Abstract

Distributed deep learning algorithms have shown eminent performance in learning from data that are privately allocated between several agents. Recent advances in sensor technology have enabled the cheap collection of spatial and temporal high-resolution data for agriculture across a wide geographical area. This continuous increase in the amount of data collected has created both the opportunity for, as well as the need to deploy distributed deep learning algorithms for a wide variety of decision support tasks in agriculture. Distributed deep learning algorithms are typically divided into two major categories: centralized vs decentralized learning algorithms, depending on whether a central parameter server exists for gathering information from participating agents. In the case of rural agriculture applications, transferring a large amount of high-resolution data (e.g., images, videos) collected with IoT devices to a central server/cloud could be very expensive especially with limited communication infrastructure. This suggests the need for decentralized learning approaches, which also naturally provide some measure of privacy. Here, autoencoders are trained using a decentralized optimization algorithm to create a latent representation of growing maize plants in a large-scale field experiment involving several hundred cameras deployed in a maize genome diversity growth experiment. We trained the autoencoders for different communication network topologies of the field-deployed cameras. The feature representations from these autoencoders are then utilized to solve downstream tasks such as anomaly detection and image retrieval. Experimental results show that distributed deep learning is effective in learning from large datasets distributed among several learning agents associated with different cameras. Anomaly detection in particular was useful to make course corrections in imaging protocol and identify localized crop management.

Introduction

Recent dramatic advances in sensor technology have enabled remote (drone, ground) as well as proximal (soil sensors, touch sensors) data acquisition that can target a large range of features (coarse vs. fine spatial resolution, high vs. low temporal frequency, visible vs. hyperspectral wavelengths, chemical vs. physiological attributes) for different decision support tasks in agriculture. Distributed AI-enabled advances are helpful to extract informative agronomic and physiological traits from the large amounts of raw sensor data collected from agricultural fields. Typically distributed deep learning algorithms are divided into two main categories; centralized McMahan et al. [2017], Kairouz et al. [2019] and decentralized Lian et al. [2017], Nedić et al. [2018] learning depending on whether a central parameter server is participating in the learning process Tang et al. [2020]. Centralized learning refers to the class of algorithms which generally contain a parameter server that aggregates information i.e. gradients, model parameters, etc. from the participating agents and performs parameter updates. On the other hand, in decentralized learning, agents communicate...
based on some graph topologies to update parameters \cite{Lyu2021}. However, in agriculture applications, transferring a large amount of data (e.g., images, videos) which are collected with IoT devices to a central server/cloud is computationally expensive. Additionally, the central parameter server can act as a single point of failure. Therefore, decentralized deep learning algorithms \cite{Kempe2003,Xiao2004,Boyd2006,Dekel2012,Lian2017,Jiang2017,Yu2019,Koloskova2019,Nadiradze2019,Balu2021,Esfandiari2021} appear to act as a better alternative. As a result, we consider \textit{DPMSGD} \cite{Lian2017} which is a decentralized learning algorithm as the method to train deep learning models in a distributed fashion.

Identification of inputs that lie far away from the training distribution (in-distribution) is called Out-of-distribution (OOD) detection, anomaly detection or outlier detection \cite{Grubbs1969}. OOD detection has been used in broad range of safety-critical applications including medical diagnosis \cite{Caruana2015}, autonomous driving \cite{Evtimov2017}, cyber-security \cite{Kruegel2003}, and biometric authentication \cite{Gunther2017}. In this paper, we deal with the challenging scenario of fully unsupervised outlier detection where the goal is to detect outliers from data containing both normal and outlier patterns. We are using \textit{LOF} \cite{Breunig2000} and cluster-conditioned detection \cite{Sehwag2021} methods for performing anomaly detection using the autoencoders which are trained in a distributed fashion. Content-based image retrieval (CBIR) technique \cite{Knorr2000} is the computer vision based process of retrieving images that are similar to visual content of a query image from an extensive archive. In the case of anomaly detection, the query image would be an anomaly example. Early research for CBIR considered the global (shape, color, and texture) and local descriptors \cite{Mikolajczyk2005,Arampatzis2013} of an image as a feature vector to perform the retrieval procedure \cite{Flickner1995,Huang2001}. Recently, representations from deep learning models have been highly efficient for image retrieval tasks \cite{Chen2021}. In this work, we use the nearest neighbors algorithm on the features from the CNN model for the image retrieval task.

\textbf{Contributions:} Specifically, in this paper (i) the use of distributed learning algorithms to train models from real-world agricultural datasets is investigated. (ii) the efficacy of the autoencoder models are shown, (iii) downstream tasks such as anomaly detection, and image retrieval using these models are performed followed by the discussion on the results and future work.

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{ workflow.png}
\caption{Workflow of the proposed method}
\end{figure}

\textbf{Framework}

Figure 1 shows the different algorithms used in the training and inference pipeline. \textbf{Distributed Deep Learning:} As discussed before, there are several methods for decentralized distributed learning among them \textit{DPMSGD} \cite{Lian2017} has shown to produce prominent results in learning from IID data distributions. In this algorithm, each agent is assigned a model for its portion of the data set. After each agent has computed a local stochastic gradient, it then fetches the optimization variables from neighbors and calculates the neighborhood weighted average. The local optimization variables are then updated using the neighborhood weighted average and the gradients. The consensus model is then achieved by averaging all the local model parameters. \textbf{Anomaly Detection: Local Outlier Factor (LOF)} is a density-based unsupervised outlier detection algorithm that detects outliers by calculating the local deviation of a given data point \cite{Breunig2000}. The points that contains substantially lower density than its neighbors are considered outliers. We used euclidean distance for
identifying the nearest neighbors and used 20 nearest neighbor points to calculate the density of a data point. **Cluster-conditioned detection** [Sehwag et al. (2021)] method uses the feature representations learnt by the autoencoder for the outlier detection. We assume that majority of the samples in the data are inliers. Then we calculate the centroid of the feature representations of the complete data to represent the inlier data. Then, we use the Mahalanobis [Mahalanobis (1936)] distance metric to identify the outlier data points that are farthest away from the centroid feature representation. **Image Retrieval:** In the first step, feature maps of the dataset and query image are extracted using the trained CNN model. Then, we use the ball tree method [Andoni and Indyk (2017)] based nearest neighbors search on the feature maps to identify the \( k \) nearest neighbors of the query feature.

**Experimental Setup and Results**

We empirically evaluate the effectiveness of our framework by using the trained model to perform downstream tasks such as anomaly detection and image retrieval on real-world agricultural dataset. To explore the algorithm performance under different topologies, the experiments are performed using 10 learning agents. The agricultural dataset consists of images of 655 rows of field-grown Maize plants. Figure 2a shows the imaging setup of the field-deployed stationary camera. The images were collected at an interval of 20 minutes from 8:00 AM to 5:00 PM over a period of two and half months in 2019 in the midwestern United States. Each row consists of 6 maize plants of a specific genotype. These field photos of maize plants are high-resolution \( 5152 \times 3864 \) RGB images. A stationary camera was used to collect data from each row separately and a total of 655 cameras were used. In our experiments, data from a diverse subset of 30 cameras are used to create a dataset of 74,000 images. The images were resized to 128 \( \times \) 128 pixels each for computational efficiency and the data collection process is described thoroughly elsewhere. We use a deep convolutional neural network (CNN) autoencoder (with 3 convolutional layers with 12, 24, 48 filters in the encoder section and 3 layers with 24, 12, 3 filters for the decoder, ReLU activation is used in convolutional layers). A mini-batch size of 128 is used, the initial step-size is set to 0.01. The step size is decayed with a constant 0.981. The stopping criterion is a fixed number of epochs and the momentum parameter (\( \beta \)) is set to be 0.98. We start our analysis in learning real-world agricultural datasets. In these experiments, data from 30 cameras are divided between 10 learning agents so each agent has access to data from 3 distinct cameras. Figure 2b shows that DPMSGD is converging for the agricultural dataset for all the three graph topologies. We then compare the original images in the dataset with their reconstructed counterparts to further analyze the model. Figure 3 shows that the autoencoder is moderately effective in reconstructing the images.

![Figure 2: a) Illustration of imaging setup used for data collection from a single row of infield Maize plants b) Average training loss for DPMSGD method on agricultural dataset with 10 training agents](image)

The trained autoencoder model is then used to perform anomaly detection and image retrieval tasks. Figure 4 shows the anomaly predictions of cluster-conditioned detection and LOF methods. Identification of anomalies was particularly useful in making course corrections to imaging protocols of field-deployed stationary cameras. We then use the same model to perform image retrieval. Figure 5 shows some examples of image retrieval results by the nearest neighbors algorithm. These results show that the models trained in decentralized distributed fashion are effective to do image retrieval tasks on diverse images (normal weather, rainy weather, or from tilted cameras).
Conclusion

In this paper, the viability of distributed learning to train autoencoders which learn from real-world agricultural datasets gathered from field-deployed stationary cameras was discussed. Experiments conducted on this dataset showed that our proposed framework is effective to perform anomaly detection and image retrieval tasks. Future work includes: comparing different distributed learning algorithms, proposing graph topologies to optimize the communications between the cameras, and conducting experiments using larger datasets.
References


