



HAL
open science

Multi-View Self-Attention for Regression Domain Adaptation with Feature Selection

Mehdi Hennequin, Khalid Benabdeslem, Haytham Elghazel, Thomas Ranvier,
Eric Michoux

► **To cite this version:**

Mehdi Hennequin, Khalid Benabdeslem, Haytham Elghazel, Thomas Ranvier, Eric Michoux. Multi-View Self-Attention for Regression Domain Adaptation with Feature Selection. 29th International Conference on Neural Information Processing, Nov 2022, New Delhi, India. pp.177-188, 10.1007/978-3-031-30105-6_15 . hal-04230643

HAL Id: hal-04230643


<https://hal.science/hal-04230643v1>

Submitted on 6 Oct 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Multi-View Self-Attention for Regression Domain Adaptation with Feature Selection

Mehdi Hennequin ^{1,2}[0000-0001-8074-6520], Khalid Benabdeslem², Haytham Elghazel², Thomas Ranvier²[0000-0001-9250-9530], and Eric Michoux¹

¹ Galilé Group, 28 Bd de la République, 71100 Chalon-sur-Saône, France
`m.hennequin@fondation.galile.fr`

² Université Lyon 1, LIRIS,
UMR CNRS 5205, F-69622, France
`{khalid.benabdeslem,haytham.elghazel,thomas.ranvier}@univ-lyon1.fr`

Abstract. In this paper, we address the problem of unsupervised domain adaptation in a regression setting considering that source data have different representations (multiple views). In this work, we investigate an original method which takes advantage of different representations using a discrepancy distance while using attention-based neural networks mechanism to estimate feature importance in domain adaptation. For this purpose, we will begin by introducing a novel formulation of the optimization objective. Then, we will develop an adversarial network domain adaptation algorithm adjusting weights given to each feature, ensuring that those related to the target receive higher weights. Finally, we will evaluate our method on public dataset and compare it to other domain adaptation baselines to demonstrate the improvement for regression tasks.

Keywords: Domain Adaptation · Feature Selection · Multi-view · Regression.

1 Introduction

In most predictive maintenance problems, data are collected from various production lines, assembly lines, or are captured by different devices. Those industrial processes there define several domains each with different distribution. In this context, an algorithm trained for predictive maintenance for a specific domain (referred as source domain) cannot be correctly generalized to another domain (referred as target domain). Therefore, it is common practice to retrain the predictive maintenance models. However, this retraining leads to delayed forecast actions until enough data are available for accurate prediction. To address this issue, predictive models, trained with a specific domain, have to adapt to data with different distributions and limited or non-existing fault information. In machine learning, this situation is often referred to as *domain adaptation* or *covariate shift* [28]. In general, domain adaptation methods attempt to solve the learning problem when the main learning task is the same but the domains have different feature spaces or different marginal conditional probabilities [22,35,25].

On the other hand, data can be represented by several independent sets of features. For instance, in the example of the aforementioned predictive maintenance, data are collected from diverse sensors and exhibit heterogeneous properties. Thus, data from different sensors can be naturally partitioned into independent groups [37]. Each group is referred to as a particular view. Multi-view learning [37,15] aims to improve predictors by taking advantage of the redundancy and consistency between these multiple views.

In the domain adaptation context, views are generally concatenated into one single view to adapt to the learning task. However, this concatenation might cause negative transfer [39], (*i.e.* introduce source domain data/knowledge undesirably) because each view has a specific statistical property. This will result in a decreased learning performance in the target domain. Furthermore, the risk of negative transfer might also come from one or several features being prejudicial to adaptation [35]. It is particularly true with adversarial methods trying to match source and target domains. Therefore, to avoid negative transfer we want to find a way to give high weights to features most related to the target domain. We can find little research on multi-view domain adaptation [38,36] where considerable attention has been given on the classification problem, while regression task and selection features remains largely under-studied.

In this paper, we propose a novel approach for multi-view domain adaptation using self-attention for regression tasks. This work makes two main contributions: first, we propose to extend the measure between distributions Source-guided Discrepancy [13] to multi-views learning concept, and we also adapt this measure to Adversarial method. The second main contribution is the introduction of self-attention to select important features to avoid negative transfer. We conduct experiments on real-world datasets and improve on state-of-the-art results for multi-view adversarial domain adaptation for regression.

2 Learning scenario

This section introduces the definitions and concepts needed for the following sections. For the most part, we follow the definitions and notations of Cortes and Mohri [5]. Let $\mathcal{X} \in \mathbb{R}^p$ and $\mathcal{Y} \in \mathbb{R}$, denote respectively input and output spaces. We define a domain as a pair formed by a distribution over \mathcal{X} and a target labeling function mapping from \mathcal{X} to \mathcal{Y} . Throughout the paper, (Q, f_Q) denotes the source domain and (P, f_P) the target domain with Q the source and P the target distribution over \mathcal{X} and with $f_Q, f_P : \mathcal{X} \rightarrow \mathcal{Y}$ the source and target labeling functions, respectively. In the scenario of *multi-view domain adaptation* the learning algorithm receives a labeled sample \mathcal{S} of m points from the source domain, and the data instances can be represented in M different views. More formally, for $v \in \{1, \dots, M\}$, $\mathcal{S}_v = \{(x_1^{(v)}, y_1^{(v)}), \dots, (x_m^{(v)}, y_m^{(v)})\} \in (\mathcal{X} \times \mathcal{Y})^m$ where $S_{x_v} = \{x_1^{(v)}, \dots, x_m^{(v)}\}$ is supposed to be drawn i.i.d. according to distribution Q and $y_i = f_Q(x_i)$ for all $i \in [1, m]$. In the same way, we define unlabeled samples from the target domain, $\mathcal{T} = \{x'_1, \dots, x'_n\} \in \mathcal{X}^n$ where $\mathcal{T}_x = \{x'_1, \dots, x'_n\}$ is assumed drawn iid according to P and $y_i = f_P(x'_i)$ for all $i \in [1, n]$. We denote by

\hat{Q} and \hat{P} the empirical distributions of the respective samples \mathcal{S}_x and \mathcal{T}_x . We consider in the following that the covariate shift assumption holds, i.e. $f = f_Q = f_P$.

We also consider a loss function $L : \mathcal{Y} \times \mathcal{Y} \rightarrow \mathbb{R}_+$ jointly convex in its two arguments. The L_p loss functions commonly used in regression and defined by $L_p(y, y') = |y - y'|^p$ for $p > 1$ are special instances of this definition. We define a hypothesis class \mathcal{H} of hypotheses $h : X \rightarrow Y$. For any two functions $h, h' : X \rightarrow Y$ and any distribution D over X , we denote by $L_D(h, h')$, the expected loss of $h(x)$ and $h'(x) : L_D(h, h') = \mathbb{E}_{x \sim D}[L(h(x), h'(x))]$.

Objectif. The goal of Domain Adaptation is to minimize the target risk $\mathcal{L}_P(h, f_P) = \mathbb{E}_{x \sim P}[L(h(x), f_P(x))]$. In unsupervised domain adaptation, no label is available in the target task and we cannot directly estimate f_P . Consequently we want to leverage the information about the labels in the source domains f_Q to adapt to the target domain.

3 Adversarial algorithm for Multi-view Domain Adaptation

3.1 Source-guided Discrepancy (S-disc)

The Source-guided Discrepancy (S-disc) introduced in [13], is defined as the maximal difference between source and target risk over a set of hypotheses. We recall below its definition.

Definition 1. (*Source-guided discrepancy*). Let \mathcal{H} be a hypothesis class and $h, h^* \in \mathcal{H}$. S-disc between two distributions Q and P is defined as:

$$\varsigma_{\mathcal{H}}^l(P, Q) = \max_{h \in \mathcal{H}} |\mathcal{L}_P(h, h_S^*) - \mathcal{L}_Q(h, h_S^*)|. \quad (1)$$

where $h_Q^* = \arg \min_{h \in \mathcal{H}} \mathcal{L}_Q(h, f_Q)$ in the source domain. Here, note that the risk minimizer h_Q^* is not necessarily equal to the labeling function f_Q as we consider a restricted hypothesis class. S-disc offers several advantages compared to existing discrepancy measures [13], nevertheless in the context of multi-view learning S-disc is not adapted. Consider M hypotheses, $h_v^* \in \mathcal{H}_v$, for $v \in \{1 \dots M\}$ (for reasons of simplification we abbreviate $h_1 \in \mathcal{H}_1, \dots, h_M \in \mathcal{H}_M$ with $h_v \in \mathcal{H}_v$), such as $h_v^* = \arg \min_{h_v \in \mathcal{H}_v} \mathcal{L}_{Q_v}(h_v, f_{Q_v})$, with $\bigcup_{v \in M} (Q_v, f_{Q_v}) \subseteq (Q, f_Q)$, where the pair (Q_v, f_{Q_v}) is the v^{th} subset of (Q, f_Q) (we note that $f_{Q_v} = f_Q$). In our context a view is considered as a subset of source domain. In this case, it is difficult to choose the appropriate predictor h_v^* to measure the difference between the two domains with S-disc. To overcome this problem we define a novel discrepancy measure Multi-Views-guided Discrepancy (MV-Disc):

Definition 2. (*Multi-Views-guided Discrepancy*) For any $h_v^* \in \mathcal{H}_v$:

$$MV\text{-Disc}(P, Q) = \max_{h \in \mathcal{H}} \left| \frac{1}{M} \sum_{v=1}^M \mathcal{L}_P(h, h_v^*) - \frac{1}{M} \sum_{v=1}^M \mathcal{L}_{Q_v}(h, h_v^*) \right| \quad (2)$$

3.2 Propositional Self-Attention Feature Importance

In this section we explore the use of attention-based neural networks mechanism for estimating feature importance in domain adaptation. This section is inspired from the seminal works on the attention mechanism [3,34,29]. We took inspiration from [29] regarding feature importance, with a different implementation of the attention mechanism, we defined it as follows:

$$\Omega(X) = \frac{1}{k} \bigoplus_k \left[\text{softmax}(f(q^k(W^k X + b^k))) \right] \quad (3)$$

Input vectors $X \in \mathcal{X}$ are first used as input to a softmax-activated layer containing the number of neurons equal to the number of features p , where the softmax function applied to the j_i -th element of a weight vector v is defined as:

$$\text{softmax}(v_{j_i}) = \frac{\exp(v_{j_i})}{\sum_{j=1}^p \exp(v_j)} \quad (4)$$

where $v \in \mathbb{R}^p$. Note that k represents the number of attention heads distinct matrices representing relations between the input features. The \otimes sign corresponds to the Hadamard product, the \oplus refers to the Hadamard summation across individual heads and f corresponds to the activation function. For the activation function f we use a *tanh* as proposed in [1]. We extend the idea of integrating a weight vector q following the attention layer as proposed in [24]. The proposed architecture maintains a bijection between the set of features p and the set of weights in W , thereby the weights in the head can be understood as relations between features [29]. Ω is a mapping from the feature space to the space of non-negative real values, i.e. $\Omega : p \rightarrow \mathbb{R}_0^+$, to obtain the pondered features we multiply the output of Ω by the input space X , and we define $\Omega_R = \Omega(X) \otimes X$.

3.3 Propositional Algorithm

A min-max problem. Since the Multi-Views-guided Discrepancy is defined as a maximum on a functional space, we propose to use adversarial training to align domains. We introduce a feature extractor called generator $G : \mathcal{X} \rightarrow \mathcal{Z}$, typically a neural network parametrized by ϕ . The generator aims to produce a latent space \mathcal{Z} where domains are not distinguished by any predictor $h \in \mathcal{H}_{\mathcal{Z}}$, such as $\mathcal{H}_{\mathcal{Z}} : \mathcal{Z} \rightarrow \mathcal{Y}$. Using the proposed attention mechanism Ω_R and the definition of MV-Disc, we formulate the following objective function for our Adversarial Multi-Views Self Attention-guided Discrepancy (AMVSAD). For any $h_v^* \in \mathcal{H}_v$:

$$\begin{aligned} & \min_{\Omega_{R_v} \in \mathcal{H}_v, g_\phi \in \mathcal{H}, \Omega_{R_T} \in \mathcal{H}} \max_{h \in \mathcal{H}} \left| \frac{1}{M} \sum_{v=1}^M \mathcal{L}_P(h \circ G_\phi \circ \Omega_{R_T}, h_v^* \circ G_\phi \circ \Omega_{R_T}) \right. \\ & \left. - \sum_{v=1}^M \beta_v \mathcal{L}_{Q_v}(h \circ G_\phi \circ \Omega_{R_v}, h_v^* \circ G_\phi \circ \Omega_{R_v}) \right| + \lambda_1 \sigma(h) + \lambda_2 \|\beta\|_2 \end{aligned} \quad (5)$$

Where σ is the spectral norm, λ_1 and λ_2 are hyperparameters. We add spectral normalization [20] to control the discriminator and avoid instability during the training. We also propose to attribute weights to each view [26,16], β ensuring that the most related views to the target receive higher weights. Ω_{R_v} is the v^{th} attention mechanism for the v^{th} view and Ω_{R_T} is the target attention mechanism. For any given $h \in \mathcal{H}$, $h_v^* \in \mathcal{H}_v$ the discrepancy term constrains all three representations ϕ_θ , weights β and Ω_R to align domains.

While computing the true solution of this min-max problem is still impossible in practice, we derive an alternate optimization algorithm. Similarly to most other adversarial methods, we sequentially optimize different parameters of our networks according to different objectives. At a given iteration, losses are minimized/maximized sequentially (the general structure of our algorithm is available at the following github repository³):

Step 1 First, we train the predictors and generator on labeled source data with the different views. Our aim is for the v^{th} predictor to predict correctly the v^{th} view to obtain h_v^* . For $v \in \{1, \dots, M\}$:

$$\min_{h_v \in \mathcal{H}_v, \Omega_{R_v} \in \mathcal{H}_v, G_\phi \in \mathcal{H}} \mathcal{L}_{Q_v}(h_v \circ G_\phi \circ \Omega_{R_v}, f_{Q_v}). \quad (6)$$

Step 2 Thus, we update the predictor h as a discriminator to increase the MV-Disc loss for a fixed generator:

$$\begin{aligned} \max_{h \in \mathcal{H}} & \left| \frac{1}{M} \sum_{v=1}^M \mathcal{L}_P(h \circ G_\phi \circ \Omega_{R_T}, h_v^* \circ G_\phi \circ \Omega_{R_T}) \right. \\ & \left. - \sum_{v=1}^M \beta_v \mathcal{L}_{Q_v}(h \circ G_\phi \circ \Omega_{R_v}, h_v^* \circ G_\phi \circ \Omega_{R_v}) \right| + \lambda_1 \sigma(h) \end{aligned} \quad (7)$$

Step 3 We train the generator, the attention mechanism, and β to minimize the MV-Disc loss for fixed predictor h :

$$\begin{aligned} \min_{\Omega_{R_v} \in \mathcal{H}_v, \Omega_{R_T} \in \mathcal{H}, g_\phi \in \mathcal{H}} & \left| \frac{1}{M} \sum_{v=1}^M \mathcal{L}_P(h \circ G_\phi \circ \Omega_{R_T}, h_v^* \circ G_\phi \circ \Omega_{R_T}) \right. \\ & \left. - \sum_{v=1}^M \beta_v \mathcal{L}_{Q_v}(h \circ G_\phi \circ \Omega_{R_v}, h_v^* \circ G_\phi \circ \Omega_{R_v}) \right| + \lambda_2 \|\beta\|_2 \end{aligned} \quad (8)$$

³ <https://github.com/HennequinMehdi/Adversarial-Multi-View-Attention-guided-Discrepancy.git>

4 Related work

Adversarial Domain Adaptation. Adversarial techniques, for Domain Adaptation was introduced in [8]. Based on the $\mathcal{H}\Delta\mathcal{H}$ -divergence [2], authors found a new representation of the input features where source and target instances cannot be distinguished by any discriminative hypothesis. [32,40,27], follow a similar idea. The above mentioned papers give considerable attention to classification setting, while regression task and selection features remains largely under-studied. Nonetheless, the authors in [19,26] propose methods tailored for regression task. Compared to above mentioned methods we add an attention mechanism to assist the generator to find a subspace shared by domains selecting the most relevant features.

Discrepancy Minimization. The present work is in line with discrepancy minimization methods, which were first introduced in [17], and further developed in [4,21,5,13,40,26]. More specifically, our algorithm aims at minimizing the empirical S-disc introduced in [13]. Discrepancy is the key measure of the difference between two distributions in the context of domain adaptation and has several advantages over other common divergence measures such as the l_1 distance. Besides, several generalizations bound for adaptation in terms of discrepancy were proposed [4,21,5,6]. In comparison to others methods, we introduce the concept of subset in the source domain, in this way we can use different views to compare two distribution.

Feature selection Domain Adaptation. Classical feature selection methods [10] are not designed for domain adaptation. For instance, in [14], the authors searched a latent subspace and deploys $l_{2,1}$ -norm to select common features shared by the domains. Another example of feature selection methods in domain adaptation are [33] and [9]. The contribution of the former paper consists in the use of parametric maximum mean discrepancy distance in order to find a weight matrix that allows to identify invariant and shifting features in the original space. The method described in the latter paper proposes a similar idea using optimal transport to find a shared feature representation. The biggest advantage using our method over the above mentioned ones, is the search of domains shared features during the training of the regression task.

5 Experiments

In this section, we evaluate our AMVSAD method. It should be noted that unsupervised Domain Adaptation with multi-view for regression is hard to evaluate as we have no real public database that corresponds entirely to the problem we described in the introduction. Consequently, we build scenarios and, for each one, we will describe the protocol. We report the results of the AMVSAD algorithm compared to other domain adaptation methods. The experiments are

Table 1. Superconductivity experiments MSE

| Expe. | l → ml | l → mh | l → h | ml → l | ml → mh | ml → h | mh → l |
|--------------|---------------|---------------|---------------|---------------|---------------|---------------|---------------|
| WANN | 0.0844 | 0.0469 | 0.0343 | 0.0404 | 0.0544 | 0.0276 | 0.0391 |
| KLIEP | 0.0619 | 0.0418 | 0.0446 | 0.0377 | 0.0400 | 0.0372 | 0.0268 |
| KMM | 0.0667 | 0.0694 | 0.0273 | 0.0513 | 0.0428 | 0.0282 | 0.0342 |
| DANN | 0.0885 | 0.0501 | 0.0333 | 0.0335 | 0.1134 | 0.3368 | 0.0578 |
| ADDA | 0.0450 | 0.0448 | 0.1155 | 0.0340 | 0.0310 | 0.1626 | 0.0478 |
| DeepCORAL | 0.0672 | 0.0431 | 0.0493 | 0.0502 | 0.0553 | 0.0324 | 0.0538 |
| MDD | 0.0691 | 0.0450 | 0.0446 | 0.0395 | 0.0483 | 0.0325 | 0.0343 |
| TrAdaBoostR2 | 0.0627 | 0.0499 | 0.0417 | 0.0480 | 0.0538 | 0.0284 | 0.0410 |
| AHD-MSDA | 0.0801 | 0.0324 | 0.0264 | 0.0679 | 0.0559 | 0.0592 | 0.0259 |
| AMVSAD | 0.0281 | 0.0252 | 0.0275 | 0.0780 | 0.0570 | 0.0772 | 0.0496 |
| Expe. | mh → ml | mh → h | h → l | h → ml | h → mh | Avg. | |
| WANN | 0.0630 | 0.0661 | 0.0300 | 0.0712 | 0.0395 | 0.0497 | |
| KLIEP | 0.0685 | 0.0337 | 0.0273 | 0.0656 | 0.0429 | 0.0440 | |
| KMM | 0.0587 | 0.0955 | 0.0350 | 0.0680 | 0.0410 | 0.0515 | |
| DANN | 0.1052 | 0.0262 | 0.0498 | 0.1235 | 0.0472 | 0.0888 | |
| ADDA | 0.0815 | 0.0256 | 0.0264 | 0.1877 | 0.0322 | 0.0695 | |
| DeepCORAL | 0.0769 | 0.0642 | 0.0586 | 0.0694 | 0.0507 | 0.0559 | |
| MDD | 0.0667 | 0.0499 | 0.0477 | 0.0762 | 0.0578 | 0.0510 | |
| TrAdaBoostR2 | 0.0654 | 0.0744 | 0.0427 | 0.0664 | 0.0466 | 0.0517 | |
| AHD-MSDA | 0.0514 | 0.0386 | 0.0292 | 0.0662 | 0.0325 | 0.0471 | |
| AMVSAD | 0.0204 | 0.0528 | 0.0368 | 0.0206 | 0.0299 | 0.0419 | |

conducted on public dataset. The following competitors are selected to compare the performance of our algorithm:

- Weighting Adversarial Neural Network (WANN) [19] is a semi-supervised domain adaptation method based on the empirical \mathcal{Y} -discrepancy [21]. It is used for regression tasks.
- Discriminative Adversarial Neural Network (DANN) [8] is an unsupervised domain adaptation method. It is used here for regression tasks by considering the mean squared error as task loss instead of the binary cross-entropy proposed in the original algorithm.
- Adversarial Discriminative Domain Adaptation (ADDA) [32] performs a DANN algorithm in two-stage: it first learns a source encoder and a task hypothesis using labeled data and then learns the target encoder with adversarial training.
- Deep Correlation Alignment (DeepCORAL) [31] is an unsupervised domain adaptation method that aligns the second-order statistics of the source and target distributions with a linear transformation.
- Margin Disparity Discrepancy (MDD) [?] is an unsupervised domain adaptation, it learns a new feature representation by minimizing the disparity discrepancy.
- TrAdaBoostR2 [23] is a semi-supervised domain adaptation method for regression tasks. The method is based on a reverse-boosting principle where the

weight of source instances poorly predicted are decreased at each boosting iteration.

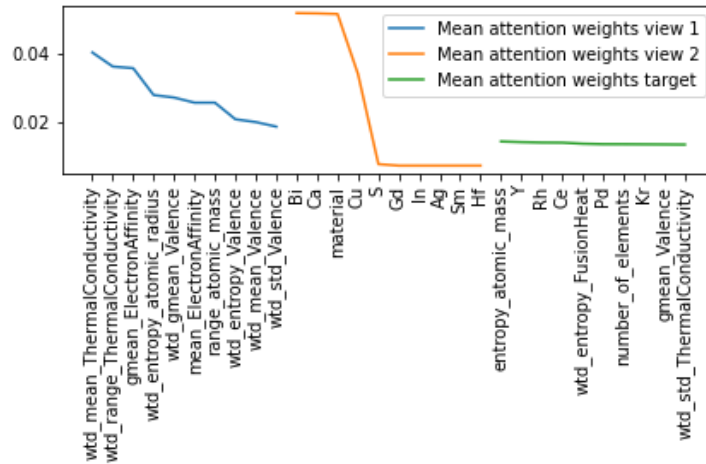
- Kullback-Leibler Importance Estimation Procedure (KLIEP) [30] is a sample bias correction method minimizing the KL-divergence between a reweighted source and target distributions.
- Kernel Mean Matching (KMM) [12] reweights source instances in order to minimize the MMD between domains.
- Adversarial Hypothesis-Discrepancy Multi-Source Domain Adaptation (AHD-MSDA) [26] is a multi-source unsupervised domain adaptation.

We propose here to demonstrate the efficiency of AMVSAD on the UCI dataset superconductivity [7,11]. The goal is to predict the critical temperature of superconductors. This is a common regression problem in the industry, as industrialists are particularly interested in modeling the relationship between a material and its properties. The dataset contains two views: the first view contains 81 features extracted from 21263 superconductors, while the second view contains the chemical formula broken up for all the 21263 superconductors, whose format is binary. We divide this dataset into separate domains as per the setup of [23]. We select an input feature with a moderate correlation factor with the output (0.3). We then sort the set according to this feature and split it into four parts: low (l), middle-low (ml), middle-high (mh), high (h). Each part defines a domain with around 5000 instances. We conduct an experiment for each pair of domains which leads to 12 experiments. For each pair of domains we also randomly select different features from the two views. Therefore, the source domain and the target domain do not have the same features. 10 target labeled instances are used in the training except for our method, AHD-MSDA, which benefit multi-view/multi-source learning method. The other target data are used to compute the results. For the multi-source methods such as AHD-MSDA, we consider a view to be a source, while for the other baseline methods that do not consider multi-source learning, we merge the views. We reported the results in tables, We also report the average MSE over the 12 experiments. For all baseline methods implementation except AHD-MSDA, the python library ADAPT is used [?]. The optimization parameters used in the presented experiments for baseline methods are $lr = 0.01/0.001$, and the loss function is the mean squared error (MSE). The base hypothesis used to learn the task is a neural network with two hidden fully-connected layers of 100 neurons each, ReLU activation functions, weights clipping $C = 1$ and Adam optimizer; 250/350 epochs with a batch size of 128 are performed. Cross-validation is also applied to select best parameters and best scores for each baseline method. Our method and AHD-MSDA have been implemented using the Pytorch library, and the network architecture in table. For more detail, the codes and experiments are available at the following github repository⁴.

⁴ <https://github.com/HennequinMehdi/Adversarial-Multi-View-Attention-guided-Discrepancy.git>

Table 2. Architectures AMVSAD

| Generator | Discriminator |
|--------------------------------------|---|
| Dense(size(features),50 , LeakyReLU) | Spectral_Norm(Dense(25,9 , ReLU)) |
| Dropout(0.1) | Spectral_Norm(Dense(9,1,ReLU)) |
| Dense(50,25,Tanh) | $\lambda_1 = 0.001$ for Spectral Norm |
| Predictors | Optimization parameters |
| Dense(25,9,ReLU) | Adam lr = 0.001, epochs = 100, $\lambda_2 = 0.01$ |
| Dense(9,1,ReLU) | Attention Mechanism Head: Dense(size(features), size(features)) |
| | Attention Mechanism q: Dense(size(features), size(features)) |
| | Num Head and q = 2 |

Fig. 1. Features importance of experiments $l \rightarrow ml$ 

Discussion Overall, we find that our method performs better than state-of-the-art methods in the target domain. However, for a pair of domains our method performs as well or less than some methods. The reason for this is that some methods leverage information from a few target labeled instances during training, thus penalizing our performance. Nevertheless, the advantage of our method is that we can access the attention level associated with each feature averaged over predictions (see Fig. 1). Since we compute feature importance shared by domains, we can visualize the feature’s ranking that contributes to adaptation.

6 Conclusion

In this work, we proposed an adversarial domain adaptation algorithm based on a new discrepancy, MV-Disc, tailored for multi-view regression. We demonstrated the efficiency of our method in real dataset especially with feature importance.

For our future work, we aim to extend our MV-disc to classification problems. In the future we hope to access to more real database in our problematic to perform more exhaustive experiments. We also intend to investigate the self-supervised learning and active learning settings, to try labeling target data with a high degree of confidence.

References

1. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate. CoRR abs/1409.0473 (2015)
2. Ben-David, S., Blitzer, J., Crammer, K., Pereira, F.: Analysis of representations for domain adaptation. In: Schölkopf, B., Platt, J., Hoffman, T. (eds.) *Advances in Neural Information Processing Systems*. vol. 19. MIT Press (2006)
3. Chorowski, J.K., Bahdanau, D., Serdyuk, D., Cho, K., Bengio, Y.: Attention-based models for speech recognition. In: Cortes, C., Lawrence, N., Lee, D., Sugiyama, M., Garnett, R. (eds.) *Advances in Neural Information Processing Systems*. vol. 28. Curran Associates, Inc. (2015)
4. Cortes, C., Mohri, M.: Domain adaptation in regression. In: *Algorithmic Learning Theory - 22nd International Conference, ALT 2011, Espoo, Finland, October 5-7, 2011. Proceedings*. Lecture Notes in Computer Science, vol. 6925, pp. 308–323. Springer (2011)
5. Cortes, C., Mohri, M.: Domain adaptation and sample bias correction theory and algorithm for regression. *Theoretical Computer Science* 519 (2014)
6. Cortes, C., Mohri, M., Medina, A.M.: Adaptation based on generalized discrepancy. *Journal of Machine Learning Research* 20(1), 1–30 (2019)
7. Dua, D., Graff, C.: UCI machine learning repository (2017), <http://archive.ics.uci.edu/ml>
8. Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., Marchand, M., Lempitsky, V.S.: Domain-adversarial training of neural networks. In: *J. Mach. Learn. Res.* (2016)
9. Gautheron, L., Redko, I., Lartizien, C.: Feature selection for unsupervised domain adaptation using optimal transport. CoRR abs/1806.10861 (2018)
10. Guyon, I., Elisseeff, A.: An introduction to variable and feature selection. *J. Mach. Learn. Res.* 3 (2003): 1157–1182.
11. Hamidieh, K.: A data-driven statistical model for predicting the critical temperature of a superconductor. *Computational Materials Science* 154, 346–354 (2018)
12. Huang, J., Gretton, A., Borgwardt, K., Schölkopf, B., Smola, A.: Correcting sample selection bias by unlabeled data. In: Schölkopf, B., Platt, J., Hoffman, T. (eds.) *Advances in Neural Information Processing Systems*. vol. 19. MIT Press (2007)
13. Kuroki, S., Charoenphakdee, N., Bao, H., Honda, J., Sato, I., Sugiyama, M.: Un-supervised domain adaptation based on source-guided discrepancy (2018)
14. Li, J., Zhao, J., Lu, K.: Joint feature selection and structure preservation for domain adaptation. In: *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*. p. 1697–1703. IJCAI'16, AAAI Press (2016)
15. Li, Y., Yang, M., Zhang, Z.: Multi-view representation learning: A survey from shallow methods to deep methods. CoRR abs/1610.01206 (2016)
16. Mansour, Y., Mohri, M., Rostamizadeh, A.: Domain adaptation with multiple sources. In: *Advances in Neural Information Processing Systems 21, Proceedings*

- of the Twenty-Second Annual Conference on Neural Information Processing Systems, Vancouver, British Columbia, Canada, December 8-11, 2008. pp. 1041–1048. Curran Associates, Inc. (2008)
17. Mansour, Y., Mohri, M., Rostamizadeh, A.: Domain adaptation: Learning bounds and algorithms. In: COLT 2009 - The 22nd Conference on Learning Theory, Montreal, Quebec, Canada, June 18-21, 2009 (2009)
 18. de Mathelin, A., Deheeger, F., Richard, G., Mougeot, M., Vayatis, N.: ADAPT : Awesome domain adaptation python toolbox. CoRR abs/2107.03049 (2021)
 19. de Mathelin, A., Richard, G., Mougeot, M., Vayatis, N.: Adversarial weighting for domain adaptation in regression. CoRR abs/2006.08251 (2020)
 20. Miyato, T., Kataoka, T., Koyama, M., Yoshida, Y.: Spectral normalization for generative adversarial networks. CoRR abs/1802.05957 (2018)
 21. Mohri, M., Muñoz Medina, A.: New analysis and algorithm for learning with drifting distributions. In: Proceedings of the 23rd International Conference on Algorithmic Learning Theory. p. 124–138. ALT'12, Springer-Verlag, Berlin, Heidelberg (2012)
 22. Pan, S.J., Yang, Q.: A survey on transfer learning. IEEE Transactions on Knowledge and Data Engineering 22(10), 1345–1359 (2010)
 23. Pardoe, D., Stone, P.: Boosting for regression transfer. In: Proceedings of the 27th International Conference on International Conference on Machine Learning. p. 863–870. ICML'10, Omnipress, Madison, WI, USA (2010)
 24. Ranvier, T., Benabdeslem, K., Bourhis, K., Canitia, B.: Deep multi-view learning for tire recommendation. In: 2021 International Joint Conference on Neural Networks (IJCNN). pp. 1–8 (2021).
 25. Redko, I., Morvant, E., Habrard, A., Sebban, M., Bennani, Y.: A survey on domain adaptation theory. CoRR abs/2004.11829 (2020)
 26. Richard, G., de Mathelin, A., Hébrail, G., Mougeot, M., Vayatis, N.: Unsupervised multi-source domain adaptation for regression. In: Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2020, Ghent, Belgium, September 14-18, 2020, Proceedings, Part I. Lecture Notes in Computer Science, vol. 12457, pp. 395–411. Springer (2020)
 27. Saito, K., Watanabe, K., Ushiku, Y., Harada, T.: Maximum classifier discrepancy for unsupervised domain adaptation. In: 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018. pp. 3723–3732. IEEE Computer Society (2018)
 28. Shimodaira, H.: Improving predictive inference under covariate shift by weighting the log-likelihood function. Journal of Statistical Planning and Inference 90(2), 227–244 (Oct 2000)
 29. Skrlj, B., Dzeroski, S., Lavrac, N., Petkovic, M.: Feature importance estimation with self-attention networks. CoRR abs/2002.04464 (2020)
 30. Sugiyama, M., Nakajima, S., Kashima, H., Buenau, P., Kawanabe, M.: Direct importance estimation with model selection and its application to covariate shift adaptation. In: Platt, J., Koller, D., Singer, Y., Roweis, S. (eds.) Advances in Neural Information Processing Systems. vol. 20. Curran Associates, Inc. (2008)
 31. Sun, B., Saenko, K.: Deep coral: Correlation alignment for deep domain adaptation. In: Hua, G., Jégou, H. (eds.) Computer Vision – ECCV 2016 Workshops. pp. 443–450. Springer International Publishing, Cham (2016)
 32. Tzeng, E., Hoffman, J., Saenko, K., Darrell, T.: Adversarial discriminative domain adaptation. CoRR abs/1702.05464 (2017)

33. Uguroglu, S., Carbonell, J.: Feature selection for transfer learning. In: Gunopulos, D., Hofmann, T., Malerba, D., Vazirgiannis, M. (eds.) *Machine Learning and Knowledge Discovery in Databases*. pp. 430–442. Springer Berlin, Heidelberg (2011)
34. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L.u., Polosukhin, I.: Attention is all you need. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (eds.) *Advances in Neural Information Processing Systems*. vol. 30. Curran Associates, Inc. (2017)
35. Wilson, G., Cook, D.J.: A survey of unsupervised deep domain adaptation. *ACM Trans. Intell. Syst. Technol.* 11(5) (Jul 2020)
36. Xia, Y., Yang, D., Yu, Z., Liu, F., Cai, J., Yu, L., Zhu, Z., Xu, D., Yuille, A.L., Roth, H.R.: Uncertainty-aware multi-view co-training for semi-supervised medical image segmentation and domain adaptation. *Medical image analysis* 65, 101766 (2020)
37. Xu, C., Tao, D., Xu, C.: A survey on multi-view learning. *CoRR* abs/1304.5634 (2013)
38. Yang, P., Gao, W., Tan, Q., Wong, K.F.: Information-theoretic multi-view domain adaptation. In: *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. pp. 270–274. Association for Computational Linguistics, Jeju Island, Korea (Jul 2012)
39. Zhang, W., Deng, L., Wu, D.: Overcoming negative transfer: A survey. *CoRR* abs/2009.00909 (2020)
40. Zhang, Y., Liu, T., Long, M., Jordan, M.: Bridging theory and algorithm for domain adaptation. In: Chaudhuri, K., Salakhutdinov, R. (eds.) *Proceedings of the 36th International Conference on Machine Learning*. *Proceedings of Machine Learning Research*, vol. 97, pp. 7404–7413. PMLR (09–15 Jun 2019)