Semi-supervised Dual Relation Learning for Multi-label Classification

Lichen Wang, Yunyu Liu, Hang Di, Can Qin, Gan Sun, and Yun Fu, *Fellow, IEEE*

***Abstract*—In a real-world scenario, an object could contain multiple tags instead of a single categorical label. To this end, multi-label learning (MLL) emerged. In MLL, the feature distributions are long-tailed and the complex semantic label relation and the long-tailed training samples are the main challenges. Semi-supervised learning is a potential solution. While, existing methods are mainly designed for single class scenario while ignoring the latent label relations. In addition, they cannot well handle the distribution shift commonly existing across source and target domains. To this end, a Semi-supervised Dual Relation Learning (SDRL) framework for multi-label classification is proposed. SDRL utilizes a few labeled samples as well as large scale unlabeled samples in the training stage. It jointly explores the inter-instance feature-level relation and the intra-instance label-level relation even from the unlabeled samples. In our model, a dual-classifier structure is deployed to obtain domain invariant representations. The prediction results from the classifiers are further compared and the most confident predictions are extracted as pseudo labels. A trainable label relation tensor is designed to explicitly explore the pairwise latent label relations and refine the predicted labels. SDRL is able to effectively and efficiently explore the feature-label relation as well as the label-label relation knowledge without any extra semantic knowledge. We evaluated SDRL in general and zero-shot multi-label classification tasks and we concluded that SDRL is superior to other SOTA baselines. Furthermore, extensive ablation studies have been done which reveal the effectiveness of each component in our framework.**

***Index Terms*—Label relation learning, semi-supervised learning, multi-label learning, image retrieval, image annotation.**

I. Introduction

IN real-world application, there could be dozens or even hundreds of semantic descriptions related to a single object. For instance, an image which shows *"A man is walking near a lake on a sunny day"*. The labels including *"Sunny"*, *"Lake"*, *"Man"*, and *"Walking"* are selected as the positive labels.

The uniqueness of Multi-label Learning (MLL) is whether there are multiple labels in a given instance [1]. Several challenges exist in MLL. First, most MLL databases (e.g., AWA [2], CUB [3], and SUN [4]) are small-scale consider creating and labeling a multi-label dataset is considerably costly. Usually, there are tens of positive labels which should be extracted from a large-scale candidate label pool. Some of the candidate labels are subjective labels (e.g., *"Stressful"*), which leads to non-negligible noises. Second, consider the natural characteristics of multi-label, most of the labels follow a long-tailed distribution where some labels are significantly popular than others. For instance, in SUN dataset [4], with a total of 14340 data points, the *"Man-made"* label exists in 8089 samples, while the *"Fire"* label only exists in 73 samples. This phenomenon always causes considerable unbalanced training samples. Third, the semantic connections across labels provide extra and important knowledge. As illustrated in Figure 1, some labels (e.g., *"Sunny"*, *"Rainy"*) have strong connections than others. Effectively utilizing such label relations could considerably improve the performance [5], [6]. Unfortunately, few existing datasets provide such knowledge.

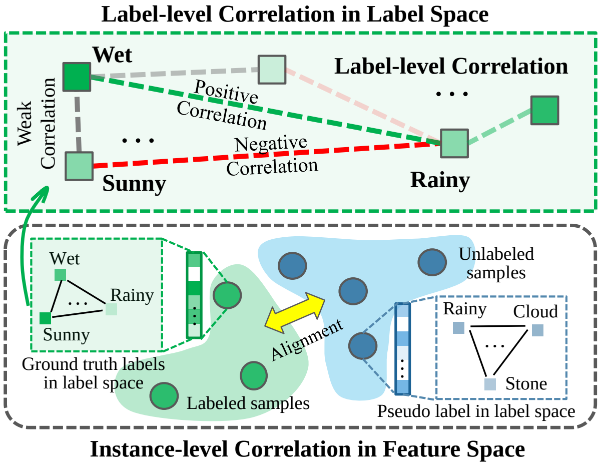


Fig. 1. There are two major difficulties in MLL. First, the labeled (green circle) and unlabeled (blue circle) samples could have different distributions. How to effectively align the distribution is difficult. Second, the latent label-label relations are crucial knowledge for improving the MLL performance. For instance, ***"Sunny"*** and ***"Rainy"*** are almost impossible to show up together (negative). ***"Rainy"*** and ***"Wet"*** are more likely to appear together (positive), and the relation between ***"Wet"*** and ***"Sunny"*** is weak (weak). How to explore this semantic relation knowledge is challenging.

In general, a large-scale training set is a solution. However, collecting such a dataset is expensive. Moreover, building the semantic relation knowledge requires specialized semantic knowledge, and the defined relation map is task-specific which can not be extended to other tasks. Although creating a large-scale dataset is difficult, related and unlabeled data is everywhere and easy to obtain. Therefore, semi-supervised learning [7], [8], [9], [10], [11], [12] are proposed which aim to explore the source domain and further enhance the final performance. Conventional semi-supervised methods mainly explore the data distribution in feature space. [13] designed a pipeline which mutually reinforce the learning from one

Lichen Wang, Yunyu Liu, Hang Di, Can Qin, and Gan Sun are with the Department of Electrical and Computer Engineering, Northeastern University, Boston, USA (Email: wanglichenxj@gmail.com, liu.yuny@northeastern.edu, di.h@northeastern.edu, canqinn@gmail.com, sungan1412@gmail.com).

Yun Fu is with the Department of Electrical and Computer Engineering, and Khoury College of Computer Science, Northeastern University, Boston, USA (Email: yunfu@ece.neu.edu).

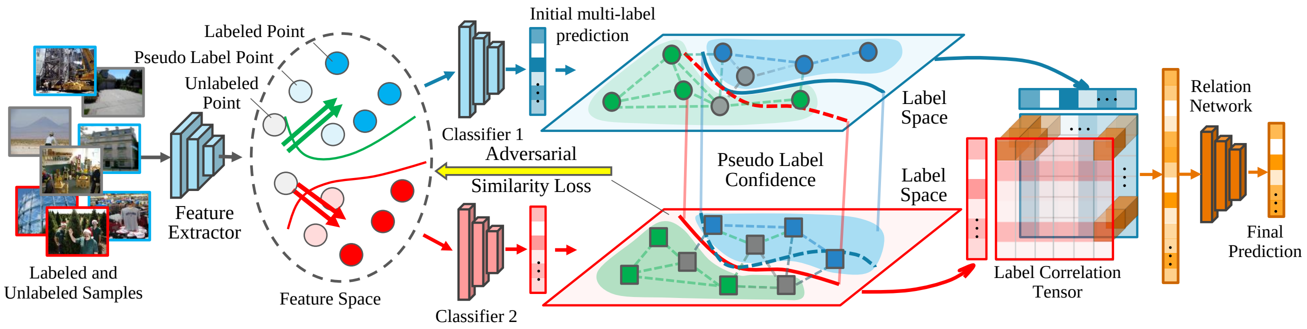


Fig. 2. Illustration of our Semi-supervised Dual Relation Learning (SDRL) framework. There are four networks, a feature extractor $E(\cdot)$, two multi-label classifiers $C_1(\cdot)$, $C_2(\cdot)$, and a label correlation learning network $C_R(\cdot)$. Specifically, $E(\cdot)$ extracts image features from the given labeled and unlabeled samples, then $C_1(\cdot)$ and $C_2(\cdot)$ obtain the initial multi-label prediction results respectively. A specifically designed adversarial learning mechanism is used to justify the representation distributions of labeled and unlabeled samples. In addition, by comparing the prediction results from two classifiers, the most confident predictions are set as pseudo labels and involved in the training strategy. Moreover, a label correlation tensor is proposed to explore the pairwise prediction results from the classifiers. By this way, both the latent label correlations and the confidence of two classifiers will be explored and further refine the prediction results. All the modules are alternatively optimized which fully reveals the latent knowledge from source and target samples to obtain the best performance.

task to other tasks. [14] presented a generalized and flexible graph CNN. [15] proposed a novel multi-view dimensional reduction approach based an on adaptive graph. [16] proposed an enhanced categorical alignment strategy which explores multiple mutually complementary techniques. However, most of these frameworks ignore the distribution gap issues, and deriving the similarity from the original feature space could reduce the learning performance dramatically. Second, most of the works focus on the single-category classification settings, which ignore the label relation knowledge [9], [10].

In this work, our model simultaneously discovers both the feature-label relation and the cross label relation in semi-supervised scenario. It fully utilizes existing samples (especially the unlabeled samples), and further mitigates the distribution shift between unlabeled and labeled samples. A novel Semi-supervised Dual Relation Learning (SDRL) framework is proposed. The framework of SDRL is illustrated in Figure 2. Specifically, SDRL considers labeled and unlabeled samples as two inconsistent domains, and it continuously updates the representations in a common subspace through a dual-classifier domain adaptive strategy. At the same time, the prediction results from two classifiers are compared and the most confident labels are extracted as pseudo labels for the following training iterations. Moreover, a label relation tensor is proposed to explicitly explore the label relations. By this way, feature-label and label-label relations are fully considered. The contributions of SDRL are listed below:

- A two-classifier domain adaptation mechanism is designed. It effectively mitigates the distribution shift between labeled and unlabeled samples, which improves and stabilizes the final performance.
- An active pseudo-label assignment strategy is proposed based on the two-classifier structure. It assigns and includes the most confident labels of the unlabeled samples in the training iteration. This strategy efficiently explores the label relations even in the unlabeled samples.
- A multi-label relation learning structure is proposed associated with a label relation tensor. It is designed to explicitly explore the latent relations across the labels and enhance the effectiveness and robustness of the model.

Our SDRL framework is a data-driven approach which automatically and effectively explores feature and label relations. All the networks and the co-training procedure can be run jointly, and the prediction results can be directly obtained without extra steps. It makes our model feasible for practical applications without additional modifications. SDRL is an extension of our previous work [17]. Compared with [17], there are three major improvements. First, a learnable relation tensor is designed to explicitly reveal the label relations. Second, the explored knowledge gives the model more capacity and flexibility to effectively improve the final performance. Moreover, extensive experiments and comprehensive ablation studies are done to prove the effectiveness of each module.

II. Related Work

A. Multi-Label Learning (MLL)

MLL is a general setting where multiple labels could be assigned to one instance [1]. A lot of practical applications are related to this problem, including text classification [18], image annotation [1], and video concept recognition [19]. A simple solution is deploying multiple single-label classifiers. However, the disadvantage of this strategy is that it does not take the relations across labels into consideration. A pre-defined label relation knowledge is considerably helpful for MLL. [20], [6], [11] use semantic knowledge to build a label dependency graph. [5] implements a label semantic structure, which covers different labels and avoids label noise. However, building such kind of label relation knowledge requires expert knowledge. [5] explicitly includes the semantic relations as a graph structured data as prior knowledge, which considerably improve the classification performance. While, this strategy required expert knowledge which is difficult and costly to obtain. Moreover, this pre-defined relation is based on unique tasks which is not feasible to be deployed to other tasks. [21] projects the labels into a subspace and then learn the latent relations in the subspace. [22] studies the object relations using attention and RNN. [22] deploys recurrent networks as well as attention strategy for label relation learning. [23] utilizes a

X-Transformer framework, which fine-tunes deep transformer models for the extreme MLL. A LightXML framework is proposed in [24] which adopts dynamic negative label sampling strategies. An efficient method was proposed to eliminate the negative effect of label noise in [25]. [26] introduces a multi-modality merging mechanism called MEFF for MLL. [27] proposed a scalable deep learning architecture that incorporates label text and label connections which provides effective and efficient real-time inferring. [28] designed a novel APLC-XLNet structure which fine-tunes the auto regressive strategy and obtains a dense representation of the target labels. [29] proposed a DECAF framework which obtains enriched models from the label metadata and jointly learns the model parameters and the feature representations. However, these approaches require large-scale datasets. While, the sizes of the related databases [30], [31], [3], [4] are relatively small which limits the potential performance. A few general works proposed to handle these challenges [32], however it is not feasible for MLL scenario.

In this paper, a semi-supervised learning model is proposed which focuses on the label relation exploration. Previous approaches mainly explore the label relations globally while neglecting the label information residing in each instance. In addition, our method not only learns the label relation from the labeled data, but also extends it to the unlabeled data.

B. Semi-Supervised Learning (SSL)

SSL jointly explores the source and target data points [8], [33], [7], [34], [35]. It is a useful mechanism for the scenario where a large amount of samples can be obtained easily while the labeling procedure is expensive. More detailed introduction could be found in [8], [36]. Graph-based approach is an effective direction [9], [7], [37], [10], [38]. [9] actively extracts informative samples from the training set by an initialization independent approach. A continuous relaxation mechanism is proposed in [7] where the Gaussian random fields and harmonic function method are deployed. However, these approaches highly depend on whether the unlabeled data has the similar distribution as the labeled data. Distribution differences could easily cause negative effects. To solve this, [37] generated an adaptive similarity graph to measure the similarity in a more flexible way. [10] introduced a graph optimization strategy to solve the unsupervised feature selection. [17] deployed a domain adaptation method for distribution alignment of labeled and unlabeled samples. [35] utilized an adaptive graph for robust label prediction. While, the methods mainly reveal extra information from feature space. While, in each instance, we consider the label-level relation is also crucial. A few methods explore the label correlations in semi-supervised scenario. The nuclear normalization is used in [39] with the singular value decomposition to reveal the correlation knowledge. [40] combines the label correlation learning and feature selection based on sparse constraints. [41] deploys a soft label matrix to learn the label distributions, recover missing labels, and predict unlabeled samples simultaneously. However, these methods indirectly explore the correlations which limit the potential performance, and most of the methods are computational costly.

Our approach explicitly explores the label correlations via a novel correlation tensor. Since deep networks are utilized, the capacity and efficiency are further enhanced. Moreover, our model considers labeled data and unlabeled data as two domains and utilizes domain adaptation strategy to mitigate the negative effect of the distribution shift.

III. Our Approach

A. Preliminaries and Motivation

$\{X_l, Y_l\}$ is the given labeled data, where $X_l \in \mathbb{R}^{d \times n_l}$ denotes the matrix of all feature vectors, and $Y_l \in \mathbb{R}^{d_l \times n_l}$ represents the corresponding labels. n_l is the number of labeled instances. d_l and d is the label vector dimension and feature dimension respectively. A column of X_l, $x_i \in \mathbb{R}^d$, denotes a single sample and y_i is the label vector of x_i. $X_u \in \mathbb{R}^{d \times n_u}$ and $Y_u \in \mathbb{R}^{d_l \times n_u}$ are the feature and label matrix from of the unlabeled instances. In the semi-supervised scenario, the goal is to recognize Y_u when X_l, X_u, and Y_l are given. The definition summary is shown in Table I.

B. Our Method

Figure 2 shows the framework of SDRL method. There are four networks and a trainable label relation tensor in SDRL. Specifically, it contains a feature encoder $E(\cdot)$, a label relation network $C_R(\cdot)$, and two multi-label classifiers $C_1(\cdot)$ and $C_2(\cdot)$. At the beginning, $E(\cdot)$ encodes the representations of the labeled and unlabeled samples into a common subspace:

$$\begin{aligned} Z_l &= E(X_l), \\ Z_u &= E(X_u), \end{aligned} \tag{1}$$

where $Z_u \in \mathbb{R}^{d_z \times n_u}$ and $Z_l \in \mathbb{R}^{d_z \times n_l}$ are the obtained features in the subspace. d_z is the feature dimension. As introduced above, X_l and X_u could be obtained from different resources, which means that the feature distributions could be slightly different. Training the model in the original feature space would lead the performance decrease. To this end, inspired by the MDA idea [42], we deployed a two-classifier structure to achieve the initial multi-label classification and label/unlabeled domain adaptation jointly. In our model, $C_1(\cdot)$ and $C_2(\cdot)$ are utilized to obtain the initial prediction results where the subspace features are set as the inputs:

$$L_C(X_l, Y_l) = \frac{1}{2}\left[\|C_1(Z_l) - Y_l\|_{\mathrm{F}}^2 + \|C_2(Z_l) - Y_l\|_{\mathrm{F}}^2\right]. \tag{2}$$

$C_1(Z_l)$ and $C_2(Z_l)$ are the prediction of the labeled samples, and $L_C(\cdot, \cdot)$ is the loss function. In the optimization pipeline, $E(\cdot)$, $C_1(\cdot)$, and $C_2(\cdot)$ are jointly optimized:

$$\min_{E, C_1, C_2} L_C(X_l, Y_l), \tag{3}$$

In training procedure, the supervision information from labeled samples are utilized to initially train $C_1(\cdot)$, $C_2(\cdot)$, and $E(\cdot)$. However, the unlabeled samples are not explored yet. As we mentioned, there could be domain shift between labeled and unlabeled samples. Thus, we reused the two classifiers as a domain adaptation framework which effectively aligns the distributions together. Specifically, an adversarial training strategy is used to update $E(\cdot)$, $C_1(\cdot)$ and $C_2(\cdot)$. When

TABLE I
DESCRIPTION TABLE OF SYMBOLS

Symbol	Descriptions
x_i	Original feature of the i-th data.
y_i	Groundtruth label vector of x_i.
X_l, X_u	Feature matrix of labeled and unlabeled samples.
f_{1i}	Predicted label vector of sample x_i from classifier 1.
f_{2i}	Predicted label vector of sample x_i from classifier 2.
Y_l	Groundtruth label of X_l.
d, d_l	Dimensions of feature space and label space.
n_l, n_u	Labeled and unlabeled data point numbers.
α	Hyper-parameter.

TABLE II
DATASETS STATISTICAL SUMMARY

Datasets	Setting	Training	Testing	Labels	Ave
Corel5K [30]	General	4,500	499	260	3.4
ESP Game [31]	General	18,689	2,081	268	4.7
IAPRTC-12 [43]	General	17,665	1,962	291	5.7
SUN [4]	General	6,387	6,513	102	6.3
	Zero-shot	12,900	1,440		
CUB [3]	General	4,374	4,468	312	31.4
	Zero-shot	8,842	2,946		
AWA [44]	General	12,154	12,141	85	15.0
	Zero-shot	24,295	6,180		

encoder $E(\cdot)$ is not updated, $C_1(\cdot)$ and $C_2(\cdot)$ are updated for maximizing the multi-label variance of the unlabeled instances Z_u. The variance between two predictions could be evaluated by a l_1-norm metric:

$$d(f_{1i}, f_{2i}) = \frac{1}{d_l}\|f_{1i} - f_{2i}\|_1, \tag{4}$$

where $\|\cdot\|_1$ is l_1-norm operation. $f_{1i} \in \mathbb{R}^{d_l}$ and $f_{2i} \in \mathbb{R}^{d_l}$ are the prediction results from $C_1(\cdot)$ and $C_2(\cdot)$. We consider Eq. (4) is an effective and simple difference metric, while other algorithms such as l_2-norm could also be utilized. Then, the loss function of the Z_u prediction differences is shown below:

$$L_{DA}(X_u) = d(C_1(Z_u), C_2(Z_u)). \tag{5}$$

In this step, we aim to train $C_1(\cdot), C_2(\cdot)$ for maximizing the classification difference:

$$\min_{C1,C2} -L_{DA}(X_u) + \lambda L_C(X_l, Y_l), \tag{6}$$

The second term is the supervision training loss which is used to keep the performance stable on the labeled sets, which is crucial to avoid the model collapse. λ is a hyper-parameter which controls the training weights between L_C and L_{DA}. In addition, $E(\cdot)$ aims to learn the feature representation in subspace which reduces the differences between the classification results. The objective function for updating $E(\cdot)$ is shown below:

$$\min_{E} L_{DA}. \tag{7}$$

In summary, the adversarial learning strategy is deployed to alternately update $E(\cdot)$, $C_1(\cdot)$, and $C_2(\cdot)$. Base on this strategy, the samples from both labeled and unlabeled are will aligned, and the domain shift challenge is mitigated.

As introduced above, label relation is another crucial knowledge for improving the performance. $C_1(\cdot)$ and $C_2(\cdot)$ could provide the initial multi-label predictions. However, they are not capable enough to capture and utilize the sophisticated label relations. To this end, we consider the continuous prediction scores should contain extra information which could be further explored. Thus, a specifically designed relation network is proposed. As illustrated in Figure 2, we extend the initial prediction result from $C_1(\cdot)$, f_{1i}, to a matrix by horizontally padding. By this way, we could obtain the label matrix F_{1i}^H:

$$F_{1i}^H = [f_{1i}, f_{1i}, f_{1i}, \cdots, f_{1i}], \tag{8}$$

where $F_{1i}^H \in \mathbb{R}^{d_l \times d_l}$. Similarly, we can have the vertical padding matrix of f_{2i} via the operation below:

$$F_{2i}^V = [f_{2i}, f_{2i}, f_{2i}, \cdots, f_{2i}]^\top, \tag{9}$$

where $F_{2i}^V \in \mathbb{R}^{d_l \times d_l}$. By this way, we obtain the two padding matrices F_{1i}^H and F_{2i}^V. The major goal for the padding operation is to make calculation of the label correlation tensor easier in the implementation process. Given a position (m, n), the combination of the prediction score of F_{1i}^H and F_{2i}^V denote to the pairwise prediction scores of m-th and n-th candidate labels. Specifically, $F_{1i}^H(m, n)$ is the prediction score of the m-th label, x_i. $F_{2i}^V(m, n)$ denotes the prediction score of n-th label of x_i. Instead of simple operations between the pairwise label scores (e.g., sum or multiplication), we proposed a trainable label relation tensor which explicitly explores the correlations of each pair of the labels. Specifically, the label relation tensor $T_R \in \mathbb{R}^{d_l \times d_l \times 2}$, which could be considered as the stack of two matrices, as $T_R = [T_R^H, T_R^V]$, where $T_R^H, T_R^V \in \mathbb{R}^{d_l \times d_l}$. We obtain the element-wise multiplication between the label matrices and the correlation tensor, then we sum the obtained matrices. The equation is shown below:

$$F_i^C = \delta(F_{1i}^H \circ T_R^H + F_{2i}^V \circ T_R^V), \tag{10}$$

where $\circ$ denotes the element-wise multiplication, and $F_i^C \in \mathbb{R}^{d_l \times d_l}$ is the label combination matrix, where each element of F_i^C is the fusion of a pairwise initial predictions obtained from T_R. The elements of T_R are considered as the weights of the pairwise predictions. $\delta(\cdot)$ is a nonlinear activation such as ReLU. F_i^C is then reshaped to a vector as $\mathbb{R}^{d_l^2}$ and be forwarded to a final relation learning network, $C_R(\cdot)$, which is used to obtain the final predictions. The loss function is illustrated below:

$$L_{C_R} = \sum_{i=1}^{n_l} \|y_i - C_R(F_i^C)\|_2^2. \tag{11}$$

In the training procedure, $C_R(\cdot)$ and T_R are trained simultaneously with the other networks:

$$\min_{E,C_1,C_2,C_R,T_R} \frac{\alpha}{2} L_C + (1-\alpha) L_{C_R}, \tag{12}$$

where $\alpha \in [0, 1]$ is the trade-off value which is used to balance the training between $C_1(\cdot)$, $C_2(\cdot)$, and $C_R(\cdot)$. In our implementation, we set $\alpha = 0.5$ as default for simplicity. Further parameter tuning (e.g., cross validation) could slightly improve the performance. The detailed parameter sensitivity analysis is provided in Section IV.

In conventional active learning scenario, the prediction confidence could be directly obtained by finding the highest prediction scores. However, MLL contains multiple positive predictions which are not feasible by this strategy. To solve this

TABLE III
CLASSIFICATION PERFORMANCE

Datasets	Methods	Pre	Rec	F1	N-R	mAP
Corel	Regression	0.2859	0.3211	0.3025	128	0.3630
	SSMLDR	0.2741	0.3366	0.3022	143	0.3410
	FastTag	0.3123	0.3657	0.3369	143	0.3871
	ML-PGD	0.2575	0.2911	0.2732	122	0.3727
	SAE	0.2962	0.3442	0.3184	141	0.3823
	AG^{2}E	0.3011	0.3520	0.3245	**157**	0.3568
	DRML	0.3154	0.3775	0.3437	148	0.4127
	Ours	**0.3341**	**0.3798**	**0.3555**	150	**0.4303**
ESP	Regression	0.3793	0.2038	0.2653	215	0.3440
	SSMLDR	0.3298	0.1885	0.2399	226	0.3156
	FastTag	0.4011	0.1927	0.2617	208	0.3904
	ML-PGD	0.3239	0.2012	0.2482	210	0.4077
	SAE	0.3861	0.1743	0.2402	194	0.3842
	AG^{2}E	0.3548	0.1525	0.2133	213	0.3730
	DRML	0.4373	0.2189	0.2918	227	0.4105
	Ours	**0.4396**	**0.2258**	**0.2984**	**231**	**0.4231**
IAP	Regression	0.4287	0.2041	0.2765	199	0.4211
	SSMLDR	0.3491	0.2520	0.2927	229	0.3981
	FastTag	0.4346	0.2267	0.2980	227	0.4596
	ML-PGD	0.4132	0.2441	0.3011	230	0.4674
	SAE	0.3537	0.2282	0.2774	213	0.4309
	AG^{2}E	0.3829	0.2330	0.2897	229	0.4353
	DRML	**0.4570**	0.2531	0.3258	230	0.5148
	Ours	0.4513	**0.2719**	**0.3393**	**235**	**0.5257**
SUN	Regression	0.6209	0.1473	0.2457	102	0.6807
	SSMLDR	0.6879	0.1700	0.2726	102	0.6723
	FastTag	0.6816	0.1473	0.2457	102	0.6914
	ML-PGD	0.7110	0.1614	0.2631	101	0.7087
	SAE	0.7183	0.1638	0.2668	98	0.7012
	AG^{2}E	0.7685	0.1765	0.2871	99	0.6778
	DRML	0.7906	0.1793	0.2923	102	0.6800
	Ours	**0.7918**	**0.1912**	**0.2994**	**102**	**0.7102**
CUB	Regression	0.2010	0.0239	0.0428	157	0.0638
	SSMLDR	0.3410	0.0473	0.0832	178	0.2329
	FastTag	0.2147	0.0359	0.0615	167	0.3144
	ML-PGD	0.3334	0.0451	0.0794	155	0.3288
	SAE	0.3383	0.0514	0.0908	196	0.3255
	AG^{2}E	0.3409	0.0531	0.0911	190	0.3106
	DRML	0.3714	0.0548	0.0955	202	0.3542
	Ours	**0.3755**	**0.0559**	**0.0973**	**205**	**0.3720**
AWA	Regression	0.8798	0.0821	0.1500	75	0.8626
	SSMLDR	0.7812	0.0858	0.1546	67	0.8346
	FastTag	0.7861	0.0949	0.1694	72	0.8791
	ML-PGD	0.5395	0.0635	0.1136	57	0.9121
	SAE	**0.9683**	**0.0957**	**0.1742**	73	0.9397
	AG^{2}E	0.8483	0.0827	0.1507	73	0.9033
	DRML	0.8689	0.0835	0.1523	75	0.9441
	Ours	0.9593	0.0856	0.1571	**82**	**0.9476**

TABLE IV
CLASSIFICATION PERFORMANCE WITH AUGMENTED LABEL SETS

Datasets	Methods	Pre	Rec	F1	N-R	mAP
Corel-A	Regression	0.2842	0.2304	0.2545	103	0.3762
	SSMLDR	0.3036	0.2791	0.2908	134	0.3660
	FastTag	0.3329	0.3145	0.3234	136	0.4127
	ML-PGD	0.3245	0.3011	0.3124	140	0.4275
	SAE	0.3168	0.3037	0.3101	128	0.4192
	AG^{2}E	0.3273	0.3172	0.3221	143	0.3985
	DRML	0.3345	**0.3671**	0.3500	147	0.4315
	Ours	**0.3461**	0.3582	**0.3520**	**147**	**0.4515**
ESP-A	Regression	0.3848	0.1256	0.1894	178	0.3913
	SSMLDR	0.3253	0.1697	0.2231	202	0.3357
	FastTag	0.3886	0.1531	0.2197	196	0.4254
	ML-PGD	0.3713	0.1184	0.1795	162	0.4211
	SAE	0.3153	0.1425	0.1966	156	0.4050
	AG^{2}E	0.3518	0.1492	0.2095	196	0.4030
	DRML	0.4202	0.1744	0.2465	209	0.4121
	Ours	**0.4335**	**0.1815**	**0.2559**	**213**	**0.4325**

challenge, we proposed a method which reuse the prediction results from $C_1(\cdot)$ and $C_2(\cdot)$ to determine the prediction confidence. In our model, the unlabeled samples are forwarded to $C_1(\cdot)$ and $C_2(\cdot)$, then the prediction differences is obtained by Eq. (13). The prediction difference evaluation we used here is l_2-norm, which is different compared with Eq. (5):

$$d_f(x_i) = \|C_1(E(x_i)) - C_2(E(x_i))\|_2^2, \tag{13}$$

where x_i is a feature extracted from the unlabeled set. When the differences are obtained, we sort $d_f(x_i)$ in an ascending way and select the first multiple predictions as the pseudo labels. After that, we include the pseudo labels and the samples to the labeled set in the future training epoch. Since databases have their unique characteristics (e.g., labels formats and scale), which leads to slightly different label assignment pipeline. For the CUB dataset [3], a threshold value $d = 1$ is set. We consider x_i as a training sample when $d_f(x_i) \leq d$.

There are two major phases in the training procedure. In the first phase, the encoder and two classifiers (i.e., $E(\cdot)$, $C_1(\cdot)$ and $C_2(\cdot)$) are first initialized based on the supervised scenario with the labeled samples. This is important and necessary which would make the initial correct prediction for the pseudo label assignment in the next phase. The loss function is $L_C(X_l, Y_l)$ as shown in Eq. (3). In our implementation, we trained $E(\cdot)$, $C_1(\cdot)$ and $C_2(\cdot)$ for 50 epochs. From the experiments, we observe that $L_C(X_l, Y_l)$ converges in all evaluated datasets. The second phase is the loop of the pseudo label assignment and extra training procedure. Specifically, we set the ratio $\mathcal{R}$ which extracts a small portion of the most confident predicted samples from the unlabeled samples. $\mathcal{R}$ is different for different datasets, in our experiment, we set $\mathcal{R}$ in $[0.01, 0.03]$. For example, if $\mathcal{R} = 0.01$ and when 1000 unlabeled samples are given, we averagely extract 10 samples at the beginning, and when only 500 samples are left, we extract 5 samples. We extract at least 1 sample in each pseudo label assignment procedure. Then, the assigned pseudo label would be considered as the ground truth sample for training $E(\cdot)$, $C_1(\cdot)$, and $C_2(\cdot)$ alternatively based on Eq. (6) and Eq. (7). In the training procedure, the relation tensor T_R and the final classifier $C_R(\cdot)$ are consistently updated in both the first and the second phases. The pseudo label is also obtained from the output of $C_R(\cdot)$. Moreover, another fixed number-based assignment strategy also works which extracts a consistent number (e.g., $[1, 20]$) of confident samples for pseudo label assignment. The relatively small ratio of the label assignment is to assure that the most confident and correct pseudo labels are assigned. In our implementation, $C_1(\cdot)$ and $C_2(\cdot)$, and $C_R(\cdot)$ are fully connected networks with different layers. Specifically, the $E(\cdot)$ is a 1-layer structure. $C_1(\cdot)$ and $C_2(\cdot)$ are 1-layer structure associated with Sigmoid function. The input to $C_R(\cdot)$ consists of the results from $C_1(\cdot)$, $C_2((\cdot)$.

IV. EXPERIMENTS

A. Datasets

Six multi-label datasets are utilized in our experiments. The statistical summary is illustrated in Table II.

- **Corel5K Dataset** [30] is an image dataset containing photos from the Corel CD database. There are $4,500$ and 499 samples for training and testing respectively. The total candidate labels is 260 and 3.40 labels per sample on average.

TABLE V
ZERO-SHOT MULTI-LABEL CLASSIFICATION RESULTS

Datasets	Methods	Pre	Rec	F1	N-R	mAP
SUN	Regression	0.7047	0.1548	0.2539	97	0.6616
	SSMLDR	0.6637	0.1481	0.2422	95	0.6581
	FastTag	0.6906	0.1522	0.2494	90	0.6706
	ML-PGD	0.7037	0.1471	0.2433	95	0.6829
	SAE	0.6978	0.1710	0.2747	**100**	0.6513
	AG^{2}E	0.7125	0.1618	0.2637	88	0.6693
	DRML	0.7512	0.1794	0.2896	97	0.6924
	Ours	**0.7583**	**0.1862**	**0.2990**	99	**0.7010**
CUB	Regression	0.2600	0.0307	0.0549	160	0.2693
	SSMLDR	0.2926	0.0383	0.0677	166	0.2329
	FastTag	0.2231	0.0434	0.0726	143	0.2967
	ML-PGD	0.2392	0.0365	0.0635	117	0.3178
	SAE	0.2552	0.0469	0.0798	**167**	0.3102
	AG^{2}E	0.2808	0.0481	0.0821	163	0.2693
	DRML	0.2981	**0.0486**	0.0835	153	0.3338
	Ours	**0.3110**	0.0484	**0.0838**	164	**0.3341**
AWA	Regression	0.7555	0.0766	0.1392	66	0.8809
	SSMLDR	0.7017	0.0764	0.1378	66	0.7858
	FastTag	0.8610	0.0912	0.1649	81	0.8918
	ML-PGD	0.4338	0.0623	0.1091	49	0.8677
	SAE	0.9015	**0.0926**	**0.1679**	78	0.8918
	AG^{2}E	0.8247	0.0811	0.1476	71	0.8874
	DRML	0.9023	0.0832	0.1524	81	0.8985
	Ours	**0.9152**	0.0857	0.1567	**81**	**0.9019**

TABLE VI
ABLATION STUDY FOR RELATION LEARNING NETWORK

Networks Structure	Pre	Rec	F1	N-R	mAP
CON 1-layer	0.7593	0.1782	0.2887	101	0.6518
CON 2-layer	0.7818	0.1834	0.2971	101	0.6857
CON 3-layer	0.7832	0.1819	0.2952	102	0.6872
CON 4-layer	0.7830	0.1800	0.2927	102	0.6869
AVE 1-layer	0.7531	0.1683	0.2751	100	0.6683
AVE 2-layer	0.7792	0.1762	0.2874	102	0.6791
AVE 3-layer	0.7811	0.1781	0.2901	102	0.6854
AVE 4-layer	0.7821	0.1791	0.2915	102	0.6842
ADD 1-layer	0.7459	0.1613	0.2651	100	0.6435
ADD 2-layer	0.7482	0.1657	0.2713	101	0.6651
ADD 3-layer	0.7510	0.1654	0.2711	102	0.6686
ADD 4-layer	0.7550	0.1641	0.2696	102	0.6704
Ours	**0.7918**	**0.1912**	**0.2994**	**102**	**0.7102**

• **ESP Game Dataset** [31] developed an interactive system between human and computer for data labeling, and the interaction is designed like a game. The training and testing numbers are $18,689$ and $2,081$ respectively. Moreover, the number of candidate labels is 268 with 4.69 label on average.

• **IAPRTC-12 Dataset** [43] is used for MLL and cross-language scenario. The categories such as landscapes, actions, and animals are included. The total number of labels is 291 and the average label per instance is 5.72.

• **SUN Dataset** [4] is proposed for detailed scene recognition and analysis tasks. There are more than $14,000$ samples corresponding to 700 different categories. The number of candidate labels is 102 and the average active label is 6.3.

• **CUB Dataset** [3] is a bird image dataset which involves 200 bird species. The numbers of training and testing split are $4,374$ and $4,468$. There are 312 label candidates with averagely 31.4 labels for each instance.

• **AWA Dataset** [44] is an animal with attribute dataset. The total number of the animal images are $30,000$ corresponding to 50 different animals. The average label is 15 selected from a 85 label candidates. The range of the label value is $[0, 100]$.

For AWA, CUB, and SUN dataset, we deploy VGG Networks [45] to obtain the visual features. The original extracted features are $4,096$ dimensional vector for each instance. VGG is pre-trained on ImageNet [46] and fixed in the whole procedure. We utilize 15 different visual descriptors for Corel5K, ESP Game, and IAPRTC datasets, which are extracted by [47].

B. Experimental Setup

We pretrained $C_1(\cdot)$ and $C_2(\cdot)$ for several epochs, then all the networks are jointly trained. When $C_R(\cdot)$ becomes stable, the most confident predictions are assigned as pseudo labels to the corresponding samples. Meanwhile, the strategy for assigning pseudo labels is slightly different across different datasets. Specifically, for ESP, Corel5K, and IAPRTC-12 datasets, the pseudo label is binary (i.e., $\{0, 1\}$) based on 0.5 as the threshold. For the SUN dataset, the pseudo label is illustrated by the combinations of $\{0, 0.33, 0.66, 1\}$ to match the original label assignments in ground-truth. For CUB and AWA databases, the assigned label is the prediction results from $C_R(\cdot)$ since they utilize the continuous label value.

Multi-label prediction performance is evaluated in general and zero-shot [2], [48] settings in our experiments. In the general setting, the labeled and unlabeled samples are evenly and randomly selected from the complete data points, where each set has roughly half of the samples in the dataset. In zero-shot experiment, there are no overlap in the labeled and unlabeled sets. Considering the distribution shift is more considerable than the general setting, it is more difficult for keeping high performance. For zero-shot test, there are default splits in AWA, CUB, and SUN datasets. We evaluate our methods as well as other benchmark multi-label approaches. The benchmark methods are briefly introduced below:

• **Least Squares Regression (Regression)** is a traditional method which projects the feature space to label space based on a matrix without nonlinear transformations.

• **FastTag** [25] is specifically designed for addressing noisy and incomplete training samples. It designs two linear projectors for completing missing labels and prediction respectively.

• **Semi-Supervised Multi-Label Dimensionality Reduction (SSMLDR)** [49] explores feature distribution structural knowledge via a transformation matrix, and transfers knowledges across labeled and unlabeled samples.

• **Multi-Label with a Mixed Graph (ML-PGD)** [20] reveals the latent label interdependence via a novel hybrid diagram. In the proposed graph, the nodes are the candidate labels and the edges are the latent relations of different nodes.

• **Semantic AutoEncoder (SAE)** [50] utilizes linear autoencoder strategy for solving label prediction problem. The encoder and decoder share the same weight to project the feature space to label space, and then back to feature space.

• **Adaptive Graph Guided Embedding (AG2E)** [35] explores the potential of adaptive graph in MLL task. The pairwise similarity between all data points are optimized. Then, the graph is adaptively learned associated with other weights to achieve the best performance.

• **Dual Relation Multi-label learning (DRML)** [17] proposed

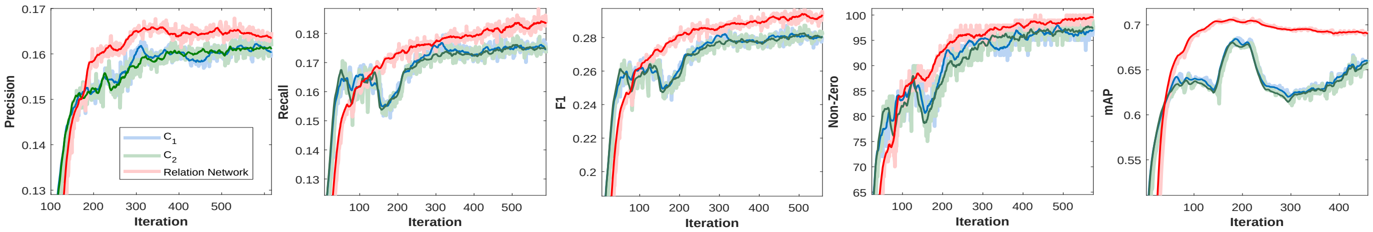


Fig. 3. Classification performance of $C_1(\cdot)$, $C_2(\cdot)$, and the relation network $C_R(\cdot)$ as the training iteration increase, where **blue** and **green** lines indicate $C_1(\cdot)$ and $C_2(\cdot)$, and $C_R(\cdot)$ is represented by the **red** line. It shows that after around 150 iterations, our relation network could consistently outperform $C_1(\cdot)$ and $C_2(\cdot)$, which demonstrates the effectiveness of the relation tensor as well as the relation network structure. We observed that the performance become stable after 300 iterations while the precision and mAP slightly decreased, we assume this is due to overfitting issue and an early stop could be deployed for solving it in other real-world practical applications.

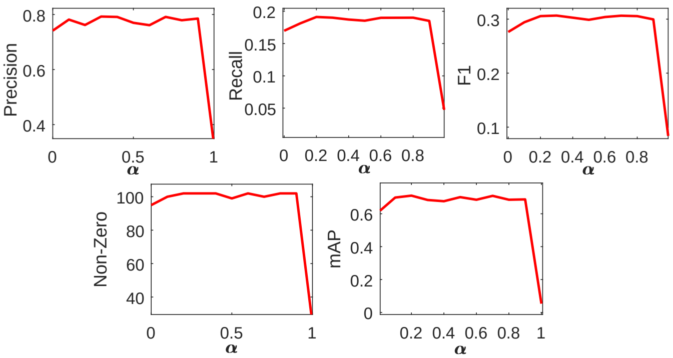


Fig. 4. Parameter sensitivity analysis of α. α balances the weights between $C_1(\cdot)$, $C_2(\cdot)$, and relation networks $C_R(\cdot)$ (Eq. (12)). Our approach achieves high and stable performance of α selection in the wide range (i.e., $[0.1, 0.9]$). $\alpha = 1.0$ denotes no losses for training $C_R(\cdot)$. The result demonstrates effectiveness of $C_R(\cdot)$ and parameter insensitive of our framework.

a two-classifier structure to align the distribution shift between labeled and unlabeled samples, and a relation learning network is further designed to explore the label relations.

SSMLDR, ML-PGD, AG2E, and DRML are semi-supervised methods. The labeled samples, their corresponding labels, and the unlabeled instances are given in the optimization procedure. Regression, FastTag, and SAE are supervised baselines. We only provide the labeled samples to train the model and the unlabeled samples are set as test samples. For evaluation purposes, we mainly deploy the metrics in [47]. First, the precision and the recall are calculated. The harmonic mean (F1-score, recall as well as precision) is also provided for comprehensive comparison. The non-zero recall represents the non-zero predictions in the results. In addition, we applied the mAP (mean average precision) which is introduced in [20] for convenient and overall assessment. Higher value represents better predictions in all the metrics.

C. Multi-label Classification

The evaluations of the general setting are shown in Table III. We could observe that SDRL considerably improves the prediction results than other benchmarks. Specifically, our approach achieves up to 3.0% performance improvements in mAP metric. It also achieves the highest performance in almost all metrics in most datasets. In addition, [20] provides a more comprehensive and refined label sets for ESP Game and Corel5K datasets. It increases the average number of labels in Corel5K from 3.40 to 4.84 and the number of labels in ESP Game from 4.69 to 7.27. We further tested the performance (Table IV) and it also shows the high performance compared with other methods.

D. Zero-shot Evaluation

We evaluated SDRL in zero-shot MLL setting. As introduced above, the training and testing categories are non-overlapped. Specifically, the samples still have the exact multi-label candidates while the categories are different between the labeled and unlabeled samples (e.g., *zebra* and *horse*). To this end, the domain shift is more significant. We tested our approach in AWA, CUB, and SUN databases. The databases are assigned the default split of training and testing for zero-shot setting. The comprehensive statistical summary is shown in Table II.

E. Model Analysis

The relation network, $C_R(\cdot)$, is one of the crucial modules. To demonstrate the usefulness of $C_R(\cdot)$, we show the training curve of $C_1(\cdot)$, $C_2(\cdot)$, and $C_R(\cdot)$ in Figure 3. It illustrates the results of each network as the training iteration increases. We observe that $C_R(\cdot)$ considerably outperforms $C_1(\cdot)$ and $C_2(\cdot)$, which denotes the effectiveness of $C_R(\cdot)$. In addition, there are two interesting phenomena. First, the performance of $C_1(\cdot)$ and $C_2(\cdot)$ are higher in the first tens of iterations, then $C_R(\cdot)$ outperforms others eventually. We assume this is mainly due to the natural lag characteristic of $C_R(\cdot)$ since the good training of $C_R(\cdot)$ is based on the roughly correct label predictions obtained from $C_1(\cdot)$ and $C_2(\cdot)$. Second, there is a slight performance drop before the final stable status, we conjecture this is the overfitting issue, extra cross-validation or early stop strategies could solve this issue.

To further demonstrate the superiority of $C_R(\cdot)$ over other network structures, we utilize a conventional multi-layer classifier to replace $C_R(\cdot)$. The input is the initial label predictions from $C_1(\cdot)$ and $C_2(\cdot)$, and the output is the final prediction. We tested three different structures. Concatenation ("CON") directly concatenates the predictions together. Average ("AVE") obtains the average predictions. Addition ("ADD") adds each pairwise of the label predictions together, which could be considered as a simplified version where all the elements in the correlation tensor, T_R, are equal, and the only functional module is the fully connected network. This setting separates the performance contribution of the correlation tensor and

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2021.3122003, IEEE Transactions on Image Processing

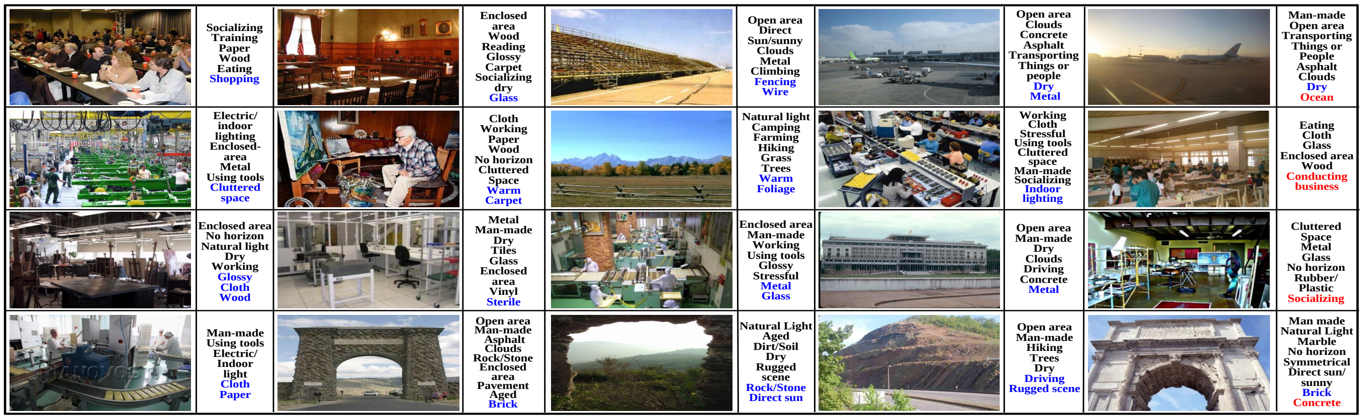


Fig. 5. Case study of the multi-label prediction results. Specifically, **Black** font shows the correct predictions, and **Red** font is assigned to show the incorrect predictions. Moreover, we observe several reasonable positive predictions while they are not in the ground truth sets. We consider these are the missing ground truth while our approach still effectively recovers these labels. From the result we can conclude that our approach is effective and robust.

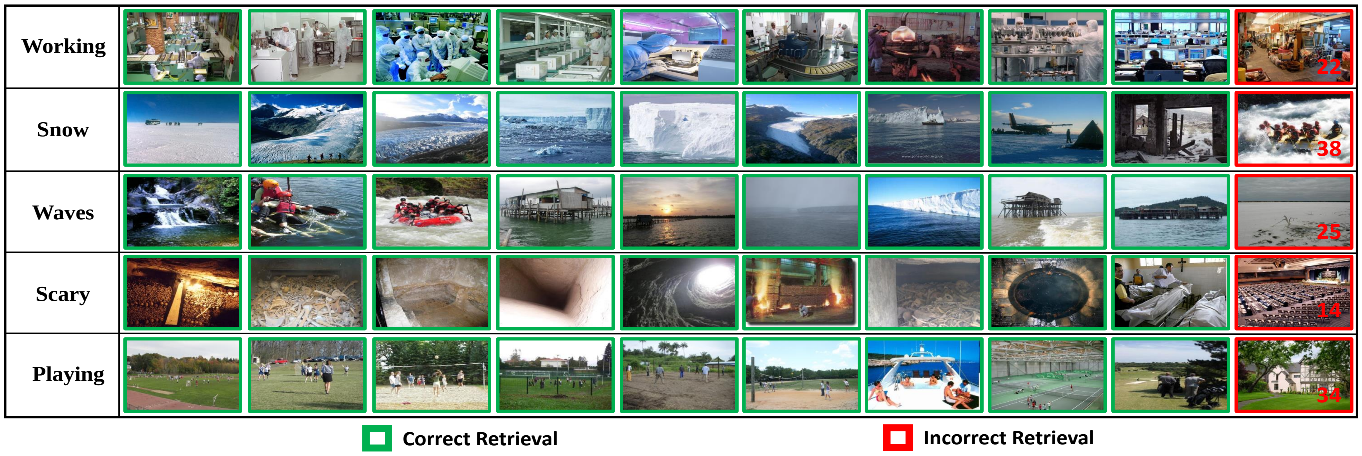


Fig. 6. Case studies of the image retrieval application, where the images are extracted based on a given target label. The **red** and **green** bounding-boxes denote the correct and incorrect retrievals, and the red numbers are the first incorrect retrieval. The results illustrate the effectiveness of our model.

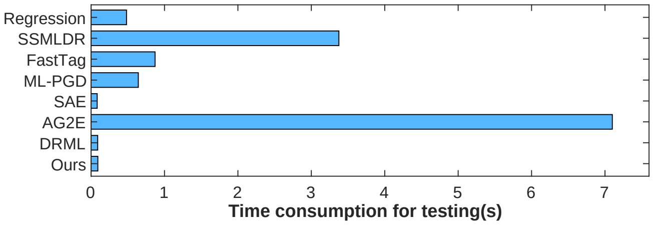


Fig. 7. Time consumption of inferring 2081 test samples of the ESP-Game dataset. Our SDRL achieves comparable efficiency, which is one of the most fast methods in all benchmarks.

the following fully connected network. The ablation studies are illustrated in Table VI, where we tested 1, 2, 3, and 4 layer networks. From Table VI, we can see that the general network structure is effective to slightly improve the performance, while the performance is saturated when the network achieves 3 or 4 layers. The result concretely demonstrates the effectiveness of correlation tensor.

F. Image Annotation

Image annotation aims to recover the multiple tags from the given images. We follow the zero-shot training/testing split of SUN dataset which means the categories of the target images are not existing in the optimization phase, and case studies are shown in Figure 5. In Figure 5, different font colors denote different prediction labels. We can see that most of the predictions are correct, which denotes that SDRL is capable of recovering multiple extra “missing” labels from the given images, and only a few incorrect predictions exist. This result further illustrates the robustness and effectiveness of our approach.

G. Image Retrieval

In our experiment, image retrieval searching and retrieving images from the test set based on a given label. We still follow the zero-shot setting to make the task be more practical. In our experiment, we first predict the multi-label vectors of all the candidate images, then we rank the prediction scores based on a target label (e.g., *“working”*) in Figure 6. Figure 6 illustrates the retrieved samples in SUN database, green and red denote correct and incorrect results. We can see that SDRL framework is effective for retrieval scenario even if the target instances are unseen in the training phase. This characteristic is more feasible for practical applications.

H. Time Consumption

Time consumption is an important consideration for real-world applications. We tested the time consumption of inferring 2081 test samples from the ESP-Game dataset. Figure 7 shows the time consumptions and our approach achieves competitive speed compared with other baselines. This is achieved by parallel computing based on the GPU acceleration.

V. Conclusion

We proposed a Semi-supervised Dual Relation Learning (SDRL) method in multi-label scenario. SDRL is designed to reveal the latent relations in given samples, including the instance-level relations in feature space between labeled and unlabeled data, and the label-level relations residing inside each sample. A two-classifier domain adaptation structure is deployed to effectively align the shifted feature distributions. Moreover, a relation tensor is proposed to efficiently and effectively learn the label-level relations and obtains more performance improvement without extra syntactical prior knowledge. All modules have been jointly to achieve the best performance. SDRL is evaluated on six benchmark databases in four 4 various tasks. The experimental results have shown that the performance had been significantly improved. Moreover, extensive ablation studies demonstrated the necessities of all proposed modules and the case studies further illustrate the robustness of our approach.

Acknowledgment

This research is supported by the U.S. Army Research Office Award W911NF-17-1-0367.

References

[1] M. R. Boutell, J. Luo, X. Shen, and C. M. Brown, "Learning multi-label scene classification," *Pattern Recognition*, vol. 37, no. 9, pp. 1757–1771, 2004.

[2] C. H. Lampert, H. Nickisch, and S. Harmeling, "Learning to detect unseen object classes by between-class attribute transfer," in *Proceedings of IEEE Computer Vision and Pattern Recognition*, 2009, pp. 951–958.

[3] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie, "The Caltech-UCSD Birds-200-2011 Dataset," California Institute of Technology, Tech. Rep. CNS-TR-2011, 2011.

[4] G. Patterson and J. Hays, "SUN attribute database: Discovering, annotating, and recognizing scene attributes," in *Proceedings of IEEE Computer Vision and Pattern Recognition*, 2012, pp. 2751–2758.

[5] B. Wu, W. Chen, P. Sun, W. Liu, B. Ghanem, and S. Lyu, "Tagging like humans: Diverse and distinct image annotation," in *Proceedings of IEEE Computer Vision and Pattern Recognition*, 2018, pp. 7967–7975.

[6] B. Wu, F. Jia, W. Liu, B. Ghanem, and S. Lyu, "Multi-label learning with missing labels using mixed dependency graphs," *International Journal of Computer Vision*, pp. 1–22, 2018.

[7] X. Zhu, Z. Ghahramani, and J. D. Lafferty, "Semi-supervised learning using Gaussian fields and Harmonic functions," in *Proceedings of International Conference on Machine Learning*, 2003, pp. 912–919.

[8] X. Zhu, "Semi-supervised learning literature survey," *University of Wisconsin-Madison*, 2005.

[9] F. Nie, D. Xu, and X. Li, "Initialization independent clustering with actively self-training method," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 42, no. 1, pp. 17–27, 2012.

[10] F. Nie, W. Zhu, and X. Li, "Unsupervised feature selection with structured graph optimization," in *Proceedings of AAAI Conference on Artificial Intelligence*, 2016, pp. 1302–1308.

[11] C. Zhaomin, W. Xiushen, W. Peng, and G. Yanwen, "Multi-label image recognition with graph convolutional networks," in *Proceedings of IEEE Computer Vision and Pattern Recognition*, 2019.

[12] L. Wang, Z. Ding, and Y. Fu, "Low-rank transfer human motion segmentation," *IEEE Transactions on Image Processing*, vol. 28, no. 2, pp. 1023–1034, 2019.

[13] C. Liu, X. Zeng, K. Wang, Q. Guo, and M. Xu, "Multi-task learning for macromolecule classification, segmentation and coarse structural recovery in cryo-tomography," *arXiv preprint arXiv:1805.06332*, 2018.

[14] R. Li, S. Wang, F. Zhu, and J. Huang, "Adaptive graph convolutional neural networks," in *Proceedings of AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018.

[15] X. Xu, C. Deng, and F. Nie, "Adaptive graph weighting for multi-view dimensionality reduction," *Signal Processing*, 2019.

[16] K. Li, C. Liu, H. Zhao, Y. Zhang, and Y. Fu, "ECACL: A holistic framework for semi-supervised domain adaptation," in *Proceedings of International Conference on Computer Vision*, 2021.

[17] L. Wang, Y. Liu, C. Qin, G. Sun, and Y. Fu, "Dual relation semi-supervised multi-label learning," in *Proceedings of AAAI Conference on Artificial Intelligence*, 2020.

[18] N. Ghamrawi and A. McCallum, "Collective multi-label classification," in *Proceedings of ACM Conference on Information and Knowledge Management*, 2005, pp. 195–200.

[19] G. Qi, X. Hua, Y. Rui, J. Tang, T. Mei, and H. Zhang, "Correlative multi-label video annotation," in *Proceedings of ACM Multimedia*, 2007, pp. 17–26.

[20] B. Wu, S. Lyu, and B. Ghanem, "ML-MG: Multi-label learning with missing labels using a mixed graph," in *Proceedings of IEEE International Conference on Computer Vision*, 2015, pp. 4157–4165.

[21] F. Tai and H.-T. Lin, "Multilabel classification with principal label space transformation," *Neural Computation*, vol. 24, no. 9, pp. 2508–2542, 2012.

[22] S. Chen, Y. Chen, C. Yeh, and Y. F. Wang, "Order-free RNN with visual attention for multi-label classification," in *Proceedings of AAAI Conference on Artificial Intelligence*, 2018.

[23] W. Chang, H. Yu, K. Zhong, Y. Yang, and I. S. Dhillon, "Taming pretrained transformers for extreme multi-label text classification," in *Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2020, pp. 3163–3171.

[24] T. Jiang, D. Wang, L. Sun, H. Yang, Z. Zhao, and F. Zhuang, "LightXML: Transformer with dynamic negative sampling for high-performance extreme multi-label text classification," *Proceedings of AAAI Conference on Artificial Intelligence*, 2021.

[25] M. Chen, A. Zheng, and K. Weinberger, "Fast image tagging," in *Proceedings of International Conference on Machine Learning*, 2013, pp. 1274–1282.

[26] W. Ge, S. Yang, and Y. Yu, "Multi-evidence filtering and fusion for multi-label classification, object detection and semantic segmentation based on weakly supervised learning," in *Proceedings of IEEE Computer Vision and Pattern Recognition*, June 2018.

[27] A. Mittal, N. Sachdeva, S. Agrawal, S. Agarwal, P. Kar, and M. Varma, "ECLARE: Extreme classification with label graph correlations," in *Proceedings of ACM International World Wide Web Conference*, April 2021.

[28] H. Ye, Z. Chen, D.-H. Wang, and B. Davison, "Pretrained generalized autoregressive model with adaptive probabilistic label clusters for extreme multi-label text classification," in *Proceedings of International Conference on Machine Learning*. PMLR, 2020, pp. 10 809–10 819.

[29] A. Mittal, K. Dahiya, S. Agrawal, D. Saini, S. Agarwal, P. Kar, and M. Varma, "DECAF: Deep extreme classification with label features," in *Proceedings of ACM International Conference on Web Search and Data Mining*, March 2021.

[30] P. Duygulu, K. Barnard, J. F. de Freitas, and D. A. Forsyth, "Object recognition as machine translation: Learning a lexicon for a fixed image vocabulary," in *Proceedings of European Conference on Computer Vision*, 2002, pp. 97–112.

[31] L. Von Ahn and L. Dabbish, "Labeling images with a computer game," in *Proceedings of ACM SIGCHI*, 2004, pp. 319–326.

[32] G. Mittal, C. Liu, N. Karianakis, V. Fragoso, M. Chen, and Y. Fu, "HyperSTAR: Task-aware hyperparameters for deep networks," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2020, pp. 8736–8745.

[33] L. Wang, Z. Ding, and Y. Fu, "Generic multi-label annotation via adaptive graph and marginalized augmentation," *ACM Transactions on Knowledge Discovery from Data*, vol. 16, no. 1, Jul. 2021.

[34] M. Belkin, I. Matveeva, and P. Niyogi, "Regularization and semi-supervised learning on large graphs," in *Proceedings of International Conference on Computational Learning Theory*, 2004, pp. 624–638.

[35] L. Wang, Z. Ding, and Y. Fu, "Adaptive graph guided embedding for multi-label annotation." in *Proceedings of International Joint Conferences on Artificial Intelligence*, 2018, pp. 2798–2804.

[36] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2021.3122003, IEEE Transactions on Image Processing

[37] J. Liu, M. Li, W. Ma, Q. Liu, and H. Lu, "An adaptive graph model for automatic image annotation," in *Proceedings of ACM SIG Multimedia*, 2006, pp. 61–70.

[38] L. Wang, B. Zong, Q. Ma, W. Cheng, J. Ni, W. Yu, Y. Liu, D. Song, H. Chen, and Y. Fu, "Inductive and unsupervised representation learning on graph structured objects," in *International Conference on Learning Representations*, 2020.

[39] L. Jing, C. Shen, L. Yang, J. Yu, and M. K. Ng, "Multi-label classification by semi-supervised singular value decomposition," *IEEE Transactions on Image Processing*, vol. 26, no. 10, pp. 4612–4625, 2017.

[40] X. Chang, H. Shen, S. Wang, J. Liu, and X. Li, "Semi-supervised feature analysis for multimedia annotation by mining label correlation," in *Proceedings of Pacific-Asia Conference on Knowledge Discovery and Data Mining*, 2014, pp. 74–85.

[41] J. Ma and T. W. Chow, "Robust non-negative sparse graph for semi-supervised multi-label learning with missing labels," *Information Sciences*, vol. 422, pp. 336–351, 2018.

[42] K. Saito, K. Watanabe, Y. Ushiku, and T. Harada, "Maximum classifier discrepancy for unsupervised domain adaptation," in *Proceedings of IEEE Computer Vision and Pattern Recognition*, 2018, pp. 3723–3732.

[43] M. Grubinger, P. Clough, H. Müller, and T. Deselaers, "The IAPR TC-12 benchmark: A new evaluation resource for visual information systems," in *Proceedings of OntoImage*, 2006.

[44] C. H. Lampert, H. Nickisch, and S. Harmeling, "Attribute-based classification for zero-shot visual object categorization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 3, pp. 453–465, 2014.

[45] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

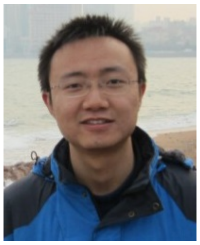
[46] J. Deng, W. Dong, R. Socher, L. Li, K. Li, and F. Li, "ImageNet: A large-scale hierarchical image database," in *Proceedings of IEEE Computer Vision and Pattern Recognition*, 2009, pp. 248–255.

[47] M. Guillaumin, T. Mensink, J. Verbeek, and C. Schmid, "TagProp: Discriminative metric learning in nearest neighbor models for image annotation," in *Proceedings of IEEE International Conference on Computer Vision*, 2009, pp. 309–316.

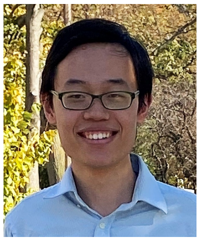
[48] M. Palatucci, D. Pomerleau, G. E. Hinton, and T. M. Mitchell, "Zero-shot learning with semantic output codes," in *Proceedings of Neural Information Processing Systems*, 2009, pp. 1410–1418.

[49] B. Guo, C. Hou, F. Nie, and D. Yi, "Semi-supervised multi-label dimensionality reduction," in *Proceedings of IEEE International Conference on Data Mining*, 2016, pp. 919–924.

[50] E. Kodirov, T. Xiang, and S. Gong, "Semantic autoencoder for zero-shot learning," in *Proceedings of IEEE Computer Vision and Pattern Recognition*, 2017.



Lichen Wang (S'16) received the B.Eng. degree in automation from Harbin Institute of Technology, Harbin, China in 2013. And the M.Eng. degree in control science and engineering from Xi'an Jiaotong University, Xi'an, China in 2016. He is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering, Northeastern University, Boston, MA, USA. His research interests include computer vision, machine learning and data mining. He has served as the reviewer for TIP, TPAMI, ICML, ICLR, NeurIPS, CVPR, ICCV, ECCV, IJCAI, AAAI, etc.

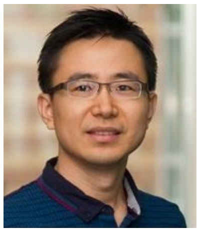


Yunyu Liu has received his B.Eng. degree in Information Engineering from Shanghai Jiao Tong University, China, in 2018 and M.S. degree in Electrical and Computer Engineering from Northeastern University, Boston, MA, USA, in 2020. He is currently pursuing the Ph.D. degree in the Computer Science Department at Purdue University. His research interests include multiview learning, deep learning and graph neural network. He has some top-tier conference papers accepted at ICCV, AAAI, ECCV et al.

Hang Di has received his B.Eng. degree in Electrical Engineering And Automation from Xi'an Jiao Tong University, China, in 2018 and M.S. degree in Electrical and Computer Engineering from Northeastern University, Boston, MA, USA, in 2020.

Can Qin has received the B.E. degree from the School of Microelectronics, Xidian University (XDU), China, in 2018. Currently, he is a Ph.D. student in the Department of Electrical and Computer Engineering, Northeastern University, under the supervision of Prof. Yun Raymond Fu. He has been awarded the Best Paper Award in ICCV Workshop on Real-World Recognition from Low-Quality Images and Videos 2019. He also has some top-tier conference papers accepted at NeurIPS, AAAI, ECCV et al. His research interests broadly include the transfer learning, semi-supervised learning and deep learning.

Gan Sun (S'19-M'20) is an associate professor in State Key Laboratory of Robotics, Shenyang Institute of Automation, Chinese Academy of Sciences. He received the B.S. degree from Shandong Agricultural University in 2013, the Ph.D. degree from Shenyang Institute of Automation, Chinese Academy of Sciences in 2020, and has been visiting Northeastern University from April 2018 to May 2019, Massachusetts Institute of Technology from June 2019 to November 2019. He also has some top-tier conference papers accepted at CVPR, ICCV, ECCV, AAAI, IJCAI, ICDM et al, and some top-tier journal papers accepted at TPAMI, TNNLS, TIP, TMM, TCSVT, Pattern Recognition et al. His current research interests include lifelong machine learning, multitask learning, medical data analysis, domain adaptation, deep learning and 3D computer vision.



Yun Fu (S'07-M'08-SM'11-F'19) received the B.Eng. degree in information engineering and the M.Eng. degree in pattern recognition and intelligence systems from Xi'an Jiaotong University, China, respectively, and the M.S. degree in statistics and the Ph.D. degree in electrical and computer engineering from the University of Illinois at Urbana-Champaign, respectively. He is an interdisciplinary faculty member affiliated with College of Engineering and the Khoury College of Computer Sciences at Northeastern University since 2012. His research interests are Machine Learning, Computational Intelligence, Big Data Mining, Computer Vision, Pattern Recognition, and Cyber-Physical Systems. He has extensive publications in leading journals, books/book chapters and international conferences/workshops. He serves as associate editor, chairs, PC member and reviewer of many top journals and international conferences/workshops. He received seven Prestigious Young Investigator Awards from NAE, ONR, ARO, IEEE, INNS, UIUC, Grainger Foundation; eleven Best Paper Awards from IEEE, ACM, IAPR, SPIE, SIAM; many major Industrial Research Awards from Google, Samsung, and Adobe, etc. He is currently an Associate Editor of the IEEE Transactions on Neural Networks and Leaning Systems (TNNLS). He is fellow of IEEE, IAPR, OSA and SPIE, a Lifetime Distinguished Member of ACM, Lifetime Senior Member of AAAI and Institute of Mathematical Statistics, member of ACM Future of Computing Academy, Global Young Academy, AAAS, INNS and Beckman Graduate Fellow during 2007-2008.