Wuhan University
Journal of Natural Sciences

2022, Vol.27 No.6, 499-507

Article ID 1007-1202(2022)06-0499-09
DOI https://doi.org/10.1051/wujns/2022276499

A Federated Domain Adaptation Algorithm Based on Knowledge Distillation and Contrastive Learning

□ **HUANG Fang, FANG Zhijun†, SHI Zhicai, ZHUANG Lehui, LI Xingchen,HUANG Bo**

School of Electrical and Electronic Engineering, Shanghai University of Engineering Science, Shanghai 201600, China

Abstract: Smart manufacturing suffers from the heterogeneity of local data distribution across parties, mutual information silos and lack of privacy protection in the process of industry chain collaboration. To address these problems, we propose a federated domain adaptation algorithm based on knowledge distillation and contrastive learning. Knowledge distillation is used to extract transferable integration knowledge from the different source domains and the quality of the extracted integration knowledge is used to assign reasonable weights to each source domain. A more rational weighted average aggregation is used in the aggregation phase of the center server to optimize the global model, while the local model of the source domain is trained with the help of contrastive learning to constrain the local model optimum towards the global model optimum, mitigating the inherent heterogeneity between local data. Our experiments are conducted on the largest domain adaptation dataset, and the results show that compared with other traditional federated domain adaptation algorithms, the algorithm we proposed trains a more accurate model, requires fewer communication rounds, makes more effective use of imbalanced data in the industrial area, and protects data privacy.

Key words: federated learning; multi-source domain adaptation; knowledge distillation; contrastive learning

CLC number: TP 399

Received date: 2022-09-18

Foundation item: Supported by the Scientific and Technological Innovation 2030—Major Project of "New Generation Artificial Intelligence" (2020AAA 0109300)

Biography: HUANG Fang, female, Master candidate, research direction: federated learning. E-mail: 2431835009@qq.com

† To whom correspondence should be addressed. E-mail: zjfang@sues.edu.cn

0 Introduction

In unsupervised deep learning, to avoid the costly annotation process, other similar datasets (i.e. source domains) are used to train models that can be applied to new datasets (i. e. target domains), and the knowledge from the source domains is used to determine the soft labels of the target domains. In the workflow collaboration process of the whole industry chain of smart manufacturing, the lack of uniform knowledge representation across disciplines, the lack of global collaboration in the process, the existence of information silos and the lack of privacy protection can lead to domain shifts between source domains, resulting in poor overall control performance. There have been many studies and solutions based on the domain adaptation. The earlier unsupervised multi-source domain adaptation (UMDA) [1] approach solves the domain shift by extracting transferable features on multiple source domains, and the more recent UMDA[2,3] approach performs knowledge transfer by constructing Source-Target pairs. However, these traditional UMDA approaches do not consider the unavailability of source domain data and cannot be applied in the general environment of privacy protection policies.

In order to train models that can be applied to the target domain even if the unavailability of source domain data, FADA[4] first proposed the concept of federated domain adaptation, which is an application of federated learning to the domain adaptation. Federated learning is a distributed machine learning method for solving the data security problem and responding to data secu-

rity regulations. It enables multiple clients to collaboratively train models under the coordination of a central server, and at the same time stores the training data on a local client, reducing the risk of privacy breaches and data transfer costs associated with traditional centralized machine learning methods[5]. Federated learning has been used in numerous fields and scenarios such as computer vision, domain adaptation, natural language processing, and recommender systems. Although it is promising, it has been facing numerous practical challenges due to the data heterogeneity. According to existing studies[6,7], data heterogeneity among clients will degrade the performance of global models, leading to slow convergence and even scattering, which is a more prominent challenge in the federated domain adaptation. There are many studies based on knowledge distillation to improve the efficiency of global model aggregation. Knowledge distillation uses integrated knowledge from local models to mitigate the impact of data heterogeneity, but does not adequately address the inherent heterogeneity among local models, and the problem persists when using knowledge distillation to implement federated domain adaptation.

In this paper, we combine the idea of model contrastive learning with the method of knowledge distillation, using contrastive learning to constrain the training of local models, so that the optimal of local models are closer to the optimal of global models; at the same time, we use knowledge distillation to improve the efficiency of model aggregation, obtain higher quality global models, better solve the problems of domain shift and data heterogeneity in federated domain adaptation, and the performance of federated domain adaptation on Non-independent identical distribution (non-IID) source domain datasets is improved. This paper uses the largest domain adaptation dataset DomainNet[8] for experimental verification. Based on extensive experimental results, the main advantages of the proposed algorithm in this paper are as follows:

1) It compensates for the fact that knowledge distillation can only optimize global models using integrated knowledge and has no way of mitigating the inherent heterogeneity between local models.

2) The accuracy is significantly better than that of many existing federated domain adaptation methods.

3) Fewer communication costs are required to achieve the same model accuracy, thus also reducing the risk of privacy breaches during the upload and download of model parameters.

1 Related Work

1.1 Federated Learning

Federated learning has received increasing attention for its ability to protect data privacy and to maximize the computing power of end devices in cloud systems. FedAvg[9] is the most classical federated learning algorithm, which can be mainly divided into four steps, as shown in Fig. 1. First, the parties download the initialized model parameters from the server, then the selected parties use the local data to train E (E is the number of local epochs) periods to update the model, and the updated model is uploaded back to the server. Finally, the server aggregates the received local model parameters or gradients on average to get the updated global model.

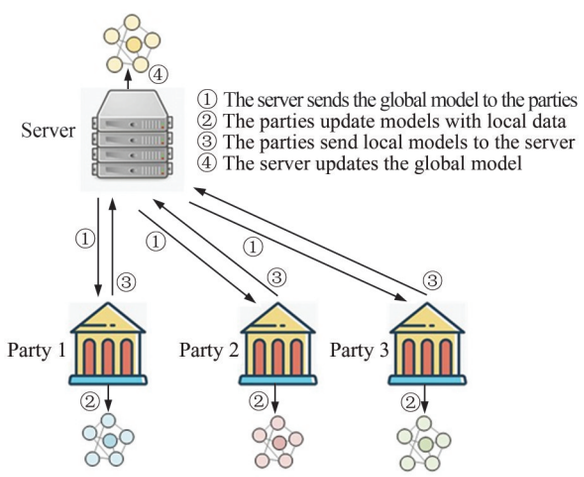

Fig.1 Framework of FedAvg

The size of E is crucial to the convergence speed of the global model. When E is greater than 1, the number of communication rounds will be reduced at the cost of increasing local computation, which solves the problem of high communication overhead in federated learning. However, in practical scenarios, due to the data heterogeneity among parties and the non-IID problem among local data, too large E will cause each party to be close to the optimal of its own local objective function , but far away from the optimal of the global objective function, and even affect the convergence of the global model[10]. Data heterogeneity among parties is the biggest challenge in federated learning, and many algorithms have been proposed in recent years to deal with data heterogeneity. These algorithms can be mainly divided into two

categories: optimization in the local training phase and improvement in the aggregation phase of the server. In addition, personalized federated learning and robust federated learning algorithms are also the research directions to solve the data heterogeneity problem.

1.2 Federated Domain Adaptation

Federated domain adaptation refers to federated multi-source domain adaptation, which is more challenging than traditional domain adaptation. Traditional domain adaptation aims to transfer knowledge from a labeled source domain to an unlabeled target domain[11], whereas in many real-world settings, labeled data comes from multiple domains. In multi-source domain adaptation, several approaches[12,13] apply difference alignment to reduce the gap between source and target domains, while minimizing *h*-divergence requires the pairwise computation of data from the source and target domains, which are not available in the background of privacy protection. This is why the federated learning algorithm framework is used for multi-source domain adaptation.

FADA[4] first proposed federated domain adaptation, using generative adversarial networks to optimize *h*-divergence without accessing data. However, in the adversarial training process, after each batch of data training is completed, the models in the source and target domains are required to be synchronized, which leads to significant communication costs. To address this problem, some studies have applied knowledge distillation[14] to the area of multi-source domain adaptation with the help of teacher-student networks. Multiple teacher models are trained in the source domains, and the teacher models are used to guide the training of student models in the target domain, avoiding unnecessary communication costs. However, in the presence of malicious source domains, the knowledge obtained through knowledge distillation may be wrong or inaccurate, leading to poor accuracy of the final global model. KD3A[15] alleviates the impact of malicious source domains by assigning high weights to source domains with high contributions while reducing the weights of source domains with low contributions. Although KD3A[15] is robust to negative transfer, knowledge distillation only improves the global model in federated learning and does not fully utilize the integrated knowledge to guide local model training, which can affect the quality of knowledge integration. FEDGEN[16] proposed a data-free knowledge distillation approach to integrate user information by learning a lightweight generator at the server, then broadcasting it to parties and using the learned knowledge as an inductive bias to regulate local model training.

1.3 Contrastive Learning

Self-supervised learning has become a popular research direction due to the fact that it does not require label information and directly uses the data itself as supervised information to learn feature representations of sample data, saving the human, material and financial resources required in the manual annotation process. Contrastive learning is a type of self-supervised learning method, which trains models by reducing the distance between different augmented view representations of the same image while increasing the distance between augmented view representations of different images, and has shown great potential for learning visual features with first-class results[17,18]. The most typical contrastive learning framework is SimCLR[19], which first generates two different augmented views x_i and x_j for the images x through the augmented operator, and then passes the views through the base encoder and the projection head to obtain the representation vectors z_i and z_j of the augmented views x_i and x_j. Equation (1) is the definition of contrastive loss for the sample images, where N refers to the number of sample images, t denotes the temperature parameter, sim() denotes the calculated cosine similarity, and the final loss is obtained by summing the contrastive loss of a batch of sample images.

$$l_{i,j}=-\log\frac{\exp\left(\mathrm{sim}\left(z_i,z_j\right)/t\right)}{\sum_{k=1}^{2N} l_{[k\neq i]}\exp\left(\mathrm{sim}\left(z_i,z_k\right)/t\right)} \tag{1}$$

FedCA[20] is the first algorithm designed for the federated self-supervised feature learning problem and also the first algorithm that combines federated learning with contrastive learning. FedCA consists of a dictionary module and an alignment module, enabling the local model to learn consistent and aligned feature representations while ensuring data security. MOON[21] proposed contrastive learning at the model level, aiming to reduce the distance between the representation learned by the local model and the representation learned by the global model, and to increase the distance between the representation learned by the local model and the representation learned by the previous round of local models. MOON improved the model performance and model sta-

bility of federated learning on non-IID data, but it was only for supervised learning.

2 Method

2.1 Preliminary Knowledge

The K source domains in multi-source domain adaptation (UMDA) are denoted by $S=\{D_s^k\}_{k=1}^K$, each source domain with n_k labeled examples is denoted as $D_s^k=\{(X_i^k, Y_i^k)\}_{i=1}^{n_k}$. Let D_t denote the target domain with n_t unlabeled examples as $D_t=\{X_i^t\}_{i=1}^{n_t}$. Our goal is to use the K source domains with annotated information to train a model h that can be used in the target domain. In general, we record the local model trained in the k-th source domain as $\{h_k^d\}_{d=1}^D$, and the corresponding global model as $\{h_t^d\}_{d=1}^D$, where D is the iteration round of server aggregation in the federated learning. To avoid negative transfer, different source domains will be given different domain weights $\{\alpha^k\}_{k=1}^K$, where $\sum_{k=1}^K \alpha^k=1$. Then $h_t^d=\sum_{k=1}^K \alpha^k h_k^d$.

2.2 Improved Knowledge Distillation

Traditional knowledge distillation belongs to the teacher-student network paradigm, which aims to use knowledge distilled from one or more teacher models to learn a lightweight student model, and it has now become an effective solution for improving the model aggregation in federated learning. However, due to the existence of malicious or irrelevant source domains, the integration strategies in traditional knowledge distillation (e.g. maximal and average integration) may not result in high quality knowledge, so we improve the quality of knowledge by improving integration strategy.

First , each source domain is trained with local data to obtain local models $\{h_k^d\}_{k=1}^K$ (as h_1^d, h_2^d, h_K^d in Fig. 2) , then put each target domain sample $X_i^t \in D_t$ into those local models for each of the K source domains successively to obtain a confidence prediction $\{q_s^k(X_i^t)\}_{k=1}^K$ for each class (shown as the table in the bottom left corner in Fig. 2) , and use the class with the highest confidence level as the target domain sample label. As shown in Fig. 2, the improved knowledge distillation has three main processes.

1) Set a relatively high confidence threshold gate (as gate = 0.9 in Fig. 2), filter out all local models whose confidence predictions are below the gate (as q_s^4 filtered out in Fig. 2). The purpose of setting the gate is to filter out the unconfident teacher model in $\{h_k^d\}_{k=1}^K$.

2) For the remaining teacher models, the confidence degrees of the same classes are summed (shown as tables summed in the same column in Fig. 2), and the class with the largest summed confidence degree is set as the consensus class (as the consensus class is Dragon in Fig. 2). Then, we filter out the teacher models whose confidence degree of the consensus class is smaller than the gate (as q_s^3 filtered out in Fig. 2).

3) At this point, we have obtained a set of teacher models that all support consensus classes. These models were integrated on average to obtain high quality knowledge p_i while recording the support for the number of source domains n_{p_i}.

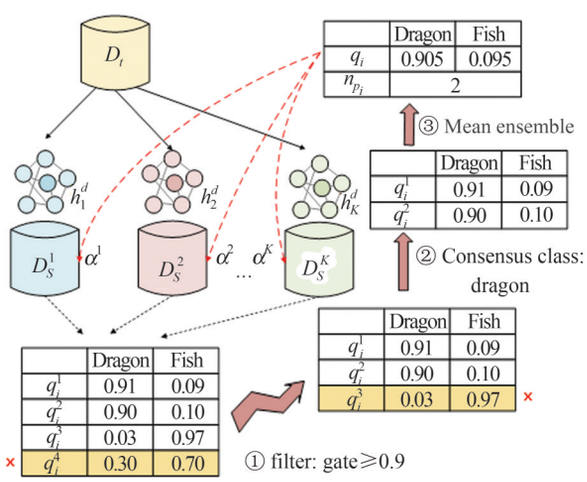

Fig.2 Flowchart of the improved knowledge distillation

The contribution of each source domain is calculated according to the obtained high-quality knowledge p_i and the number of models n_{p_i} supporting the consensus class. We first calculate the integrated knowledge quality $Q(S)$ about the set S of source domains by using equation (2), then consider the contribution of the source domain D_s^k to the integrated knowledge quality by removing the k-th source domain D_s^k from the set S of source domains to obtain a new set $\left(S\setminus\{D_s^k\}\right)$, and similarly calculate the integrated knowledge quality $Q\left(S\setminus\{D_s^k\}\right)$ using equation (2), and the degree of decrease in knowledge quality $Q(S)$ and knowledge quality $Q\left(S\setminus\{D_s^k\}\right)$ indicates the degree of contribution $C\left(D_s^k\right)$ of source domain D_s^k to the set S, which is calculated as shown in equation (3).

$$Q(S)=\sum_{X_i^t\in D_t} n_{p_i}(S)*\max p_i(S) \tag{2}$$

$$C\left(D_s^k\right)=Q(S)-Q\left(S\setminus\left\{D_s^k\right\}\right) \tag{3}$$

And through equation (4), high weights are given to source domains with high contribution degree and low weights are given to source domains with low contribution degree.

$$\alpha^k=\frac{C\left(D_s^k\right)}{\sum_{k=1}^{K}C(D_s^k)} \tag{4}$$

2.3 Network Architecture and Loss Function

Although the improvement of integration strategy of knowledge distillation brings higher quality transferable integration knowledge, knowledge distillation can only use this integration knowledge to improve the aggregation process of servers in federated learning, but does not fully utilize the integration knowledge to guide local model training. As each client updates its local model, its local optimum may be far from the global optimum, a distance that grows worse as the number of federated learning iterations increases, and which in turn affects the quality of knowledge integration. Contrastive learning can bring local models closer to the global model, so this paper adds contrastive learning to the training process of local models to mitigate the inherent heterogeneity among local models. Minor but effective modifications are mainly made to the network architecture and loss functions of the local models.

As shown in Fig. 3, the network architecture of the local model is divided into two parts: the feature extraction and the classifier. The feature extraction is used to obtain a feature representation of each image in the same dimension, and the classifier is used to generate predicted confidence for each class. The loss function of the local model also consists of two parts, one is the most typical cross-entropy loss function defined as l_{cro} and the other is the model contrastive loss function defined as l_{con}.

During every iteration of federated learning, each client trains its local model h_k^d and uploads the model to the server. After server aggregation, the server sends the global model h_t^d to each client, and then the client uses the local data to update the global model to get a new local model h_k^{d+1}. As shown in Fig. 3, in a federated learning round we call the model h_k^d as the previous model, and define the feature representation of the local sample data obtained from the previous model as z_{prev}. Model h_t^d is called the globe model, and the feature representation of the local sample data through the globe model is defined as z_{glob}. Model h_k^{d+1} is called the local model, and the feature representation of the local sample data obtained from the local model is defined as z.

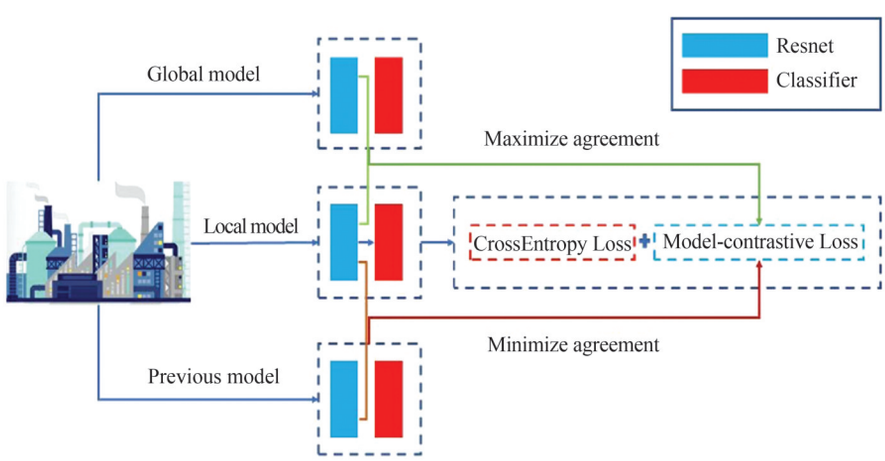

Fig.3 Network architecture and loss function for local models

Then the model contrastive loss function is defined as:

$$l_{\text{con}}=-\log\frac{\exp\left(\text{sim}\left(z,z_{\text{glob}}\right)/t\right)}{\exp\left(\text{sim}\left(z,z_{\text{glob}}\right)/t\right)+\exp\left(\text{sim}\left(z,z_{\text{prev}}\right)/t\right)} \tag{5}$$

With such model contrastive loss, each client can constrain the similarity between z and z_{glob} to be increasing and the similarity between z and z_{prev} to be decreasing as it updates its local model. This allows its local model to be closer to the global model, and mitigates the inherent heterogeneity among local models.

The loss function of the final local model is defined as:

$$l=l_{\text{cro}}+\mu l_{\text{con}} \tag{6}$$

where μ is a hyperparameter that controls the weight of the model contrastive loss.

2.4 Overall Flow of the Algorithm

The entire algorithm workflow is shown in Algorithm 1. Except that model contrastive learning is not available in the first round of communication, we can combine the improved knowledge distillation method with model contrastive learning in all other communication rounds, mitigate negative transfer by assigning weights to each domain during the aggregation stage of federated learning, and optimize the inherent heterogeneity of local models through model contrastive learning during the local training stage of federated learning.

Algorithm 1 The whole process of the algorithm

Input: source domain $S=\{D_s^k\}_{k=1}^K$, target domain D_t, source model $\{h_k^d\}_{k=1}^K$, target model h_t^d, number of communication rounds D, confidence threshold gate, temperature t, hyper-parameter μ

Output: the final target model h_t^D

(1) **while** $d=1$ **do**

(2) **for** D_s^k **in** S **do**

(3) source model initialize: h_k^d //train h_k^d with classification loss

(4) $\{\alpha^k\}_{k=1}^K=$ **ImprovedKnowledgeDistillation**($\{h_k^d\}_{k=1}^K$, gate)

(5) target model $h_t^d=\sum_{k=1}^K \alpha^k h_k^d$

(6) **for** $d=2,3,\cdots,D$ **do**

(7) **for** $k=1,2,\cdots,K$ **in parallel do**

(8) send the target model h_t^{d-1} to D_s^k

(9) $h_k^d \leftarrow$ **LocalTraining**($h_t^{d-1}, h_k^{d-1}, t, \mu$)

(10) //train h_k^d with classification loss and model-contrastive loss

(11) $\{\alpha^k\}_{k=1}^K=$ **ImprovedKnowledgeDistillation**($\{h_k^d\}_{k=1}^K$, gate)

(12) target model $h_t^d=\sum_{k=1}^K \alpha^k h_k^d$

(13) return h_t^d

3 Experiments and Analysis of Results

3.1 Experimental Setup

3.1.1 Datasets

We chose DomainNet[8], the largest domain adaptation dataset, which contains six domains: clipart, infograph, painting, sketch, real, quickdraw, and each domain contains 345 common objects. For the clipart and infograph domains, there are about 150 images per category on average; for the painting and sketch domains, there are about 220 images per category on average; and for the real domain, there are about 510 images; for the quickdraw domain, there are 500 images per category, and the entire dataset contains 5.96 million images. There are obvious problems of domain shift and data heterogeneity among various domains, which is suitable to be used to verify the feasibility of our method. During the experimental setup, we conventionally select each domain in turn as the target domain and use the remaining domains as the source domains.

3.1.2 Baselines

We conducted extensive horizontal comparison experiments with three classical federated multi-source domain adaptation methods, namely the SHOT[22], the FADA[4] and the KD3A[15]. Among them, the KD3A also adopted knowledge distillation to improve the efficiency of model aggregation. Five sets of vertical comparison experiments were also conducted by selecting each domain in turn as the target domain. Comparisons were made in terms of model accuracy and for the number of communication rounds when the global model reached convergence.

3.1.3 Experimental environment configuration

After summarizing some of our previous work, we implemented our algorithm by the PyTorch, using a pretrained ResNet to extract image features, and using a fully-connected layer as a classifier to output the confidence for each class. For model optimization, we used stochastic gradient descent (SGD) with 0.9 momentum as the optimizer and used a cosine annealing strategy to reduce the learning rate from 0.005 to 0.001, with batch size set to 50 and communication rounds set to 80. In the process of performing knowledge distillation, we slowly increased gate from 0.8 to 0.95 to find the most appropriate gate. We set the temperature parameter t to 0.5 and the weight μ of model contrastive loss to 1 by default.

3.2 Model Accuracy

The top-1 model accuracy was compared between the different algorithms under the same experimental setup as described above. Five experiments were conducted for each algorithm, and the results with the highest accuracy were selected from the five experiments, as shown in Table 1. By comparing the model accuracy of multiple UMDA methods, it can be found that the model accuracy of our proposed algorithm is the highest, with an average accuracy of 1.4% higher than the KD3A, 7% higher than the SHOT, and 9.7% higher than the FADA. It suggests that knowledge distillation and contrastive learning can be additive to each other, although they act

Table 1 UMDA accuracy on the DomainNet dataset %

Target domain	SHOT	FADA	KD3A	Ours
Clipart	61.7	59.1	73.3	74.2
Infographics	22.2	21.7	21.8	23.2
Quickdraw	12.2	8.8	15.3	16.7
Painting	52.6	47.9	59.5	60.6
Real	67.7	60.8	69.9	72.7
Sketch	48.6	50.4	58.7	59.8
Avg	44.2	41.5	49.8	51.2

in different stages of federated domain adaptation.

3.3 Privacy and Security

Table 2 shows the number of communication rounds required by KD3A and our proposed algorithm to achieve the same model accuracy. We can observe that for the target domain of Quickdraw, the proposed algorithm achieves the model accuracy of KD3A with only one-third of the communication rounds of KD3A; for the target domains of Infograph, Painting and Real, the proposed algorithm requires less than half of the communication rounds than KD3A. This indicates that the proposed algorithm is much more efficient than the KD3A in terms of communication. This means that contrastive learning is an effective solution for reducing the distance between the local and global models, and that the contrastive loss of our models can effectively improve accuracy without slowing down convergence.

Table 2 The number of rounds of different approaches to achieve the same accuracy

Target domain	KD3A		Ours	
	round	speedup	round	speedup
Clipart	80	1×	55	1.4×
Infograph	80	1×	32	2.5×
Quickdraw	80	1×	26	3×
Painting	80	1×	38	2.1×
Real	80	1×	33	2.4×
Sketch	80	1×	49	1.6×

An attacker can recover the image from the gradients passed during the communication[23], which means that the more the rounds of communication, the longer the model gradients of the local models are in the communication process, while leading to privacy leakage and making the process of training the models through federated learning insecure. In this regard, the higher the communication efficiency of the algorithm, the higher the privacy security of the algorithm for the data.

3.4 Communication Efficiency

Both KD3A and the proposed algorithm optimize the global model using knowledge distillation in the server aggregation phase, so a comparison in communication efficiency with the KD3A better illustrates that the proposed algorithm can compensate for the shortcoming that knowledge distillation only improves the global model but does not address the inherent heterogeneity among local models.

Figure 4 shows the variation in accuracy for each round during the training period. As seen in Fig. 4, the

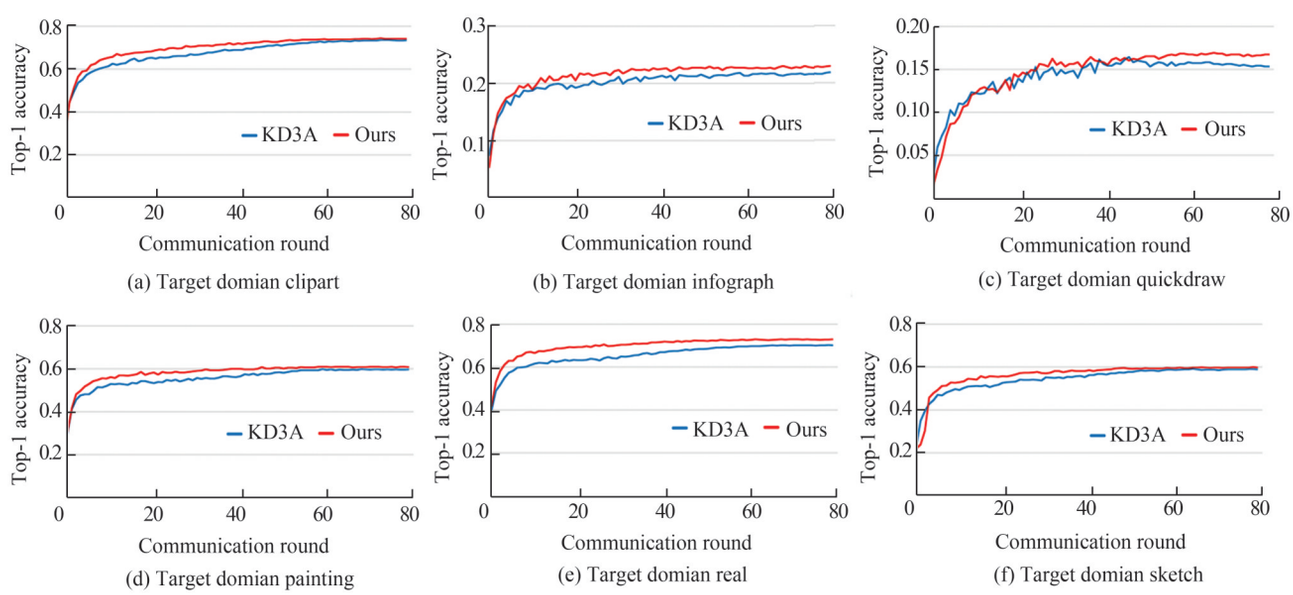

Fig. 4 Variation of accuracy with communication rounds

accuracy of the proposed algorithm improves faster than that of the KD3A before 10 rounds, regardless of which domain is used as the target domain; for each round after 10 rounds, the accuracy of the algorithm we proposed is higher than that of the KD3A. This indicates that the proposed algorithm has higher communication efficiency and can improve accuracy effectively without slowing down the convergence rate.

4 Conclusion

This paper presents a federated domain adaptation algorithm based on knowledge distillation and contrastive learning. The impact of data heterogeneity on model accuracy and convergence is mitigated by a two-pronged approach in the local model training phase and the server aggregation phase. To make better use of the transferable integration knowledge, knowledge distillation is combined with contrastive learning so that the integration knowledge obtained through the knowledge distillation can be used to tune the local model while allowing better optimization of the integration knowledge quality. At the same time, the combination of model-level contrastive learning with knowledge distillation is extended to self-supervised learning.

The algorithm proposed in this paper can effectively protect the privacy of industrial data, break the information silos, and solve the problem of unsafe knowledge sharing caused by privacy invasion and leakage of industrial data stream. The algorithm also has strong robustness in the face of unbalanced industrial data, which can be used more effectively to exploit the unbalanced data in the industrial field and uncover greater value.

References

[1] Yang Q, Liu Y, Chen T J, *et al*. Federated machine learning: Concept and applications[J]. *ACM Transactions on Intelligent Systems and Technology* (*TIST*), 2019, **10**(2): 1-19.

[2] Chang W G, You T, Seo S, *et al*. Domain-specific batch normalization for unsupervised domain adaptation[C]//*Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. Washington D C: IEEE, 2019: 7354-7362.

[3] Zhao S C, Wang G Z, Zhang H H, *et al*. Multi-source distilling domain adaptation[C]// *Proceedings of the AAAI Conference on Artificial Intelligence*,2020, **34**(7):12975-12983.

[4] Peng X C, Huang Z J, Zhu Y Z, *et al*. Federated Adversarial Domain Adaptation[EB/OL]. [2019-05-15]. *https://. www. arXiv preprint arXiv*:1911.02054.

[5] Kairouz P, McMahan H B, Avent B, *et al*. Advances and open problems in federated learning[J]. *Foundations and Trends in Machine Learning*, 2021,**14**(1-2): 1-210.

[6] Karimireddy S P, Kale S, Mohri M , *et al*. SCAFFOLD: Stochastic Controlled Averaging for On-Device Federated Learning[EB/OL]. [2019-05-15]. *https://arxiv. org/abs/* 1910.06378.

[7] Li X, Huang K X, Yang W H, *et al*. On the Convergence of FedAvg on Non-iid Data[EB/OL]. [2019-04-27]. *https:// www.arXiv preprint arXiv*:1907.02189.

[8] Peng X C, Bai Q X, Xia X D, *et al*. Moment matching for multi-source domain adaptation[C]// *Proceedings of the IEEE/CVF International Conference on Computer Vision*. Wasnington D C: IEEE, 2019: 1406-1415.

[9] McMahan H B, Moore E, Ramage D, *et al*. Communication-efficient learning of deep networks from decentralized data [C]//*Artificial Intelligence and Statistics*. New York: PMLR, 2017: 1273-1282.

[10] Kallista B, Hubert E, Wolfgang G, *et al*. Towards federated learning at scale: System design[C]// *Proceedings of Machine Learning and Systems*, 2019, 1: 374-388. https://doi.org/10.48550/arXiv.1902.01046.

[11] Zhao H, Combes R T D, Zhang K, *et al*. On learning invariant representations for domain adaptation[C]// *International Conference on Machine Learning*. New York: PMLR, 2019: 7523-7532.

[12] Long M S, Cao Y, Wang J M, *et al*. Learning transferable features with deep adaptation networks[C]// *International Conference on Machine Learning*. New York: ACM, 2015, **37**: 97-105.

[13] Lee C Y, Batra T, Baig M H, *et al*. Sliced wasserstein discrepancy for unsupervised domain adaptation[C]//*Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. Washington D C: IEEE , 2019: 10285-10295.

[14] Chen D F, Mei J P, Wang C, *et al*. Online knowledge distillation with diverse peers[C]// *Proceedings of the AAAI Conference on Artificial Intelligence.* Washington D C: IEEE , 2020, **34**(4): 3430-3437.

[15] Feng H Z, You Z Y, Chen M H, *et al*. KD3A: Unsupervised multi-source decentralized domain adaptation via knowledge distillation[C]// *International Conference on Machine Learning*. New York: PMLR, 2021: 3274-3283.

[16] Zhu Z D, Hong J Y, Zhou J Y. Data-free knowledge distillation for heterogeneous federated learning[C]// *International*

Conference on Machine Learning. New York: PMLR, 2021, **139**: 12878-12889.

[17] Oord A V D, Li Y Z, Vinyals O. Representation Learning with Contrastive Predictive Coding[EB/OL]. [2018-03-28]. *https*://*arXiv* : 1807.03748.

[18] Bachman P, Hjelm R D, Buchwalter W. Learning Representations by Maximizing Mutual Information Across Views [EB/OL]. [2019-09-27]. *https*://*arxiv.org*/*abs*/1906.00910.

[19] Chen H Y, Chao W L. Fedbe: Making Bayesian Model Ensemble Applicable to Federated Learning[EB/OL]. [2020-02-15]. *https*://*arXiv preprint* arXiv:2009.01974.

[20] Lin T, Kong L J, Stich Sebastian U, *et al*. Ensemble distillation for robust model fusion in federated learning[J]. *Advances in Neural Information Processing Systems*, 2020, **33**: 2351-2363.

[21] Li Q B, He B S, Song D. Model-contrastive federated learning[C]//*Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. Washington D C: IEEE , 2021: 10713-10722.

[22] Liang J, Hu D P, Feng J S. Do we really need to access the source data source hypothesis transfer for unsupervised domain adaptation[C]//*Proceedings of the* 37*th International Conference on Machine Learning.* New York: ACM, 2020: 6028-6039.

[23] Geiping J, Bauermeister H, Dröge H, *et al*. Inverting gradients-how easy is it to break privacy in federated learning? [C]//*Proceedings of the* 34*th International Conference on Neural Information Processing Systems.* New York: ACM, 2020, **33**: 16937-16947.

□