An Optimal Transport-based Latent Mixer for Robust Multi-modal Learning

Fengjiao Gong[1], **Angxiao Yue**[1], **Hongteng Xu**[1,2*]

[1]Gaoling School of Artifical Intelligence, Renmin University of China, Beijing, China
[2]Beijing Key Laboratory of Big Data Management and Analysis Methods, Beijing, China
gongfengjiao2021@ruc.edu.cn, angxiaoyue@ruc.edu.cn, hongtengxu@ruc.edu.cn

Abstract

Multi-modal learning aims to learn predictive models based on the data from different modalities. However, due to the requirement of data security and privacy protection, real-world multi-modal data are often scattered to different agents and cannot be shared across the agents, which limits the application of existing multi-modal learning methods. To achieve robust multi-modal learning in such a challenging scenario, we propose a novel optimal transport-based mixer (OTM), which works as an effective latent code alignment and augmentation method for unaligned and distributed multi-modal data. In particular, we train a Wasserstein autoencoder (WAE) for each agent, which encodes its single modal samples in a latent space. Through a central server, the proposed OTM computes a stochastic fused Gromov-Wasserstein barycenter (FGWB) to mix different modalities' latent codes, so that each agent applies the barycenter to reconstruct its samples. This method neither requires well-aligned multi-modal data nor assumes the data to share the same latent distribution, and each agent can learn a specific model based on multi-modal data while achieving inference based on its local modality. Experiments on multi-modal clustering and classification demonstrate that the models learned with the OTM method outperform the corresponding baselines.

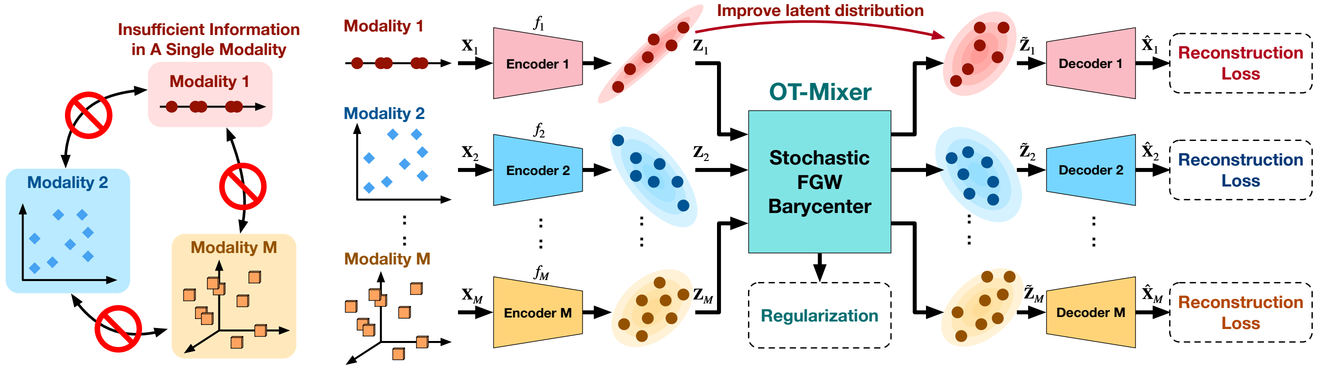
Introduction

Real-world data, such as healthcare records and multimedia signals, are multimodal in general, consisting of multiple modalities describing different aspects of the same object (Sharma and Giannakos 2020). Multi-modal learning aims to obtain a comprehensive representation across all modalities within the data, uncovering and fusing the high-level semantics lying in different modalities (Baltrušaitis, Ahuja, and Morency 2018) for downstream tasks. This learning task is important for many applications, e.g., diagnosing diseases based on heterogeneous clinical records and test results (Zhang et al. 2019), fraud detection based on various social behaviors (Abilov et al. 2021), and so on.

The common approach to represent all modalities simultaneously is to fuse them together to get an unified representation in a latent space through supervised or unsupervised learning (de Cheveigné et al. 2019; Wang et al. 2015; Guo et al. 2014). However, this learning paradigm requires well-aligned multi-modal data and assumes different modalities to share the same latent space, which are questionable even unavailable in practical applications, especially in those distributed scenarios. In particular, as shown in Figure 1(a), real-world multi-modal data are often scattered to different local agents, and each agent can only access the data in a single modality. Due to privacy protection and data security, sharing data directly across different agents is forbidden in many applications. What is worse, for some agents, the data associated with its modality may be insufficient for downstream tasks because the number of the data can be limited and the features can be not informative enough for representation learning. Take healthcare records as an example. Different hospitals might have different modalities of patients data in various formats, which are not supposed to be shared directly due to legal regulations, private information leakage risks, and financial considerations (Vepakomma et al. 2018). The precise diagnose of diseases and the development of efficient drugs urgently demand plenty of patients data with complementary modalities, especially when the number of patients are rare and distributed across various places. Besides, raw data from different modalities are likely unaligned since patients can choose medical tests autonomously dependent on their symptoms. Therefore, the multi-modal data in practice are often unaligned and distributed, and thus a robust multi-modal learning framework is required to learn representation models based on such data.

In this study, we propose a novel optimal transport-based mixer (OTM) for achieving robust multi-modal learning based on unaligned and distributed multi-modal data. As illustrated in Figure 1(b), our method leverages a multi-head Wasserstein auto-encoder (MWAE) (Tolstikhin et al. 2018) to map different modalities' samples to the common latent space, in which each WAE head is deployed locally in an agent. A central server receives latent codes from the agents and fuses them by solving a stochastic fused Gromov-Wasserstein barycenter (FGWB) problem, leading to the proposed OT-based mixer module. The central server pushes the learned barycenter forward to different modalities, and the agents receive the pushforward results and reconstruct the corresponding modalities' samples accordingly. The MWAE model is learned by minimizing the reconstruction loss of each modality and regularizing the la-

[*]Corresponding author



(a) Practical multi-modal case

(b) The scheme of OTM-based multi-modal learning

Figure 1: (a) An illustration of the practical multi-modal learning scenario with unaligned and distributed multi-modal data. (b) The scheme of our multi-modal learning framework based on the OT-mixer module. In this framework, for the modality with less informative data, the OT-based mixer can leverage other modalities' information to train the model of this modality, which helps improve model performance.

tent codes (e.g., enhancing the clustering structure of the barycenter for unsuperivsed learning and predicting labels for supervised learning).

Due to the merits of the Fused Gromov-Wasserstein (FGW) barycenter (Vayer et al. 2020; Ma et al. 2024), the OTM is suitable for handling the heterogeneous and totally unaligned multi-modal data while remaining the instinct relationship among each modality. For each agent, the received pushforward results contain the information of other modalities, which helps improve its WAE model when the local modality is less informative. In addition, the proposed framework does not require the agents to send raw data to the central server, which can avoid data leakage. Experiments on multi-modal clustering and classification demonstrate the effectiveness of our method compared with the existing baselines, especially in the unaligned multi-modal scenarios.

Related Work and Preliminaries

Multi-modal Learning

As aforementioned, traditional multi-modal learning methods such as MCCA (de Cheveigné et al. 2019), DCCAE (Wang et al. 2015), MVKSC (Guo et al. 2014) and MultiNMF (Liu et al. 2013) are dependent on the well-aligned data, while in practice the distributions of each modality might be different from each other due to technical restrictions (Ramachandram and Taylor 2017) and the correspondence among modalities might be unknown because of privacy protection (Yu et al. 2021). To extend these methods for unaligned multi-modal data, the method in (Ma et al. 2021) proposes a multi-modal imputation method for incomplete or partially unaligned multi-modal data. The method in (Guo et al. 2022) proposes a supervised multi-modal learning framework for totally-unaligned multi-modal data. The work in (Duan et al. 2022; Yu et al. 2021) achieves the multi-modal alignment in the latent space for two modalities. The deep generative learning methods in (Ramachandram and Taylor 2017; Hu, Nie, and Li 2019) also show great potentials in the multi-modal learning for missing or unaligned modalities. However, these methods seldom consider aligning more than three modalities due to their scalability issues. In addition, these methods often require the interaction and fusion of raw data, which makes them inapplicable in the distributed scenarios with privacy-preserving considerations.

Optimal Transport-based Mixup

Data mixing (Zhang et al. 2018) is an effective technique to augment data, which helps enhance the diversity of training data and, accordingly, the generalization power of target model. Typical mixing strategies, e.g., Mixup (Zhang et al. 2018) and CutMix (Yun et al. 2019), are proposed to augment data in the Euclidean space by linear interpolation. For example, given a pair of labeled samples, denoted as $(\boldsymbol{x}_1, \boldsymbol{y}_1)$ and $(\boldsymbol{x}_2, \boldsymbol{y}_2)$, Mixup generates a new sample by

$$\boldsymbol{x}_\lambda = \lambda \boldsymbol{x}_1 + (1-\lambda)\boldsymbol{x}_2, \boldsymbol{y}_\lambda = \lambda \boldsymbol{y}_1 + (1-\lambda)\boldsymbol{y}_2.$$

For multi-modal data, however, data mixing becomes challenging since the data of different modalities are likely to be heterogeneous and even unaligned in distributed scenarios. In such a situation, the simple linear interpolation of raw data becomes less interpretable even inapplicable (Lee et al. 2020a), and mixing their latent codes would be more efficient and reasonable (Bengio et al. 2013).

From the viewpoint of optimal transport theory, data mixing can be treated as the pushforward of data distribution (Peyré, Cuturi et al. 2019). Such OT-based data mixing has been widely used in domain adaptation tasks (Courty et al. 2016), and recently, some OT-based mixing method is proposed for multi-modal learning. The work in (Luo, Xu, and Carin 2022) proposes a differentiable hierarchical optimal transport model to learning multi-modal representation model for unaligned data, in which different modalities' samples are fused by sliced Wasserstein distance. The method GWMAC in (Gong, Nie, and Xu 2022) shows great potentials of Gromov-Wasserstein barycenter (Peyré, Cuturi, and Solomon 2016) to aggregate information across different modalities' kernel matrices in the totally unaligned situation. The work in (Ma et al. 2024) makes use of the Fused Gromov-Wasserstein (FGW) distance to mix up two graphs for the classification tasks, which considers both the structural information and the node attribute relationship, and thus gets an excellent performance. Inspired by the above work, we implement a new OT-based mixer based on stochastic fused Gromov-Wasserstein barycenter (FGWB), which fuses as well as aligns the latent codes of different modalities for robust multi-modal representation learning.

Proposed OT-based Mixer

Problem Statement

Suppose that we have a set of multi-modal data, denoted as $\mathcal{D} = \{\boldsymbol{X}_m\}_{m=1}^{M}$, where M is the number of modalities. The data of the m-th modality, i.e., $\boldsymbol{X}_m = \{\boldsymbol{x}_{m,j}\}_{j=1}^{N_m} \in \mathbb{R}^{N_m \times D_m}$, contains N_m D_m-dimensional samples. We aim to learn M WAE models, i.e., $\{f_m, g_m\}_{m=1}^{M}$, in an unsupervised way. The encoder $f_m : \mathcal{X}_m \mapsto \mathcal{Z}$ maps the samples of the m-th modality from the sample space $\mathcal{X}_m$ to the latent space $\mathcal{Z} \subset \mathbb{R}^d$, and the decoder $g_m : \mathcal{Z} \mapsto \mathcal{X}_m$ maps latent codes in $\mathcal{Z}$ back to the corresponding data space.

As aforementioned, the data of different modalities are scattered to different agents and sharing data across the agents is forbidden. In addition, some modalities are less informative than others, which cannot well support the learning of the corresponding WAE models. Therefore, we would like to propose a robust multi-modal learning framework, leveraging multi-modal information during training while avoiding data sharing. In the following content, we will show that we can achieve this aim by introducing an OT-based mixer based on stochastic FGW barycenter.

An OTM Based on Stochastic FGW Barycenter

As shown in Figure 1(b), in the training phase, each agent first derives the latent codes of its data locally, i.e., $\boldsymbol{Z}_m = [\boldsymbol{z}_{m,j}] = f_m(\boldsymbol{X}_m) \in \mathbb{R}^{N_m \times d}$. A central server collects all the latent codes from the M agents and fuses the latent codes by solving the following FGW barycenter problem:

$$\boldsymbol{Z}_B, \{\boldsymbol{T}_m^*\}_{m=1}^{M} = \arg\min_{\boldsymbol{Z}} \sum_{m=1}^{M} \mathrm{FGW}\left(\boldsymbol{Z}, \boldsymbol{Z}_m; \alpha\right), \quad (1)$$

where $\boldsymbol{Z}_B \in \mathbb{R}^{N_B \times d}$ denotes the barycenter consisting N_B fused latent codes. FGW $(\boldsymbol{Z}, \boldsymbol{Z}_i; \alpha)$ is the Fused Gromov-Wasserstein distance (Vayer et al. 2020) defined as

$$\begin{aligned} &\mathrm{FGW}\left(\boldsymbol{Z}, \boldsymbol{Z}_m; \alpha\right) \\ &= \min_{\boldsymbol{T}_m \in \Pi(\boldsymbol{\mu}, \boldsymbol{\mu}_m)} \sum_{i,j,k,l} \alpha \underbrace{d_{\mathcal{Z}}^2(\boldsymbol{z}_i, \boldsymbol{z}_{m,j}) t_{m,ij}}_{\text{Wasserstein term}} \\ &\quad + (1-\alpha) \underbrace{|\boldsymbol{A}(i,k) - \boldsymbol{A}_m(j,l)|^2 t_{m,ij} t_{m,kl}}_{\text{Gromov-Wasserstein term}}, \end{aligned} \quad (2)$$

where $d_{\mathcal{Z}}(\boldsymbol{z}_i, \boldsymbol{z}_{m,j})$ is the distance metric between the fused code $\boldsymbol{z}_i$ in the barycenter and the j-th latent code of the m-th modality (i.e., $\boldsymbol{z}_{m,j}$). $\boldsymbol{A} = [d_{\mathcal{Z}}(\boldsymbol{z}_i, \boldsymbol{z}_{i'})] \in \mathbb{R}^{N_B \times N_B}$ and $\boldsymbol{A}_m = [d_{\mathcal{Z}}(\boldsymbol{z}_{m,j}, \boldsymbol{z}_{m,j'})] \in \mathbb{R}^{N_m \times N_m}$ are the distance matrices that capture the relations between the latent codes within the barycenter and the modality, respectively. $\boldsymbol{T}_m = [t_{m,ij}]$ is the transport map, whose feasible domain is $\Pi(\boldsymbol{\mu}_B, \boldsymbol{\mu}_m) = \{\boldsymbol{T}_m \in \mathbb{R}_+^{N_B \times N_m} \mid \boldsymbol{T}_m \boldsymbol{1}_{N_m} = \boldsymbol{\mu}_B, \boldsymbol{T}_m^\top \boldsymbol{1}_{N_B} = \boldsymbol{\mu}_m\}$, where $\boldsymbol{1}$ denotes all-one vector, and $\boldsymbol{\mu}_B$ and $\boldsymbol{\mu}_m$ are empirical sample distributions of $\boldsymbol{Z}$ and $\boldsymbol{Z}_m$, respectively, which are set to be uniform distributions in our implementation. According to the definition, FGW distance in (2) jointly considers the distance between two latent code sets and that between the pairwise latent code relations within the two sets, in which the hyperparameter $\alpha \in [0, 1]$ achieves the trade-off between the corresponding Wasserstein and Gromov-Wasserstein terms.

Algorithm 1: Computation of FGW distance

```
Input: The latent codes and their distributions $\{\boldsymbol{Z}_m, \boldsymbol{\mu}_m, \boldsymbol{Z}, \boldsymbol{\mu}_B\}$, the hyperparameter $\alpha$.
Initialize $\boldsymbol{T} = \boldsymbol{\mu}_B \boldsymbol{\mu}_m^\top$.
Construct $\boldsymbol{A}$, $\boldsymbol{A}_m$, and $\boldsymbol{D}_m = [d_{\mathcal{Z}}^2(\boldsymbol{z}_i, \boldsymbol{z}_{m,j})]$.
while not converge do
    ($i$) Apply the network flow algorithm: obtain $\tilde{\boldsymbol{T}}$ by $\min_{\boldsymbol{T} \in \Pi(\boldsymbol{\mu}_B, \boldsymbol{\mu}_m)} \langle \alpha \boldsymbol{D}_m - 2(1-\alpha) \boldsymbol{A}^T \boldsymbol{T} \boldsymbol{A}_m, \boldsymbol{T} \rangle$.
    ($ii$) Apply the line search method:
    $a = -2(1-\alpha) \langle \boldsymbol{A}^\top \tilde{\boldsymbol{T}} \boldsymbol{A}_m, \tilde{\boldsymbol{T}} \rangle$,
    $g = (\boldsymbol{A} \odot \boldsymbol{A}) \boldsymbol{\mu}_B \boldsymbol{1}_{N_m}^\top + \boldsymbol{1}_{N_B} \boldsymbol{\mu}_m^\top (\boldsymbol{A}_m \odot \boldsymbol{A}_m)$,
    $b = \langle \alpha \boldsymbol{D}_m + (1-\alpha) g, \tilde{\boldsymbol{T}} \rangle - 2(1-\alpha)(\langle \boldsymbol{A} \tilde{\boldsymbol{T}} \boldsymbol{A}_m, \boldsymbol{T} \rangle + \langle \boldsymbol{A} \boldsymbol{T} \boldsymbol{A}_m, \tilde{\boldsymbol{T}} \rangle)$,
    $c = \langle \alpha \boldsymbol{D}_m + (1-\alpha) g - 2(1-\alpha) \boldsymbol{A} \boldsymbol{T} \boldsymbol{A}_m^\top, \boldsymbol{T} \rangle$
    if $a > 0$ then
        $\tau = \min(1, \max(0, \frac{-(b+c)}{2a}))$
    else
        $\tau = 1$ if $a + b + c < 0$ else $\tau = 0$
    end if
    ($iii$) Update OT matrix: $\boldsymbol{T} \leftarrow (1-\tau) \boldsymbol{T} + \tau \tilde{\boldsymbol{T}}$
end while
Output: $\boldsymbol{T}_m^* := \boldsymbol{T}$.
```

Algorithm 2: Computation of FGW barycenter

```
Input: The number of fused latent code $N_B$, the latent codes and their distributions $\{\boldsymbol{Z}_m, \boldsymbol{\mu}_m\}_{m=1}^{M}$.
Initialize $\boldsymbol{Z}$ randomly.
while not converge do
    for $m = 1, \dots, M$ do
        Given $\boldsymbol{Z}$ and $\boldsymbol{Z}_m$, compute $\boldsymbol{T}_m^*$ by Algorithm 1.
    end for
    Update $\boldsymbol{Z}_B$ by $\sum_m \mathrm{diag}\left(\frac{N_B}{\boldsymbol{\mu}_B}\right) \boldsymbol{T}_m^* \boldsymbol{Z}_m$.
end while
Output: $\{\boldsymbol{T}_m^*\}_{m=1}^{M}$ and $\boldsymbol{Z}_B$.
```

Computation of FGW Barycenter The FGW barycenter can be computed in an alternating optimization framework. Specifically, we need to compute the M FGW distances in (1) to derive optimal transport matrices and then update the barycenter accordingly. Repeating the above two steps till convergence, we obtain the optimal transport matrices and the barycenter. When computing the optimal transport matrices, we apply the conditional gradient algorithm in (Vayer et al. 2020) in this study, which can obtain sparse optimal transport matrices with relatively lower computational cost (compared to the proximal gradient algorithm (Peyré, Cuturi, and Solomon 2016; Xu et al. 2019) and the Bregman ADMM algorithm (Xu 2020)). Algorithm 1 shows the computational pipeline of optimal transport matrix, and Algorithm 2 summarizes the scheme of FGW barycenter, where $\odot$ is Hadamard product of matrix and $\langle\cdot,\cdot\rangle$ is the inner product operation.

Stochastic Mixing By solving the FGW barycenter problem in (1), we can derive a set of fused latent codes (i.e., $\boldsymbol{Z}_B$), whose FGW distance to the latent codes of each modality is minimized. In addition, the optimal transport matrix between $\boldsymbol{Z}_B$ and each $\boldsymbol{Z}_m$, denoted as $\boldsymbol{T}_m^*$, is derived, which indicates the correspondence between the latent codes in $\boldsymbol{Z}_m$ and the fused ones in $\boldsymbol{Z}_B$. In particular, given $\boldsymbol{T}_m^*$, the large value $t_{m,ij}^*$ implies that the latent code $\boldsymbol{z}_{m,j}$ contributes significantly to the fused latent code $\boldsymbol{z}_i$. As a result, the optimal transport matrices provide strong evidence for data mixing.

Based on the optimal transport matrices, the proposed OTM derives the mixing result as follows:

$$\tilde{\boldsymbol{Z}}_B = N_B \sum_{i=1}^{M} \boldsymbol{M}_m \odot (\boldsymbol{T}_m^* \boldsymbol{Z}_m), \text{ with } \sum_{m=1}^{M} \boldsymbol{M}_m = \boldsymbol{1}_{N_B \times d},$$

where $\boldsymbol{T}_m^*\boldsymbol{Z}_m$ is the pushforward of $\boldsymbol{Z}_m$ based on the optimal transport matrix $\boldsymbol{T}^*$ (Courty et al. 2016), and $\odot$ is Hadamard product of matrix. Instead of using the barycenter directly as the data mixing result, OTM imposes random masks, denoted as $\{\boldsymbol{M}_m \in \{0,1\}^{N_B\times d}\}$, on the pushforward results, which introduces randomness to the mixing results. Note that, the summation of the masks equals to an all-one matrix, which means that for each fused code in $\tilde{\boldsymbol{Z}}_B$, each of its feature dimensions is determined by a single modality. Compared with the barycenter $\boldsymbol{Z}_B$, which equals to $N_B\sum_m \boldsymbol{T}_m^*\boldsymbol{Z}_m$, introducing the masks helps avoid over-smoothed mixing results, which leads to the proposed stochastic FGW barycenter.

As a result, the m-th agent receives $\tilde{\boldsymbol{Z}}_B$ and $\boldsymbol{T}_m^*$ from the central server and augments their local latent codes by

$$\tilde{\boldsymbol{Z}}_m = (\boldsymbol{T}_m^*)^\top \tilde{\boldsymbol{Z}}_B. \tag{3}$$

As shown in Figure 1(b), such augmented latent codes can improve the latent distributions of those less information modalities, which improves the learning of the corresponding WAE model.

In our framework, different modalities only share their unaligned latent codes rather than the raw data when applying our OT-mixer. Because the raw data are not shared and each modality only accesses its own decoder, it cannot recover the data of the other modalities based on its own decoder. In addition, like differential privacy strategies, the random mask matrix in Eq.(3) further introduces additional uncertainty into the barycenter. In Eq.(4), although the barycenter contains the information of other modalities, it has aggregated all the modalities' information, making them indistinguishable. The m-th modality cannot recover the raw data of the other modalities purely based on T_m and Z_b.

Connections to Existing OT-based Methods Similar to the OT-based multi-modal fusion methods in (Luo, Xu, and Carin 2022; Gong, Nie, and Xu 2022), our OTM is applicable for unaligned multi-modal data with the help of the optimal transport matrices. In addition, because the fusion is applied to latent codes, our method does not require the share of raw data, which is friendly to privacy-preserving distributed scenarios. Note that, the GWMAC in (Gong, Nie, and Xu 2022) only considers the GW barycenter when fusing latent codes, while the DHOT in (Luo, Xu, and Carin 2022) only considers the sliced Wasserstein distance between paired modalities. Our OTM jointly considers both the Wasserstein distance and the GW distance between latent codes, which can lead to more reliable fusion results. In addition, the stochastic masking used in OTM mitigates the over-smoothness issue that is common in the existing OT-based fusion methods.

Robust Multi-modal Learning Framework

As shown in Figure 1(b), we can learn the WAE models by minimizing the Wasserstein distance between each modality's data distribution and model distribution, i.e., $\min_{\{f_m,g_m\}_{m=1}^M} W(p_m,q_m)$, where p_m is the data distribution of $\boldsymbol{X}_m$, and q_m is the distribution parametrized by the model. As shown in (Tolstikhin et al. 2018), this learning task can be approximately implemented as the reconstruction loss of each modality with a regularization on the latent representations (Xu et al. 2020).

$$\min_{\{f_m,g_m\}_{m=1}^M} \sum_{m=1}^{M} \|\boldsymbol{X}_m - \hat{\boldsymbol{X}}_m\|_F^2 + \lambda R(\{\boldsymbol{Z}_m\}_{m=1}^M), \tag{4}$$

where $\hat{\boldsymbol{X}}_m$ is the data reconstructed by latent codes, $R(\cdot)$ is the regularizer of latent codes, and $\lambda > 0$ is the weight of the regularizer.

In our framework, the reconstruction loss of each modality is derived based on both its original data and the mixing results, i.e.,

$$\mathcal{L}_{mix} = \sum_{m=1}^{M} \|\boldsymbol{X}_m - g_m(\boldsymbol{Z}_m)\|_F^2 + \|\boldsymbol{Z}_m - \tilde{\boldsymbol{Z}}_m\|_F^2, \tag{5}$$

where the first term is the reconstruction loss for samples, while the second term penalizes the discrepancy between the latent codes of a single modality and the fused latent codes, which impose the information of other modalities to the learning of the target modality.

For the regularizer, we can implement it according to the availability of data label and downstream tasks.

Unsupervised Multi-modal Clustering In our approach, the unsupervised clustering is performed on the fusion of the latent representation from each modality, i.e., $\tilde{\boldsymbol{Z}}_B$. Following the work (Gong, Nie, and Xu 2022), we perform the spectral clustering on $\tilde{\boldsymbol{Z}}_B$ by computing the Gromov-Wasserstein distance between $\tilde{\boldsymbol{Z}}_B$ and a predefined identity matrix with size $C \times C$, i.e., $\boldsymbol{I}_C$, where C is the number of clusters. As a result, the regularizer is implemented as

$$\underbrace{\min_{\boldsymbol{T}\in\Pi(\boldsymbol{\mu}_B,\boldsymbol{\mu}_C)} \sum\nolimits_{i,j,k,l} |\boldsymbol{K}(i,k) - \boldsymbol{I}_C(j,l)|^2 t_{ij} t_{kl},}_{GW(\tilde{\boldsymbol{Z}}_B, \boldsymbol{I}_C)} \tag{6}$$

where $\boldsymbol{K} = [\kappa(\tilde{\boldsymbol{z}}_i, \tilde{\boldsymbol{z}}_{i'})] \in \mathbb{R}^{N_B \times N_B}$ is the similarity matrix of $\tilde{\boldsymbol{Z}}_B$, which is a Gaussian kernel. $\boldsymbol{\mu}_C = \frac{1}{C}\boldsymbol{1}_C$ denotes the distribution of the clusters. $\boldsymbol{T} = [t_{ij}]$ is the optimal transport map, whose feasible domain is $\Pi(\boldsymbol{\mu}_B, \boldsymbol{\mu}_C)$.

Supervised Multi-modal Learning For supervised multi-modal classification and regression tasks, we combine OTM with existing multi-modal fusion paradigms to obtain the augmented latent representations. Given a labeled modality $(\boldsymbol{Z}_m, \boldsymbol{y}_m)$, where $\boldsymbol{y}_m \in \mathbb{R}^{N_m}$ is the labels of the samples (the latent codes), we can augment the data as $(\tilde{\boldsymbol{Z}}_m, \boldsymbol{y}_m)$ Accordingly, the regularizer becomes

$$\mathcal{L}_s(h_m(\tilde{\boldsymbol{Z}}_m), \boldsymbol{y}_m) + \mathcal{L}_s(h_m(\boldsymbol{Z}_m), \boldsymbol{y}_m), \tag{7}$$

where $\mathcal{L}_s$ is the supervised loss (e.g., MAE (Willmott and Matsuura 2005) or MSE for regression and CrossEntropy for classification). $h_m : \mathcal{Z} \mapsto \mathcal{Y}$ is the local predictor for the m-th modality.

Experiments

To demonstrate the effectiveness our OTM method on robust multi-model learning, we apply it to unsupervised multi-modal clustering tasks and superivsed mulit-modal classification and regresstion tasks, respectively, and compare it with state-of-the-art methods. In addition, the impact of the stochastic mixing strategy on the robustness of our method is analyzed as well. The code and more experimental results can be found at https://github.com/redLinmumu/OTM.

Experimental Setup

Datasets For clustering tasks, we conduct the experiments on four conventional multi-modal dataset used in (Hu, Nie, and Li 2019; Guo et al. 2014; Gong, Nie, and Xu 2022). Each dataset contains well-aligned samples and corresponding labels, which are used only in the validation stage. The dataset for the classification and regression tasks are chosen from Multibench (Liang et al. 2021), which is a well-known systematic large-scale multi-modal learning benchmark. The dataset for the classification and regression tasks are pre-processed and filtered in the same way as the framework Multibench (Liang et al. 2021) does. The information of the dataset is summarized in Table 1. To imitate unaligned multi-modal data, we permute the training data of each modality randomly and train multi-modal models by various methods on well-aligned and unaligned multi-modal data, respectively. The validation and testing data are maintained to be well-aligned for a fair comparison.

Dataset	Size	Dimensions $\{D_m\}_{m=1}^{M}$	C	M
Caltech7	1474	[48, 40, 254, 1984, 512, 928]	7	6
ORL	400	[288, 288]	40	2
Movies	617	[1878, 1398]	17	2
Prokaryotic	551	[438, 3, 393]	4	3
ENRICO	1458	[98304, 98304]	2	2
CMU-MOSI	2183	[1750, 3700, 15000]	2	3
AV-MNIST	70000	[784, 12544]	2	2
MUJOCO	37990	[48, 112, 16384, 112]	-	4

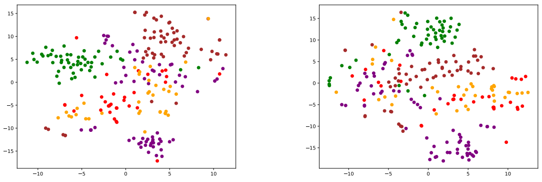
Table 1: Summary of datasets.

Data type	Datasets	Caltech 7	ORL	Movies	Prokaryotic
	Algorithms	Purity	Purity	Purity	Purity
Aligned	MCCA	0.5313	0.3475	0.0989	0.5620
	DCCAE	0.4110	0.5625	0.1572	0.5070
	AttnAE	0.4600	0.4600	0.1880	0.5390
	MVKSC	0.5196	0.3013	0.2285	**0.6188**
	MultiNMF	0.4525	**0.6900**	0.1726	0.5771
	MWAE+OTM	**0.6072**	0.6563	**0.3177**	0.5952
Unaligned	MVC-UM	0.3112	0.5431	0.1841	0.4451
	GWMAC	0.3568	0.5118	0.1928	**0.5479**
	MWAE+OTM	**0.5788**	**0.6550**	**0.2925**	0.5438

Table 2: The performance of different clustering methods.

Baselines For clustering tasks, we compare our method with five methods for well-aligned data (i.e., MCCA (de Cheveigné et al. 2019), DCCAE (Wang et al. 2015), AttnAE, MVKSC (Guo et al. 2014), and MultiNMF (Liu et al. 2013)) and two state-of-the-art unaligned clustering methods (i.e., GWMAC (Gong, Nie, and Xu 2022) and MVC-UM (Yu et al. 2021)). For each method, we concatenate the latent codes of all the modalities and apply K-means (Teknomo 2006) to obtain the clustering result. For the classification and regression tasks, we combine OTM with five common multi-modal methods: Late fusion (Baltrušaitis, Ahuja, and Morency 2018), Multimodal Factorized Model (MFM) (Tsai et al. 2018), Tensor Fusion (TF) (Zadeh et al. 2017), Low-rank Tensor Fusion (LRTF) (Liu et al. 2018), and Multiplicative Interactions (MI) MATRIX (Jayakumar et al. 2019), all of which are implemented in MultiBench (Liang et al. 2021).

Backbones and Evaluation Metrics In the following clustering experiments, we implement all encoders and decoders of MWAE by two-layer multi-layer perceptrons (MLPs). We choose the clustering purity to evaluate the clustering performance. For a fair comparison, the baselines use the same model architecture. For the classification and regression tasks, we apply the backbone models provided by Multibench and plug OTM into their training phase. Each model is trained by five-fold cross validation. For the clustering models, we apply the clustering purity to evaluate their performance. For the classification and regression models, we apply the classification accuracy and MAE (Willmott and Matsuura 2005) to evaluate them, respectively.



(a) Well-aligned latent codes (b) Fused latent codes

Figure 2: The t-SNE plots of the well-aligned latent codes and the fused ones achieved by OTM. The latent codes are learned on Caltech7 and different colors indicate different image classes. For the fused latent codes, we determine their colors by the labels of the 6-th modality.

Dataset (Task)	Method	Result
AV-MNIST (Classification)	Late fusion	0.7295
	Late fusion + OTM	**0.7316**
ENRICO (Classification)	MI matrix	0.4815
	MI matrix + OTM	**0.5034**
	Tensor matrix	0.4814
	Tensor matrix + OTM	**0.4911**
CMU-MOSI (Classification)	Late fusion	0.5194
	Late fusion + OTM	**0.5368**
	LRTF	0.5245
	LRTF + OTM	**0.5327**
	MFM	0.5391
	MFM + OTM	**0.5410**
CMU-MOSI (Regression)	Late fusion	1.3710
	Late fusion + OTM	**1.3630**
	Tenser fusion	1.3691
	Tenser fusion + OTM	**1.3644**
MUJOCO (Regression)	Tensor fusion	1.583×10^{-3}
	Tensor fusion + OTM	$\mathbf{1.369}\times10^{-3}$

Table 3: Classification accuracy and regression MAE under 5-fold cross validation.

Numerical Comparisons

Clustering The clustering performance of different methods is summarized in Table 2. We can find that for well-aligned multi-modal data, applying OTM achieves reliable data augmentation and thus leads to at least comparable clustering performance for the original data. In other words, the augmented latent codes achieved by OTM yield the same latent distribution of original well-aligned data. For unaligned multi-modal data, the clustering can only be achieved based on the fused latent codes. In such a situation, our OTM works better than its competitors — compared with the matrix factorization-based method MVC-UM and the GW barycenter-based method GWMAC, our OTM achieves significant improvements in three of four datasets. These results verify that applying OTM helps achieve competitive performance for multi-modal clustering.

The random mask helps augment barycenter and introduce uncertainty for privacy protection. To demonstrate the rationality of applying the mask, our method is compared with the classic weighted barycenter, i.e., $\sum_m \lambda_m T_m Z_m$, means setting $M_m = [\lambda_m]$. From this viewpoint, Eq.(3) can be treated as a randomized extension of the classic setting. From Table 2, we can find that replacing Eq.(3) with the classic weighted barycenter (WB) leads to performance degradation on clustering purity.

Dataset (Task)	Method	Selected Modality		
		1	2	3
Prokaryotic (Clustering)	MWAE	0.4936	**0.6261**	0.4791
	MWAE+OTM	**0.5554**	0.5209	**0.5426**
CMU-MOSI (Classification)	Late fusion	0.5369	**0.5373**	0.5163
	Late fusion+OTM	**0.5428**	0.5328	**0.5268**
	LRTF	**0.5190**	**0.5241**	0.5131
	LRTF+OTM	0.5131	0.5222	**0.5209**
CMU-MOSI (Regression)	Late fusion	**1.3721**	**1.3581**	1.4055
	Late fusion+OTM	1.3804	1.3629	**1.3698**
	Tenser fusion	1.3684	1.3680	1.3853
	Tenser fusion+OTM	**1.3674**	**1.3669**	**1.3716**

Table 4: The performance on single modality.

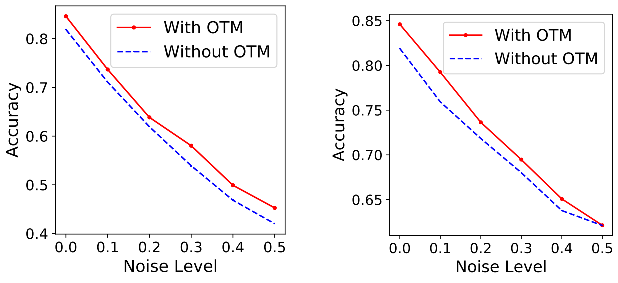
In addition, we show the t-SNE plots of the latent codes learned on Caltech7 in Figure 2. We can find that even if the multi-modal data are unaligned, applying our OTM can align different modalities well. In particular, after training MWAE+OTM, the fused latent codes $\tilde{Z}_B$ have clustering structures that are similar to the latent codes learnt from well-aligned data. This visualization demonstrates that the alignment obtained by OTM is reasonable, leading to semantic clustering structure.

Classification and regression For classification and regression tasks, we combine OTM with a series of multi-modal methods in Multibench (Liang et al. 2021), varying from traditional methods to neural learnable ones. Table 3 shows that OTM helps improve classification accuracy and MAE for various methods. These results demonstrate that our OTM is also applicable in supervised multi-modal learning tasks. In particular, the augmented data achieved by OTM are aligned well to the original ones, so that they can share the same labels. As a result, the augmented data help improve the generalization power of model.

Robustness Analysis

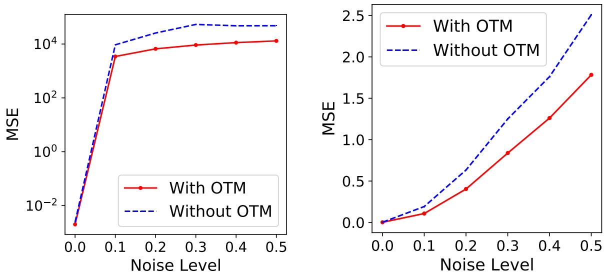
Robustness to missing modalities In distributed multi-modal learning scenarios, each agent may have to rely on its local modality to make prediction, which corresponds to a challenging learning task with missing modalities. In such a situation, learning a predictor based a single modality often leads to sub-optimal performance, especially when the modality is less informative. Applying our OTM module, each agent can leverage the information of other modalities in the training phase and thus improve the latent distribution of its own modality, which helps learn a robust model.

To verify our claim, for each multi-modal dataset used in the previous experiments, we consider learning a multi-modal model with or without OTM, respectively. In the test-



(a) Image Modality (b) Set Modality

Figure 3: Robustness test on ENRICO dataset for classification task with MI matrix fusion strategy.



(a) Image Modality (b) Haptics Modality

Figure 4: Robustness test on MUJOCO dataset for regression task with Tensor fusion strategy.

ing phase, we only apply the samples of the selected modality to evaluate the performance of the learned models. For those multi-modal models, only the latent codes of the selected modality are applied to make predictions. Table 4 shows the experimental results, we can find that applying OTM helps improve the robustness of multi-modal model to the missing modality issue in most situations. For the less-informative modalities in each dataset, applying OTM introduces the information of other modalities into their representation models and thus improves model performance.

Robustness to data noise Besides missing modalities, multi-modal data are likely to be corrupted by data noise. Therefore, we further evaluate the robustness of models trained with noisy-free data by testing on noisy test data for classification and regression tasks and analyze the impact of OTM accordingly. In the classification task, we choose the MI matrix fusion strategy on the ENRICO dataset that has two modalities, the visual image and the set of wireframe image. In the regression task, we choose the Tensor fusion method on the MUJOCO dataset and add noise to its image and haptics modalities, respectively. For each modality, we add random noise with increasing noise levels in the range $[0, 0.5]$. Zero means no noise, and 0.5 means the energy of noise is a half of data energy.

The comparison results for the models learnt with and without OTM are illustrated in Figures 3 and 4. We can find that applying OTM helps achieve higher classification accuracy and lower MAE consistently. Table 5 further provides the numerical comparisons for more models and more datasets when the noise level is around 0.2, which further verifies the effectiveness of OTM. All these results demonstrate that applying OTM can improve the robustness of multi-modal model to data noise.

Dataset (Task)	Method	Result
CMU-MOSI (Classification)	Late fusion	0.4922
	Late fusion+OTM	**0.5062**
	LRTF	0.5073
	LRTF+OTM	**0.5122**
	MFM	0.5078
	MFM+OTM	**0.5264**
	Late fusion	0.4877
	Late fusion+OTM	**0.5270**
	Tenser fusion	0.4959
	Tenser fusion+OTM	**0.5106**
MUJOCO (Regression)	Tensor fusion	0.1971
	Tensor fusion+OTM	**0.1906**

Table 5: The robustness test on noisy data.

Conclusion

We propose a novel optimal transport-based mixer (OTM) that achieves data alignment and augmentation for robust multi-modal learning. Our method leverages a multi-head Wasserstein autoencoder (MWAE) to encode different modalities' samples in the same latent space, and the proposed OTM computes a stochastic fused Gromov-Wasserstein barycenter to mix different modalities' latent codes, then reconstructing the modalities' samples accordingly. Experiments on multi-modal dataset demonstrate that our OTM could achieve promising performance in various multi-modal learning tasks, which can deal with unaligned and distributed multi-modal data and enhance the robustness of model to data noise and missing modalities.

Limitations and future work. As shown in Table 4, although applying OTM is useful for less informative modalities, it leads to performance degradation for some other modalities. It implies that our method introduces useless noise to some significant modalities. How to suppress this issue is one of our future work. In addition, we would like to improve the efficiency of data mixing by accelerating the computation of FGW barycenter, e.g., replacing the FGW distance to sliced FGW distance (Xu et al. 2020). In the future, we plan to test our method in real-world applications, e.g., federated learning for healthcare data modeling.

Acknowledgments

This work was supported by National Natural Science Foundation (62106271, 92270110), the Fundamental Research Funds for the Central Universities, and the Research Funds of Renmin University of China. We also acknowledge the support provided by the fund for building world-class universities (disciplines) of Renmin University of China and by the funds from Engineering Research Center of Next-Generation Intelligent Search and Recommendation, Ministry of Education, and from Intelligent Social Governance Interdisciplinary Platform, Major Innovation & Planning Interdisciplinary Platform for the "Double-First Class" Initiative, Renmin University of China.

References

Abilov, A.; Hua, Y.; Matatov, H.; Amir, O.; and Naaman, M. 2021. Voterfraud2020: a multi-modal dataset of election fraud claims on twitter. In *Proceedings of the international AAAI conference on web and social media*, volume 15, 901–912.

Aronoff, S.; et al. 1982. Classification accuracy: a user approach. *Photogrammetric Engineering and Remote Sensing*, 48(8): 1299–1307.

Baldi, P.; Brunak, S.; Chauvin, Y.; Andersen, C. A.; and Nielsen, H. 2000. Assessing the accuracy of prediction algorithms for classification: an overview. *Bioinformatics*, 16(5): 412–424.

Baltrušaitis, T.; Ahuja, C.; and Morency, L.-P. 2018. Multimodal machine learning: A survey and taxonomy. *IEEE transactions on pattern analysis and machine intelligence*, 41(2): 423–443.

Bengio, Y.; Mesnil, G.; Dauphin, Y.; and Rifai, S. 2013. Better mixing via deep representations. In *International conference on machine learning*, 552–560. PMLR.

Courty, N.; Flamary, R.; Tuia, D.; and Rakotomamonjy, A. 2016. Optimal transport for domain adaptation. *IEEE transactions on pattern analysis and machine intelligence*, 39(9): 1853–1865.

de Cheveigné, A.; Di Liberto, G. M.; Arzounian, D.; Wong, D. D.; Hjortkjær, J.; Fuglsang, S.; and Parra, L. C. 2019. Multiway canonical correlation analysis of brain data. *neuroimage*, 186: 728–740.

Duan, J.; Chen, L.; Tran, S.; Yang, J.; Xu, Y.; Zeng, B.; and Chilimbi, T. 2022. Multi-modal alignment using representation codebook. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 15651–15660.

Gong, F.; Nie, Y.; and Xu, H. 2022. Gromov-Wasserstein multi-modal alignment and clustering. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, 603–613.

Guo, D.; Zhang, J.; Liu, X.; Cui, Y.; and Zhao, C. 2014. Multiple kernel learning based multi-view spectral clustering. In *2014 22nd International conference on pattern recognition*, 3774–3779. IEEE.

Guo, J.; Tang, J.; Dai, W.; Ding, Y.; and Kong, W. 2022. Dynamically adjust word representations using unaligned multimodal information. In *Proceedings of the 30th ACM International Conference on Multimedia*, 3394–3402.

Hu, D.; Nie, F.; and Li, X. 2019. Deep multimodal clustering for unsupervised audiovisual learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 9248–9257.

Jayakumar, S. M.; Czarnecki, W. M.; Menick, J.; Schwarz, J.; Rae, J.; Osindero, S.; Teh, Y. W.; Harley, T.; and Pascanu, R. 2019. Multiplicative Interactions and Where to Find Them. In *International Conference on Learning Representations*.

LeCun, Y.; Bottou, L.; Bengio, Y.; and Haffner, P. 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11): 2278–2324.

Lee, K.; Zhu, Y.; Sohn, K.; Li, C.-L.; Shin, J.; and Lee, H. 2020a. i-Mix: A Domain-Agnostic Strategy for Contrastive Representation Learning. In *International Conference on Learning Representations*.

Lee, M. A.; Yi, B.; Martín-Martín, R.; Savarese, S.; and Bohg, J. 2020b. Multimodal sensor fusion with differentiable filters. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 10444–10451. IEEE.

Leiva, L. A.; Hota, A.; and Oulasvirta, A. 2020. Enrico: A high-quality dataset for topic modeling of mobile UI designs. *Proc. MobileHCI extended abstracts*.

Liang, P. P.; Lyu, Y.; Fan, X.; Wu, Z.; Cheng, Y.; Wu, J.; Chen, L.; Wu, P.; Lee, M. A.; Zhu, Y.; et al. 2021. Multibench: Multiscale benchmarks for multimodal representation learning. *arXiv preprint arXiv:2107.07502*.

Liu, J.; Wang, C.; Gao, J.; and Han, J. 2013. Multi-view clustering via joint nonnegative matrix factorization. In *Proceedings of the 2013 SIAM international conference on data mining*, 252–260. SIAM.

Liu, Z.; Shen, Y.; Lakshminarasimhan, V. B.; Liang, P. P.; Zadeh, A. B.; and Morency, L.-P. 2018. Efficient Low-rank Multimodal Fusion With Modality-Specific Factors. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2247–2256.

Luo, D.; Xu, H.; and Carin, L. 2022. Differentiable hierarchical optimal transport for robust multi-view learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(6): 7293–7307.

Ma, M.; Ren, J.; Zhao, L.; Tulyakov, S.; Wu, C.; and Peng, X. 2021. Smil: Multimodal learning with severely missing modality. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 2302–2310.

Ma, X.; Chu, X.; Wang, Y.; Lin, Y.; Zhao, J.; Ma, L.; and Zhu, W. 2024. Fused Gromov-Wasserstein Graph Mixup for Graph-level Classifications. *Advances in Neural Information Processing Systems*, 36.

Marutho, D.; Handaka, S. H.; Wijaya, E.; et al. 2018. The determination of cluster number at k-mean using elbow method and purity evaluation on headline news. In *2018 international seminar on application for technology of information and communication*, 533–538. IEEE.

Peyré, G.; Cuturi, M.; and Solomon, J. 2016. Gromov-wasserstein averaging of kernel and distance matrices. In *International conference on machine learning*, 2664–2672. PMLR.

Peyré, G.; Cuturi, M.; et al. 2019. Computational optimal transport: With applications to data science. *Foundations and Trends® in Machine Learning*, 11(5-6): 355–607.

Ramachandram, D.; and Taylor, G. W. 2017. Deep multimodal learning: A survey on recent advances and trends. *IEEE signal processing magazine*, 34(6): 96–108.

Sharma, K.; and Giannakos, M. 2020. Multimodal data capabilities for learning: What can multimodal data tell us about learning? *British Journal of Educational Technology*, 51(5): 1450–1484.

Sripada, S. C.; and Rao, M. S. 2011. Comparison of purity and entropy of k-means clustering and fuzzy c means clustering. *Indian journal of computer science and engineering*, 2(3): 343–346.

Teknomo, K. 2006. K-means clustering tutorial. *Medicine*, 100(4): 3.

Tolstikhin, I.; Bousquet, O.; Gelly, S.; and Schoelkopf, B. 2018. Wasserstein Auto-Encoders. In *International Conference on Learning Representations*.

Tsai, Y.-H. H.; Liang, P. P.; Zadeh, A.; Morency, L.-P.; and Salakhutdinov, R. 2018. Learning Factorized Multimodal Representations. In *International Conference on Learning Representations*.

Vayer, T.; Chapel, L.; Flamary, R.; Tavenard, R.; and Courty, N. 2020. Fused Gromov-Wasserstein distance for structured objects. *Algorithms*, 13(9): 212.

Vepakomma, P.; Gupta, O.; Swedish, T.; and Raskar, R. 2018. Split learning for health: Distributed deep learning without sharing raw patient data. *arXiv preprint arXiv:1812.00564*.

Vielzeuf, V.; Lechervy, A.; Pateux, S.; and Jurie, F. 2018. Centralnet: a multilayer approach for multimodal fusion. In *Proceedings of the European Conference on Computer Vision (ECCV) Workshops*.

Wang, W.; Arora, R.; Livescu, K.; and Bilmes, J. 2015. On deep multi-view representation learning. In *International conference on machine learning*, 1083–1092. PMLR.

Willmott, C. J.; and Matsuura, K. 2005. Advantages of the mean absolute error (MAE) over the root mean square error (RMSE) in assessing average model performance. *Climate research*, 30(1): 79–82.

Xu, H. 2020. Gromov-Wasserstein factorization models for graph clustering. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, 6478–6485.

Xu, H.; Luo, D.; Henao, R.; Shah, S.; and Carin, L. 2020. Learning autoencoders with relational regularization. In *International Conference on Machine Learning*, 10576–10586. PMLR.

Xu, H.; Luo, D.; Zha, H.; and Carin, L. 2019. Gromov-wasserstein learning for graph matching and node embedding. In *International conference on machine learning*, 6932–6941. PMLR.

Yu, H.; Tang, J.; Wang, G.; and Gao, X. 2021. A novel multi-view clustering method for unknown mapping relationships between cross-view samples. In *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*, 2075–2083.

Yun, S.; Han, D.; Oh, S. J.; Chun, S.; Choe, J.; and Yoo, Y. 2019. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF international conference on computer vision*, 6023–6032.

Zadeh, A.; Chen, M.; Poria, S.; Cambria, E.; and Morency, L.-P. 2017. Tensor Fusion Network for Multimodal Sentiment Analysis. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, 1103–1114.

Zadeh, A.; Zellers, R.; Pincus, E.; and Morency, L.-P. 2016. Mosi: multimodal corpus of sentiment intensity and subjectivity analysis in online opinion videos. *arXiv preprint arXiv:1606.06259*.

Zhang, F.; Li, Z.; Zhang, B.; Du, H.; Wang, B.; and Zhang, X. 2019. Multi-modal deep learning model for auxiliary diagnosis of Alzheimer's disease. *Neurocomputing*, 361: 185–195.

Zhang, H.; Cisse, M.; Dauphin, Y. N.; and Lopez-Paz, D. 2018. mixup: Beyond Empirical Risk Minimization. In *International Conference on Learning Representations*.

Datasets

- **Clustering Datasets.** We conduct the clustering experiments on four widely used multi-modal dataset to evaluate our method. The multi-modal samples in each dataset are well-aligned and have the corresponding labels, which are only used in the validation stage. The links of these four clustering dataset from (Hu, Nie, and Li 2019; Guo et al. 2014; Gong, Nie, and Xu 2022) are given as follows:
  - **Caltech7** https://data.caltech.edu/records/20086
  - **ORL** https://cam-orl.co.uk/facedatabase.html
  - **Movies** http://membres-lig.imag.fr/grimal/data.html
  - **Prokaryotic** https://github.com/mbrbic/Multi-view-LRSSC/blob/master/datasets
- **Classification & Regression Datasets.** The dataset for the classification and regression tasks are chosen from Multibench (Liang et al. 2021), and are pre-processed and filtered in the same way as Multibench does. Detailed information of all dataset can be found in section C of Multibench (Liang et al. 2021)'s Appendix.
  - **CMU-MOSI** (Zadeh et al. 2016) CMU-MOSI is used in affective computing areas, which contains language, video, and audio time-series data, to predict sentiment. According to Multibench (Liang et al. 2021), CMU-MOSI was originally downloaded from https://github.com/A2Zadeh/CMU-MultimodalSDK.
  - **MUJOCO** (Lee et al. 2020b) The MUJOCO dataset we use in this paper is short for MUJOCO PUSH, which is a large-scale dataset in robotics to predict the pose of the object being pushed by the robot end-effector (Liang et al. 2021). It has records on the manipulation of simulated and real robotic arms equipped with visual (RGB and depth), force, and proprioception sensors. It can be downloaded from https://github.com/brentyi/multimodalfilter/.

  - **ENRICO** (Leiva, Hota, and Oulasvirta 2020) ENRICO (Enhanced Rico) dataset is a benchmarks for data-driven models of design in scaffolding the creation of mobile apps (Liang et al. 2021), which consists of two modalities for app classification: (1) the app screenshot image and (2) the set of unordered view hierarchy, which describes the spatial and structural layout of UI elements. It can be downloaded from https://github.com/luileito/enrico.
  - **AV-MNIST** (Vielzeuf et al. 2018) AV-MNIST pairs the audio of a human reading digits from the FSDD dataset in https://github.com/Jakobovski/free-spoken-digit-dataset with written digits in the MNIST dataset (LeCun et al. 1998), and the task is to predict the class of the digit from 0 to 9. According to Multibench (Liang et al. 2021), the preprocessing code is provided in https://github.com/slyviacassell/_MFAS/blob/master/datasets/avmnist_gen.py.

Metrics

- **Clustering Metric.** Purity is a traditional metric to measure the quality of clustering task and is widely used (Sripada and Rao 2011; Marutho et al. 2018). Purity value is computed by equation (8). The purity value is no more than 1, and the higher the purity is, the better of the clustering performance.

$$\text{Purity} = \frac{1}{N}\sum_{k=1}^{K}\max_{j}|C_k \cap L_j| \tag{8}$$

  where N is the total number of samples, K is the number of clusters, C_k is the k-th cluster, L_j is the set of samples with true label j, $|C_k \cap L_j|$ is the number of samples in the k-th cluster that belong to the true label j.
- **Classification and Regression Metric.** Accuracy (Aronoff et al. 1982; Baldi et al. 2000) is a traditional metric to measure the quality of classification task, while MSE (Willmott and Matsuura 2005) is a traditional metric to measure the quality of regression task. For regression, MAE is better than MSE for evaluating the performance as stated in (Willmott and Matsuura 2005). The less MSE or MAE is, the better of the regression performance.

Implementation Details

The random seed is setup at the beginning of the experiment, which can be found in the code files. For classification and regression experiments, the initial seed is 42. For clustering experiments, the initial seed is 123. We run our experiments on two sets of computing infrastructures. One has CPU Intel(R) Xeon(R) Gold 5218 CPU @ 2.30GHz, GPU model NVIDIA RTX A6000 with OS 20.04.1-Ubuntu x86_64 GNU, and the amount of memory is 251Gi. The other one has CPU Intel(R) Xeon(R) Gold 5218 CPU @ 2.30GHz, GPU NVIDIA GeForce RTX 3090, with OS 83∼20.04.1-Ubuntu x86_64 GNU/Linux, and the amount of memory is 251Gi. The relevant software libraries and frameworks are given in the code repositories by the yml files.

- **Clustering.** The details of the parameters and models can be found in the config directory of the repository name mwae. We conduct the clustering training and validation on the full dataset as the baselines did. The optimizer is Adam, the early stop is adopted with patience 5. α is the hyper-parameter of FGW, which varies from 0 to 1, and we tune it in [0,1] by uniform with interval 0.095333. The weight of OTM is tuned in [1e-2, 1e2] by loguniform. The weight of reconstruction loss is tuned in [1e-2, 1e2] by loguniform. The learning rate is tuned in [1e-5, 5e-2] by loguniform.
- **Classification.** The hyper-parameters of the classification and regression tasks are listed in Table 7. We only tune the OTM weight in [1e-2,1e2] by loguniform, and other parameters are consistent with Multibench. For CMU-MOSI, the max number of epochs is 100, the batch size is 32, the early stop is not applied except for the LRTF with patience 7, the optimizer is AdamW, the weight decay is 0.01, and the learning rate is 0.001. For ENRICO, the max number of epochs is 50, the batch size is 32, the early stop is not applied, the optimizer is Adam, the learning rate is 0.0001, and there is no weight decay. For AV-MNIST, the max number of epochs is 30, the batch size is 32, the early stop is not applied, the optimizer is SGD, the weight decay is 0.0001, and the learning rate is 0.1.
- **Regression.** For CMU-MOSI, the max number of epochs is 100, the batch size is 32, the early stop is not applied, the optimizer is AdamW, the weight decay is 0.01 and the learning rate is 0.001. For MUJOCO, the max number of epochs is 20, the batch size is 32, the early stop is not applied, the opimizer is AdamW, the learning rate is 0.00001, and there is no weight decay. We only tune the OTM weight in [1e-2,1e2] by loguniform, and other parameters are consistent with Multibench.
- **Robustness to Missing Modalities.** For Prokaryotic, we use the same settings of the multi-modal training to train each modality one by one. For CMU-MOSI, we loaded the model learnt with or without OTM in the previous classification and regression tasks. Then we select the dimension of the single modality's head to match the dimension of the model's output to perform classification or regression tasks. Finally, we train the head by means of the single modality training data, and valid the models one modality by one modality on the single modality validation data by 5-fold cross-validation. We implement the head by MLP, and the dimensions of the head are listed in Table 8.
- **Robustness to Data Noise.** For CMU-MOSI and MUJOCO dataset, we add noise to all modalities with different noise levels around 0.2 for robustness comparison. The hyper-parameters are listed in 9. The validation task for classification task on CMU-MOSI is classification, while the validation task for regression task on CMU-MOSI is posneg-classification, which is consistent with Multibench. The validation task for regression task on MUJOCO is regression. For this experiment, we tune the loss level in [0.1, 0.3] with interval 0.1.

Dataset	Aligned	α	Construction Weight	OTM Weight
Caltech7	False	0.8579	98.6208	0.4575
	True	0.0953	1.6462	0.0138
Movies	False	0.0953	47.2208	0.1701
	True	0.5719	33.6095	16.1002
ORL	False	0.7626	99.0937	0.1391
	True	0.4766	15.2616	15.6488
Prokaryotic	False	0.1906	1.6734	0.7819
	True	0.7626	0.0232	5.6824

Table 6: Hyper-parameter for clustering task on various dataset.

Dataset	Task	Fusion	OTM weight
CMU-MOSI	Classification	Late fusion	100.0
		LRTF	0.01
		MFM	1.0
	Regression	Late fusion	100.0
		TF	10.0
ENRICO	Classification	MI matrix	0.1
		Tensor matrix	100.0
AV-MNIST	Classification	Late fusion	0.01
MUJOCO	regression	TF	0.1

Table 7: Hyper-parameter for classification and regression tasks on various dataset.

Experiments

Rationality of Random mask

As aforementioned, the random mask helps augment barycenter and introduce uncertainty for privacy protection in Eq.(3). The classic weighted barycenter, i.e., $\sum_m \lambda_m T_m Z_m$ means setting $M_m = [\lambda_m]$. From this viewpoint, Eq.(3) can be treated as a randomized extension of the classic setting. We demonstrate the rationality of applying the mask by the following clustering experiments, comparing original OTM to OTM(WB) which is constructed by replacing Eq.(3) with the classic weighted barycenter (WB). From Table 10, we can see that OTM(WB) has performance degradation on clustering purity in unaligned case while they achieve comparable performance in aligned case.

Task	Fusion	Modality	Head
Classification	Late fusion	1	[70, 870, 2]
		2	[200, 870, 2]
		3	[600, 870, 2]
	LRTF	1	[128, 512, 2]
		2	[128, 512, 2]
		3	[128, 512, 2]
Regressioin	Late fusion	1	[70, 870, 1]
		2	[200, 870, 1]
		3	[600, 870, 1]
	Tensor Fusion	1	[4,4,1]
		2	[19,19,1]
		3	[79,79,1]

Table 8: Head parameters of CMU-MOSI's single modality.

Dataset	Train Task	Fusion	Noise	β	Epoch
CMU-MOSI	Classification	Late fusion	0.2	100.0	300
		LRTF	0.3	0.01	100
		MFM	0.2	0.1	100
	Regressioin	Late fusion	0.2	100.0	100
		Tensor Fusion	0.2	10.0	300
MUJOCO	Regressioin	Tensor Fusion	0.3	0.1	20

Table 9: Hyper-parameters of noisy test on CMU-MOSI and MUJOCO dataset.

Data type	Datasets	Caltech 7	ORL	Movies	Prokaryotic
	Algorithms	Purity	Purity	Purity	Purity
Aligned	MWAE+OTM	0.6072	**0.6563**	0.3177	**0.5952**
	MWAE+OTM(WB)	**0.6097**	0.6525	**0.3184**	0.5541
Unaligned	MWAE+OTM	**0.5788**	**0.6550**	**0.2925**	**0.5438**
	MWAE+OTM(WB)	0.5667	0.6350	0.2860	0.5299

Table 10: The performance of OTM with and without stochastic mask.