GRADUAL FINE-TUNING WITH GRAPH ROUTING FOR MULTI-SOURCE UNSUPERVISED DOMAIN ADAPTATION

Yao Ma[*,†], **Samuel Louvan**[†], **Zhunxuan Wang**[†]
Amazon
United Kingdom
{yaoom,slouvan,wzhunxua}@amazon.co.uk

ABSTRACT

Multi-source unsupervised domain adaptation aims to leverage labeled data from multiple source domains for training a machine learning model to generalize well on a target domain without labels. Source domain selection plays a crucial role in determining the model's performance. It relies on the similarities amongst source and target domains. Nonetheless, existing work for source domain selection often involves heavyweight computational procedures, especially when dealing with numerous source domains and the need to identify the best ones from them. In this paper, we introduce a framework for gradual fine tuning (GFT) of machine learning models on multiple source domains. We represent multiple source domains as an undirected weighted graph. We then give a new generalization error bound for GFT along any path within the graph, which is used to determine the optimal path corresponding to the optimal training order. With this formulation, we introduce three lightweight graph-routing strategies which tend to minimize the error bound. Our best strategy improves 2.3% of accuracy over the state-of-the-art on Natural Language Inference (NLI) task and achieves competitive performance on Sentiment Analysis (SA) task, especially a 3.9% improvement on a more diverse subset of data we use for SA.

1 INTRODUCTION

Domain adaptation has been shown to succeed in training deep neural networks with limited data, particularly when the acquisition of labeled data can be costly in real-world applications. In practice, it is often favorable to train a model with data from related domains. Accordingly, the effectiveness of domain adaptation highly depends on the quality and similarity of the source domains' datasets. In cases where few or no labeled samples are available in the domain we target, developing a methodology to train a model without direct supervision on target domain becomes necessary. This approach is known as unsupervised domain adaptation.

Extensive research has been conducted on theoretical analysis and empirical algorithms that minimize the generalization error (risk) of the trained model on target domain. Mansour et al. (2009) has demonstrated that the generalization error depends on both generalization error on the source domain and the discrepancy between source and target domains. In the pursuit of minimizing the discrepancy, certain approaches (Ruder & Plank, 2017; Liu et al., 2019) have been proposed to select source domains that are close to the target domain. Nevertheless, this selection process can be costly as it introduces an additional step prior to the model training. Moreover, the source domain selection tends to discard distant domains, while we assert that the distant domains can actually provide valuable training benefits for the target domain. For this reason, we propose a lightweight and efficient gradual fine-tuning (GFT) framework that can take advantage of all available source domains. By sequentially fine-tuning a model on multiple data sources, we aim to address the limitation of existing methods and unlock the potential benefits from distant domains during the training process for the target domain. The underlying intuition of our approach is to *gradually* guide a model to its optimal solution through sequentially fine-tuning the model on different source data whose distribution progressively aligns with the target domain. With this formulation, the model learns from data that spans a wide range of distributions, and ultimately leading it towards a better performance on the target domain. The gradual alignment of source data distributions with the target domain is crucial in enhancing adaptability and performance.

The contribution of our work is two-fold. First, motivated by Wang et al. (2022), we construct theoretical analysis and give the generalization error of the proposed GFT algorithms. Based on our theory, we introduce graph routing

[*] Correspondence to: Yao Ma <yaoom@amazon.co.uk>.
[†] These authors contributed equally to this work.

strategies for determining the optimal paths through an undirected graph constructed by source domains and Wasserstein distances thereof. Second, we present empirical results for the GFT algorithms on two Natural Language Processing (NLP) tasks that are commonly used to demonstrate the effectiveness of a domain adaptation algorithms. We show that the performance of GFT algorithms does not highly depends on the discrepancy of sources and target domains as long as there exists a path along domains such that the distance between consecutive domains is small and the data magnitude on the path is large. Both empirical and theoretical results indicate that through GFT framework the model achieves better performance than baselines.

2 Related Work

Domain Adaptation aims to learn a model from source domains that generalize well to a target domain. In general, there are three types of approaches on domain adaptation, namely model based, data centric, or hybrid approaches (Pan & Yang, 2010; Ramponi & Plank, 2020). Model based approaches typically aim to learn an invariant representation for domain shift (Ganin & Lempitsky, 2015; Zhao et al., 2019; Li et al., 2021). However, there is no theoretical analysis presented in these works. Existing works (Huang et al., 2006; Mansour et al., 2009; Courty et al., 2017) also have proposed on determining the value of sources, including the number of samples, the quality of data, and the discrepancy between source and target. The data centric approaches typically perform data selection to select source domains that are more similar to the target domain in terms of data distribution. Different kinds of metrics have been used to measure domain similarity, for example in NLP, Jensen Shannon similarity over word distribution Ruder & Plank (2017) is one of the common metrics to be used. One of the downside of data selection methods is, it usually involves expensive computation in addition to the model training such as data selection using Bayesian Optimization (Ruder & Plank, 2017) and Reinforcement Learning (Liu et al., 2019). Instead of performing data selection, our work attempts to eliminate this source selection stage through gradual fine-tuning.

Gradual Domain Adaptation (GDA) is proposed for the problem of unsupervised domain adaption which assumes the existence of a set of unsupervised datasets from intermediate domains. A pre-trained model is trained using labeled data from source domain. And then the model is trained and updated sequentially with pseudo-labeled predicted by the current model by minimizing the empirical loss w.r.t. the pseudo-labels. Kumar et al. (2020) have shown the GDA achieves a small generalization error when the distribution shift between two consecutive domains is small and the error in source domain is small. Wang et al. (2022) further proved an improved generalization bound which only grows linearly with the number of intermediate domains. Chen & Chao (2021) considers the problem of gradual domain adaption when the intermediate domains are not clearly defined, and proposed Intermediate Domain Labeler (IDOL) to assign scores to all unlabeled samples and group samples into different domains accordingly.

Learning from multiple sources. As the data from a single domain could be very limited, Mancini et al. (2018); Peng et al. (2019); Zhao et al. (2018) consider the problem of multi-source unsupervised domain adaptation which assumes the source domain examples are multi-modal, i.e., the samples are drawn from different distributions. Multi-source domain adaptation considers the setting when training with joint data samples from multiple sources and little or no labeled training data from target domain. Crammer et al. (2008) presented a bound on the expected error incurred by using $\mathcal{K}$ number of data sources. By applying the bound, an optimal number of data sources to train a model can be achieved by measuring the discrepancy of data sources. Mansour et al. (2008) considers a similar problem which assumes the target distribution is a mixture of distributions of multiple sources. The results show that there exists a distribution weighted mixture combining rule that has a small enough loss with respect to any consistent target function and any mixture of the data distributions. Furthermore, Mansour et al. (2009) relaxes the assumption and shows that there exists a distribution weighted combination of the source hypotheses whose loss can be bounded with respect to the maximum loss of the source hypotheses and the Renyi divergence.

3 Problem Setup

In this paper, we consider a binary classification problem. We denote $\mathcal{X}$ as the feature space and $\mathcal{Y} = \{-1, +1\}$ as the label space. We assume that the feature space is compact and bounded by an L^2 ball, i.e., $\mathcal{X} \subseteq \{\boldsymbol{x} \in \mathbb{R}^d : \|\boldsymbol{x}\|_2 \leq 1\}$. A domain is defined by a joint data distribution D with sample space $\mathcal{X} \times \mathcal{Y}$, where $\mathcal{X}$ and $\mathcal{Y}$ are mapped by a labeling function $f : \mathcal{X} \to \mathcal{Y}$. For any domain, we assume that the data distribution is unknown, and we draw n sample pairs $S = \{\boldsymbol{x}_i, y_i\}_{i=1}^n$ from D independently. A hypothesis (classifier) is represented as a function $h : \mathcal{X} \to \mathcal{Y}$. Let $\mathcal{H}$ denotes the hypothesis class, which is a set of classifiers. In this paper, we assume any classifier $h \in \mathcal{H}$ is Lipschitz continuous with respect to the feature vector $\boldsymbol{x}$. More precisely, for any classifier $h \in \mathcal{H}$, there exists a real constant $R \geq 0$ such that $\forall \boldsymbol{x}, \boldsymbol{x}' \in X$, $|h(\boldsymbol{x}) - h(\boldsymbol{x}')| \leq R \cdot \|\boldsymbol{x} - \boldsymbol{x}'\|_2$.

Then, for any h, we can define its expected loss (risk) with loss function $\mathcal{L}$ on data distribution D as

$$\epsilon_D(h) = \mathbb{E}_{\{\boldsymbol{x},y\}\sim D}[\mathcal{L}(h(\boldsymbol{x}), y)].$$

Similarly, we denote the empirical loss on n samples as

$$\hat{\epsilon}_D(h, f) = \frac{1}{n}\sum_{i=1}^{n}\mathcal{L}(\boldsymbol{x}_i, y_i).$$

To provide bounds on the expected loss, we make an assumption that the considered loss function $\mathcal{L}$ is also Lipschitz continuous w.r.t. the input of the function. That is, there exists a real constant $L \geq 0$, such that $\forall y, y' \in \mathcal{Y}$, we have $|\mathcal{L}(\cdot, y) - \mathcal{L}(\cdot, y')| \leq L\|y - y'\|_2$.

In standard supervised learning, a model is trained to minimize the empirical loss (i.e., training loss) using n samples. The trained model is expected to perform well on a test dataset when the test data and training data are drawn from the same distribution D. However, this assumption does not hold in the context of domain adaptation, where the test data (target domain) is not drawn from the same distribution as the training data (source domain(s)).

Problem Statement. The objective of domain adaptation is to learn a classifier that minimizes the expected loss or risk on test data from the target domain, given the training data is from multiple source domains without knowing the underlying joint distribution. Formally, the loss is

$$\epsilon_T(h) = \min_{h\in\mathcal{H}} \mathbb{E}_{\{\boldsymbol{x},y\}\in D_T}[\mathcal{L}(h(\boldsymbol{x}), y)].$$

We assume the model has access to D_T as a distribution but does not have access to its data. Instead, certain number of samples drawn from other different distributions are the only labeled resource available for training. Different from most of the single domain adaptation set-ups, we consider a problem where there are K source domains and a single target domain. We denote the data distribution and data samples for the t-th domain with D_t and S_t respectively. All data distributions are unknown, we are only given K set of data samples $S_t, t = 1, ..., K$ drawn from distributions $D_1, ..., D_K$ respectively. Each D_t has different level of discrepancy with D_T. Existing work (Ramesh Kashyap et al., 2021; Ruder & Plank, 2017) has been proposed to evaluate discrepancy measurement between source and target, e.g., KL-divergence (Kullback & Leibler, 1951), Jansen-Shannon divergence (Lin, 1991), Renyi divergence (Rényi, 1961), and Wasserstein-p distance (Villani, 2009). In this paper, we use Wasserstein-p as a distance metric on a space of probability measures, which has constantly been a reliable measurement for achieving good transfer performance (Villani, 2009; Shen et al., 2018). For any distribution D_1 and D_2, the Wasserstein-p distance is defined by the value of the following minimization problem:

$$W_p(D_1, D_2) = \inf_{\pi\in\Gamma(D_1,D_2)} \int \|\boldsymbol{z}_1 - \boldsymbol{z}_2\|_p \mathrm{d}\pi(\boldsymbol{z}_1, \boldsymbol{z}_2),$$

where $\Gamma(D_1, D_2) = \{\pi | \int \pi(\boldsymbol{z}_1, \boldsymbol{z}_2)\mathrm{d}\boldsymbol{z}_1 = D_1(\boldsymbol{z})$ and $\int \pi(\boldsymbol{z}_1, \boldsymbol{z}_2)d\boldsymbol{z}_2 = D_2(\boldsymbol{z}), \boldsymbol{z} \in \mathcal{X} \times \mathcal{Y}\}$ is the set of joint distributions with marginals D_1 and D_2. In reality, the estimation of Wasserstein-p distance of two distributions with finite number of samples could be challenging. The Sinkhorn algorithm (Chizat et al., 2020) provides a practical way to estimate Wasserstein distance by solving an entropy regularized minimization problem. In this work, we apply this estimator to evaluate the distance between domains.

4 Gradual Fine-Tuning

In this section, we present our GFT approach for training on multiple source domains sequentially. Our method is inspired by the fact that the generalization error of a trained classifier increases linearly with the distance between the initial and the final parameter values. Previous research (Mansour et al., 2009) has shown that the target error depends on both the source error and the discrepancy between the source domain data distributions D_S and target domain data distribution D_T. This relationship explains why domain adaptation often works well in practice. However, existing methods have not fully exploited these insights.

Our GFT approach addresses this gap by gradually updating the model based on the source domains in sequence. We use graph routing algorithms to determine the order of updates, ensuring that each update minimizes the total error on all previous source domains while maximizing the accuracy on the current one. This way we ensure that the model converges faster and performs better overall compared to traditional single-source fine-tuning approaches. The GFT approach provides a principled framework for handling multi-source domain adaptation problems, allowing us to leverage the benefits of distant sources without sacrificing performance on the target domain.

Given K labeled datasets $S_k, k = 1, ..., K$ from K sources and an unlabeled dataset T from the target domain, we want to quantify the similarity between the source domains and the target domain. To achieve this, we employ the Wasserstein-p distance, specifically utilizing the Sinkhorn divergence estimation method (Chizat et al., 2020), to

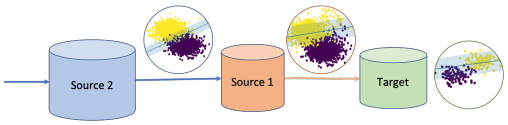


Figure 1: GFT illustration for 2-source domain adaptation with linear binary SVM. Source 1 has a distribution close to target but its scale is diminutive, whereas Source 2 has a large size but diverges further from the target. Model is first trained from scratch on Source 2, giving a clear hyperplane splitting two classes. Then fine-tuned on Source 1, shifting the hyperplane towards Source 1 distribution. Evaluation on target demonstrates the efficacy of GFT.

measure the distance between each pair of source domains. This yields a symmetric $(K+1)\times(K+1)$ matrix W whose i,j-th entry is $W_p(D_i, D_j)$.

For a given disparity matrix W, we model those multiple domains as a Wasserstein geometric graph $G=(V,E)$, where vertices V are domains and edges E are pairs of domains weighted by their Wasserstein distances. Any path to the target domain in G represents a GFT trajectory to train a classifier. Notably, the resulting graph G is complete when no threshold is applied to the maximum Wasserstein distance. However, to ensure that the disparity graph G is meaningful and easier to interpret, we introduce a threshold value τ for the Wasserstein distance. Specifically, the disparity graph G satisfies $\forall i,j\in V, (S_i,S_j)\in E$ if and only if the distance $W_{i,j}<\tau$. The threshold parameter τ plays a crucial role in ensuring the efficacy of the GFT algorithm. Intuitively, it serves as a bound on the lengths of edges in the graph, preventing excessive errors from propagating through the network. Specifically, the threshold guarantees that every edge in the graph is bounded by a small positive constant, which is essential for maintaining a low expected error rate. By applying the threshold, the disparity graph induced by W gets pruned, thereby enhancing the accuracy of the algorithm because it ensures that paths between high-discrepancy sources are not possible. On the other hand, we lack the access to target labels, indicating that direct Wasserstein distance calculation is not feasible. Therefore, we adopt a common approach in unsupervised domain adaptation, where pseudo-labels are generated by a pre-trained classifier on sources to make it Wasserstein measurable between source and target domains.

Now, we formally present our GFT algorithm, which leverages the disparity graph G defined earlier. For any path in G, our approach trains the model iteratively along the path using empirical risk minimization for each dataset. At each step, we start with the previously train model $\hat{h}_{t-1}$ and fine-tune it on S_t by minimizing the empirical loss $\mathcal{L}_t(h)$ defined as

$$\hat{h}_t = \frac{1}{n_t}\arg\min_{h\in\mathcal{H}} \sum_{(x,y)\in S_t} \mathcal{L}(h(x),y). \tag{1}$$

In summary, our GFT algorithm provides a flexible framework to train a model on multiple datasets without merging them into a single one. While the framework itself does not explicitly define a specific path for domain adaptation, a path from the most distant source to the closest source with the minimum sum of weights may lead to better performance in unsupervised domain adaptation. An illustration of example GFT training with two sources is shown in Figure 1, where the path is "Source 2 $\rightarrow$ Source 1". We also conduct initial experiments on the 2-source example and GFT achieves the best test accuracy compared to training on individual and combined sources. See appendix B for details.

5 Theoretical Analysis

In this section, we present the generalization error bound of the classifier trained with GFT algorithm along any path in the disparity graph G. We first recall the result from Wang et al. (2022) which shows the error difference of any classifier h over shifted data distribution is bounded by the Wasserstein-1 distance.

Lemma 5.1. *Given two joint distributions D_1 and D_2 over $X\times Y$, the expected loss of a classifier h satisfies*

$$|\epsilon_{D_1}(h)-\epsilon_{D_2}(h)| \le L\sqrt{R^2+1}W_1(D_1,D_2). \tag{2}$$

Lemma 5.1 gives the performance discrepancy bound of a classifier between two different datasets. A classifier produces similar errors on two data distributions that has smaller Wasserstein-1 distance. Applying this lemma, we bound the expected errors of two consecutive classifiers for the proposed GFT algorithm as

$$\epsilon_{t+1}(\hat{h}_{t+1})-\epsilon_t(\hat{h}_t) \le \frac{4B\sqrt{2}L}{\sqrt{n_{t+1}}} + 4B\sqrt{\frac{\log 1/\delta}{2n_{t+1}}} + L\sqrt{R^2+1}W_p(D_{t+1},D_t),$$

where $\Delta_{t,t+1}=\Delta_{t+1,t}=W_p(D_{t+1},D_t)$.The first and second terms are from generalization error bound under the assumption that the Rademacher complexity of the hypothesis space satisfies $\mathcal{R}_n(\mathcal{H})\le\frac{B}{\sqrt{n}}$. Note that although the above result is very similar to the bound in Wang et al. (2022), this result is different since the difference between labels

of two consecutive datasets is considered in this paper. This bound is essential to analyze the generalization error. The intuition is that the gradual fine-tuning works well only when the consecutive datasets are close enough.

We then present the theorem for bounding the generalization error of the gradual fine-tuned model under stated assumptions. We follow the same line of the analysis in Wang et al. (2022) which treats the training procedure as an online learning problem. More specifically, we sequentially train the final classifier with κ datasets consists of $\sum_{t=1}^{\kappa} n_t$ training samples. The result in Kuznetsov & Mohri (2020) shows the discrepancy measurement between distributions provides a tight generalization error bound for the final classifier with the target data distribution. For any $\mathbf{q}_\kappa = (q_1^1, ..., q_1^{n_1}, q_2^1 ..., q_2^{n_2}, ..., q_\kappa^1, ..., q_\kappa^{n_\kappa}) \in \mathcal{R}^{\sum_{t=1}^{\kappa} n_t}$, the discrepancy measurement is defined as

$$\text{disc}(\mathbf{q}_\kappa) = \sup_{h \in \mathcal{H}} \left(\epsilon_\kappa(h) - \sum_{t=1}^{\kappa} \sum_{\tau=1}^{n_t} q_t^\tau \epsilon_t(h) \right).$$

The following bound holds for any real number sequence $\mathbf{q}_\kappa$ with at least probability $1 - \delta$,

$$\epsilon_T(h_\kappa) \leq \sum_{t=1}^{\kappa} \sum_{i=1}^{n_t} q_t^i \epsilon_i(h_\kappa) + \|\mathbf{q}_\kappa\|_2 + \text{disc}(\mathbf{q}_\kappa) + 6B \sqrt{4\pi \log \sum_{t=1}^{\kappa} n_t \mathcal{R}^{\text{seq}}_{n_{1:\kappa}}} + B\|\mathbf{q}_\kappa\|_2 \sqrt{8 \log 1/\delta},$$

where $\mathcal{R}^{\text{seq}}_{n_{1:\kappa}}$ is the sequential Rademacher complexity Rokhlin (2017) of $\mathcal{H}$ with loss function $\mathcal{L}$.

By applying this result, we are able to analyze the expected error of the GFT algorithm with the optimal weights for discrepancy measure as $\mathbf{q}_\kappa = \left(\frac{1}{n_1 \kappa}, ... \frac{1}{n_1 \kappa}, ..., \frac{1}{n_\kappa \kappa}, ..., \frac{1}{n_\kappa \kappa} \right)$.Here, we present our main theory for the generalization error bound of the proposed gradual fine-tuning algorithm as follows whose proof is provided in the appendix.

Theorem 5.2. *The expected error of the final classifier h_κ in the target domain T is upper bounded with probability at least $1 - \delta$ as*

$$\begin{aligned} \epsilon_T(\hat{h}_\kappa) \leq \quad & L\sqrt{R^2 + 1} W_p(D_T, D_\kappa) + \hat{\epsilon}_1(\hat{h}_1) + (1 + \tfrac{1}{\kappa}) L \sqrt{R^2 + 1} \sum_{t=1}^{\kappa - 1} \Delta_{t,t+1} \\ & + \frac{(4\sqrt{2}LB + 2\sqrt{2}B\sqrt{\log(1/\delta)})(\kappa - 1)}{\kappa} \sum_{t=0}^{\kappa - 1} \frac{1}{\sqrt{n_{t+1}}} + 6B\sqrt{4\pi \log \sum_{t=1}^{\kappa} n_t} \mathcal{R}^{\text{seq}}_{\kappa}(\mathcal{H}) \\ & + \frac{B\sqrt{8 \log(1/\delta)} + 1}{\kappa} \sqrt{\sum_{t=1}^{\kappa} \frac{1}{n_t}}. \end{aligned} \tag{3}$$

Theorem 5.2 indicates that GFT achieves the minimum error bound when the model is sequentially trained along the optimal path from the furthest domain to the closest source domain with respect to the target domain. The **1-st** term in Equation 3 is proportional to Wasserstein distance between the last domain and the target. When all domains are distant from the target, the **1-st** term naturally becomes large. The **2-nd** term represents the path length across the selected source domains, which depends on the distance between source domains and the number of sources selected. On the other hand, sample sizes dominate the **4,5,6-th** terms. As the number of samples increases, the **4,5,6-th** term decreases, indicating that using far-away but large-size domains in learning can still lead to low generalization bound on target domain. Conversely, the opposite also holds for these terms when conditions are reversed. As long as the consecutive domains are similar enough, the generalization error remains bounded. This verify our intuition that even distant domains help learn the target domain when a good connecting path exists. Note that this generalization bound acts as a worst-case scenario for prediction errors. In practice, it guarantees that actual errors remain below this upper bound, but it doesn't necessarily mean achieving the minimum prediction error itself. We focus on the optimization of this worst-case scenario in the following.

For comparison, we also analyze the expected error bounds for two baselines (detailed analysis is in Appendix C and D). First one is training with joint data from all source domains. The expected error of the trained classifier scales as the weighted Wasserstein-1 distance between each source and the target. When a domain has dominate number of samples and large enough Wasserstein distance with the target, the error on this domain will dominate the final trained classifier. The second baseline is training a model only on the closest domain. As in standard learning theorem, the risk decreases monotonically as the number of sample grows. In the case of the closest domain does not have contain enough samples, a trade-off between Wasserstein distance and the number of samples needs to be carefully considered and selected.

6 GRAPH ROUTING

Pivoting around the minimization of generalization error bound in Theorem 5.2 for the best worst-case scenario, we present our GFT trajectory selection strategies based on classical graph routing algorithms. As justified previously, a path in G represents a GFT trajectory. Theorem 5.2 thereby indicates that corresponding error bound of the GFT trajectory

can be represented by length of the path and sizes of source domain data sets on the path. Qualitatively, our objective is to route a path in G that minimizes the error bound. Because the number of paths in G is limited, we can exhaust every single path in G to get the optimal path that minimizes the bound. However, the number of paths in G grows larger than exponential functions and even factorial when G is near complete (Jokić & Van Mieghem, 2022), which is obviously multiplied in the computational complexity of obtaining the optimal path. Therefore, in this section, we aim to explore more efficient ways to get paths with acceptable loss of optimality.

By observation, the error bound in Theorem 5.2 increases w.r.t. weights of the path $\sum_{t=1}^{\kappa-1} \Delta_{t,t+1}$, which naturally derives the first principle of path search: minimize path weights. Another more intricate factor is the magnitude, i.e. the sum of source domain data sizes n_i on the path, which exists in the last three terms in Theorem 5.2. For a fixed κ, all the terms decreases[1] w.r.t. n_i. The bound decreases as adding new large sized sources into training, because $\mathcal{R}_\kappa^{\text{seq}}$ mostly dominates sum of magnitude terms by its relatively large constant. We then get the second path search principle: maximize path magnitudes. Following the two principles, we obtain the optimal path $\boldsymbol{\pi}^*$ by formulation

$$\boldsymbol{\pi}^* \approx \arg\max_{\boldsymbol{\pi} \in P^*} \operatorname{mag}(\boldsymbol{\pi}), \text{ where } P^* = \left\{ \arg\min_{\boldsymbol{\rho} \in P_G(S_i, T)} \Delta(\boldsymbol{\rho}),\ i \in [K] \right\}, \tag{4}$$

where $\Delta(\boldsymbol{\rho})$ is the sum of weights (i.e. Wasserstein distance) along any path $\boldsymbol{\rho}$. For each source dataset S_i, we first minimize the weights over all the paths from S_i to T in G, denoted by $P_G(S_i, T)$. Then we maximize magnitudes over all minimal weight paths, denoted by P^*, each of which corresponds each S_i as start source. The first stage from $P_G(S_i, T)$ to P^* is guided by graph routing. The second stage from P^* to approximately optimal path is evaluated on the magnitude metric we defined.

Repetitive Nearest Neighbor Search. Our first proposed graph routing strategy employs the repetitive nearest neighbor algorithm, which selects the closest unvisited neighbor from one vertex to another until it reaches dead end. We utilize nearest neighbor algorithm with T as the starting vertex and accumulate the path at each stop. Assuming we don't prune the original graph G in this strategy, by going K steps we exhaust all vertices and obtain an approximate length optimal path for each S_i by backtracking its cumulative path until T:

$$\boldsymbol{\rho}_{\text{nn}}(S_i) = \left\{ (T, f(T)), \left(f(T), f^2(T)\right), \ldots, \left(f^{K-1}(T), S_i\right) \right\}$$
$$\text{where } f(u) = \arg\min_{v,\,(u,v) \in E} W_p(u, v),\ f^K(T) = S_i.$$

Note that nearest neighbor graph routing cannot get the exact minimal length path for each S_i, but it applies a greedy strategy that guarantees every next stop on the path comes from its closest non-successor, which acts as a 1-gram safe move in GFT sense: fine-tuning on the source closest to the next target achieves the closest behaviour to it in one-move scope. It also guarantees maximum magnitudes, as the fact that it exhausts every source domain brings a Hamilton path as one of approximate optima.

Shortest Paths. Classical shortest path (SP) routing gets the exact minimal length paths P^* in Equation 4 by definition. For each source domain S_i, we apply Dijkstra's algorithm (Cormen et al., 2022) to calculate the SP from S_i to T in G, i.e. the path with minimal sum of W_p between every pair of adjacent domains on the path:

$$\boldsymbol{\rho}_{\text{sp}}(S_i) = \{(S_i, u_1), (u_1, u_2), \ldots, (u_{\kappa-1}, T)\}, \text{ where } \boldsymbol{u} = \arg\min_{\substack{\{u_1, \ldots, u_{\kappa-1}\} \subset V \\ (u_i, u_{i+1}) \in E \\ u_0 = S_i,\, u_{t+1} = T}} \sum_{i=0}^{\kappa-1} W_p(u_i, u_{i+1}).$$

This strategy gives exactly P^* and guarantees to get the optimal path length in the generalization bound, but it generally doesn't generate large magnitudes because shortest paths only contain a small portion of vertices in a general graph. In this case, we define the magnitude as the sum of data size of every domain on the path. Note that we apply edge weight thresholds to G for this strategy, as a compromise that enhances the magnitude of the path while making the path longer.

Minimum Spanning Tree. This strategy is based on minimum spanning tree (MST), which is the tree as a subgraph of G that makes all vertices in G connected while its edge subset has the minimal weight sum. We prune G to its most lightweight connected acyclic form whose edges are short and every pair of vertices have one and only one path in between. For domain graph G, we apply Kruskal's algorithm (Cormen et al., 2022) to calculate its MST. Then for each source domain S_i, we use the exactly one path from S_i to T to form the approximate P^*:

$$\boldsymbol{\rho}_{\text{mst}}(S_i) \in P_{\text{MST}(G)}(S_i, T), \text{ where } \forall u, v \in V,\ |P_{\text{MST}(G)}(u, v)| = 1.$$

[1] For the term with sequential Rademacher complexity $\mathcal{R}_\kappa^{\text{seq}}$, since $\mathcal{R}_\kappa^{\text{seq}}$ has the same order as $1/\sum n_i$, it also decreases w.r.t. n_i.

Table 1: Accuracy comparison on 5 target domains from the `MultiNLI` dataset, in mean ± std. Subscripts of average accuracies denote relative decreases to the best performance. Repeated experiments are conducted above identical set of seeds for training.

Method	Target Domain					Avg Acc.
	Fiction	Government	Telephone	Slate	Travel	
ALL SOURCES	76.62 ± 0.67	72.34 ± 1.57	71.94 ± 1.37	71.09 ± 1.12	72.47 ± 2.02	$72.89_{(\downarrow 4.7\%)}$
CLOSEST	74.97 ± 0.34	72.88 ± 1.19	72.24 ± 0.71	73.50 ± 1.28	71.36 ± 0.85	$72.99_{(\downarrow 4.6\%)}$
SEAL-SHAP	74.70 ± 1.62	75.39 ± 0.75	$\mathbf{74.63} \pm 2.05$	73.37 ± 0.69	75.70 ± 3.07	$74.75_{(\downarrow 2.3\%)}$
Xu et al. (2021)	$\mathbf{78.82} \pm 1.62$	75.29 ± 1.11	74.97 ± 0.59	75.47 ± 1.11	73.25 ± 2.37	$75.56_{(\downarrow 1.2\%)}$
TGFT	77.43 ± 1.78	$\mathbf{77.19} \pm 2.13$	72.89 ± 2.08	74.35 ± 1.59	74.68 ± 4.43	$75.30_{(\downarrow 1.6\%)}$
NNGFT	78.03 ± 2.34	76.95 ± 2.14	73.74 ± 2.19	$\mathbf{77.03} \pm 6.27$	$\mathbf{76.76} \pm 2.19$	$\mathbf{76.50}_{(0.0\%)}$
SPGFT	76.40 ± 1.31	73.91 ± 5.31	73.05 ± 1.69	71.00 ± 3.39	73.14 ± 2.85	$73.50_{(\downarrow 3.9\%)}$
MSTGFT	76.18 ± 4.39	73.91 ± 5.31	73.05 ± 1.69	71.00 ± 3.39	73.14 ± 2.85	$73.45_{(\downarrow 4.0\%)}$

This strategy employs a further trade-off with the shortest path strategy between path lengths and magnitudes in the generalization bound. The Cut Property of MST states that any newly added edge to an MST forms a cycle and has the maximum weight on the cycle. As a result, MST can omit potential edges from a shortest path and take alternative routes that involve smaller edges, enhancing its magnitudes.

The proposed strategies leverage the power of graph-based representations to facilitate the seamless knowledge transfer from a source domain to a target domain while accounting for their distributional differences and magnitudes.

7 EXPERIMENTAL SETUP

We evaluate our proposed GFT methods, focusing on sentiment analysis (SA) and Natural Language Inference (NLI) text classification tasks.

Multi-domain Datasets. For the SA task, we use the Amazon Review dataset (Blitzer et al., 2007; Liu et al., 2017), which contains product reviews from 20 domains, annotated with binary sentiment labels (positive or negative sentiment). In this experiment, we randomly select 8 domains as shown in Table 2 for simplicity of the experiment setting. The language used in the dataset is English and Spanish. Additionally, to understand the performances of different strategies under a more difficult scenario for this task, we manually select and experiment on 4 domains out of the 8 from Amazon Review that are more diverging: *books*, *music*, *electronics*, *grocery*, which have the greatest Wassserstein-1 distance between each pair. For the NLI task, we use multi-genre Natural Language Inference (Williams et al., 2018, `MultiNLI`), which contains a sentence pair of premise and hypothesis from 5 domains. The language in the dataset is English. Each sentence pair is annotated with entailment, neutral, or contradiction labels. We binarize the label into entailment or not, by following the procedure in Ma et al. (2019).

Implementation & Evaluation. The base model for the gradual fine-tuning experiments is a BERT-based model (Devlin et al., 2019). Our gradual fine-tuning implementation is built on top of the Huggingface framework (Wolf et al., 2020), and we use Geomloss (Feydy et al., 2019) to compute Wasserstein distances between domains in a dataset. For the evaluation metric, we use accuracy to compare performance between different methods on each task.

Baselines. We experiment with our gradual fine-tuning (GFT) methods, namely nearest-neighbor (NNGFT), shortest path (SPGFT), and minimum spanning tree (MSTGFT) graph routing. Additionally, we also conduct the aforementioned brute-force that exhausts every possible path in G and chooses the one that minimizes the theoretical generalization bound, named TGFT. We compare our GFT strategies to several baselines: (i) All sources 1-stage (ALL SOURCES): We use all source domains combined for training and evaluate it on the target domain. (ii) Closest source 1-stage (CLOSEST): We use the closest source domain to the target domain and evaluate it on the target domain. We determine the closest domain by Wasserstein distance. (iii) SEAL-SHAP: A state-of-the-art method Parvez & Chang (2021) that uses Shapley-based score to measure the usefulness of individual sources for transfer learning. (iv) Xu et al. (2021), which gradually fine tunes on mixtures of in- and out-domain data with descending amount of out-domain data.

Table 2: Accuracy comparison on 8 target domains from the multi-domain sentiment analysis dataset, in mean ± std. Subscripts of average accuracies denote relative decreases to the best performance. Repeated experiments are conducted above identical set of seeds for training.

Method	Target Domain								Avg Acc.
	Apparel	Baby	Dvd	Electronics	Beauty	Books	Grocery	Music	
ALL SOURCES	91.89 ±0.28	91.31 ±0.72	90.48 ±0.56	89.96 ±0.85	90.85 ±0.52	90.56 ±0.45	90.83 ±0.67	89.96 ±0.43	90.73 (↓1.1%)
CLOSEST	88.62 ±0.77	88.40 ±0.52	89.31 ±0.31	88.34 ±0.31	88.49 ±0.54	88.60 ±0.36	88.15 ±0.52	88.82 ±0.57	88.59 (↓3.4%)
SEAL-SHAP	**92.13** ±0.70	**93.13** ±0.13	**91.00** ±0.13	**91.68** ±0.25	**93.90** ±0.32	**93.23** ±0.15	89.43 ±0.45	89.12 ±0.07	**91.70** (0.0%)
Xu et al. (2021)	91.40 ±0.11	92.13 ±0.33	89.81 ±0.14	90.08 ±0.03	90.67 ±0.33	89.81 ±0.31	**91.56** ±0.62	89.54 ±0.53	90.63 (↓1.2%)
TGFT	91.93 ±0.6	88.22 ±0.54	90.18 ±0.62	89.43 ±0.49	90.51 ±0.92	90.41 ±0.8	89.98 ±0.48	**90.22** ±0.26	90.11 (↓1.7%)
NNGFT	91.95 ±0.6	90.68 ±0.4	90.31 ±0.8	90.10 ±0.72	90.19 ±0.53	90.21 ±0.36	90.37 ±0.65	89.95 ±0.66	90.47 (↓1.3%)
SPGFT	91.36 ±0.43	88.05 ±0.60	88.31 ±0.64	89.50 ±0.61	89.68 ±1.02	88.77 ±1.33	89.68 ±0.43	88.52 ±1.07	89.23 (↓2.7%)
MSTGFT	91.35 ±0.42	88.44 ±0.88	87.91 ±0.64	89.35 ±0.87	88.98 ±0.94	88.96 ±0.72	89.09 ±0.66	89.14 ±0.98	89.15 (↓2.8%)

Table 3: Accuracy comparison on 4 distant domains from the multi-domain sentiment analysis dataset, in mean ± std. Subscripts of average accuracies denote relative decreases to the best performance. Repeated experiments are conducted above identical set of seeds for training.

Method	Target Domain				Avg Acc.
	Books	Music	Electronics	Grocery	
ALL SOURCES	89.71 ± 0.31	88.83 ± 0.67	**87.75** ± 0.56	88.66 ± 0.53	88.73 (↓0.5%)
CLOSEST	**89.69** ± 0.41	88.98 ± 0.22	83.66 ± 0.78	88.62 ± 0.88	87.73 (↓1.6%)
SEAL-SHAP	84.85 ± 1.80	85.91 ± 1.52	88.18 ± 0.47	84.21 ± 1.50	85.79 (↓3.8%)
Xu et al. (2021)	88.87 ± 0.85	88.90 ± 0.37	87.56 ± 0.44	88.72 ± 1.52	88.51 (↓0.7%)
NNGFT	89.33 ± 0.53	**89.85** ± 0.32	87.65 ± 0.04	**89.85** ± 0.82	**89.17** (0.0%)

8 RESULTS & DISCUSSION

Performance on `MultiNLI`. As shown in Table 1, on overall average accuracy, NNGFT outperforms all the baselines including the state-of-the-art, SEAL-SHAP. On per-domain performance, TGFT and NNGFT outperform SEAL-SHAP on 3 and 4 target target domains, respectively. However, the results for SPGFT and MSTGFT are less positive compared to TGFT and NNGFT, this is possibly because SPGFT and MSTGFT produces shorter path consisting only 1-2 source domains, hence discarding the distant domains that can offer benefit for the performance in the target domain in the `MultiNLI` dataset. In 4 target domains, the results between TGFT and NNGFT are the same because the produced paths are identical from both methods. The fact that NNGFT surpasses CLOSEST on all target domains indicates that by using only the closest source domain to the target domain is not optimal for the `MultiNLI` dataset. Additionally, the fact that NNGFT is better than ALL SOURCES suggests that, although in terms of training data we use all source domains on both ALL SOURCES and NNGFT, gradual fine-tuning is evidently better than one-stage fine-tuning.

Performance on SA. SEAL-SHAP yields the best overall results as shown in Table 2. Per-domain wise, it outperforms all the methods in 6 out of 8 target domains. By overall average accuracy, all GFT variants can only outperform the CLOSEST baseline. TGFT obtains one better performance than SEAL-SHAP on the Music domain. Other routing

Table 4: Elapsed real time of each method on SA task, for pathfinding and training respectively. All runs are on the same computing instance with `NVIDIA Tesla V100` with cold start.

Method	Pathfinding	Training	Notes
SEAL-SHAP	$<$ 6 hours	$<$ 10 minutes	long pathfinding time
Xu et al. (2021)	$\approx$ 20 minutes		pathfinding is online in parallel with training
All GFT variants	$\approx$ 30 minutes	$<$ 10 minutes	only $\approx$ 5 minutes pathfinding without pseudo label generation

strategies, SPGFT and MSTGFT, are not necessary optimal but they are still comparable to SEAL-SHAP and other graph strategies because of their close performance to the best with low resource use of data and computation. From Table 3, with the 4 distant domains, SEAL-SHAP gets significantly hit by increased discrepancy among source domains. NNGFT outperforms all other baselines in 2 out of 4 domains and on average accuracy while stay comparable in the other 2 domains. It is also worth noting that, SEAL-SHAP is hugely more expensive in terms of computation. During experiments, SEAL-SHAP needs more than 5 hours to obtain the Shapley scores of the candidate source domains for each target domain, while GFT only around 30 minutes in average.[2] Last but not least, our approaches do not need target labels whereas SEAL-SHAP needs to access them in its scoring function.

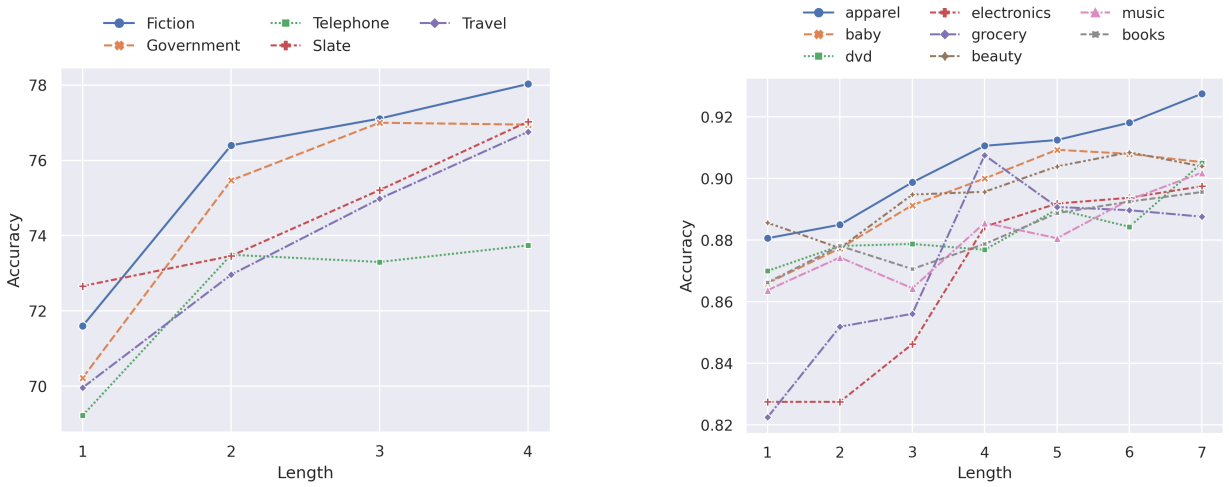
Nature of SA versus NLI. We analyze from another aspect why GFT performs not as significantly in the 8-domain SA experiment: SA is a relatively less complex task than NLI. In SA, detecting the sentiment polarity of some adjective keywords is empirically effective to determine the sentiment within a sentence (Hutto & Gilbert, 2014). Such approach is common in the context of SA, where having a broader lexicon can enhance overall performance. This correlation also explains why training on all sources has strong performance in SA. On the other hand, NLI is generally considered to be a harder problem in NLP due to its requirement for more advanced linguistic comprehension and reasoning skills. NLI is essentially a textual entailment task which requires reasoning and world knowledge (Bowman et al., 2015) to determine the relation between a premise and hypothesis. It is not sufficient for the NLI model to only rely on lexicon (word) encountered in a sentence, the model needs to capture the "meaning" of each sentence to then determine the relation between premise and hypothesis text fragments. We use the preceding discussion to further support GFT's capability of solving more complex tasks based on the experimental results on those tasks.

Observation on Domain Distance. Based on the results from both tasks, although NNGFT performs the best on the NLI task, its performance on the SA task is less effective although it's still comparable with ALL SOURCES and CLOSEST baselines. Observing the pairwise of the domain distance, we notice that the distance between domains in the Amazon dataset is relatively smaller compared to domains in the `MultiNLI` dataset. Based on this observation, we hypothesize that NNGFT is more effective when the domains has more diverging distance. NNGFT always exhausts all source domains, but in an efficient way that it continuously seek the closest domain at each search step to try to minimize total distance travelled. As the domains in a dataset become more distant, other methods that use all/most source domains e.g. all-source, SEAL-SHAP experience a decline in their performances. This is because growing divergence between domains cause distortion in methods that merge domains during training, and methods that choose domains in pointwise manners overlook the increasing cross-domain distance and the path length, resulting in significant prediction error bound increase, according to Theorem 5.2. However, GFT approaches follow a specific discipline that they actively controls the distance between domains which mitigates such performance regression. Experiment results reported in Table 3 also verifies this hypothesis. When we manually choose domains that are farther apart within the same dataset, the effectiveness of our proposed algorithm becomes more apparent.

Ablation Study on Path Length. We also perform an ablation study by experimenting with different path lengths of GFT algorithms. Given a particular sequence of fine-tuning of a target domain produced by GFT we try to examine the behaviour of the model when we exclude a number of source domains from the sequence. For example, if the GFT sequence of *travel* target domain in `MultiNLI` is: *slate* $\rightarrow$ *telephone* $\rightarrow$ *government* $\rightarrow$ *fiction* (path length of 4). To produce path length of 3, we simply remove the furthest source domain, *slate*, and measure the performance. We repeat the same step for different path length. Figure 2 shows the accuracy for NNGFT with different path lengths on `MultiNLI` and SA. We observe that as we include more source domains in NNGFT, the trend of accuracy on most target domains are also increasing for both datasets.

Notes on TGFT. In the two experiments conducted, we notice that TGFT generally performs less effectively than NNGFT. It is true that generalization bound value for TGFT is theoretically the lowest among all GFT approaches, as it

[2] See Table 4 for a more detailed wall clock time report that shows the computational efficiency of GFT approaches.



(a) MultiNLI (b) Sentiment Analysis

Figure 2: Accuracy ablation on different path length on two datasets. The x-axis indicates the path length of NNGFT, i.e. number of source domains that is included in NNGFT. The y-axis indicates the accuracy on a particular path length.

explores all possible paths and selects the one that minimizes the bound. However, as discussed above, TGFT only guarantees the *minimax* prediction error, but not the *minimum* prediction error. This observation indicates that optimizing the worst-case scenario does not always yield the most practical results in GFT. It inspires deeper exploration into the factors influencing what's under the upper bound, e.g. its distribution, to approach more optimal pathfinding strategies.

Notes on SPGFT and MSTGFT. For the other two graph routing strategies, i.e. SPGFT and MSTGFT, we also highlight that they make significant trade-offs between resource consumption which reflects training computational cost, and slight performance reduction, regardless of low pathfinding cost already. Unlike NNGFT which exhaust all the domains on the fine tuning path, SPGFT and MSTGFT only select small portion of the domains and follow short paths that link those domains together, indicating low resource consumption and faster training, though they lead to a slight performance decrease. All GFT strategies work significantly fast in pathfinding, while remaining the flexibility between training costs, as determined by the identified paths connecting selected source domains, and model performance.

9 CONCLUSION

We conduct theoretical and experimental studies on gradual fine-tuning (GFT) in multi-source unsupervised domain adaptation setting. We show that theoretically, using all source domains through GFT minimizes the generalization error. Our experiment results show that even *without* source domains selection, the adapted model from GFT outperforms state-of-the-art method in Natural Language Inference (NLI) task and achieve comparable performance in the sentiment analysis (SA) task. We observe that (i) GFT is more effective when the Wasserstein distance between source domains and target are more diverge. Including distant source domain through gradual fine-tuning can improve the adapted model on the intermediate domains which is beneficial for the final target domain eventually. (ii) Path optimality for GFT is still an open question as our graph routing strategies are focused on mitigating worst-case scenario, and are only close to but not strictly optimal. (iii) Current graph routing strategies can hardly scale large graphs constructed by too many source domains because of computational complexity quadratic increase. We believe that our findings can be applied to more complex NLP tasks in the context of multi-source domain adaptation.

ACKNOWLEDGMENTS

We would like to thank Gerrit van den Burg, Jean Baptiste Faddoul, Pavel Tyletski, Nithish Kannen, Wei Liu and the anonymous reviewers for their valuable and constructive feedback. We also thank Murat Sensoy and Abhishek Tripathi for directional advice.

REFERENCES

John Blitzer, Mark Dredze, and Fernando Pereira. Biographies, Bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pp. 440–447, Prague, Czech Republic, June 2007. Association for Computational Linguistics. URL https://aclanthology.org/P07-1056.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. A large annotated corpus for learning natural language inference. In Lluís Màrquez, Chris Callison-Burch, and Jian Su (eds.), *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 632–642, Lisbon, Portugal, September 2015. Association for Computational Linguistics. doi: 10.18653/v1/D15-1075. URL https://aclanthology.org/D15-1075.

Hong-You Chen and Wei-Lun Chao. Gradual domain adaptation without indexed intermediate domains. In M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (eds.), *Advances in Neural Information Processing Systems*, volume 34, pp. 8201–8214. Curran Associates, Inc., 2021. URL https://proceedings.neurips.cc/paper_files/paper/2021/file/45017f6511f91be700fda3d118034994-Paper.pdf.

Lénaïc Chizat, Pierre Roussillon, Flavien Léger, François-Xavier Vialard, and Gabriel Peyré. Faster wasserstein distance estimation with the sinkhorn divergence. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 2257–2269. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/17f98ddf040204eda0af36a108cbdea4-Paper.pdf.

Thomas H Cormen, Charles E Leiserson, Ronald L Rivest, and Clifford Stein. *Introduction to algorithms*. MIT press, 2022.

Nicolas Courty, Rémi Flamary, Amaury Habrard, and Alain Rakotomamonjy. Joint distribution optimal transportation for domain adaptation. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/0070d23b06b1486a538c0eaa45dd167a-Paper.pdf.

Koby Crammer, Michael Kearns, and Jennifer Wortman. Learning from multiple sources. *Journal of Machine Learning Research*, 9(57):1757–1774, 2008. URL http://jmlr.org/papers/v9/crammer08a.html.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL https://aclanthology.org/N19-1423.

Jean Feydy, Thibault Séjourné, François-Xavier Vialard, Shun-ichi Amari, Alain Trouve, and Gabriel Peyré. Interpolating between optimal transport and mmd using sinkhorn divergences. In Kamalika Chaudhuri and Masashi Sugiyama (eds.), *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, volume 89 of *Proceedings of Machine Learning Research*, pp. 2681–2690. PMLR, 16–18 Apr 2019. URL https://proceedings.mlr.press/v89/feydy19a.html.

Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. In Francis Bach and David Blei (eds.), *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 1180–1189, Lille, France, 07–09 Jul 2015. PMLR. URL https://proceedings.mlr.press/v37/ganin15.html.

Jiayuan Huang, Arthur Gretton, Karsten Borgwardt, Bernhard Schölkopf, and Alex Smola. Correcting sample selection bias by unlabeled data. In B. Schölkopf, J. Platt, and T. Hoffman (eds.), *Advances in Neural Information Processing Systems*, volume 19. MIT Press, 2006. URL https://proceedings.neurips.cc/paper/2006/file/a2186aa7c086b46ad4e8bf81e2a3a19b-Paper.pdf.

Clayton Hutto and Eric Gilbert. Vader: A parsimonious rule-based model for sentiment analysis of social media text. *Proceedings of the International AAAI Conference on Web and Social Media*, 8(1):216–225, May 2014. doi: 10.1609/icwsm.v8i1.14550. URL https://ojs.aaai.org/index.php/ICWSM/article/view/14550.

Ivan Jokić and Piet Van Mieghem. Number of paths in a graph. *arXiv preprint arXiv:2209.08840*, 2022.

S. Kullback and R. A. Leibler. On Information and Sufficiency. *The Annals of Mathematical Statistics*, 22(1):79 – 86, 1951. doi: 10.1214/aoms/1177729694. URL https://doi.org/10.1214/aoms/1177729694.

Ananya Kumar, Tengyu Ma, and Percy Liang. Understanding self-training for gradual domain adaptation. In Hal Daumé III and Aarti Singh (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 5468–5479. PMLR, 13–18 Jul 2020. URL https://proceedings.mlr.press/v119/kumar20c.html.

Vitaly Kuznetsov and Mehryar Mohri. Discrepancy-based theory and algorithms for forecasting non-stationary time series. *Ann. Math. Artif. Intell.*, 88(4):367–399, 2020. doi: 10.1007/s10472-019-09683-1. URL https://doi.org/10.1007/s10472-019-09683-1.

Jingjing Li, Erpeng Chen, Zhengming Ding, Lei Zhu, Ke Lu, and Heng Tao Shen. Maximum density divergence for domain adaptation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(11):3918–3930, 2021. doi: 10.1109/TPAMI.2020.2991050.

Jianhua Lin. Divergence measures based on the shannon entropy. *IEEE Transactions on Information Theory*, 37(1): 145–151, 1991. doi: 10.1109/18.61115.

Miaofeng Liu, Yan Song, Hongbin Zou, and Tong Zhang. Reinforced training data selection for domain adaptation. In Anna Korhonen, David Traum, and Lluís Màrquez (eds.), *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 1957–1968, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1189. URL https://aclanthology.org/P19-1189.

Pengfei Liu, Xipeng Qiu, and Xuanjing Huang. Adversarial multi-task learning for text classification. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1–10, Vancouver, Canada, July 2017. Association for Computational Linguistics. doi: 10.18653/v1/P17-1001. URL https://aclanthology.org/P17-1001.

Xiaofei Ma, Peng Xu, Zhiguo Wang, Ramesh Nallapati, and Bing Xiang. Domain adaptation with BERT-based domain classification and data selection. In *Proceedings of the 2nd Workshop on Deep Learning Approaches for Low-Resource NLP (DeepLo 2019)*, pp. 76–83, Hong Kong, China, November 2019. Association for Computational Linguistics. doi: 10.18653/v1/D19-6109. URL https://aclanthology.org/D19-6109.

Massimiliano Mancini, Lorenzo Porzi, Samuel Rota Bulò, Barbara Caputo, and Elisa Ricci. Boosting domain adaptation by discovering latent domains. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3771–3780, 2018. doi: 10.1109/CVPR.2018.00397.

Yishay Mansour, Mehryar Mohri, and Afshin Rostamizadeh. Domain adaptation with multiple sources. In D. Koller, D. Schuurmans, Y. Bengio, and L. Bottou (eds.), *Advances in Neural Information Processing Systems*, volume 21. Curran Associates, Inc., 2008. URL https://proceedings.neurips.cc/paper/2008/file/0e65972dce68dad4d52d063967f0a705-Paper.pdf.

Yishay Mansour, Mehryar Mohri, and Afshin Rostamizadeh. Multiple source adaptation and the rényi divergence. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, UAI '09, pp. 367–374, Arlington, Virginia, USA, 2009. AUAI Press. ISBN 9780974903958.

Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359, 2010. doi: 10.1109/TKDE.2009.191.

Md Rizwan Parvez and Kai-Wei Chang. Evaluating the values of sources in transfer learning. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 5084–5116, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.402. URL https://aclanthology.org/2021.naacl-main.402.

Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1406–1415, 2019. doi: 10.1109/ICCV.2019.00149.

Abhinav Ramesh Kashyap, Devamanyu Hazarika, Min-Yen Kan, and Roger Zimmermann. Domain divergences: A survey and empirical analysis. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 1830–1849, Online, June 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.naacl-main.147. URL https://aclanthology.org/2021.naacl-main.147.

Alan Ramponi and Barbara Plank. Neural unsupervised domain adaptation in nlp—a survey. In *Proceedings of the 28th International Conference on Computational Linguistics*, pp. 6838–6855, 2020.

Alfréd Rényi. On measures of entropy and information. In *Proceedings of the fourth Berkeley symposium on mathematical statistics and probability, volume 1: contributions to the theory of statistics*, volume 4, pp. 547–562. University of California Press, 1961.

Dmitry B Rokhlin. Asymptotic sequential rademacher complexity of a finite function class. *Archiv der Mathematik*, 108:325–335, 2017.

Sebastian Ruder and Barbara Plank. Learning to select data for transfer learning with Bayesian optimization. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 372–382, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. doi: 10.18653/v1/D17-1038. URL https://aclanthology.org/D17-1038.

Jian Shen, Yanru Qu, Weinan Zhang, and Yong Yu. Wasserstein distance guided representation learning for domain adaptation. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence and Thirtieth Innovative Applications of Artificial Intelligence Conference and Eighth AAAI Symposium on Educational Advances in Artificial Intelligence*, AAAI'18/IAAI'18/EAAI'18. AAAI Press, 2018. ISBN 978-1-57735-800-8.

Cédric Villani. *The Wasserstein distances*, pp. 93–111. Springer Berlin Heidelberg, Berlin, Heidelberg, 2009. ISBN 978-3-540-71050-9. doi: 10.1007/978-3-540-71050-9_6. URL https://doi.org/10.1007/978-3-540-71050-9_6.

Haoxiang Wang, Bo Li, and Han Zhao. Understanding gradual domain adaptation: Improved analysis, optimal path and beyond. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 22784–22801. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/wang22n.html.

Adina Williams, Nikita Nangia, and Samuel Bowman. A broad-coverage challenge corpus for sentence understanding through inference. In Marilyn Walker, Heng Ji, and Amanda Stent (eds.), *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pp. 1112–1122, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/N18-1101. URL https://aclanthology.org/N18-1101.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. Transformers: State-of-the-art natural language processing. In Qun Liu and David Schlangen (eds.), *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 38–45, Online, October 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-demos.6. URL https://aclanthology.org/2020.emnlp-demos.6.

Haoran Xu, Seth Ebner, Mahsa Yarmohammadi, Aaron Steven White, Benjamin Van Durme, and Kenton Murray. Gradual fine-tuning for low-resource domain adaptation. In Eyal Ben-David, Shay Cohen, Ryan McDonald, Barbara Plank, Roi Reichart, Guy Rotman, and Yftah Ziser (eds.), *Proceedings of the Second Workshop on Domain Adaptation for NLP*, pp. 214–221, Kyiv, Ukraine, April 2021. Association for Computational Linguistics. URL https://aclanthology.org/2021.adaptnlp-1.22.

Han Zhao, Shanghang Zhang, Guanhang Wu, José M. F. Moura, Joao P Costeira, and Geoffrey J Gordon. Adversarial multiple source domain adaptation. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc., 2018. URL https://proceedings.neurips.cc/paper_files/paper/2018/file/717d8b3d60d9eea997b35b02b6a4e867-Paper.pdf.

Han Zhao, Remi Tachet des Combes, Kun Zhang, and Geoffrey J. Gordon. On learning invariant representations for domain adaptation. In Kamalika Chaudhuri and Ruslan Salakhutdinov (eds.), *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pp. 7523–7532. PMLR, 2019. URL http://proceedings.mlr.press/v97/zhao19a.html.

A Proofs for Theoretical Analysis

Let's first recall the following general theorem.

Lemma A.1. *Let D be a joint distribution over $\mathcal{X} \times \mathcal{Y}$ and l be a B-bounded loss function that is L-Lipschitz in the 2-norm in the first argument. For a given function space $\mathcal{F}$ and $f \in \mathcal{F}$, let $f^* = \arg\min_{f\in\mathcal{F}} \mathrm{E}_C[l(x, y)]$ and $\hat{f} = \arg\min_{f\in\mathcal{F}} \frac{1}{n}\sum_{i=1}^{n} l(f(x_i), y_i)$ be the empirical and population loss minimizers. Then for any $\delta > 0$, with probability at least $1 - \delta$,*

$$l(\hat{f}) - l(f^*) \leq 2\sqrt{2}L\mathcal{R}(\mathcal{F}) + 2B\sqrt{\frac{\log(1/\delta)}{2m}}.$$

The next lemma shows the error difference over shifted domains.

Lemma A.2. *We further assume that the classifier in $\mathcal{F}$ is R-Lipschitz continuous, then*

$$|l(f, D_1) - l(f, D_2)| \leq L\sqrt{R^2 + 1}W_p(D_1, D_2).$$

Lemma A.2 provides a model independent bound on the difference of errors for a classifier under distribution shift. By utilizing this bound, we can bound the difference of errors for the classifies generated by GFT algorithm under the distribution shift.

$$\begin{aligned}
\epsilon_2(\hat{h}_2) - \epsilon_1(\hat{h}_1) &\leq \epsilon_2(\hat{h}_2) - \epsilon_2(\hat{h}_1) + L\sqrt{R^2 + 1}W_p(D_1, D_2) \\
&\leq \hat{\epsilon}_2(\hat{h}_2) - \hat{\epsilon}_2(\hat{h}_1) + 4\sqrt{2}L\mathcal{R}_{n_2}(\mathcal{F}) + 4B\sqrt{\tfrac{\log 1/\delta}{2n_2}} + L\sqrt{R^2 + 1}W_p(D_1, D_2) \\
&\leq 4\sqrt{2}L\mathcal{R}_{n_2}(\mathcal{F}) + 4B\sqrt{\tfrac{\log 1/\delta}{2n_2}} + L\sqrt{R^2 + 1}W_p(D_1, D_2)
\end{aligned}$$

where the third inequality hold as $\hat{h}_2$ is the minimizer of the empirical loss ϵ_2. By iteratively applying this result, we have for any $t \in \{1, ..., \kappa\}$

$$\begin{aligned}
\epsilon_t(\hat{h}_\kappa) - \epsilon_1(\hat{h}_1) &\leq \epsilon_\kappa(\hat{h}_\kappa) - \epsilon_1(\hat{h}_1) + L\sqrt{R^2 + 1}W_p(D_t, D_\kappa) \\
&\leq \textstyle\sum_{i=1}^{\kappa-1}\left(\epsilon_{i+1}(\hat{h}_{i+1}) - \epsilon_i(\hat{h}_i)\right) + L\sqrt{R^2 + 1}\sum_{i=t}^{\kappa-1} W_p(D_i, D_{i+1}) \\
&\leq \textstyle\sum_{i=1}^{\kappa-1}\left[4\sqrt{2}L\mathcal{R}_{n_{i+1}}(\mathcal{F}) + 4B\sqrt{\tfrac{\log(1/\delta)}{2n_{i+1}}} + L\sqrt{R^2 + 1}W_p(D_i, D_{i+1})\right] + L\sqrt{R^2 + 1}W_p(D_t, D_\kappa).
\end{aligned}$$

Summarize over $t = 2, ..., \kappa$, we have

$$\begin{aligned}
&\textstyle\sum_{t=2}^{\kappa} \epsilon_t(\hat{h}_\kappa) - (\kappa - 1)\epsilon_1(\hat{h}_1) \\
\leq\ & 4\sqrt{2}L(\kappa - 1)\textstyle\sum_{i=1}^{\kappa-1} \mathcal{R}_{n_{i+1}}(\mathcal{F}) + 4B(\kappa - 1)\sum_{i=1}^{\kappa-1}\sqrt{\tfrac{\log(1/\delta)}{2n_{i+1}}} \\
&+L\sqrt{R^2 + 1}(\kappa - 1)\textstyle\sum_{i=1}^{\kappa-1} W_p(D_i, D_{i+1}) + L\sqrt{R^2 + 1}\sum_{t=2}^{\kappa} W_p(D_t, D_\kappa)
\end{aligned}$$

Now, we present proofs for results in the theoretical analysis section.

Proof. In this proof, we follow the same line as Wang et al. (2022). By applying Corollary 2 of Kuznetsov & Mohri (2020), we can bound the population loss of the classifier $\hat{h}_K$ from the final stage of GFT in the target domain D_T as

$$\begin{aligned}
\epsilon_T(\hat{h}_\kappa) \leq\ & \textstyle\sum_{t=1}^{\kappa}\sum_{i=1}^{n_t} q_t^i \epsilon_t(\hat{h}_\kappa) + \mathrm{disc}(\mathbf{q}_\kappa) + \|\mathbf{q}_\kappa\|_2 \\
&+6B\sqrt{4\pi \log \textstyle\sum_{t=1}^{\kappa} n_t}\mathcal{R}_\kappa^{seq}(\mathcal{H}) + B\|\mathbf{q}_\kappa\|_2\sqrt{8\log 1/\delta},
\end{aligned}$$

where $\mathcal{R}_\kappa^{\mathrm{seq}}$ is the sequential Rademacher complexity of the hypothesis space $\mathcal{H}$ with loss function $\mathcal{L}$.

By setting the optimal weights for discrepancy measurement as $\mathbf{q}_\kappa = \left(\frac{1}{n_1\kappa}, ...\frac{1}{n_1\kappa}, ..., \frac{1}{n_\kappa\kappa}, ..., \frac{1}{n_\kappa\kappa}\right)$, we can bound the L^2 norm of $\mathbf{q}_\kappa$ as

$$\|\mathbf{q}_\kappa\|_2 \leq \frac{1}{\kappa}\sqrt{\frac{1}{n_1} + \frac{1}{n_2} + ... + \frac{1}{n_\kappa}}.$$

Then by applying the same weights, we can bound the discrepancy measurement as

$$\begin{aligned}
\mathrm{disc}(\mathbf{q}_\kappa) &= \sup_{h\in\mathcal{H}}\left(\epsilon_\kappa(h) - \textstyle\sum_{t=1}^{\kappa}\sum_{i=1}^{n_t} q_i^t \epsilon_t(h)\right) \\
&\leq \sup_{h\in\mathcal{H}}\left(\textstyle\sum_{t=1}^{\kappa}\sum_{i=1}^{n_t} q_i^t |\epsilon_\kappa(h) - \epsilon_t(h)|\right) \\
&\leq \tfrac{L}{\kappa}\sqrt{R^2 + 1}\textstyle\sum_{t=1}^{\kappa-1} W_p(D_t, D_\kappa)
\end{aligned}$$

Finally, we provide the bound for the first term as

$$\begin{aligned}\frac{1}{\kappa}\sum_{t=1}^{\kappa}\epsilon_t(\hat{h}_t) \quad \leq \quad & \epsilon_1(\hat{h}_1)+\frac{4\sqrt{2L}(\kappa-1)}{\kappa}\sum_{t=1}^{\kappa-1}\mathcal{R}_{n_{t+1}}+\frac{4B(\kappa-1)}{\kappa}\sum_{t=1}^{\kappa-1}\sqrt{\frac{\log(1/\delta)}{2n_{t+1}}} \\ & +\frac{L\sqrt{R^2+1}}{\kappa}\sum_{t=1}^{\kappa-1}W_p(D_t,D_{t+1})+\frac{L\sqrt{R^2+1}}{\kappa}\sum_{t=1}^{\kappa-1}W_p(D_t,D_\kappa)\end{aligned}$$

Then, we can rewrite the above result as

$$\begin{aligned}\epsilon_\kappa(\hat{h}_\kappa) \quad \leq \quad & \epsilon_1(\hat{h}_1)+\frac{\kappa-1}{\kappa}L\sqrt{R^2+1}\Delta+(\kappa-1)L\sqrt{R^2+1}\Delta \\ & +\frac{4\sqrt{2}L(\kappa-1)}{\kappa}\sum_{t=1}^{\kappa-1}\mathcal{R}_{n_{t+1}}+\frac{4B(\kappa-1)}{\kappa}\sum_{t=1}^{\kappa-1}\sqrt{\frac{\log(1/\delta)}{2n_{t+1}}} \\ & +\frac{1}{\kappa}\sqrt{\sum_{t=1}^{\kappa}\frac{1}{n_t}}+6B\sqrt{4\pi\log\sum_{t=1}^{\kappa}n_t}\mathcal{R}_\kappa^{seq}(\mathcal{H})+\frac{B}{\kappa}\sqrt{8\log(1/\delta)\sum_{t=1}^{\kappa}\frac{1}{n_t}}\end{aligned}$$

Furthermore, we expand $\epsilon_1\hat{h}_1$ and obtain the following result

$$\begin{aligned}\epsilon_\kappa(\hat{h}_\kappa) \quad \leq \quad & \hat{\epsilon}_1(\hat{h}_1)+\frac{2\sqrt{2}BL}{\sqrt{n_1}}+2B\sqrt{\frac{\log(1/\delta)}{2n_1}}+(\kappa-\tfrac{1}{\kappa})L\sqrt{R^2+1}\Delta \\ & +\frac{4\sqrt{2}LB(\kappa-1)}{\kappa}\sum_{t=1}^{\kappa-1}\frac{1}{\sqrt{n_{t+1}}}+\frac{4B(\kappa-1)}{\kappa}\sum_{t=1}^{\kappa-1}\sqrt{\frac{\log(1/\delta)}{2n_{t+1}}} \\ & +\frac{1}{\kappa}\sqrt{\sum_{t=1}^{\kappa}\frac{1}{n_t}}+6B\sqrt{4\pi\log\sum_{t=1}^{\kappa}n_t}\mathcal{R}_\kappa^{seq}(\mathcal{H})+\frac{B}{\kappa}\sqrt{8\log(1/\delta)\sum_{t=1}^{\kappa}\frac{1}{n_t}}\end{aligned}$$

which concludes the proof. □

For comparison, we also provide the expected error bounds for two baselines. The first one is training with the joint of data from all source domains. The second baseline is training a model only on the closest domain.

B Simulation Results

We present the results of the 2-source initial simulation experiments for our GFT framework. We compared three training strategies: 1) We train two linear classifiers on Source 1 and Source 2, separately. 2) We train a single classifier on joint of Source 1 and Source 2. 3) We apply GFT following path "Source 2 → Source 1". The result is shown in Figure 3. GFT achieves the highest test accuracy among the three strategies. The intuition is the classifier in training strategy 3 achieve a better result by following the guidance of our graph routing GFT.

C Baseline: All Sources

One important baseline is the risk of the classifier trained with data from all sources. Let's denote the classifier that minimize the empirical loss over all samples as

$$\hat{h}=\arg\min_{h\in\mathcal{H}}\frac{1}{n_{1:K}}\sum_{t=1}^{K}\sum_{(x,y)\in S_t}\mathcal{L}(h(x),y).$$

By applying the same theory from Kuznetsov & Mohri (2020), we are able to write the upper bound of the risk of $\hat{h}$ on the target domain.

Lemma C.1. *Suppose at each time step, a sample is i.i.d. drawn from the entire training data-set. With probability* $1-\delta$*, the expected error of classifier* $\hat{h}$ *is bounded as*

$$\begin{aligned}\epsilon_T(\hat{h}) \quad \leq \quad & L\sqrt{R^2+1}\sum_{t=1}^{K}\left(\frac{n_t}{n_{1:K}}\right)W_P(D_t,D_T) \\ & +\sum_{t=1}^{K}\left(\frac{n_t}{n_{1:K}}\right)\hat{\epsilon}_t(\hat{h})+\sum_{t=1}^{K}\frac{\sqrt{n_t}B}{n_{1:K}} \\ & +\sum_{t=1}^{K}\frac{\log(1/\delta)\sqrt{n_t}}{n_{1:K}}\end{aligned}$$

The expected error of $\hat{h}$ scales as the weighted Wasserstein-1 distance between each source and the target. When a domain has dominate number of sample n_i and large enough $\Delta_{i,T}$, the error on this domain will dominate the final trained classifier.

D BASELINE: CLOSEST SOURCE

As in most domain adaption algorithms, training on the closest source has been shown to achieve good performance in many real applications. But the drawback of limited number of samples always exists when the closest source does not contain enough number of labeled training samples. Here, we denote the classifier trained on the closest domain c as $\hat{h}_c = \arg\min_{h \in \mathcal{H}} \frac{1}{n_c} \sum_{i=1}^{n_c} \mathcal{L}(h(x), y)$.

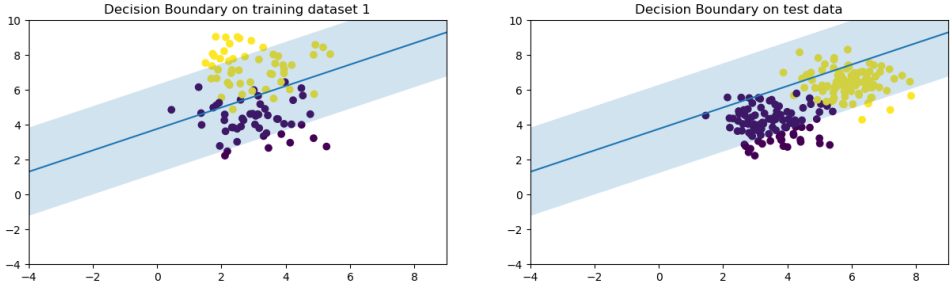
Lemma D.1. *With probability $1 - \delta$, the expected error of the learned classifier $\hat{h}_c$ satisfies*

$$\epsilon_T(\hat{h}_c) \leq L\sqrt{R^2 + 1} W_p(D_c, D_T) \\ + \hat{\epsilon}_c(\hat{h}_c) + \frac{B}{\sqrt{n_c}} + \frac{\log(1/\delta)}{\sqrt{n_c}}$$

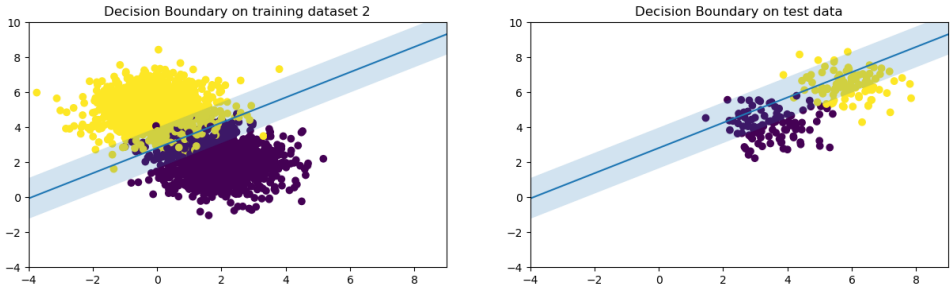
As in standard learning theorem, the risk decreases monotonically as the number of sample n_c grows. In the case of n_c does not have contain enough samples, a trade-off between $\Delta_{c,T}$ and n_c need to be carefully considered and selected.

E PAIRWISE DISTANCE MATRIX

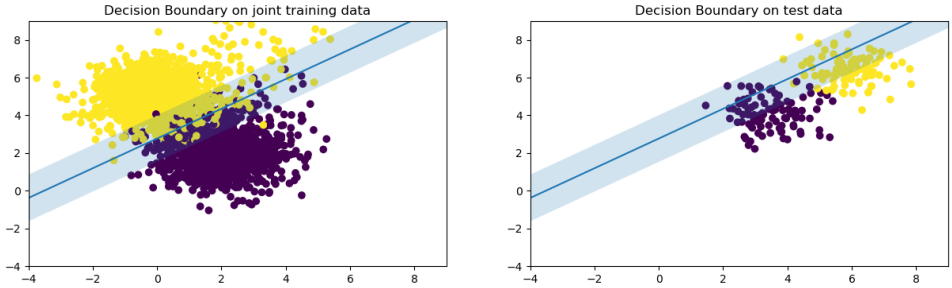
We include the W_1 adjacency matrix of `MultiNLI` and Amazon (for sentiment analysis) domains in Figure 4 and 5 respectively.



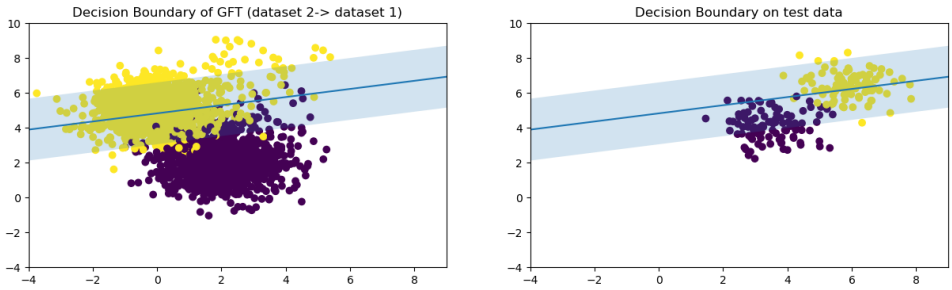
(a) Classifier trained on dataset 1



(b) Classifier trained on dataset 2



(c) Classifier trained on joint of datasets



(d) Classfier trained with GFT

Figure 3: Two datasets drawn from different distributions are available as training data. The test data from target domain has higher discrepancy to dataset 1 than dataset 2. An linear model trained on dataset 1 achieves 0.555 accuracy on test data as shown in subfigure (a). The same model trained only on dataset 2 achieves 0.54 accuracy on test data. Although dataset 1 is more similar to the test dataset, it still achieves performance since the number of samples is very limited. By jointing the two sources, the model's accuracy is 0.53 as dataset 2 has much more samples than 1. In the last subfigure, we applied GFT algorithm by following the order Source $2 \rightarrow$ Source 1. The modeled achieves 0.805 accuracy.

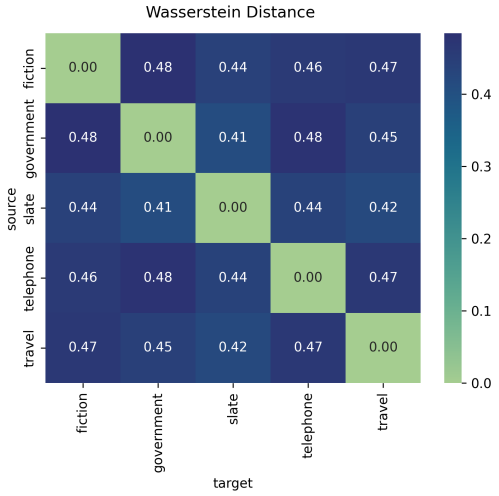


Figure 4: The matrix of pairwise Wasserstein-1 distance between domains in the `MultiNLI` dataset.

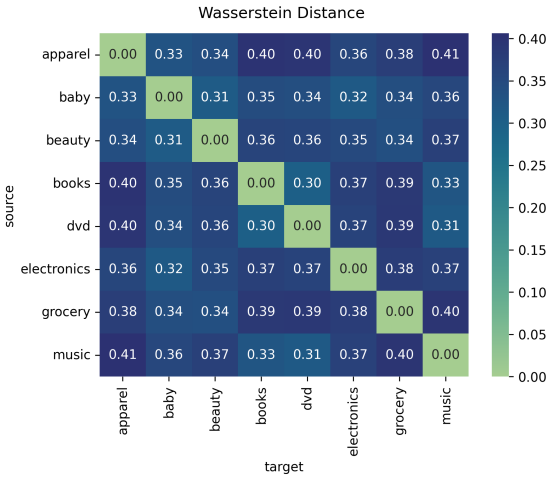


Figure 5: The matrix of pairwise Wasserstein-1 distance between domains in the Amazon dataset.