Disjoint Label Space Transfer Learning with Common Factorised Space

Citation for published version:

Link:
Link to publication record in Edinburgh Research Explorer

Document Version:
Peer reviewed version

Published In:
Proceedings of the Thirty-Third AAAI Conference on Artificial Intelligence (AAAI-19)
Disjoint Label Space Transfer Learning with Common Factorised Space

Xiaobin Chang\textsuperscript{1}, Yongxin Yang\textsuperscript{2}, Tao Xiang\textsuperscript{1}, Timothy M. Hospedales\textsuperscript{2}

Queen Mary University of London\textsuperscript{1}, The University of Edinburgh\textsuperscript{2}

x.chang@qmul.ac.uk yongxin.yang@ed.ac.uk t.xiang@qmul.ac.uk t.hospedales@ed.ac.uk

\begin{abstract}
In this paper, a unified approach is presented to transfer learning that addresses several source and target domain label-space and annotation assumptions with a single model. It is particularly effective in handling a challenging case, where source and target label-spaces are disjoint, and outperforms alternatives in both unsupervised and semi-supervised settings. The key ingredient is a common representation termed Common Factorised Space. It is shared between source and target domains, and trained with an unsupervised factorisation loss and a graph-based loss. With a wide range of experiments, we demonstrate the flexibility, relevance and efficacy of our method, both in the challenging cases with disjoint label spaces, and in the more conventional cases such as unsupervised domain adaptation, where the source and target domains share the same label-sets.
\end{abstract}

\section{Introduction}

Deep learning methods are now widely used in diverse applications. However, their efficacy is largely contingent on a large amount of labelled data in the target task and domain of interest. This issue continues to motivate intense interest in cross-task and cross-domain knowledge transfer. A wide range of transfer learning settings are considered which differ in whether the label spaces of source and target domains are overlapped (i.e., aligned or disjoint), as well as the amount of supervision/labelled training samples available in the target domain (see Figure 1). The standard practice of fine-tuning (Yosinski et al. 2014) treats a pre-trained source model as a good initialisation for training a target problem model. It is adopted when the label spaces of both domains are either aligned or disjoint, but always requires a significant amount of labelled data from the target, albeit less than learning from scratch. Another popular problem is the unsupervised domain adaptation (UDA), where knowledge is transferred from a labelled source domain to an unlabelled target domain (Tzeng et al. 2017; Ganin et al. 2016; Cao, Long, and Wang 2018). UDA makes the simplifying assumption that the label space of source and target domains are the same, and focuses on narrowing the distribution gap between source and target domains without any labelled samples from the target.

An important but less-studied transfer learning problem setting is one where the source and target domains are with disjoint label spaces, recently highlighted by (Luo et al. 2017). In these problems, which we term Disjoint Label Space Transfer Learning (DLSTL), there are both a domain shift between source and target, as well as a new set of target classes to recognise with few (semi-supervised case) or no labelled (unsupervised case) sample per category. Thus, two main challenges exist simultaneously. On one hand, there is few or no target label to drive the adaptation. On the other hand, no clear path is provided to transfer source supervision to target domain due to the disjoint label spaces. As an example, consider object recognition in two cameras (domains) where the object categories (label-space) are different in each camera, and one source camera has dense labels, while the target camera has data but few or no labels. The traditional fine-tuning (Yosinski et al. 2014) and multi-domain training (Rebuffi, Bilen, and Vedaldi 2017) can address the supervised (few label) DLSTL variant, but break down if the labels are very few, and cannot exploit unlabelled data in the target camera, i.e., semi-supervised learning. Meanwhile UDA approaches (Ganin et al. 2016) based on distribution alignment are ineffective since the label-spaces are disjoint and feature distributions thus should not be indistinguishable. One approach that has the potential to handle DLSTL under both unsupervised and semi-supervised settings is based on modelling attributes, which can serve as a bridge across domains for better transferring the discriminative power (Chen et al. 2015; Gebru, Hoffman, and Li 2017;
related methods alleviate the domain gap by either minimising distribution matching (Tzeng et al. 2017; Ganin et al. 2016). Once the domain gap is eliminated, a classifier trained on another task can be used to transfer knowledge from the source domain to the target. However, this method is not effective for very sparsely supervised DLSTL, and it is not applicable to unsupervised DLSTL. Moreover, vanilla TL does not exploit available unlabelled samples for the target problem (i.e., semi-supervised TL). The most related method to ours is Luo et al. (2017) which exploits both unlabelled and few labelled data. CFSM achieves state-of-the-art results on both unsupervised and semi-supervised DLSTL problems and performs competitively in standard UDA. The effectiveness and flexibility of the proposed model on transfer learning problems is thus demonstrated.

Related Work

Transfer Learning

Transfer learning (TL) aims to transfer knowledge from one domain/task to improve performance on another (Pan, Yang, and others 2010). The most widely used TL technique for deep networks is fine-tuning (Yosinski et al. 2014; Chen et al. 2018; Ren et al. 2015). Instead of training a target network from scratch, its weights are initialised by a pretrained model from another task such as ImageNet (Deng et al. 2009) classification. While fine-tuning reduces label requirement compared to learning the target problem from scratch, it is prone to over-fitting if target labels are very few (Yosinski et al. 2014). Therefore, it is ineffective for very sparsely supervised DLSTL, and it is not applicable to unsupervised DLSTL. Moreover, vanilla TL does not exploit available unlabelled samples for the target problem (i.e., semi-supervised TL). The most related method to ours is Luo et al. (2017) which does exploit both unlabelled and few labelled data, i.e., semi-supervised DLSTL. However, like other TL methods, it does not generalise to the unsupervised DLSTL setting where no target annotations are available.

Another popular setting, unsupervised domain adaptation (UDA) focuses on transferring the source supervision to the unlabelled target domain in order to obtain a model that performs well on the latter data. The typical assumption of UDA is that both domains share the same label space. Existing methods alleviate the domain gap by either minimising the distribution discrepancy (Cao, Long, and Wang 2018; Sun and Saenko 2016) or making the dataset representations indistinguishable by adversarial learning (Tzeng et al. 2017; Ganin et al. 2016). Once the domain gap is eliminated, a classifier trained on source-domain labels can be applied to the target data directly. However, distribution matching is inappropriate in the disjoint label space setting. Open set domain adaptation (OSDA) (Busto and Gall 2017) generalises
UDA by allowing target domain to have some novel categories in addition to the shared ones. It focuses on identifying shared categories and aligning those. DLSTL is a more general problem setting than OSDA, since there is no assumption of any shared categories. Related to our approach that exploits a common factorisation space to discover shared latent factors/attributes, semantic attributes have been used to improve domain adaptation performance (Su et al. 2016), for example by enabling new types of self-training (Chen et al. 2015; Wang et al. 2018) and consistency losses (Gebru, Hoffman, and Li 2017). However these methods require the attribute definition and annotation, at least in the source domain. In contrast, no expensive attribute annotation is required in our model.

Deep Binary Representation Learning

The use of binary codes for hashing with deep networks goes back to (Salakhutdinov and Hinton 2009). In computer vision, hashing layers were inserted between feature- and classification-layers to provide a hashing code (Lin et al. 2015; Zhu et al. 2016). To produce a binary representation for fast retrieval, a threshold is applied on the sigmoid activated hashing layer (Lin et al. 2015). Our method is similar in working with a near-binary penultimate layer. However there are several key differences: First, our CFS serves a very different purpose to a hash code. We focus on TL to a new domain with new label-space, and the role of our CFS is to provide a representation with which different domains can be more aligned for knowledge transfer, rather than for efficient retrieval. In contrast, existing hashing methods follow the conventional supervised learning paradigm within a single domain. Second, the proposed CFS is only near-binary due to a low-entropy loss, rather than sacrificing representation power for an exactly binary code.

Semi-supervised Learning

Graph-based regularisation is popular for semi-supervised learning (SSL) which uses both labelled and unlabelled data to achieve better performance than learning with labelled data only (Zhu 2006; Belkin, Niyogi, and Sindhwani 2006). In SSL, graph based regularisation is applied to regularise model predictions to respect the feature-space manifold (Yue et al. 2017; Nadler, Srebro, and Zhou 2009; Belkin, Niyogi, and Sindhwani 2006). Moreover, exploiting the graph from lower-level to regularise higher-level features is widely adopted in other scenarios, e.g., unsupervised learning (Jia et al. 2015; Yang et al. 2017). Due to the source→target knowledge transfer, the more “trustworthy” layer in our method is the penultimate CFS layer, as it is closer to the supervision, rather than the feature space layer. Therefore our regularisation is applied to encourage the feature-extractor to learn representations that respect the CFS manifold shared by both domains, i.e., the regularisation direction is opposite to that in existing models.

Entropy loss for unlabelled data is another widely used SSL regulariser (Zhu 2006; Long et al. 2016). It is applied at the classification layer in problems where the unlabelled and labelled data share the same label-space — and reflects the inductive bias that a classification boundary should not cut through the dense unlabelled data regions. Its typical use is on softmax classifier outputs where it encourages a classifier to pick a single label. In contrast we use entropy-loss to solve DLSTL problems by applying it element-wise on our intermediate CFS layer in order to weakly align domains by encouraging them to share a near-binary representation.

Methodology

Definition and notation For Disjoint Label Space Transfer Learning (DLSTL), there is a source (labelled) domain $S$ and a target (unlabelled or partially labelled) domain $T$. The key characteristic of DLSTL is the disjoint label space assumption, i.e., the source $Y_S$ and target $Y_T$ label spaces are potentially disjoint: $Y_S \cap Y_T = \emptyset$. Instances from source/target domains are denoted $X_S$ and $X_T$ respectively. The combined inputs $\{X_S, X_T\}$ are denoted as $X$. To present our model, we stick mainly to the most challenging unsupervised DLSTL setting where target labels are totally absent. The easier cases, e.g., semi-supervised DLSTL and UDA, can then be handled with minor modifications.

Model Architecture

The proposed model architecture consists of three modules, a feature extractor $F = \Psi_{\theta_M}(X)$ that can be any deep neural network and is shared between all domains. This is followed by a fully connected layer and sigmoid activation that goes back to (Salakhutdinov and Hinton 2009). In computer vision, hashing layers were inserted between feature- and classification-layers to provide a hashing code (Lin et al. 2015; Zhu et al. 2016). To produce a binary representation for fast retrieval, a threshold is applied on the sigmoid activa-

Regularised Model Optimisation

The parameters of the proposed CFSM are $\theta := \{\theta_M, \theta_C, \theta_S\}$ including parameters of the feature extractor $\theta_M$, CFS layer $\theta_C$ and source classifier $\theta_S$. The training procedure can be formulated as a maximum-a-posterior (MAP) learning given labelled source $\{X_S, Y_S\}$ and unlabelled target data $X_T$,

$$\hat{\theta} = \arg\max_{\theta} p(\theta|X_S, Y_S, X_T),$$

(1)

where $p(\theta|X_S, Y_S, X_T)$ is the posterior of model parameter $\theta$ given data $X_S, Y_S, X_T$. This can be rewritten as

$$p(\theta|X_S, Y_S, X_T) \propto p(\theta, X_S, Y_S, X_T) \propto p(Y_S|X_S, X_T, \theta)p(\theta|X_S, X_T).$$

(2)

So the optimisation in Eq. 1 is equivalently

$$\hat{\theta} = \arg\max_{\theta} p(Y_S|X_S, \theta)p(\theta|X).$$

(3)

The proposed model can be easily extended to deal with multiple source and target domains.
The first term \( p(Y_S|X_S, \theta) \) in Eq. 3 represents the likelihood of source labels w.r.t. \( \theta \). Optimising this term is a conventional supervised learning task with a loss denoted \( \ell_{\text{sup}}(X_S, Y_S; \theta) \).

The second term \( p(\theta|X) \) in Eq. 3 is a prior depending on the input data \( X \) of both source and target datasets. From an optimisation perspective, this is the regulariser that will play the key role in solving unsupervised DLSTL problems since it requires no labels. Given the model architecture, it can be further decomposed as:

\[
p(\theta|X) = p(\theta_M, \theta_C|X) = p(\theta_C|\theta_M, X)p(\theta_M|X),
\]

where \( \theta_S \) is excluded since no supervision is used. Specifically, the first term \( p(\theta_C|\theta_M, X) \) serves as the regulariser on the CFS layer while the second term \( p(\theta_M|X) \) regularises the deep feature extractor \( \Phi_{\theta_M} \).

### Low-Entropy Regularisation: Unsupervised Adaptation

We first discuss how to define the prior \( p(\theta_C|\theta_M, X) \) regulariser for CFS layer. The sigmoid activated outputs \( F_C \) from CFS layer \( \Psi_{\theta_C} \) can be interpreted as multi-label predictions on latent factors. The uncertainty measure for label prediction can be defined by using its entropy

\[
h(\theta_C|\theta_M, X) = -\sum_{i=1}^{N} < F_{C,i}, \log(F_{C,i}) >
\]

\[
= -\sum_{i=1}^{N} < \Psi_{\theta_C}(x_i), \log(\Psi_{\theta_C}(x_i)) >
\]

where \( F_{C,i} \) denotes the common factor representation \( \Psi_{\theta_C}(x_i) \) of instance \( x_i \in X \). This is applied on both source and target data, so \( N \) is the number of instances in both datasets. \( \log(\cdot) \) is applied element-wise, and \(< \cdot, \cdot > \) is vector inner product. According to the low-uncertainty criterion (Carlucci et al. 2017), optimising the prior term \( p(\theta_C|\theta_M, X) \) can be achieved by minimising this uncertainty measure. Eq. 5 is thus the regulariser corresponding to the prior \( p(\theta_C|\theta_M, X) \). Specifically, this loss biases the representation \( F_C \) to contain more certain predictions, e.g., closer to 0 or 1 for each discovered latent factor. Therefore, we denote it as unsupervised factorisation loss.

In summary, the low-entropy regulariser on CFS is built upon the assumption that the two domains share a set of latent attributes and that if a source classifier is well adapted to the target, then the presence/absence of these attributes should be certain for each instance. Therefore, it essentially generalises the low-uncertainty principle (widely used in existing unsupervised and semi-supervised learning literature) to the disjoint label space setting.

### Graph Regularisation: Robust Feature Learning

The second prior in Eq. 4 is \( p(\theta_M|X) \) which acts as the regulariser for the feature extractor \( \Phi_{\theta_M} \). The unique property of our setup so far is that the knowledge transfer into the target domain is via the CFS layer; therefore we are interested in ensuring that the feature extractor network extracts features whose similarity structure reflects that of the latent factors in the CFS layer. Unlike conventional graph Laplacian losses that regularise higher-level features with a graph built on lower-level features (Belkin, Niyogi, and Sindhwani 2006; Zhu 2006), we do the reverse and regularise the feature extractor \( \Phi_{\theta_M} \) to reflect the similarity structure in \( F_C \). This is particularly important for applications where the target problem is retrieval, because we use deep features \( F = \Phi_{\theta_M}(\cdot) \) as an image representation.

The proposed graph loss is expressed as

\[
\text{Tr}(F^T \Delta_{F_C} F),
\]

where \( \Delta_{F_C} \) is the graph Laplacian (Cai et al. 2011) built on the common space features \( F_C \).

### Summary

We unify the proposed model architecture \( \theta := \{ \theta_M, \theta_C, \theta_D_S \} \) with source \( \{ X_S, Y_S \} \) and target \( \{ X_T \} \) data for unsupervised DLSTL problems from an maximum-a-posterior (MAP) perspective. This decomposes into a standard supervised term \( p(Y_S|X_S, \theta) \) (source data only) and data-driven priors for the CFS layer and feature extraction module. They correspond to supervised loss \( \ell_{\text{sup}}(X_S, Y_S; \theta) \), unsupervised factorisation loss (Eq. 5) and the graph loss (Eq. 6) respectively. Taking all terms into account, the final optimisation objective of Eq. 3 is

\[
L(\theta) = \ell_{\text{sup}}(X_S, Y_S; \theta) + \beta_M \text{Tr}(F^T \Delta_{F_C} F) - \beta_C \frac{1}{N} \sum_{i=1}^{N} < F_{C,i}, \log(F_{C,i}) >.
\]

where \( \beta_C \) and \( \beta_M \) are balancing hyper-parameters. In order to select \( \beta_C \) and \( \beta_M \), the model is first run by setting all weights to 1; after the first few iterations, we check the values of each loss. We then set the two hyper-parameters to rescale the losses to a similar range so that all three terms contribute approximately equally to the training.

### Mini-batch organisation

Convolutional Neural Networks (CNNs) are usually trained with SGD mini-batch optimisation, but Eq. 7 is expressed in a full-batch fashion. Converting Eq. 7 to mini-batch optimisation is straightforward. However, it is worth mentioning the mini-batch
scheduling: each mini-batch contains samples from both source and target domains. The supervised loss is applied only to source samples with corresponding supervision, the entropy and graph losses are applied to both, and the graph is built per-mini-batch. In this work, the number of source and target samples are equally balanced in a mini-batch.

Experiments

The proposed model is evaluated on progressively more challenging problems. First, we evaluate CFSM on unsupervised domain adaptation (UDA). Second, different DLSTL settings are considered, including semi-supervised DLSTL classification and unsupervised DLSTL retrieval. CFSM handles all these scenarios with minor modifications. The effectiveness of CFSM is demonstrated by its superior performance compared to the existing work. Finally, insight is provided through ablation study and visualization analysis.

Unsupervised Domain Adaptation: SVHN-MNIST

Dataset and Settings We evaluate the UDA setting from (Ganin et al. 2016) where SVHN (Netzer et al. 2011) is the labelled source dataset and MNIST (LeCun et al. 1998) is the unlabelled target. For fair comparison we use an identical feature extractor network to (Luo et al. 2017). Our CFSM is pre-trained on the source dataset with cross-entropy supervision and $d_C = 50$, followed by joint training on source and target with our regularisers as in Eq. 7. Since the label-space is shared in UDA, we also apply entropy loss on the softmax classification of the target (Long et al. 2016). We set $\beta_M = 0.001$ and $\beta_C = 0.01$.

Results We compare our method with two baselines. Source only: Supervised training on the source and directly apply to target data. Joint FT: Model is initialised with source pre-train, and fine-tuning on both domains with supervised loss for source and semi-supervised entropy loss for target. We also compare several deep UDA methods including Gradient Reversal (Ganin et al. 2016), Domain Confusion (Tzeng et al. 2015), ADDA (Tzeng et al. 2017), Label Efficient Transfer (LET) (Luo et al. 2017), Asym. tri-training (Saito, Ushiku, and Harada 2017) and Respara (Rozantsev, Salzmann, and Fua 2018).

As shown in Table 1, CFSM boosts the performance on both baselines with clear margin ($25.5\%$ and $9.3\%$ vs. Source only and Joint FT respectively). Moreover, it is $5.5\%$ higher than LET (Luo et al. 2017), the nearest competitor and only alternative that also addresses the DLSTL setting.

Semi-supervised DLSTL: Digit Recognition

Dataset and Settings We follow the semi-supervised DLSTL recognition experiment of (Luo et al. 2017) where again two digit datasets, SVHN and MNIST, are used. Images of digits $0$ to $4$ from SVHN are fully labelled as source data while images of digits $5$ to $9$ from MNIST are target data. The target dataset has sparse labels ($k$ labels per class) and unlabelled images available. Thus we also add a classifier $\chi_{\theta_M}$ after the CFS layer $\Psi_{\theta_M}$ for the target categories.

The feature extractor architecture $\Phi_{\theta_M}$ is exactly the same as in (Luo et al. 2017) for fair comparison. We pre-train CFSM on source data as initialisation, and then train it with both source and target data using only loss in Eq. 7. We set $d_C = 10$, $\beta_M = \beta_C = 0.01$. The learning rate is $0.001$ and the Adam (Kingma and Ba 2014) Optimiser is used.

Results The results for several degrees of target label sparsity $k = 2, 3, 4, 5$ (corresponding to $10, 15, 20, 25$ labelled samples, or $0.03\%, 0.05\%, 0.066\%, 0.086\%$ of total target training data respectively), are reported in Table 2. Results are averaged over ten random splits as in (Luo et al. 2017). Besides the FT matching nets (Vinyls et al. 2016) and state-of-the-art LET results from (Luo et al. 2017), we run two baselines: Train Target: Training CFSM architecture from scratch with partially labelled target data only, and FT Target: The standard pre-train/fine-tune pipeline, i.e., pre-train on the labelled source, and fine-tune on the labelled target samples only.

As shown in Table 2, the performances of baseline models are significantly lower than LET and the proposed CFSM. The Train Target baseline performs poorly as it is hard to achieve good performance with few target samples and no knowledge transfer from source. The Fine-Tune Target baseline performs poorly as the annotation here is too sparse for effective fine-tuning on the target problem. Fine-Tune matching nets follows the 5-way $(k - 1)$-shot learning with sparsely labelled target data only, but no improvement is shown over the other baselines. Our proposed CFSM consistently outperforms the state-of-the-art LET alternative. For example, under the most challenging setting $(k = 2)$, CFSM is $1.8\%$ higher than LET on mean accuracy and $0.2\%$ lower on standard error.

Unsupervised DLSTL: ReID and SBIR

ReID The person re-identification (ReID) problem is to match person detections across camera views. Annotating person image identities in every camera in a camera network for training supervised models is infeasible. This motivates the topical unsupervised Re-ID problem of adapting a ReID model trained on one dataset with annotation to a new dataset without annotation. Although they are evaluated with retrieval metrics, contemporary Re-ID models are trained using identity prediction (classification) losses. This means that unsupervised Re-ID fits the unsupervised DLSTL setting, as the label-spaces (person identities) are different in different Re-ID datasets, and the target dataset has no labels.

We adopt two highly contested large-scale benchmarks.
for unsupervised person Re-ID: Market (Zheng et al. 2015) and Duke (Zheng, Zheng, and Yang 2017). ImageNet pre-trained Resnet50 (He et al. 2016) is used as the feature extractor \( \Phi_{\theta_M} \). Cross-entropy loss with label smoothing and triplet loss are used for the source domain as supervised learning objectives. We set \( dc = 2048, \beta_M = 2.0, \beta_C = 0.01 \). Adam optimiser is used with learning rate \( 3.5e^{-4} \). We treat each dataset in turn as source/target and perform unsupervised transfer from one to the other. Rank 1 (R1) accuracy and mean Average Precision (mAP) results on the target datasets are used as evaluation metrics.

In Table 3, We show that our method outperforms the state-of-the-art alternatives purpose-designed for unsupervised person Re-ID: UMDL (Peng et al. 2016), PTCGAN (Wei et al. 2018), PUL (Fan, Zheng, and Yang 2017), CAMEL (Yu, Wu, and Zheng 2017), TJ-AIDL (Wang et al. 2018), SPGAN (Deng et al. 2018) and MMFA (Lin et al. 2018). Note that TJ-AIDL and MMFA exploit attribute labels to help alignment and adaptation. The proposed method automatically discovers latent factors with no additional annotation. However, CFSM improves at least 3.0% over TJ-AIDL and MMFA on the R1 accuracy of both settings.

**FG-SBIR** Fine-grained Sketch Based Image Retrieval (SBIR) focuses on matching a sketch with its corresponding photo (Sangkloy et al. 2016). As demonstrated in (Sangkloy et al. 2016), object category labels play an important role in retrieval performance, so existing studies make a closed world assumption, i.e., all testing categories overlap with training categories. However, if deploying SBIR in a real application such as e-commerce (Yu et al. 2016), one would like to train the SBIR system once on some source object categories, and then deploy it to provide sketch-based image retrieval of new categories without annotating new data and re-training for the target object category. Unsupervised adaptation to new categories without sketch-photo pairing labels is therefore another example of the unsupervised DLSTL problem. Comparing to Re-ID, where instances are person images in different camera views, instances in SBIR are either photos or hand-drawn sketches of objects.

There are 125 object classes in the Sketchy dataset (Sangkloy et al. 2016). We randomly split 75 classes as a labelled source domain and use the remaining 50 classes to define an unlabelled target domain with disjoint label space. ImageNet pre-trained Inception-V3 (Szegedy et al. 2016) is used as the feature extractor \( \Phi_{\theta_M} \). Cross-entropy and triplet loss are used for source supervision. We set \( dc = 512, \beta_M = 10^{-3}, \beta_C = 0.1 \). Adam optimiser with learning rate \( 10^{-4} \) is used. As a baseline, Source Only is the direct transfer alternative that uses the same architecture but trains on the source labelled data only, and is applied directly to the target without adaptation. The retrieval performance on unseen classes (tar. cls.) is reported. Results are averaged over 10 random splits. As shown in Table 4, the proposed CFSM improves the retrieval accuracy on unseen cases by 2.48%.

![Table 3: Unsupervised transfer for person Re-ID (%). M2D indicates Market as source dataset and Duke as target, vice versa. Target Dataset Performance is reported.](image)

<table>
<thead>
<tr>
<th>model</th>
<th>M2D</th>
<th>D2M</th>
</tr>
</thead>
<tbody>
<tr>
<td>UMDL CVPR’16</td>
<td>18.5</td>
<td>54.5</td>
</tr>
<tr>
<td>PTGAN CVPR’18</td>
<td>27.4</td>
<td>38.6</td>
</tr>
<tr>
<td>PUL arXiv’17</td>
<td>30.0</td>
<td>45.5</td>
</tr>
<tr>
<td>CAMEL ICCV’17</td>
<td>54.5</td>
<td>26.3</td>
</tr>
<tr>
<td>TJ-AIDL CVPR’18</td>
<td>44.3</td>
<td>58.2</td>
</tr>
<tr>
<td>SPGAN CVPR’18</td>
<td>46.4</td>
<td>57.7</td>
</tr>
<tr>
<td>MMFA BMVC’18</td>
<td>45.3</td>
<td>56.7</td>
</tr>
<tr>
<td>CFSM</td>
<td>49.8</td>
<td>28.3</td>
</tr>
</tbody>
</table>

![Table 4: SBIR: Sketch-photo retrieval results (%). Averaged Rank 1 accuracy and standard error.](image)

<table>
<thead>
<tr>
<th></th>
<th>Source only</th>
<th>CFSM</th>
</tr>
</thead>
<tbody>
<tr>
<td>tar. cls.</td>
<td>23.74 ± 0.24</td>
<td>26.22 ± 0.25</td>
</tr>
</tbody>
</table>

**Further Analysis**

**Ablation study** Unsupervised person Re-ID is chosen as the main benchmark for an ablation study. Firstly because it is a challenging and realistic large-scale problem in the unsupervised DLSTL setting, and secondly because it provides a bidirectional evaluation for more comprehensive analysis.

The following ablated variants are proposed and compared with the full CFSM. Source Only: The proposed architecture is learned with source data and supervised losses only. Source+Regs: The regularisers, unsupervised factorisation and graph losses can be added with source dataset only. CFSM—Graph: Our method without the proposed graph loss. CFSM+ClassicGraph: Replacing our proposed graph loss with a conventional Laplacian graph (i.e., graphs constructed in lower-level feature space extracted by \( \Phi_{\theta_M} \) to regularise the proposed CFS). AE: Other regularisers such as feature reconstruction as in autoencoder (AE) is used to provide the prior term \( p(\theta | X) \). We reconstruct the deep features \( F \) using the outputs of CFS layer as hidden representations.
In this case both source and target data are used and the reconstruction error provides the regularisation loss.

The results are shown in Table 5. Firstly, by comparing the variants that use source data only (Source Only and Source+Regs) with the joint training methods, we find they are consistently inferior. This illustrates that it is crucial to leverage target domain data for adaptation. Secondly, CFSM and its variants consistently achieve better results than AE, illustrating that our unsupervised factorisation loss and graph losses provide better regularisation for cross-domain/cross-task adaptation. The effectiveness of our graph loss is illustrated by two comparisons: (1) CFSM–Graph is worse than CFSM, showing the contribution of the graph loss; and (2) replacing our graph loss with the conventional Laplacian graph loss (CFSM+ClassicGraph) shows worse results than ours, justifying our choice of regularisation direction. Finally, we note that applying our regularisers to the source only (Source+Regs) still improves the performance slightly on target dataset vs Source Only. This shows that training with these regularisers has a small benefit to representation transferability even without adaptation.

**Visualisation analysis** To understand the impact of unsupervised factorisation loss, Figure 3 illustrates the distribution of target CFS activations in the semi-supervised DLSTL setting (SVHN→MNIST). The left plot shows the activations without any such loss, leading to a distribution of moderate predictions peaked around 0.5. In contrast, the right plot shows the activation distribution on the target dataset of CFSM. We can see that our regulariser has indeed induced the target dataset to represent images with a low–entropy near-binary code. We also compare training a source model by adding low-entropy CFS loss, and then applying it to the target data. This leads to a low-entropy representation of the source data, but the middle plot shows that when transferred to the target dataset or adaptation the representation becomes high-entropy. That is, joint training with our losses is crucial to drive the adaptation that allows target dataset to be represented with near-binary latent factor codes.

**Qualitative Analysis** We visualise the discovered latent attributes qualitatively. For each element in \( F_C \), we rank images in both source and target domains by their activation. Person images corresponding to the highest ten values of a specific \( F_C \) are recorded. Figure 4 shows two example factors with images from the source (first row) and target (second row) dataset. We can see that the first example in Figure 4(a) is a latent attribute for the colour ‘red’ cov-erering both people’s bags and clothes. The second example in Figure 4(b) is a higher-level latent attribute that is selective for both females, as well as textured clothes and bag-carrying. Importantly, these factors have become selective for the same latent factors across datasets, although the target dataset has no supervision (i.e., unsupervised DLSTL).

**Conclusion**

We studied a challenging transfer learning setting DLSTL, where the label space between source and target labels are disjoint, and the target dataset has few or no labels. In order to transfer the discriminative cues from the labelled source to the target, we propose a simple yet effective model which uses an unsupervised factorisation loss to discover a common set of discriminative latent factors between source and target datasets. And to improve feature learning for subsequent tasks such as retrieval, a novel graph-based loss is further proposed. Our method is both the first solution to the unsupervised DLSTL, and also uniquely provides a single framework that is effective at both unsupervised and semi-supervised DLSTL as well as the standard UDA.

**Acknowledgements** This work was supported by the EPSRC grant EP/R026173.
References

Belkin, M.; Niyogi, P.; and Sindhwani, V. 2006. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *JMLR.*

Busto, P. P., and Gall, J. 2017. Open set domain adaptation. In *ICCV.*

Cai, D.; He, X.; Han, J.; and Huang, T. S. 2011. Graph regularized nonnegative matrix factorization for data representation. *TPAMI.*

Cao, Y.; Long, M.; and Wang, J. 2018. Unsupervised domain adaptation with distribution matching machines. *AAAI.*


Chen, Q.; Huang, J.; Feris, R.; Brown, L. M.; Dong, J.; and Yan, S. 2015. Deep domain adaptation for describing people based on fine-grained clothing attributes. *CVPR.*


Deng, J.; Dong, W.; Socher, R.; Li, L.-J.; Li, K.; and Fei-Fei, L. 2009. Imagenet: A large-scale hierarchical image database. *CVPR.*


Nadler, B.; Srebro, N.; and Zhou, X. 2009. Semi-supervised learning with the graph laplacian: The limit of infinite unlabeled data. *NIPS.*


Pan, S. J.; Yang, Q.; et al. 2010. A survey on transfer learning. *TKDE.*


Rastegari, M.; Farhadi, A.; and Forsyth, D. 2012. Attribute discovery via predictable discriminative binary codes. *ECCV.*

Rebuffi, S.-A.; Bilen, H.; and Vedaldi, A. 2017. Learning multiple visual domains with residual adapters. *NIPS.*


Rozantsev, A.; Salzmann, M.; and Fua, P. 2018. Residual parameter transfer for deep domain adaptation. *CVPR.*

Saito, K.; Ushiku, Y.; and Harada, T. 2017. Asymmetric tri-training for unsupervised domain adaptation. *ICML.*


Sangkloy, P.;Burnell, N.; Ham, C.; and Hays, J. 2016. The sketchy database: Learning to retrieve badly drawn bunnies. *SIGGRAPH.*


Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; and Wojna, Z. 2016. Rethinking the inception architecture for computer vision. *CVPR.*


Tzeng, E.; Hoffman, J.; Saenko, K.; and Darrell, T. 2017. Adversarial discriminative domain adaptation. *CVPR.*


Wei, L.; Zhang, S.; Gao, W.; and Tian, Q. 2018. Person transfer gan to bridge domain gap for person re-identification. *CVPR.*


Yosinski, J.; Clune, J.; Bengio, Y.; and Lipson, H. 2014. How transferable are features in deep neural networks? *NIPS.*

Yu, Q.; Liu, F.; Song, Y.-Z.; Xiang, T.; Hospedales, T. M.; and Loy, C. C. 2016. Sketch me that shoe. *CVPR.*


Zheng, Z.; Zheng, L.; and Yang, Y. 2017. Unlabeled samples generated by gan improve the person re-identification baseline in vitro. *ICCV.*
