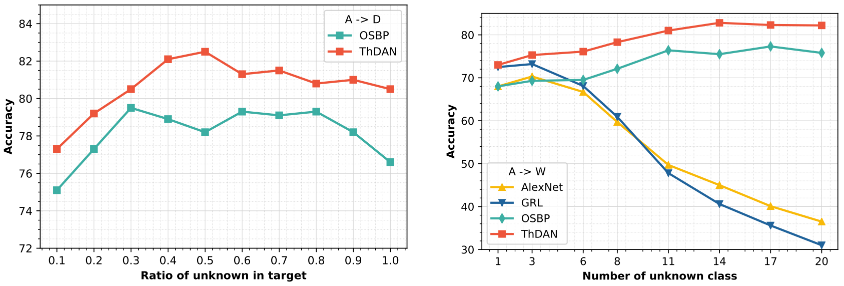
Reply to Reviewer#2

In the main idea, it is not clear the sampling proportion between the data from known and unknown classes. How does this proportion affect the subsequent training?

Response:

As you mentioned, it's essential to indicate the affection of samples from unknown classes in the setting of open set domain adaptation. However, in real scenarios, we can not manually set the sampling proportion due to the lack of target label information. During the training phase, ThDAN is confined to randomly sample training data from the target domain. So we conduct experiments to implicitly change the sampling proportion by varying the number of unknown data and the number of unknown classes. All experiments are conducted on Office-31 datasets. Results show as follows,



(a) Accury *w.r.t.* Ratio of Unknown Samples

(b) Accury *w.r.t.* Number of Unknown Classes

Figure 1: (**a**): The prediction accuracy of our method when we changed the ratio of unknown samples in the adaptation task $A \rightarrow D$. (**b**): The prediction accuracy of our method when we changed the number of unknown classes in the adaptation task $A \rightarrow W$.

We first fix the number of unknown classes to 10, and then change the sample number of each unknown class. **Ratio 0.1** of sample number indicates that only one-tenth samples of each unknown class are used for training and testing, while the others are discarded from the dataset. Figure1a shows the result. When the ratio is low, the model can hardly identify samples from unknown classes due to lack of training samples. On the contrary, when the ratio is high, the negative transfer of unknown class would undermine the classification of the known class. Therefore the performance of the model is defective around the endpoint of the interval.

We further vary the number of unknown classes in Figure1b. As the number of unknown classes increases, the performance of the methods proposed for close set domain adaptation will decrease significantly. That is because they cannot alleviate the negative transfer brought by the unknown classes. The model proposed for open set domain adaptation would gains performance improvement by correctly rejecting samples from unknown classes. That is because it's easier to reject samples as the "unknown" than categorize samples from the diverse known classes. The accuracy has the trend to decay since too many unknown classes would interfere with the model on classifying the known class.

The experiments show that the sampling proportion between the data from known and unknown classes could substantially affect model training for both known and unknown classes. Low proportion of sample from unknown classes would affect the discrimination of unknown samples, and high proportion of that would undermine classifications of samples from known classes.

From their reported results in Figure 7, the adjust threshold of γ_0 is not sensitive to the accuracy of training, it is a bit strange. Authors are encouraged to carefully check their claims by using more detailed reasons or more diverse data sets.

Response:

Thanks for your suggestions. In our model, the threshold is calculated by averaging the transferability score of a batch of source samples and the predefined γ as follows,

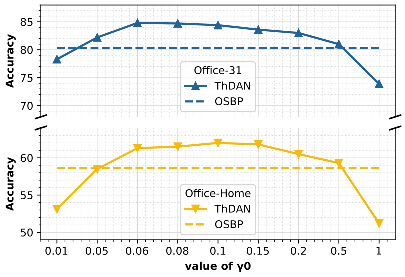
$$\beta(X_s, \gamma) = \mathbb{E}_{x \in X_s} w(x) - \gamma. \tag{1}$$

Then we dynamically increase γ from 0 to γ_0 so that more samples could be selected for adversarial training. The changing function of γ as follows,

$$\gamma = \begin{cases} 0 & , n \in N_1 \\ \gamma_0 \times \sigma(n) & , n \in N_2 \end{cases} \tag{2}$$

where γ is a *monotonically increasing function* with an upper bound of 1.

In experiments, we verify that the model is insensitivity to the value of γ_0 on **6 tasks of Office-31** and **12 tasks of Office-Home**. We include Figure7 of the original paper as follows, which shows the average prediction accuracy on these datasets when γ_0 varies,



(a) Accury *w.r.t.* value of γ_0

Figure 2: The prediction accuracy of our method when we change the value of σ_0 on dataset Office-31 and Office-Home.

Compared with the baseline model OSBP, our model can deliver plausible results on both Office-31 and Office-Home datasets when the γ_0 in the range of $[0.06, 0.5]$. That is because when we gradually increase the γ during training, we also gradually decrease the learning rate. As a result, the transferable samples selected in early stage would ‘dominate’ the model training, because they engage in training sooner, and the gradient produced by them can update the model with a greater learning rate. As the γ approaches γ_0, the learning rate is relatively small, and the previously selected samples would still be selected for training, thus the newcomer has little impact on model performance. This directly reflects on the insensitivity of value of γ_0 to the accuracy of training.

The above explanation has been added in the revised manuscript. Please refer to Section 4.3.6 in the revised manuscript for details.

1. **The abstract needs to be redesigned. Open set domain adaptation (OSDA) should be a scenario. Authors claim to solve the OSDA problem. What is the problem?? This is unclear. It seems the author should firstly explain the OSDA setting and then present which issue they want to address. The first sentence of the abstract is a bit far away from their focus. More seriously, the introduction don not connect the abstract. For example, the beginning of the abstract explains the unsupervised domain adaptation, but the beginning of the introduction suddenly use deep learning to start this paper without unsupervised domain adaptation. Authors are encouraged to carefully reorganize their work, where the introduction must connect the abstract tightly. One sentence in the abstract should connect one logic in the introduction section.**

Response:

Thanks for your precious suggestions, which is very helpful for us in improving paper writing. The **abstract**, **introduction** and **method** sections of revised manuscript have been reorganized for tighter logical connections.

The **abstract** has being redesigned as follows,

> *Unsupervised domain adaptation (UDA) is a paradigm to tackle domain shift problem. Most existing UDA methods are proposed for Close Set Domain Adaptation (CSDA) which assumes source and target domains share the same label space. More practically, target domain contains unknown class different from the known ones in the source domain, i.e., Open Set Domain Adaptation (OSDA). Due to the presence of unknown classes, aligning the whole distribution of the source and target domain for OSDA as in the previous method will lead to negative transfer. Existing methods developed for OSDA try to assign smaller weights to target samples of unknown class. Despite promising performance, the samples of unknown class are still used for training, which exposes the model to the risk of negative transfer. Instead of reweighting, this paper presents a novel method namely Thresholded Domain Adversarial Network (ThDAN), which progressively selects transferable target samples for distribution alignment. Based on the fact that samples from the known classes must be more transferable than target samples of the unknown one, we derive a criterion to quantify the transferability by constructing classifiers to categorize known classes and to discriminate unknown class. In ThDAN, an adaptive threshold is calculated by averaging transferability scores of source domain samples to select target samples for training. The threshold is tweaked progressively during the training process so that more and more target samples from the known classes can be correctly selected for adversarial training. Extensive experiments show that the proposed method outperforms state-of-the-art domain adaptation and open set recognition approaches on benchmarks.*

In the revised abstract, we first introduce the unsupervised domain adaptation (UDA). Then we make an introduction to the open set domain adaptation (OSDA) and the challenge of it. To clarify the motivation of our work, we then introduce the working mechanism of existing models proposed for OSDA and state the drawbacks of it. Finally, we present the general idea of the proposed Thresholded Domain Adversarial Network (ThDAN).

Following the logic of the redesigned abstract, we modified the **introduction** to make it more tightly connected to the abstract. For example, in the first paragraph of the introduction, we first introduce the domain shift problem, then lead to the topic of unsupervised domain adaptation, which is logically related to the first sentence of the abstract.

We also reorganized the subsections of **Method** section to make it more logical. In the revised manuscript, a section of the introduction of domain adversarial training [1] is added as the preliminary of the proposed model. Then in the next section, we introduce the general idea and training procedure of the proposed ThDAN. In the following sections, we detailed introduce the transferability based sample selection algorithm used in ThDAN. We consider such an organization can help readers better understand our work.

Please refer to Abstract, Section 1 and 3 in the revised manuscript for details.

2. What is ”transferable”? How to evaluate it? Any definition to support this term?

Response:

Unfortunately, there is no formal definition or mathematical formula to define “transferable” in the context of domain adaptation. Generally, the word “transferable’ is used to describe features. Here we quote from [2]:

> *the transferable features are the features that generalize well to novel tasks for domain adaption*

While the later work [3] uses “transferable” to describe samples that contribute to the transfer task of domain adaptation.

In the scenario of open set domain adaptation, we consider the target samples from the known classes are transferable samples since they can boost the classification performance of the model. On the contrary, the target samples from the unknown classes are considered as untransferable samples, that is because aligning distribution with them will incur negative transfer.

In this work, we evaluate transferability based on two observations:

Target samples from the known classes are able to confuse G_d. Here G_d is the domain discriminator that gives the probability of being target samples. For a target sample, if $G_d^*(z)$ approaches to 1, then the sample has a high probability of coming from the unknown classes. That is because the unknown classes only included in the target domain and can be almost perfectly discriminated from the source samples. On the other hand, if $G_d^*(z)$ approaches to 0, then the sample is more likely from the known classes that shared by domains. Therefore, the transferable target samples are able to confuse G_d to label them as the source samples. Then we can define the transferability $w_d(x)$ as inversely related to $G_d^*(z)$ as ,

$$w_d(x) = 1 - G_d^*(z). \tag{3}$$

Target samples from the known classes can be categorized by $G_{c,known}$. Here $G_{c,known}$ is the classifier that gives the probability distribution among known classes. Due to the overlapping in the marginal distributions, the target samples from the known classes should be categorized by the classifier G_c that trained on the source samples, leading to low classification entropy. And for samples from the unknown classes, because they cannot be aligned with a specific class from the source domain, the prediction tends to be uncertain over K known categories, leading to high classification entropy. Therefore the transferability $w_c(x)$ can be defined as inversely related to the *normalized* classification entropy H as,

$$w_c(x) = 1 - H(G_{c,\,known}(z)). \tag{4}$$

Since the domain discriminate G_d in Eq.3 and the classifier $G_{c,known}$ in Eq.4 work independently, we can unify the transferability criterion as,

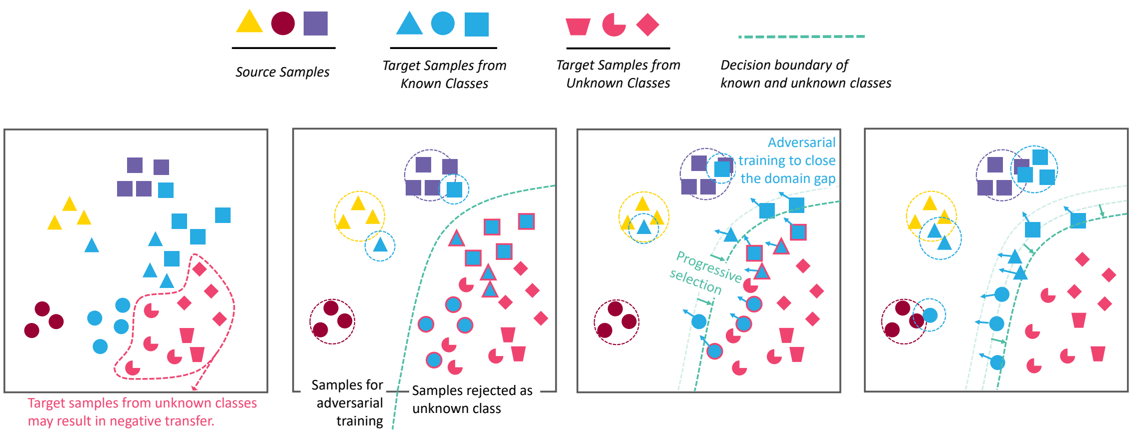
$$w(x) = 1 - G_d(z) \cdot H(G_{c,\,known}(z)). \tag{5}$$

The experiments in Section4 show that the transferability criterion of Eq.5 works well for our model to select target samples from the known classes.

3. In Figure1, it is unclear why (d) must outperform the other methods.

Response:

Thanks for your comments. The Figure1 in the original manuscript is as follows,


Source Samples
Target Samples from Known Classes
Target Samples from Unknown Classes
Decision boundary of known and unknown classes
Target samples from unknown classes may result in negative transfer.
Samples for adversarial training
Samples rejected as unknown class
Adversarial training to close the domain gap
Progressive selection

(a) (b) (c) (d)

Figure 3: Figure1 of the original manuscript.

The claim of Figure3 in the original paper could be confused. As a matter of fact, we want to use Figure3 to illustrate the general idea of the proposed Thresholded Domain Adversarial Network. To deliver the idea clearer, we revised the caption of Figure3 in the revised manuscript as follows,

> *The general idea of the proposed Thresholded Domain Adversarial Network (**ThDAN**). (**a**): Samples for the domain adaptation task. (**b**): The ThDAN builds a decision boundary based on the transferability threshold to separate known classes and unknown ones, then align the distributions of the source and the selected samples. (**c**): The ThDAN tweaks the transferability threshold to collect more samples of known classes to enhance distribution alignment. (**d**): The ThDAN progressively selects more known class samples for training. All the unselected samples are rejected as 'unknown'.*

4. What is ” domain-invariant features ”?

Response:

Generally speaking, the aligning distributions of the source and target domain will deliver domain-invariant features in feature space. In domain-invariant feature space, the source and target domains have the same (or similar) marginal distributions, and the posterior distributions of the labels are the same across domains too. Hence, a classifier trained on the labeled source would likely perform well on the target.

Specifically, the domain-invariant features can be obtained by utilizing domain adversarial training [1] to align the distribution of the source and target domain. The domain adversarial training has been clarified in Section 3.1 of the revised manuscript. The domain adversarial training is a two-player minmax game: the domain discriminator G_d as the first player aims to separate the feature representation of the source domain from the target domain, at the same time, feature generator G_f as the second player is trained to deceive the domain discriminator. Formally, the domain adversarial training can be written as:

$$\begin{aligned}\min_{G_f}\max_{G_d} \mathcal{L}(G_f,G_d) = {} & \mathbb{E}_{x\sim p_t(x)}\left[\log\left(G_d\left(G_f\left(x\right)\right)\right)\right] \\ & + \mathbb{E}_{x\sim p_s(x)}\left[\log\left(1-G_d\left(G_f\left(x\right)\right)\right)\right].\end{aligned} \tag{6}$$

The optimization of Eq.(6) makes G_f to generate features that invariant to domains, thus the classifier G_c that trained on the source domain can perform well for target samples by leveraging features extracted by G_f.

5. In the related work, CSDA does not have references.

Response:

Thanks for pointing out the mistakes. References [4, 5, 6] for Close Set Domain Adaptation (CSDA) has been cited in the revised manuscript.

6. In section 2.1, ”Another way to solve domain adaptation” is not suitable. Domain adaptation is a setting in detailed ML tasks or a probability distribution issue. Authors can obtain more information from S Ben-David’s paper.

Response:

Thanks for pointing out the mistakes. Ben-David’s paper [4] gives us a better understanding of domain adaptation and has been cited in the revised manuscript. The phrase “Another way to solve domain adaptation” has been revised as “Another approach for domain adaptation”. Similar mistakes have bee corrected in the revised manuscript.

7. Section 2.2 missed some new work from open-set domain adaptation.

Response:

Thanks for your suggestions. We add some new work [7, 8] of open set domain adaptation in Section2.2, they are respectively from ICML2019 and CVPR2019.

Further more, we implements *Factorized Representations For Open Set Domain Adaptation* ***(FRFOSDA)*** model proposed in [7] in experiments on **Office-Hone** and **Office31** for comparisons. Please refer to Section 4 for more details.

8. In Section 3.1, what is “untransferable ones”?

Response:

Thanks for your comments. This question is related to Question2. The untransferable one indicate the untransferable samples that will deteriorate model performance if we try to align distributions for them. On the contrary, we uses “transferable” to describe samples that contribute to transfer task of domain adaptation. Please refer to Question2 for more details of the definition and measure criterion of “transferable”.

9. In Eq.(2), more strong reasons need to be presented to explain the G_f and G_d. Why the use such settings to define them? Only based other’s work?

Response:

Thanks for your comments. The Eq.(2) in original manuscript is as follows,

$$G_d^*(z) = \frac{p_s(z)}{p_s(z) + p_t(z)}. \tag{7}$$

In the original manuscript, we use the statement " *the G_f and G_d can be considered as the generative network and discriminate network of GAN respectively* " to explains Eq.(7). We find this explanation is unconvincing, and has been removed from the revised manuscript. We give the proof of Eq.(7) in Section3.1 of the revised manuscript, please refer to the reply of the next question for details.

However, we still like to explain on why " *the G_f and G_d can be considered as the generative network and discriminate network of GAN respectively* ". The proposed model is built upon the *Domain Adversarial Neural Network* ***(DANN)*** [1] which aligns domain by performing domain adversarial trianing. The domain adversarial training is a two-player minmax game: the domain discriminator G_d as the first player aims to separate the feature representation of the source domain from the target domain, at the same time, feature generator G_f as the second player is trained to deceive the domain discriminator. Formally, the domain adversarial training can be written as:

$$\begin{aligned}\min_{G_f}\max_{G_d}\mathcal{L}(G_f,G_d) &= \mathbb{E}_{x\sim p_t(x)}\left[\log\left(G_d\left(G_f(x)\right)\right)\right] \\ &+ \mathbb{E}_{x\sim p_s(x)}\left[\log\left(1-G_d\left(G_f(x)\right)\right)\right].\end{aligned} \tag{8}$$

where G_f is the feature extractor for samples, and G_d is a binary domain classifier with all the source samples labelled as 0 and all the target samples labelled as 1. The training procedure of Eq.(8) is very similar to the training function of the original GAN [9], therefore the G_f and G_d can be considered as the generative network and discriminate network of GAN respectively.

10. In the following, why does G_d converge to the optimal?

Response:

The optimal G_d for Eq.(8) is Eq.(7). We give the proof as follows,

Proof. For any G_f, we train G_d to maximize Eq.(6):

$$\begin{aligned}\max_{G_d}\mathcal{L}(G_f,G_d) &= \int_x p_s(x)\log\left(G_d\left(G_f(x)\right)\right) + p_t(x)\log\left(1-G_d\left(G_f(x)\right)\right)\,dx. \\ &= \int_x p_s(x)\log\left(G_d(z)\right) + p_t(x)\log\left(1-G_d(z)\right)\,dx.\end{aligned} \tag{9}$$

For any $(a,b)\in\mathbb{R}^2\backslash\{0,0\}$, the function $y\to a\log(y)+b\log(1-y)$ achieves its maximum in $[0,1]$ at $\frac{a}{a+b}$. Therefore the optimal G_d is Eq.(6). □

The proof above has also been added in Section 3.1 of the revised manuscript.

11. In Eq.(2), how to define p_s and p_t?

Response:

Here p_s is probability distribution of source domain, and p_t is that of target domain. z is the deep represent of samples in feature space, *i.e.,* $z=G_f(x)$. Therefore $p_s(z)$ is the probability of source samples in feature space and $p_t(z)$ is that of target samples.

12. In Eq.(9), what is X_t^k?

Response:

Thanks for your comments. The Eq.(9) in the original manuscript is as follows,

$$\begin{aligned} X_t^k &= \{x|w(x) \geq \beta, x \in X_t\}, \\ X_t^u &= \{x|w(x) < \beta, x \in X_t\}. \end{aligned} \tag{10}$$

Here β is the transferability threshold, w is the transferability calculator. For a batch of target training data X_t, we use the threshold β to split it into two parts based on transferability. The first part is transferable target samples that selected as the samples from the known classes, denoted as X_t^k. And the second part is untransferable target samples that regraded as the samples from the unknown classes, denoted as X_t^u. For target samples in X_t^k, we train the model to align distribution with the source samples. For target samples in X_t^k, we train the G_d to identify them as "unknown" samples.

In the revised manuscript, we add Section 3.2 to give an overview idea of the proposed method, in which X_t^k and X_t^u are explained.

13. A same controversial issue appears again: Sample selection is the key step for this model to address open set domain adaptation.

Response:

Thanks for pointing out the mistakes. This sentence has been modified as " Sample selection is the key step for this model to tackle the challenge of open set domain adaptation "

Similar mistakes have bee corrected in the revised manuscript.

14. In the experiments, why do you use different setting to begin the experiments in Section 4.2.1? Authors are encouraged to explain the inherent reasons.

Response:

Thanks for your suggestions.

We want to perform an ablation study to evaluate the efficacy of the proposed threshold tweaking techniques proposed in Section3.5. Ablation studies on Office-31, OfficeHome and VisDA datasets involve 3 kinds of ThDAN variants. (**1**) ***ThDAN-m-dy*** is the original setting which calculates transferability threshold based on mini-batch samples and a fixed γ_0. (**2**) ***ThDAN-dy*** is the variant applies exponential moving average to update transferability threshold. (**3**) ***ThDAN*** further takes advantage of dynamic γ to tweak transferability threshold.

We found the writing of the ablation study in the original is confusing since it separated in different subsections. In the revised manuscript, we unify the ablation study and reorganized in one particular section, please refer to Section 4.3.1 for details.

15. A same issue appear again in the settings of Section 4.2.2.

Response:

Thanks for your comments.

This question is related to the previous question. In the revised manuscript, the ablation study in the original Section 4.2.2 has been reorganized to Section 4.3.1.

16. The results analysis in Section 4.2.3 is unclear. More detailed reasons need to be present to support your conclusion.

Response:

The results of Section 4.2.3 are as follows,

Table 1: Accuracy (%) of each method with 6 shared class on VisDA.

Method	class-wise accuracy on VisDA								
	Bcycle	Bus	Car	Mcycle	Train	Truck	Unkown	*Avg.All*	*Avg.Known*
OSVM [ECCV, 2014]	4.8	45.0	44.2	43.5	59.0	10.5	57.4	*37.8*	*34.5*
MMD [NIPS, 2007]	0.2	30.9	49.1	54.8	56.1	8.1	61.3	*37.2*	*33.2*
GRL [JMLR, 2016]	9.1	50.5	53.9	79.8	69.0	8.1	42.5	*44.7*	*45.1*
OSBP [ECCV, 2018]	48.0	67.4	39.2	80.2	69.4	**24.9**	80.3	*58.5*	*54.8*
ThDAN-dy	52.6	69.5	58.8	83.1	72.3	12.3	84.5	*61.8*	*58.1*
ThDAN	**55.2**	**70.8**	**61.4**	**85.6**	**74.8**	10.0	**88.8**	***63.8***	***59.6***

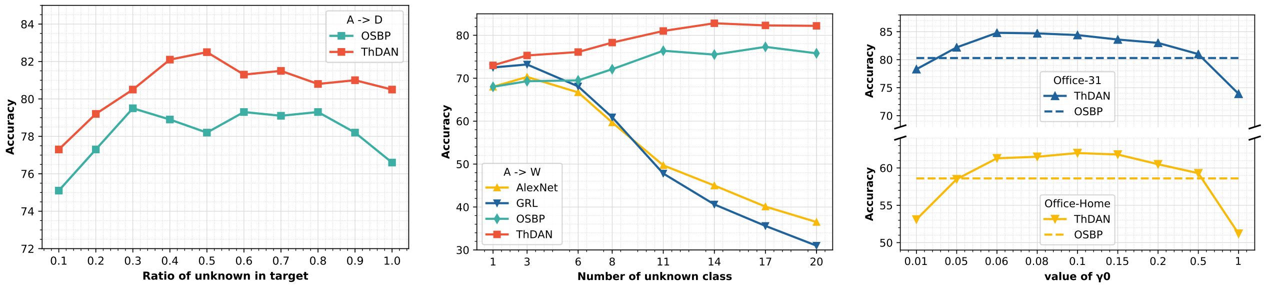
In the experiments of VisDA, the training split is used as the source domain and validation one as the target domain. We choose 6 categories, i.e., bicycle, bus, car, motorcycle, train and truck as the known classes, and the other 6 categories as the unknown classes. As Table. 2.18.1 shows the classification results, where *Avg.All* and *Avg.Known* indicate the accuracy averaged over all classes and known classes. For classification of known classes, ThDAN could exceeds other methods in almost every classes and in the average, which means our method can effetely transfer knowledge to the known classes. Also, our model improves the classification accuracy for Unknown classes by a big margin. It states that our method can perform well on both labeling the samples from the unknown classes and classifying the samples from the unknown classes. Also, the ablation study shows the superiority of dynamically adjusting threshold during training.

The above analysis of experiments in VisDA has been added into the corresponding section in the revised manuscript.

17. Please carefully check the claim in Figure 7.

Response:

Thanks for your reminding. The Figure 7 in the original manuscript is as follows,



(a) Accury *w.r.t.* Ratio of Unknown Samples

(b) Accury *w.r.t.* Number of Unknown Classes

(c) Accury *w.r.t.* value of γ_0

Figure 4: Figure of the original paper.

In the revised manuscript, the claim of it has been revised as,

> (***a***): *The prediction accuracy of our method when we changed the ratio of unknown samples in the adaptation task A→D.* (***b***): *The prediction accuracy of our method when we changed the number of unknown classes in the adaptation task A→W.* (***c***): *The prediction accuracy of our method when we change the value of* σ_0 *on dataset Office-31 and Office-Home.*

18. Some equations missed "," or "." at their ends.

Response:

Thanks for pointing out the mistakes. The equations in the revised manuscript have been corrected.

References

[1] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, V. Lempitsky, Domain-adversarial training of neural networks, The Journal of Machine Learning Research 17 (1) (2016) 2096–2030.

[2] M. Long, Y. Cao, J. Wang, M. I. Jordan, Learning transferable features with deep adaptation networks, arXiv preprint arXiv:1502.02791 (2015).

[3] X. Wang, L. Li, W. Ye, M. Long, J. Wang, Transferable attention for domain adaptation, in: Thirty-Third AAAI Conference on Artificial Intelligence, 2019.

[4] S. Ben-David, J. Blitzer, K. Crammer, A. Kulesza, F. Pereira, J. W. Vaughan, A theory of learning from different domains, Machine learning 79 (1-2) (2010) 151–175.

[5] M. Wang, W. Deng, Deep visual domain adaptation: A survey, Neurocomputing 312 (2018) 135–153.

[6] S. J. Pan, Q. Yang, A survey on transfer learning, IEEE Transactions on Knowledge and Data Engineering 22 (10) (2010) 1345–1359.

[7] M. Baktashmotlagh, M. Faraki, T. Drummond, M. Salzmann, Learning factorized representations for open-set domain adaptation, arXiv preprint arXiv:1805.12277 (2018).

[8] H. Liu, Z. Cao, M. Long, J. Wang, Q. Yang, Separate to adapt: Open set domain adaptation via progressive separation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 2927–2936.

[9] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, Generative adversarial nets, in: Advances in Neural Information Processing Systems, 2014, pp. 2672–2680.