CSG: COMPLEMENTARY SLACKNESS GUIDANCE FOR SOLVING OPTIMAL TRANSPORT PROBLEM

Anonymous authors
Paper under double-blind review

ABSTRACT

Optimal Transport (OT) has become a cornerstone mathematical tool in machine learning for measuring discrepancies between probability distributions. Directly solving the optimal transport problem can be relatively difficult with high complexity. Conventional solvers often employ relaxation approaches (i.e., entropic regularization) to achieve estimated matching results. However, these methods frequently yield dense and ambiguous outputs and thereby degrading the model performance. To address this issue, we propose the Complementary Slackness Guidance (CSG) approach to tackle the OT problem. That is, CSG introduces a novel Conditional Regularization Mask (CRM) that incorporates multiplier information by enlarging the irrelevant cost distances, thereby enabling more precise solutions. Specifically, CSG involves exploration and utilization stages to further estimate and leverage the Lagrange multipliers for the whole computation. We conduct several numerical experiments to demonstrate the effectiveness of our proposed CSG methods with the CRM term in addressing OT problems.

1 INTRODUCTION

The Optimal Transport (OT) technique is a fundamental mathematical tool for measuring domain discrepancies between source and target domains. OT has emerged as a versatile tool and has been extensively applied in numerous fields, including domain adaptation Damodaran et al. (2018); Xu et al. (2020); Wang et al. (2024), representation learning Wang et al. (2022); Liu et al. (2023b; 2025); Kerdoncuff et al. (2021), vision-language alignment Chen et al. (2023); Shi et al. (2024); Cao et al. (2024); Kassraie et al. (2024), and generative models Sanjabi et al. (2018); Gu et al. (2023); Kornilov et al. (2024). However, directly employing traditional linear programming methods Liu & Nocedal (1989) to solve the optimal transport problem incurs a high cost with a cubic time complexity Montesuma et al. (2024). Therefore, developing efficient approximation techniques that incorporate more useful information for solving OT problems while maintaining a high matching accuracy remains a major challenge.

Although several conventional OT solvers have attempted to address this issue, these approaches cannot effectively balance accuracy and efficiency. To improve computational efficiency, Cuturi (2013) first introduced an entropic regularization term and proposed the Sinkhorn algorithm to efficiently obtain matching results. However, the Sinkhorn algorithm inevitably leads to dense and inaccurate solutions Liu et al. (2023a); Torres et al. (2021), further deteriorating the model performance. To achieve sparser and more accurate transport plans, various regularization terms have been investigated, including the square norm Blondel et al. (2018) and q-deformed entropy Bao & Sakaue (2022). Recently, Tang & Qiu (2024) proposed the Safe and Sparse Newton Method (SSNS), which adopts the sparsification mechanism Tang et al. (2023) in the Hessian matrix to avoid ambiguous results. These sparsity-promoting approaches competitively improve the quality and interpretability of the coupling outputs by filtering out irrelevant or low-confidence matching pairs. Nevertheless, these methods can be relatively complex and time-consuming (e.g., solving the Hessian linear system), which limits their potential. Moreover, these approaches are always sensitive to the given hyperparameters (e.g., determining the sparsity threshold Ouyang et al. (2026)), making it difficult to implement them in real-world applications. Therefore, leveraging inherent optimization guidance to find precise solutions for the optimal transport problem requires further investigation.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

In this paper, we propose a novel method called Complementary Slackness Guidance (CSG) to obtain clearer and more precise matching results for optimal transport problems. Specifically, we propose a Conditional Regularization Mask (CRM) that leverages the useful KKT complementary slackness information encoded in the multipliers to guide the entire optimization process. Incorporating the CRM term yields several important benefits: (1) the cost distances of irrelevant pairs are enlarged, and (2) the cost distances of matched pairs are further reduced. Leveraging the above properties can avoid ambiguous outputs and lead to more precise matching results which mitigates the negative effects introduced by relaxation terms (e.g., entropic or ℓ_2-norm regularization). The CSG solver includes two main stages, namely exploration and utilization, for the entire optimization scheme. In the exploration stage, the CSG exploits the dual variables of the OT problem via the proposed Centering-BFGS method to achieve a smooth approximation. Meanwhile, CSG is equipped with the CRM term in the utilization stage to achieve the final OT solutions. Moreover, we provide a sufficient approximate error bound analysis (e.g., multiplier estimation) to verify the efficacy of our approaches. Notably, the proposed CSG is model-agnostic and easy to implement across a wide range of real-world applications to boost model performance.

We summarize our main contributions as follows: (1) We propose a novel solver framework, that is, Complementary Slackness Guidance (CSG), to solve the optimal transport problem. That is, the proposed CSG solver will enlarge the irrelevant cost distance and yield clearer and more precise matching results. (2) We first introduce the technical details of CSG by designing a novel Conditional Regularization Mask (CRM), which consists of an exploration stage and a utilization stage. These two stages enable the effective estimation and leveraging of the multiplier information throughout the entire optimization process. (3) We conduct extensive experiments on both synthetic and real-world datasets (e.g., for unsupervised domain adaptation and cross-domain retrieval) to demonstrate the performance of the proposed CSG algorithm.

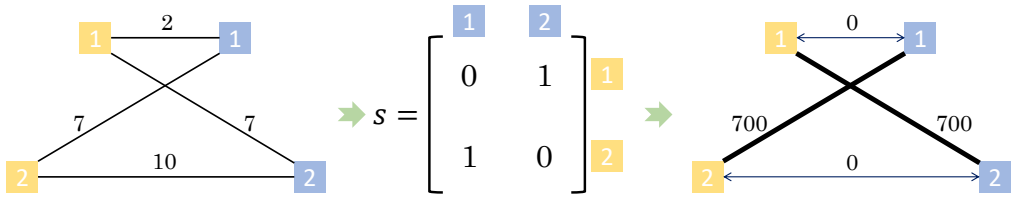
2 Related Works

The Optimal Transport (OT) technique is well-positioned as a powerful mathematical tool to measure distances between different probability distributions Gasteiger et al. (2021); Li et al. (2023). However, the exact optimal transport problem is typically formulated as a linear program with prohibitive computational complexity Wu et al. (2025); Chewi et al. (2025) and thus limits its generalization. To address this cost and make OT viable for modern machine learning tasks, researchers have developed various regularized formulations and efficient numerical solvers Montesuma et al. (2024); Tang et al. (2024); An et al. (2022); Xie et al. (2020); Gracyk & Chen (2022). A prominent milestone in this direction is the work of Cuturi (2013), who first involved an entropic regularization term into the Kantorovich formulation. This smoothed objective enables the use of the Sinkhorn algorithm, which computes the transport plan through highly iterative matrix-vector multiplications. To promote sparser and more interpretable transport plans, various additional regularization terms have been introduced, including quadratic matrix-norm regularization Blondel et al. (2018); Lorenz et al. (2021) and q-deformed entropy Bao & Sakaue (2022). More recently, researchers have proposed methods that incorporate sparsity constraints into the Newton descent framework, such as the Safe and Sparse Newton Method (SSNS) Tang & Qiu (2024) to achieve higher precision in optimal transport solutions. However, these approaches heavily depend on manually specified thresholds and suffer from high computational complexity (e.g., finding the inverse matrix) which significantly limits their practical potential Boyd & Vandenberghe (2004). Therefore, achieving precise and sparse matching results without sacrificing computational efficiency remains a challenge in the optimal transport literature.

3 Methodology

In this section, we present a detailed description of the Complementary Slackness Guidance (CSG) approach equipped with a Conditional Regularization Mask (CRM) for solving the optimal transport problem. To start with, we first consider two sets of data samples $\boldsymbol{X} \in \mathbb{R}^{M\times D}$ and $\boldsymbol{Z} \in \mathbb{R}^{N\times D}$ in domain X and domain Z, where M, N denote the number of samples and D denotes the data dimension. Then we can reach the definition of optimal transport as shown:

$$\begin{aligned} &\min_{\boldsymbol{\pi}\in\Delta} L_{\mathrm{OT}} = \langle \boldsymbol{C}, \boldsymbol{\pi} \rangle \\ &s.t.\ \Delta = \{\boldsymbol{\pi}\mathbf{1}_N = \boldsymbol{\alpha}, \quad \boldsymbol{\pi}^\top \mathbf{1}_M = \boldsymbol{\beta}, \quad \boldsymbol{\pi} \geq 0\}, \end{aligned} \tag{1}$$



(a) Original cost matrix $\boldsymbol{C}$ (b) Multiplier $\boldsymbol{s}$ (c) Update cost matrix

Figure 1: The toy example for the proposed CSG solver with the conditional regularization mask.

where $\boldsymbol{C} \in \mathbb{R}^{M\times N}$ and $\boldsymbol{\pi} \in \mathbb{R}^{M\times N}$ denote the pairwise distances and matching matrix among different data samples respectively. $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$ denote the marginal probabilities for each data samples and $\boldsymbol{\pi}$ should obey the boundary constraints Δ on both the rows and columns of $\boldsymbol{\pi}$. The corresponding dual Fenchel-Lagrange dual form can be further expressed as:

$$\begin{aligned} &\max_{\boldsymbol{f},\boldsymbol{g}} \left[\langle \boldsymbol{f}, \boldsymbol{\alpha}\rangle + \langle \boldsymbol{g}, \boldsymbol{\beta}\rangle\right] \\ &s.t.\ f_i + g_j + s_{ij} = C_{ij}, \quad s_{ij} \geq 0, \end{aligned} \tag{2}$$

where $\boldsymbol{f}$, $\boldsymbol{g}$ and $\boldsymbol{s}$ denote the Lagrange multipliers and it satisfies KKT condition $\pi_{ij}s_{ij} = 0$. However, it is rather time-consuming to directly optimize L_{OT} via linear programming Liu & Nocedal (1989). Previous researchers always involve an entropy regularization term (e.g., Sinkhorn Cuturi (2013)) to accelerate the optimization process. However, such a regularization term could easily result in ambiguous matching results Blondel et al. (2018) and lead to the negative transfer in real practice. To tackle this issue, we first propose the Complementary Slackness Guidance method for solving the optimal transport problem as **CSG**. We will introduce the motivation and technical details of CSG solver for OT in the following section.

3.1 CSG for Optimal Transport

3.1.1 Motivation for the CSG solver

In this section, we start by motivating the CSG solver which seeks to provide more precise solutions for the classical optimal transport problem. Specifically, CSG is motivated by the KKT complementary slackness condition (i.e., $\pi_{ij}s_{ij} = 0$) and incorporates the multiplier $\boldsymbol{s}$ from Eq.equation 2. In particular, multipliers inherit key properties from this relationship: when $s_{ij} = 0$ implies $\pi_{ij} > 0$, and conversely, when $\pi_{ij} = 0$ implies $s_{ij} > 0$. Therefore, considering $s_{ij}\pi_{ij}$ information can provide informative matching guidance and lead to more precise results. To further illustrate the key insights of the CSG solver, we provide a demonstration as shown in Fig.1. As we mask the cost matrix $\boldsymbol{C}$ with its multiplier $\boldsymbol{s}$, the irrelevant cost distances are enlarged while the remaining ones drop to zero. The whole process is targeted while avoiding the uniform scaling of all elements for the cost matrix. Unfortunately, computing the exact value of $\boldsymbol{s}$ as defined in Eq.equation 2 is time-consuming and rather difficult in practice. To tackle this issue, we first propose CSG solver with exploration and utilization stages to provide better estimation of $\boldsymbol{s}$. Specifically, CSG solver first explores the dual variables of the optimal transport problem L_{OT} in Eq.equation 2 using an efficient iterative procedure. Subsequently, CSG utilizes these dual variables and multipliers by incorporating a newly proposed Conditional Regularization Mask (CRM) into the original objective for more accurate solutions. By fully exploiting these useful inherit guidances, we can obtain more accurate matching solutions.

3.1.2 Exploration Stage of CSG solver

To start with, we first introduce the exploration stage of the CSG solver. Specifically, this stage is set to exploit the dual variables of Eq.equation 2 by reference to the following Lemma 1:

Lemma 1. *Given the dual formulation in Eq.equation 2, we adopt the c-transform* $g_j = \inf_{k\in[M]}\left(C_{kj} - f_k\right)$ *without loss of optimality Villani et al. (2009). Substituting this relation into the dual problem eliminates the explicit constraints and yields the unconstrained formulation:*

$$\min_{\boldsymbol{f}} \mathcal{J}_{\mathrm{OT}} = \sum_{j=1}^{N} \beta_j \sup_{k\in[M]} \left(f_k - C_{kj}\right) - \sum_{i=1}^{M} f_i\alpha_i \quad . \tag{3}$$

Although $\mathcal{J}_{\mathrm{OT}}$ is convex without any constraints, the presence of $\inf(\cdot)$ renders it a non-smooth function which preventing efficient optimization. Moreover, $\mathcal{J}_{\mathrm{OT}}$ is translation invariance which leads to multiple solutions. To further accelerate the entire optimization process, we adopt the smooth approximation of replacing $\sup(\cdot)$ with $\mathrm{LogSumExp}(\cdot)$ following Nesterov (2005) in CSG with given ϵ as $|\epsilon \log[\sum_{k=1}^{M} \exp((f_k - C_{kj})/\epsilon)] - \sup_{k\in[M]}[f_k - C_{kj}]| \leq \epsilon \log M$. Apparently, smaller ϵ (e.g., $\epsilon \to 0$) will lead to a tighter approximation. Hence we can obtain the approximate dual Fenchel-Lagrange form of optimal transport as shown:

$$\min_{\widehat{\boldsymbol{f}}^\top \mathbf{1}_{\mathrm{M}}=0} \mathfrak{J}_{\mathrm{OT}} = \sum_{j=1}^{N} \epsilon\beta_j \log\left[\sum_{i=1}^{M} \exp\left(\frac{\widehat{f}_i - C_{ij}}{\epsilon}\right)\right] - \sum_{i=1}^{M} \widehat{f}_i \alpha_i. \tag{4}$$

Meanwhile the centering constraint $\widehat{\boldsymbol{f}}^\top \mathbf{1}_{\mathrm{M}} = 0$ can avoid multiple solutions for optimization.

Proposition 1. *The objective $\mathfrak{J}_{\mathrm{OT}}$ in Eq. equation 4 has a $1/\epsilon$-Lipschitz continuous gradient. $\mathfrak{J}_{\mathrm{OT}}$ is μ-strongly convex restricted to the subspace $\mathcal{M} := \{\widehat{\boldsymbol{f}}^\top \mathbf{1}_{\mathrm{M}} = 0\}$ where $\mu = \lambda_{\min}\big(\nabla^2 \mathfrak{J}_{\mathrm{OT}}\big|_{\mathcal{M}}\big)$.*

The detailed proof of Proposition 1 can be found in Appendix. CSG proposes the Centering-BFGS method to optimize Eq.equation 4 as shown in Alg.1 with the computational analysis in the remark:

Remark. *The optimization of Centering-BFGS satisfies the following inequality Jin et al. (2024) as:*

$$\frac{\mathfrak{J}_{\mathrm{OT}}(\widehat{\boldsymbol{f}}^{(k)}) - \mathfrak{J}_{\mathrm{OT}}(\widehat{\boldsymbol{f}}^{*})}{\mathfrak{J}_{\mathrm{OT}}(\widehat{\boldsymbol{f}}^{(0)}) - \mathfrak{J}_{\mathrm{OT}}(\widehat{\boldsymbol{f}}^{*})} \leq \left(1 - \frac{2\omega_a(1-\omega_b)}{L/\mu}\right)^k \tag{5}$$

where ω_a and ω_b denote the line search parameters as:

$$\begin{aligned} \mathfrak{J}_{\mathrm{OT}}(\widehat{\boldsymbol{f}}^{(k)} + \eta_t \boldsymbol{d}_t) &\leq \mathfrak{J}_{\mathrm{OT}}(\widehat{\boldsymbol{f}}^{(k)}) + \omega_a \eta_t \nabla \mathfrak{J}_{\mathrm{OT}}(\widehat{\boldsymbol{f}}^{(k)})^\top \boldsymbol{d}_t \\ \nabla \mathfrak{J}_{\mathrm{OT}}(\widehat{\boldsymbol{f}}^{(k)} + \eta_t \boldsymbol{d}_t)^\top \boldsymbol{d}_t &\geq \omega_b \nabla \mathfrak{J}_{\mathrm{OT}}(\widehat{\boldsymbol{f}}^{(k)})^\top \boldsymbol{d}_t \end{aligned} \tag{6}$$

where $\boldsymbol{d}_t$ denotes the search direction. This indicates that the algorithm achieves superlinear convergence Jin et al. (2024) since it is strongly convex with $1/\epsilon$-Lipschitz continuous gradient.

Note that the proposed Centering-BFGS approach differs from the standard BFGS in the following aspects. It adopts a more effective initialization for the approximated inverse Hessian matrix by setting $\boldsymbol{H}_0 = \epsilon \boldsymbol{I}$. This initialization not only exploits the inherent $1/\epsilon$-Lipschitz continuity of the gradient but also avoids the search directions deviating from the true descent path during the calculation Jin et al. (2024). At the k-th iteration, we compute $\widehat{\boldsymbol{f}}^{(k+1)} = \widehat{\boldsymbol{f}}^{(k)} - \zeta_k \boldsymbol{H}_k \nabla \mathfrak{J}_{\mathrm{OT}}(\widehat{\boldsymbol{f}}^{(k)})$ using a step size ζ_k obtained via the Armijo–Wolfe line search, and subsequently update $\boldsymbol{H}_k$ according to the BFGS update rule. Such simple settings can provide valuable characteristics that ensure a global superlinear convergence rate via Proposition 2:

Proposition 2. *$\widehat{\boldsymbol{f}}^{(k+1)}$ restricted to the subspace $\mathcal{M}$ as $\sum_{i=1}^{M} \widehat{f}_i^{(k+1)} = 0$ on Centering-BFGS.*

The proof of Proposition 2 is provided in Appendix. This observation is particularly novel and interesting because the constraints are automatically satisfied without requiring any additional operations on Centering-BFGS. Once the optimal $\widehat{\boldsymbol{f}}$ is obtained, the multiplier $\widehat{\boldsymbol{s}}$ can be further estimated via:

$$\widehat{s}_{ij} = C_{ij} - \widehat{f}_i + \epsilon \log\left[\sum_{k=1}^{M} \exp\left(\frac{\widehat{f}_k - C_{kj}}{\epsilon}\right)\right] \geq 0. \tag{7}$$

Moreover, we also provide a detailed error analysis between the approximate results $\widehat{\boldsymbol{f}}$, $\widehat{\boldsymbol{s}}$ on Eq.equation 4 and the exact results $\boldsymbol{f}$, $\boldsymbol{s}$ on Eq.equation 2 within the following Proposition 3:

Proposition 3. *Given the optimal solutions of $\boldsymbol{s}^*$ in $\mathcal{J}_{\mathrm{OT}}$ and $\widehat{\boldsymbol{s}}^*$ in $\mathfrak{J}_{\mathrm{OT}}$, the general error bound is:*

$$0 \leq \left\langle \boldsymbol{\alpha} \otimes \boldsymbol{\beta}, \widehat{\boldsymbol{s}}^* - \boldsymbol{s}^* \right\rangle \leq \epsilon \log M. \tag{8}$$

More specifically, the ℓ_1-error bound between $\boldsymbol{s}^*$ and $\widehat{\boldsymbol{s}}^*$ is given as:

$$\left\| \widehat{\boldsymbol{s}}^* - \boldsymbol{s}^* \right\|_1 \leq N \sqrt{\frac{2\epsilon M \log M}{\mu}} + NM \sqrt{\frac{2\epsilon \log M}{\mu}} + \epsilon NM \log M \tag{9}$$

The detailed proof of Proposition 3 can be found in Appendix. That is, we observe that as the regularization parameter ϵ decreases (and particularly as $\epsilon \to 0$), the approximation $\widehat{\boldsymbol{s}}^*$ becomes increasingly

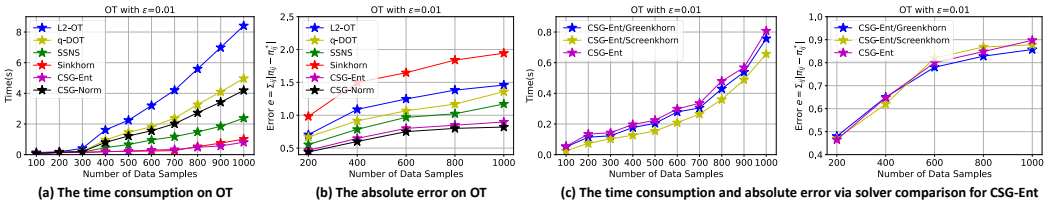


Figure 2: The time consumption and error analysis on classic optimal transport on synthetic datasets.

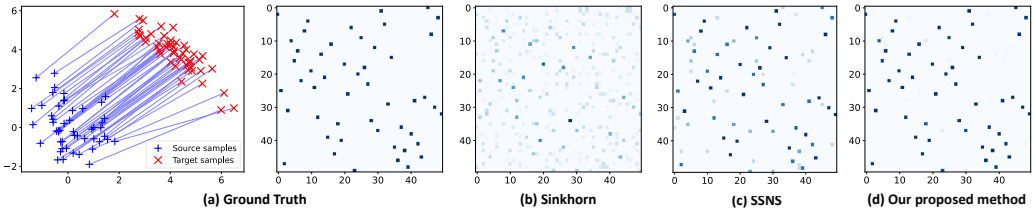


Figure 3: The optimal transport plans $\boldsymbol{\pi}^*$ produced by the Sinkhorn Cuturi (2013), the Safe and Sparse Newton Method (SSNS) Tang & Qiu (2024), and our proposed method CSG-Ent with $L_{\text{Reg}}(\boldsymbol{\pi}) = \langle \boldsymbol{\pi}, \log \boldsymbol{\pi} - \mathbf{1} \rangle$. Experiments are conducted with $\eta = 100$ and $\epsilon = 0.01$.

close to the true optimal multiplier $\boldsymbol{s}^*$. This behavior empirically confirms the effectiveness of the exploration stage in the CSG solver with the smoothness approximation.

Algorithm 1 Centering-BFGS approach for the exploration stage of CSG solver

1: Initialize the value of $\widehat{\boldsymbol{f}}^{(0)} = \mathbf{0}, \boldsymbol{H}_0 = \epsilon \boldsymbol{I}$.
2: **for** $k = 0$ to T **do**
3: Update $\widehat{\boldsymbol{f}}^{(k+1)} = \widehat{\boldsymbol{f}}^{(k)} - \zeta_k \boldsymbol{H}_k \nabla \mathfrak{J}_{\text{OT}}(\widehat{\boldsymbol{f}}^{(k)})$ where ζ_k obtained via Armoji-Wolfe line search.
4: Update matrix $\boldsymbol{H}_{k+1}$ with $\boldsymbol{v}_k = \widehat{\boldsymbol{f}}^{(k+1)} - \widehat{\boldsymbol{f}}^{(k)}$ and $\boldsymbol{y}_k = \nabla \mathfrak{J}_{\text{OT}}(\widehat{\boldsymbol{f}}^{(k+1)}) - \nabla \mathfrak{J}_{\text{OT}}(\widehat{\boldsymbol{f}}^{(k)})$:

$$\boldsymbol{H}_{k+1} = \left(\boldsymbol{I} - \frac{\boldsymbol{v}_k \boldsymbol{y}_k^\top}{\boldsymbol{v}_k^\top \boldsymbol{y}_k} \right) \boldsymbol{H}_k \left(\boldsymbol{I} - \frac{\boldsymbol{y}_k \boldsymbol{v}_k^\top}{\boldsymbol{v}_k^\top \boldsymbol{y}_k} \right) + \frac{\boldsymbol{v}_k \boldsymbol{v}_k^\top}{\boldsymbol{v}_k^\top \boldsymbol{y}_k} \tag{10}$$

5: **end for**
Ensure: The optimal solution $\widehat{\boldsymbol{f}}$.

3.1.3 Utilization Stage of CSG solver

Since we have obtained the multipliers (i.e., $\widehat{\boldsymbol{f}}$, $\widehat{\boldsymbol{g}}$ and $\widehat{\boldsymbol{s}}$) via the exploration step in CSG, we aim to utilize this useful information to figure out the precise matching solutions on $\boldsymbol{\pi}$. That is, we innovatively propose the Conditional Regularization Mask (CRM) as $\eta \widehat{\boldsymbol{s}}$ for the utilization step in CSG to find the results of $\boldsymbol{\pi}$ as shown in the Definition 1:

Definition 1. *Given the approximate multipliers $\widehat{\boldsymbol{s}}$ obtained via Proposition 2, one can incorporate a conditional regularization mask into the optimal transport problem:*

$$\min_{\boldsymbol{\pi} \in \Delta} \widehat{L}_{\text{R}} = \langle \boldsymbol{C} \odot \eta \widehat{\boldsymbol{s}}, \boldsymbol{\pi} \rangle + \epsilon L_{\text{Reg}}(\boldsymbol{\pi}), \tag{11}$$

where η denotes the hyperparameter for the conditional regularization mask on the cost matrix.

Note that the CRM term leverages the information from these obtained dual variables and is straightforward to implement. In this utilization stage, we adopt Proposition 4 to optimize Eq.equation 11:

Proposition 4. *The dual Fenchel-Lagrange form of $\widehat{L}_{\text{R}}$ with multipliers $\boldsymbol{\phi}$ and $\boldsymbol{\psi}$ can be obtained:*

$$\max_{\boldsymbol{\psi}, \boldsymbol{\phi}} \mathcal{L} = \langle \boldsymbol{\alpha}, \boldsymbol{\psi} \rangle + \langle \boldsymbol{\beta}, \boldsymbol{\phi} \rangle - \epsilon \sum_{i=1}^{M} \sum_{j=1}^{N} L^*_{\text{Reg}} \left(\frac{\psi_i + \phi_j - \eta \widehat{s}_{ij} C_{ij}}{\epsilon} \right), \tag{12}$$

where $L^*_{\text{Reg}}(\cdot)$ denotes its conjugate function. We can adopt alternative minimization scheme to optimize $\boldsymbol{\phi}$ and $\boldsymbol{\psi}$. Thus the matching results can be determined via solving the following equation $\nabla_{\pi_{ij}} L_{\text{Reg}}(\pi_{ij}) = (\psi_i + \phi_j - \eta \widehat{s}_{ij} C_{ij})/\epsilon$ accordingly.

Specifically, one can iteratively optimize $\mathcal{L}$ by alternately solving the stationarity conditions $\nabla_{\boldsymbol{\phi}} \mathcal{L} = 0$ and $\nabla_{\boldsymbol{\psi}} \mathcal{L} = 0$. The utilization stage of CSG solver aligns the KKT complementary condition of the

Table 1: Classification accuracy (%) on *Office-31* and *ImageCLEF* for unsupervised domain adaptation

Method for UDA	A→W	D→W	W→D	A→D	D→A	W→A	Avg	I→P	P→I	I→C	C→I	C→P	P→C	Avg
Source-only	82.3	84.5	99.8	98.0	75.6	74.5	85.8	74.8	83.9	91.5	78.0	65.5	91.3	80.7
DANN	84.5	78.6	99.6	96.8	63.6	62.8	81.6	75.0	86.0	96.2	87.0	74.3	91.5	85.0
CDAN	94.1	92.9	100.0	98.6	71.0	69.3	87.7	77.7	90.7	97.7	91.3	74.2	94.3	87.7
DeepJDOT	88.9	88.2	99.6	98.5	72.1	70.1	86.2	77.5	90.5	95.0	88.3	74.9	94.2	86.7
RWOT	95.1	94.5	**100.0**	**99.5**	77.5	77.9	90.8	81.3	92.9	97.9	92.7	79.1	96.5	90.0
PPOT	94.6	96.1	**100.0**	98.5	86.3	85.4	93.5	82.0	95.0	98.2	96.1	81.8	97.8	91.9
DeepJDOT+CSG-Ent	90.5	90.0	**100.0**	98.8	75.4	73.6	88.1	79.2	91.1	95.6	89.2	76.0	94.9	87.7
PPOT+CSG-Ent	95.0	96.6	**100.0**	98.9	87.2	86.8	94.1	82.7	95.5	98.4	96.6	82.5	98.1	92.3
DeepJDOT+CSG-Norm	91.2	90.7	**100.0**	99.1	77.0	74.3	88.7	80.4	92.3	96.5	91.1	78.2	96.4	89.2
PPOT+CSG-Norm	**95.8**	**97.3**	**100.0**	**99.5**	**88.4**	**87.2**	**94.7**	**83.9**	**96.8**	**99.1**	**97.7**	**83.6**	**98.8**	**93.3**

Table 2: Classification accuracy (%) on *Office-Home* for unsupervised domain adaptation

Method for UDA	Ar→Cl	Ar→Pr	Ar→Rw	Cl→Ar	Cl→Pr	Cl→Rw	Pr→Ar	Pr→Cl	Pr→Rw	Rw→Ar	Rw→Cl	Rw→Pr	Avg
Source-only	46.3	67.5	75.9	59.1	59.9	62.7	58.2	41.8	74.9	67.4	48.2	74.2	61.4
DANN	45.6	59.3	70.1	47.0	58.5	60.9	46.1	43.7	68.5	63.2	51.8	76.8	57.6
CDAN	50.7	70.6	76.0	57.6	70.0	70.0	57.4	50.9	77.3	70.9	56.7	81.6	65.8
DeepJDOT	48.2	69.2	74.5	58.5	69.1	71.1	56.3	46.0	76.5	68.0	52.7	80.9	64.3
RWOT	55.2	72.5	78.0	63.5	72.5	75.1	60.2	48.5	78.9	69.8	54.8	82.5	67.6
PPOT	57.5	75.8	83.8	68.2	76.9	76.0	66.3	55.4	82.5	75.6	61.1	84.4	72.0
DeepJDOT+CSG-Ent	50.8	70.4	76.2	59.3	69.7	72.4	58.0	47.5	77.6	69.9	53.4	81.7	65.6
PPOT+CSG-Ent	59.6	78.2	86.1	70.7	78.3	77.9	66.9	56.0	83.4	76.8	62.5	85.1	73.5
DeepJDOT+CSG-Norm	52.1	71.8	77.5	61.2	71.0	73.6	59.4	49.2	78.8	71.3	55.6	82.4	67.0
PPOT+CSG-Norm	**61.4**	**79.5**	**87.2**	**72.3**	**79.8**	**79.1**	**68.7**	**57.8**	**84.9**	**78.2**	**64.3**	**86.5**	**75.0**

underlying classical optimal transport problem, effectively discouraging spurious or overly incorrect matchings. Meanwhile, adding the CRM term will not affect the optimization stability or convergence. The whole optimization procedure for the proposed CSG solver is shown in Alg.2.

Algorithm 2 CSG with CRM mechanism for optimal transport

Require: Cost matrix $\boldsymbol{C}$; initial marginal probability $\boldsymbol{\alpha}, \boldsymbol{\beta}$; hyper-parameters ϵ, η.

1: Initialize the value of $\widehat{\boldsymbol{f}}, \boldsymbol{\psi}, \boldsymbol{\phi}$.
2: Optimize $\widehat{\boldsymbol{f}}$ via the Centering-BFGS method in Alg.1.
3: Obtain $\widehat{\boldsymbol{g}}$ via $\widehat{g}_j = -\epsilon \log[\sum_{k=1}^{M} \exp((\widehat{f}_k - C_{kj})/\epsilon)]$.
4: Obtain $\widehat{\boldsymbol{s}}$ via $\widehat{s}_{ij} = \max(0, C_{ij} - \widehat{f}_i - \widehat{g}_j)$.
5: Involve CRM mechanism into the optimal transport as Eq. equation 11
6: Optimize $\boldsymbol{\psi}$ and $\boldsymbol{\phi}$ via Proposition 4.
7: Obtain the matching results of $\boldsymbol{\pi}$ via solving $\nabla_{\pi_{ij}} L_{\mathrm{Reg}}(\pi_{ij}) = (\psi_i + \phi_j - \eta \widehat{s}_{ij} C_{ij})/\epsilon$.

Ensure: The optimal solution $\boldsymbol{\pi}$.

4 Experiment

4.1 Experimental setup

Synthetic Datasets. We first conduct the experiments on the synthetic datasets. That is, we set the source and target domain distributions as $\mathbb{P}_X = \mathcal{N}\left(\begin{bmatrix}-1\\-1\end{bmatrix}, \begin{bmatrix}1 & 0\\0 & 1\end{bmatrix}\right)$ and $\mathbb{P}_Z = \mathcal{N}\left(\begin{bmatrix}4\\4\end{bmatrix}, \begin{bmatrix}1 & -0.8\\-0.8 & 1\end{bmatrix}\right)$ following previous works Flamary et al. (2021); Chapel et al. (2021) for optimal transport.

Real-World Datasets. We also involve several real-world datasets to validate our proposed method. We conduct the unsupervised domain adaptation tasks on *Office-Home* Venkateswara et al. (2017), *Office-31* Saenko et al. (2010) and *ImageCLEF* Caputo et al. (2014). Meanwhile we conduct cross-domain retrieval task on *Office-Home* Venkateswara et al. (2017) and *DomainNet* Peng et al. (2019). The **Office-Home** Venkateswara et al. (2017) dataset consists of images from 4 different domains: Artistic images (**Ar**), Clip Art (**Cl**), Product images (**Pr**), and Real-World (**Rw**) images. For each domain, the dataset contains images of 65 object categories found typically in Office and Home settings. The **Office-31** Saenko et al. (2010) dataset is a classification dataset contains 4,110 images of 31 commonplace office objects in three different domains: Amazon (**A**) images, DSLR (**D**) images, and Webcam (**W**) images, which offers 6 possible transfer tasks for which a model is trained with labeled source domain data and adapted to the unlabeled target domain data. The **ImageCLEF** Caputo et al. (2014) dataset contains 3 domains with 12 classes, i.e., *Caltech* (**C**), *ImageNet* (**I**) and *Pascal* (**P**), whose target domain has the first 6 classes (alphabetical order) following Luo et al. (2020). The **DomainNet** dataset Peng et al. (2019) consists of six domains: *Clipart*, *Infograph*, *Painting*, *Quickdraw*, *Real*, and *Sketch* with significant variations in visual style.

Table 3: Experimental results on Cross-Domain Retrieval on Office-Home.

	Art → Real		Real → Art		Art → Product		Product → Art		Clipart → Real		Real → Clipart	
	P@1	P@5	P@1	P@5	P@1	P@5	P@1	P@5	P@1	P@5	P@1	P@5
ID	0.3589	0.3313	0.3989	0.3442	0.2588	0.2491	0.3217	0.2594	0.2948	0.2648	0.3551	0.3217
ProtoNCE	0.4050	0.3639	0.4453	0.3926	0.2954	0.2789	0.3573	0.3061	0.2525	0.2266	0.4115	0.3766
CDS	0.4508	0.4115	0.4471	0.4075	0.3276	0.3147	0.3575	0.3248	0.3251	0.3030	0.3888	0.3648
PCS	0.4170	0.3851	0.4496	0.3988	0.3329	0.3150	0.3924	0.3477	0.2907	0.2606	0.4060	0.3811
DD	0.4512	0.4233	0.4795	0.4368	0.3539	0.3467	0.4251	0.3794	0.3331	0.3057	0.4466	0.4147
ProtoOT	0.4994	0.4924	0.5490	0.5181	0.4425	0.4595	0.5343	0.5207	0.4092	0.4068	0.5111	0.5212
ProtoOT + CSG-Ent	0.5123	0.5031	0.5578	0.5264	0.4512	0.4678	0.5429	0.5295	0.4175	0.4132	0.5198	0.5287
ProtoOT + CSG-Norm	**0.5287**	**0.5174**	**0.5692**	**0.5381**	**0.4635**	**0.4789**	**0.5556**	**0.5413**	**0.4298**	**0.4256**	**0.5334**	**0.5426**
	Product → Real		Real → Product		Product → Clipart		Clipart → Product		Art → Clipart		Clipart → Art	
	P@1	P@5	P@1	P@5	P@1	P@5	P@1	P@5	P@1	P@5	P@1	P@5
ID	0.5073	0.4503	0.4512	0.4146	0.3152	0.2855	0.2401	0.2242	0.2678	0.2479	0.2117	0.1786
ProtoNCE	0.5384	0.4825	0.4774	0.4485	0.3613	0.3399	0.2117	0.2063	0.2897	0.2615	0.2133	0.1740
CDS	0.5400	0.5007	0.4939	0.4727	0.3769	0.3499	0.2724	0.2646	0.2559	0.2377	0.2241	0.2034
PCS	0.5645	0.5078	0.4990	0.4711	0.3951	0.3751	0.2639	0.2586	0.3123	0.2874	0.2451	0.2127
DD	0.5742	0.5269	0.5171	0.4848	0.4226	0.3742	0.2779	0.2726	0.3267	0.3079	0.2726	0.2394
ProtoOT	0.6401	0.5922	0.5485	0.5349	0.4476	0.4264	0.2992	0.3015	0.3539	0.3457	0.2896	0.2558
ProtoOT + CSG-Ent	0.6514	0.6037	0.5572	0.5438	0.4568	0.4359	0.3087	0.3104	0.3621	0.3542	0.2978	0.2635
ProtoOT + CSG-Norm	**0.6678**	**0.6189**	**0.5695**	**0.5552**	**0.4693**	**0.4472**	**0.3214**	**0.3231**	**0.3745**	**0.3668**	**0.3102**	**0.2764**

Table 4: Experimental results on Cross-Domain Retrieval on DomainNet.

	Clipart → Sketch		Sketch → Clipart		Infograph → Real		Real → Infograph		Infograph → Sketch		Sketch → Infograph	
	P@50	P@100	P@50	P@100	P@50	P@100	P@50	P@100	P@50	P@100	P@50	P@100
ID	0.4946	0.4609	0.5438	0.4712	0.2827	0.2744	0.3998	0.3177	0.3035	0.2904	0.4220	0.3494
ProtoNCE	0.4685	0.4267	0.5452	0.4504	0.2841	0.2853	0.5701	0.4184	0.2824	0.2679	0.3983	0.3199
CDS	0.4584	0.4237	0.5913	0.4883	0.2851	0.2792	0.5669	0.3976	0.3055	0.2951	0.4627	0.3611
PCS	0.5101	0.4687	0.5970	0.5067	0.3056	0.3027	0.5542	0.4213	0.3027	0.2836	0.4258	0.3409
DD	0.5631	0.5274	0.6307	0.5726	0.3552	0.3524	0.5774	0.4669	0.3129	0.2933	0.4366	0.3614
ProtoOT	0.7046	0.6941	0.8279	0.7868	0.4065	0.4035	0.7702	0.6733	0.3716	0.3630	0.6359	0.5330
ProtoOT + CSG-Ent	0.7125	0.7018	0.8356	0.7942	0.4153	0.4127	0.7789	0.6821	0.3794	0.3708	0.6437	0.5412
ProtoOT + CSG-Norm	**0.7258**	**0.7143**	**0.8473**	**0.8065**	**0.4276**	**0.4249**	**0.7912**	**0.6945**	**0.3918**	**0.3832**	**0.6564**	**0.5539**
	Painting → Clipart		Clipart → Painting		Painting → Quickdraw		Quickdraw → Painting		Quickdraw → Real		Real → Quickdraw	
	P@50	P@100	P@50	P@100	P@50	P@100	P@50	P@100	P@50	P@100	P@50	P@100
ID	0.6467	0.5441	0.4237	0.3961	0.2034	0.1959	0.2112	0.1981	0.2827	0.2746	0.2345	0.2279
ProtoNCE	0.5544	0.4374	0.3913	0.3587	0.2163	0.2124	0.2395	0.2284	0.2638	0.2570	0.2510	0.2481
CDS	0.6315	0.4730	0.3775	0.3518	0.1875	0.1889	0.2137	0.2144	0.1928	0.1914	0.1536	0.1557
PCS	0.6347	0.5321	0.4883	0.4621	0.2512	0.2465	0.2403	0.2324	0.3482	0.3392	0.2898	0.2885
DD	0.6642	0.5684	0.5258	0.5010	0.3972	0.3859	0.3345	0.3381	0.4279	0.4275	0.4190	0.4210
ProtoOT	0.9021	0.8738	0.7113	0.7115	0.6396	0.6275	0.6001	0.6033	0.6032	0.6064	0.5517	0.5556
ProtoOT + CSG-Ent	0.9104	0.8826	0.7205	0.7201	0.6487	0.6362	0.6098	0.6124	0.6125	0.6158	0.5603	0.5642
ProtoOT + CSG-Norm	**0.9237**	**0.8959**	**0.7328**	**0.7324**	**0.6612**	**0.6489**	**0.6224**	**0.6251**	**0.6251**	**0.6284**	**0.5736**	**0.5775**

Baselines. We compare the CSG method with the following OT solvers on the synthetic datasets. (1) **Sinkhorn** Cuturi (2013) is a widely-used solver that employs entropic regularization to address the classical OT problem. (2) **L2-OT** Blondel et al. (2018) adopts a square-norm regularization term to achieve sparser matching results. (3) q**-DOT** Bao & Sakaue (2022) further involves deformed q-entropy with a q-exponential distribution for tackling OT. (4) **SSNS** Tang & Qiu (2024) is the latest optimal transport solver that proposed a safe and sparse Newton algorithm during the calculation procedure. Meanwhile, we also involve **DANN** Ganin et al. (2016), **CDAN** Long et al. (2018), **DeepJDOT** Damodaran et al. (2018), **RWOT** Xu et al. (2020) and **PPOT** Wang et al. (2024) as the model baselines for the unsupervised domain adaptation task. Moreover, we include **ID** Wu et al. (2018), **ProtoNCE** Li et al., **CDS** Kim et al. (2021), **PCS** Yue et al. (2021), **DD** Hu & Lee (2022) and **ProtoOT** Li et al. (2024) as the model baselines for the unsupervised cross-domain retrieval task. More detailed illustrations of these baselines are provided in the Appendix E, F.

Implemented details. For both synthetic and real-world datasets, we set $\epsilon = 0.01$ and $\eta = 100$ on CSG solver with the proposed CRM term. Specifically, we mainly investigate two variants of CSG: CSG-Ent with the entropic regularization term $L_{\text{reg}}(\boldsymbol{\pi}) = \langle \boldsymbol{\pi}, \log(\boldsymbol{\pi}) - 1 \rangle$, and CSG-Norm with the ℓ_2-norm regularization term $L_{\text{reg}}(\boldsymbol{\pi}) = \langle \boldsymbol{\pi}, \boldsymbol{\pi} \rangle$. The initial values of $\widehat{\boldsymbol{f}}^{(0)}$ and $\boldsymbol{\psi}^{(0)}$ are set as zero vectors. The initial sample weights are set to be equal, i.e., $a_i = \frac{1}{M}$ and $b_j = \frac{1}{N}$. For the real-world experiments, we adopt the same framework and settings as previous methods accordingly with the only modification being the replacement of their OT solvers with proposed CSG. Meanwhile we adopt classification accuracy and precision (i.e., P@k for top k) as the evaluation metrics for unsupervised domain adaptation and cross-domain retrieval tasks respectively. For all experiments, we conduct 20 independent runs with different random seeds and report the average results.

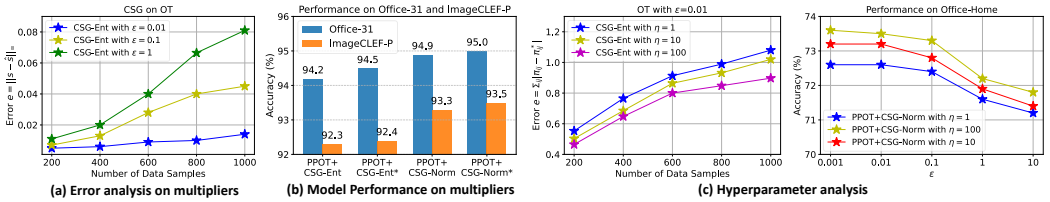


Figure 4: The error analysis, model performance on multipliers and hyperparameter tuning of CSG.

4.2 Performance on Synthetic and Real-World Datasets

Performance on Synthetic Datasets. We first sample N two-dimensional points from both the source and target distributions with N ranging from 100 to 1000 for computing the optimal transport plan $\boldsymbol{\pi}^*$. We then compute the absolute computation error between the transport plan $\boldsymbol{\pi}$ obtained from the OT solvers and the ground-truth plan $\boldsymbol{\pi}^*$ solved via CVXPY Diamond & Boyd (2016) defined as $e = \sum_{i,j} |\boldsymbol{\pi}_{ij} - \boldsymbol{\pi}^*_{ij}|$, and report the results in Fig.2(a). Additionally, we provide a runtime analysis as shown in Fig.2(b). We can further observe that Sinkhorn solver could lead to more inaccurate results against other methods. Meanwhile L2-OT, q-DOT and SSNS can lead to more precise output results through the corresponding sparsification mechanism. However, their computing time consumption is considerably high (e.g., for matrix inversion) especially when N increases. Our proposed CSG-based approach can provide more accurate matching solutions with lower absolute error e. Note that CSG-Norm achieves a lower absolute error compared to CSG-Ent, while CSG-Ent maintains comparable computational efficiency to the Sinkhorn algorithm, underscoring the practical efficacy of the proposed CSG method. Moreover, we further investigate the method generalization via replacing the original Sinkhorn solver in CSG-Ent with Greenkhorn Altschuler et al. (2017) and Screenkhorn Alaya et al. (2019), resulting in the variants CSG-Ent/Greenkhorn and CSG-Ent/Screenkhorn, respectively. We also conduct the experiments on synthetic datasets and report the time consumption in Fig.2(c)-(d). Notably, CSG-Ent/Greenkhorn and CSG-Ent/Screenkhorn achieve absolute errors comparable to standalone CSG-Ent, while offering substantial acceleration in computation. These results demonstrate the strong generalization capability of our CSG approach, which can be readily combined with other acceleration techniques.

Performance on Real-World Datasets. We further conduct the experiments on the real-world datasets to validate the proposed CSG-based method. The experimental UDA task results on *Office-31*, *ImageCLEF* and *Office-Home* are shown in Table 1 and Table 2. We can observe that replacing conventional OT solvers with more accurate methods (e.g., CSG-Ent) will definitely boost model performance. This phenomenon underscores the importance of choosing more accurate optimal transport solvers. Meanwhile our proposed PPOT+CSG-Norm approach can achieve better performance than the original baseline model PPOT and PPOT+CSG-Ent which is consistent with absolute error analysis on the synthetic datasets shown in Fig.2. Moreover, we report the experimental results of the cross-domain retrieval task on Office-Home and DomainNet in Table 3 and Table 4. Likewise, ProtoOT+CSG-Ent and ProtoOT+CSG-Norm also boost the model potentials against ProtoOT with more accurate matching results. Overall, the proposed CSG approach is model-agnostic and can be easily incorporated into different baseline models to boost their performance. It further exhibits superior capability in addressing OT problems across a wide range of real-world applications.

4.3 Analysis

Visualization on Matching Results. To better illustrate our proposed method, we further provide the visualization analysis on matching results. Specifically, we first sample 50 data samples on both source and target distributions to find $\boldsymbol{\pi}^*$ as shown in Fig.3(a). The coupling matrices $\boldsymbol{\pi}^*$ obtained by the Sinkhorn, SSNS, and CSG-Ent solvers are then presented in Fig.3(b), (c), and (d), respectively. We can observe that Sinkhorn algorithm with the entropic regularization term could lead to dense and ambiguous matching results for $\boldsymbol{\pi}^*$. Meanwhile SSNS can obtain a sparser solution with its sparsification on the Hessian matrix. However, SSNS requires computing the inverse Hessian matrix based on a predefined threshold which poses significant implementation challenges in practice. Compared to other approaches, our proposed CSG-Ent solver produces clearer and more accurate results against other baseline methods which demonstrating its superior efficacy.

Table 5: Recommendation performances of NDCG@k and Recall@k on Yahoo! R3, Coat and KuaiRec for method extension on partial optimal transport.

Dataset	Yahoo! R3			Coat			KuaiRec		
Metrics	NDCG@1	NDCG@3	NDCG@5	NDCG@1	NDCG@3	NDCG@5	NDCG@1	NDCG@3	NDCG@5
RMF Saito et al. (2020)	0.632	0.666	0.706	0.374	0.403	0.442	0.211	0.209	0.212
SCF Li et al. (2021)	0.661	0.690	0.722	0.447	0.456	0.493	0.455	0.442	0.460
UPL Ren et al. (2023)	0.759	0.796	0.829	0.512	0.508	0.539	0.250	0.265	0.281
DCF He et al. (2024)	0.762	0.796	0.826	0.544	0.538	0.566	0.448	0.457	0.466
WeaklyRec Wang et al. (2025)	0.784	0.814	0.843	0.552	0.555	0.589	0.498	0.486	0.488
WeaklyRec+CSG-Ent	0.797	0.825	0.852	0.566	0.570	0.594	0.501	0.505	0.509
WeaklyRec+CSG-Norm	**0.803**	**0.831**	**0.859**	**0.572**	**0.578**	**0.601**	**0.506**	**0.510**	**0.517**
Dataset	Yahoo! R3			Coat			KuaiRec		
Metrics	Recall@1	Recall@3	Recall@5	Recall@1	Recall@3	Recall@5	Recall@1	Recall@3	Recall@5
RMF Saito et al. (2020)	0.042	0.132	0.226	0.054	0.170	0.285	0.008	0.022	0.037
SCF Li et al. (2021)	0.060	0.152	0.261	0.073	0.213	0.324	0.115	0.166	0.201
UPL Ren et al. (2023)	0.129	0.268	0.346	0.118	0.256	0.369	0.023	0.070	0.117
DCF He et al. (2024)	0.131	0.262	0.339	0.134	0.272	0.386	0.119	0.231	0.288
WeaklyRec Wang et al. (2025)	0.146	0.280	0.354	0.131	0.302	0.425	0.137	0.245	0.296
WeaklyRec+CSG-Ent	0.155	0.289	0.368	0.145	0.306	0.437	0.142	0.254	0.303
WeaklyRec+CSG-Norm	**0.162**	**0.296**	**0.371**	**0.150**	**0.312**	**0.445**	**0.148**	**0.259**	**0.307**

Approximation Error Analysis. To further validate the quality of the smoothness approximation in the exploration stage of the CSG solver, we analyze the error $e_s = \|\widehat{\boldsymbol{s}} - \boldsymbol{s}\|_\infty$ between the multiplier $\widehat{\boldsymbol{s}}$ learned by CSG and the ground-truth $\boldsymbol{s}$ computed using CVXPY. That is, we conduct the experiments on two-dimensional synthetic datasets by varying different number of data samples N and ϵ in range of $\epsilon \in \{0.01, 0.1, 1\}$ and report the results in Fig.4(a). We observe that the computation error e_s remains relatively small particularly when $\epsilon = 0.01$ which aligns well with the theoretical approximation presented in Section 3.1.2. It is worth noting that larger values of ϵ will inevitably lead to inaccurate approximations, thereby deteriorating the final results. Moreover, we conduct the experiments on real-world Office-31 and ImageCLEF datasets involving ground-truth $\boldsymbol{s}$ with PPOT+CSG-Ent* and PPOT+CSG-Norm* then report the results in Fig.4(b). From the analysis, we conclude that decreasing ϵ improves the accuracy of the multiplier estimation without significantly sacrificing model performance compared to using the ground-truth $\boldsymbol{s}$ in the real-world applications.

Method Extension. To evaluate the generalizability of the proposed algorithm, we apply the CSG method to tackle the partial optimal transport Chapel et al. (2020). We conduct the experiments on Yahoo!R3 Marlin & Zemel (2009), Coat Schnabel et al. (2016) and KuaiRec Gao et al. (2022) for the positive-unlabeled recommendation problem. We follow the same settings as the state-of-the-art model WeaklyRec Wang et al. (2025) via simply replacing Sinkhorn with CSG method and report the results of Recall@k and NDCG@k Wang et al. (2019) in Table 5. We observe that both WeaklyRec+CSG-Ent and WeaklyRec+CSG-Norm further boost performance compared to the other baseline approaches Saito et al. (2020); He et al. (2024); Ren et al. (2023); Li et al. (2021) demonstrates the effectiveness of the proposed CSG method.

Parameter Sensitivity. We finally study the effects of hyperparameters on model performance. That is, we tune η on CSG in a range of $\eta \in \{1, 10, 100\}$ using the synthetic and real-world datasets then report the experimental results in Fig.4(c). We can observe that when η is smaller (e.g., $\eta = 1$), the proposed CRM term will fail to play a significant role during the optimization procedure with limited gains. Meanwhile, setting $\eta = 100$ enables the CSG solver to deliver more accurate matching outputs, resulting in improved overall performance. Consequently, we empirically set $\eta = 100$.

5 Conclusion

In this paper, we propose the Complementary Slackness Guidance (CSG) approach with the Conditional Regularization Mask (CRM) to solve optimal transport problem. In particular, we analyze the key properties of the CSG solver that help reduce the numerical duality gap, mitigate the adverse effects of L_{Reg}, and yield more accurate results. Specifically, the CSG solver first performs an exploration stage to estimate the Lagrange multipliers using a smoothed approximation with Centering-BFGS. Then CSG conducts the utilization stage to leverage this valuable information throughout the entire optimization process. We provide an error bound analysis (e.g., estimated $\widehat{\boldsymbol{s}}$ and groundtruth $\boldsymbol{s}$) to verify the efficacy of the CSG method. We conduct experiments to demonstrate the superior performance of CSG solver on both synthetic and real-world datasets of different tasks and applications, indicating that CSG method is model-agnostic and straightforward to implement.

AI USE STATEMENT

In this work, we have not used generative AI tools for generating synthetic data sets, helping develop theoretical models or conceptual frameworks, formulating mathematical claims, providing critical ingredients for proving mathematical claims, assisting in the writing of proofs, proposing or refining hypotheses, designing or providing feedback on research methodology or experiments, implementing methods, assisting with translation, cleaning and reformatting dataset, supporting qualitative and thematic data analysis, and interpreting results. We have reviewed all AI-assisted work. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Mokhtar Z Alaya, Maxime Berar, Gilles Gasso, and Alain Rakotomamonjy. Screening sinkhorn algorithm for regularized optimal transport. *Advances in Neural Information Processing Systems*, 32, 2019.

Jason Altschuler, Jonathan Niles-Weed, and Philippe Rigollet. Near-linear time approximation algorithms for optimal transport via sinkhorn iteration. *Advances in neural information processing systems*, 30, 2017.

Dongsheng An, Na Lei, Xiaoyin Xu, and Xianfeng Gu. Efficient optimal transport algorithm by accelerated gradient descent. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 10119–10128, 2022.

Han Bao and Shinsaku Sakaue. Sparse regularized optimal transport with deformed q-entropy. *Entropy*, 24(11):1634, 2022.

Mathieu Blondel, Vivien Seguy, and Antoine Rolet. Smooth and sparse optimal transport. In *International conference on artificial intelligence and statistics*, pp. 880–889. PMLR, 2018.

Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

He Cao, Yunzhou Zhang, Shangdong Zhu, and Lei Wang. Pro-clip: A clip-based category measurement network through prototype and regularized optimal transportation. *IEEE Transactions on Instrumentation and Measurement*, 2024.

Barbara Caputo, Henning Müller, Jesus Martinez-Gomez, Mauricio Villegas, Burak Acar, Novi Patricia, Neda Marvasti, Suzan Üsküdarlı, Roberto Paredes, Miguel Cazorla, et al. Imageclef 2014: Overview and analysis of the results. In *Information Access Evaluation. Multilinguality, Multimodality, and Interaction: 5th International Conference of the CLEF Initiative, CLEF 2014, Sheffield, UK, September 15-18, 2014. Proceedings 5*, pp. 192–211. Springer, 2014.

Laetitia Chapel, Mokhtar Z Alaya, and Gilles Gasso. Partial optimal tranport with applications on positive-unlabeled learning. *Advances in Neural Information Processing Systems*, 33:2903–2913, 2020.

Laetitia Chapel, Rémi Flamary, Haoran Wu, Cédric Févotte, and Gilles Gasso. Unbalanced optimal transport through non-negative penalized linear regression. *Advances in Neural Information Processing Systems*, 34:23270–23282, 2021.

Guangyi Chen, Weiran Yao, Xiangchen Song, Xinyue Li, Yongming Rao, and Kun Zhang. Plot: Prompt learning with optimal transport for vision-language models. In *ICLR*, 2023.

Sinho Chewi, Jonathan Niles-Weed, and Philippe Rigollet. *Statistical optimal transport*. Springer, 2025.

Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26, 2013.

Bharath Bhushan Damodaran, Benjamin Kellenberger, Rémi Flamary, Devis Tuia, and Nicolas Courty. Deepjdot: Deep joint distribution optimal transport for unsupervised domain adaptation. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 447–463, 2018.

Steven Diamond and Stephen Boyd. Cvxpy: A python-embedded modeling language for convex optimization. *Journal of Machine Learning Research*, 17(83):1–5, 2016.

Rémi Flamary, Nicolas Courty, Alexandre Gramfort, Mokhtar Z Alaya, Aurélie Boisbunon, Stanislas Chambon, Laetitia Chapel, Adrien Corenflos, Kilian Fatras, Nemo Fournier, et al. Pot: Python optimal transport. *The Journal of Machine Learning Research*, 22(1):3571–3578, 2021.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario March, and Victor Lempitsky. Domain-adversarial training of neural networks. *Journal of machine learning research*, 17(59):1–35, 2016.

Chongming Gao, Shijun Li, Wenqiang Lei, Jiawei Chen, Biao Li, Peng Jiang, Xiangnan He, Jiaxin Mao, and Tat-Seng Chua. Kuairec: A fully-observed dataset and insights for evaluating recommender systems. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, pp. 540–550, 2022.

Johannes Gasteiger, Marten Lienen, and Stephan Günnemann. Scalable optimal transport in high dimensions for graph distances, embedding alignment, and more. In *International Conference on Machine Learning*, pp. 5616–5627. PMLR, 2021.

Andrew Gracyk and Xiaohui Chen. Geonet: a neural operator for learning the wasserstein geodesic. *arXiv preprint arXiv:2209.14440*, 2022.

Xiang Gu, Liwei Yang, Jian Sun, and Zongben Xu. Optimal transport-guided conditional score-based diffusion model. *Advances in Neural Information Processing Systems*, 36:36540–36552, 2023.

Zhuangzhuang He, Yifan Wang, Yonghui Yang, Peijie Sun, Le Wu, Haoyue Bai, Jinqi Gong, Richang Hong, and Min Zhang. Double correction framework for denoising recommendation. In *Proceedings of the 30th ACM SIGKDD conference on knowledge discovery and data mining*, pp. 1062–1072, 2024.

Conghui Hu and Gim Hee Lee. Feature representation learning for unsupervised cross-domain image retrieval. In *European Conference on Computer Vision*, pp. 529–544. Springer, 2022.

Qiujiang Jin, Ruichen Jiang, and Aryan Mokhtari. Non-asymptotic global convergence analysis of bfgs with the armijo-wolfe line search. *Advances in Neural Information Processing Systems*, 37: 16810–16851, 2024.

Parnian Kassraie, Aram-Alexandre Pooladian, Michal Klein, James Thornton, Jonathan Niles-Weed, and Marco Cuturi. Progressive entropic optimal transport solvers. *Advances in Neural Information Processing Systems*, 37:19561–19590, 2024.

Tanguy Kerdoncuff, Remi Emonet, and Marc Sebban. Sampled gromov wasserstein. *Machine Learning*, 110(8):2151–2186, 2021.

Donghyun Kim, Kuniaki Saito, Tae-Hyun Oh, Bryan A Plummer, Stan Sclaroff, and Kate Saenko. Cds: Cross-domain self-supervised pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9123–9132, 2021.

Nikita Kornilov, Petr Mokrov, Alexander Gasnikov, and Aleksandr Korotin. Optimal flow matching: Learning straight trajectories in just one step. *Advances in Neural Information Processing Systems*, 37:104180–104204, 2024.

Bin Li, Ye Shi, Qian Yu, and Jingya Wang. Unsupervised cross-domain image retrieval via prototypical optimal transport. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 3009–3017, 2024.

Junnan Li, Pan Zhou, Caiming Xiong, and Steven Hoi. Prototypical contrastive learning of unsupervised representations. In *International Conference on Learning Representations*.

Mengyu Li, Jun Yu, Tao Li, and Cheng Meng. Importance sparsification for sinkhorn algorithm. *arXiv preprint arXiv:2306.06581*, 2023.

Xiucheng Li, Jin Yao Chin, Yile Chen, and Gao Cong. Sinkhorn collaborative filtering. In *Proceedings of the web conference 2021*, pp. 582–592, 2021.

Dong C Liu and Jorge Nocedal. On the limited memory bfgs method for large scale optimization. *Mathematical programming*, 45(1-3):503–528, 1989.

Tianlin Liu, Joan Puigcerver, and Mathieu Blondel. Sparsity-constrained optimal transport. *ICLR*, 2023a.

Yanbei Liu, Chongxu Wang, Zhitao Xiao, Lei Geng, Yanwei Pang, and Xiao Wang. Hgot: Self-supervised heterogeneous graph neural network with optimal transport. *arXiv preprint arXiv:2506.02619*, 2025.

Yang Liu, Zhipeng Zhou, and Baigui Sun. Cot: Unsupervised domain adaptation with clustering and optimal transport. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 19998–20007, 2023b.

Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. Conditional adversarial domain adaptation. *Advances in neural information processing systems*, 31, 2018.

Dirk A Lorenz, Paul Manns, and Christian Meyer. Quadratically regularized optimal transport. *Applied Mathematics & Optimization*, 83(3):1919–1949, 2021.

You-Wei Luo, Chuan-Xian Ren, Dao-Qing Dai, and Hong Yan. Unsupervised domain adaptation via discriminative manifold propagation. *IEEE transactions on pattern analysis and machine intelligence*, 44(3):1653–1669, 2020.

Benjamin M Marlin and Richard S Zemel. Collaborative prediction and ranking with non-random missing data. In *Proceedings of the third ACM conference on Recommender systems*, pp. 5–12, 2009.

Eduardo Fernandes Montesuma, Fred Maurice Ngole Mboula, and Antoine Souloumiac. Recent advances in optimal transport for machine learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2024.

Yu Nesterov. Smooth minimization of non-smooth functions. *Mathematical programming*, 103: 127–152, 2005.

Xiaxue Ouyang, Hao Zheng, Haoxian Liang, Jingyi Zhang, Yixuan Qiu, Cheng Meng, and Mengyu Li. Sparsification techniques for large-scale optimal transport problems. *Wiley Interdisciplinary Reviews: Computational Statistics*, 18(1):e70056, 2026.

Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 1406–1415, 2019.

Yi Ren, Hongyan Tang, Jiangpeng Rong, and Siwen Zhu. Unbiased pairwise learning from implicit feedback for recommender systems without biased variance control. In *Proceedings of the 46th international ACM SIGIR conference on research and development in information retrieval*, pp. 2461–2465, 2023.

Kate Saenko, Brian Kulis, Mario Fritz, and Trevor Darrell. Adapting visual category models to new domains. In *Computer Vision–ECCV 2010: 11th European Conference on Computer Vision, Heraklion, Crete, Greece, September 5-11, 2010, Proceedings, Part IV 11*, pp. 213–226. Springer, 2010.

Yuta Saito, Suguru Yaginuma, Yuta Nishino, Hayato Sakata, and Kazuhide Nakata. Unbiased recommender learning from missing-not-at-random implicit feedback. In *Proceedings of the 13th international conference on web search and data mining*, pp. 501–509, 2020.

Maziar Sanjabi, Jimmy Ba, Meisam Razaviyayn, and Jason D Lee. On the convergence and robustness of training gans with regularized optimal transport. *Advances in Neural Information Processing Systems*, 31, 2018.

Tobias Schnabel, Adith Swaminathan, Ashudeep Singh, Navin Chandak, and Thorsten Joachims. Recommendations as treatments: Debiasing learning and evaluation. In *international conference on machine learning*, pp. 1670–1679. PMLR, 2016.

Liangliang Shi, Jack Fan, and Junchi Yan. Ot-clip: Understanding and generalizing clip via optimal transport. In *Forty-first International Conference on Machine Learning*, 2024.

Jianheng Tang, Weiqi Zhang, Jiajin Li, Kangfei Zhao, Fugee Tsung, and Jia Li. Robust attributed graph alignment via joint structure learning and optimal transport. In *2023 IEEE 39th International Conference on Data Engineering (ICDE)*, pp. 1638–1651. IEEE, 2023.

Xun Tang, Michael Shavlovsky, Holakou Rahmanian, Elisa Tardini, Kiran Koshy Thekumparampil, Tesi Xiao, and Lexing Ying. Accelerating sinkhorn algorithm with sparse newton iterations. *arXiv preprint arXiv:2401.12253*, 2024.

Zihao Tang and Yixuan Qiu. Safe and sparse newton method for entropic-regularized optimal transport. *Advances in Neural Information Processing Systems*, 37:129914–129943, 2024.

Luis Caicedo Torres, Luiz Manella Pereira, and M Hadi Amini. A survey on optimal transport for machine learning: Theory and applications. *arXiv preprint arXiv:2106.01963*, 2021.

Hemanth Venkateswara, Jose Eusebio, Shayok Chakraborty, and Sethuraman Panchanathan. Deep hashing network for unsupervised domain adaptation. *CoRR*, abs/1706.07522, 2017.

Cédric Villani et al. *Optimal transport: old and new*, volume 338. Springer, 2009.

Hao Wang, Zhichao Chen, Haotian Wang, Yanchao Tan, Licheng Pan, Tianqiao Liu, Xu Chen, Haoxuan Li, and Zhouchen Lin. Unbiased recommender learning from implicit feedback via weakly supervised learning. In *Forty-second International Conference on Machine Learning*, 2025.

Wei Wang, Fei Wen, Zeyu Yan, and Peilin Liu. Optimal transport for unsupervised denoising learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(2):2104–2118, 2022.

Xiang Wang, Xiangnan He, Yixin Cao, Meng Liu, and Tat-Seng Chua. Kgat: Knowledge graph attention network for recommendation. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pp. 950–958, 2019.

Yan Wang, Chuan-Xian Ren, Yi-Ming Zhai, You-Wei Luo, and Hong Yan. Probability-polarized optimal transport for unsupervised domain adaptation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pp. 15653–15661, 2024.

Di Wu, Ling Liang, and Haizhao Yang. Pins: Proximal iterations with sparse newton and sinkhorn for optimal transport. *arXiv preprint arXiv:2502.03749*, 2025.

Zhirong Wu, Yuanjun Xiong, Stella X Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3733–3742, 2018.

Yujia Xie, Xiangfeng Wang, Ruijia Wang, and Hongyuan Zha. A fast proximal point method for computing exact wasserstein distance. In *Uncertainty in artificial intelligence*, pp. 433–453. PMLR, 2020.

Renjun Xu, Pelen Liu, Yin Zhang, Fang Cai, Jindong Wang, Shuoying Liang, Heting Ying, and Jianwei Yin. Joint partial optimal transport for open set domain adaptation. In *IJCAI*, pp. 2540–2546, 2020.

Xiangyu Yue, Zangwei Zheng, Shanghang Zhang, Yang Gao, Trevor Darrell, Kurt Keutzer, and Alberto Sangiovanni Vincentelli. Prototypical cross-domain self-supervised learning for few-shot unsupervised domain adaptation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 13834–13844, 2021.

Appendix

A Notation Table

We provide the important notations and their descriptions for clarification on Table.6.

Symbol	Description
$\boldsymbol{C}$	The cost matrix
η, ϵ	The hyperparameters
$\boldsymbol{\alpha}, \boldsymbol{\beta}$	The marginal probabilities for $\mathcal{J}_{\mathrm{OT}}$
$\boldsymbol{f}, \boldsymbol{s}$	The ground truth of multiplier for $\mathcal{J}_{\mathrm{OT}}$
$\widehat{\boldsymbol{f}}, \widehat{\boldsymbol{s}}$	The learned multiplier for $\mathfrak{J}_{\mathrm{OT}}$
$\boldsymbol{H}$	The approximated Hessian matrix in Centering-BFGS
ζ	The step size in Centering-BFGS
μ	The strongly convex coefficient for $\mathcal{J}_{\mathrm{OT}}$
κ	The strongly convex coefficient for L_{Reg}

Table 6: Important notations

B Proof for Exploration Stage of CSG solver (Proposition 1-3)

Proposition 1. *The objective $\mathfrak{J}_{\mathrm{OT}}$ in Eq. equation 4 has a $1/\epsilon$-Lipschitz continuous gradient. Moreover, it is μ-strongly convex restricted to the subspace $\mathcal{M} := \{\widehat{\boldsymbol{f}}^\top \mathbf{1}_{\mathrm{M}} = 0\}$ where $\mu = \lambda_{\min}\big(\nabla^2 \mathfrak{J}_{\mathrm{OT}}\big|_{\mathcal{M}}\big)$.*

Proof. Given the consideration on the differentiation on $\mathfrak{J}_{\mathrm{OT}}$, we can obtain the following results:

$$\begin{aligned}\frac{\partial \mathfrak{J}_{\mathrm{OT}}}{\partial \widehat{f}_s} &= \sum_{j=1}^{N} p_{sj}\beta_j - \alpha_s \\ &= \sum_{j=1}^{N}\left[\frac{\exp\left(\frac{\widehat{f}_s - C_{sj}}{\epsilon}\right)}{\sum_{k=1}^{M}\exp\left(\frac{\widehat{f}_k - C_{kj}}{\epsilon}\right)}\beta_j\right] - \alpha_s,\end{aligned} \tag{13}$$

where the coupling p_{sj} is given by the Softmax expression.

Furthermore, the second-order partial derivatives (Hessian entries) can be derived as follows:

$$\frac{\partial^2 \mathfrak{J}_{\mathrm{OT}}}{\partial \widehat{f}_s \partial \widehat{f}_i} = \begin{cases} -\dfrac{1}{\epsilon}\displaystyle\sum_{j=1}^{N}\left[p_{ij}p_{sj}\right]\beta_j, & s \neq i \\ \dfrac{1}{\epsilon}\displaystyle\sum_{j=1}^{N}\left[p_{ij} - p_{ij}^2\right]\beta_j, & s = i \end{cases} \tag{14}$$

The Hessian matrix $\nabla^2 \mathfrak{J}_{\mathrm{OT}}$ is positive semidefinite, as it can be expressed as $\nabla^2 \mathfrak{J}_{\mathrm{OT}} = \frac{1}{\epsilon}\sum_{j=1}^{N}\beta_j\left(\mathrm{Diag}(\boldsymbol{p}_{\cdot j}) - \boldsymbol{p}_{\cdot j}\boldsymbol{p}_{\cdot j}^\top\right)$ where each term $\mathrm{Diag}(\boldsymbol{p}_{\cdot j}) - \boldsymbol{p}_{\cdot j}\boldsymbol{p}_{\cdot j}^\top$ is positive semidefinite. To bound the Lipschitz constant of the gradient, we should note that the spectral norm satisfies $\|\nabla^2 \mathfrak{J}_{\mathrm{OT}}\|_2 = \lambda_{\max}(\nabla^2 \mathfrak{J}_{\mathrm{OT}}) \leq \mathrm{tr}(\nabla^2 \mathfrak{J}_{\mathrm{OT}})$ since the matrix is positive semidefinite and all eigenvalues are nonnegative. Computing the trace yields:

$$\begin{aligned}\mathrm{tr}\Big(\nabla^2 \mathfrak{J}_{\mathrm{OT}}\Big) &= \frac{1}{\epsilon}\sum_{j=1}^{N}\left[\sum_{i=1}^{M}\Big(p_{ij} - p_{ij}^2\Big)\right]\beta_j \\ &\leq \frac{1}{\epsilon},\end{aligned} \tag{15}$$

where the inequality follows from $\sum_i p_{ij}^2 \geq 0$ and $\sum_j \beta_j = 1$. Thus, the gradient of $\mathfrak{J}_{\mathrm{OT}}$ is Lipschitz continuous with a Lipschitz constant bounded by $1/\epsilon$.

Moreover, we consider the direction vector $\boldsymbol{e}$ which is constructed to meet the constraint $\sum_{k=1}^{M} e_k = 0$. By considering the minimal value of $\boldsymbol{p}$ as $\gamma = \min(p_{ij}) > 0$ we have:

$$\begin{aligned}
\boldsymbol{e}^\top \nabla^2 \mathfrak{J}_{\mathrm{OT}} \boldsymbol{e} = \frac{1}{\epsilon} \sum_{j=1}^{N} b_j \left[\sum_{k=1}^{M} p_{kj} e_k^2 - \left(\sum_{k=1}^{M} p_{kj} e_k \right)^2 \right] &= \frac{1}{\epsilon} \sum_{j=1}^{N} b_j \left[\sum_{k=1}^{M} p_{kj} \Big(w_k + \bar{e}_j \Big)^2 - \bar{e}_j^2 \right] \\
&= \frac{1}{\epsilon} \sum_{j=1}^{N} b_j \left[\sum_{k=1}^{M} p_{kj} w_k^2 \right] \\
&\geq \frac{1}{\epsilon} \sum_{j=1}^{N} b_j \left[\gamma \sum_{k=1}^{M} w_k^2 \right] \\
&= \frac{\gamma}{\epsilon} \sum_{j=1}^{N} b_j \left[\sum_{k=1}^{M} \Big(e_k - \bar{e}_j \Big)^2 \right] \\
&\geq \frac{\gamma}{\epsilon} ||\boldsymbol{e}||_2^2 \\
&> 0
\end{aligned} \tag{16}$$

Meanwhile we set $\bar{e}_j = \sum_{k=1}^{M} p_{kj} e_k$ and $e_k = w_k + \bar{e}_j$ and therefore we have $\sum_{k=1}^{M} p_{kj} w_k = \sum_{k=1}^{M} p_{kj} \left(e_k - \bar{e}_j \right) = 0$. At that time, the claim that the optimization function is strongly convex on the affine constraint subspace $\mathcal{M}$. □

Proposition 2. *$\widehat{\boldsymbol{f}}^{(k+1)}$ restricted to the subspace $\mathcal{M}$ as $\sum_{i=1}^{M} \widehat{f}_i^{(k+1)} = 0$ on Centering-BFGS.*

Proof. To fulfill the proof, we need to prove the following basic lemmas.

Lemma. *The gradient of $\mathfrak{J}_{\mathrm{OT}}(\cdot)$ satisfies the following relationship:*

$$\sum_{s=1}^{M} \frac{\partial \mathfrak{J}_{\mathrm{OT}}}{\partial \widehat{f}_s^{(\ell)}} = \sum_{s=1}^{M} \left[\sum_{j=1}^{N} \left[\frac{\exp\left(\frac{\widehat{f}_s^{(\ell)} - C_{sj}}{\epsilon} \right)}{\sum_{k=1}^{M} \exp\left(\frac{\widehat{f}_k^{(\ell)} - C_{kj}}{\epsilon} \right)} \beta_j \right] - \alpha_s \right] = 0 \tag{17}$$

Then we start the proof by deduction. That is, it is easy to verify that:

$$\begin{aligned}
\boldsymbol{H}_0 \mathbf{1} = \epsilon \boldsymbol{I} \quad \text{and} \quad \mathbf{1}^\top \widehat{\boldsymbol{f}}^{(1)} &= \mathbf{1}^\top \widehat{\boldsymbol{f}}^{(0)} - \zeta_0 \, \mathbf{1}^\top \boldsymbol{H}_k \nabla \mathfrak{J}_{\mathrm{OT}}(\widehat{\boldsymbol{f}}^{0)}) \\
&= -\epsilon \zeta_0 \mathbf{1}^\top \nabla \mathfrak{J}_{\mathrm{OT}}(\widehat{\boldsymbol{f}}^{(0)}) \\
&= 0
\end{aligned} \tag{18}$$

Therefore we assume that $\boldsymbol{H}_k \mathbf{1} = \epsilon \boldsymbol{I}$ and $\mathbf{1}^\top \widehat{\boldsymbol{f}}^{(k)} = 0$ at the k-th iteration. Then we can verify that:

$$\begin{aligned}
\mathbf{1}^\top \widehat{\boldsymbol{f}}^{(k+1)} &= \mathbf{1}^\top \widehat{\boldsymbol{f}}^{(k)} - \zeta_k \, \mathbf{1}^\top \boldsymbol{H}_k \nabla \mathfrak{J}_{\mathrm{OT}}(\widehat{\boldsymbol{f}}^{(k)}) \\
&= -\epsilon \zeta_k \mathbf{1}^\top \nabla \mathfrak{J}_{\mathrm{OT}}(\widehat{\boldsymbol{f}}^{(k)}) \\
&= 0
\end{aligned} \tag{19}$$

Meanwhile we notice that $\boldsymbol{v}_k^\top = \widehat{\boldsymbol{f}}^{(k+1)} - \widehat{\boldsymbol{f}}^{(k)}$ then we have:

$$\boldsymbol{v}_k^\top \mathbf{1} = \mathbf{1}^\top \left(\widehat{\boldsymbol{f}}^{(k+1)} - \widehat{\boldsymbol{f}}^{(k)} \right) = 0 \tag{20}$$

Likewise we have the results on $\boldsymbol{y}_k = \nabla \mathfrak{J}_{\mathrm{OT}} \left(\widehat{\boldsymbol{f}}^{(k+1)} \right) - \nabla \mathfrak{J}_{\mathrm{OT}} \left(\widehat{\boldsymbol{f}}^{(k)} \right)$ since:

$$\boldsymbol{y}_k^\top \mathbf{1} = \mathbf{1}^\top \left(\nabla \mathfrak{J}_{\mathrm{OT}} \left(\widehat{\boldsymbol{f}}^{(k+1)} \right) - \nabla \mathfrak{J}_{\mathrm{OT}} \left(\widehat{\boldsymbol{f}}^{(k)} \right) \right) = 0 \tag{21}$$

Then we can show that:

$$\begin{aligned}\boldsymbol{H}_{k+1}\mathbf{1} &= \left(\boldsymbol{I} - \frac{\boldsymbol{v}_k\boldsymbol{y}_k^\top}{\boldsymbol{v}_k^\top\boldsymbol{y}_k}\right)\boldsymbol{H}_k\left(\boldsymbol{I} - \frac{\boldsymbol{y}_k\boldsymbol{v}_k^\top}{\boldsymbol{v}_k^\top\boldsymbol{y}_k}\right)\mathbf{1} + \frac{\boldsymbol{v}_k\left(\boldsymbol{v}_k^\top\mathbf{1}\right)}{\boldsymbol{v}_k^\top\boldsymbol{y}_k} \\ &= \left(\boldsymbol{I} - \frac{\boldsymbol{v}_k\boldsymbol{y}_k^\top}{\boldsymbol{v}_k^\top\boldsymbol{y}_k}\right)\boldsymbol{H}_k\mathbf{1} \\ &= \epsilon\mathbf{1} - \epsilon\frac{\boldsymbol{v}_k\left(\boldsymbol{y}_k^\top\mathbf{1}\right)}{\boldsymbol{v}_k^\top\boldsymbol{y}_k} \\ &= \epsilon\mathbf{1}\end{aligned} \tag{22}$$

We verify that $\boldsymbol{H}_{k+1}\mathbf{1} = \epsilon\boldsymbol{I}$ and $\mathbf{1}^\top\widehat{\boldsymbol{f}}^{(k+1)} = 0$ at the $(k+1)$-th iteration and we claim the proof. □

Remark. *The optimization of Centering-BFGS satisfies the following inequality Jin et al. (2024) as:*

$$\frac{\mathfrak{J}_{\mathrm{OT}}(\widehat{\boldsymbol{f}}^{(k)}) - \mathfrak{J}_{\mathrm{OT}}(\widehat{\boldsymbol{f}}^*)}{\mathfrak{J}_{\mathrm{OT}}(\widehat{\boldsymbol{f}}^{(0)}) - \mathfrak{J}_{\mathrm{OT}}(\widehat{\boldsymbol{f}}^*)} \leq \left(1 - \frac{2\omega_a(1-\omega_b)}{L/\mu}\right)^k \tag{23}$$

where ω_a and ω_b denote the line search parameters as:

$$\begin{aligned}\mathfrak{J}_{\mathrm{OT}}(\widehat{\boldsymbol{f}}^{(k)} + \eta_t\boldsymbol{d}_t) &\leq \mathfrak{J}_{\mathrm{OT}}(\widehat{\boldsymbol{f}}^{(k)}) + \omega_a\eta_t\nabla\mathfrak{J}_{\mathrm{OT}}(\widehat{\boldsymbol{f}}^{(k)})^\top\boldsymbol{d}_t \\ \nabla\mathfrak{J}_{\mathrm{OT}}(\widehat{\boldsymbol{f}}^{(k)} + \eta_t\boldsymbol{d}_t)^\top\boldsymbol{d}_t &\geq \omega_b\nabla\mathfrak{J}_{\mathrm{OT}}(\widehat{\boldsymbol{f}}^{(k)})^\top\boldsymbol{d}_t\end{aligned} \tag{24}$$

where $\boldsymbol{d}_t$ denotes the search direction. This indicates that the algorithm achieves superlinear convergence Jin et al. (2024) since it is strongly convex with $1/\epsilon$-Lipschitz continuous gradient.

Proposition 3. *Given the optimal solutions of $\boldsymbol{s}^*$ in $\mathcal{J}_{\mathrm{OT}}$ and $\widehat{\boldsymbol{s}}^*$ in $\mathfrak{J}_{\mathrm{OT}}$, the general error bound can be calculated with ϵ:*

$$0 \leq \left\langle \boldsymbol{\alpha}\otimes\boldsymbol{\beta}, \widehat{\boldsymbol{s}}^* - \boldsymbol{s}^* \right\rangle \leq \epsilon\log M. \tag{25}$$

More specifically, the ℓ_1-error bound between $\boldsymbol{s}^*$ and $\widehat{\boldsymbol{s}}^*$ is given as:

$$\left\|\widehat{\boldsymbol{s}}^* - \boldsymbol{s}^*\right\|_1 \leq N\sqrt{\frac{2\epsilon M\log M}{\mu}} + NM\sqrt{\frac{2\epsilon\log M}{\mu}} + \epsilon NM\log M \tag{26}$$

Proof. Given the optimal solutions of $\boldsymbol{f}^*$ in $\mathcal{J}_{\mathrm{OT}}$ and $\widehat{\boldsymbol{f}}^*$ in $\mathfrak{J}_{\mathrm{OT}}$, it is obvious to verify the following relationships:

$$(\text{Optimal Condition}): \quad \begin{cases}\mathfrak{J}_{\mathrm{OT}}\left(\widehat{\boldsymbol{f}}^*\right) \leq \mathfrak{J}_{\mathrm{OT}}\left(\boldsymbol{f}^*\right) \\ \mathcal{J}_{\mathrm{OT}}\left(\boldsymbol{f}^*\right) \leq \mathcal{J}_{\mathrm{OT}}\left(\widehat{\boldsymbol{f}}^*\right).\end{cases} \tag{27}$$

Meanwhile we should remind the approximation error bound as shown below:

$$\begin{aligned}\text{error} &= \mathfrak{J}_{\mathrm{OT}}\left(\boldsymbol{f}^*\right) - \mathcal{J}_{\mathrm{OT}}\left(\boldsymbol{f}^*\right) \\ &= \sum_{j=1}^N \epsilon b_j\log\left[\sum_{k=1}^M\exp\left(\frac{f_k^* - C_{kj}}{\epsilon}\right)\right] - \sum_{j=1}^N\sup_{k\in[M]}\left[f_k^* - C_{kj}\right]b_j \\ &\leq \sum_{j=1}^N \epsilon b_j\left[\log\left(M\exp\left[\sup_{k\in[M]}\left(\frac{f_k^* - C_{kj}}{\epsilon}\right)\right]\right) - \log\left(\exp\left[\sup_{k\in[M]}\left(\frac{f_k^* - C_{kj}}{\epsilon}\right)\right]\right)\right] \\ &= \epsilon\log M.\end{aligned} \tag{28}$$

Finally we should further obtain that:

$$\sum_{j=1}^N b_j\left(\sup_{k\in[M]}\left[\widehat{f}_k^* - C_{kj}\right] - \epsilon\log\left[\sum_{k=1}^M\exp\left(\frac{\widehat{f}_k^* - C_{kj}}{\epsilon}\right)\right]\right) \leq 0 \quad \Rightarrow \quad \mathcal{J}_{\mathrm{OT}}\left(\widehat{\boldsymbol{f}}^*\right) - \mathfrak{J}_{\mathrm{OT}}\left(\widehat{\boldsymbol{f}}^*\right) \leq 0. \tag{29}$$

Therefore we can achieve the first important results as shown:

$$\begin{cases} \mathfrak{J}_{\mathrm{OT}}\left(\widehat{\boldsymbol{f}}^*\right) \leq \mathfrak{J}_{\mathrm{OT}}\left(\boldsymbol{f}^*\right) \\ \mathfrak{J}_{\mathrm{OT}}\left(\boldsymbol{f}^*\right) - \mathcal{J}_{\mathrm{OT}}\left(\boldsymbol{f}^*\right) \leq \epsilon \log M \end{cases} \Rightarrow \quad \mathfrak{J}_{\mathrm{OT}}\left(\widehat{\boldsymbol{f}}^*\right) - \mathcal{J}_{\mathrm{OT}}\left(\boldsymbol{f}^*\right) \leq \epsilon \log M. \tag{30}$$

Likewise we can obtain that:

$$\begin{cases} \mathcal{J}_{\mathrm{OT}}\left(\boldsymbol{f}^*\right) \leq \mathcal{J}_{\mathrm{OT}}\left(\widehat{\boldsymbol{f}}^*\right) \\ \mathcal{J}_{\mathrm{OT}}\left(\widehat{\boldsymbol{f}}^*\right) - \mathfrak{J}_{\mathrm{OT}}\left(\widehat{\boldsymbol{f}}^*\right) \leq 0 \end{cases} \Rightarrow \quad \mathfrak{J}_{\mathrm{OT}}\left(\widehat{\boldsymbol{f}}^*\right) - \mathcal{J}_{\mathrm{OT}}\left(\boldsymbol{f}^*\right) \geq 0. \tag{31}$$

By combining the above results, we have:

$$0 \leq \mathfrak{J}_{\mathrm{OT}}\left(\widehat{\boldsymbol{f}}^*\right) - \mathcal{J}_{\mathrm{OT}}\left(\boldsymbol{f}^*\right) \leq \epsilon \log M. \tag{32}$$

Therefore, we can finally obtain the results:

$$\begin{aligned} \Delta_s = \left\langle \boldsymbol{\alpha} \otimes \boldsymbol{\beta}, \widehat{\boldsymbol{s}}^* - \boldsymbol{s}^* \right\rangle &= \sum_{i=1}^{M}\sum_{j=1}^{N} \alpha_i \beta_j \left(\widehat{s}_{ij}^* - s_{ij}^*\right) \\ &= \sum_{i=1}^{M}\sum_{j=1}^{N} \alpha_i \beta_j \left(C_{ij} - \widehat{f}_i^* - \widehat{g}_j^* - C_{ij} + f_i^* + g_j^*\right) \\ &= \sum_{i=1}^{M}\sum_{j=1}^{N} \alpha_i \beta_j \left(- \widehat{f}_i^* - \widehat{g}_j^* + f_i^* + g_j^*\right) \\ &= -\sum_{j=1}^{N} \widehat{g}_j \beta_j - \sum_{i=1}^{M} \widehat{f}_i \alpha_i - \left(-\sum_{j=1}^{N} g_j^* \beta_j - \sum_{i=1}^{M} f_i^* \alpha_i\right) \\ &= \sum_{j=1}^{N} \epsilon \beta_j \log\left[\sum_{i=1}^{M} e^{\frac{\widehat{f}_i - C_{ij}}{\epsilon}}\right] - \sum_{i=1}^{M} \widehat{f}_i \alpha_i - \left(\sum_{j=1}^{N} \beta_j \sup_{k\in[M]} \left(f_k^* - C_{kj}\right) - \sum_{i=1}^{M} f_i^* \alpha_i\right) \\ &= \mathfrak{J}_{\mathrm{OT}}\left(\widehat{\boldsymbol{f}}^*\right) - \mathcal{J}_{\mathrm{OT}}\left(\boldsymbol{f}^*\right). \end{aligned} \tag{33}$$

Therefore we can claim the proof as:

$$0 \leq \Delta_s = \langle \boldsymbol{\alpha} \otimes \boldsymbol{\beta}, \widehat{\boldsymbol{s}}^* - \boldsymbol{s}^* \rangle = \mathfrak{J}_{\mathrm{OT}}\left(\widehat{\boldsymbol{f}}^*\right) - \mathcal{J}_{\mathrm{OT}}\left(\boldsymbol{f}^*\right) \leq \epsilon \log M. \tag{34}$$

More importantly, if we consider the general case that $\alpha_i = \frac{1}{M}$ and $\beta_j = \frac{1}{N}$, the error bound can be determined as:

$$0 \leq \frac{1}{MN} \sum_{i=1}^{M}\sum_{j=1}^{N} \left(\widehat{s}_{ij}^* - s_{ij}\right) \leq \epsilon \log M. \tag{35}$$

It indicates that with given smaller ϵ will lead to much better approximation for $\widehat{\boldsymbol{s}}$. Meanwhile we can obtain the following results due to the strongly convex property:

$$\mathfrak{J}_{\mathrm{OT}}\left(\boldsymbol{f}^*\right) \geq \mathfrak{J}_{\mathrm{OT}}\left(\widehat{\boldsymbol{f}}^*\right) + \frac{\mu}{2}\left\|\boldsymbol{f}^* - \widehat{\boldsymbol{f}}^*\right\|_2^2 \tag{36}$$

Therefore the error between $\boldsymbol{f}$ and $\widehat{\boldsymbol{f}}$ can be bounded as:

$$\left\|\boldsymbol{f}^* - \widehat{\boldsymbol{f}}^*\right\|_2 \leq \sqrt{\frac{2\epsilon \log M}{\mu}} \tag{37}$$

We can further utilize the property of $|\inf(\boldsymbol{x}) - \inf(\boldsymbol{y})| \leq \max_k |x_k - y_k| = \|\boldsymbol{x} - \boldsymbol{y}\|_\infty$ with $\|\cdot\|_\infty \leq \|\cdot\|_2$:

$$\left|\inf_{k\in[M]}\left(C_{kj} - f_k^*\right) - \inf_{k\in[M]}\left(C_{kj} - \widehat{f}_k^*\right)\right| \leq \left\|\boldsymbol{f}^* - \widehat{\boldsymbol{f}}^*\right\|_\infty \leq \left\|\boldsymbol{f}^* - \widehat{\boldsymbol{f}}^*\right\|_2 \leq \sqrt{\frac{2\epsilon \log M}{\mu}} \tag{38}$$

Table 7: Classification accuracy (%) on *Office-31* and *ImageCLEF* for unsupervised domain adaptation

Method for UDA	A→W	D→W	W→D	A→D	D→A	W→A	Avg	I→P	P→I	I→C	C→I	C→P	P→C	Avg
Source-only	82.3	84.5	99.8	98.0	75.6	74.5	85.8	74.8	83.9	91.5	78.0	65.5	91.3	80.7
PPOT	94.6	96.1	**100.0**	98.5	86.3	85.4	93.5	82.0	95.0	98.2	96.1	81.8	97.8	91.9
PPOT+SSNS	94.7	96.3	**100.0**	98.6	86.9	85.8	93.7	82.3	95.2	98.4	96.3	82.0	97.9	92.0
PPOT+CSG-Ent	95.0	96.6	**100.0**	98.9	87.2	86.8	94.1	82.7	95.5	98.4	96.6	82.5	98.1	92.3
PPOT+CSG-Norm	**95.8**	**97.3**	**100.0**	**99.5**	**88.4**	**87.2**	**94.7**	**83.9**	**96.8**	**99.1**	**97.7**	**83.6**	**98.8**	**93.3**

Table 8: Classification accuracy (%) on *Office-Home* for unsupervised domain adaptation

Method for UDA	Ar→Cl	Ar→Pr	Ar→Rw	Cl→Ar	Cl→Pr	Cl→Rw	Pr→Ar	Pr→Cl	Pr→Rw	Rw→Ar	Rw→Cl	Rw→Pr	Avg
Source-only	46.3	67.5	75.9	59.1	59.9	62.7	58.2	41.8	74.9	67.4	48.2	74.2	61.4
PPOT	57.5	75.8	83.8	68.2	76.9	76.0	66.3	55.4	82.5	75.6	61.1	84.4	72.0
PPOT+SSNS	58.2	77.0	85.1	68.8	77.2	76.6	66.5	55.7	82.9	76.1	61.8	84.6	72.5
PPOT+CSG-Ent	59.6	78.2	86.1	70.7	78.3	77.9	66.9	56.0	83.4	76.8	62.5	85.1	73.5
PPOT+CSG-Norm	**61.4**	**79.5**	**87.2**	**72.3**	**79.8**	**79.1**	**68.7**	**57.8**	**84.9**	**78.2**	**64.3**	**86.5**	**75.0**

Furthermore, the error bound on $\boldsymbol{g}$ and $\widehat{\boldsymbol{g}}$ can be obtained as:

$$\begin{aligned}\left\|\boldsymbol{g}^* - \widehat{\boldsymbol{g}}^*\right\|_1 \leq \sum_{j=1}^{N}\Bigg|\inf_{k\in[M]}\left(C_{kj} - f_k^*\right) - \inf_{k\in[M]}\left(C_{kj} - \widehat{f}_k^*\right)\Bigg| \\ + \sum_{j=1}^{N}\Bigg|\inf_{k\in[M]}\left(C_{kj} - \widehat{f}_k^*\right) + \epsilon\log\left[\sum_{k=1}^{M}\exp\left(\frac{\widehat{f}_k^* - C_{kj}}{\epsilon}\right)\right]\Bigg| \\ \leq N\sqrt{\frac{2\epsilon\log M}{\mu}} + \epsilon N\log M\end{aligned} \tag{39}$$

Finally, the error bound over multipliers can be shown with $||\max(0,\boldsymbol{x}) - \max(0,\boldsymbol{y})||_1 \leq ||\boldsymbol{x}-\boldsymbol{y}||_1$*:*

$$\begin{aligned}\left\|\boldsymbol{s}^* - \widehat{\boldsymbol{s}}^*\right\|_1 \leq \sum_{i=1}^{M}\sum_{j=1}^{N}\left|\left(f_i^* - \widehat{f}_i^*\right) + \left(g_j^* - \widehat{g}_j^*\right)\right|_1 \leq N\left\|\boldsymbol{f}^* - \widehat{\boldsymbol{f}}^*\right\|_1 + M\left\|\boldsymbol{g}^* - \widehat{\boldsymbol{g}}^*\right\|_1 \\ \leq N\sqrt{\frac{2\epsilon M\log M}{\mu}} + NM\sqrt{\frac{2\epsilon\log M}{\mu}} + \epsilon NM\log M\end{aligned} \tag{40}$$

This statement means that as the value of ϵ becomes infinitely small, the estimate $\widehat{\boldsymbol{s}}$ converges toward the true value $\boldsymbol{s}$. □

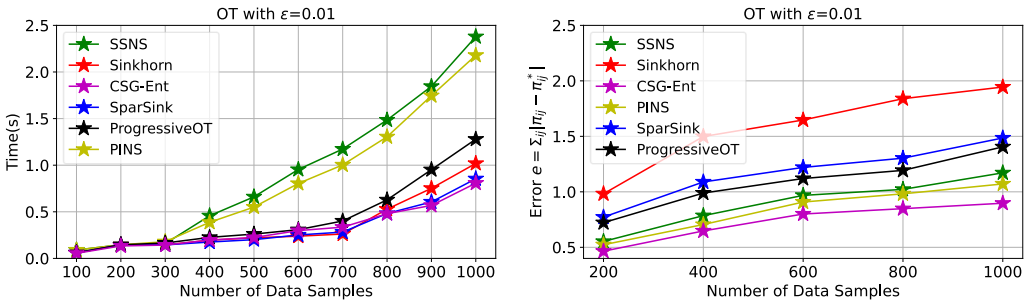
C Experiments

C.1 Experimental setup

Baselines. We involve **DANN** Ganin et al. (2016), **CDAN** Long et al. (2018), **DeepJDOT** Damodaran et al. (2018), **RWOT** Xu et al. (2020) and **PPOT** Wang et al. (2024) as the model baselines for the unsupervised domain adaptation task.

- **DANN** Ganin et al. (2016) is a new representation learning approach for domain adaptation, in which data at training and test time come from similar but different distributions.
- **CDAN** Long et al. (2018) is a principled framework that conditions the adversarial adaptation models on the discriminative information conveyed in the classifier predictions.
- **DeepJDOT** Damodaran et al. (2018) first adopts optimal transport to solve the domain adaptation problem within a deep learning framework.
- **RWOT** Xu et al. (2020) fully utilizes information of not only the labeled source domain but also the discriminative representation of the unknown class in the target domain.
- **PPOT** Wang et al. (2024) is a novel Probability-Polarized OT framework, which can characterize the structure of OT plan explicitly.

We involve **ID** Wu et al. (2018), **ProtoNCE** Li et al., **CDS** Kim et al. (2021), **PCS** Yue et al. (2021), **DD** Hu & Lee (2022) and **ProtoOT** Li et al. (2024) as the model baselines for unsupervised cross-domain retrieval task.



(a) The time consumption on OT **(b) The absolute error on OT**

Figure 5: The time consumption and error analysis on classic optimal transport on synthetic datasets with more baselines.

- **ID** Wu et al. (2018) formulates a non-parametric classification problem at the instance-level, with noise-contrastive estimation for computational efficiency.
- **ProtoNCE** Li et al. introduces prototypes as latent variables to find the maximum-likelihood estimation of parameters in the Expectation-Maximization framework.
- **CDS** Kim et al. (2021) uses in-domain instance discrimination in order to learn discriminative features for new data and cross-domain matching is used to learn domain-invariant features.
- **PCS** Yue et al. (2021) is an prototypical cross-domain self-supervised framework, which leverages in-domain prototypical contrastive learning and cross-domain prototypical self-supervision.
- **DD** Hu & Lee (2022) introduces cluster-wise contrastive learning for class semantic-aware features, with a distance-of-distance loss for domain discrepancy minimization.
- **ProtoOT** Li et al. (2024) is an optimal transport formulation for UCIR, which unifies intra-domain representation learning and cross-domain alignment.

Experimental results on synthetic datasets. We conduct experiments on the synthetic datasets with more baselines including **SparSink** Li et al. (2023), **ProgressiveOT** Kassraie et al. (2024) and **PINS** Wu et al. (2025). (1) **SparSink** Li et al. (2023) adopts the importance sparsification mechanism to filter matching noise. (2) **ProgressiveOT** Kassraie et al. (2024) involves progressive entropic map with interpolation on solving OT. (3) **PINS** Wu et al. (2025) adopts proximal iterations with sparse Newton with Sinkhorn solver on OT. We report the time consumption and absolute error in Fig.5. Notably, both PINS and ProgressiveOT require higher computational time due to their multiple complex procedures. Meanwhile, CSG-Ent achieves a computational efficiency comparable to SparSink, maintaining lower absolute error with similar time consumption. In summary, the results indicates the efficacy of the proposed CSG solver.

Experimental results on real-world applications. We also conduct more experiments on Office-31, ImageCLEF and Office-Home in Table 6-7. Specifically, we incorporate SSNS into the PPOT framework, referred to as PPOT+SSNS. The performance gains achieved by PPOT+SSNS over PPOT validate the significance of precise OT computation. Moreover, the fact that PPOT+CSG surpasses both PPOT and PPOT+SSNS underscores the effectiveness of our proposed approach.