sustainability

MDPI

Article

Particle Swarm-Based Federated Learning Approach for Early Detection of Forest Fires

Y. Supriya † and **Thippa Reddy Gadekallu** *,†

School of Information Technology and Engineering, Vellore Institute of Technology, Vellore 632014, India
* Correspondence: thippareddy.g@vit.ac.in
† These authors contributed equally to this work.

Abstract: Forests are a vital part of the ecological system. Forest fires are a serious issue that may cause significant loss of life and infrastructure. Forest fires may occur due to human or man-made climate effects. Numerous artificial intelligence-based strategies such as machine learning (ML) and deep learning (DL) have helped researchers to predict forest fires. However, ML and DL strategies pose some challenges such as large multidimensional data, communication lags, transmission latency, lack of processing power, and privacy concerns. Federated Learning (FL) is a recent development in ML that enables the collection and process of multidimensional, large volumes of data efficiently, which has the potential to solve the aforementioned challenges. FL can also help in identifying the trends based on the geographical locations that can help the authorities to respond faster to forest fires. However, FL algorithms send and receive large amounts of weights of the client-side trained models, and also it induces significant communication overhead. To overcome this issue, in this paper, we propose a unified framework based on FL with a particle swarm-optimization algorithm (PSO) that enables the authorities to respond faster to forest fires. The proposed PSO-enabled FL framework is evaluated by using multidimensional forest fire image data from Kaggle. In comparison to the state-of-the-art federated average model, the proposed model performed better in situations of data imbalance, incurred lower communication costs, and thus proved to be more network efficient. The results of the proposed framework have been validated and 94.47% prediction accuracy has been recorded. These results obtained by the proposed framework can serve as a useful component in the development of early warning systems for forest fires.

Keywords: Federated Learning; Federated Averaging; Particle Swarm Optimization; forest fires; disaster management

Citation: Supriya, Y.; Gadekallu, T.R. Particle Swarm-Based Federated Learning Approach for Early Detection of Forest Fires. *Sustainability* **2023**, *15*, 964. https://doi.org/10.3390/su15020964

Academic Editor: Hong Tang

Received: 21 November 2022
Revised: 21 December 2022
Accepted: 30 December 2022
Published: 5 January 2023

1. Introduction

Over the last few years, more than 3 billion people have been affected by many natural disasters such as landslides, tsunamis, earthquakes, forest fires, heat waves, cyclones, floods, and landslides, as well as various pandemics. The world is now challenged by many disasters, of which forest fires are more hazardous and cause great ecological and economic damage. Forests are a significant component of the ecological system. They play a crucial role by handing over a variety of resources such as minerals and materials that are required in the manufacturing sector. The abundant trees in forests help in absorbing carbon dioxide and releasing oxygen, which is a basic need for the survival of human beings. Furthermore, forests are a natural habitat for several animals. Forests play a key role in maintaining the ecological balance. The rich heritage of forests is majorly being destroyed by forest fires. Forest fires, also referred to as bushfires or vegetation fires, occur due to climatic changes which are caused by high emissions of greenhouse gases and global warming. Figure 1 gives a clear picture of the death rate due to natural disasters in the world, including forest fires.

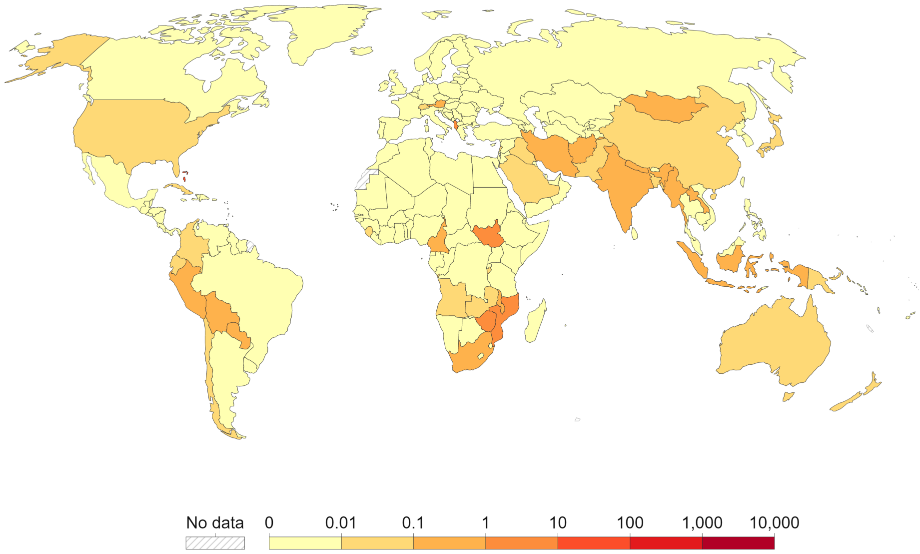

Figure 1. World death rate from natural disasters, 1990 to 2019 [1].

A number of forest fires occurred in recent times and one of the deadliest was the Australian bushfire (2019–2020) which incurred damage to the extent of 18 million hectares and 400 deaths [2]. A forest fire in the Indian state of Uttarakhand (2016) which lasted for many days, resulted in a heavy loss for the forest department [2]. The problem of heatwaves (1990 to present) [3] are on the rise in China and causing a negative impact on social, environmental [4,5], ecosystem, and health factors. The early detection and mitigation of forest fires have recently become a research hot spot for forest fire prevention authorities across the globe. Authorities all around the world are trying to protect forests from forest fires. They need to follow the steps of a forest fire management cycle in order to be informed about an unanticipated threat as discussed below:

1.1. Stages of Forest Fire Management

In a fire crisis, relief activities must be well-planned and executed in order to minimize the damage as much as possible. It is crucial to give timely, pertinent information to the concerned authorities when dealing with forest fires. A framework that enables meaningful and accurate communication is desperately needed to mitigate the problems that occur due to a forest fire [6].

Designing, implementing, and calculating strategies, policies, and measures to promote and improve preparedness for a disaster, response, and recovery procedures are known as disaster management. The disaster management cycle is a process that is followed by governmental institutions, society, and NGOs to be prepared and alert when a disaster occurs. As mentioned below, the fire executive plan involves the following steps [7]:

- Disaster mitigation
- Disaster preparedness
- Disaster response
- Disaster recovery

Disaster mitigation is generally referred to as the measures that are taken to prevent further adverse conditions that can happen after identifying the forest fire. This helps the authorities to react promptly when a forest fire recurs. Disaster preparedness includes the creation of emergency plans and early warning systems [8]. It serves as a link between disaster management and disaster risk reduction. It aids authorities in averting some of the forest fire's negative effects and recognizes the potential for ongoing, unmitigated risk. Disaster response is a group of actions taken in the wake of a disaster to determine needs, ease suffering, stop the fire's spread and effects, and pave the way for recovery [9]. Disaster recovery frequently starts while emergency response efforts are still in full swing. Communities affected by a forest fire are restored, developed, and revitalized as part of the disaster recovery process [10]. Figure 2 gives the process of the Disaster management cycle.

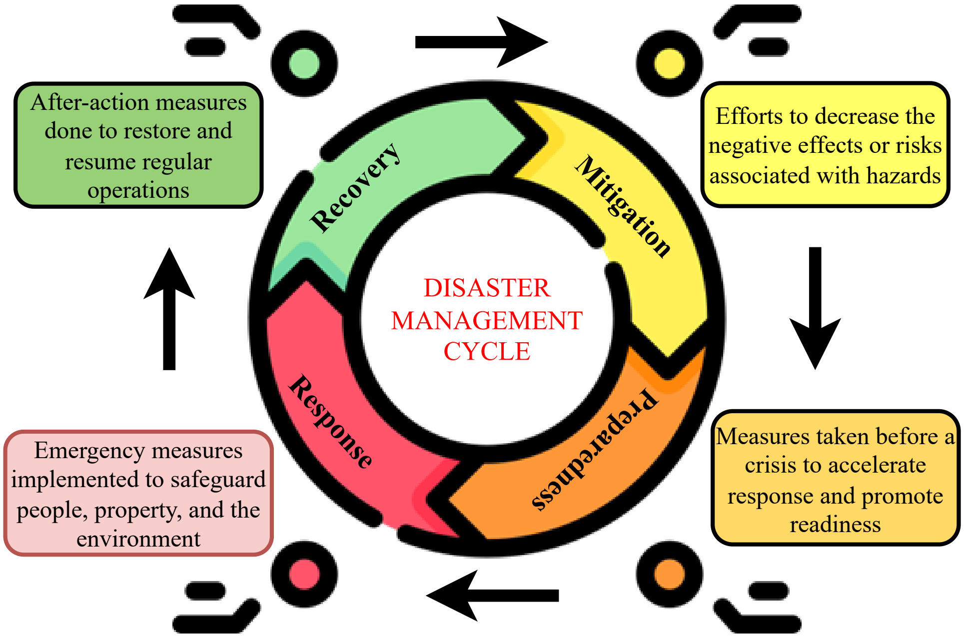

Figure 2. Disaster Management Cycle.

Around the world, different government authorities, and NGOs have been leveraging different methodologies to protect forests from fire. Currently, many kinds of sensors are used to monitor forest fires. A ZigBee-based wireless sensor is one such example [11]. A Deep Learning (DL) framework called FireNet uses separable and residual convolutional blocks used to detect active fires [12]. An asynchronous FL model is proposed in [13], which assists the edge nodes in smart remote sensing with a forest fire detection use case. The majority of current forest fire prediction systems employ a modeling technique and manually created features to make forecasts.

Recently several researchers proposed interesting solutions for the early prediction of forest fires. Energy-efficient Internet of Things (IoT) and fog-cloud computing have been used for forest fire prediction in [14]. In other interesting work a smart forest fire forecasting strategy namely a learning-based forest fire prediction system (LBFFPS) has been designed with the help of technologies such as DL, the IoT, and smart sensors to help manage the environment efficiently [15]. The authors in [16] have used a Machine Learning approach called Random Forest (RF) that maps the regional distribution of fire risk and determines how climatic and human-caused factors affect the likelihood of a fire occurring. Yolov5 and EfficientDet are used by the authors in [17] to extract the forest fire features and prove to be more efficient than regular manual feature extraction.

Recently, there has been an increased interest in applying artificial intelligence (AI) to aid disaster management [18]. ML is one such technology that has supplemented and improved existing disaster response strategies [19]. ML can handle large and complex datasets that are generated continuously when a forest fire occurs and they also assist in response and recovery activities after a forest fire occurs.

However, there are a few challenges and open issues when ML-based algorithms are used to deal with forest fires, such as:

- Cost of data collection: Network communication and storage costs for collecting and managing large amounts of original data on the server are high.

- Network Latency: The time taken for communication between the client and server in ML is more, the response time is increased and which causes delays in response to the disaster.
- Low computation capability: The processors in mobile devices do not have sufficient computing capabilities for ML.
- Security threat: Collecting or storing private data increases the likelihood of data breaches.

These challenges lead to a great loss of human life, the economy, and nature even before the authorities respond. FL is a recent technique that has the potential to solve the aforementioned issues. However, FL takes more time in communication than computation. Therefore, it is necessary to cut down on network communication time and increase the speed of network transmission to increase the effectiveness of FL. Particle Swarm Optimization (PSO) works effectively in contexts that are dynamic and heterogeneous, such as Federated Learning (FL). As a result, we suggest a new Convolutional neural network (CNN) model that incorporates FL and PSO. A PSO algorithm can set the basic parameters of the models that are trained on the local data of the clients. In short, to optimize the FL performance and make it feasible for reducing the latency in disaster response, we propose a PSO-enabled FL approach to create an effective communication strategy. The proposed PSO-enabled FL framework can help in coordinating responses to forest fires among the numerous participants.

1.2. Key Contributions

The main contributions of this paper are summarized below:

- We propose a methodology that addresses the latency in communication. Our method introduces a new paradigm of effective holistic integration of FL and PSO, bringing the widely recognized advantages of swarm intelligence to distributed learning applications.
- The experimental evaluations show that the proposed methodology outperforms the conventional FL approaches. The proposed PSO-enabled FL has an accuracy improvement.
- The findings from the extensive experimental analysis demonstrate that the proposed PSO-enabled FL outperforms better than the benchmark techniques in terms of achieving higher testing accuracy.

1.3. Paper Organization

The rest of the paper is organized in the following way. Recent works in FL and PSO are discussed in Section 2. The proposed methodology is briefed in Section 3. This is followed by Section 4, which gives a complete understanding of the results. The paper is finally concluded in Section 5.

2. Background and Related Work

In the following section, the literature survey on traditional ML approaches, FL, PSO, and natural disaster analysis are discussed.

2.1. Traditional Machine Learning Approaches

AI is the branch of science that has been used to develop intelligent machines. AI has emerged with many distinct approaches including ML, DL, neural networks (NN), and other techniques that have been applied to forest fire prediction. ML algorithms can be embedded with Wireless Sensor Networks (WSN) to prevent a false alarm [20]. The authors in [12] use a decision tree approach to detect fire events. Furthermore, WSN was used to detect forest fires in the initial stage, and furthermore to obtain accurate fire detection, regression is used. The authors in [21] used a DL-based framework to detect active forest fires using a methodology called Fire-Net. The combination of color-motion-shape features and ML is proposed for the faster detection of disaster. In another work, the authors in [22] proposed the CNN model to detect fire, which outperforms other existing methods such as smoke sensors that are installed in buildings. Unmanned Aerial Vehicles (UAV) are used to constantly patrol over potentially threatened fire areas. AI benefits UAV to use vision methods for the recognition and detection of smoke or fire based on the images. The authors in [23] provide an innovative, low-cost, machine-learning approach for forest fire prediction in Indonesia using remote sensing data. The study in [24] proposed a light-weighted CNN model which uses the three-channel color images. The study in [25] predicts quality of soil in greater depths and heterogeneous environments is evaluated using Cation exchange capacity and Closer transect spacing is needed to improve areal prediction. The study in [26] aims to predict topsoil and subsoil clay at the district scale.

2.2. Federated Learning

FL is considered the new dawn of AI. The notion of FL was initially introduced by Google in 2016 [27]. It was first applied in the Google keyboard to understand the combined data from different Android devices. FL has gained a lot of interest recently and has led to successful attempts to develop learning-based applications across distributed devices. FL permits distributed learning without the exchange of raw data thereby improving the privacy [28] of the data by keeping them on the client side [29]. In addition to this, FL also ensures the reduction of the communication cost that is incurred between the server and the client [30]. Since the data at the client are not sent to the server for training purposes, the latency in communication between the client and server is reduced. FL can also use vast amounts of data on remote devices [31]. FL is implemented in scenarios where security [32] and privacy are very important [33]. Figure 3 explains the framework of FL.

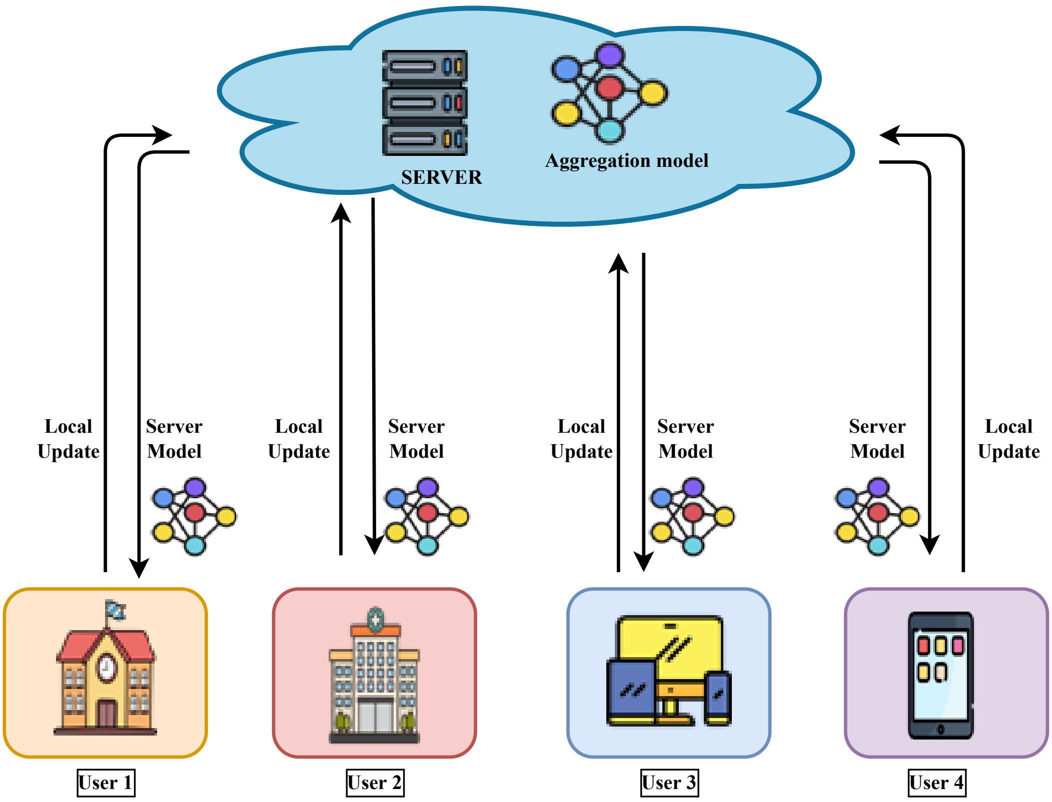

Figure 3. Federated Learning Architecture.

The following steps brief the working of FL:

1. The server creates a model based on the data available.
2. Sends a copy of the model to all the clients and the model is trained on each client based on the local data.

3. The models that are trained at clients are sent back to the server.
4. The models sent from each client are aggregated on the server side using aggregation algorithms.
5. The server sends the new updates to the client and this process repeats till the optimal model is created.

There are several FL services that perform step 4, such as FedSGD [30], Fedavg, FedPer [34], FedMA [35] etc.

However, the conventional FL methods encounter numerous difficulties, including limited capacity and resources in the IoT networks [36], high communication overhead when transferring the local updates to the server in more extensive networks [37], and occasionally being vulnerable to certain attacks such as Byzantine attacks [38,39].

2.3. Particle Swarm Optimization

The PSO algorithm is a bio-inspired algorithm that was first proposed by Kennedy and Eberhart in 1995 [40]. The behavior of a school of fish or a flock of birds mainly inspires this algorithm. It starts with a population of candidate solutions. Each particle moving with a certain velocity is considered a solution for the specified problem. Each particle in the group will update its velocity based on its and other colleagues' flying experience. Every particle in the group keeps track of its personal best (pb) and the global best (gb). Amongst the chosen pb values, the gb is selected such that it is the optimal value among the best $(gb = max_i(pb))$. The particle's position is modified based on its present position, the distance between pb and the present position, the present velocity, and the distance between the gb and the present position. It is much different from other optimization algorithms that it needs only one objective function and not anything else. The PSO algorithm has many advantages when compared to other optimization techniques. It is easy and fast in implementation, highly robust, more scalable, has early convergence, and uses simple mathematical calculations [41]. The PSO method aims at optimizing a problem iteratively. Every particle computes the speed to move on to the next step using Equation (1).

$$S_i^a = \alpha \cdot S_i^{a-1} + d_1 \cdot rd_1 \cdot (pb - Si^{a-1}) + d_2 \cdot rd_2 \cdot (gb - S_i^{a-1}) \quad (1)$$

In Equation (1), *inertia* weight is represented by the constant α, the acceleration constant for pb is represented by d_1, and the acceleration constant for gb is represented by d_2. The values of rd_1 and rd_2 are any random value between 0 and 1.

2.4. Natural Disaster Analysis

Disasters impact many people and countries in so many ways economically, and socially. The researchers, professionals, and policymakers should assess the potential impacts and how to manage the situations when disasters arise [42]. Social media has a growing importance in collecting and analyzing multimedia data related to natural disasters. The analysis of multimedia content has gained attention these days [43]. Satellite imagery also has been supporting the collection of image data related to natural disasters. Over the years many solutions have been found after analyzing natural disasters. The authors in [44] presented a detailed survey on the analysis and detection of disasters from the data collected from social media. The authors in [45] majorly summarize the seven aspects of how to identify and mitigate disasters. Moreover, the scientific challenges in handling disasters are discussed. Among all the natural disasters forest fires are very dangerous and when not detected very early can spread across a large area and cause major damage to wildlife and the country's economic status.

2.5. Related Work

Various studies are prevalent on improving the performance of FL. FL has numerous issues due to the unstable network, crashing and shifting of nodes, and increased latency when nodes increase. Sometimes the volume of data also matters as the network transmission between client and server is more. Recently, the studies in [46–48] have proposed techniques to handle the limited bandwidth bottleneck which is accomplished by joint device selection and beamforming design.

PSO has been used to address challenging optimization issues without relying on the convexity and differentiable assumptions, leveraging the swarm biological intelligence of animal flocks [49,50]. There have been a few recent attempts to apply PSO concepts to enhance ML performance. CNNs are optimized using PSO in the centralized setup for improved recognition accuracy and image classification [51–53]. Many works have been found on the integration of FL and PSO to increase the performance of FL [54,55]. FL is used in [54] for learning, whereas PSO is only used to look for the optimal hyperparameters. For the idealized distributed settings with i.i.d. data and no attacks, which cannot be guaranteed for edge IoT systems [56], PSO and FL are integrated [55] in a simplistic manner. The majority of earlier works emphasized client communication and broad optimization to boost the performance of FL. However, PSO has never been used to improve the performance of global models via network communication. Table 1 gives a complete summary of the research works on the existing systems.

Table 1. Research Works on the Existing Systems.

Ref. No	Methods	Advantages	Research Challenges
[35]	Federated matched averaging (FedMA) algorithm	FedMA builds the shared global model layer by layer based on the feature extraction signatures of hidden elements	No Privacy protection measures, data bias
[31]	Federated Optimization	Mobile Devices are built as nodes for computation	Lack of dataset, no theoretical justification
[41]	FL and Red Fox Optimization	The worker and the server operation is combined as a small operation	Parameters are chosen randomly and the execution time is extended
[57]	Aqua-Fel PSO	Detects the pollution in water and also estimates the quality of the water	Multiple water quality parameter models are not estimated
[58]	PSO + FL = PAASO	The function that is needed to be optimized by agents can be understood	Does not show good performance in heterogeneous environment
[59]	PSO and FL	PSO optimizes the eight of the clients that are sent to the aggregation	FL models are not stable and thus can be improvised further
[60]	FPSO-FS algorithm	PSO can search for optimal private subset and FL solves the privacy issues in multi-participants involvement	Execution time is very high
This Paper	PSO and FL	This unified framework will help in early detection of forest fires.	Does not analyze other nature-inspired algorithms

PSO is generally used to select an appropriate client model in every round of the global model update. PSO optimizes the weight coefficients of the clients that are involved in the aggregation [59]. The combination of PSO and FL helps us to understand the function to be optimized by a set of agents [58]. A water-monitoring system is proposed by the authors in [57], which has two phases, namely exploration and exploitation, using autonomous surface vehicles which are equipped with water quality sensors which are based on PSO and FL techniques. In other work, the authors in [60] fulfilled feature selection and privacy requirements with the combination of PSO and FL.

From the above literature survey, it is found out that PSO can solve the challenges of the FL. To this effect, we have used PSO-enabled FL to predict forest fires by training local data models based on ML on-site and then transferring those models to the central server where a global data model is developed by aggregating them. To predict forest fires, a CNN classifier has been trained on a server data model. The PSO-enabled FL framework for early forest fire prediction is presented in this research effort for the first time, to the best of our knowledge.

3. Research Methodology and Research Questions

There are three stages to this research. Phase one begins with a thorough assessment of the prior literature on the areas of ML, FL, PSO, and fire disaster management. Journal publications, conference proceedings, and news stories all contribute to the prior research. Phase two provides a detailed experimental investigation of the proposed methodology for predicting forest fires. In the last phase, the findings that were gathered from the in-depth experimental analysis in the second phase are discussed. The research's final conclusions are based on the data gathered and the discussion of various issues and potential solutions for enhancing the fire management system. To reach the research aim, the following research questions are selected.

How can PSO-enabled FL help in detecting forest fires?
Why only PSO is used to optimize FL and why not other bio-inspired algorithms?
How better is the proposed framework performing than other traditional approaches?

4. Proposed Methodology

The main objective of our work is to create a model that detects forest fires as early as possible and stops the loss that occurs due to them.

The accuracy of the CNN model is boosted because of more layers in the model. The number of variables that need training rises as the layers go deeper. When a model that has been trained on a client is transferred to the server in a normal FL system, network latency increases. So, using the characteristics of PSO to send the trained model regardless of size, we propose the PSO-enabled FL technique, which sends the best score (either accuracy or loss) to the server. We will analyze the standard FedAvg algorithm first, before the proposed methodology. FL makes use of the flow as in Algorithm 1. In line 4, the client which is taking part in the round is chosen. Through lines 5 and 6, it is possible to learn how to receive weights from different clients. Line 7 is used for weight average calculation. The global weights are then determined. After aggregation, the global model is sent to the client from the server and continues the procedure till a proper model is achieved. Figure 4 discusses the process of FedAvg.

Algorithm 1 Federated Average algorithm

1: **function** SERVERAGGREGATION(η_N)
2: initialise w_0
3: **for** every iteration $a = 1, 2, \ldots$ **do**
4: $S_a \leftarrow$ (clients are chosen randomly from set of $max(CK, 1)$)
5: **for** every client tES_a in parallel **do**
6: $w^k_{a+1} \leftarrow$ UPDATECLIENT(k, w_a)
7: $w_{a+1} \leftarrow$ average of the weights that are collected w^k_{a+1} of S_a clients
8: **function** UPDATECLIENT(k, w)
9: Carry out the process of learning on the client t that has weight w till the client arrives E epoch
10: $w \leftarrow$ (revised weight following learning)
11: **return** w to the server

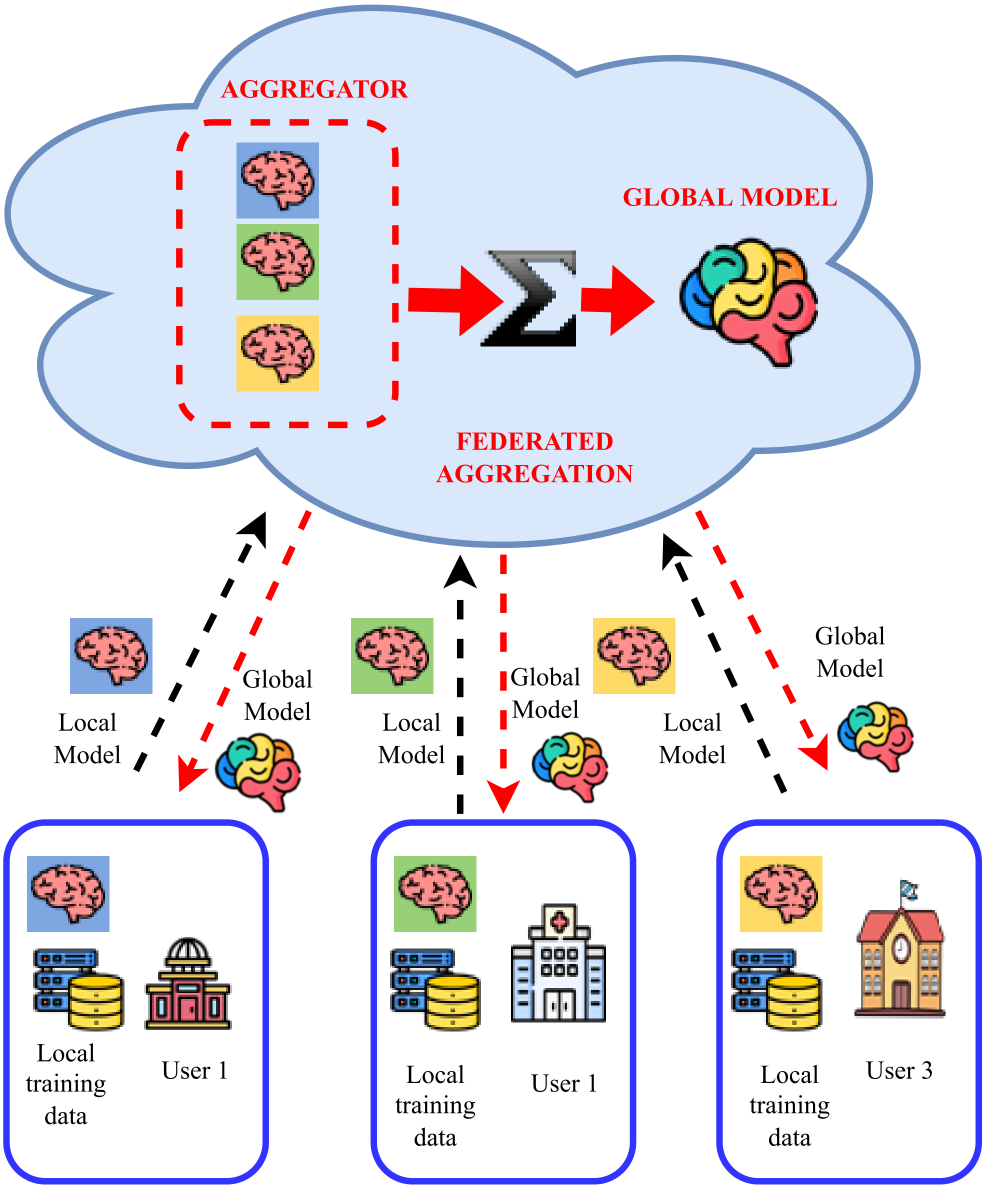

Figure 4. Federated Averaging Process.

The proposed model will receive the model weights only from the client that has provided the best weight and restricts other clients from sending their model weights to the server. The process is shown in Figure 5. The best score is chosen based on the lowest loss value that is derived after client training. Two variables, namely pb and gb, are needed to identify the best model in the PSO-enabled FL. Variable V is used to update the model in the proposed framework.

The weights in CNN were updated using Equation (1), the weight update for PSO-enabled FL can be represented as follows:

$$\begin{aligned} S_i^a &= \alpha \cdot S_i^{a-1} + d_1 \cdot rd_1 \cdot (pb - S_i^{a-1}) + d_2 \cdot rd_2 \cdot (gb - S_i^{a-1}) \\ w_i^a &= w_i^{a-1} + S^a \end{aligned} \tag{2}$$

According to Equation (2), for each layer of weight w, S in CNN has a value. S is added to the previous step weight w_{a-1} to obtain the current step weight w_a. As already discussed in Equation (1), *inertia* weight is represented by the constant α, the acceleration constant for pb is represented by d_1, and the acceleration constant for gb is represented by

d_2. The values of rd_1 and rd_2 are any random value between 0 and 1. Table 2 refers the symbols used in the paper.

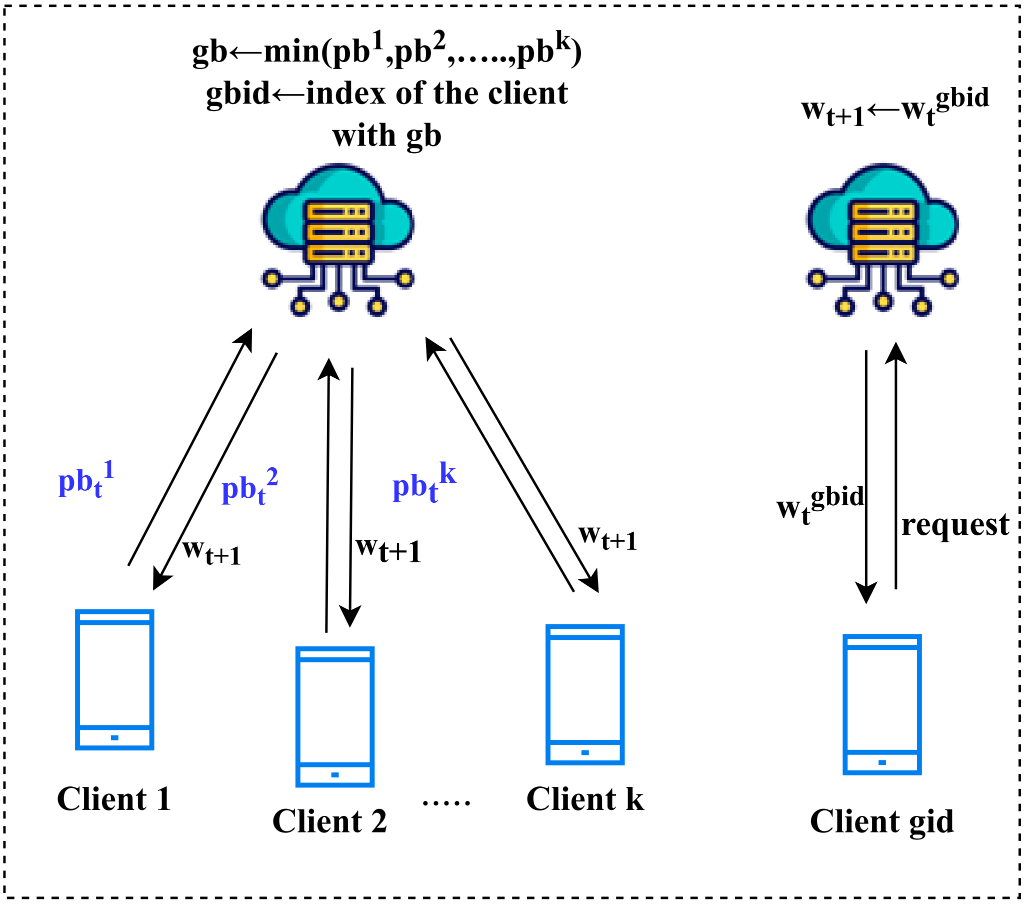

Figure 5. PSO-enabled Federated Learning.

The conceptual algorithm of PSO-enabled FL is presented in Algorithm 2. Algorithm 2 is the extension of Algorithm 1 by applying PSO. The function, ServerAggregation will receive only the *pb* values but not the weight w value from the client on Line 5. Lines 6–8 implement the search for the client with the lowest *pb* value among the data gathered. CNN then uses the UpdateClient function to apply the PSO. Variable S, which was used in the previous phase, the user's ideal w^{pb} value, and the w^{gb} value sent to the server are all calculated in lines 13–14. The process is repeated for each layer weight. In Line 15, variable V is summed up with w fetched from the previous round that is used to calculate the w that is to be used in the present round. As many times as the client epoch E, lines 16-18 are repeated. Function GetAptModel (Lines 20–23) requests the client model that has the ideal score on the server. Figure 6 gives the block diagram of PSO-enabled FL.

Table 2. Symbols used in the paper.

Symbols	Description
N	number of customers
δ	Learning rate
w^0	weights of model

Algorithm 2 PSO-enabled FL algorithm

function SERVERAGGREGATION(η_N)
  initialize w_0, $gbid$, pb, gb,
  for every iteration $a = 1, 2, \ldots$ **do**
    for every parallel client t **do**
      $pb \leftarrow$ UPDATECLIENT(t, w_t^{gbid})
      if $gb > pb$ **then**
        $gb \leftarrow pb$
        $gbid \leftarrow t$
      $w_{a+1} \leftarrow$ GETAPTMODEL$(gbid)$
function UPDATECLIENT(t, w_a^{gbid})
  initialize S, w,w^{pb}, α, d_1, d_2
  $\beta \leftarrow$ (divide p_t into batches each of size B)
  for every layer with weight $r = 1, 2, \ldots$ **do**
    $S_r \leftarrow \alpha.S_r + d_r.rd.(w^{pb} - S_r) + d_2.rd.(w_a^{gb} - S_r)$
  $w \leftarrow w + S$
  for every epoch of client i from 1 to G **do**
    for batch $b \in B$ **do**
      $w \leftarrow w - \eta(w; b)$
  return pb to the server
function GETAPTMODEL(gid)
  send a request to the client (gbid)
  receive w from Client
  return w to the server

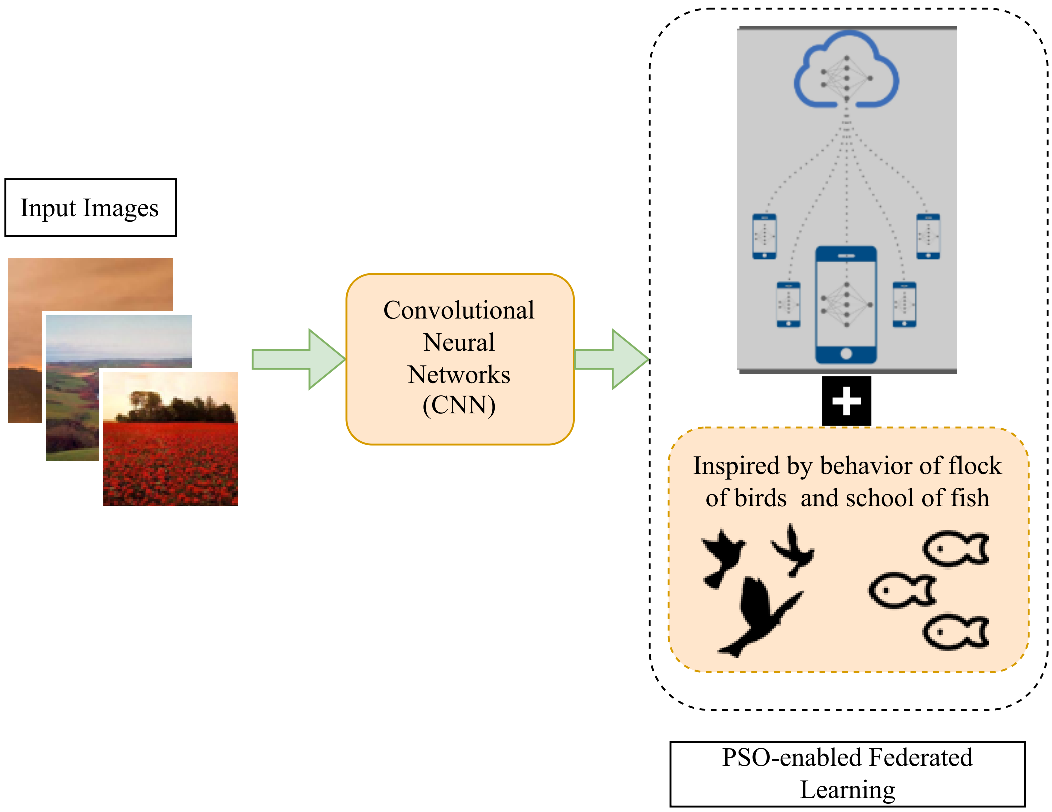

Figure 6. Block Diagram of PSO-enabled Federated Learning.

5. Experiments and Results

This section discusses the experimental analysis and discussions and future directions.

5.1. Experimental Setup

The experiments were carried out on a laptop with an Intel(R) Core(TM) i5-6200U CPU, 250 GB memory, and two NVIDIA GeForce RTX 2070 Super GPUs with 4 GB DRAM each. We used Keras version 2.4.3 and TensorFlow version 2.3.0 for simulating the FL. The proposed work enhances the communication performance of FL. A PSO algorithm is used to update the weights that are sent back from the client to the server.

5.2. Dataset Collection and Preparation

This work uses an image dataset from Kaggle https://www.kaggle.com/datasets/alik05/forest-fire-dataset (accessed on 20 November 2022). This dataset contains a variety of images from wildfires and bushfires. In total, the dataset has 1900 images related to the forest fire. Figure 7 depicts the sample images from the forest fire dataset.

Figure 7. Sample images from forest fire dataset. The images are of two classes namely fire and no-fire.

5.2.1. Dataset Pre-Processing

The dataset contains images from a variety of view angles. This kind of dataset can tune a model in distinguishing the fire and no-fire images. The dataset gives the model, the ability to identify a forest fire in two separate ways:

(1) by identifying fire flames or
(2) by detecting smoke from fire flames.

At this point, we have only taken into account these criteria and have evenly divided the number of fire (1) and no-fire (0) images in the dataset. The following categories apply to the data:

- Fire (1): Pictures of forest and mountain fires with flames and/or smoke.

- No-Fire (0): Images of the forest and mountains are taken from various angles but without any fire or smoke.

The dataset has to be cleaned up for better training and better results. As a result, we applied the required pre-processing, including trimming the images that were relevant to the specified issue, such as an image showing fire in a forest or a mountain. Following cropping, we scaled each image to a constant resolution of 150 × 150 pixels. The model was able to easily learn about forest fires because of these pre-processing procedures.

5.2.2. Dataset Partitioning

The dataset is distributed equally, with 950 photos belonging to the Fire version and the remaining 950 to the No-Fire version. We used 80% of the data for training and 20% for testing. Table 3 shows the data partitioning details.

Table 3. Dataset Division.

Dataset	Training	Testing	Total
Fire	760	190	950
No-fire	760	190	950
Total	1520	380	1900

5.3. CNN

CNN is a method for automatically learning different classification models using neural network backpropagation. Currently, deep neural networks are the predominant form. CNNs have the ability to successfully reduce network complexity, minimize the number of training parameters, and produce models that are somewhat invariant to translation, distortion, and scaling thanks to their local connection, weight sharing, and pooling operations. It is straightforward and fault-tolerant. In our experiment we used a three-layer CNN model to conduct the experiments (the first layer with 32 channels, the second layer with 64, and the third layer with 128 channels, each followed by 2 × 2 max pooling). Table 4 shows the layers of the CNN model.

Table 4. Parameters settings for CNN.

Layer	Shape
Layer 1	Conv2D(32, 3, 3) ReLU, MaxPool2D(2, 2)
Layer 2	Conv2D(64, 3, 3) ReLU, MaxPool2D(2, 2)
Layer 3	Conv2D(128, 3, 3) ReLU, MaxPool2D(2, 2)
Layer 4	Dense(512) ReLU
Layer 5	Dense(2) Sigmoid

5.3.1. Pooling

In order to further reduce the dimension of the feature image obtained from the convolution layer, the pooling layer, also known as the downsampling layer, replaces a piece of the image with a value. Popular strategies include maximum pooling and average pooling. In our proposed work, we used maximum pooling.

5.3.2. Activation Functions

Activation functions increase the nonlinearity of a neural network and are crucial for the optimization process. We employed ReLU and sigmoid activation functions in our suggested methodology. The ReLU does not saturate and is simple to compute. The ReLU discovers the intricate details of the data and returns the input and the element-wise maximum 0. The sigmoid function, often known as the logistic function, provides the output prediction probability with a value ranging from 0 to 1.

5.3.3. Optimizer

We used an adaptive moment estimation (Adam) optimizer and the model was trained using a binary cross-entropy loss function. used empirical testing to identify the ideal values for each hyper-parameter. Four different values between 0.1 and 1 were selected for the testing with learning rate. In a similar manner to this, we assessed the model using batch size 32.

5.4. Experimental Results

In this section, an overview of the experiments conducted to assess and evaluate the PSO-enabled FL architecture is presented. The performance of PSO-enabled FL on the forest fire dataset is examined in the next section, along with a comparison with the conventional FL algorithm.

5.4.1. Performance of the Proposed Approach

The model's training and performance evaluation is assessed in this subsection. The server model is trained first using the forest fire dataset samples. Then, clients are allocated the server model. In general, we use 10 clients to test the performance of the model. We selected the learning rate value as 0.0025. For each client device in the dataset, observations are chosen at random. The successful performance of the model against each round is depicted in Figure 8.

PSO-enabled FL showed a higher accuracy (94.47%) than FedAvg in all cases for 10 epochs. The number of clients that are to be used for training in FedAvg is restricted using a constant C. C is a constant between 0 and 1. In every round of communication, the experiment was conducted by choosing a client as high as C from the remaining clients. The accuracy is higher when the value of C is higher, but the data transmitted between the server and client increases accordingly. However, in the case of PSO-enabled FL, convergence occurs in fewer epochs.

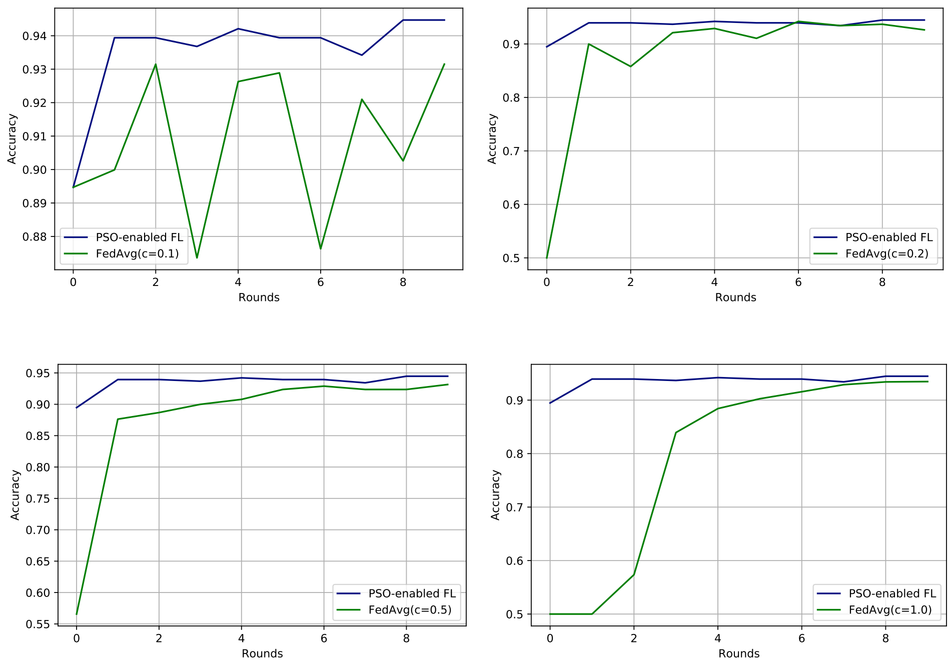

Figure 8. Accuracy results of PSO-enabled FL and FedAvg.

5.4.2. Performance Comparison with Other Models

This section compares the performance of the suggested method on our dataset of forest fires to that of other models, including FedAvg and Genetic Clustered Federated Learning (Genetic CFL) [61]. The Genetic CFL showed the lowest accuracy of the three models. The accuracy results for the three models are presented in Figure 9. The graphs are designed based on the test accuracy.

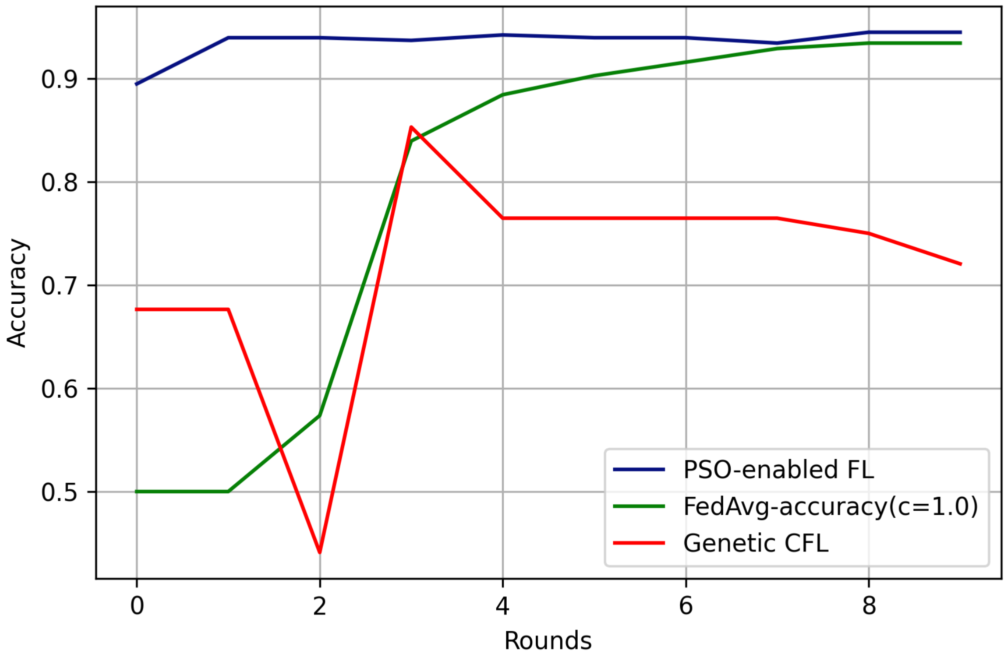

Figure 9. Comparision between the other three models.

This work also compares the performance of the proposed framework with the forest fire dataset with the outcomes of previous works which use their local datasets. We used our earlier research and classification works from 2020 and 2021 as a point of reference. The comparative analysis of the methodologies is shown in Table 5. In comparison to the earlier methods, our method correctly detected 94.47% of the forest fires in our dataset. The proposed approach to the forest fire dataset shows superiority over the other methods.

Table 5. Performance comparison of the proposed framework with previous fire detection methods.

Work	Accuracy	Dataset	Method
Proposed work	94.47%	Forest-Fire Dataset	FL and PSO
Sousa et al. [62]	93.6%	open-source dataset	Transfer Learning
Govil et al. [63]	91%	Cal Fire Dataset	Deep Learning
Sun et al. [64]	94.1%	Random Dataset	Multi-convolution kernels
Tang et al. [65]	92%	Random Dataset	Deep Learning
Lin, Chen, Li, Yu, Jia, Zhang and Liang [66]	54%	FengYun-2G S-VISSR data	Contextual based

5.5. Analysis and Discussion

With the findings depicted in the earlier subsections, the suggested model shows the potential and higher performance of the method when we have tried it with the forest fire image dataset. To evaluate the performance of the proposed PSO enabled FL, we have compared the results obtained with recent state-of-the-art. The authors in [62] proposed a wildfire prediction model that uses transfer learning. the author used an Inceptionv3 model to classify the fire and not fire data. Wildfire detection using terrestrial cameras was suggested by Govil et al. in [63]. For the purpose of detecting wildfires, their suggested technique utilized the Inceptionv3 model-based classification of the smoke and non-smoke images. The authors in [64] used a CNN model for the classification of smoke in forest fires. In [65], the authors used deep learning models such as the ForestResNet method and the ResNet50 model for fire classification. The authors in [66] used sensor data from the Visible Infrared Imaging Radiometer Suite (VIIRS) to study the detection of active fires. The aforementioned methods for classifying forest fires have proven effective in resolving the categorization issue. The biggest limitation, and a major barrier to addressing the detection of forest fires, continues to be the forest fire dataset constraint and decrease in the frequency of false alerts while also increasing the accuracy of prediction. Moreover, the above mentioned works use ML and DL algorithms where there is a need to send the data from base locations to the central server which incurs a lot of time delay in responding to the forest fires. To address the aforementioned problems, we provide a PSO-enabled forest fire detection approach for early warning to prevent significant disasters.

PSO uses a computational method that will optimize the problem iteratively by improving the candidate solution related to the weights of the FL model. PSO enables the optimization of FL by sending the best weights to the server for aggregation. Our proposed methodology, PSO-enabled FL performance is much better than the traditional FL approaches. For the forest fire dataset, our methodology is more efficient. Furthermore, it proves that it is flexible in optimizing the hyper-parameters in FL, thereby ensuring faster response to the disaster. The other techniques need a lot of effort to optimize the hyper-parameters. Our mechanism ensures that the FL performance can be increased by optimizing the weights of the client model sent to the server.

6. Conclusions and Future Directions

In this article, the PSO-enabled FL technique has been applied for forest fire prediction that allowed the clients at different locations to collectively learn a shared prediction model without transferring the training data from their origin. Forest fires create disturbances to the natural resources and environment and ecological system. Additionally, accurate and effective classification of forest fire imagery against no-fire is necessary for a reliable forest fire detection system. We used a forest fire dataset for the forest fire binary challenge. There are around 1900 multiple-colored images of which 950 belong to fire and the other 950 belong to no-fire. We have also explored the CNN algorithm for classifying forest fires. We assessed the efficiency of the FedAvg and our proposed PSO-enabled FL algorithms. The simulation results showed that the suggested approach outperformed other algorithms, obtaining a prediction accuracy of 94.47%. Overall, the performance of the suggested method on the Forest fire dataset indicated good results for classifying forest fires. As part of future work, we intend to test the proposed approach with other state-of-the-art nature-inspired algorithms such as the Firefly algorithm, Whale optimization, and Artificial Bee colony optimization to improve the FL algorithm's effectiveness in spotting disasters.

Author Contributions: Conceptualization, Y.S.; Data curation, Y.S. and T.R.G.; Formal analysis, Y.S.; Methodology, Y.S.; Resources, T.R.G.; Software, Y.S.; Supervision, T.R.G.; Validation, T.R.G.; Visualization, T.R.G.; Writing—original draft, Y.S. and T.R.G. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The Training data samples of the images are collected from https://www.kaggle.com/datasets/alik05/forest-fire-dataset (accessed on 20 November 2022).

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

FL	Federated Learning
ML	Machine Learning
IoT	Internet of Things
LBFFPS	Learning-based forest fire prediction system
CNN	Convolutional Neural Network
RF	Random Forest
PSO	Particle Swarm Optimization
DL	Deep Learning
NN	Neural Networks
UAV	Unmanned Aerial Vehicles
FedAvg	Federated Averaging Algorithm
FedSGD	Federated Stochastic Gradient Descent
FedPer	Federated Personalization Approach
FedMA	Federated Matched Averaging

References

1. Our World in Data. Research and Data to Make Progress against the World's Largest Problems. 2021. Available online: https://ourworldindata.org/ (accessed on 20 November 2022).
2. Kumar, M. 10 Natural Disasters That Have Taken Place Around The World in 2022. 2020. Available online: https://www.holidify.com/pages/natural-disasters-in-the-world-in-2020-4836.html (accessed on 20 November 2022).
3. He, B.J.; Wang, J.; Zhu, J.; Qi, J. Beating the urban heat: Situation, background, impacts and the way forward in China. *Renew. Sustain. Energy Rev.* **2022**, *161*, 112350. [CrossRef]
4. Bhattacharya, S.; Victor, N.; Chengoden, R.; Ramalingam, M.; Selvi, G.C.; Maddikunta, P.K.R.; Donta, P.K.; Dustdar, S.; Jhaveri, R.H.; Gadekallu, T.R. Blockchain for Internet of Underwater Things: State-of-the-Art, Applications, Challenges, and Future Directions. *Sustainability* **2022**, *14*, 15659. [CrossRef]
5. Pandya, S.; Gadekallu, T.R.; Maddikunta, P.K.R.; Sharma, R. A Study of the Impacts of Air Pollution on the Agricultural Community and Yield Crops (Indian Context). *Sustainability* **2022**, *14*, 13098. [CrossRef]
6. Li, Y. A Survey on Communication Networks in Emergency Warning Systems. *Sci. Comput.* **2011**, *100*, 314.
7. Alamgir, M.; Campbell, M.; Sloan, S.; Engert, J.; Word, J.; Laurance, W. Emerging challenges for sustainable development and forest conservation in Sarawak, Borneo. *PLoS ONE* **2020**, *15*, e0229614. [CrossRef]
8. Su, L.; Zhao, J.; Wang, W. Distributed Adaptive Fuzzy Modeling for Industrial Gas-Electricity Networks with Multi-Operation Modes. *IEEE Trans. Ind. Inform.* **2022**, 1–12. [CrossRef]
9. Hemmati, R.; Saboori, H.; Jirdehi, M.A. Stochastic planning and scheduling of energy storage systems for congestion management in electric power systems including renewable energy resources. *Energy* **2017**, *133*, 380–387 [CrossRef]
10. Elliott, R.T.; Fernández-Blanco, R.; Kozdras, K.; Kaplan, J.; Lockyear, B.; Zyskowski, J.; Kirschen, D.S. Sharing Energy Storage Between Transmission and Distribution. *IEEE Trans. Power Syst.* **2019**, *34*, 152–162. [CrossRef]
11. Zhang, J.; Li, W.; Han, N.; Kan, J. Forest fire detection system based on ZigBee wireless sensor network. *Front. For. China* **2008**, *3*, 369–374. [CrossRef]
12. Seydi, S.T.; Saeidi, V.; Kalantar, B.; Ueda, N.; Halin, A.A. Fire-Net: A deep learning framework for active forest fire detection. *J. Sensors* **2022**, *2022*, 8044390. [CrossRef]
13. Fadlullah, Z.M.; Kato, N. On Smart IoT Remote Sensing over Integrated Terrestrial-Aerial-Space Networks: An Asynchronous Federated Learning Approach. *IEEE Netw.* **2021**, *35*, 129–135. [CrossRef]
14. Aljumah, A. IoT-inspired Framework for Real-time Prediction of Forest Fire. *Int. J. Comput. Commun. Control* **2022**, *17*, 1–18. [CrossRef]
15. Ananthi, J.; Sengottaiyan, N.; Anbukaruppusamy, S.; Upreti, K.; Dubey, A.K. Forest fire prediction using IoT and deep learning. *Int. J. Adv. Technol. Eng. Explor.* **2022**, *9*, 246.
16. Singh, M.; Huang, Z. Analysis of Forest Fire Dynamics, Distribution and Main Drivers in the Atlantic Forest. *Sustainability* **2022**, *14*, 992. [CrossRef]
17. Qian, J.; Lin, H. A Forest Fire Identification System Based on Weighted Fusion Algorithm. *Forests* **2022**, *13*, 1301. [CrossRef]
18. Sun, W.; Bocchini, P.; Davison, B.D. Applications of artificial intelligence for disaster management. *Nat. Hazards* **2020**, *103*, 2631–2689. [CrossRef]

19. Kuglitsch, M.M.; Pelivan, I.; Ceola, S.; Menon, M.; Xoplaki, E. Facilitating adoption of AI in natural disaster management through collaboration. *Nat. Commun.* **2022**, *13*, 1579. [CrossRef]
20. Dampage, U.; Bandaranayake, L.; Wanasinghe, R.; Kottahachchi, K.; Jayasanka, B. Forest fire detection system using wireless sensor networks and machine learning. *Sci. Rep.* **2022**, *12*, 46. [CrossRef]
21. Wahyono; Harjoko, A.; Dharmawan, A.; Adhinata, F.D.; Kosala, G.; Jo, K.H. Real-Time Forest Fire Detection Framework Based on Artificial Intelligence Using Color Probability Model and Motion Feature Analysis. *Fire* **2022**, *5*, 23. [CrossRef]
22. Khan, A.; Hassan, B.; Khan, S.; Ahmed, R.; Abuassba, A. DeepFire: A Novel Dataset and Deep Transfer Learning Benchmark for Forest Fire Detection. *Mob. Inf. Syst.* **2022**, *2022*, 5358359. [CrossRef]
23. Yang, S.; Lupascu, M.; Meel, K.S. Predicting forest fire using remote sensing data and machine learning. In Proceedings of the AAAI Conference on Artificial Intelligence, Virtual, 2–9 February 2021; Volume 35, pp. 14983–14990.
24. Li, Z.; Jiang, H.; Mei, Q.; Li, Z. Forest Fire Recognition Based on Lightweight Convolutional Neural Network. *J. Internet Technol.* **2022**, *23*, 1147–1154.
25. Zhao, D.; Li, N.; Zare, E.; Wang, J.; Triantafilis, J. Mapping cation exchange capacity using a quasi-3d joint inversion of EM38 and EM31 data. *Soil Tillage Res.* **2020**, *200*, 104618. [CrossRef]
26. Zhao, D.; Wang, J.; Zhao, X.; Triantafilis, J. Clay content mapping and uncertainty estimation using weighted model averaging. *CATENA* **2022**, *209*, 105791. [CrossRef]
27. Mammen, P.M. Federated learning: Opportunities and challenges. *arXiv* **2021**, arXiv:2101.05428.
28. Kandati, D.R.; Gadekallu, T.R. Genetic Clustered Federated Learning for COVID-19 Detection. *Electronics* **2022**, *11*, 2714. [CrossRef]
29. Konečnỳ, J.; McMahan, H.B.; Yu, F.X.; Richtárik, P.; Suresh, A.T.; Bacon, D. Federated learning: Strategies for improving communication efficiency. *arXiv* **2016**, arXiv:1610.05492.
30. McMahan, B.; Moore, E.; Ramage, D.; Hampson, S.; Y Arcas, B.A. Communication-efficient learning of deep networks from decentralized data. *arXiv* **2017**, arXiv:1602.05629.
31. Konečnỳ, J.; McMahan, B.; Ramage, D. Federated optimization: Distributed optimization beyond the datacenter. *arXiv* **2015**, arXiv:1511.03575.
32. Alazab, M.; RM, S.P.; Parimala, M.; Maddikunta, P.K.R.; Gadekallu, T.R.; Pham, Q.V. Federated Learning for Cybersecurity: Concepts, Challenges, and Future Directions. *IEEE Trans. Ind. Inform.* **2021**, *18*, 3501–3509. [CrossRef]
33. Mothukuri, V.; Parizi, R.M.; Pouriyeh, S.; Huang, Y.; Dehghantanha, A.; Srivastava, G. A survey on security and privacy of federated learning. *Future Gener. Comput. Syst.* **2021**, *115*, 619–640. [CrossRef]
34. Arivazhagan, M.G.; Aggarwal, V.; Singh, A.K.; Choudhary, S. Federated learning with personalization layers. *arXiv* **2019**, arXiv:1912.00818.
35. Wang, H.; Yurochkin, M.; Sun, Y.; Papailiopoulos, D.; Khazaeni, Y. Federated learning with matched averaging. *arXiv* **2020**, arXiv:2002.06440.
36. Taheri, R.; Shojafar, M.; Alazab, M.; Tafazolli, R. FED-IIoT: A robust federated malware detection architecture in industrial IoT. *IEEE Trans. Ind. Inform.* **2020**, *17*, 8442–8452. [CrossRef]
37. Agrawal, S.; Sarkar, S.; Aouedi, O.; Yenduri, G.; Piamrat, K.; Alazab, M.; Bhattacharya, S.; Maddikunta, P.K.R.; Gadekallu, T.R. Federated learning for intrusion detection system: Concepts, challenges and future directions. *Comput. Commun.* **2022**, *195*, 346–361. [CrossRef]
38. Yang, Z.; Gang, A.; Bajwa, W.U. Adversary-resilient distributed and decentralized statistical inference and machine learning: An overview of recent advances under the Byzantine threat model. *IEEE Signal Process. Mag.* **2020**, *37*, 146–159. [CrossRef]
39. Agrawal, S.; Chowdhuri, A.; Sarkar, S.; Selvanambi, R.; Gadekallu, T.R. Temporal weighted averaging for asynchronous federated intrusion detection systems. *Comput. Intell. Neurosci.* **2021**, *2021*, 5844728. [CrossRef]
40. Wang, D.; Tan, D.; Liu, L. Particle swarm optimization algorithm: An overview. *Soft Comput.* **2018**, *22*, 387–408. [CrossRef]
41. Połap, D.; Woźniak, M. Meta-heuristic as manager in federated learning approaches for image processing purposes. *Appl. Soft Comput.* **2021**, *113*, 107872. [CrossRef]
42. Barnes, B.; Dunn, S.; Wilkinson, S. Natural hazards, disaster management and simulation: A bibliometric analysis of keyword searches. *Nat. Hazards* **2019**, *97*, 813–840. [CrossRef]
43. Gadekallu, T.R.; Pham, Q.V.; Huynh-The, T.; Bhattacharya, S.; Maddikunta, P.K.R.; Liyanage, M. Federated learning for big data: A survey on opportunities, applications, and future directions. *arXiv* **2021**, arXiv:2110.04160.
44. Said, N.; Ahmad, K.; Riegler, M.; Pogorelov, K.; Hassan, L.; Ahmad, N.; Conci, N. Natural disasters detection in social media and satellite imagery: A survey. *Multimed. Tools Appl.* **2019**, *78*, 31267–31302. [CrossRef]
45. Cui, P.; Peng, J.; Shi, P.; Tang, H.; Ouyang, C.; Zou, Q.; Liu, L.; Li, C.; Lei, Y. Scientific challenges of research on natural hazards and disaster risk. *Geogr. Sustain.* **2021**, *2*, 216–223. [CrossRef]
46. Jing, S.; Xiao, C. Federated Learning via Over-the-Air Computation with Statistical Channel State Information. *IEEE Trans. Wirel. Commun.* **2022**, *21*, 9351–9365. [CrossRef]
47. Fan, X.; Wang, Y.; Huo, Y.; Tian, Z. Joint optimization of communications and federated learning over the air. *IEEE Trans. Wirel. Commun.* **2021**, *21*, 2798–2803. [CrossRef]
48. Yang, K.; Jiang, T.; Shi, Y.; Ding, Z. Federated learning based on over-the-air computation. In Proceedings of the ICC 2019—2019 IEEE International Conference on Communications (ICC), Shanghai, China, 20–24 May 2019; pp. 1–6.

49. Eberhart, R.; Kennedy, J. A new optimizer using particle swarm theory. In Proceedings of the Sixth International Symposium on Micro Machine and Human Science, MHS'95, Nagoya, Japan, 4–6 October 1995; pp. 39–43.
50. Poli, R.; Kennedy, J.; Blackwell, T. Particle swarm optimization. *Swarm Intell.* **2007**, *1*, 33–57. [CrossRef]
51. Serizawa, T.; Fujita, H. Optimization of convolutional neural network using the linearly decreasing weight particle swarm optimization. *arXiv* **2020**, arXiv:2001.05670.
52. Junior, F.E.F.; Yen, G.G. Particle swarm optimization of deep neural networks architectures for image classification. *Swarm Evol. Comput.* **2019**, *49*, 62–74. [CrossRef]
53. Syulistyo, A.R.; Purnomo, D.M.J.; Rachmadi, M.F.; Wibowo, A. Particle swarm optimization (PSO) for training optimization on convolutional neural network (CNN). *J. Ilmu Komput. Dan Inf.* **2016**, *9*, 52–58. [CrossRef]
54. Qolomany, B.; Ahmad, K.; Al-Fuqaha, A.; Qadir, J. Particle swarm optimized federated learning for industrial IoT and smart city services. In Proceedings of the GLOBECOM 2020—2020 IEEE Global Communications Conference, Taipei, Taiwan, 7–11 December 2020; pp. 1–6.
55. Park, S.; Suh, Y.; Lee, J. FedPSO: Federated learning using particle swarm optimization to reduce communication costs. *Sensors* **2021**, *21*, 600. [CrossRef]
56. Yarradoddi, S.; Gadekallu, T.R. Federated Learning Role in Big Data, Jot Services and Applications Security, Privacy and Trust in Jot A Survey. In *Trust, Security and Privacy for Big Data*; CRC Press: Boca Raton, FL, USA, 2022; pp. 28–49.
57. Kathen, M.J.T.; Johnson, P.; Flores, I.J.; Reina, D.G.E. AquaFeL-PSO: A Monitoring System for Water Resources using Autonomous Surface Vehicles based on Multimodal PSO and Federated Learning. *arXiv* **2022**, arXiv:2211.15217.
58. Torra, V.; Galván, E.; Navarro-Arribas, G. PSO + FL = PAASO: Particle swarm optimization + federated learning = privacy-aware agent swarm optimization. *Int. J. Inf. Secur.* **2022**, *21*, 1–11. [CrossRef]
59. Li, Y.; Chen, Y.; Zhu, K.; Bai, C.; Zhang, J. An Effective Federated Learning Verification Strategy and Its Applications for Fault Diagnosis in Industrial IoT Systems. *IEEE Internet Things J.* **2022**, *9*, 16835–16849. [CrossRef]
60. Hu, Y.; Zhang, Y.; Gao, X.; Gong, D.; Song, X.; Guo, Y.; Wang, J. A federated feature selection algorithm based on particle swarm optimization under privacy protection. *Knowl.-Based Syst.* **2022**, *260*, 110122. [CrossRef]
61. Agrawal, S.; Sarkar, S.; Alazab, M.; Maddikunta, P.K.R.; Gadekallu, T.R.; Pham, Q.V. Genetic CFL: Hyperparameter optimization in clustered federated learning. *Comput. Intell. Neurosci.* **2021**, *2021*, 7156420. [CrossRef]
62. Sousa, M.J.; Moutinho, A.; Almeida, M. Wildfire detection using transfer learning on augmented datasets. *Expert Syst. Appl.* **2020**, *142*, 112975. [CrossRef]
63. Govil, K.; Welch, M.L.; Ball, J.T.; Pennypacker, C.R. Preliminary Results from a Wildfire Detection System Using Deep Learning on Remote Camera Images. *Remote Sens.* **2020**, *12*, 166. [CrossRef]
64. Sun, X.; Sun, L.; Huang, Y. Forest fire smoke recognition based on convolutional neural network. *J. For. Res.* **2021**, *32*, 1921–1927. [CrossRef]
65. Tang, Y.; Feng, H.; Chen, J.; Chen, Y. ForestResNet: A deep learning algorithm for forest image classification. In *Journal of Physics: Conference Series*; IOP Publishing: Changsha, China, 2021; p. 012053.
66. Schroeder, W.; Oliva, P.; Giglio, L.; Csiszar, I.A. The New VIIRS 375 m active fire detection data product: Algorithm description and initial assessment. *Remote Sens. Environ.* **2014**, *143*, 85–96. [CrossRef]