# Label Hungry? Not Anymore: Towards Federated Semi-supervised Learning

Team: Deepthi Hegde, Ishna Kaul, Geoffrey Li, Roshan Ramkeesoon
Google mentors: Peter Kairouz, Boqing Gong, Zachary Charles

## Abstract

Federated learning is a promising paradigm to train models on privacy sensitive data, without the data ever leaving the device. Several recent efforts have focused on commensurating the performance of Federated Learning algorithms with their centralized equivalents for a wide range of applications. However, deploying federated algorithms for image based applications largely remains an unsolved problem due to the lack of large-scale labelled training data on client devices. To this end, we propose a semi-supervised federated learning framework that leverages unlabelled device-data to train better image classification models. Through experiments on federated EMNIST dataset, we show that semi-supervised methods outperform fully-supervised methods by large margins, especially in the low supervision regime.

## Introduction

As mobile device usage increases, much of the world's richest data will be generated on user devices. While this data can be used to train intelligent models that deliver value to people, usage of on-device data suffers from a number of issues including user privacy and the lack of ground-truth annotations. Federated learning, a distributed optimization technique in which data remains on client devices, can provide value in this domain, allowing us to build increasingly intelligent tools while preserving user privacy. However, the lack of labels remains a challenge in fully benefiting from on-device data.

This work is an exploration of formulations that enable leveraging large image data sources from devices in a federated, privacy-preserving way. We propose two frameworks to train Federated Learning (FL) models in a semi-supervised way where devices contain primarily unlabeled data, and access to a small pool of labels simulating user generated labels: (1) First stage involves pre-training a local model on a self-supervised [7] proxy task using unlabelled data followed by Federated Averaging[1] [6] to obtain the global model. The second stage comprises fine-tuning on the downstream task locally followed by Federated Averaging to obtain a task specific global model. We call this method Self-supervised Pre-Trained Federated Averaging (SSPT-FedAvg) (2) Jointly optimized, two headed network trained locally; one head trained in a self-supervised fashion on an proxytask using all of the available data, the other head trained on a downstream task using only the labelled data. We

[1] FederatedAveraging is an algorithm, which combines local stochastic gradient descent (SGD) on each client with a server that performs model averaging.

call the resulting method Semi-Supervised Jointly Optimized Federated Averaging (SSJO-FedAvg).

# Related Work

**Federated Learning**
Federated Learning (FL) [6] is a distributed machine learning approach which enables training on a large corpus of decentralized data residing on client devices, such as mobile phones. When combined with techniques from differential privacy and secure aggregation, it has the potential to learn from large amounts of sensitive data generated by users without impeding on user privacy. A popular implementation of federated learning involves training for a number of epochs on replicas of a model distributed over client devices, which then send the parameter updates to a centralized server that applies the updates using a technique such as federated averaging. While federated learning has already been deployed in applications, such as next word prediction in Gboard (Google's keyboard software), there are many areas of active research. Challenges in federated settings include the non-IID nature of client data as well as the lack of user labels.

**Representation Learning**
In recent years, the deep learning community has seen a surge towards finding alternatives to strong supervision during training, owing largely to the vast amount of cheap, unlabelled data available today. Self-supervised learning, a form of unsupervised learning where supervision is derived from the data itself, has narrowed the performance gap between fully-supervised and unsupervised methods. Many clever methods [1][2][3] have been devised by researchers that have allowed for representation learning on unlabelled datasets using proxy tasks. One simple method is to use an autoencoder network, which attempts to reconstruct the input via a learned embedding. We wish to adopt this method in our formulation for the self-supervised part of our network.

While we evaluate our approach using image reconstruction as the proxy task, this can be swapped out for any other representation learning task. In fact, the nature and choice of the proxy task may strongly influence the model performance depending on the dataset of interest and compatibility of the supervised and self-supervised tasks.

# Data Summary

The EMNIST (Extended MNIST) dataset is a federated dataset comprising handwritten digits derived from the original NIST Database. It has been converted to a 28x28 pixel image format and has 341,873 images distributed across 3,383 users (clients) over 10 classes: digits 0 to 9. The data is keyed by the original writer of the digits. Since each writer has a unique style, this dataset exhibits the kind of non-i.i.d. behavior expected of federated datasets.

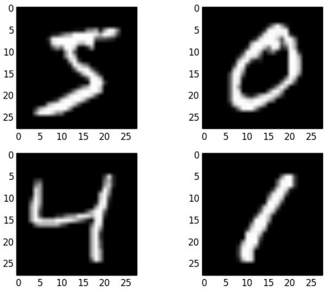

Figure 1. Handwritten digits demonstrate a variety of styles.

The number of images per client vary from a minimum of 9 to a maximum of 132. The mode of this distribution of images is 114 which belongs to 300 clients.

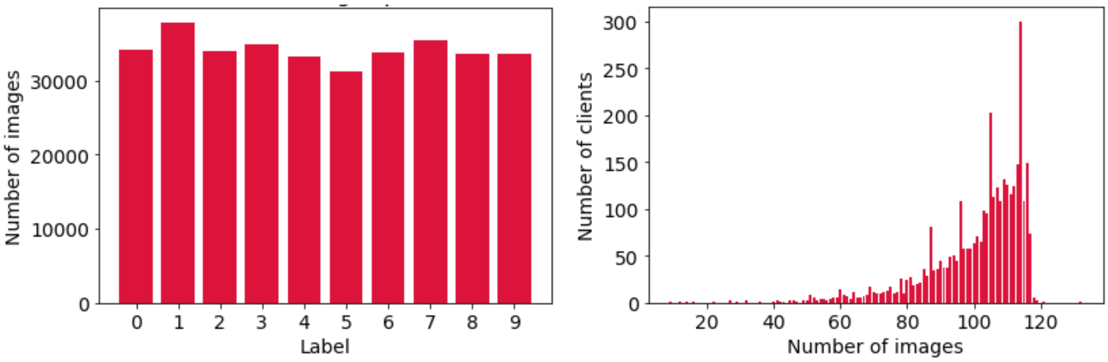

Figure 2. (left) Distribution of image classes in the dataset, (right) distribution of images over clients

# Methods

The Semi-Supervised Federated Learning (SSFL) approaches detailed in this section are designed to learn task specific encodings from examples that have ground-truth labels while still extracting generic image representations from examples that do not have labels. This allows the models to leverage large amounts of unlabelled image data sitting on the device end. An overview of our two approaches, SSPT-FedAvg and SSJO-FedAvg are shown in Fig. 3 and Fig. 4 respectively.

## 1. **Self-supervised Pre-trained Federated Averaging (SSPT-FedAvg)**

SSPT-FedAvg is a two stage process wherein the first stage of learning involves self-supervised representation learning on the client-side using autoencoder reconstruction loss. This is followed by Federated Averaging to update the global server model using local gradients from the clients. The process continues until convergence while randomly sampling a fixed number of clients from the pool of clients, to participate in each round. The second stage involves supervised training on the target task, which in our case is image classification. The local and global updates are similar to the first stage. While we evaluate our approach using reconstruction as the proxy task in this work, this can be swapped out for any other representation learning task.

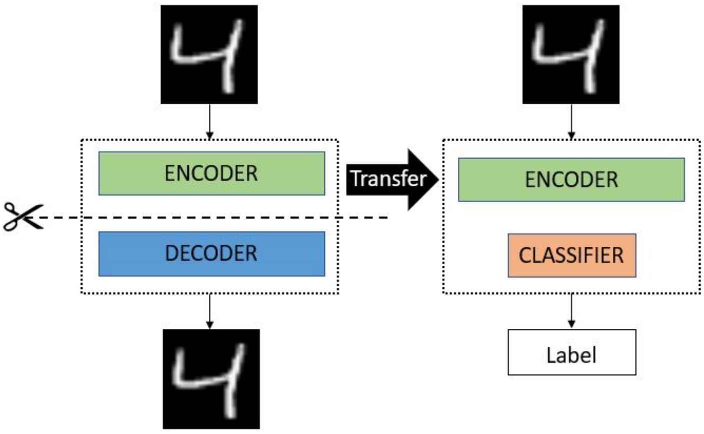

Figure 3. Depicts the local training involved in SSPT-FedAvg. Shown on the left is the pretrained model on self-supervised reconstruction. The ENCODER weights are transferred to the classification model and fine-tuned.

## 2. **Semi-supervised joint optimized Federated Learning (SSJO-FedAvg)**

SSJO-FedAvg is an end-to-end learning method that comprises a two headed network - the supervised head and the self-supervised head. The self-supervised head is trained on image reconstruction. Loss for this head is computed over all the images in the local dataset. The supervised head is trained on image classification and the loss is computed only on the images that have ground-truth labels. The two heads are trained jointly by randomly sampling batches of labelled or unlabelled data in each round on the client side. A fixed number of clients participate in each round and at the end of the round, the updates from the clients is aggregated using Federated Averaging.

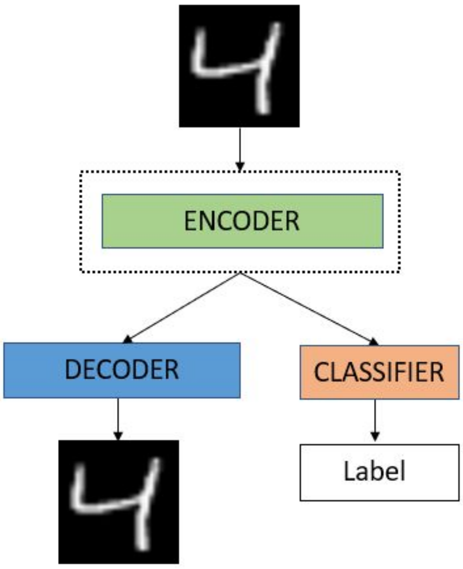

Figure 4. Depicts the local training phase of SSJO-FedAvg. Shown is the shared network(ENCODER) and the two heads from which gradients are propagated. For unlabelled data, gradients are only accumulated at the self-supervised reconstruction head(DECODER). For labelled data, gradients are computed at both DECODER and CLASSIFIER heads.

## Experimental Setup

Our experiments evaluate (1) the degradation in performance as device mask-ratio varies (2) the number of communication rounds required to attain 50% test accuracy and (3) the relationship of test performance with communication rounds. The evaluations are two fold: we compare test performances of our two proposed methods to the fully supervised baseline under both central and federated settings[2].

**Fully Supervised Learning Baseline (FSL)**

- Central setting: We train a two layer dense model on the dataset in a central server method
- Federated setting: Same network architecture but trained in a federated setting, where we treat client IDs as simulated user devices.

**Transfer Learning Model (SSPT)**

- We first train a self-supervised autoencoder model with 2 dense layers. We use the learned weights from the first encoding layer to transfer to the supervised classifiers. We train in the following settings:
    - Central self-supervised pre-training + central supervised training
        - Baseline
    - Federated self-supervised pre-training + federated supervised training
        - Experimental

**Semi Supervised Joint Model (SSJO)**

- **Central setting:** We jointly train a 2 headed network - a supervised classification head and an unsupervised reconstruction head. This is done by aggregating the data from all clients onto the central server. We randomly sample batches to contain either all supervised or unsupervised examples in each round.
- **Federated setting:** Everything else remains the same, except there is no data aggregation and the model is trained in a decentralized fashion

We train our three classifiers on the EMNIST dataset while varying the ratio of examples with masked labels. The mask ratios we use are 0.0, 0.8, 0.9, 0.95, 0.98, and 0.99 which correspond to a minimum of 3,414 labeled training examples.

[2] Source code available at https://github.com/r-o-s-h-a-n/semisupervisedFL.

| **Hyperparameter** | **Central** | **Federated** |
|---|---|---|
| Batch Size | 20 | 20 |
| Learning Rate | 1e-4 | 1e-3 |
| Momentum | 0.99 | 0.99 |
| *Federated only:* | | |
| Num epochs/client | - | 10 |
| Num clients/round | - | 100 |
| Num communication rounds | - | 100 |
| *Central Only:* | | |
| Num epochs | 10 epochs | - |

Chart 1. Hyperparameters used for our experiments were chosen after running a few initial experiments on EMNIST dataset.

# Experimental Results

## Test accuracy

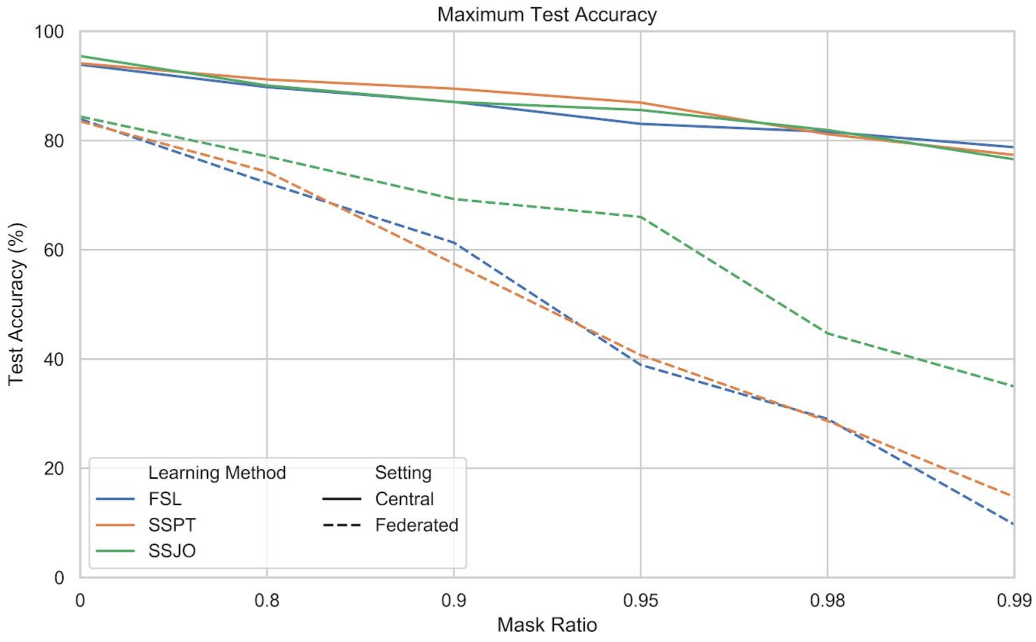

Figure 5. Maximum test accuracy achieved at various mask ratios of the EMNIST dataset. Note: the horizontal axis is not to scale.

Maximum Test Accuracy

| | Central | | | Federated | | |
|---|---|---|---|---|---|---|
| **Mask Ratio** | **FSL** | **SSPT** | **SSJO** | **FSL-FedAvg** | **SSPT-FedAvg** | **SSJO-FedAvg** |
| 0 | 93.90 | 94.14 | **95.46** | 83.94 | 83.52 | **84.39** |
| 0.8 | 89.78 | **91.18** | 90.08 | 72.24 | 74.30 | **77.10** |
| 0.9 | 87.07 | **89.50** | 87.06 | 61.31 | 57.46 | **69.29** |
| 0.95 | 83.05 | **86.95** | 85.59 | 38.93 | 40.73 | **66.02** |
| 0.98 | 81.60 | 81.16 | **81.92** | 29.04 | 28.66 | **44.70** |
| 0.99 | **78.77** | 77.36 | 76.55 | 9.70 | 14.76 | **34.98** |

Chart 2. Maximum achievable test accuracy at various mask ratios of the EMNIST dataset.

Centrally trained models obtain higher test accuracy than their federated counterparts. Among the federated models, we observe that the federated semi-supervised learning methods achieve higher test accuracy than the federated fully supervised learning method in the high mask ratio regime. Further experimentation with hyperparameters, especially for the federated models, may result in higher test accuracies. In addition, running the federated experiments for a greater number of communication rounds may result in higher test accuracies.

Notably, in the high mask ratio regime, we see the federated SSJO-FedAvg method achieving higher test accuracy than the federated SSPT-FedAvg method. Due to the similarity between the SSPT-FedAvg and fully supervised results, we believe the autoencoder task may not be a helpful pre-training task for digit classification. In addition, in the joint optimization method, further research is required to understand the role the autoencoder proxy task plays in training. It may be the case that the autoencoder proxy task provides noise to the encoder which may provide helpful regularization.

We note that the centrally trained models all achieve similar test accuracy with the centrally trained semi-supervised methods achieving slightly higher test accuracy. The similarity in the results is likely due to the saturation of the EMNIST dataset at the given mask ratio with the given model architecture.

### Number of communication rounds

In addition to improving the test accuracy of our image classifier, a related goal of ours is to use semi-supervised learning to allow federated models to converge faster and reduce the number of communication rounds required to achieve a target test accuracy.

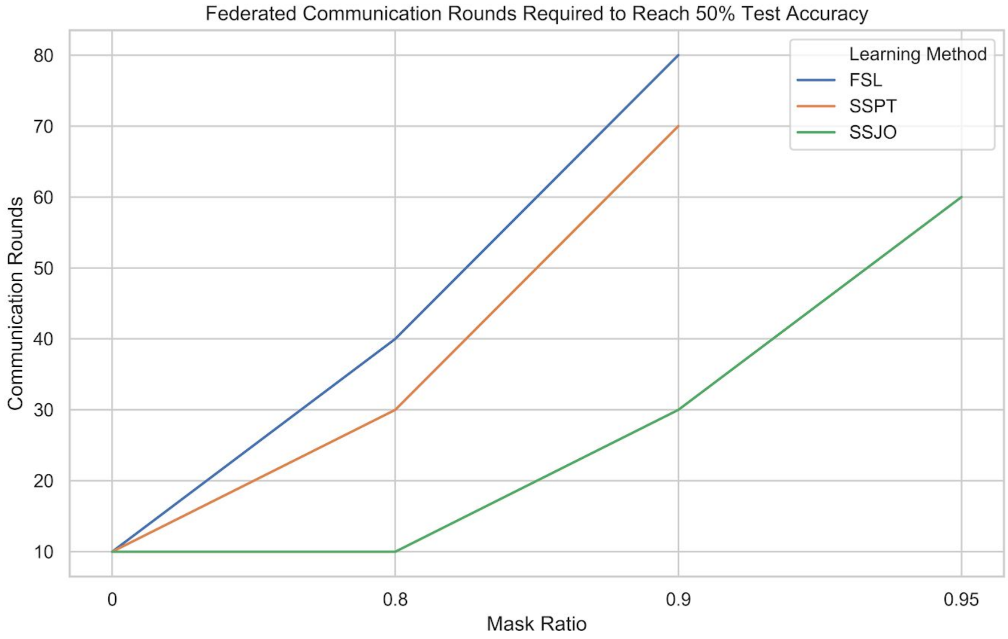

Figure 6. Number of communication rounds to achieve 50% test accuracy at various mask ratios on the EMNIST dataset. Note: the horizontal axis is not to scale.

### Federated Communication Rounds Required to Reach 50% Test Accuracy

| Mask Ratio | FSL-FedAvg | SSPT-FedAvg | SSJO-FedAvg |
|---|---|---|---|
| 0 | 10 | 10 | 10 |
| 0.8 | 40 | 30 | 10 |
| 0.9 | 80 | 70 | 30 |
| 0.95 | N/A | N/A | 60 |
| 0.98 | N/A | N/A | N/A |
| 0.99 | N/A | N/A | N/A |

Chart 3. Number of communication rounds to achieve 50% test accuracy at various mask ratios on the EMNIST dataset.

FSL-FedAvg training requires more communication rounds than federated SSPT-FedAvg and SSJO-FedAvg to reach 50% test accuracy, with SSJO-FedAvg requiring the least. Reducing the number of communication rounds to achieve a target test accuracy is an important goal for federated learning tasks. Semi-supervised learning methods may play an important part in achieving this.

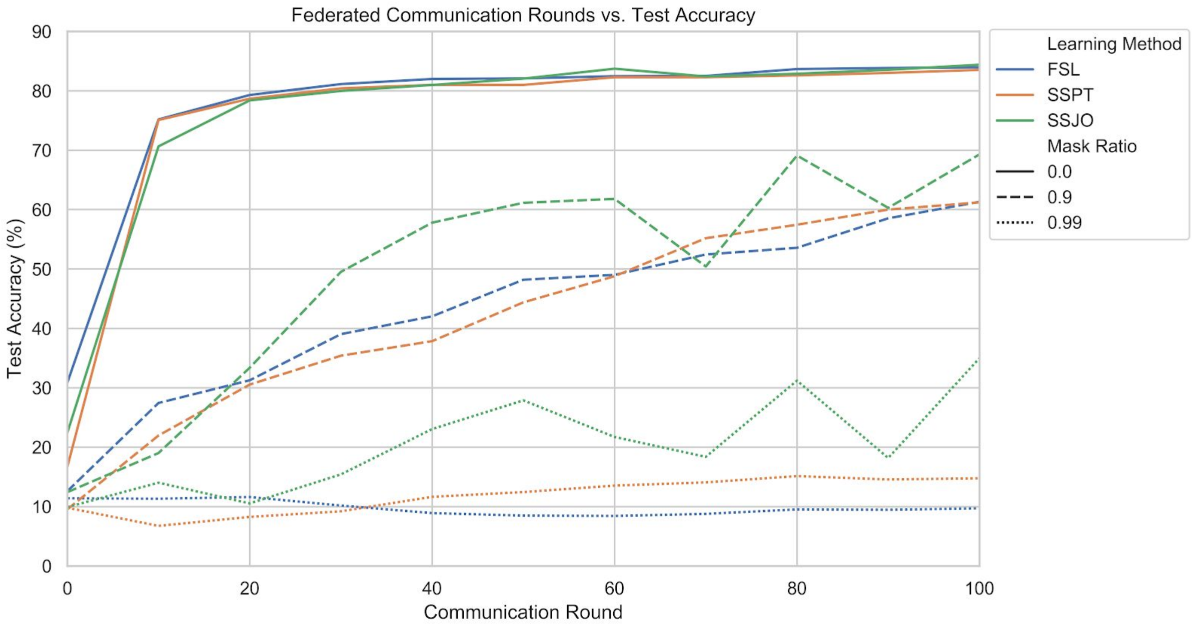

Figure 7. Test accuracy over communication rounds during federated training of fully supervised and semi-supervised methods.

## Federated Communication Rounds vs. Test Accuracy

| | FSL-FedAvg | | | SSPT-FedAvg | | | SSJO-FedAvg | | |
|---|---|---|---|---|---|---|---|---|---|
| **Communication Round** | **MR 0.0** | **MR 0.9** | **MR 0.99** | **MR 0.0** | **MR 0.9** | **MR 0.99** | **MR 0.0** | **MR 0.9** | **MR 0.99** |
| 0 | 30.83 | 12.58 | 11.39 | 16.65 | 9.52 | 9.80 | 22.40 | 12.42 | 9.91 |
| 10 | 75.18 | 27.44 | 11.32 | 75.08 | 21.96 | 6.75 | 70.64 | 19.01 | 14.02 |
| 20 | 79.29 | 31.26 | 11.61 | 78.67 | 30.54 | 8.26 | 78.38 | 33.37 | 10.53 |
| 30 | 81.13 | 39.03 | 10.17 | 80.41 | 35.40 | 9.20 | 79.97 | 49.52 | 15.42 |
| 40 | 81.99 | 42.03 | 8.90 | 81.00 | 37.85 | 11.64 | 80.98 | 57.80 | 23.05 |
| 50 | 82.09 | 48.19 | 8.48 | 80.99 | 44.38 | 12.44 | 82.05 | 61.14 | 27.88 |
| 60 | 82.46 | 49.01 | 8.42 | 82.28 | 48.78 | 13.53 | 83.72 | 61.80 | 21.72 |
| 70 | 82.50 | 52.44 | 8.77 | 82.28 | 55.18 | 14.07 | 82.41 | 50.42 | 18.37 |
| 80 | 83.66 | 53.57 | 9.52 | 82.59 | 57.45 | 15.13 | 82.87 | 69.10 | 31.20 |
| 90 | 83.84 | 58.53 | 9.46 | 83.01 | 60.00 | 14.56 | 83.53 | 60.29 | 18.18 |
| 100 | 83.94 | 61.31 | 9.70 | 83.52 | 61.21 | 14.76 | 84.39 | 69.29 | 34.98 |

Chart 4. Test accuracy over communication rounds during federated training of fully supervised and semi-supervised methods.

However, the training of semi-supervised learning methods is not without issues. We observe that the SSJO-FedAvg test accuracy shows oscillation over communication rounds. The joint optimization may be causing the training to shift between local optima between the two tasks. Future work may look into decaying the learning rate over communication rounds to prevent oscillation.

## Conclusions

We demonstrate the viability of Semi-supervised learning techniques in reducing the number of communication rounds to achieve target test accuracies for federated image classification tasks in which there are many unlabeled examples and few labeled examples. Using a simple dense autoencoder architecture, we show that the joint optimization of the self-supervised and supervised tasks provides higher test accuracy in the federated setting. In future work, we would like to example the use of learning rate decay to stabilize SSJO-FedAvg training, the use of more specialized proxy tasks such as predicting rotations, and the performance of these methods on richer datasets such as CIFAR 100 federated.

## References

[1] Doersch, Carl, Abhinav Gupta, and Alexei A. Efros. “Unsupervised Visual Representation Learning by Context Prediction.” 2015 IEEE International Conference on Computer Vision (ICCV) (2015).

[2] Noroozi, Mehdi, and Paolo Favaro. “Unsupervised Learning of Visual Representations by Solving Jigsaw Puzzles.” Lecture Notes in Computer Science (2016): 69–84.

[3] Gidaris, Spyros, Praveer Singh, and Nikos Komodakis. "Unsupervised representation learning by predicting image rotations." arXiv preprint arXiv:1803.07728 (2018).

[4] Liu, Shikun, Edward Johns, and Andrew J. Davison. “End-To-End Multi-Task Learning With Attention.” 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2019).

[5] Aleksei Triastcyn and Boi Faltings. “Federated generative privacy.” In IJCAI Workshop on Federated Machine Learning for User Privacy and Data Confidentiality (FML 2019), 2019.

[6] McMahan, H. Brendan, et al. "Communication-efficient learning of deep networks from decentralized data." arXiv preprint arXiv:1602.05629 (2016).

[7] Wang, Xialong, Abhinav Gupta. "Unsupervised Learning of Visual Representations using Video" arXiv preprint arXiv:1505.00687(2015)