations, and patient and hospital confidentiality [7]. In contrast to non-FL, FL requires collaboration among numerous hospitals for the development of the models [8–10]. Each client or hospital trains the model and passes the information to the central server or global model and that's how confidentiality is maintained.

Our suggested work has two components. We employed both FL-based and non-FL-based models and compared them after comprehensive analysis. As a starting point, we used pre-processed X-ray pictures of lung

Author's address:

cancer and tuberculosis as our dataset, dividing each set of data into affected lung and unaffected lung. Following the FL approach, we used the ensemble method to put together the VGG19, DenseNet, and Inception DL models. We focused our attention on the non-IID characteristics in this instance. A three-layer sequential model with conv2D and max pooling 2D is used separately. These Deep Learning models are fit to the data to detect the diseases. Likewise, we applied the same architecture to the non-FL model. Just like before, we used both the ensemble and sequential methods to detect lung cancer as well as tuberculosis from the images. At the end, a comparative analysis of each model for both FL and non-FL approaches has been performed and shown which approach and model performs the best.

2 RELATED WORK

Over the past years, so many computer-aided diagnosis (CAD) systems have been developed based on Deep Learning models to diagnose a variety of diseases. CT scan images are mostly used for less noise. Paper [11] suggests a 3D multipath VGG-like network with two classifications. Lung nodules and non-nodules are categorized in one category, and benign and malignant nodules are classified in another. U-Net is adapted for data segmentation using image processing techniques. Lung cancer predictions from the 3D multipath VGG-like network and U-Net are combined to get the ultimate results. Using this architecture, the malignancy degree of lung nodules is identified and classified with 95.60% accuracy. Another research [12] describes a technique for classifying lung cancers as malignant or benign from CT images using a convolutional neural network (CNN), DenseNet. The effectiveness of CNN's outcomes is 81 percent, which is higher than that of the conventional neural network.

In paper [13], computer-aided diagnostic (CAD) systems that employ deep learning to analyze chest X-rays to detect pulmonary tuberculosis (TB) are reviewed in detail. Within Deep Learning, for the detection of TB, convolutional neural networks (CNNs) are frequently employed. Convolutional layers, subsampling/pooling layers, and fully linked layers are frequently found in CNN models. For the purpose of categorizing TB symptoms in chest X-ray pictures, two deep learning techniques with DCNN are examined by the authors in this work [14]. The first is a number of preprocessing techniques, while the second is a hybrid approach. The hybrid approach combines DCNN and Computer Aided Detection (CAD). The findings from both approaches are very good, with the hybrid approach having the highest accuracy. The authors in paper [15] proposed to train a ConvNet CNN model to detect TB from chest X-rays and compared it with other five different pre-trained models. In comparison to other deep CNN models for the datasets with image augmentation techniques, the Exception, ResNet50, and VGG16 models performed the best with the accuracy over 90% in all cases. Whereas, the proposed model achieved the classification accuracy of 87%.

Paper [16] says the federated learning approach offers two different types of advantages. First, a more accurate model can be created by using a larger pool of data points than could be achieved by relying just on the information stored by one centralized database. This is crucial for modern machine learning models since they frequently incorporate a high number of parameters and, thus, need a lot of training data. The second advantage is privacy; everyone keeps their local and private raw and, generally, sensitive data. Paper [17] described the reported work, which emphasizes several studies in the areas of drug development, disease prediction systems, and electronic health record systems leveraging the FL model.

The Federated Learning approach for disease detection has expanded significantly. Paper [18] suggests a system that would detect objects with an accuracy of roughly 90% by combining a Deep Learning model with a federated learning approach. In paper [19], the study addresses the issue of federated learning for COVID-19 identification on X-ray pictures. The authors show that the proposed federated learning framework is robust

and achieves outcomes that are comparable to those of a centralized learning process, despite the decentralized data, non-IID, and imbalanced aspects of the data distribution. For the purpose of detecting lung nodules and predicting their severity, paper [20] suggests decentralizing the ML model in distributed databases.

In order to mitigate model overfitting, enhance model generalization with better outcomes, and save model training time, the research [21] suggested a sampling-based content diversity approach that is tested on luna16 data. Authors compared 3D ResNet18 Dual Path Faster R-CNN of federated learning algorithm with other federated learning algorithms of deep learning in order to further verify it. According to the experimental findings, the 3D ResNet18 Dual Path Faster R-CNN of federated learning algorithm produces the best outcomes.

In this study [22], authors suggest a federated learning-based assisted diagnosis paradigm for cancer patients. Various physical examination indicators of patients were used as input for a convolutional neural network based on the federated learning architecture. Here, the multilayer perceptron neural network technique, gradient ascending tree, linear regression, support vector regression, and Bayesian regression were all applied. The accuracy of CNN's federated prediction model, which is based on enhanced joint modeling and simulation on data from the five different types of cancer, attained more than 90%; this accuracy surpasses that of single modeling machine learning tree models, linear models, and neural networks.

Lastly, this study [23] provides a detailed summary for academics and reviews the most recent deep ensemble models. The ensemble models are widely divided into explicit/implicit ensembles, homogeneous/heterogeneous ensembles, decision fusion techniques based deep ensemble models, bagging, boosting, stacking and negative correlation based deep ensemble models. The use of deep ensemble models in many fields is also briefly covered. Ensemble methods are being used in detection of diseases using chest X-ray [24].

3 PROPOSED FRAMEWORK

In this segment, we will discuss our proposed method to differentiate between affected and non-affected lung diseases. Firstly we will talk about the basic working procedure of federated learning (FL) and non-federated learning (non-FL). Secondly, we will give an overview of our both frameworks. Thirdly, we will discuss each model's architecture. Lastly, we will give a brief idea about the client-side model part and server-side model part for FL and also the test-train process for Non-FL.

3.1 Introduction

We know that for health purposes, data security is one of the most important parts. That's why, we have run two models for both FL and Non-FL framework and compared them. For Fl, we will consider only Non-IID properties. As we are working for health sectors and images are from hospitals, so datasets can create many problems. The relation between clients and local data distribution and also non-overlapping in sending and receiving data from clients side are also the key points of this framework. There will be a fixed number of clients, K (in our experiments, it was 10). Starting of every loop, clients C will be selected randomly and the current global algorithm state is sent to each client. After That the server will receive the updates and apply it to the global state. Here, learning object function is $f_i(w) = \ell(x_i, y_i, w)$. Besides, for each clients K, there will be a set of data points, Pk. Moreover, Nk is the particular data points for that client. The final objective of this form is :

$$\min_{w \in \mathbb{R}^d} f(w) \quad \textit{where} \quad f(w) \stackrel{\text{def}}{=} \frac{1}{n} \sum_{i=1}^{n} f_i(w) \tag{1}$$

From the paper [25], FedAvg can work as a central model when it is connected with IID data. Besides, in non-IID cases, the accuracy rate of fedAvg is reduced, at the time of training on highly skewed non-IID data. After that for the Non-FL, we have trained and tested the whole data centrally to classify the images. Lastly, we have calculated the accuracy results using both frameworks.

3.2 Overview of frameworks

Basically our framework is divided into two parts. In the first part, we have used a federated learning(FL) framework which is client-server architecture implementing FedAvg algorithm to classify affected and unaffected lungs in lung cancer and also affected and unaffected chest in Tuberculosis(TB). In the second part, we have worked with another framework, non-federated learning (non-FL). Using the splitting method in central datasets, we classify the same things like FL.

In FL, we have worked with a deep convolutional neural network (CNN) to classify the images for both lung cancer and Tuberculosis diseases. This framework follows some steps to complete the procedure. Initially there is a global server (g) which is maintained by the central server and the server is shared with a subset of all randomly selected clients, S_t (for our case, clients are hospitals). After that each client (k) which belongs to the subset (S_t) performs training operations and sends the model updates, w_K^t to the server. After receiving the updates, the server calculates the average model (w^t) in every round (t) to update the global server. The equation looks like:

$$w^t \leftarrow \sum_{K=1}^{K} \frac{n_k}{n} w_k^t \tag{2}$$

Here, N is the total number of data points and N_k is the data points for each client. These steps are combinedly run for each round to update the global server. Moreover, for non-FL, we do not have the client-server architecture. We simply set epochs, split and batch-size functions to train the central dataset. In this case, after completing each epoch, the train model will update.

3.3 Ensemble model architecture

There are many models to train the datasets. We know that for health sectors, more accuracy is always appreciated. To get better accuracy results, we have used a high performance model which is an ensemble model. This model is combined with the VGG19, Inception and DenseNet models. For each model, the whole process is completed with 3 layers . Firstly, each model starts a new keras sequential model and then adds the convolutional part of each model. After that, flatten the output of each model because it is from a convolutional layer and lastly add a dense layer for combining features so that the model can recognize the image. At first we ran VGG19, then Inception and finally we ran denseNet .Though we need a GPU to run this whole model but in the end, this model gives us more better results than individual ones.

3.4 Sequential model architecture

For this model, we have run three layers sequentially. For each layer, we have run conv2D and max pooling 2D. Conv2D takes some arguments like kernel size, activation, input shape etc. We set the filter value to 32 and for the second, third layer we set the value to 64, 128 respectively. It helps the model to remove the complexity of our dataset. Before running this model, we have reshaped our images so that we can decrease the ram uses. After that, reshaped image's size are passed through the input shape argument. Besides, we have run max pooling 2D for downsampling the input along its height and width.

3.5 Client-side part for FL

In the client-side model, the training operation is performed. First of all, we have to create our clients function. In this function, we have placed the sharding data at each client. After that, we have used a batch-data function which returns us a tfds object for each client. Tfds object basically gives us the information about version, features, split etc. Lastly, using the batch data function, we batch the training and test data for each client. In this way, this model has maintained the data privacy for training and testing for each client.

3.6 Server-side part for FL

Server-side model has a global model who manages the overall process. Server-side model is a central model who receives the update from each client for each federated round. After that it sums them up to build a new average model and finally update the global model.

Algorithm of the whole process of federated learning is given below.

Algorithm 1 Federated Learning for lung diseases Detection. The functions test_model,weight_scalling_factor and 'scale_model_weights' are run by each client. Sum_scaled_weights to be run by a global model.

for each communication round comms_round= 0,1,2,..,100 **do**
  ASSIGN EMPTY LIST to scaled_local_weight_list
  for each client in clients_names **do**
  receive w G comms_round from Central Serve
  local_model ← $fit(\text{clientsData}_{client}^{client\_names})$
  test_model ← $(X\_test_{client}^{client\_names}, Y\_test_{client}^{client\_names}, local\_model, comms\_round = 0, 1, 2, .., 100)$
  $scaled\_factor \leftarrow weight\_scalling\_factor(clients\_batched_{client}^{client\_names}, clients)$
  scaled_weights ← $scale\_model\_weights(scaled\_factor) scaled\_local\_weight\_list.insert(scaled\_weights)$
end for
average_weights ← $sum\_scaled\_weights(scaled\_local\_weight\_list)$
$update \leftarrow global\_model(average\_weights_{client}^{client_n ames})$
$test \leftarrow global\_model$
end for

3.7 Test-train process for Non-FL

In the Non-FL part, there is no client server and global server. At first, we set the epoch value, image size and batch size. After that we split our data set with a fixed value for train and test and finally, by running the model, we can get the accuracy result.

4 EXPERIMENTS

As we have worked for both FL and Non-FL architecture, the experiments are divided into two types. One for client-global server architecture and another one for central server. Besides, we have used the same shape of images for both architectures. As we know that collecting medical images is hard for normal humans, we have used the kaggle datasets. There are a total of 6598 for lung images and 2700 for tuberculosis images.

For client-global server architecture, we have created 10 clients (here, clients mean hospital) for both lung and tuberculosis classification. We took 6006 images for training purposes and 592 images for test purposes for lung images. There are a total 3 categories in lung images set such as normal lung images, adenocarcinomas cancer

images and squamous cell carcinoma cancer images. Considering real life scenarios, we took a different number of images for these 3 categories. After that, we randomly shuffled our whole training and test data. Then we divided the training data points into 10 clients. As we have worked with a non-IID training case using weight scaling factor function, we divided the data points randomly among those clients like for one simulation it was like 21%, 10%, 6%, 9%, 11%, 5%, 4%, 7%, 12%, 15%.

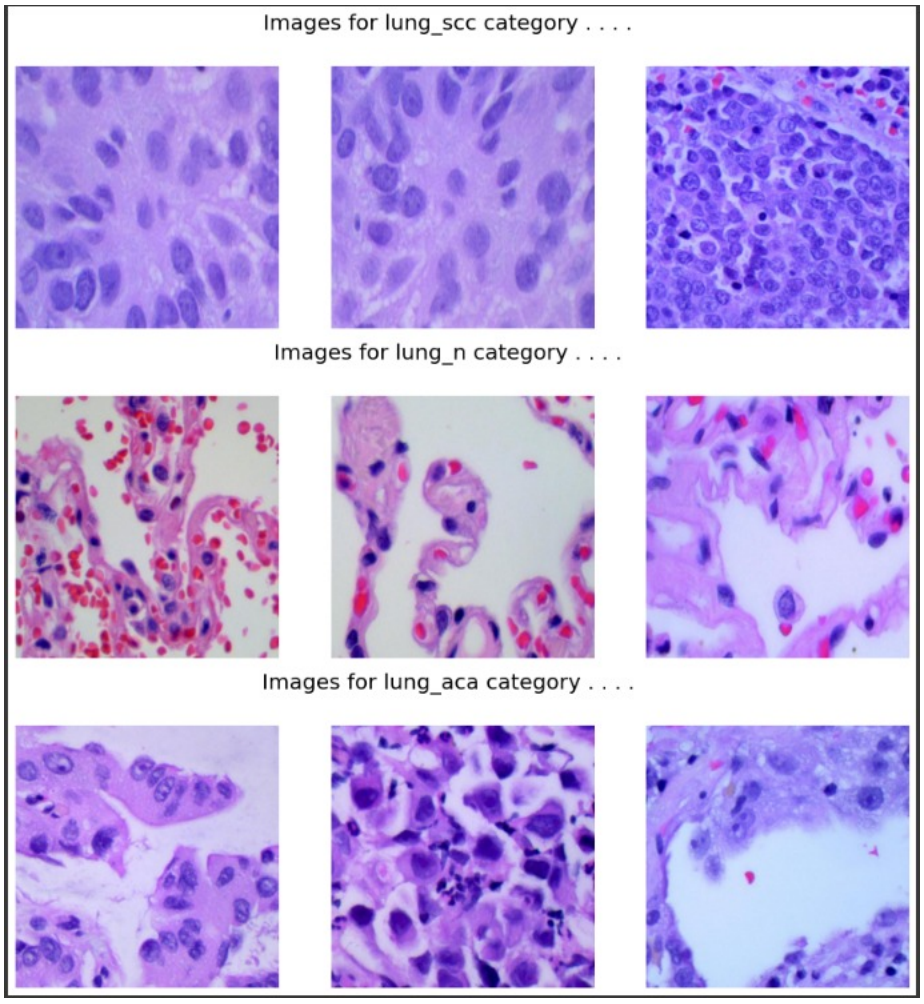

Fig. 1. Classification of lung images

In the same way, to make it similar, we created 10 clients for tuberculosis also. Here, we took 2160 images for training purposes and 540 images for test purposes. But in this case, there are only two categories such as tuberculosis X-ray and normal chest X-ray. After that, we fitted these in different models, calculated the accuracy rate and plotted them in different graphs. It is mentionable that all the images of each patient are unique. In this

kaggle data set, the image size was much larger and so we had to reduce the size. We made the 75 for all images.

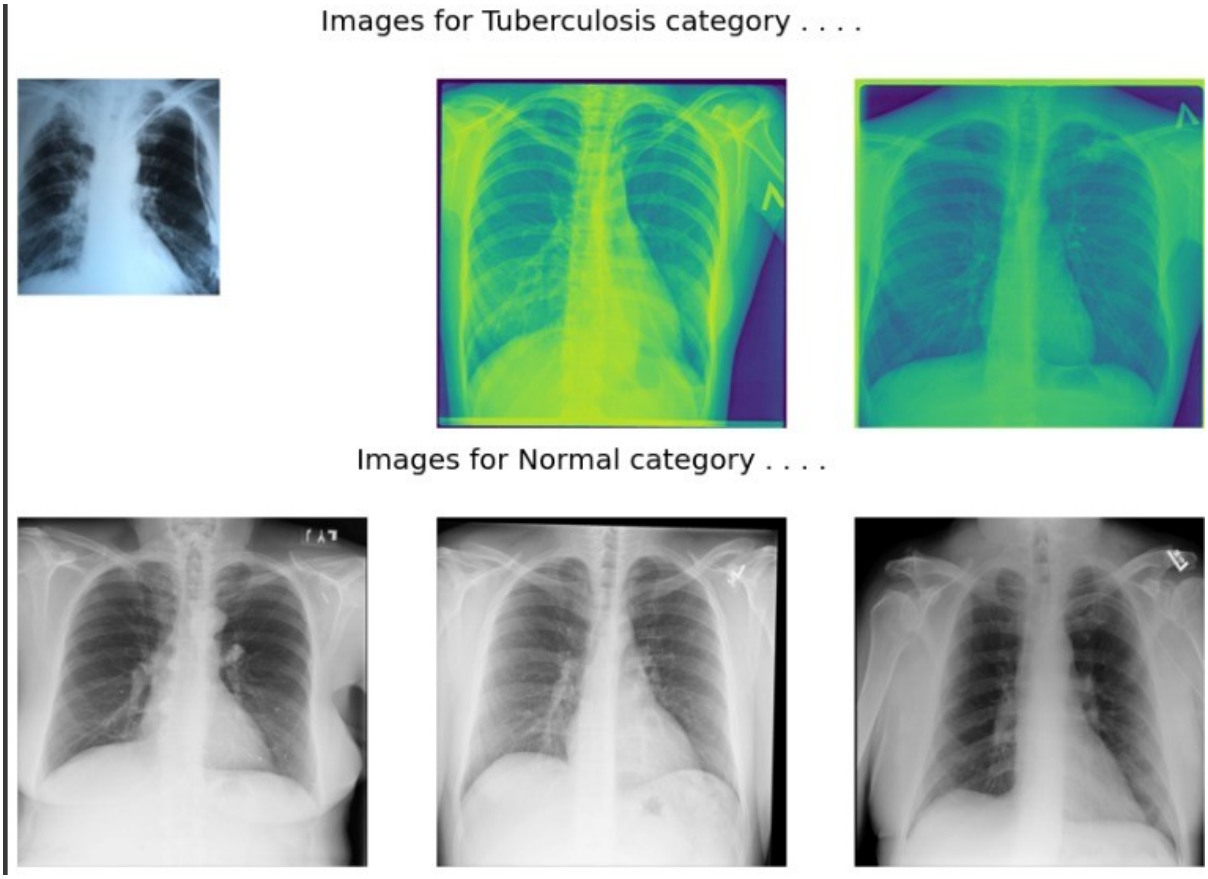

Fig. 2. Classification of Tuberculosis images

Moreover, for central server architecture, there are no clients and so we do not need to divide the whole dataset for test and training manually. Here, we took the full dataset. For tuberculosis the images were 2700 and for lung, were 6598 as mentioned before. We want to make a perfect comparison, that's why, we used the same image size for non-FL also and it was 75. In this case, we split our whole data into 30% by 70% so that in tuberculosis test images and training images were 810,1890 respectively and in lung 4620, 1979 for test and training respectively. After that, we fitted both models on non-FL and checked the accuracy and finally, we plotted them into graph.

As our main goal for this research is to make a perfect comparison for non-FL and non-IID-FL architecture, we took the same amount of data points and made the images size the same for all. Finally, we got the accuracy and graphs which helped us to get the idea about which architecture and model are more perfectful for medical diseases.

5 RESULT AND ANALYSIS

Our main goal in performing this study is to demonstrate that decentralized learning is a viable alternative to centralized learning if data privacy is a top priority. This research was carried out and carefully examined in order to acquire the best results from our Federated Learning models. We explain our analysis component in two contexts as it was conducted for two distinct disorders. We first contrast the federated learning strategy with the centralized learning approach to diagnose lung cancer. Lastly, a discussion of the same comparison is made for tuberculosis detection.

We maintained the non-IID and unbalanced data qualities when we shuffled the datasets for federated learning's local models. We have selected the batch size of 64 for FL 2D sequential and FL Ensemble model.The Non-FL models, on the other hand, are chosen with a batch size of 16. However, for both the FL and Non-FL models epoch is 100. The models were run multiple times while varying various parameters, including learning rate, data preparation options, and more. For a fair comparison of FL and Non-FL learning, the 2D Sequential model and Ensemble model were fitted inside the customized key architecture. The test accuracy, precision, recall, and F1-score were used to evaluate models in order to find the best one.

5.1 FL vs Non-FL for diagnosis of Lung Cancer

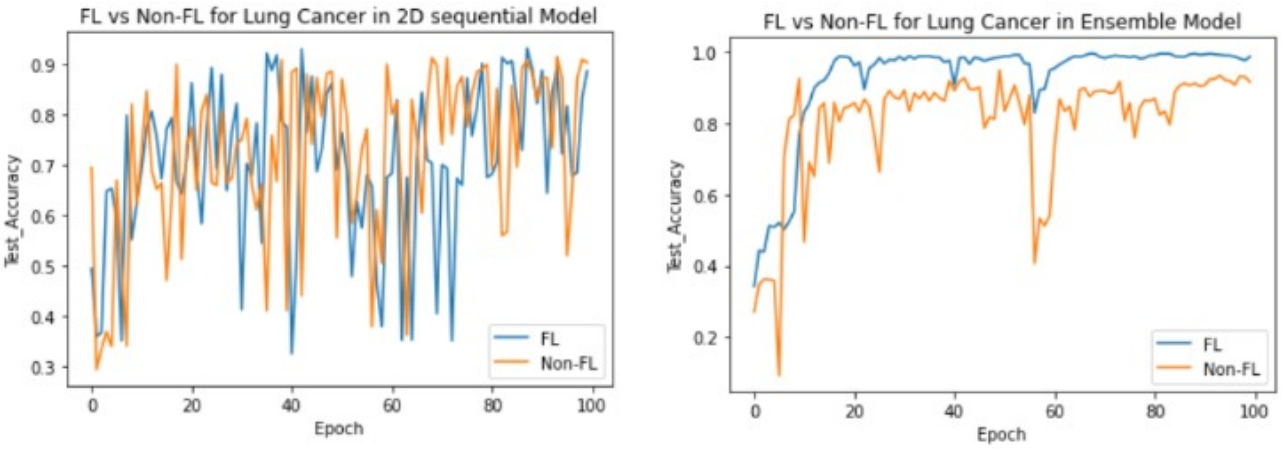

Fig. 3. Test Accuracy of Lung Cancer detection in 2D sequential Model and Ensemble Model for Federated Learning and Non Federated Learning

We have run our Lung Cancer dataset with two models under CNN architecture. The Ensemble model combines VGG19, Inception, and DenseNet models in its various layers, as opposed to the 2D Sequential Model, which featured three Convolution, Max-pooling, and Dense layers with ReLu activation function.

The test accuracy comparison for 2D sequential models between FL and Non-FL is shown in Fig. 3. With accuracy rates of 90% and 89%, it is possible to demonstrate that centralized learning is slightly more accurate than decentralized learning. Moreover, the recall score is almost 100% for FL for detecting the normal lungs.

With a 99% accuracy rate on the test data, the Ensemble Model for FL unquestionably offers the finest results. The precision, recall, and F1 scores of these models are all exceptional, as are the other scores. On the other hand, the Ensemble model for Non-FL scores 92%. Another aspect is that, for our dataset on lung cancer, the FL ensemble model outperforms the FL 2D sequential model in terms of consistency. Ensemble model FL obtains nearly the same accuracy from round 60 to round 100.

5.2 FL vs Non-FL for diagnosis of Tuberculosis

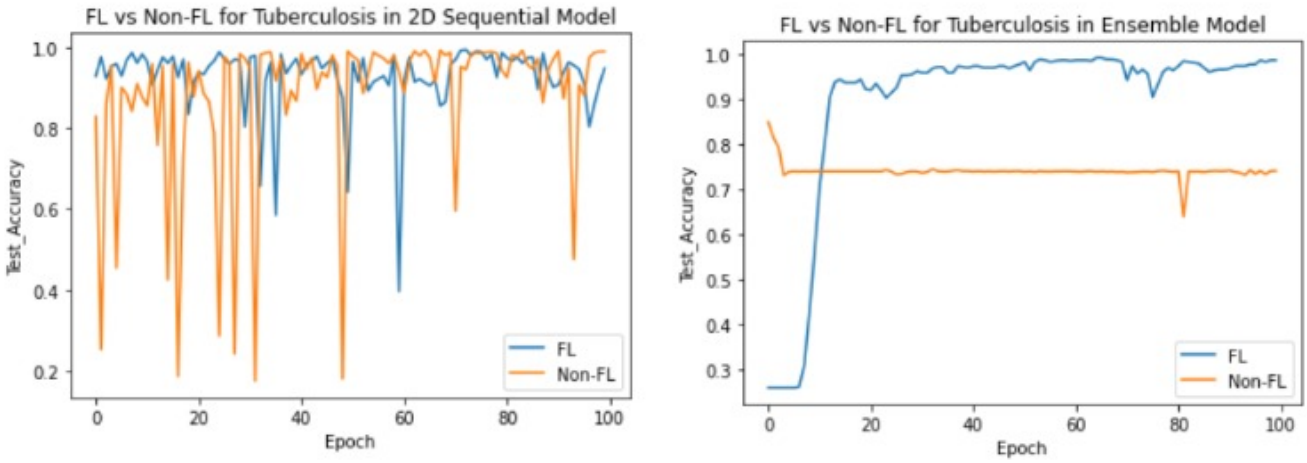

Fig. 4. Test Accuracy of Tuberculosis detection in 2D sequential Model and Ensemble Model for Federated Learning and Non Federated Learning

Fig. 4 shows that for the ensemble model for FL, it acquires the same result as for the lung cancers with 99% accuracy. There is no doubt that the combination of three different models can diagnose various diseases more accurately than the non-fl ensemble model. For example, the ensemble model for Non-FL gives nearly 74% accuracy which is the lowest among all the models. The 2D sequential model in this case gives 99% accuracy for the Non-Fl model which is higher than the FL model models.

6 CONCLUSION

In conclusion, local training of our model allows us to maintain data confidentiality, and centralized training allows us to diagnose diseases with nearly perfect accuracy. It is now possible to train data in a local model while maintaining the privacy of every patient's data, and only provide the model weight to the global model for further computation. And by doing this, we have shown that the ensemble model works better for detecting tuberculosis and lung cancer, respectively, with both 99% accuracy. As a result, we can say these models complement one another better. It can, however, also be a data loss solution. We can lose all of the data when a server crashes. But if the data is dispersed, that won't be a problem. Then, we consult the medical records from various hospitals. Nevertheless, hospital and diagnostic centers must continuously update their server and information in order for our approach to operate successfully in the field.

Finally, there are other things that may be done to enhance the model. For a better view of the image, more data pre-processing could be applied here, such as data augmentation or canny filter application. For a more

accurate result, cross validation might be performed here. We have only applied our concept to two diseases, though. Other diseases that need computer-aided diagnosis can be found using it.

REFERENCES

[1] Nikita Banerjee and Subhalaxmi Das. Prediction lung cancer– in machine learning perspective. In *2020 International Conference on Computer Science, Engineering and Applications (ICCSEA)*, pages 1–5, 2020.

[2] Deepak Rawat, Meenakshi, Lokesh Pawar, Gaurav Bathla, and Ravi Kant. Optimized deep learning model for lung cancer prediction using ann algorithm. In *2022 3rd International Conference on Electronics and Sustainable Communication Systems (ICESC)*, pages 889–894, 2022.

[3] Shimazaki A. and Ueda D. Deep learning-based algorithm for lung cancer detection on chest radiographs using the segmentation method | scientific reports. https://www.nature.com/articles/s41598-021-04667-w#citeas, 2022.

[4] A. Asuntha and Andy Srinivasan. Deep learning for lung cancer detection and classification. *Multimedia Tools and Applications*, 79, 03 2020.

[5] Pragya Chaturvedi, Anuj Jhamb, Meet Vanani, and Varsha Nemade. Prediction and classification of lung cancer using machine learning techniques. *IOP Conference Series: Materials Science and Engineering*, 1099(1):012059, mar 2021.

[6] Akshita Tiwari and Srabanti Maji. Machine learning techniques for tuberculosis prediction. *SSRN Electronic Journal*, 01 2019.

[7] Theodora S. Brisimi, Ruidi Chen, Theofanie Mela, Alex Olshevsky, Ioannis Ch. Paschalidis, and Wei Shi. Federated learning of predictive models from federated electronic health records. *International Journal of Medical Informatics*, 112:59–67, 2018.

[8] Ashish Rauniyar, Desta Haileselassie Hagos, Debesh Jha, Jan Erik Håkegård, Ulas Bagci, Danda B Rawat, and Vladimir Vlassov. Federated learning for medical applications: A taxonomy, current trends, challenges, and future research directions. *arXiv preprint arXiv:2208.03392*, 2022.

[9] Nicola Rieke, Jonny Hancox, Wenqi Li, Fausto Milletarì, Holger R. Roth, Shadi Albarqouni, Spyridon Bakas, Mathieu N. Galtier, Bennett A. Landman, Klaus Maier-Hein, Sébastien Ourselin, Micah Sheller, Ronald M. Summers, Andrew Trask, Daguang Xu, Maximilian Baust, and M. Jorge Cardoso. The future of digital health with federated learning. *npj Digital Medicine*, 3(1), sep 2020.

[10] Alexander Chowdhury, Hasan Kassem, Nicolas Padoy, Renato Umeton, and Alexandros Karargyris. *A Review of Medical Federated Learning: Applications in Oncology and Cancer Research*, pages 3–24. 07 2022.

[11] Ruchita Tekade and K. Rajeswari. Lung cancer detection and classification using deep learning. In *2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)*, pages 1–5, 2018.

[12] N Kalaivani, N Manimaran, Dr Sophia, and D Devi. Deep learning based lung cancer detection and classification. *IOP Conference Series: Materials Science and Engineering*, 994:012026, 12 2020.

[13] Murali Puttagunta and Ravi Subban. Detection of tuberculosis based on deep learning based methods. *Journal of Physics: Conference Series*, 1767:012004, 02 2021.

[14] Michael Norval, Zenghui Wang, and Yanxia Sun. Pulmonary tuberculosis detection using deep learning convolutional neural networks. In *Proceedings of the 3rd International Conference on Video and Image Processing*, ICVIP 2019, page 47–51, New York, NY, USA, 2020. Association for Computing Machinery.

[15] Eman Showkatian, Mohammad Salehi, Hamed Ghaffari, Reza Reiazi, and Nahid Sadighi. Deep learning-based automatic detection of tuberculosis disease in chest x-ray images. *Polish Journal of Radiology*, 87(1):118–124, 2022.

[16] Adam Sadilek, Luyang Liu, Dung Nguyen, Methun Kamruzzaman, Benjamin Rader, Alex Ingerman, Stefan Mellem, Peter Kairouz, Elaine O. Nsoesie, Jamie MacFarlane, Anil Vullikanti, Madhav Marathe, Paul C. Eastham, John S. Brownstein, Blaise Aguera y. Arcas, Michael Howell, and John Hernandez. Privacy-first health research with federated learning. *npj Digital Medicine*, 2021.

[17] Yogesh Kumar and Ruchika Singla. Federated learning systems for healthcare: Perspective and recent progress. 2021.

[18] Saadat Hasan Khan and Md. Golam Rabiul Alam. A federated learning approach to pneumonia detection. In *2021 International Conference on Engineering and Emerging Technologies (ICEET)*, pages 1–6, 2021.

[19] Ines Feki, Sourour Ammar, Yousri Kessentini, and Khan Muhammad. Federated learning for covid-19 screening from chest x-ray images. *Applied Soft Computing*, 106:107330, 2021.

[20] Pragati Baheti, Mukul Sikka, K. V. Arya, and Rajesh Rajagopal. Federated learning on distributed medical records for detection of lung nodules. pages 445–451, 01 2020.

[21] Lixin Liu, Kefeng Fan, and Mengzhen Yang. Federated learning: a deep learning model based on resnet18 dual path for lung nodule detection. *Multimedia Tools and Applications*, pages 1–14, 11 2022.

[22] Zezhong Ma. Frontiers | an assisted diagnosis model for cancer patients based on federated learning. https://www.frontiersin.org/articles/10.3389/fonc.2022.860532/full, 2022.

[23] M.A. Ganaie, Minghui Hu, A.K. Malik, M. Tanveer, and P.N. Suganthan. Ensemble deep learning: A review. *Engineering Applications of Artificial Intelligence*, 115:105151, 2022.

[24] Mohd Hijazi, Stefanus Hwa, Abdullah Bade, Razali Yaakob, and Mohammad Jeffree. Ensemble deep learning for tuberculosis detection using chest x-ray and canny edge detected images. *IAES International Journal of Artificial Intelligence (IJ-AI)*, 8:429, 12 2019.

[25] H. McMahan, Eider Moore, Daniel Ramage, and Blaise Agüera y Arcas. Federated learning of deep networks using model averaging. 02 2016.