







Embracing Semi-supervised Domain Adaptation for Federated Knowledge Transfer

Madhureeta Das¹, Zhen Liu², Xianhao Chen³, Xiaoyong Yuan⁴,
and Lan Zhang¹

¹ Department of Electrical and Computer Engineering, Michigan Technological University, Houghton, MI, USA

{mdas1, lanzhang}@mtu.edu

² Department of Civil, Environmental, and Geospatial Engineering, Michigan Technological University, Houghton, MI, USA

zhenl@mtu.edu

³ Department of Electrical and Electronic Engineering, University of Hong Kong, Pok Fu Lam, Hong Kong, China

xchen@eee.hku.hk

⁴ College of Computing, Michigan Technological University, Houghton, MI, USA

xyyuan@mtu.edu

Abstract. Given rapidly changing machine learning environments and expensive data labeling, semi-supervised domain adaptation (SSDA) is imperative when the labeled data from the source domain is statistically different from the partially labeled target data. Most prior SSDA research is centrally performed, requiring access to both source and target data. However, data in many fields nowadays is generated by distributed end devices. Due to privacy concerns, the data might be locally stored and cannot be shared, resulting in the ineffectiveness of existing SSDA. This paper proposes an innovative approach to achieve SSDA over multiple distributed and confidential datasets, named by Federated Semi-Supervised Domain Adaptation (FSSDA). FSSDA integrates SSDA with federated learning based on strategically designed knowledge distillation techniques, whose efficiency is improved by performing source and target training in parallel. Moreover, FSSDA controls the amount of knowledge transferred across domains by properly selecting a key parameter, *i.e.*, the imitation parameter. Further, the proposed FSSDA can be effectively generalized to multi-source domain adaptation scenarios. Extensive experiments demonstrate the effectiveness and efficiency of FSSDA design.

Keywords: Federated Learning · Semi-Supervised Domain Adaptation · Knowledge Distillation · Imitation Parameter

1 Introduction

The data generated by end devices, such as IoT devices, are essential to creating machine intelligence and actively shaping the world. However, when using a well-trained machine learning model, one common challenge is domain shift due to the diverse data distribution. Taking object detection as an example, a model trained for autonomous driving using data from sunny weather may perform poorly on foggy or snowy days. Domain adaptation addresses such situations. Typically, there is ample labeled data from the source domain to train the original model (e.g., sunny day object detection) but little labeled data from the target domain for domain adaptation (e.g., snowy day object detection). Given the fast-changing machine learning environments and expensive labeling, it is critical to develop domain adaptation approaches to handle the domain shift when there is limited labeled data and abundant unlabeled data from the target domain, *i.e.*, semi-supervised domain adaptation (SSDA).

Prior SSDA efforts are mainly conducted in a centralized manner, requiring access data from both source and target domains [6, 8, 33]. However, data in many fields nowadays is generated by distributed end devices. Given the widespread impact of recent data breaches [29], end users may become reluctant to share their local data due to privacy concerns. Although federated learning (FL) [32] offers a promising way to enable knowledge sharing across end devices without migrating the private end data to a central server, it is non-trivial to marry existing SSDA approaches with the FL paradigm. First, data from both the source and target domains is stored at end devices and cannot be shared in federated settings, resulting in the ineffectiveness of the existing centralized SSDA. Second, efficiency has been a well-recognized concern for FL. With distributed data from both source and target domains, more iterations need to be involved in obtaining a well-trained target model. Last but not least, the entangled knowledge across domains may lead to negative transfer [22], which becomes more challenging in federated settings with unavailable data from source and target domains across devices.

Enlightened by a popular model fusion approach, knowledge distillation (KD), that allows knowledge transfer across different models [14], we enable knowledge transfer between models from different domains without accessing the original domain data. Specifically, the target model can be learned with the help of the soft labels that are predictions of target samples by using the source model. Considering the distributed data from both source and target domains in federated settings, instead of waiting for a well-trained source model, we propose a parallel training paradigm to generate soft labels along with the source model to improve SSDA efficiency. However, due to domain discrepancy, the soft labels generated from the source model can be different from the ground truth target labels. Moreover, the soft labels derived at the initial federated training stage may perform poorly on SSDA. To address the above issues, we intend to align the source and target domains by adaptively leveraging both soft labels and ground truth labels. One major challenge here is the limited ground truth target labels in SSDA. To effectively leverage the few ground truth labels, we balance the knowl-

edge transferred from the soft and ground truth labels by properly selecting a key parameter, *i.e.*, the imitation parameter. Inspired by recent multi-task learning research [21], we control the amount of knowledge transferred from the source domain by adaptively selecting the imitation parameter based on the stochastic multi-subgradient descent algorithm (SMSGDA). The adaptively derived imitation parameters can be effectively used to handle multi-source SSDA problems under federated settings.

By integrating the above ideas, we propose an innovative SSDA approach for federated settings, named Federated Semi-Supervised Domain Adaptation (FSSDA). To the best of our knowledge, the research we present here is the first SSDA approach over distributed and confidential datasets. Our main contributions are summarized as follows: (i) To achieve SSDA over multiple distributed and confidential datasets, we propose FSSDA to integrate SSDA and FL, which enables knowledge transfer between a source domain(s) and target domain by leveraging domain models rather than original domain data based on strategically designed knowledge distillation techniques. (ii) Considering distributed data from both source and target domains in federated settings, we develop a parallel training paradigm to facilitate domain knowledge generation and domain adaptation concurrently, improving the efficiency of FSSDA. (iii) Due to different domain gaps in various SSDA problems, we control the amount of knowledge transferred from different domains to avoid negative transfer, where the imitation parameter, a key parameter of FSSDA, is properly selected based on the SMSGDA algorithm. (iv) Extensive experiments are conducted on the office dataset under both iid and non-iid federated environments. Experimental results validate the effectiveness and efficiency of the proposed FSSDA approach.

2 Related Work

2.1 Semi-Supervised Domain Adaptation (SSDA)

SSDA intends to address the domain shift when the labeled data from the source domain is statistically different from the partially labeled data from the target domain [31]. Classical SSDA exploits the knowledge from the source domain by mitigating the domain discrepancy [6, 8, 33]. Daumé *et al.* [6] proposed to compensate for the domain discrepancy by augmenting the feature space of source and target data. Donahue *et al.* [8] solved the domain discrepancy problem by optimizing the auxiliary constraints on labeled data. Yao *et al.* [33] proposed an SDASL framework to learn a subspace that can reduce the data distribution mismatch. Saito *et al.* [27] minimize the distance between unlabeled target samples and class prototypes through minimax training on entropy. Some recent research proposed adversarial-based methods, such as DANN [12], to adversarially learn discriminative and domain-invariant representations. However, all the above SSDA research requires access to both source and target domain data. Although one recent work, GDSDA [2], relaxed the source data requirement, it is designed to learn a shallow SVM model, and target samples are still required. Hence, GDSDA is ineffective in deep learning-based SSDA over distributed and confidential datasets from both source and target domains.

2.2 Label-Limited Federated Learning (FL)

FL has gained popularity in transferring knowledge across distributed and confidential datasets. Most existing FL focuses on supervised learning with ground-truth labeled samples at end devices. However, end data is often unlabeled in practice since annotating requires both time and domain knowledge [35,37]. Some recent research has focused on label-limited FL problems, mainly on semi-supervised FL and unsupervised domain adaptation (UDA). To handle semi-supervised FL, Albaseer *et al.* [1] proposed FedSem by developing distributed processing schemes based on pseudo-labeling techniques. Similarly, Jeong *et al.* [15] introduced the inter-client consistency loss to transfer labeling knowledge from labeled samples to nearby unlabeled ones with high confidence. Another line of label-limited FL on UDA problems is more challenging due to data requirements in prior centralized UDA research [31]. Peng *et al.* [23] proposed FADA to transfer source knowledge across multiple distributed nodes to a target node by using adversarial approaches. Peterson *et al.* [24] leveraged a prior domain expert to guide per-user domain adaptation. Zhuang *et al.* [38] predicted pseudo labels using a new clustering algorithm. However, the above UDA research targets either a single source or target dataset, while our design is under multiple distributed sources and target datasets for a more general domain adaptation setting. Moreover, UDA problems assume unknown target labels, making them ineffective in extracting target knowledge from the target labels in SSDA.

2.3 Knowledge Distillation (KD)

KD was initially proposed to compress a large neural model (teacher) down to a smaller model (student) [4,14]. Typically, KD compresses the well-trained teacher model into an empty student model by steering the student’s prediction towards the teacher’s prediction [25]. Urban *et al.* [30] used a small network to simulate the output of large depths using layer-by-layer distillation. Similarly, [18] used ℓ_2 loss to train a compressed student model from a teacher model for face recognition. Previous works [3,11,34] also show distilling a teacher model into a student model of the same architecture can improve student over teacher. Furlanello *et al.* [11] and Bagherinezhad *et al.* [3] demonstrated that by training the student using softmax outputs of the teacher as ground truth over generations. Some recent works [2,20,36] use KD to address domain adaptation problems through a teacher-student training strategy: train multiple teacher models on the source domain and integrate them to train the target student model. However, the above KD-based domain adaptation research requires access to either source or target data, which cannot be used to solve SSDA over multiple distributed and confidential datasets from both source and target domains.

3 Federated Semi-supervised Domain Adaptation

3.1 Problem Statement

As shown in Fig. 1, this work focuses on a typical SSDA problem over distributed K confidential datasets. Each dataset $\mathcal{D}^k = \{\mathcal{D}_s^k, \mathcal{D}_t^k\}$ includes data from two

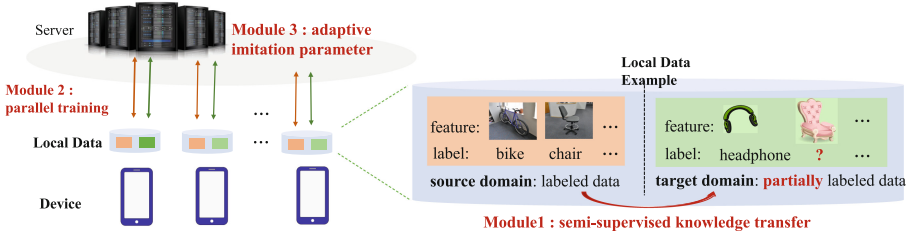


Fig. 1. FSSDA overview with three key modules.

domains, which is held by an end device k in a set of \mathcal{K} , $|\mathcal{K}| = K$. Specifically, the source domain data at device $k \in \mathcal{K}$ is fully labeled and denoted by $\mathcal{D}_s^k = (\mathcal{X}_s^k, \mathcal{Y}_s^k)$; the target domain data is partially labeled and denoted by $\mathcal{D}_t^k = \{\mathcal{D}_{t_l}^k, \mathcal{D}_{t_u}^k\}$. In particular, the labeled target data $\mathcal{D}_{t_l}^k = (\mathcal{X}_{t_l}^k, \mathcal{Y}_{t_l}^k)$ is much less than the unlabeled target data $\mathcal{D}_{t_u}^k = (\mathcal{X}_{t_u}^k)$. The datasets cannot be shared. The ultimate *goal* of this work is to obtain a global target model W_t that performs well on the distributed target domain data $\mathcal{D}_t = \{\mathcal{D}_t^k\}_{k \in \mathcal{K}}$ without accessing any data from both source and target domains $\mathcal{D} = \{\mathcal{D}^k\}_{k \in \mathcal{K}}$.

3.2 FSSDA Design

To achieve this goal, we propose Federated Semi-Supervised Domain Adaptation (FSSDA), including three key modules. First, a semi-supervised knowledge transfer module is developed to integrate SSDA with federated learning. Next, to improve the efficiency of FSSDA, the parallel training module is proposed to enable concurrent training between source and target domains. Finally, a key parameter of FSSDA, *i.e.*, the imitation parameter, is improved through the imitation parameter selection module to further boost the domain adaptation along with parallel training. The overall procedures of FSSDA are illustrated in Algorithm 1. In the following, we elaborate on the key modules of FSSDA design, respectively.

Semi-supervised Knowledge Transfer. Knowledge distillation (KD) [5, 14] has been a well-known technology to transfer knowledge from one or more models (teacher) into a new model (student). Typically, the student model is generated by mimicking the outputs of the teacher model on the same dataset. Note that the dataset here is not necessarily the one on which the teacher model was trained, which motivates our design for transferring knowledge in a semi-supervised manner. In FSSDA, KD is used to exploit the knowledge of unlabeled target data, where the source model is the teacher and the target model is the student. Specifically, to enable SSDA, FSSDA assigns each target sample a hard label y_t and a soft label y_t^* . The hard label for a labeled target sample is its actual label in a one-hot manner. For an unlabeled target sample, we use a “fake label” strategy that assigns all 0s as the label. Thus, all samples in the target

Algorithm 1: FSSDA

```

1 INPUT: for each device  $k \in \mathcal{K}$ , source domain data  $\mathcal{D}_s^k = (\mathcal{X}_s^k, \mathcal{Y}_s^k)$  of size  $N_s^k$ 
   and target domain data  $\mathcal{D}_t^k = \{(\mathcal{X}_{t_l}^k, \mathcal{Y}_{t_l}^k), (\mathcal{X}_{t_u}^k)\}$  of size  $N_t^k$ ; the number of
   rounds  $R$ .
2 OUTPUT:  $W_t$ 
3 initialize the global source and target model as  $W_s(0)$  and  $W_t(0)$ ;
4 initialize the local source and target model as  $w_s^k(0)$  and  $w_t^k(0)$  for each device
    $k \in \mathcal{K}$ ;
5 for each round  $r = 1, 2, \dots, R$  do
6   for each device  $k \in \mathcal{K}$  do // device source domain update
7     | update  $w_s^k$  using gradient descent.
8   end
9    $W_s(r) \leftarrow \sum_{i=1}^k \frac{N_s^i}{N_s} w_s^i(r)$  // server update
10  for each device  $k \in \mathcal{K}$  do // device target domain update
11    | Compute  $y^{*k}$  using equation (1)
12    | Calculate  $\lambda^k$  using equation (4) and Update  $w_t^k$  using equation (2)
13  end
14   $W_t(r) \leftarrow \sum_{i=1}^k \frac{N_t^i}{N_t} w_t^i(r)$  // server update
15 end

```

domain have hard labels. It should be mentioned that although the fake label may introduce some noise, the impact is subtle and controllable. On the one hand, only one class (the ground truth) will be affected among all classes (e.g., 31 classes in the office datasets [26]). On the other hand, the noise from hard labels can be controlled by properly selecting imitation parameters to balance the uncertainty from both the hard and soft labels. More discussions can be found in Sect. 15. Similar findings were shown in recent research [2]. Besides, the soft label of a target sample is derived by the prediction of the source model, which is a class probability value. By leveraging the source data and the target data with hard and soft labels, the process of training the target model is as follows: (i) Train the source model w_s^k for device $k \in \mathcal{K}$ with \mathcal{D}_s^k ; (ii) Use the learned source model to generate the soft label y_t^* for each sample $x_t \in \mathcal{X}_t$ in the target domain using softmax function σ . The soft label is defined by

$$y_t^* = \sigma(W_s(x_t)/T), \quad (1)$$

where W_s is the global source model by element-wise averaging local source model w_s^k for all device $k \in \mathcal{K}$ [19], and T is the temperature parameter to control the smoothness of the soft label. (iii) Train the target model w_t^k at device k using the hard and soft labels for each target data by

$$\arg \min \frac{1}{N_t^k} \sum_{i=1}^{N_t^k} [\lambda^k \ell_t(y_t^i, w_t^k(x_t^i)) + (1 - \lambda^k) \ell_t(y_t^{*i}, w_t^k(x_t^i))], \quad (2)$$

where N_t^k denotes the number of target domain samples at device k ; ℓ_t is the loss function; w_t is the local target model; λ^k is the imitation parameter for device k to balance the importance between the hard label y_t and the soft label y_t^* .

Parallel Training Between Source and Target. Efficiency has been a well-known concern in distributed machine learning. Since both source and target data are distributed across devices, instead of waiting for a well-trained source model, we propose a parallel training paradigm to accelerate FSSDA. As shown in Algorithm 1 (line 6–10), each device trains the source and target model simultaneously. Although the source model does not perform well in the initial stage, it still promotes domain alignment and thus accelerates the generation of the target model. Thus the main purpose of parallel computing is to train the source and target models simultaneously, speeding up the overall training process. Our parallel design is empirically evaluated in the experiment section below. It should be mentioned that parallel training does not incur additional communication costs since target model updates can be appended to source updates.

Adaptive Imitation Parameters. Although the semi-supervised knowledge transfer module integrates SSDA and FL in Sect. 15, FSSDA suffers negative transfer from the noisy hard and soft labels. Specifically, due to limited labeling in the target domain (e.g., three labeled samples per class in the experiments), most hard labels are fake ones with limited ground truth knowledge, which restricts the domain alignment performance. Besides, the soft labels during parallel training upon the above module can be noisy during initial training. Due to the domain gap between source and target, even the well-trained source model may generate improper soft labels, and the entangled knowledge learned from the source may lead to serious negative transfer [22]. These problems become more challenging in federated settings, where target devices do not have access to any source domain data. To properly balance the importance between hard labels and soft labels, we develop an adaptive approach for selecting the imitation parameter λ in (2). Specifically, the imitation parameter controls how much knowledge can be transferred from the source domain, whose importance has been shown in prior KD research [9, 17]. However, prior research determines the imitation parameter using either a brute-force search or domain knowledge, which cannot flexibly handle different domain discrepancies and noisy labels in FSSDA. Especially under heterogeneous federated settings, end devices have statistically heterogeneous data (non-iid) for both source and target domains.

To effectively select imitation parameters to adaptively use the noisy soft and hard labels, we consider problem (1) as a multi-task learning problem, where the soft loss and hard loss are the two task objectives. Since, in each federated training iteration, each device holds its own target domain data and the updated global source model, imitation parameters can be determined independently on the device side, which also addresses the data heterogeneity concern in federated settings. Specifically, we leverage the stochastic multi-subgradient descent algorithm (SMSGDA) [21], a well-known multi-task learning approach,

to adaptively select the imitation parameter at each federated iteration for each individual device. The objective function can be given by

$$\min_{\lambda \in [0,1]} \|\lambda \nabla_w \ell_t(y_t, w_t(x_t)) + (1 - \lambda) \nabla_w \ell_t(y_t^*, w_t(x_t))\|^2, \quad (3)$$

where ℓ_t is the loss function, y_t is the hard label, and y_t^* is the soft label generated by the source model W_s for the target domain dataset. w_t is the local target model. The analytical solution to the above problem can be given by

$$\lambda = \nabla_w \ell_t(y_t^*, w_t(x_t)) \times \frac{(\nabla_w \ell_t(y_t^*, w_t(x_t)) - \nabla_w \ell_t(y_t, w_t(x_t)))^T}{\|\nabla_w \ell_t(y_t, w_t(x_t)) - \nabla_w \ell_t(y_t^*, w_t(x_t))\|^2}, \quad (4)$$

where λ is clipped between $[0,1]$. Therefore, each device can efficiently derive its local imitation parameter with the above closed-form solution.

3.3 FSSDA over Multi-source Domains

This part introduces the extension of the proposed FSSDA to multi-source scenarios. When the distributed source data includes multiple source domains, then it is essential to extract the inter-domain knowledge to align the domain-specific representations better. Define the total number of source domains by S . Thus, the overall learning objective at device $k \in \mathcal{K}$ for S source domains can be extended from (2) to

$$\begin{aligned} \arg \min & \frac{1}{N_t^k} \sum_{i=1}^{N_t^k} [\lambda_1^k \ell_t(y_t^i, w_t^k(x_t^i)) + \sum_{j=1}^S \lambda_{j+1}^k \ell_t(y_t^{*ij}, w_t^k(x_t^i))], \\ \text{s.t.} & \sum \lambda_i^k = 1, \end{aligned} \quad (5)$$

where N_t^k is the total number of data samples in the target domain at device k , w_t is the local target model, y_t^{*ij} is the soft-label generated by the j th source model W_S^j for local data x_i , and λ^k is the imitation parameter for device $k \in \mathcal{K}$. In (5), imitation parameters are used to control more than two objective functions, *i.e.*, in total $S + 1$ losses, to jointly optimize the target model. Thus, given the new condition for imitation parameters, problem (5) cannot be solved by the closed-form solution in (4). We propose to use the Frank-Wolfe-based optimizer to solve the constrained optimization, which can scale to high-dimensional problems with low computational overhead [10, 28].

4 Experiments

4.1 Experimental Setup

We evaluate our models on the office dataset [26], which is widely used in domain adaptation. The office dataset includes 3 subsets: Webcam (795 samples) contains images captured by the web camera, Amazon (2,817 samples) contains

Table 1. Performance comparison between FSSDA and baseline approaches. Six cases are considered between Amazon (A), Webcam (W), and DSLR (D).

	A \rightarrow W	A \rightarrow D	W \rightarrow A	W \rightarrow D	D \rightarrow A	D \rightarrow W
SSDAOnly (iid)	66.83%	66.10%	56.98%	75.67%	49.67%	73.37%
FLOnly (iid)	64.08%	70.10%	41.07%	70.10%	41.07%	64.08%
FSSDA (iid)	83.01%	84.94%	66.23%	98.45%	71.39%	97.63%
SSDAOnly (non-iid)	64.81%	59.39%	52.51%	69.80%	46.83%	69.33%
FLOnly (non-iid)	52.47%	63.65%	38.71%	63.65%	38.71%	52.47%
FSSDA (non-iid)	82.15%	82.15%	66.09%	97.20%	69.67%	95.48%

images downloaded from amazon.com, and DSLR (498 samples) contains images captured by a digital SLR camera, sharing 31 classes. In the following, we use W, A, and D to represent the above three subsets, respectively.

We consider both iid and non-iid data distributions in federated settings. We use the distribution-based label imbalance [16] to generate non-iid data distributions, where each end device is allocated a proportion of the samples whose labels follow Dirichlet distribution. Specifically, we sample $p_l \sim \text{DirN}(\beta)$ and allocate a $p_{l,k}$ proportion of the instances of class l to each device k . In our setting, we set the β value as 0.1. Besides, we consider practical SSDA settings, where limited labeled samples are given in the target domain. In iid and non-iid settings, only 93 labeled examples (3 per class) are distributed across all the end devices. We use ResNet-101 [13] for the baseline methods and the proposed method. All models are pre-trained on ImageNet [7]. The model parameters are optimized using stochastic gradient descent with a learning rate of 0.001.

For baseline approaches, existing SSDA requires access to data from different domains, which is ineffective in federated settings. Besides, none of the existing FL targets SSDA. Hence, to evaluate the proposed FSSDA, we consider two baseline approaches. (i) *SSDAOnly*: Without using FL, device local knowledge cannot be transferred due to privacy concerns. Each device performs SSDA to generate a local target model with its own data but does not participate in federated learning. (ii) *FLOnly*: Without effective SSDA in federated settings, end devices can only leverage labeled target data to learn the target model collaboratively. There is no knowledge transfer from the source domain.

4.2 Experimental Results

Effectiveness Evaluation. We consider six cases for domain adaptations between Amazon, Webcam, and DSLR under both iid and non-iid federated settings. As shown in Table 1, we observe that *FSSDA outperforms both SSDAOnly and FLOOnly in all the cases*. We get the most promising result in the case of Webcam to DSLR both in iid and non-iid settings. The SSDAOnly and FLOOnly get around 70%, whereas our proposed FSSDA methods achieve more than 97% accuracy. FLOOnly cannot leverage the unlabeled samples, resulting in the

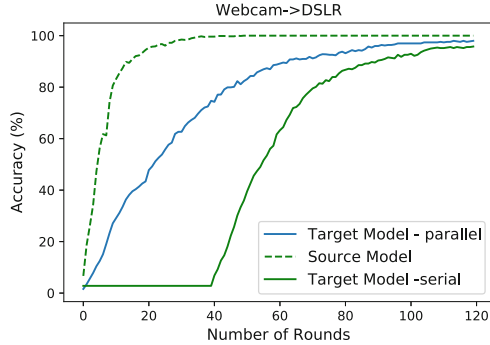


Fig. 2. Impact of the parallel training module of FSSDA ($W \rightarrow D$).

worst performance in most cases. Although SSDAOnly leverages unlabeled target domains via knowledge transfer, SSDAOnly cannot utilize the shared knowledge from other end-devices, which makes the learning ineffective. Especially in the non-iid cases, the number of data varies for each end device; the performance degradation of one of the local models affects the aggregated global model. Both in iid and non-iid settings, DSLR as a target is able to achieve good performance of over 82% accuracy even when the domain gap is large ($A \rightarrow D$). Moreover, due to the large domain gap between Webcam/DSLR and Amazon as well as the limited samples in Webcam/DSLR compared to Amazon, it is challenging to transfer knowledge to Amazon ($W \rightarrow A$ and $D \rightarrow A$). However, we still achieve better results compared to baselines. SSDAOnly and FLOnly can only get around 50% accuracy, while the FSSDA can achieve accuracy close to 70%, demonstrating the effectiveness of FSSDA in challenging SSDA scenarios.

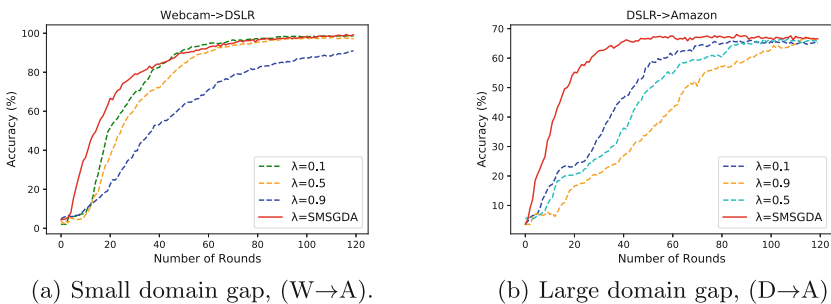


Fig. 3. Impacts of imitation parameters for FSSDA with different domain gaps.

Efficiency Evaluation. We compare the parallel training discussed in Sect. 3 with the serial training between the source (Webcam) and target (DSLR) models, as shown in Fig. 2. In serial training, the target model starts SSDA under

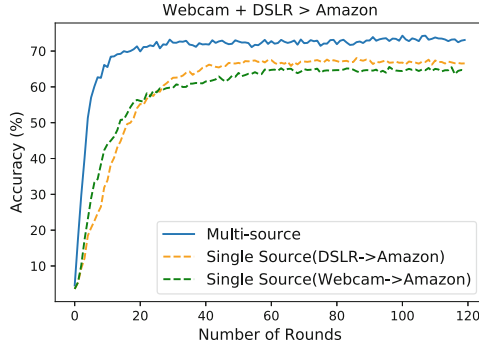


Fig. 4. FSSDA for multi-source SSDA (Target: Amazon).

federated settings until the source model is converged at around the 40th round. We observe that parallel training continuously outperforms serial training, which confirms the source model’s positive impact on SSDA. Compared with serial training, the convergence rate of parallel training is significantly improved by around 33%.

Impact of Imitation Parameters. We illustrate the impact of the imitation parameter on FSSDA by using static and adaptive (SMSGDA) values. In our design, a large λ indicates learning more knowledge from the target domain (hard label) and less from the source domain (soft label), and vice versa. We conduct experiments under two different domain shift scenarios in Fig. 3: the small domain gap from Webcam to Amazon and the large domain gap from DSLR to Amazon. We observe that when the domain gap is large (Fig. 3(b)), at the initial stage, a lower value of the imitation parameter ($\lambda = 0.1$) will speed up the performance of the target model, but at the end, performance degrades, which shows the impact of negative transfer. Besides, a larger imitation parameter ($\lambda = 0.9$) finally achieves good accuracy but does not converge quickly compared to our adaptive design. From Fig. 3(a), when the domain gap is small, the negative transfer will not be significant ($\lambda = 0.1$), and thus we can able to rely more on the source domain. However, the proposed adaptive imitation parameter scheme is irrespective of the domain difference, which performs well for both small and large domain shifts. Overall, our adaptive design trains the target model faster and converges quickly.

Effectiveness Evaluation for Multi-source SSDA. We evaluate the performance of FSSDA under a multi-source scenario. We focus on FSSDA with Amazon as the target since Amazon has a large domain gap compared to the other two domains (DSLR and Webcam), which is the most challenging FSSDA setting under the office dataset. As shown in Fig. 4, multi-source FSSDA outperforms both single-source results from Webcam and DSLR, which demonstrates the effectiveness of our FSSDA in multi-source scenarios. Meanwhile,

multi-source FSSDA further speeds up the overall federated training process to converge faster.

5 Conclusion

This paper proposed FSSDA to achieve semi-supervised domain adaptation (SSDA) over multiple distributed and confidential datasets. FSSDA integrates SSDA with federated learning based on adaptive and controllable knowledge transfer techniques, which include three key modules: semi-supervised knowledge transfer, parallel training, and adaptive imitation parameter selection. FSSDA can be used in single- or multiple-source SSDA problems. We empirically explored SSDA performance under iid and non-iid federated settings to validate the effectiveness and efficiency of our design.

Acknowledgement. The authors thank all anonymous reviewers for their insightful feedback. This work was supported by the National Science Foundation under Grants CCF-2106754, CCF-2221741, CCF-2153381, and CCF-2151238. The work of Zhen Liu was supported in part by Federal Highway Administration grant FHWA693JJ320-C000022, and the work of Xianhao Chen was supported in part by the HKU IDS Research Seed Fund under grant IDS-RSF2023-0012.

References

1. Albaseer, A., Ciftler, B.S., Abdallah, M., Al-Fuqaha, A.: Exploiting unlabeled data in smart cities using federated learning. arXiv preprint [arXiv:2001.04030](https://arxiv.org/abs/2001.04030) (2020)
2. Ao, S., Li, X., Ling, C.: Fast generalized distillation for semi-supervised domain adaptation. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 31 (2017)
3. Bagherinezhad, H., Horton, M., Rastegari, M., Farhadi, A.: Label refinery: Improving imagenet classification through label progression. arXiv preprint [arXiv:1805.02641](https://arxiv.org/abs/1805.02641) (2018)
4. Chen, D., Mei, J.P., Wang, C., Feng, Y., Chen, C.: Online knowledge distillation with diverse peers. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, pp. 3430–3437 (2020)
5. Cho, J.H., Hariharan, B.: On the efficacy of knowledge distillation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 4794–4802 (2019)
6. Daumé III, H., Kumar, A., Saha, A.: Frustratingly easy semi-supervised domain adaptation. In: Proceedings of the 2010 Workshop on Domain Adaptation for Natural Language Processing, pp. 53–59 (2010)
7. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: a large-scale hierarchical image database. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition, pp. 248–255. IEEE (2009)
8. Donahue, J., Hoffman, J., Rodner, E., Saenko, K., Darrell, T.: Semi-supervised domain adaptation with instance constraints. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 668–675 (2013)
9. Duan, L., Xu, D., Tsang, I.: Learning with augmented features for heterogeneous domain adaptation. arXiv preprint [arXiv:1206.4660](https://arxiv.org/abs/1206.4660) (2012)

10. Frank, M., Wolfe, P.: An algorithm for quadratic programming. *Naval Res. Logist. Q.* **3**(1–2), 95–110 (1956)
11. Furlanello, T., Lipton, Z., Tschannen, M., Itti, L., Anandkumar, A.: Born again neural networks. In: *International Conference on Machine Learning*, pp. 1607–1616. PMLR (2018)
12. Ganin, Y., et al.: Domain-adversarial training of neural networks. *J. Mach. Learn. Res.* **17**(1), 2030–2096 (2016)
13. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778 (2016)
14. Hinton, G., Vinyals, O., Dean, J.: Distilling the knowledge in a neural network. *arXiv preprint [arXiv:1503.02531](https://arxiv.org/abs/1503.02531)* (2015)
15. Jeong, W., Yoon, J., Yang, E., Hwang, S.J.: Federated semi-supervised learning with inter-client consistency & disjoint learning. *arXiv preprint [arXiv:2006.12097](https://arxiv.org/abs/2006.12097)* (2020)
16. Li, Q., Diao, Y., Chen, Q., He, B.: Federated learning on non-IID data silos: an experimental study. *arXiv preprint [arXiv:2102.02079](https://arxiv.org/abs/2102.02079)* (2021)
17. Lopez-Paz, D., Bottou, L., Schölkopf, B., Vapnik, V.: Unifying distillation and privileged information. *arXiv preprint [arXiv:1511.03643](https://arxiv.org/abs/1511.03643)* (2015)
18. Luo, P., Zhu, Z., Liu, Z., Wang, X., Tang, X.: Face model compression by distilling knowledge from neurons. In: *Thirtieth AAAI Conference on Artificial Intelligence* (2016)
19. McMahan, B., Moore, E., Ramage, D., Hampson, S., Arcas, B.A.: Communication-efficient learning of deep networks from decentralized data. In: *Artificial Intelligence and Statistics*, pp. 1273–1282 (2017)
20. Meng, Z., Li, J., Gong, Y., Juang, B.H.: Adversarial teacher-student learning for unsupervised domain adaptation. In: *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 5949–5953. IEEE (2018)
21. Milojkovic, N., Antognini, D., Bergamin, G., Faltings, B., Musat, C.: Multi-gradient descent for multi-objective recommender systems. *arXiv preprint [arXiv:2001.00846](https://arxiv.org/abs/2001.00846)* (2019)
22. Pan, S.J., Yang, Q.: A survey on transfer learning. *IEEE Trans. Knowl. Data Eng.* **22**(10), 1345–1359 (2009)
23. Peng, X., Huang, Z., Zhu, Y., Saenko, K.: Federated adversarial domain adaptation. *arXiv preprint [arXiv:1911.02054](https://arxiv.org/abs/1911.02054)* (2019)
24. Peterson, D., Kanani, P., Marathe, V.J.: Private federated learning with domain adaptation. *arXiv preprint [arXiv:1912.06733](https://arxiv.org/abs/1912.06733)* (2019)
25. Polino, A., Pascanu, R., Alistarh, D.: Model compression via distillation and quantization. *arXiv preprint [arXiv:1802.05668](https://arxiv.org/abs/1802.05668)* (2018)
26. Saenko, K., Kulis, B., Fritz, M., Darrell, T.: Adapting visual category models to new domains. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) *ECCV 2010*. LNCS, vol. 6314, pp. 213–226. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-15561-1_16
27. Saito, K., Kim, D., Sclaroff, S., Darrell, T., Saenko, K.: Semi-supervised domain adaptation via minimax entropy. In: *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 8050–8058 (2019)
28. Sener, O., Koltun, V.: Multi-task learning as multi-objective optimization. In: *Advances in Neural Information Processing Systems*, vol. 31 (2018)
29. SOBERS, R.: 98 must-know data breach statistics for 2021 (2021). <https://www.varonis.com/blog/data-breach-statistics/>

30. Urban, G., et al.: Do deep convolutional nets really need to be deep (or even convolutional)? (2016)
31. Wang, M., Deng, W.: Deep visual domain adaptation: a survey. *Neurocomputing* **312**, 135–153 (2018)
32. Yang, Q., Liu, Y., Chen, T., Tong, Y.: Federated machine learning: concept and applications. *ACM Trans. Intell. Syst. Technol. (TIST)* **10**(2), 1–19 (2019)
33. Yao, T., Pan, Y., Ngo, C.W., Li, H., Mei, T.: Semi-supervised domain adaptation with subspace learning for visual recognition. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2142–2150 (2015)
34. Yim, J., Joo, D., Bae, J., Kim, J.: A gift from knowledge distillation: fast optimization, network minimization and transfer learning. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4133–4141 (2017)
35. Zhang, L., Yuan, X.: Fedzkt: zero-shot knowledge transfer towards heterogeneous on-device models in federated learning. arXiv preprint [arXiv:2109.03775](https://arxiv.org/abs/2109.03775) (2021)
36. Zhou, K., Yang, Y., Qiao, Y., Xiang, T.: Domain adaptive ensemble learning. *IEEE Trans. Image Process.* **30**, 8008–8018 (2021)
37. Zhu, X., Goldberg, A.B.: Introduction to semi-supervised learning. *Synth. Lect. Artif. Intell. Mach. Learn.* **3**(1), 1–130 (2009)
38. Zhuang, W., Gan, X., Wen, Y., Zhang, X., Zhang, S., Yi, S.: Towards unsupervised domain adaptation for deep face recognition under privacy constraints via federated learning. arXiv preprint [arXiv:2105.07606](https://arxiv.org/abs/2105.07606) (2021)