EXPLORING SYNONYMS AS CONTEXT IN ZERO-SHOT ACTION RECOGNITION

Ioannis Alexiou[⋆] *Tao Xiang*[†] *Shaogang Gong*[⋆]

[⋆]Queen Mary University of London
[†]Vision Semantics Ltd

ABSTRACT

Zero shot learning (ZSL) provides a solution to recognising unseen classes without class labelled data for model learning. Most ZSL methods aim to learn a mapping from a visual feature space to a semantic embedding space, e.g. attribute or word vector spaces. The use of word vector space is particularly attractive as compared to attribute, it offers vast auxiliary classes with free parts embedding without human annotation. However, using the word vector embedding often provides weaker discriminative power than manually labelled attributes of the auxiliary classes. This is compounded further in zero-shot action recognition due to richer content variations among action classes. In this work we propose to explore a broader semantic contextual information in the text domain to enrich the word vector representation of action classes. We show through extensive experiments that this method improves significantly the performance of a number of existing word vector embedding ZSL methods. Moreover, it also outperforms attribute embedding ZSL with human annotation.

Index Terms— zero-shot learning, action recognition, semantic embedding

1. INTRODUCTION

Conventional approaches to visual recognition are based on supervised learning. That is, given a large labelled training dataset of a known set of classes (e.g., hundreds of instances per class), a classifier is learned to classify each instance in a test dataset into the same set of classes. Collecting large quantities of annotated instances for each class is a bottleneck, especially when visual recognition tasks are moving towards a finer granularity on details. To overcome this bottleneck, Zero-Shot Learning (ZSL) aims to recognise a new class without seeing visual samples.

The underlying principle of ZSL is that each unseen class name can be embedded into a semantic space into which low-level feature representation of unseen visual data can be projected and their similarity to the unseen class names (known) can be estimated [1, 2, 3, 4, 5, 6]. The semantic embedding spaces considered by most early works are attribute spaces [1, 2]. However, to represent an object class in an attribute space, an attribute ontology has to be defined manually and each class needs be annotated by an attribute vector. Moreover, the labelling of different datasets of the same attributes is often inconsistent therefore non-scalable. Such an approach hinders the scalability of an attribute space based ZSL method. To overcome this, more recent works [5, 6] explore the semantic (text) word vector space [7], which is learned using large corpus of unannotated text for natural language processing tasks such as sentence completion. The text corpus is vast that any class label or textual description of the class can be embedded in this space and they are universal (dataset independent), effectively mitigating the scalability issue. In this paper, we focus on the word vector space embedding methods.

Most existing works [8, 6] show that attribute space embedding is more informative than the word vector space embedding. This is due to: (a) Most attributes used by existing work are visual whilst the dimensions of a word vector space have no corresponding visual meanings. (b) More importantly, using a word vector representation of single words (class name) to represent the rich appearance variation of a whole class is over-simplified. This problem is particularly acute for action recognition with each action class consisting of discriminative constituent parts. If an image is worth 1,000 words, a video is perhaps worth tens of thousands.

In this paper, we propose to enrich the word vector representation of unseen action class names by exploring broader semantic content. Specifically we automatically mine a set of *synonyms* for each class name to supplement and extend its semantic dimension. In doing so, instead of using a single word vector, a set of word vectors are used to represent an unseen class in a word vector embedding space. Our method can be applied to any existing word vector embedding ZSL models. In addition, to address the domain shift problem [9] suffered by all existing ZSL methods, the proposed synonym based semantic word vector representation of unseen classes can be further extended by exploiting the embedded unseen class visual data in a self-training manner.

Extensive experiments are performed on two large action recognition datasets (UCF-101 and HMDB-51). The results demonstrate that when the proposed method is applied in con-

This research was funded by the European Unions Seventh Framework Programme managed by REA-Research Executive Agency http://ec.europa.eu/research/rea under Grant No 606952 - SmartPrevent.

junction with a number of existing ZSL methods, their performance is improved significantly. In particular, we show that with our method, a ZSL model with word vector embedding can outperform the same ZSL model with attribute space embedding. This is important as it addresses the inherent limitation of the existing word vector space ZSL in the lack of informative representation.

2. METHODOLOGY

The proposed zero-shot learning approach is illustrated in Fig. 1. Assume there is a training set containing a set of labelled training data from seen classes and a test set containing unlabelled test class data. The seen class-samples are defined by $S = \{X_s, Y_s, Z_s\}$ accordingly. S is large collection of visual representations $X_s = [x_1, \ldots, x_N]$ with $X_s \in \mathcal{R}^{N \times d}$. N is the total number of samples and d their dimensionality (e.g., x_i has d elements). Y_s represents the semantic embedding space which can be either attribute space or word vector space with $Y_s = [y_1, \ldots, y_N]$ where $Y_s \in \mathcal{R}^{N \times l}$. In this work we focus on the word vector space. The dimensionality l of each intermediate space vector y_i depends on the employed ZSL method. The class identification vector $Z_s = [z_1, \ldots, z_N]$ with $Z_s \in \mathcal{N}^{N \times 1}$ contains integer values that identify the classes. This info is also useful in ZSL approach to test the efficiency of the proposed model. Similarly, $U = \{X_u, Y_u, Z_u\}$ unseen samples can be defined with the absolute condition that the class labels from S and U are disjoint ($S \notin U$) only if $Z_s \cap Z_u = \emptyset$.

2.1. Visual Representations

In this work, we focus on action recognition under a zero-shot setting. Two recent action representations can be considered. The first is the unsupervised approach of Fisher Vectors [10, 11] and the other is based on deep convolutional neural networks (CNNs) in which their output layers can be used as features [12, 6, 13, 14, 15, 16]. The CNNs can learn powerful representations given the fact that vast amounts of data exists. A drawback of CNNs is that cannot produce high quality features on unseen classes unless they have been trained on large datasets. This is not the objective of ZSL where no visual knowledge of the unknown classes is mandatory. We opt for the Fisher Vector (FV) representation due to low computational cost and the unsupervised nature of this method. FVs do not require labelled data during training and testing.

FVs computation involves GMM model which is trained on local features. As local features, we adopt the MBH [17] features. A random subset of MBH descriptors x are fed to a GMM to learn the following model parameters $\lambda = \{w_k, \mu_k, \Sigma_k, k = 1, \ldots, K\}$. A collection of Gaussian models is learnt with w_k weight, μ_k the centre of the Gaussian (mean) and its covariance matrix Σ_k (its extent in the feature space). The Fisher vectors of MBH descriptors

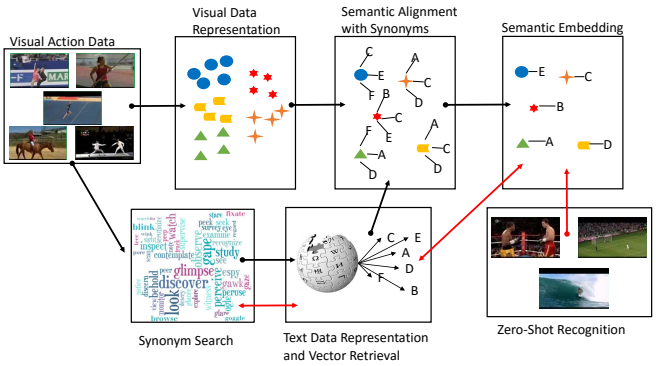

Fig. 1: A schematic representation of the proposed zero-shot learning model. The black arrows show the data flow during the training while the red arrows show the testing phase of the model.

(λ) are then computed from X_s and X_u to construct the independent training and testing data sets.

2.2. Word Vector Space

The word vector space proposed in [7] is used which is termed as word2vec. A skip-gram neural network is trained on a large corpus to derive vector representations of billions of words. Specifically the word2vec net is trained on the latest release of Wikipedia data which contains roughly 3.8 billion different words (with the skip-gram option). The word2vec net will produce a list of the words $W = [w_1, \ldots, w_L]$ and the corresponding set of vectors $Y = [y_1, \ldots, y_N]$ which form the intermediate word semantic space. y_1 is a d dimensional vector as defined by the user.

A mapping needs to be learnt from the visual domain to the word-vector domain. An SVM regressor is trained for that purpose with a Hellinger kernel choice as in [11, 10]. We consider enough correlation among the word space vectors as many words have overlapping meanings. Bearing that, two mappings are obtained with the first being the raw word vectors and the second obtained after the vectors have been decorrelated. The vectors were decorrelated using the PCA algorithm to maintain the most variant dimensions.

2.3. Synonym Modelling and Self Training

In the conventional word vector embedding ZSL method, a single word vector is used to represent a class. More recently, the idea of summing phrases corresponding to class labels by a simple averaging of multiple words was introduced [18] as:

$$Y'_s = \frac{1}{M} \sum_{m=1}^{M} f(Z_s, W, Y_s); \quad (1)$$

However, this simple approach by averaging can blur and distort the word-vectors which may misalign the semantic representation of the word-vectors with the class labels resulting

poor performance. Here we wish to explore a broader semantic context of the target action class name. Specifically, we consider to search a set of words that have similar semantic meaning as the class name in a wider context. We introduce a new class word vector representation by taking into account *all relevant words* from multiple text domains. To that end, we formulate an algorithm 1 to address the inherent label (mis)alignment between multiple domains (domain shift). More precisely, many class label names do not have a good single noun description that can effectively represent the underlying action. This will negatively affect the retrieved word-vectors of the corresponding class labels by adding randomness to the visual-word space mappings. To address this problem, we mine synonyms of the action class labels from multiple internet dictionaries including *Google*, *TheFreeDictionary*, *OxfordDictionary* and *WordReferrence*. All these synonyms $R = \{r_1, \ldots, r_N\}$ are stacked for each class label $Z = [z_1, \ldots, z_N]$. Once all the synonyms R have been gathered for each action class label, the Algorithm 1 is performed to identify the most suitable synonym (word-vector) for each action class label respectively. On rare occasions when synonyms of an action class cannot be found, the simple averaging model (Eq. (1) is adopted.

Algorithm 1: Improved word vector representation of class names.

Data: Class Labels Z, Word Synonyms Dictionary R, Word-Vectors Y
Result: Improved Word Space Alignment Y'
while *converge to max performace* **do**
  for z_i *in* Z **do**
    $z_i' \leftarrow R(z_i)$;
    // Synonym as ClassLabel;
    $Location \leftarrow SeekIn(Y, z_i)$;
    // Find Vector in Pool;
    if $Location \neq empty$ **then**
      $Y'_{zi} \leftarrow Y(Location)$;
      // Found vector becomes class label
    else
      $Y'_{zi} \leftarrow$ Eq. (1) //update ;
    end
    $Y_u = SemanticEmbedding(X)$;
    $Y' \leftarrow$ Eq.(2) //update ;
    $Z_u = NN(Y', Y_u)$;
    //Perform nearest neighbour to assign class labels
  end
end

Having enriched the class name representation in the word vector space, there is still one problem remaining. In particular, the visual-word space domain shift occurs when applying the learnt embedding to map from the new visual representations to the word space (see Fig. 1). Specifically, the estimated word vectors are distorted by the applied embedding, thus transductive self-training is applied to reduce that effect. This is implemented by a k nearest neighbour smoothing (k-NN) on the estimated vectors to compensate this distortion. Our approach differs from [18] because in our approach weights w_k are applied onto the k-NN smoothing as follows.

$$y_i = \frac{1}{K} \sum_{k=1}^{K} w_k \underset{k}{NN}(Y_u) \tag{2}$$

The weights are estimated by Euclidean distance among the k nearest neighbours. The estimated distance is inverted and assigned for each corresponding weight w_k, where all weights are $L1$ normalised. The final class estimation Z_u is achieved by re-estimating the NN between the mapped visual representations to the word-space and the actual word vectors using cosine distance.

3. EXPERIMENTS

Experiments are performed on the UCF-101 [19] and HMDB-51 [20] dataset containing 101 and 51 action classes respectively. **Data Splits:** The zero-shot experiments are performed in the following fashion. First, each dataset is split to half for training and half for zero-shot testing. This is repeated 10 times randomly. Specifically 50% of the action classes are kept seen during the training process and the other 50% is kept unseen for zero-shot testing. **Visual Representation:** For the visual representations we use only unsupervised features as zero-shot classification has no label information. Each video is converted into motion features. These MBH features are acquired by the improved dense trajectory implementation by [17]. Only the MBH descriptors per video file are used to compute a holistic representation of each video. To achieve this the MBH features of each video are converted to fisher vectors using a 256-components GMM. Prior to fisher vector encoding, the dimensionality of each MBH descriptor is reduced to 96. **Word-Space Model:** The word space is obtained by using the skip-gram (a 2 layer neural net) [7]. The word2vec model is trained on the latest wikipedia data release. The text data are preprocessed to remove xml-based characters, spell out digits and convert all letters to lowercase. The skip-gram model is set to generate 300 dimensional vectors. Next, comparative evaluations are given to show the performance advantage of the proposed synonym modelling with self-training in enhancing existing word-vector embedding models for ZSL action recognition.

3.1. Comparative Evaluations

Tables 1 and 2 compared four existing word vector embedding models (IAP-WV, DAP-WV, TM-CLSI, SESA) against the proposed synonym mining and alignment enhancement on these models. Note, IAP [2] was proposed originally for attribute based ZSL. In this experiment the attributes are

replaced with word vectors (-"WV") for fair comparison. These models are trained on the wikipedia pages that contain descriptions and definitions for almost any word. The TM-CLSI [21] method extracts textual features in two phases, a widely used approach in document retrieval. The first phase is an indexing phase that generates textual features with TFIDF (Term Frequency-Inverse Document Frequency) configuration The second phase is a dimensionality reduction step, in which Clustered Latent Semantic Indexing (CLSI) algorithm is used. The SESA model [18] constructs an intermediate semantic space using word vectors. Its semantic embedding is achieved by applying support vector regressors. As evident in Table 1, the SESA model performs the best among the existing word vector embedding models.

Table 1: Conventional word vector embedding for ZSL action recognition.

	UCF-101	HMDB-51
IAP-WV	7.91	8.74
DAP-WV	5.35	6.15
TM-CLSI [21]	3.10	4.12
SESA [18]	**10.90**	**13.00**

Table 2: Synonym enhancement on word vector embedding for ZSL action recognition.

	UCF-101	HMDB-51
IAP-WV-Syn	9.73	10.12
DAP-WV-Syn	6.78	7.52
TM-CLSI [21]	3.10	4.12
SESA-Syn	**12.01**	**14.38**

Table 2 shows clearly that three of the four existing word vector embedding models benefit significantly from the proposed synonym mining and alignment (Algorithm 1) for exploring broader semantic context in zero-shot action recognition, as noted by "-Syn". In the case of TM-CLSI, it is not feasible to introduce the additional synonyms mining given its current implementation because the text pool of TM-CLSI uses one-to-one correspondence between class labels and text definitions (i.e. single-word definitions only). In this experiment, we reduced the dimensionality of the word vectors down to 70 for all the methods in order to accelerate the speed of computation. Comparing Tables 1 and 2, it is evident that the most gains are achieved for SESA and IAP.

Table 3 shows the benefit of further introducing self-training ("-ST"). The self-training idea was also exploited in the original SESA [18]. However, our formulation of self-training (Eq. 2) adopts a weighting scheme, which provides slightly improved performance. Three of the existing word vector embedding methods benefit significantly from this self-training. It is worth pointing out that the IAP model benefits greatly from self-training along with synonym enhancement. This suggests that IAP can be generalised effectively to word vector space beyond attribute without the need for human annotation.

Table 3: The benefit of self-training.

	UCF-101	HMDB-51
IAP-WV-Syn-ST	25.67	**28.45**
DAP-WV-Syn-ST	12.32	14.65
TM-CLSI [21]	3.10	4.12
SESA-Syn-ST	**28.53**	16.47

Table 4: Comparisons between synonym and self-training enhanced word vector embedding vs. attribute embedding on ZSL action recognition.

	UCF-101	HMDB-51
IAP-Attributes [2]	13.08	15.64
DAP-Attributes [2]	13.37	16.12
IAP-WV-Syn-ST	25.67	**28.45**
DAP-WV-Syn-ST	12.32	14.65
TM-CLSI [21]	3.1	4.12
SESA-ST [18]	15.8	15
SESA-ST-Aux [18]	18.6	21.2
SESA-Syn-ST	28.53	16.47
SESA-Syn-ST-Aux	**35.17**	22.41

Finally, we compared the synonym enhancement with self-training on word vector embedding models against the attribute embedding models for ZSL action recognition. Note, to perform this experiment, due to the lack of full attribute annotations, we annotated and expanded the attributes for the HMDB-51 classes using the original 115 attributes from [2]. Table 4 shows the performance of all methods. It is evident that the overall strategy of "-Syn-ST" boosts the performance of IAP, DAP and SESA by a notable margin. Moreover, it is also clear that these synonym and self-training enhanced word vector embedding models outperform significantly the attribute embedding models. Additionally, the auxiliary "-Aux" data from UCF-101 and HMDB-51 were introduced to the SESA regressors from the non-testing dataset. This use of the auxiliary data improves further the performance of the SVRs in SESA.

4. CONCLUSIONS

In this work we introduced a novel synonym enhancement based word vector space embedding approach to ZSL action recognition. We show that synonyms mining and alignment can benefit word vector embedding by introducing more robust semantic context from a wider range of text domains. The approach can be further aided by self-training. Together, word vector embedding is capable of beating the attribute embedding approach and without the need for human annotation.

5. REFERENCES

[1] Jingen Liu, Benjamin Kuipers, and Silvio Savarese, "Recognizing human actions by attributes," in *CVPR*, 2011.

[2] Christoph H Lampert, Hannes Nickisch, and Stefan Harmeling, "Attribute-based classification for zero-shot visual object categorization," *TPAMI*, 2014.

[3] Dinesh Jayaraman and Kristen Grauman, "Zero-shot recognition with unreliable attributes," in *NIPS*, 2014.

[4] Andrea Frome, Greg S Corrado, Jon Shlens, Samy Bengio, Jeff Dean, Marc Aurelio Ranzato, and Tomas Mikolov, "Devise: A deep visual-semantic embedding model," in *NIPS*. 2013.

[5] Richard Socher, Milind Ganjoo, Christopher D Manning, and Andrew Ng, "Zero-shot learning through cross-modal transfer," in *NIPS*. 2013.

[6] Zhenyong Fu, Tao Xiang, Elyor Kodirov, and Shaogang Gong, "Zero-shot object recognition by semantic manifold distance," in *CVPR*, 2015.

[7] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean, "Distributed representations of words and phrases and their compositionality," in *NIPS*. 2013.

[8] Zeynep Akata, Scott Reed, Daniel Walter, Honglak Lee, and Bernt Schiele, "Evaluation of output embeddings for fine-grained image classification," in *CVPR*, 2015.

[9] Yanwei Fu, Timothy M Hospedales, Tao Xiang, Zhenyong Fu, and Shaogang Gong, "Transductive multi-view embedding for zero-shot recognition and annotation," in *ECCV*. 2014.

[10] Jorge Snchez, Florent Perronnin, Thomas Mensink, and Jakob Verbeek, "Image classification with the fisher vector: Theory and practice," *IJCV*, 2013.

[11] Xiaoyang Wang and Qiang Ji, "A unified probabilistic approach modeling relationships between attributes and objects," in *ICCV*, 2013.

[12] Andrej Karpathy, George Toderici, Sanketh Shetty, Thomas Leung, Rahul Sukthankar, and Li Fei-Fei, "Large-scale video classification with convolutional neural networks," in *CVPR*, 2014.

[13] Karen Simonyan and Andrew Zisserman, "Two-stream convolutional networks for action recognition in videos," in *NIPS*. 2014.

[14] Junjie Cai, Richard Hong, Meng Wang, and Qi Tian, "Exploring feature space with semantic attributes," in *ICME*, 2015.

[15] Junjie Cai, Michele Merler, Sharath Pankanti, and Qi Tian, "Heterogeneous semantic level features fusion for action recognition," in *ICMR*, 2015.

[16] Zhongwen Xu, Yi Yang, and Alexander G Hauptmann, "A discriminative cnn video representation for event detection," in *CVPR*, 2015.

[17] Heng Wang and Cordelia Schmid, "Action recognition with improved trajectories," in *ICCV*, 2013.

[18] Xun Xu, Timothy Hospedales, and Shaogang Gong, "Semantic embedding space for zero-shot action recognition," in *ICIP*, 2015.

[19] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah, "Ucf101: A dataset of 101 human actions classes from videos in the wild," *CRCV-TR-12-01*, 2012.

[20] Hildegard Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso Poggio, and Thomas Serre, "Hmdb: a large video database for human motion recognition," in *ICCV*, 2011.

[21] Mohamed Elhoseiny, Babak Saleh, and Ahmed Elgammal, "Write a classifier: Zero-shot learning using purely textual descriptions," in *ICCV*, 2013.