Transductive Multi-View Zero-Shot Learning

Yanwei Fu, Timothy M. Hospedales, *Member, IEEE*, Tao Xiang, and Shaogang Gong

Abstract—Most existing zero-shot learning approaches exploit transfer learning via an intermediate semantic representation shared between an annotated auxiliary dataset and a target dataset with different classes and no annotation. A projection from a low-level feature space to the semantic representation space is learned from the auxiliary dataset and applied without adaptation to the target dataset. In this paper we identify two inherent limitations with these approaches. First, due to having disjoint and potentially unrelated classes, the projection functions learned from the auxiliary dataset/domain are biased when applied directly to the target dataset/domain. We call this problem the *projection domain shift* problem and propose a novel framework, *transductive multi-view embedding*, to solve it. The second limitation is the *prototype sparsity* problem which refers to the fact that for each target class, only a single prototype is available for zero-shot learning given a semantic representation. To overcome this problem, a novel heterogeneous multi-view hypergraph label propagation method is formulated for zero-shot learning in the transductive embedding space. It effectively exploits the complementary information offered by different semantic representations and takes advantage of the manifold structures of multiple representation spaces in a coherent manner. We demonstrate through extensive experiments that the proposed approach (1) rectifies the projection shift between the auxiliary and target domains, (2) exploits the complementarity of multiple semantic representations, (3) significantly outperforms existing methods for both zero-shot and N-shot recognition on three image and video benchmark datasets, and (4) enables novel cross-view annotation tasks.

Index Terms—Transducitve learning, multi-view Learning, transfer Learning, zero-shot Learning, heterogeneous hypergraph

1 Introduction

Humans can distinguish 30,000 basic object classes [3] and many more subordinate ones (e.g. breeds of dogs). They can also create new categories dynamically from few examples or solely based on high-level description. In contrast, most existing computer vision techniques require hundreds of labelled samples for each object class in order to learn a recognition model. Inspired by humans' ability to recognise without seeing samples, and motivated by the prohibitive cost of training sample collection and annotation, the research area of *learning to learn* or *lifelong learning* [6], [35] has received increasing interests. These studies aim to intelligently apply previously learned knowledge to help future recognition tasks. In particular, a major and topical challenge in this area is to build recognition models capable of recognising novel visual categories without labelled training samples, i.e. zero-shot learning (ZSL).

The key idea underpinning ZSL approaches is to exploit knowledge transfer via an intermediate-level semantic representation. Common semantic representations include binary vectors of visual attributes [15], [27], [31] (e.g. 'hasTail' in Fig. 1) and continuous word vectors [11], [32], [44] encoding linguistic context. In ZSL, two datasets with disjoint classes are considered: a labelled auxiliary set where a semantic representation is given for each data point, and a target dataset to be classified without any labelled samples. The semantic representation is assumed to be shared between the auxiliary/source and target/test dataset. It can thus be re-used for knowledge transfer between the source and target sets: a projection function mapping low-level features to the semantic representation is learned from the auxiliary data by classifier or regressor. This projection is then applied to map each unlabelled target class instance into the same semantic space. In this space, a 'prototype' of each target class is specified, and each projected target instance is classified by measuring similarity to the class prototypes. Depending on the semantic space, the class prototype could be a binary attribute vector listing class properties (e.g., 'hasTail') [27] or a word vector describing the linguistic context of the textual class name [11].

Two inherent problems exist in this conventional zero-shot learning approach. The first problem is the *projection domain shift problem*. Since the two datasets have different and potentially unrelated classes, the underlying data distributions of the classes differ, so do the 'ideal' projection functions between the low-level feature space and the semantic spaces. Therefore, using the projection functions learned from the auxiliary dataset/domain without any adaptation to the target dataset/domain causes an unknown shift/bias. We call it the *projection domain shift* problem. This is illustrated in Fig. 1, which shows two object classes from the animals with attributes (AwA) dataset [28]: Zebra is one of the 40 auxiliary classes while Pig is one of 10 target classes. Both of them share the same 'hasTail' semantic attribute, but the visual appearance of their tails differs greatly (Fig. 1a). Similarly, many other attributes of Pig are visually different from the corresponding attributes in the auxiliary classes. Fig. 1b illustrates the projection domain

- *Y. Fu is with the Disney Research, Pittsburgh, PA 15213. E-mail: y.fu@qmul.ac.uk.*
- *T.M. Hospedales, T. Xiang, and S. Gong are with the School of Electronic Engineering and Computer Science, Queen Mary University of London E1 4NS, United Kingdom. E-mail: {t.hospedales, t.xiang, s.gong}@qmul.ac.uk.*

Manuscript received 17 July 2014; revised 2 Feb. 2015; accepted 18 Feb. 2015. Date of publication 2 Mar. 2015; date of current version 7 Oct. 2015. Recommended for acceptance by C. H. Lampert.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TPAMI.2015.2408354

0162-8828 © 2015 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.
Authorized licensed use limited to: Nanyang Technological University. Downloaded on August 31,2020 at 23:43:09 UTC from IEEE Xplore. Restrictions apply.

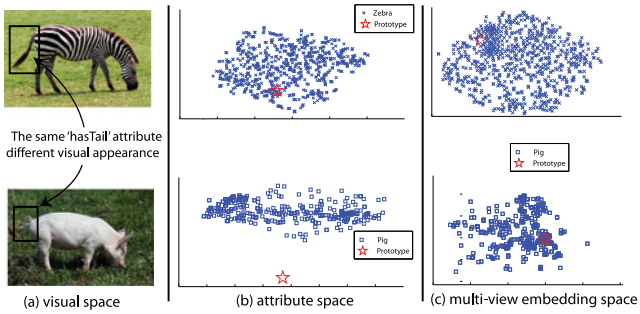


Fig. 1. An illustration of the projection domain shift problem. Zero-shot prototypes are shown as red stars and predicted semantic attribute projections (defined in Section 3.2) shown in blue.

shift problem by plotting (in 2D using t-SNE [47]) an 85D attribute space representation of the image feature projections and class prototypes (85D binary attribute vectors). A large discrepancy can be seen between the Pig prototype in the semantic attribute space and the projections of its class member instances, but not for the auxiliary Zebra class. This discrepancy is caused when the projections learned from the 40 auxiliary classes are applied directly to project the Pig instances—what 'hasTail' (as well as the other 84 attributes) visually means is different now. Such a discrepancy will inherently degrade the effectiveness of zero-shot recognition of the Pig class because the target class instances are classified according to their similarities/distances to those prototypes. To our knowledge, this problem has neither been identified nor addressed in the zero-shot learning literature.

The second problem is the *prototype sparsity problem*: for each target class, we only have a single prototype which is insufficient to fully represent what that class looks like. As shown in Figs. 6b and 6c, there often exist large intra-class variations and inter-class similarities. Consequently, even if the single prototype is centred among its class instances in the semantic representation space, existing zero-shot classifiers will still struggle to assign correct class labels—one prototype per class is not enough to represent the intra-class variability or help disambiguate class overlap [39].

In addition to these two problems, conventional approaches to zero-shot learning are also limited in *exploiting multiple intermediate semantic representations*. Each representation (or semantic 'view') may contain complementary information—useful for distinguishing different classes in different ways. While both visual attributes [9], [15], [27], [31] and linguistic semantic representations such as word vectors [11], [32], [44] have been independently exploited successfully, it remains unattempted and non-trivial to synergistically exploit multiple semantic views. This is because they are often of very different dimensions and types and each suffers from different domain shift effects discussed above.

In this paper, we propose to solve the projection domain shift problem using transductive multi-view embedding. The transductive setting means using the unlabelled test data to improve generalisation accuracy. In our framework, each unlabelled target class instance is represented by multiple views: its low-level feature view and its (biased) projections in multiple semantic spaces (visual attribute space and word space in this work). To rectify the projection domain shift between auxiliary and target datasets, we introduce a multi-view semantic space alignment process to correlate different semantic views and the low-level feature view by projecting them onto a common latent embedding space learned using multi-view canonical correlation analysis (CCA) [17]. The intuition is that when the biased target data projections (semantic representations) are correlated/aligned with their (unbiased) low-level feature representations, the bias/projection domain shift is alleviated. The effects of this process on projection domain shift are illustrated by Fig. 1c, where after alignment, the target Pig class prototype is much closer to its member points in this embedding space. Furthermore, after exploiting the complementarity of different low-level feature and semantic views synergistically in the common embedding space, different target classes become more compact and more separable (see Fig. 6d for an example), making the subsequent zero-shot recognition a much easier task.

Even with the proposed transductive multi-view embedding framework, the prototype sparsity problem remains—instead of one prototype per class, a handful are now available depending on how many views are embedded, which are still sparse. Our solution is to pose this as a semi-supervised learning [57] problem: prototypes in each view are treated as labelled 'instances', and we exploit the manifold structure of the unlabelled data distribution in each view in the embedding space via label propagation on a graph. To this end, we introduce a novel transductive multi-view hypergraph label propagation (TMV-HLP) algorithm for recognition. The core in our TMV-HLP algorithm is a new *distributed representation* of graph structure termed heterogeneous hypergraph which allows us to exploit the complementarity of different semantic and low-level feature views, as well as the manifold structure of the target data to compensate for the impoverished supervision available from the sparse prototypes. Zero-shot learning is then performed by semi-supervised label propagation from the prototypes to the target data points within and across the graphs. The whole framework is illustrated in Fig. 2.

By combining our transductive embedding framework and the TMV-HLP zero-shot recognition algorithm, our approach generalises seamlessly when none (zero-shot), or few (N-shot) samples of the target classes are available. Uniquely it can also synergistically exploit zero + N-shot (i.e., both prototypes and labelled samples) learning. Furthermore, the proposed method enables a number of novel cross-view annotation tasks including *zero-shot class description* and *zero prototype learning*.

Our contributions. Our contributions are as follows: (1) To our knowledge, this is the first attempt to investigate and provide a solution to the projection domain shift problem in zero-shot learning. (2) We propose a transductive multi-view embedding space that not only rectifies the projection shift, but also exploits the complementarity of multiple semantic representations of visual data. (3) A novel transductive multi-view heterogeneous hypergraph label propagation algorithm is developed to improve both zero-shot and N-shot learning tasks in the embedding space and overcome the prototype sparsity problem. (4) The learned embedding space enables a number of novel cross-view annotation tasks. Extensive experiments are carried out and the results show that our approach

Authorized licensed use limited to: Nanyang Technological University. Downloaded on August 31,2020 at 23:43:09 UTC from IEEE Xplore. Restrictions apply.

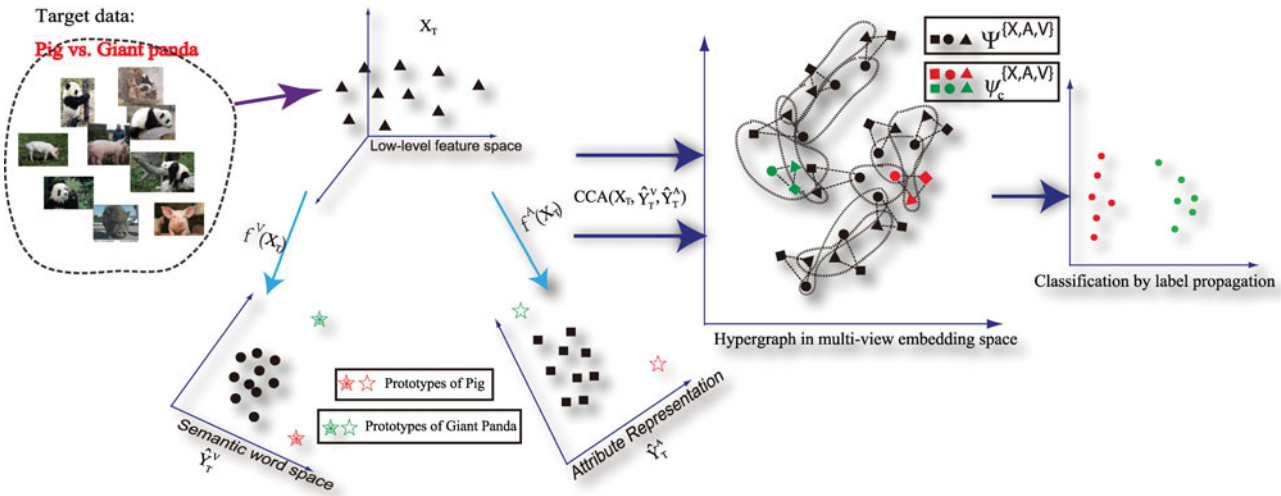


Fig. 2. The pipeline of our framework illustrated on the task of classifying unlabelled target data into two classes.

significantly outperforms existing methods for both zero-shot and N-shot recognition on three image and video benchmark datasets.

2 Related Work

Semantic spaces for zero-shot learning. To address zero-shot learning, attribute-based semantic representations have been explored for images [9], [27] and to a lesser extent videos [15], [31]. Most existing studies [1], [24], [27], [33], [34], [41], [54] assume that an exhaustive ontology of attributes has been manually specified at either the class or instance level. However, annotating attributes scales poorly as ontologies tend to be domain specific. This is despite efforts exploring augmented data-driven/latent attributes at the expense of name-ability [9], [15], [31]. To address this, semantic representations using existing ontologies and incidental data have been proposed [37], [38]. Recently, *word vector* approaches based on distributed language representations have gained popularity. In this case a word space is extracted from linguistic knowledge bases e.g., Wikipedia by natural language processing models such as [4], [32]. The language model is then used to project each class' textual name into this space. These projections can be used as prototypes for zero-shot learning [11], [44]. Importantly, regardless of the semantic spaces used, existing methods focus on either designing better semantic spaces or how to best learn the projections. The former is orthogonal to our work—any semantic spaces can be used in our framework and better ones would benefit our model. For the latter, no existing work has identified or addressed the projection domain shift problem.

Transductive zero-shot learning was considered by Fu et al. [13], [15] who introduced a generative model to for user-defined and latent attributes. A simple transductive zero-shot learning algorithm is proposed: averaging the prototype's k-nearest neighbours (KNN) to exploit the test data attribute distribution. Rohrbach et al. [36] proposed a more elaborate transductive strategy, using graph-based label propagation to exploit the manifold structure of the test data. These studies effectively transform the ZSL task into a transductive semi-supervised learning task [57] with prototypes providing the few labelled instances. Nevertheless, these studies and this paper (as with most previous work [27], [28], [37]) only consider recognition among the novel classes: unifying zero-shot with supervised learning remains an open challenge [44].

Domain adaptation. Domain adaptation methods attempt to address the domain shift problems that occur when the assumption that the source and target instances are drawn from the same distribution is violated. Methods have been derived for both classification [8], [10] and regression [45], and both with [8] and without [10] requiring label information in the target task. Our zero-shot learning problem means that most of supervised domain adaptation methods are inapplicable. Our projection domain shift problem differs from the conventional domain shift problems in that (i) it is indirectly observed in terms of the projection shift rather than the feature distribution shift, and (ii) the source domain classes and target domain classes are completely different and could even be unrelated. Consequently our domain adaptation method differs significantly from the existing unsupervised ones such as [10] in that our method relies on correlating different representations of the unlabelled target data in a multi-view embedding space.

Learning multi-view embedding spaces. Relating low-level feature and semantic views of data has been exploited in visual recognition and cross-modal retrieval. Most existing work [17], [23], [43], [51] focuses on modelling images/videos with associated text (e.g. tags on Flickr/YouTube). Multi-view CCA is often exploited to provide unsupervised fusion of different modalities. However, there are two fundamental differences between previous multi-view embedding work and ours: (1) Our embedding space is transductive, that is, learned from unlabelled target data from which all semantic views are estimated by projection rather than being the original views. These projected views thus have the projection domain shift problem that the previous work does not have. (2) The objectives are different: we aim to rectify the projection domain shift problem via the embedding in order to perform better recognition and annotation while previous studies target primarily cross-modal retrieval. Note that although in this work, the

Authorized licensed use limited to: Nanyang Technological University. Downloaded on August 31,2020 at 23:43:09 UTC from IEEE Xplore. Restrictions apply.

popular CCA model is adopted for multi-view embedding, other models [40], [49] could also be considered.

Graph-based label propagation. In most previous zero-shot learning studies (e.g., direct attribute prediction (DAP) [28]), the available knowledge (a single prototype per target class) is very limited. There has therefore been recent interest in additionally exploiting the unlabelled target data distribution by transductive learning [15], [36]. However, both [36] and [15] suffer from the projection domain shift problem, and are unable to effectively exploit multiple semantic representations/views. In contrast, after embedding, our framework synergistically integrates the low-level feature and semantic representations by transductive multi-view hypergraph label propagation. Moreover, TMV-HLP generalises beyond zero-shot to N-shot learning if labelled instances are available for the target classes.

In a broader context, graph-based label propagation [55] in general, and classification on multi-view graphs (C-MG) in particular are well-studied in semi-supervised learning. Most C-MG solutions are based on the seminal work of Zhou et al. [56] which generalises spectral clustering from a single graph to multiple graphs by defining a mixture of random walks on multiple graphs. In the embedding space, instead of constructing local neighbourhood graphs for each view independently (e.g. TMV-BLP [14]), this paper proposes a *distributed representation* of pairwise similarity using heterogeneous hypergraphs. Such a distributed heterogeneous hypergraph representation can better explore the higher-order relations between any two nodes of different complementary views, and thus give rise to a more robust pairwise similarity graph and lead to better classification performance than previous multi-view graph methods [14], [56]. Hypergraphs have been used as an effective tool to align multiple data/feature modalities in data mining [29], multimedia [12] and computer vision [19], [30] applications. A hypergraph is the generalisation of a 2-graph with edges connecting many nodes/vertices, versus connecting two nodes in conventional 2-graphs. This makes it cope better with noisy nodes and thus achieve better performance than conventional graphs [12], [21], [22]. The only existing work considering hypergraphs for multi-view data modelling is [19]. Different from the multi-view hypergraphs proposed in [19] which are homogeneous, that is, constructed in each view independently, we construct a multi-view heterogeneous hypergraph: using the nodes from one view as query nodes to compute hyperedges in another view. This novel graph structure better exploits the complementarity of different views in the common embedding space.

3 Learning a Transductive Multi-View Embedding Space

A schematic overview of our framework is given in Fig. 2. We next introduce some notation and assumptions, followed by the details of how to map image features into each semantic space, and how to map multiple spaces into a common embedding space.

3.1 Problem Setup

We have c_S source/auxiliary classes with n_S instances $S = \{X_S, Y_S^i, \mathbf{z}_S\}$ and c_T target classes $T = \{X_T, Y_T^i, \mathbf{z}_T\}$ with n_T instances. $X_S \in \mathbb{R}^{n_s \times t}$ and $X_T \in \mathbb{R}^{n_T \times t}$ denote the $t-$dimensional low-level feature vectors of auxiliary and target instances respectively. $\mathbf{z}_S$ and $\mathbf{z}_T$ are the auxiliary and target class label vectors. We assume the auxiliary and target classes are disjoint: $\mathbf{z}_S \cap \mathbf{z}_T = \varnothing$. We have I different types of semantic representations; Y_S^i and Y_T^i represent the i-th type of m_i-dimensional semantic representation for the auxiliary and target datasets respectively; so $Y_S^i \in \mathbb{R}^{n_S \times m_i}$ and $Y_T^i \in \mathbb{R}^{n_T \times m_i}$. Note that for the auxiliary dataset, Y_S^i is given as each data point is labelled. But for the target dataset, Y_T^i is missing, and its prediction $\hat{Y}_T^i$ from X_T is used instead. As we shall see, this is obtained using a projection function learned from the auxiliary dataset. The problem of zero-shot learning is to estimate $\mathbf{z}_T$ given X_T and $\hat{Y}_T^i$.

Without any labelled data for the target classes, external knowledge is needed to represent what each target class looks like, in the form of class prototypes. Specifically, each target class c has a pre-defined class-level semantic prototype $\mathbf{y}_c^i$ in each semantic view i. In this paper, we consider two types of intermediate semantic representation (i.e. $I = 2$)—attributes and word vectors, which represent two distinct and complementary sources of information. We use $\mathcal{X}$, $\mathcal{A}$ and $\mathcal{V}$ to denote the low-level feature, attribute and word vector spaces respectively. The attribute space $\mathcal{A}$ is typically manually defined using a standard ontology. For the word vector space $\mathcal{V}$, we employ the state-of-the-art skip-gram neural network model [32] trained on all English Wikipedia articles.[1] Using this learned model, we can project the textual name of any class into the $\mathcal{V}$ space to get its word vector representation. Unlike semantic attributes, it is a 'free' semantic representation in that this process does not need any human annotation. We next address how to project low-level features into these two spaces.

3.2 Learning the Projections of Semantic Spaces

Mapping images and videos into semantic space i requires a projection function $f^i : \mathcal{X} \rightarrow \mathcal{Y}^i$. This is typically realised by classifier [27] or regressor [44]. In this paper, using the auxiliary set S, we train support vector classifiers $f^{\mathcal{A}}(\cdot)$ and support vector regressors $f^{\mathcal{V}}(\cdot)$ for each dimension[2] of the auxiliary class attribute and word vectors respectively. Then the target class instances X_T have the semantic projections: $\hat{Y}_T^{\mathcal{A}} = f^{\mathcal{A}}(X_T)$ and $\hat{Y}_T^{\mathcal{V}} = f^{\mathcal{V}}(X_T)$. However, these predicted intermediate semantics have the projection domain shift problem illustrated in Fig. 1. To address this, we learn a transductive multi-view semantic embedding space to align the semantic projections with the low-level features of target data.

3.3 Transductive Multi-View Embedding

We introduce a multi-view semantic alignment (i.e. transductive multi-view embedding) process to correlate target

1. To 13 Feb. 2014, it includes 2.9 billion words from a 4.33 million-words vocabulary (single and bi/tri-gram words).

2. Note that methods for learning projection functions for all dimensions jointly exist (e.g. [11]) and can be adopted in our framework.

Authorized licensed use limited to: Nanyang Technological University. Downloaded on August 31,2020 at 23:43:09 UTC from IEEE Xplore. Restrictions apply.

instances in different (biased) semantic view projections with their low-level feature view. This process alleviates the projection domain shift problem, as well as providing a common space in which heterogeneous views can be directly compared, and their complementarity exploited (Section 4). To this end, we employ multi-view canonical correlation analysis for n_V views, with the target data representation in view i denoted Φ^i, a $n_T \times m_i$ matrix. Specifically, we project three views of each target class instance $f^{\mathcal{A}}(X_T)$, $f^{\mathcal{V}}(X_T)$ and X_T (i.e. $n_V = I + 1 = 3$) into a shared embedding space. The three projection functions W^i are learned by

$$\begin{aligned} \min_{\{\mathrm{W^i}\}_{\mathrm{i=1}}^{\mathrm{n_V}}} \quad & \sum_{i,j=1}^{n_V} \quad Trace(W^i \Sigma_{ij} W^j) \\ = & \sum_{i,j=1}^{n_V} \quad \| \Phi^i W^i - \Phi^j W^j \|_F^2 \qquad (1) \\ \text{s.t.} \quad & [W^i]^T \Sigma_{ii} W^i = I \quad [\mathbf{w}_k^i]^T \Sigma_{ij} \mathbf{w}_l^j = 0 \\ & i \neq j, k \neq l \quad i,j = 1,\ldots,n_V \quad k,l = 1,\ldots,n_T, \end{aligned}$$

where W^i is the projection matrix which maps the view Φ^i ($\in \mathbb{R}^{n_T \times m_i}$) into the embedding space and $\mathbf{w}_k^i$ is the kth column of W^i. Σ_{ij} is the covariance matrix between Φ^i and Φ^j. The optimisation problem above is multi-convex as long as Σ_{ii} are non-singular. The local optimum can be easily found by iteratively maximising over each W^i given the current values of the other coefficients as detailed in [18].

The dimensionality m_e of the embedding space is the sum of the input view dimensions, i.e. $m_e = \sum_{i=1}^{n_V} m_i$, so $W^i \in \mathbb{R}^{m_i \times m_e}$. Compared to the classic approach to CCA [18] which projects to a lower dimension space, this retains all the input information including uncorrelated dimensions which may be valuable and complementary. Side-stepping the task of explicitly selecting a subspace dimension, we use a more stable and effective soft-weighting strategy to implicitly emphasise significant dimensions in the embedding space. This can be seen as a generalisation of standard dimension reducing approaches to CCA, which implicitly define a binary weight vector that activates a subset of dimensions and deactivates others. Since the importance of each dimension is reflected by its corresponding eigenvalue [17], [18], we use the eigenvalues to weight the dimensions and define a *weighted embedding space* Γ:

$$\Psi^i = \Phi^i W^i [D^i]^\lambda = \Phi^i W^i \tilde{D}^i, \qquad (2)$$

where D^i is a diagonal matrix with its diagonal elements set to the eigenvalues of each dimension in the embedding space, λ is a power weight of D^i and empirically set to 4 [17], and Ψ^i is the final representation of the target data from view i in Γ. We index the $n_V = 3$ views as $i \in \{\mathcal{X}, \mathcal{V}, \mathcal{A}\}$ for notational convenience. The same formulation can be used if more views are available.

Similarity in the embedding space. The choice of similarity metric is important for high-dimensional embedding spaces. For the subsequent recognition and annotation tasks, we compute cosine distance in Γ by l_2 normalisation: normalising any vector $\boldsymbol{\psi}_k^i$ (the kth row of Ψ^i) to unit length (i.e. $\| \boldsymbol{\psi}_k^i \|_2 = 1$). Cosine similarity is given by the inner product of any two vectors in Γ.

4 Recognition by Multi-View Hypergraph Label Propagation

For zero-shot recognition, each target class c to be recognised has a semantic prototype $\mathbf{y}_c^i$ in each view i. Similarly, we have three views of each unlabelled instance $f^{\mathcal{A}}(X_T)$, $f^{\mathcal{V}}(X_T)$ and X_T. The class prototypes are expected to be the mean of the distribution of their class in semantic space, since the projection function f^i is trained to map instances to their class prototype in each semantic view. To exploit the learned space Γ to improve recognition, we project both the unlabelled instances and the prototypes into the embedding space.[3] The prototypes $\mathbf{y}_c^i$ for views $i \in \{\mathcal{A}, \mathcal{V}\}$ are projected as $\boldsymbol{\psi}_c^i = \mathbf{y}_c^i W^i \tilde{D}^i$. So we have $\boldsymbol{\psi}_c^{\mathcal{A}}$ and $\boldsymbol{\psi}_c^{\mathcal{V}}$ for the attribute and word vector prototypes of each target class c in Γ. In the absence of a prototype for the (non-semantic) low-level feature view $\mathcal{X}$, we synthesise it as $\boldsymbol{\psi}_c^{\mathcal{X}} = (\boldsymbol{\psi}_c^{\mathcal{A}} + \boldsymbol{\psi}_c^{\mathcal{V}})/2$. If labelled data is available (i.e., N-shot case), these are also projected into the space. Recognition could now be achieved using NN classification with the embedded prototypes/N-shots as labelled data. However, this does not effectively exploit the multi-view complementarity, and suffers from labelled data (prototype) sparsity. To solve this problem, we next introduce a unified framework to fuse the views and transductively exploit the manifold structure of the unlabelled target data to perform zero-shot and N-shot learning.

Most or all of the target instances are unlabelled, so classification based on the sparse prototypes is effectively a semi-supervised learning problem [57]. We leverage graph-based semi-supervised learning to exploit the manifold structure of the unlabelled data transductively for classification. This differs from the conventional approaches such as direct attribute prediction [28] or NN, which too simplistically assume that the data distribution for each target class is Gaussian or multinomial. However, since our embedding space contains multiple projections of the target data and prototypes, it is hard to define a single graph that synergistically exploits the manifold structure of all views. We therefore construct multiple graphs within and across views in a transductive multi-view hypergraph label propagation model. Specifically, we construct the heterogeneous hypergraphs across views to combine/align the different manifold structures so as to enhance the robustness and exploit the complementarity of different views. Semi-supervised learning is then performed by propagating the labels from the sparse prototypes (zero-shot) and/or the few labelled target instances (N-shot) to the unlabelled data using random walk on the graphs.

4.1 Constructing Heterogeneous Hypergraphs

Pairwise node similarity. The key idea behind a hypergraph based method is to group similar data points, represented as vertices/nodes on a graph, into hyperedges, so that the

3. Before being projected into Γ, the prototypes are updated by semi-latent zero shot learning algorithm in [15].

Authorized licensed use limited to: Nanyang Technological University. Downloaded on August 31,2020 at 23:43:09 UTC from IEEE Xplore. Restrictions apply.

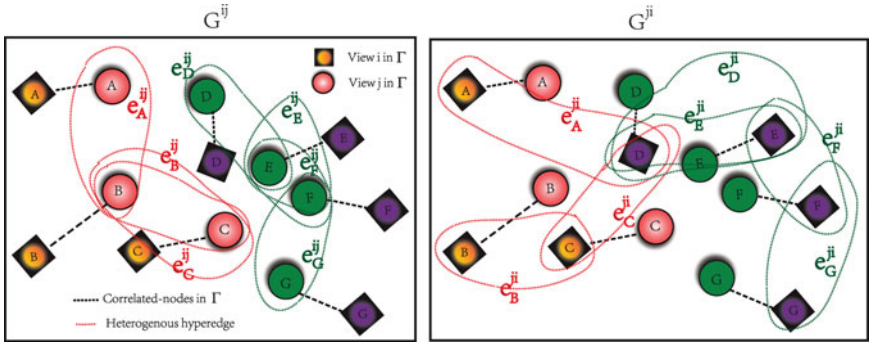


Fig. 3. An example of constructing heterogeneous hypergraphs. Suppose in the embedding space, we have 14 nodes belonging to seven data points A, B, C, D, E, F and G of two views—view i (rectangle) and view j (circle). Data points A,B,C and D,E,F,G belong to two different classes—red and green respectively. The multi-view semantic embedding maximises the correlations (connected by black dash lines) between the two views of the same node. Two hypergraphs are shown ($\mathcal{G}^{ij}$ at the left and $\mathcal{G}^{ji}$ at the right) with the heterogeneous hyperedges drawn with red/green dash ovals for the nodes of red/green classes. Each hyperedge consists of two most similar nodes to the query node.

subsequent computation is less sensitive to individual noisy nodes. With the hyperedges, the pairwise similarity between two data points are measured as the similarity between the two hyperedges that they belong to, instead of that between the two nodes only. For both forming hyperedges and computing the similarity between two hyperedges, pairwise similarity between two graph nodes needs to be defined. In our embedding space Γ, each data point in each view defines a node, and the similarity between any pair of nodes is:

$$\omega(\boldsymbol{\psi}_k^i, \boldsymbol{\psi}_l^j) = \exp\left(\frac{<\boldsymbol{\psi}_k^i, \boldsymbol{\psi}_l^j>^2}{\varpi}\right), \tag{3}$$

where $<\boldsymbol{\psi}_k^i, \boldsymbol{\psi}_l^j>^2$ is the square of inner product between the ith and jth projections of nodes k and l with a bandwidth parameter ϖ.[4] Note that Eq. (3) defines the pairwise similarity between any two nodes within the same view ($i = j$) or across different views ($i \neq j$).

Heterogeneous hyperedges. Given the multi-view projections of the target data, we aim to construct a set of across-view heterogeneous hypergraphs

$$\mathcal{G}^c = \{\mathcal{G}^{ij} \,|\, i, j \in \{\mathcal{X}, \mathcal{V}, \mathcal{A}\}, i \neq j\}, \tag{4}$$

where $\mathcal{G}^{ij} = \{\Psi^i, E^{ij}, \Omega^{ij}\}$ denotes the cross-view heterogeneous hypergraph from view i to j (in that order) and Ψ^i is the node set in view i; E^{ij} is the hyperedge set and Ω^{ij} is the pairwise node similarity set for the hyperedges. Specifically, we have the hyperedge set $E^{ij} = \{e^{ij}_{\boldsymbol{\psi}_k^i} \,|\, i \neq j, k = 1, \ldots n_T + c_T\}$ where each hyperedge $e^{ij}_{\boldsymbol{\psi}_k^i}$ includes the nodes [5] in view j that are the most similar to node $\boldsymbol{\psi}_k^i$ in view i and the similarity set $\Omega^{ij} = \{\Delta^{ij}_{\boldsymbol{\psi}_k^i} = \{\omega(\boldsymbol{\psi}_k^i, \boldsymbol{\psi}_l^j)\} \,|\, i \neq j, \boldsymbol{\psi}_l^j \in e^{ij}_{\boldsymbol{\psi}_k^i} k = 1, \ldots n_T + c_T\}$ where $\omega(\boldsymbol{\psi}_k^i, \boldsymbol{\psi}_l^j)$ is computed using Eq. (3).

4. Most previous work [36], [56] sets ϖ by cross-validation. Inspired by [26], a simpler strategy for setting ϖ is adopted: $\varpi \approx \underset{k,l=1,\cdots,n}{\text{median}} < \boldsymbol{\psi}_k^i, \boldsymbol{\psi}_l^j >^2$ in order to have roughly the same number of similar and dissimilar sample pairs. This makes the edge weights from different pairs of nodes more comparable.

5. Both the unlabelled samples and the prototypes are nodes.

We call $\boldsymbol{\psi}_k^i$ the query node for hyperedge $e^{ij}_{\boldsymbol{\psi}_k^i}$, since the hyperedge $e^{ij}_{\boldsymbol{\psi}_k^i}$ intrinsically groups all nodes in view j that are most similar to node $\boldsymbol{\psi}_k^i$ in view i. Similarly, $\mathcal{G}^{ji}$ can be constructed by using nodes from view j to query nodes in view i. Therefore given three views, we have six across view/heterogeneous hypergraphs. Fig. 3 illustrates two heterogeneous hypergrahs constructed from two views. Interestingly, our way of defining hyperedges naturally corresponds to the star expansion [46] where the query node (i.e. $\boldsymbol{\psi}_k^i$) is introduced to connect each node in the hyperedge $e^{ij}_{\boldsymbol{\psi}_k^i}$.

Similarity strength of hyperedge. For each hyperedge $e^{ij}_{\boldsymbol{\psi}_k^i}$, we measure its similarity strength by using its query nodes $\boldsymbol{\psi}_k^i$. Specifically, we use the weight $\delta^{ij}_{\boldsymbol{\psi}_k^i}$ to indicate the similarity strength of nodes connected within the hyperedge $e^{ij}_{\boldsymbol{\psi}_k^i}$. Thus, we define $\delta^{ij}_{\boldsymbol{\psi}_k^i}$ based on the mean similarity of the set $\Delta^{ij}_{\boldsymbol{\psi}_k^i}$ for the hyperedge

$$\delta^{ij}_{\boldsymbol{\psi}_k^i} = \frac{1}{|\, e^{ij}_{\boldsymbol{\psi}_k^i} \,|} \sum_{\omega(\boldsymbol{\psi}_k^i, \boldsymbol{\psi}_l^j) \in \Delta^{ij}_{\boldsymbol{\psi}_k^i}, \boldsymbol{\psi}_l^j \in e^{ij}_{\boldsymbol{\psi}_k^i}} \omega(\boldsymbol{\psi}_k^i, \boldsymbol{\psi}_l^j), \tag{5}$$

where $|\, e^{ij}_{\boldsymbol{\psi}_k^i} \,|$ is the cardinality of hyperedge $e^{ij}_{\boldsymbol{\psi}_k^i}$.

In the embedding space Γ, similarity sets $\Delta^{ij}_{\boldsymbol{\psi}_k^i}$ and $\Delta^{ij}_{\boldsymbol{\psi}_l^i}$ can be compared. Nevertheless, these sets come from heterogeneous views and have varying scales. Thus some normalisation steps are necessary to make the two similarity sets more comparable and the subsequent computation more robust. Specifically, we extend zero-score normalisation to the similarity sets: (a) We assume $\forall \Delta^{ij}_{\boldsymbol{\psi}_k^i} \in \Omega^{ij}$ and $\Delta^{ij}_{\boldsymbol{\psi}_k^i}$ should follow Gaussian distribution. Thus, we enforce zero-score normalisation to $\Delta^{ij}_{\boldsymbol{\psi}_k^i}$. (b) We further assume that the retrieved similarity set Ω^{ij} between all the queried nodes $\boldsymbol{\psi}_k^i$ ($l = 1, \ldots n_T$) from view i and $\boldsymbol{\psi}_l^j$ should also follow Gaussian distributions. So we again enforce Gaussian distribution to the pairwise similarities between $\boldsymbol{\psi}_l^j$ and all query nodes from view i by zero-score normalisation. (c) We select the first K

Authorized licensed use limited to: Nanyang Technological University. Downloaded on August 31,2020 at 23:43:09 UTC from IEEE Xplore. Restrictions apply.

highest values from $\Delta^{ij}_{\boldsymbol{\psi}^i_k}$ as new similarity set $\bar{\Delta}^{ij}_{\boldsymbol{\psi}^i_k}$ for hyperedge $e^{ij}_{\boldsymbol{\psi}^i_k}$. $\bar{\Delta}^{ij}_{\boldsymbol{\psi}^i_k}$ is then used in Eq. (5) in place of $\Delta^{ij}_{\boldsymbol{\psi}^i_k}$. These normalisation steps aim to compute a more robust similarity between each pair of hyperedges.

Computing similarity between hyperedges. With the hypergraph, the similarity between two nodes is computed using their hyperedges $e^{ij}_{\boldsymbol{\psi}^i_k}$. Specifically, for each hyperedge there is an associated incidence matrix $H^{ij} = (h(\boldsymbol{\psi}^j_l, e^{ij}_{\boldsymbol{\psi}^i_k}))_{(n_T+c_T)\times|E^{ij}|}$ where

$$h\big(\boldsymbol{\psi}^j_l, e^{ij}_{\boldsymbol{\psi}^i_k}\big) = \begin{cases} 1 & if \boldsymbol{\psi}^j_l \in e^{ij}_{\boldsymbol{\psi}^i_k}, \\ 0 & otherwise. \end{cases} \tag{6}$$

To take into consideration the similarity strength between hyperedge and query node, we extend the binary valued hyperedge incidence matrix H^{ij} to soft-assigned incidence matrix $SH^{ij} = \big(sh\big(\boldsymbol{\psi}^j_l, e^{ij}_{\boldsymbol{\psi}^i_k}\big)\big)_{(n_T+c_T)\times|E^{ij}|}$ as follows:

$$sh\big(\boldsymbol{\psi}^j_l, e^{ij}_{\boldsymbol{\psi}^i_k}\big) = \delta^{ij}_{\boldsymbol{\psi}^i_k} \cdot \omega\big(\boldsymbol{\psi}^i_k, \boldsymbol{\psi}^j_l\big) \cdot h\big(\boldsymbol{\psi}^j_l, e^{ij}_{\boldsymbol{\psi}^i_k}\big). \tag{7}$$

This soft-assigned incidence matrix is the product of three components: (1) the weight $\delta_{\boldsymbol{\psi}^i_k}$ for hyperedge $e^{ij}_{\boldsymbol{\psi}^i_k}$; (2) the pairwise similarity computed using queried node $\boldsymbol{\psi}^i_k$; (3) the binary valued hyperedge incidence matrix element $h(\boldsymbol{\psi}^j_l, e^{ij}_{\boldsymbol{\psi}^i_k})$. To make the values of SH^{ij} comparable among the different heterogeneous views, we apply l_2 normalisation to the soft-assigned incidence matrix values for all node incident to each hyperedge.

Now for each heterogeneous hypergraph, we can finally define the pairwise similarity between any two nodes or hyperedges. Specifically for $\mathcal{G}^{ij}$, the similarity between the oth and lth nodes is

$$\omega^{ij}_c\big(\boldsymbol{\psi}^j_o, \boldsymbol{\psi}^j_l\big) = \sum_{e^{ij}_{\boldsymbol{\psi}^i_k} \in E^{ij}} sh\big(\boldsymbol{\psi}^j_o, e^{ij}_{\boldsymbol{\psi}^i_k}\big) \cdot sh\big(\boldsymbol{\psi}^j_l, e^{ij}_{\boldsymbol{\psi}^i_k}\big). \tag{8}$$

With this pairwise hyperedge similarity, the hypergraph definition is now complete. Empirically, given a node, other nodes on the graph that have very low similarities will have very limited effects on its label. Thus, to reduce computational cost, we only use the K-nearest-neighbour[6] nodes of each node [57] for the subsequent label propagation step.

The advantages of heterogeneous hypergraphs. We argue that the pairwise similarity of heterogeneous hypergraphs is a distributed representation [2]. To explain it, we can use star extension [46] to extend a hypergraph into a 2-graph. For each hyperedge $e^{ij}_{\boldsymbol{\psi}^i_k}$, the query node $\boldsymbol{\psi}^i_k$ is used to compute the pairwise similarity $\Delta^{ij}_{\boldsymbol{\psi}^i_k}$ of all the nodes in view j. Each hyperedge can thus define a hyper-plane by categorising the nodes in view j into two groups: strong and weak similarity group regarding to query node $\boldsymbol{\psi}^i_k$. In other words, the hyperedge set E^{ij} is multi-clustering with linearly separated regions (by each hyperplane) per classes. Since the final pairwise similarity in Eq. (8) can be represented by a set of similarity weights computed by hyperedge, and such weights are not mutually exclusive and are statistically independent, we consider the heterogeneous hypergraph a distributed representation. The advantage of having a distributed representation has been studied by Watts and Strogatz [52], [53] which shows that such a representation gives rise to better convergence rates and better clustering abilities. In contrast, the homogeneous hypergraphs adopted by previous work [12], [19], [22] does not have this property which makes them less robust against noise. In addition, fusing different views in the early stage of graph construction potentially can lead to better exploitation of the complementarity of different views. However, it is worth pointing out that (1) The reason we can query nodes across views to construct heterogeneous hypergraph is because we have projected all views in the same embedding space in the first place. (2) Hypergraphs typically gain robustness at the cost of losing discriminative power—it essentially blurs the boundary of different clusters/classes by taking average over hyperedges. A typical solution is to fuse hypergraphs with 2-graphs [12], [19], [29], which we adopt here as well.

4.2 Label Propagation by Random Walk

Now we have two types of graphs: heterogeneous hypergraphs $\mathcal{G}^c = \{\mathcal{G}^{ij}\}$ and 2-graphs[7] $\mathcal{G}^p = \{\mathcal{G}^i\}$. Given three views ($n_V = 3$), we thus have nine graphs in total (six hypergraphs and three 2-graphs). To classify the unlabelled nodes, we need to propagate label information from the prototype nodes across the graph. Such semi-supervised label propagation [56], [57] has a closed-form solution and is explained as a random walk. A random walk requires pairwise transition probability for nodes k and l. We obtain this by aggregating the information from all graphs $\mathcal{G} = \{\mathcal{G}^p; \mathcal{G}^c\}$,

$$\begin{aligned} p(k \to l) = &\sum_{i \in \{\mathcal{X},\mathcal{V},\mathcal{A}\}} p(k \to l \,|\, \mathcal{G}^i) \cdot p(\mathcal{G}^i \,|\, k) \\ &+ \sum_{i,j \in \{\mathcal{X},\mathcal{V},\mathcal{A}\}, i \neq j} p(k \to l \,|\, \mathcal{G}^{ij}) \cdot p(\mathcal{G}^{ij} \,|\, k), \end{aligned} \tag{9}$$

where

$$p(k \to l \,|\, \mathcal{G}^i) = \frac{\omega^i_p\big(\boldsymbol{\psi}^i_k, \boldsymbol{\psi}^i_l\big)}{\sum_o \omega^i_p\big(\boldsymbol{\psi}^i_k, \boldsymbol{\psi}^i_o\big)}, \tag{10}$$

and

$$p(k \to l \,|\, \mathcal{G}^{ij}) = \frac{\omega^{ij}_c\big(\boldsymbol{\psi}^j_k, \boldsymbol{\psi}^j_l\big)}{\sum_o \omega^{ij}_c\big(\boldsymbol{\psi}^j_k, \boldsymbol{\psi}^j_o\big)},$$

and then the posterior probability to choose graph $\mathcal{G}^i$ at projection/node $\boldsymbol{\psi}^i_k$ will be:

$$p(\mathcal{G}^i|k) = \frac{\pi(k|\mathcal{G}^i)p(\mathcal{G}^i)}{\sum_i \pi(k|\mathcal{G}^i)p(\mathcal{G}^i) + \sum_{ij} \pi(k|\mathcal{G}^{ij})p(\mathcal{G}^{ij})}, \tag{11}$$

$$p(\mathcal{G}^{ij}|k) = \frac{\pi(k|\mathcal{G}^{ij})p(\mathcal{G}^{ij})}{\sum_i \pi(k|\mathcal{G}^i)p(\mathcal{G}^i) + \sum_{ij} \pi(k|\mathcal{G}^{ij})p(\mathcal{G}^{ij})}, \tag{12}$$

6. $K = 30$. It can be varied from $10 \sim 50$ with little effect in our experiments.

7. That is the K-nearest-neighbour graph of each view in Γ [14].

Authorized licensed use limited to: Nanyang Technological University. Downloaded on August 31,2020 at 23:43:09 UTC from IEEE Xplore. Restrictions apply.

where $p(\mathcal{G}^i)$ and $p(\mathcal{G}^{ij})$ are the prior probability of graphs $\mathcal{G}^i$ and $\mathcal{G}^{ij}$ in the random walk. This probability expresses prior expectation about the informativeness of each graph. The sameBayesian model averaging [14] can be used here to estimate these prior probabilities. However, the computational cost is combinatorially increased with the number of views; and it turns out the prior is not critical to the results of our framework. Therefore, uniform prior is used in our experiments.

The stationary probabilities for node k in $\mathcal{G}^i$ and $\mathcal{G}^{ij}$ are

$$\pi(k|\mathcal{G}^i) = \frac{\sum_l \omega_p^i(\boldsymbol{\psi}_k^i, \boldsymbol{\psi}_l^i)}{\sum_o \sum_l \omega_p^i(\boldsymbol{\psi}_k^i, \boldsymbol{\psi}_o^i)}, \tag{13}$$

$$\pi(k|\mathcal{G}^{ij}) = \frac{\sum_l \omega_c^{ij}(\boldsymbol{\psi}_k^j, \boldsymbol{\psi}_l^j)}{\sum_k \sum_o \omega_c^{ij}(\boldsymbol{\psi}_k^j, \boldsymbol{\psi}_o^j)}. \tag{14}$$

Finally, the stationary probability across the multi-view hypergraph is computed as:

$$\pi(k) = \sum_{i \in \{\mathcal{X},\mathcal{V},\mathcal{A}\}} \pi(k|\mathcal{G}^i) \cdot p(\mathcal{G}^i) \tag{15}$$

$$+ \sum_{i,j \in \{\mathcal{X},\mathcal{V},\mathcal{A}\}, i \neq j} \pi(k|\mathcal{G}^{ij}) \cdot p(\mathcal{G}^{ij}). \tag{16}$$

Given the defined graphs and random walk process, we can derive our label propagation algorithm (TMV-HLP). Let P denote the transition probability matrix defined by Eq. (9) and Π the diagonal matrix with the elements $\pi(k)$ computed by Eq. (15). The Laplacian matrix $\mathcal{L}$ combines information of different views and is defined as: $\mathcal{L} = \Pi - \frac{\Pi P + P^T \Pi}{2}$. The label matrix Z for labelled N-shot data or zero-shot prototypes is defined as:

$$Z(q_k, c) = \begin{cases} 1 & q_k \in class\, c, \\ -1 & q_k \notin class\, c, \\ 0 & unknown. \end{cases} \tag{17}$$

Given the label matrix Z and Laplacian $\mathcal{L}$, label propagation on multiple graphs has the closed-form solution [56]: $\hat{Z} = \eta(\eta\Pi + \mathcal{L})^{-1}\Pi Z$ where η is a regularisation parameter.[8] Note that in our framework, both labelled target class instances and prototypes are modelled as graph nodes. Thus the difference between zero-shot and N-shot learning lies only on the initial labelled instances: Zero-shot learning has the prototypes as labelled nodes; N-shot has instances as labelled nodes; and a new condition exploiting both prototypes and N-shot together is possible. This unified recognition framework thus applies when either or both of prototypes and labelled instances are available. The computational cost of our TMV-HLP is $\mathcal{O}((c_T + n_T)^2 \cdot n_V^2 + (c_T + n_T)^3)$, where K is the number of nearest neighbours in the KNN graphs, and n_V is the number of views. It costs $\mathcal{O}((c_T + n_T)^2 \cdot n_V^2)$ to construct the heterogeneous graph, while the inverse matrix of Laplacian matrix $\mathcal{L}$ in label propagation step will take $\mathcal{O}((c_T + n_T)^3)$ computational time, which however can be further reduced to $\mathcal{O}(c_T n_T t)$ using the recent work of Fujiwara and Irie [16], where t is an iteration parameter in their paper and $t \ll n_T$.

8. It can be varied from $1-10$ with little effects in our experiments.

5 Annotation and Beyond

Our multi-view embedding space Γ bridges the semantic gap between low-level features $\mathcal{X}$ and semantic representations $\mathcal{A}$ and $\mathcal{V}$. Leveraging this cross-view mapping, annotation [17], [20], [51] can be improved and applied in novel ways. We consider three annotation tasks here:

Instance level annotation. Given a new instance u, we can describe/annotate it by predicting its attributes. The conventional solution is directly applying $\hat{\mathbf{y}}_u^{\mathcal{A}} = f^{\mathcal{A}}(\mathbf{x}_u)$ for test data $\mathbf{x}_u$ [9], [17]. However, as analysed before, this suffers from the projection domain shift problem. To alleviate this, our multi-view embedding space aligns the semantic attribute projections with the low-level features of each unlabelled instance in the target domain. This alignment can be used for image annotation in the target domain. Thus, with our framework, we can now infer attributes for any test instance via the learned embedding space Γ as $\hat{\mathbf{y}}_u^{\mathcal{A}} = \mathbf{x}_u W^{\mathcal{X}} \tilde{D}^{\mathcal{X}} [W^{\mathcal{A}} \tilde{D}^{\mathcal{A}}]^{-1}$.

Zero-shot class description. From a broader machine intelligence perspective, one might be interested to ask what are the attributes of an unseen class, based solely on the class name. Given our multi-view embedding space, we can infer the semantic attribute description of a novel class. This *zero-shot class description* task could be useful, for example, to hypothesise the zero-shot attribute prototype of a class instead of defining it by experts [27] or ontology [15]. Our transductive embedding enables this task by connecting semantic word space (i.e. naming) and discriminative attribute space (i.e. describing). Given the prototype $\mathbf{y}_c^{\mathcal{V}}$ from the name of a novel class c, we compute $\hat{\mathbf{y}}_c^{\mathcal{A}} = \mathbf{y}_c^{\mathcal{V}} W^{\mathcal{V}} \tilde{D}^{\mathcal{V}} [W^{\mathcal{A}} \tilde{D}^{\mathcal{A}}]^{-1}$ to generate the class-level attribute description.

Zero prototype learning. This task is the inverse of the previous task—to infer the name of class given a set of attributes. It could be useful, for example, to validate or assess a proposed zero-shot attribute prototype, or to provide an automated semantic-property based index into a dictionary or database. To our knowledge, this is the first attempt to evaluate the quality of a class attribute prototype because no previous work has directly and systematically linked linguistic knowledge space with visual attribute space. Specifically given an attribute prototype $\mathbf{y}_c^{\mathcal{A}}$, we can use $\hat{\mathbf{y}}_c^{\mathcal{V}} = \hat{\mathbf{y}}_c^{\mathcal{A}} W^{\mathcal{A}} \tilde{D}^{\mathcal{A}} [W^{\mathcal{V}} \tilde{D}^{\mathcal{V}}]^{-1}$ to name the corresponding class and perform retrieval on dictionary words in $\mathcal{V}$ using $\hat{\mathbf{y}}_c^{\mathcal{V}}$.

6 Experiments

6.1 Datasets and Settings

We evaluate our framework on three widely used image/video datasets: Animals with Attributes, Unstructured Social Activity Attribute (USAA), and Caltech-UCSD-Birds (CUB). *AwA* [27] consists of 50 classes of animals (30,475 images) and 85 associated class-level attributes. It has a standard source/target split for zero-shot learning with 10 classes and 6,180 images held out as the target dataset. We use the same 'hand-crafted' low-level features (RGB colour histograms, SIFT, rgSIFT, PHOG, SURF and local

Authorized licensed use limited to: Nanyang Technological University. Downloaded on August 31,2020 at 23:43:09 UTC from IEEE Xplore. Restrictions apply.

TABLE 1
Comparison with the State-of-the-Art on Zero-Shot Learning on AwA, USAA and CUB

Approach	AwA ($\mathcal{H}$ [27])	AwA ($\mathcal{O}$)	AwA ($\mathcal{O},\mathcal{D}$)	USAA	CUB ($\mathcal{O}$)	CUB ($\mathcal{F}$)
DAP	40.5([27])/41.4([28])/38.4*	51.0*	57.1*	33.2([15])/35.2*	26.2*	9.1*
IAP	27.8([27])/42.2([28])	–	–	–	–	–
M2LATM [15]***	41.3	–	–	41.9	–	–
ALE/HLE/AHLE [1]	37.4/39.0/43.5	–	–	–	–	18.0*
Mo/Ma/O/D [38]	27.0/23.6/33.0/35.7	–	–	–	–	–
PST [36]***	42.7	54.1*	62.9*	36.2*	38.3*	13.2*
[54]	48.3**	–	–	–	–	–
TMV-BLP [14]***	47.7	69.9	77.8	48.2	45.2	16.3
TMV-HLP***	**49.0**	**73.5**	**80.5**	**50.4**	**47.9**	**19.5**

*Features $\mathcal{H}$, $\mathcal{O}$, $\mathcal{D}$ and $\mathcal{F}$ represent hand-crafted, OverFeat, DeCAF, and Fisher Vector respectively. Mo, Ma, O and D represent the highest results by the mined object class-attribute associations, mined attributes, objectness as attributes and direct similarity methods used in [38] respectively. '–': no result reported. *: our implementation. **: requires additional human annotations.***: requires unlabelled data, i.e. a transductive setting.*

self-similarity histograms) released with the dataset (denoted as $\mathcal{H}$); and the same multi-kernel learning (MKL) attribute classifier from [27]. *USAA* is a video dataset [15] with 69 instance-level attributes for eight classes of complex (unstructured) social group activity videos from YouTube. Each class has around 100 training and test videos respectively. USAA provides the instance-level attributes since there are significant intra-class variations. We use the thresholded mean of instances from each class to define a binary attribute prototype as in [15]. The same setting in [15] is adopted: four classes as source and four classes as target data. We use exactly the same SIFT, MFCC and STIP low-level features for USAA as in [15]. *CUB-200-2011* [48] contains 11,788 images of 200 bird classes. This is more challenging than AwA—it is designed for fine-grained recognition and has more classes but fewer images. Each class is annotated with 312 binary attributes derived from a bird species ontology. We use 150 classes as auxiliary data, holding out 50 as test data. We extract 128 dimensional SIFT and colour histogram descriptors from regular grid of multi-scale and aggregate them into image-level feature Fisher Vectors ($\mathcal{F}$) by using 256 Gaussians, as in [1]. Colour histogram and PHOG features are also used to extract global color and texture cues from each image. Due to the recent progress on deep learning based representations, we also extract OverFeat ($\mathcal{O}$) [42][9] from AwA and CUB as an alternative to $\mathcal{H}$ and $\mathcal{F}$ respectively. In addition, DeCAF ($\mathcal{D}$) [7] is also considered for AwA.

We report absolute classification accuracy on USAA and mean accuracy for AwA and CUB for direct comparison to published results. The word vector space is trained by the model in [32] with 1,000 dimensions.

6.2 Recognition by Zero-Shot Learning

6.2.1 Comparisons with State-of-the-Art

We compare our method (TMV-HLP) with the recent state-of-the-art models that report results or can be re-implemented by us on the three datasets in Table 1. They cover a wide range of approaches on utilising semantic intermediate representation for zero-shot learning. They can be roughly categorised according to the semantic representation(s) used: DAP and IAP ([27], [28]), M2LATM [15], ALE [1], [36] and [50] use attributes only; HLE/AHLE [1] and Mo/Ma/O/D [38] use both attributes and linguistic knowledge bases (same as us); [54] uses attribute and some additional human manual annotation. Note that our linguistic knowledge base representation is in the form of word vectors, which does not incur additional manual annotation. Our method also does not exploit data-driven attributes such as M2LATM [15] and Mo/Ma/O/D [38].

Consider first the results on the most widely used AwA. Apart from the standard hand-crafted feature ($\mathcal{H}$), we consider the more powerful OverFeat deep feature ($\mathcal{O}$), and a combination of OverFeat and DeCAF ($\mathcal{O},\mathcal{D}$).[10] Table 1 shows that (1) with the same experimental settings and the same feature ($\mathcal{H}$), our TMV-HLP (49.0 percent) outperforms the best result reported so far (48.3 percent) in [54] which, unlike ours, requires additional human annotation to relabel the similarities between auxiliary and target classes. (2) With the more powerful OverFeat feature, our method achieves 73.5 percent zero-shot recognition accuracy. Even more remarkably, when both the OverFeat and DeCAF features are used in our framework, the result (see the AwA ($\mathcal{O},\mathcal{D}$) column) is 80.5 percent. Even with only 10 target classes, this is an extremely good result given that we do not have any labelled samples from the target classes. Note that this good result is not solely due to the feature strength, as the margin between the conventional DAP and our TMV-HLP is much bigger indicating that our TMV-HLP plays a critical role in achieving this result. (3) Our method is also superior to the AHLE method in [1] which also uses two semantic spaces: attribute and WordNet hierarchy. Different from our embedding framework, AHLE simply concatenates the two spaces. (4) Our method also outperforms the other alternatives of either mining other semantic knowledge bases (Mo/Ma/O/D [38]) or exploring data-driven attributes (M2LATM [15]). (5) Among all compared methods, PST [36] is the only one except ours that performs label propagation based transductive learning. It yields better results than DAP in all the experiments which essentially does nearest neighbour in the semantic space. TMV-HLP consistently beats PST in all the results shown in Table 1 thanks to our multi-view embedding. (6) Compared to our TMV-BLP model [14], the superior results of TMV-HLP

9. We use the trained model of OverFeat in [42].

10. With these two low-level feature views, there are six views in total in the embedding space.

Authorized licensed use limited to: Nanyang Technological University. Downloaded on August 31,2020 at 23:43:09 UTC from IEEE Xplore. Restrictions apply.

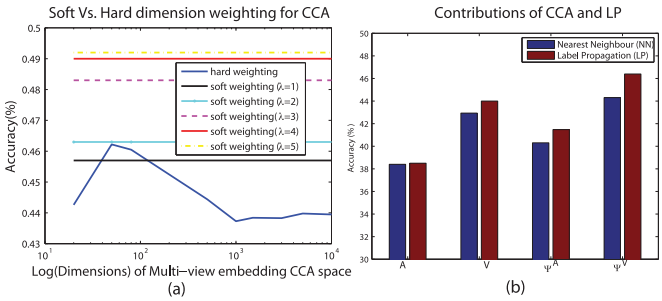


Fig. 4. (a) Comparing soft and hard dimension weighting of CCA for AwA. (b) Contributions of CCA and label propagation on AwA. $\Psi^{\mathcal{A}}$ and $\Psi^{\mathcal{V}}$ indicate the subspaces of target data from view $\mathcal{A}$ and $\mathcal{V}$ in Γ respectively; and LP is conducted on $\mathcal{G}^{\mathcal{A}}$ and $\mathcal{G}^{\mathcal{V}}$ respectively. Hand-crafted features are used in both experiments.

shows that the proposed heterogeneous hypergraph is more effective than the homogeneous 2-graphs used in TMV-BLP for zero-shot learning.

Table 1 also shows that on two very different datasets: USAA video activity, and CUB fine-grained, our TMV-HLP significantly outperforms the state-of-the-art alternatives. In particular, on the more challenging CUB, 47.9 percent accuracy is achieved on 50 classes (chance level 2 percent) using the OverFeat feature. Considering the fine-grained nature and the number of classes, this is even more impressive than the 80.5 percent result on AwA.

6.2.2 Further Evaluations

Effectiveness of soft weighting for CCA embedding. In this experiment, we compare the soft-weighting (Eq. (2)) of CCA embedding space Γ (a strategy adopted in this work) with the conventional hard-weighting strategy of selecting the number of dimensions for CCA projection. Fig. 4a shows that the performance of the hard-weighting CCA depends on the number of projection dimensions selected (blue curve). In contrast, our soft-weighting strategy uses all dimensions weighted by the CCA eigenvalues, so that the important dimensions are automatically weighted more highly. The result shows that this strategy is clearly better and it is not very sensitive to the weighting parameter λ, with choices of $\lambda > 2$ all working well.

Contributions of individual components. There are two major components in our ZSL framework: CCA embedding and label propagation. In this experiment we investigate whether both of them contribute to the strong performance. To this end, we compare the ZSL results on AwA with label propagation and without (nearest neighbour) before and after CCA embedding. In Fig. 4b, we can see that: (i) Label propagation always helps regardless whether the views have been embedded using CCA, although its effects are more pronounced after embedding. (ii) Even without label propagation, i.e. using nearest neighbour for classification, the performance is improved by the CCA embedding. However, the improvement is bigger with label propagation. This result thus suggests that both CCA embedding and label propagation are useful, and our ZSL framework works the best when both are used.

Transductive multi-view embedding. To further validate the contribution of our transductive multi-view embedding space, we split up different views with and without embedding and the results are shown in Fig. 5. In Figs. 5a and 5c, the hand-crafted feature $\mathcal{H}$ and SIFT, MFCC and STIP low-level features are used for AwA and USAA respectively, and we compare $\mathcal{V}$ vs. $\Gamma(\mathcal{X}+\mathcal{V})$, $\mathcal{A}$ vs. $\Gamma(\mathcal{X}+\mathcal{A})$ and $[\mathcal{V},\mathcal{A}]$ vs. $\Gamma(\mathcal{X}+\mathcal{V}+\mathcal{A})$ (see the caption of Fig. 5 for definitions). We use DAP for $\mathcal{A}$ and nearest neighbour for $\mathcal{V}$ and $[\mathcal{V},\mathcal{A}]$, because the prototypes of $\mathcal{V}$ are not binary vectors so DAP cannot be applied. We use TMV-HLP for $\Gamma(\mathcal{X}+\mathcal{V})$ and $\Gamma(\mathcal{X}+\mathcal{A})$ respectively. We highlight the following observations: (1) After transductive embedding, $\Gamma(\mathcal{X}+\mathcal{V}+\mathcal{A})$, $\Gamma(\mathcal{X}+\mathcal{V})$ and $\Gamma(\mathcal{X}+\mathcal{A})$ outperform $[\mathcal{V},\mathcal{A}]$, $\mathcal{V}$ and $\mathcal{A}$ respectively. This means that the transductive embedding is helpful whichever semantic space is used in rectifying the projection domain shift problem by aligning the semantic views with low-level features. (2) The results of $[\mathcal{V},\mathcal{A}]$ are higher than those of $\mathcal{A}$ and $\mathcal{V}$ individually, showing that the two semantic views are indeed complementary even with simple feature level fusion. Similarly, our TMV-HLP on all views $\Gamma(\mathcal{X}+\mathcal{V}+\mathcal{A})$ improves individual embeddings $\Gamma(\mathcal{X}+\mathcal{V})$ and $\Gamma(\mathcal{X}+\mathcal{A})$.

Embedding deep learning feature views also helps. In Fig. 5b three different low-level features are considered for AwA: hand-crafted ($\mathcal{H}$), OverFeat ($\mathcal{O}$) and DeCAF features ($\mathcal{D}$). The zero-shot learning results of each individual space are indicated as $\mathcal{V}_{\mathcal{H}}$, $\mathcal{A}_{\mathcal{H}}$, $\mathcal{V}_{\mathcal{O}}$, $\mathcal{A}_{\mathcal{O}}$, $\mathcal{V}_{\mathcal{D}}$, $\mathcal{A}_{\mathcal{D}}$ in Fig. 5b and we observe that $\mathcal{V}_{\mathcal{O}} > \mathcal{V}_{\mathcal{D}} > \mathcal{V}_{\mathcal{H}}$ and $\mathcal{A}_{\mathcal{O}} > \mathcal{A}_{\mathcal{D}} > \mathcal{A}_{\mathcal{H}}$. That is OverFeat > DeCAF > hand-crafted features. It is widely reported that deep features have better performance

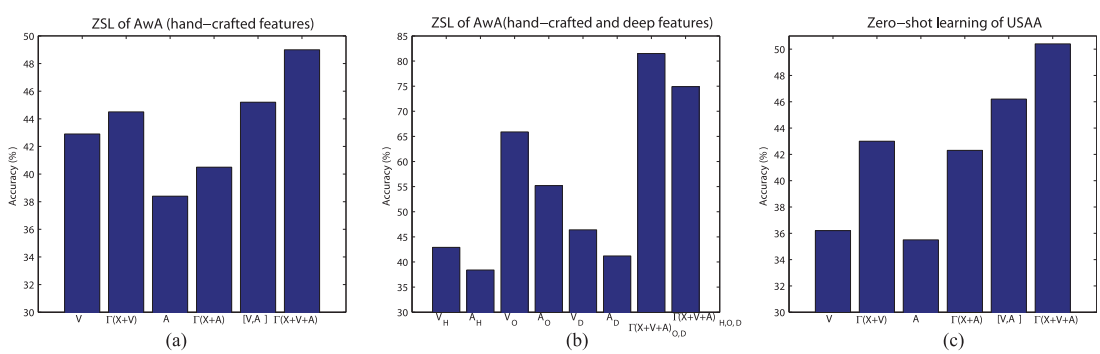


Fig. 5. Effectiveness of transductive multi-view embedding. (a) zero-shot learning on AwA using only hand-crafted features; (b) zero-shot learning on AwA using hand-crafted and deep features together; (c) zero-shot learning on USAA. $[\mathcal{V},\mathcal{A}]$ indicates the concatenation of semantic word and attribute space vectors. $\Gamma(\mathcal{X}+\mathcal{V})$ and $\Gamma(\mathcal{X}+\mathcal{A})$ mean using low-level+semantic word spaces and low-level+attribute spaces respectively to learn the embedding. $\Gamma(\mathcal{X}+\mathcal{V}+\mathcal{A})$ indicates using all three views to learn the embedding.

Authorized licensed use limited to: Nanyang Technological University. Downloaded on August 31,2020 at 23:43:09 UTC from IEEE Xplore. Restrictions apply.

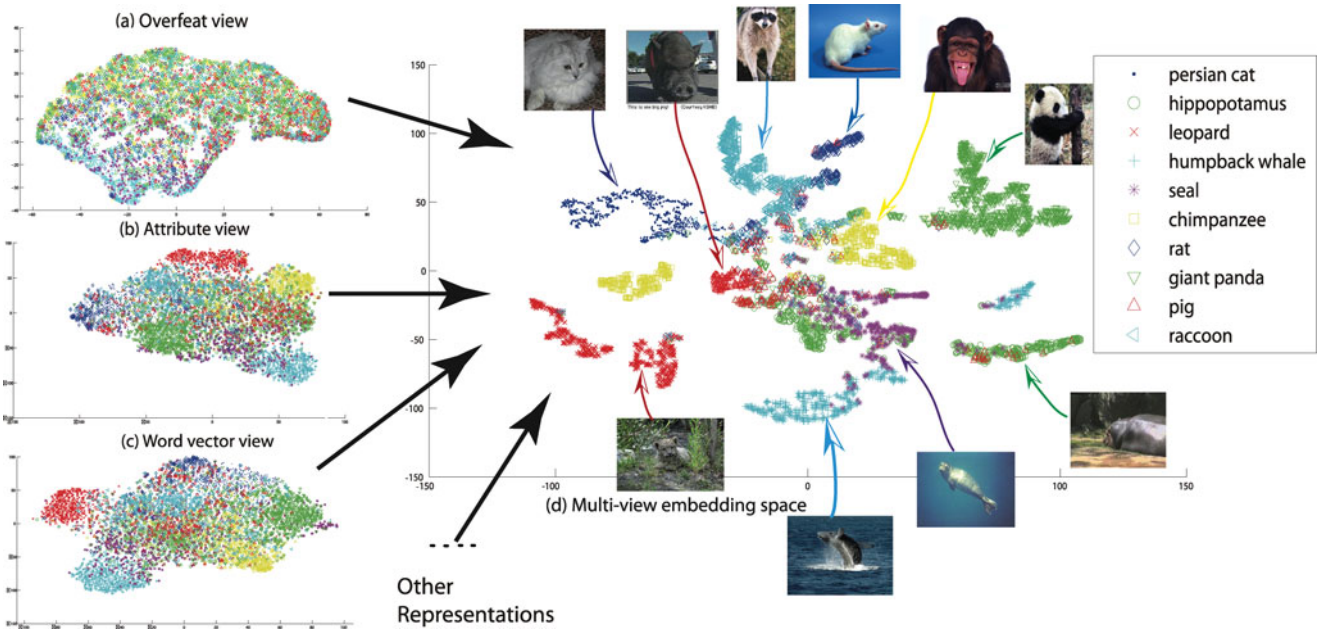


Fig. 6. t-SNE Visualisation of (a) OverFeat view ($\mathcal{X}_{\mathcal{O}}$), (b) attribute view ($\mathcal{A}_{\mathcal{O}}$), (c) word vector view ($\mathcal{V}_{\mathcal{O}}$), and (d) transition probability of pairwise nodes computed by Eq. (9) of TMV-HLP in ($\Gamma(\mathcal{X}+\mathcal{A}+\mathcal{V})_{\mathcal{O},\mathcal{D}}$). The unlabelled target classes are much more separable in (d).

than 'hand-crafted' features on many computer vision benchmark datasets [5], [42]. What is interesting to see here is that OverFeat > DeCAF since both are based on the same convolutional neural network (CNN) model of [25]. Apart from implementation details, one significant difference is that DeCAF is pre-trained by ILSVRC2012 while OverFeat by ILSVRC2013 which contains more animal classes meaning better (more relevant) features can be learned. It is also worth pointing out that: (1) With both OverFeat and DeCAF features, the number of views to learn an embedding space increases from 3 to 9; and our results suggest that the more views, the better chance to solve the domain shift problem and the data become more separable as different views contain complementary information. (2) Fig. 5b shows that when all nine available views ($\mathcal{X}_{\mathcal{H}}$, $\mathcal{V}_{\mathcal{H}}$, $\mathcal{A}_{\mathcal{H}}$, $\mathcal{X}_{\mathcal{D}}$, $\mathcal{V}_{\mathcal{D}}$, $\mathcal{A}_{\mathcal{D}}$, $\mathcal{X}_{\mathcal{O}}$, $\mathcal{V}_{\mathcal{O}}$ and $\mathcal{A}_{\mathcal{O}}$) are used for embedding, the result is significantly better than those from each individual view. Nevertheless, it is lower than that obtained by embedding views ($\mathcal{X}_{\mathcal{D}}$, $\mathcal{V}_{\mathcal{D}}$, $\mathcal{A}_{\mathcal{D}}$, $\mathcal{X}_{\mathcal{O}}$, $\mathcal{V}_{\mathcal{O}}$ and $\mathcal{A}_{\mathcal{O}}$). This suggests that view selection may be required when a large number of views are available for learning the embedding space.

Embedding makes target classes more separable. We employ t-SNE [47] to visualise the space $\mathcal{X}_{\mathcal{O}}$, $\mathcal{V}_{\mathcal{O}}$, $\mathcal{A}_{\mathcal{O}}$ and $\Gamma(\mathcal{X}+\mathcal{A}+\mathcal{V})_{\mathcal{O},\mathcal{D}}$ in Fig. 6. It shows that even in the powerful OverFeat view, the 10 target classes are heavily overlapped (Fig. 6a). It gets better in the semantic views (Figs. 6b and 6c). However, when all 6 views are embedded, all classes are clearly separable (Fig. 6d).

Running time. In practice, for the AwA dataset with hand-crafted features, our pipeline takes less than 30 minutes to complete the zero-shot classification task (over 6,180 images) using a six core 2.66 GHz CPU platform. This includes the time for multi-view CCA embedding and label propagation using our heterogeneous hypergraphs.

6.3 Annotation and Beyond

In this section we evaluate our multi-view embedding space for the conventional and novel annotation tasks introduced in Section 5.

Instance annotation by attributes. To quantify the annotation performance, we predict attributes/annotations for each target class instance for USAA, which has the largest instance level attribute variations among the three datasets. We employ two standard measures: mean average precision (mAP) and (FM) between the estimated and true annotation list. Using our multi-view embedding space, our method (FM: 0.341, mAP: 0.355) outperforms significantly the baseline of directly estimating $\mathbf{y}_u^{\mathcal{A}} = f^{\mathcal{A}}(\mathbf{x}_u)$ (FM: 0.299, mAP: 0.267).

Zero-shot description. In this task, we explicitly infer the attributes corresponding to a specified novel class, given only the textual name of that class without seeing any visual samples. Table 2 illustrates this for AwA. Clearly most of the top/bottom five attributes predicted for each of the 10 target classes are meaningful (in the ideal case, all top five should be true positives and all bottom five true negatives). Predicting the top-5 attributes for each class gives an F-measure of 0.236. In comparison, if we directly select the five nearest attribute name projection to the class name projection (prototype) in the word space, the F-measure is 0.063, demonstrating the importance of learning the multi-view embedding space. In addition to providing a method to automatically—rather than manually—generate an attribute ontology, this task is interesting because even a human could find it very challenging (effectively a human has to list the attributes of a class which he has never seen or been explicitly taught about, but has only seen mentioned in text).

Zero prototype learning. In this task we attempt the reverse of the previous experiment: inferring a class name given a list of attributes. Table 3 illustrates this for

Authorized licensed use limited to: Nanyang Technological University. Downloaded on August 31,2020 at 23:43:09 UTC from IEEE Xplore. Restrictions apply.

TABLE 2
Zero-Shot Description of 10 AwA Target Classes

AwA		Attributes
pc	T-5	active, **furry, tail, paws, ground.**
	B-5	swims, hooves, long neck, horns, arctic
hp	T-5	**old world, strong, quadrupedal**, fast, **walks**
	B-5	red, plankton, skimmers, stripes, tunnels
lp	T-5	**old world, active, fast, quadrupedal, muscle**
	B-5	plankton, arctic, insects, hops, tunnels
hw	T-5	**fish**, **smart, fast, group, flippers**
	B-5	hops, grazer, tunnels, fields, plains
seal	T-5	**old world, smart**, **fast**, chew teeth, **strong**
	B-5	fly, insects, tree, hops, tunnels
cp	T-5	**fast, smart, chew teeth, active**, brown
	B-5	tunnels, hops, skimmers, fields, long neck
rat	T-5	**active, fast, furry, new world, paws**
	B-5	arctic, plankton, hooves, horns, long neck
gp	T-5	**quadrupedal**, active, **old world**, **walks**, **furry**
	B-5	tunnels, skimmers, long neck, blue, hops
pig	T-5	**quadrupedal**, **old world**, **ground**, furry, **chew teeth**
	B-5	desert, long neck, orange, blue, skimmers
rc	T-5	**fast**, **active**, **furry**, **quadrupedal**, **forest**
	B-5	long neck, desert, tusks, skimmers, blue

Γ is learned using six views ($\mathcal{X}_{\mathcal{D}}$, $\mathcal{V}_{\mathcal{D}}$, $\mathcal{A}_{\mathcal{D}}$, $\mathcal{X}_{\mathcal{O}}$, $\mathcal{V}_{\mathcal{O}}$ and $\mathcal{A}_{\mathcal{O}}$). The true positives are highlighted in bold. pc, hp, lp, hw, cp, gp, and rc are short for Persian cat, hippopotamus, leopard, humpback whale, chimpanzee, giant panda, and raccoon respectively. T-5/B-5 are the top/bottom five attributes predicted for each target class.

USAA. Table 3a shows queries by the groundtruth attribute definitions of some USAA classes and the top-4 ranked list of classes returned. The estimated class names of each attribute vector are reasonable—the top-4 words are either the class name or related to the class name. A baseline is to use the textual names of the attributes projected in the word space (summing their word vectors) to search for the nearest classes in word space, instead of the embedding space. Table 3a shows that the predicted classes in this case are reasonable, but significantly worse than querying via the embedding space. To quantify this we evaluate the average rank of the true name for each USAA class when queried by its attributes. For querying by embedding space, the average rank is an impressive 2.13 (out of 4.33M words with a chance-level rank of 2.17 M), compared with the average rank of 110.24 by directly querying word space [32] with textual descriptions of the attributes. Table 3b shows an example of "incremental" query using the ontology definition of birthday party [15]. We first query the 'wrapped presents' attribute only, followed by adding 'small balloon' and all other attributes ('birthday songs and 'birthday caps'). The changing list of top ranked retrieved words intuitively reflects the expectation of the combinatorial meaning of the attributes.

7 Conclusions

We identified the challenge of projection domain shift in zero-shot learning and presented a new framework to solve it by rectifying the biased projections in a multi-view embedding space. We also proposed a novel label-propagation algorithm TMV-HLP based on heterogeneous across-view hypergraphs. TMV-HLP synergistically exploits multiple intermediate semantic representations, as well as the manifold structure of unlabelled target data to improve recognition in a unified way for zero shot, N-shot and zero+N shot learning tasks. As a result we achieved state-of-the-art performance on the challenging AwA, CUB and USAA datasets. Finally, we demonstrated that our framework enables novel tasks of relating textual class names and their semantic attributes.

A number of directions have been identified for future work. First, we employ CCA for learning the embedding space. Although it works well, other embedding frameworks can be considered (e.g. [49]). In the current pipeline, low-level features are first projected onto different semantic views before embedding. It should be possible to develop a unified embedding framework to combine these two steps. Second, under a realistic lifelong learning setting [6], an unlabelled data point could either belong to a seen/auxiliary category or an unseen class. An ideal framework should be able to classify both seen and unseen classes [44]. Finally, our results suggest that more views, either manually defined (attributes), extracted from a linguistic corpus (word space), or learned from visual data (deep features), can potentially

TABLE 3
Zero Prototype Learning on USAA

(a) Query by GT attributes of	Query via embedding space	Query attribute words in word space
graduation party	**party**, **graduation**, audience, caucus	cheering, proudly, dressed, wearing
music_performance	**music**, **performance**, musical, heavy metal	sing, singer, sang, dancing
wedding_ceremony	**wedding_ceremony**, wedding, glosses, stag	nun, christening, bridegroom, **wedding_ceremony**

(b) Attribute query	Top ranked words
wrapped presents	music; performance; solo_performances; performing
+small balloon	wedding; wedding_reception; birthday_celebration; birthday
+birthday song +birthday caps	**birthday_party**; prom; wedding reception

(a) Querying classes by groundtruth (GT) attribute definitions of the specified classes. (b) An incrementally constructed attribute query for the birthday_party class. Bold indicates true positive.

Authorized licensed use limited to: Nanyang Technological University. Downloaded on August 31,2020 at 23:43:09 UTC from IEEE Xplore. Restrictions apply.

give rise to better embedding space. More investigation is needed on how to systematically design and select semantic views for embedding.

Acknowledgments

The work was done while Yanwei Fu was at Queen Mary University of London. Yanwei Fu is the corresponding author.

References

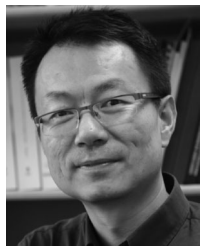
[1] Z. Akata, F. Perronnin, Z. Harchaoui, and C. Schmid, "Label-embedding for attribute-based classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2013, pp. 819–826.
[2] Y. Bengio, "Learning deep architectures for AI," *Found. Trends Mach. Learn.*, vol. 2, pp. 1–127, 2009.
[3] I. Biederman, "Recognition by components—A theory of human image understanding," *Psychological Rev.*, vol. 94, pp. 115–147, 1987.
[4] P. F. Brown, V. J. Pietra, P. V. deSouza, J. C. Lai, and R. L. Mercer, "Class-based n-gram models of natural language," *J. Comput. Linguistics*, vol. 18, pp. 467–479, 1992.
[5] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, "Return of the devil in the details: Delving deep into convolutional nets," *ArXiv e-prints*, 2014.
[6] X. Chen, A. Shrivastava, and A. Gupta, "NEIL: Extracting visual knowledge from web data," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2013, pp. 1409–1416.
[7] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell, "DeCAF: A deep convolutional activation feature for generic visual recognition," *CoRR*, abs/1310.1531, 2013.
[8] L. Duan, I. W. Tsang, D. Xu, and S. J. Maybank, "Domain transfer SVM for video concept detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2009, pp. 1375–1381.
[9] A. Farhadi, I. Endres, D. Hoiem, and D. Forsyth, "Describing objects by their attributes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2009, pp. 1778–1785.
[10] B. Fernando, A. Habrard, M. Sebban, and T. Tuytelaars, "Unsupervised visual domain adaptation using subspace alignment," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2013, pp. 2960–2967.
[11] A. Frome, G. S. Corrado, J. Shlens, S. Bengio, J. Dean, M. Ranzato, and T. Mikolov, "DeViSE: A deep visual-semantic embedding model," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 2121–2129.
[12] Y. Fu, Y. Guo, Y. Zhu, F. Liu, C. Song, and Z.-H. Zhou, "Multi-view video summarization," *IEEE Trans. Multimedia*, vol. 12, no. 7, pp. 717–729, Nov. 2010.
[13] Y. Fu, T. Hospedales, T. Xiang, and S. Gong, "Attribute learning for understanding unstructured social activity," in *Proc. 12th Eur. Conf. Comput. Vis.*, 2012, pp. 530–543.
[14] Y. Fu, T. M. Hospedales, T. Xiang, Z. Fu, and S. Gong, "Transductive multi-view embedding for zero-shot recognition and annotation," Computer Vision—ECCV 2014 Lecture Notes in Computer Science, vol. 8690, pp. 584–599, 2014.
[15] Y. Fu, T. M. Hospedales, T. Xiang, and S. Gong, "Learning multimodal latent attributes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 2, pp. 303–316, Feb. 2014.
[16] Y. Fujiwara and G. Irie, "Efficient label propagation," in *Proc. 31st Int. Conf. Mach. Learning*, 2014, pp. 784–792.
[17] Y. Gong, Q. Ke, M. Isard, and S. Lazebnik, "A multi-view embedding space for modeling internet images, tags, and their semantics," *Int. J. Comput. Vis.*, vol. 106, pp. 210–233, 2014.
[18] D. R. Hardoon, S. Szedmak, and J. Shawe-Taylor, "Canonical correlation analysis: An overview with application to learning methods," *Neural Comput.*, vol. 16, pp. 2639–2664, 2004.
[19] C. Hong, J. Yu, J. Li, and X. Chen, "Multi-view hypergraph learning by patch alignment framework," *Neurocomputing*, vol. 118, pp. 79–86, 2013.
[20] T. Hospedales, S. Gong, and T. Xiang, "Learning tags from unsegmented videos of multiple human actions," in *Proc. IEEE 11th Int. Conf. Data Mining*, 2011, pp. 251–259.
[21] Y. Huang, Q. Liu, and D. Metaxas, "Video object segmentation by hypergraph cut," in *Proc. IEEE Conf. Comput. Vis. Patern Recog.*, 2009, pp. 1738–1745.
[22] Y. Huang, Q. Liu, S. Zhang, and D. N. Metaxas, "Image retrieval via probabilistic hypergraph ranking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2010, pp. 3376–3383.
[23] S. J. Hwang and K. Grauman, "Learning the relative importance of objects from tagged images for retrieval and cross-modal search," *Int. J. Comput. Vis.*, vol. 100, pp. 134–153, 2011.
[24] S. J. Hwang, F. Sha, and K. Grauman, "Sharing features between objects and their attributes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2011, pp. 1761–1768.
[25] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.
[26] C. H. Lampert, "Kernel methods in computer vision," *Found. Trends Comput. Graph. Vis.*, vol. 4, pp. 193–285, 2009.
[27] C. H. Lampert, H. Nickisch, and S. Harmeling, "Learning to detect unseen object classes by between-class attribute transfer," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2009, pp. 951–958.
[28] C. H. Lampert, H. Nickisch, and S. Harmeling, "Attribute-based classification for zero-shot visual object categorization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 3, pp. 453–465, Mar. 2014.
[29] X. Li, W. Hu, C. Shen, A. Dick, and Z. Zhang, "Context-aware hypergraph construction for robust spectral clustering," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 10, pp. 2588–2597, Oct. 2014.
[30] X. Li, Y. Li, C. Shen, A. R. Dick, and A. van den Hengel, "Contextual hypergraph modelling for salient object detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2013, pp. 3328–3335.
[31] J. Liu, B. Kuipers, and S. Savarese, "Recognizing human actions by attributes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2011, pp. 3337–3344.
[32] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representation in vector space," in *Proc. Workshop ICLR*, 2013.
[33] M. Palatucci, G. Hinton, D. Pomerleau, and T. M. Mitchell, "Zero-shot learning with semantic output codes," in *Proc. Adv. Neural Inf. Process. Syst.*, 2009, pp. 1410–1418.
[34] D. Parikh and K. Grauman, "Relative attributes," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2011, pp. 503–510.
[35] A. Pentina and C. H. Lampert, "A PAC-Bayesian bound for lifelong learning anastasia," in *Proc. Int. Conf. Mach. Learning*, 2014, pp. 991–999.
[36] M. Rohrbach, S. Ebert, and B. Schiele, "Transfer learning in a transductive setting," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 46–54.
[37] M. Rohrbach, M. Stark, and B. Schiele, "Evaluating knowledge transfer and zero-shot learning in a large-scale setting," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2012, pp. 1641–1648.
[38] M. Rohrbach, M. Stark, G. Szarvas, I. Gurevych, and B. Schiele, "What helps where—and why semantic relatedness for knowledge transfer," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2010, pp. 910–917.
[39] E. Rosch, "Classification of real-world objects: Origins and representations in cognition," in *Thinking: Readings in Cognitive Science*. Cambridge, U.K.: Cambridge Univ. Press, 1977.
[40] R. Rosipal and N. Kramer, "Overview and recent advances in partial least squares," in *Subspace, Latent Structure and Feature Selection*, C. Saunders, M. Grobelnik, S. Gunn, and J. Shawe-Taylor, Eds. Berlin, Germany: Springer, 2006, pp. 34–51.
[41] W. J. Scheirer, N. Kumar, P. N. Belhumeur, and T. E. Boult, "Multi-attribute spaces: Calibration for attribute fusion and similarity search," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2012, pp. 2933–2940.
[42] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, "Overfeat: Integrated recognition, localization and detection using convolutional networks," in *Proc. Int. Conf. Learning Representations*, 2014.
[43] R. Socher and L. Fei-Fei, "Connecting modalities: Semi-supervised segmentation and annotation of images using unaligned text corpora," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2010, pp. 966–973.
[44] R. Socher, M. Ganjoo, H. Sridhar, O. Bastani, C. D. Manning, and A. Y. Ng, "Zero-shot learning through cross-modal transfer," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 935–943.
[45] A. J. Storkey and M. Sugiyama, "Mixture regression for covariate shift," in *Proc. Adv. Neural Inf. Process. Syst.*, 2007, pp. 1337–1344.
[46] L. Sun, S. Ji, and J. Ye, "Hypergraph spectral learning for multi-label classification," in *Proc. 14th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2008, pp. 668–676.

Authorized licensed use limited to: Nanyang Technological University. Downloaded on August 31,2020 at 23:43:09 UTC from IEEE Xplore. Restrictions apply.

[47] L. van der Maaten and G. Hinton, "Visualizing high-dimensional data using t-SNE," *J. Mach. Learning Res.*, vol. 9, pp. 2579–2605, 2008.
[48] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie, "The Caltech-UCSD Birds-200-2011 Dataset," California Inst. Technol., Pasadena, CA, USA, Tech. Rep. CNS-TR-2011-001, 2011.
[49] K. Wang, R. He, W. Wang, L. Wang, and T. Tan, "Learning coupled feature spaces for cross-modal matching," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2013, pp. 2088–2095.
[50] X. Wang and Q. Ji, "A unified probabilistic approach modeling relationships between attributes and objects," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2013, pp. 2120–2127.
[51] Y. Wang and S. Gong, "Translating topics to words for image annotation," in *Proc. 16th ACM Conf. Inf. Knowl. Manage.*, 2007, pp. 995–998.
[52] D. Watts and S. Strogatz, "Collective dynamics of 'small-world' networks," *Nature*, vol. 393, pp. 440–442, 1998.
[53] D. J. Watts, *Small Worlds: The Dynamics of Networks between Order and Randomness*, 8th ed. Oakland, CA, USA: Univ. Press California, 2004.
[54] F. X. Yu, L. Cao, R. S. Feris, J. R. Smith, and S.-F. Chang, "Designing category-level attributes for discriminative visual recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2013, pp. 771–778.
[55] D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Schölkopf, "Learning with local and global consistency," in *Proc. Adv. Inf. Process. Syst.*, 2004, pp. 321–328.
[56] D. Zhou and C. J. C. Burges, "Spectral clustering and transductive learning with multiple views," in *Proc. Int. Conf. Mach. Learning*, 2007, pp. 1159–1166.
[57] X. Zhu, "Semi-supervised learning literature survey," Dept. Comput. Sci., Univ. Wisconsin-Madison, Madison, WI, USA, Tech. Rep. 1530, 2007.

Yanwei Fu received the PhD degree from Queen Mary University of London in 2014, and the MEng degree from the Department of Computer Science & Technology, Nanjing University in 2011, China. He is a postdoctoral researcher with Leonid Sigal in Disney Research, Pittsburgh, which is co-located with Carnegie Mellon University. His research interests include attribute learning for image and video understanding, robust learning to rank, and large-scale video summarization.

Timothy M. Hospedales received the PhD degree in neuroinformatics from the University of Edinburgh in 2008. He is currently a lecturer (assistant professor) of computer science at Queen Mary University of London. His research interests include probabilistic modeling and machine learning applied variously to problems in computer vision, data mining, interactive learning, and neuroscience. He has published more than 20 papers in major international journals and conferences. He is a member of the IEEE.



Tao Xiang received the PhD degree in electrical and computer engineering from the National University of Singapore in 2002. He is currently a reader (associate professor) in the School of Electronic Engineering and Computer Science, Queen Mary University of London. His research interests include computer vision and machine learning. He has published more than 100 papers in international journals and conferences and co-authored a book, *Visual Analysis of Behaviour: From Pixels to Semantics*.

Shaogang Gong received the DPhil in computer vision from Keble College, Oxford University in 1989. He is a professor of visual computation at Queen Mary University of London, a fellow of the Institution of Electrical Engineers and a fellow of the British Computer Society. His research interests include computer vision, machine learning, and video analysis.

▷ **For more information on this or any other computing topic, please visit our Digital Library at** www.computer.org/publications/dlib.

Authorized licensed use limited to: Nanyang Technological University. Downloaded on August 31,2020 at 23:43:09 UTC from IEEE Xplore. Restrictions apply.