論文 / 著書情報
Article / Book Information

Title	Adaptation of Word Vectors using Tree Structure for Visual Semantics
Author	Nakamasa Inoue, Koichi Shinoda
Citation	Proc. ACM Multimedia, , , pp. 277-281
Issue date	2016, 10
Copyright	Copyright (c) 2016 Association for Computing Machinery
Set statement	(c) ACM, 2016. This is the author's version of the work. It is posted here by permission of ACM for your personal use. Not for redistribution. The definitive version was published in Proceedings of the 2016 ACM on Multimedia Conference pp. 277-281, http://dx.doi.org/10.1145/2964284.2967226

Adaptation of Word Vectors using Tree Structure for Visual Semantics

Nakamasa Inoue
Tokyo Institute of Technology, Tokyo, Japan
inoue@ks.cs.titech.ac.jp

Koichi Shinoda
Tokyo Institute of Technology, Tokyo, Japan
shinoda@cs.titech.ac.jp

ABSTRACT

We propose a framework of word-vector adaptation, which makes vectors of visually similar concepts close to each other. Here, word vectors are real-valued vector representation of words, e.g., *word2vec* representation. Our basic idea is to assume that each concept has some hypernyms that are important to determine its visual features. For example, for a concept *Swallow* with hypernyms *Bird*, *Animal* and *Entity*, we believe *Bird* is the most important since birds have common visual features with their feathers etc. Adapted word vectors are obtained for each word by taking a weighted sum of a given original word vector and its hypernym word vectors. Our weight optimization makes vectors of visually similar concepts close to each other, by giving a large weight for such important hypernyms. We apply the adapted word vectors to zero-shot learning on the TRECVID 2014 semantic indexing dataset. We achieved 0.083 of Mean Average Precision, which is the best performance without using TRECVID training data to the best of our knowledge.

1. INTRODUCTION

With advances in information and communication technologies, a large amount of image, video and text data have been made available on the Internet. Semantic analysis of such data has received a growing amount of attention since it has many multimedia applications such as search, surveillance, summarization, and robot vision. However, it has been a challenging topic due to the semantic gap [1], the lack of correspondence between low-level features and high-level semantic concepts such as objects, actions, and scenes.

For semantic analysis, knowing semantic relation between concepts is important. For example, relation between objects and/or scenes has shown to play important roles to bridge the semantic gap in recent works on zero-shot object recognition [2, 3, 4], concept localization [5], and event recounting [6].

A recent trend to extract semantic relation is to utilize word vectors [7, 8, 9], which represent a word by a real-valued vector. Since word vectors have interesting property that semantic regularities are captured by vector operations, e.g., the result of $v(\text{King}) - v(\text{Man}) + v(\text{Woman})$ is close to $v(\text{Queen})$ where $v(w) \in \mathbb{R}^d$ is a word vector of a word w, they can directly help to know the semantic relation between concepts. However, since word vectors are typically trained on a text corpus such as Wikipedia, similar words (concepts) found by them are not always visually similar to each other. This inspires us to propose a framework of word-vector adaptation for visual semantics.

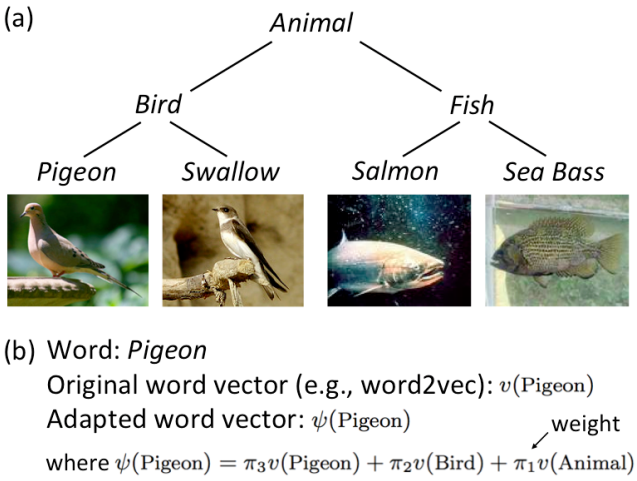

Figure 1: Adaptation of word vectors. (a) Example tree structure to obtain hypernyms of words. (b) An adapted word vector for *Pigeon*, which is obtained from weighted sum of original word vectors with its hypernyms. The weight coefficients are optimized on training data to make vectors of visually similar concepts close to each other.

To discuss visual similarity, let's have a question: *Why are Swallow and Pigeon visually similar*? If we answer to this question by *Because they are* ____*s*, which hypernym is appropriate for the blank: *Bird* or *Animal*? Because birds have common visual features with their feathers etc., we believe *Bird* is more appropriate. This implies that a hypernym *Bird* is an important word to explain the visual features of *Swallow* and *Pigeon*, and gives us an idea to use hyponyms for word-vector adaptation to image and video applications.

Our key contribution is a framework of word-vector adaptation based on hypernyms of words shown in Figure 1. From original word vectors given by a word embedding method such as Skip-gram model [7], our proposed framework ex-

DOI: http://dx.doi.org/10.1145/2964284.2967226

tracts adapted word vectors, which represent a word w by a weighted sum of its original word vector and its hypernym word vectors. Our weight optimization algorithm gives a large weight for important hyponyms, e.g., *Bird* for *Swallow*, to make vectors of visually similar concepts close to each other. In experiments, we apply adapted word vectors to zero-shot learning on the TRECVID Semantic Indexing dataset [10] to show the effectiveness of our framework. Here, zero-shot learning [2, 3, 4] is a type of learning which assumes that 1) target categories for training and testing are disjoint, and 2) traning images and/or videos are given only for training categories. We use 1,000 object categories from ImageNET [11] and 30 concept categories from TRECVID for training and testing, respectively.

The rest of this paper is organized as follows. Sec. 2 describes related works. Sec. 3 presents the proposed framework. Sec. 4 shows experimental evaluations, and Sec. 5 describes the conclusion with future work.

2. RELATED WORK

To extract word vectors, many word embedding methods have been proposed in the field of natural language processing with their applications such as machine translation [12] and web-document classification [13]. Latent semantic analysis [14] based on matrix factorization of a word frequency matrix is known to be a standard method. Some recent methods are focusing on the word analogy task, which aims to answer semantic questions about objects or places. The Skip-gram model proposed by Mikolov et al. [7] trains word vectors from a large-scale text corpus by using a neural network. The Global vector representation (GloVe) [9] introduces cooccurrence statistics to matrix factorization. With these methods, word analogies are represented by vector operations in the word-vector space. For example, the analogy of "Man is to Woman as King is to Queen" is represented by the fact that the result of $v(\text{King}) - v(\text{Man}) + v(\text{Woman})$ is close to $v(\text{Queen})$ where $v(w) \in \mathbb{R}$ is a word vector of a word w.

Since these word analogies include semantic relation between concepts such as objects and scenes, they can be directly introduced to visual applications. For example, zero-shot learning [15, 16] is known to be one of its major applications. Since zero-shot learning assumes sets of object categories for training and testing are disjoint, visual similarity between concepts often plays an important role. However, similar words (concepts) found by word vectors trained on a text corpus are not always visually similar to each other.

If we have labeled images and videos, a straightforward way to measure visual similarity is to use visual features. For feature extraction, a recent trend is to train visual features with a statistical model from large-scale datasets. For example, supervised learning of deep convolutional neural networks [17, 18] has been shown to be effective. The parameters of these networks are typically trained on ImageNET images [11], and are often fine-tuned on other data such as TRECVID videos [10]. For example, Snoek et al. have shown the effectiveness of deep neural networks in video semantic indexing and event detection [19].

Without using training images, it is also known that visual similarity between concepts can be obtained from specific metadata. For example, attribute metadata about colors and textures, which relates attributes with objects, has shown to be effective to detect objects such as animals [15, 16, 20]. Meta descriptions of events, which relate concepts to events, are often introduced to event detection [21, 22, 23]. The Skip-gram model [7] is also introduced in [24]. With these studies, visual similarity can be measured because detailed metadata often describes about visual information of objects, events, and scenes. However, manual attachment of such metadata is known to be costly.

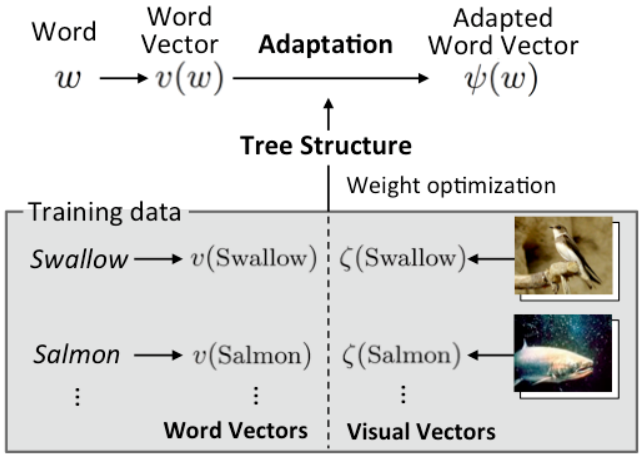

Figure 2: Overview of adaptation of word vectors. From training data, word vectors and visual vectors are extracted for each concept. They are used to optimize parameters for adaptation using tree structure.

In contrast, our framework of word-vector adaptation to visual applications aims to make word vectors of visually similar concepts close to each other without using metadata. Notably, our proposed framework is complementary to recent zero-shot learning techniques including domain adaptation [25, 26, 27, 28], distance modification [29], and joint learning [30, 31].

3. PROPOSED FRAMEWORK

This section presents our proposed framework of word-vector adaptation shown in Figure 2. Let $v(w) \in \mathbb{R}^d$ be a word vector of a word w given by a word embedding method. The proposed framework represents a word w by an adapted word vector $\psi(w)$, which is defined by a weighted sum of an original word vector and its hypernym word vectors.

For example, two words *Swallow* and *Pigeon* with hypernyms *Bird* and *Animal* are represented by

$$\psi(\text{Swallow}) = \pi_3 v(\text{Swallow}) + \pi_2 v(\text{Bird}) + \pi_1 v(\text{Animal}), \quad (1)$$

$$\psi(\text{Pigeon}) = \pi_3 v(\text{Pigeon}) + \pi_2 v(\text{Bird}) + \pi_1 v(\text{Animal}), \quad (2)$$

where π is a weight coefficient. If *Bird* is an important hypernym to decide visual features of *Swallow* and *Pigeon* as discussed in Introduction, our algorithm will give $\pi_2 > \pi_1, \pi_3$ to make the two vectors, $\psi(\text{Swallow})$ and $\psi(\text{Pigeon})$, close to each other as a result.

In the following, word-vector adaptation using tree structure is presented in Sec. 3.1 and weight optimization algorithm is presented in Sec. 3.2. To the best of our knowledge, our framework to make word vectors of visually similar concepts close to each other based on hypernyms of words is novel.

3.1 Adaptation of Word Vectors

For each word w, we assume that a set of its hypernyms $A(w)$ is given by a lexical database such as WordNet [32], with hierarchical tree structure. The adapted word vector for w is defined by

$$\psi(w) = \sum_{a \in A(w) \cup \{w\}} \pi_{d(a)} v(a) \quad (3)$$

where $d(a)$ is the depth of a word a in the tree structure.

In Eq. (3), we introduced restriction for weight coefficients: all words with depth d have the same weight. This is for solving the data insufficiency problem in zero-shot learning. For example, if we have training samples for (hyponyms of) *Bird*, this helps to determine the weight for the other words at the same depth level, e.g., *Fish* in Figure 1, without training samples. This restriction is reasonable in practice with several ten thousand of training concepts available on recent image and video datasets such as ImageNET [11].

3.2 Weight Optimization

The goal here is to determine weight coefficients π_d in Eq. (3) from training data. Our idea is to minimize distance between the following two similarity matrixes.

1. $K_{ij} = k(\psi(y_i), \psi(y_j))$: word similarity matrix where $\psi(y)$ is an adapted word vector for a concept y with indexes $i, j = 1, 2, \cdots, |\mathcal{Y}|$ for the concepts for training.

2. $K'_{ij} = k'(\zeta(y_i), \zeta(y_j))$: visual similarity matrix where $\zeta(y)$ is a visual vector of a concept y discussed later.

Here, k and k' are kernel functions such as the RBF kernel. Weight coefficients are optimized by

$$\hat{\pi} = \underset{\pi}{\operatorname{argmin}} \|K - K'\|. \quad (4)$$

The standard steepest descent method, in which the gradient is calculated by a finite difference approximation, is used to optimize π from an initial value of $\pi_d = 0$.

Finally, we discuss what the visual vector $\zeta(y_i)$ of a concept y_i is. With supervised learning methods, a concept is represented by a parametric classifier. This suggests that a concept can be represented by a vector of trained parameters. For example, for a softmax classifier, which is often used as the final layer of convolutional neural networks, is given by

$$\text{Softmax}(y_i|x) = \frac{\exp(f(y_i|x))}{\sum_j \exp(f(y_j|x))} \quad (5)$$

with a function

$$f(y_i|x) = a_i^T x + b_i. \quad (6)$$

Concatenation of the parameters a_i and b_i can be used as a visual vector, i.e., $\zeta(y_i) = (a_i^T, b_i)^T$.

4. EXPERIMENTS

We apply our word-vector adaptation to zero-shot learning experiments on two datasets, TRECVID and ImageNET. In the following, we first present a short summary of an application to zero-shot learning and then show our experimental settings and results.

Table 1: Evaluation on the TRECVID dataset. *Zero-shot baseline* uses the zero-shot learning framework in [28] with word vectors in [9] and Google Inception network [18]. *Ours* introduces the adapted word vectors to the baseline. The other three methods are from official submissions at TRECVID 2014. In the second column, *None* does not use training data, *Web Images* collects and uses training data from Google Search for each TRECVID concept, *TRECVID Videos* uses all training TRECVID videos. Mean Average Precision (Mean AP) for each method is reported.

Method	Training Data for TRECVID concepts	MeanAP(%)
Zero-shot baseline	None	6.37
Ours	None	**8.31**
Jiang et al.[33]	Web Images	1.21
McGuinness et al.[34]	Web Images	7.97
Snoek et al.[35]	TRECVID Videos	33.19

4.1 Application to Zero-shot Learning

Let $\mathcal{Y}$ and $\mathcal{Z}$ be disjoint sets of concepts for training and testing, respectively. In zero-shot learning [3, 5, 28], since training data is available only for concepts in $\mathcal{Y}$, a concept detector for $z \in \mathcal{Z}$ is build by a convex combination of trained detectors for $\mathcal{Y}$ as

$$f(z|x) = \frac{1}{C} \sum_{y \in \mathcal{Y}} \psi(y)^T \psi(z) f(y|x), \quad (7)$$

where $\psi(\cdot)$ is an adapted word vector, f is a detection function, $C = \|\psi(z)\| \| \sum_y \psi(y) f(y|x) \|$ is a normalization coefficient, and x is a testing image/video.

4.2 Experimental Settings

The TRECVID dataset consists of 800 hours of Internet videos with creative commons licenses used in the TRECVID 2014 semantic indexing task [10]. The goal is to detect 30 semantic concepts such as *Airplane*, *Dog*, and *Cityscape* from each video shot. Shot boundaries are provided with the dataset. The number of video shots is 547,634 for training and 107,806 for testing. For zero-shot experiments, we use the ImageNET images of 1,000 objects for training and TRECVID videos for testing. Note that TRECVID videos are not used for training. The evaluation measure is Mean Average Precision (Mean AP) over the 30 concepts, which is calculated by using the official toolkit and annotations.

The ImageNET dataset consists of 1,281,167 images for training, and 50,000 images for testing [1], with ground-truth labels of 1,000 object categories used at the ILSVRC 2012 competition [11]. We make a zero-shot learning task on this dataset under the following *leave-one-out* setting: repeat experiments 1,000 times so that 1) each category is selected once for testing, and 2) the rest 999 categories are used for training.

The word vectors in [9] trained on the Wikipedia dataset are used to extract adapted word vectors. The Google Inception Net [18] is used to train visual classifiers. RBF-kernels are used for the kernels in Sec. 3.2. We apply k-nearest

[1]We use the official "validation" set for testing, because the ground-truth labels for the official "test" set are not publicly available.

Table 2: Evaluation on the ImageNET dataset. *Zero-shot baseline* uses the zero-shot learning framework in [3] with word vectors in [9] and Google Inception network [18]. *Ours* introduces the adapted word vectors to the baseline. *Supervised* uses ImageNET training images. Mean Average Precision (Mean AP) and Top-5 Accuracy (Acc.) are reported.

Method	Mean AP (%)	Top-5 Acc. (%)
Zero-shot baseline	7.79	26.79
Ours	9.37	30.29
Supervised [18]	50.50	88.90

neighbor search to Eq.(7) by following the parameter settings in [3] and [28] for ImageNET and TRECVID, respectively.

4.3 Experimental Results

4.3.1 Performance Comparison

Table 1 and Table 2 show performance comparison on the TRECVID dataset and the ImageNET dataset, respectively. We see our method improves the performance on both datasets.

Compared with the other methods at the no-annotation semantic indexing task in TRECVID 2014 [10], which requires not to use TRECVID official training videos and allows to collect Web images by using search engines for supervised learning, we also see in Table 1 that our method outperforms its best performance of 7.97% in Mean AP. Note that our method did not use training images or videos for TRECVID concepts.

However, there is still a gap between supervised methods [18, 19] using official training data and our zero-shot method. To bridge the gap, improvement not only on word vectors, but also on visual classifiers is needed. For example, since we used detectors for 1,000 objects from ImageNET in our experiments, adding detectors for actions would be interesting as a next step. Audio analysis to detect concepts such as *Singing* and *MusicalInstruments* is also needed.

4.3.2 Analysis

Table 3 shows an example of tree structure for two object categories, *GoldFish* and *SnowBird*, with trained weight coefficients for each depth level. We see that the largest weight is given for the depth level of 12. From a biological point of view, this level mainly has *Classes* and *Orders* in the biological taxonomy, which are groups of animals whose physical features are similar to each other, e.g., *BonyFish* and *OscineBird*. Since physical features and visual features are often correlated, this supports our assumption that concepts have some hypernyms that decide their visual features.

Figure 3 shows the top 3 concepts from training data selected for *BoatShip* and *Baby*. We see that more visually similar concepts are selected with our method, such as *LifeBoat* and *OceanLiner* for *BoatShip*. For *Baby*, since its hypernyms or hyponyms are not included in the ImageNET dataset, the baseline method mainly selected animals, whose birth is often announced in news articles, e.g., *Panda*. On the other hand, our method improved the detection performance by selecting visually similar objects often seen with a baby, e.g. *Crib*.

Table 3: Tree structure for GoldFish and SnowBird. d: depth in the tree. π_d: obtained weights.

d	π_d	Tree structure
1	0.03	*Entity*
7	0.27	*Organism*
8	0.30	*Animal*
9	0.31	*Craniate*
10	0.29	*AquaticVertebrate* / *Bird*
11	0.28	*Fish* / *PasseriformBird*
12	**0.43**	***BonyFish*** / ***OscineBird***
13	0.31	*Teleost* / *Finch*
14	0.35	*SoftFinnedFish* / *SnowBird*
15	0.27	*CypriniformFish*
16	0.13	*CyprinidFish*
17	0.06	*GoldFish*

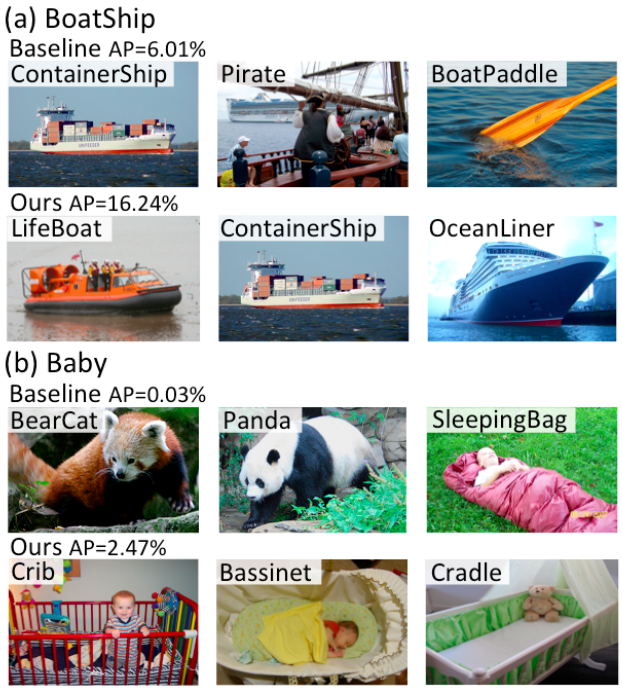

Figure 3: Top 3 ImageNET concepts selected by the baseline and our methods for two TRECVID concepts, (a) BoasShip and (b) Baby.

5. CONCLUSION

We proposed a framework of word-vector adaptation, which makes word vectors of visually similar concepts close to each other We applied our framework to zero-shot learning experiments on TRECVID and ImageNet datasets, and showed the performance improvement. Our future work will focus on audio and action analysis using word vectors.

Acknowledgments

This work was supported by JSPS KAKENHI 15K16019.

6. REFERENCES

[1] A.W.M. Smeulders, M. Worring, S. Santini, A. Gupta, and R. Jain. Content-Based Image Retrieval at the End of the Early Years. In IEEE Transactions on PAMI, vol.22, no.12, pp.1349–1380, 2000.

[2] T. Mensink, E. Gavves, and C.G.M. Snoek. Costa: Co-occurrence Statistics for Zero-shot Classification. *Proc. CVPR*, 2014.

[3] M. Norouzi, T. Mikolov, S. Bengio, Y. Singer, J. Shlens, A. Frome, G. Corrado, and J. Dean. Zero-shot Learning by Convex Combination of Semantic Embeddings. *Proc. ICLR*, 2014.

[4] A. Frome, G. S. Corrado, J. Shlens, S. Bengio, J. Dean, and T. Mikolov. Devise: A Deep Visual-semantic Embedding Model. *Proc. NIPS*, 2013.

[5] M. Jain, J.C. van Gemert, T.Mensink, and C.G.M. Snoek. Objects2action: Classifying and Localizing Actions without Any Video Example. *Proc. ICCV*, 2015.

[6] Q. Yu, J. Liu, H. Cheng, A. Divakaran, H. Sawhney. Multimedia event recounting with concept based representation. *ACM Multimedia*, 2012.

[7] T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient Estimation of Word Representations in Vector Space. *Proc. ICLR*, 2013.

[8] T. Mikolov, I. lya Sutskever, Kai Chen, Greg Corrado, and Jeffrey Dean. Distributed Representations of Words and Phrases and their Compositionality. *Proc. NIPS*, 2013.

[9] J. Pennington, R. Socher, and C.D. Manning. GloVe: Global Vectors for Word Representation. *In Empirical Methods in Natural Language Processing*, pp.1532–1543, 2014.

[10] G. Awad, A. Smeaton, W. Kraaij, G. Quéenot, R. Ordelman, and R. Aly. TRECVID 2015 – An Overview of the Goals, Tasks, Data, Evaluation Mechanisms and Metrics. *Proc. TRECVID workshop*, 2015.

[11] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A.C. Berg, and L. Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *In IJCV*, vol.115, no.3, pp.211–252, 2015.

[12] H. Schwenk. Continuous space language models. *In Computer Speech and Language*, vol. 21, 2007.

[13] C. Xing, D. Wang, X. Zhang, and C. Liu. Document classification with distributions of word vectors. *In Proc. APSIPA*, pp.1-5, 2014.

[14] S. Deerwester, S.T. Dumais, G.W. Furnas, T.K. Landauer, and R. Harshman, Indexing by Latent Semantic Analysis, *In Journal of the American Society for Information Science*, vol.41, 1990.

[15] C. Lampert, H. Nickisch, and S. Harmeling. Learning to Detect Unseen Object Classes by Between-Class Attribute Transfer. *Proc. CVPR*, 2009.

[16] Z. Akata, F. Perronnin, Z. Harchaoui, and C. Schmid. Label Embedding for Attribute-Based Classification. *Proc. CVPR*, 2013.

[17] A. Krizhevsky, I.Sutskever, and G.E. Hinton. ImageNet Classification with Deep Convolutional Neural Networks. *Proc. NIPS*, pp.1–9, 2012.

[18] C. Szegedy, W Liu, Y Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke and A. Rabinovich. Going Deeper with Convolutions *Proc. CVPR*, 2015.

[19] C.G.M. Snoek, et al., Qualcomm Research and University of Amsterdam at TRECVID 2015: Recognizing Concepts, Objects, and Events in Video. *Proc. TRECVID workshop*, 2015.

[20] M. Rohrbach, M. Stark, and B. Schiele. Evaluating Knowledge Transfer and Zero-Shot Learning in a Large-scale Setting. *Proc. CVPR*, 2011.

[21] L. Jiang, D. Meng, T. Mitamura, and A.G. Hauptmann. Easy Samples First: Self-Paced Reranking for Zero-Example Multimedia Search. *Proc. ACM Multimedia*, 2014.

[22] A. Habibian, T. Mensink, and C.G.M. Snoek. Composite Concept Discovery for Zero-shot Video Event Detection. *Proc. ICMR*, 2014.

[23] S. Wu, S. Bondugula, F. Luisier, X. Zhuang, and P. Natarajan. Zero-shot Event Detection using Multi-modal Fusion of Weakly Supervised Concepts. *Proc. CVPR*, pp.2665–2672, 2014.

[24] L. Jiang, S.-I Yu, D. Meng, T. Mitamura, and A.G. Hauptmann. Bridging the Ultimate Semantic Gap: A Semantic Search Engine for Internet Videos. *Proc. ICMR*, 2015.

[25] E. Gavves, T. Mensink, T. Tommasi, C.G.M. Snoek, and T. Tuytelaars. Active Transfer Learning With Zero-Shot Priors: Reusing Past Datasets for Future Tasks. *Proc. ICCV*, 2015.

[26] E. Kodirov, T. Xiang, Z. Fu, and S. Gong. Unsupervised Domain Adaptation for Zero-Shot Learning. *Proc. ICCV*, 2015.

[27] Z. Zhang, and V. Saligrama. Zero-Shot Learning via Semantic Similarity Embedding. *Proc. ICCV*, 2015.

[28] N. Inoue, and K. Shinoda. Vocabulary Expansion using Word Vectors for Video Semantic Indexing. *Proc. ACM Multimedia*, 2015.

[29] Z. Fu, T. Xiang, E. Kodirov, and S. Gong. Zero-Shot Object Recognition by Semantic Manifold Distance. *Proc. CVPR*, 2015.

[30] J.L. Ba, K. Swersky, S. Fidler, and R. Salakhutdinov. Predicting Deep Zero-Shot Convolutional Neural Networks Using Textual Descriptions. *Proc. ICCV*, 2015.

[31] X. Li, Y. Guo, and D. Schuurmans. Semi-Supervised Zero-Shot Classification With Label Representation Learning. *Proc. ICCV*, 2015.

[32] G. A. Miller. WordNet: A Lexical Database for English. *In Communications of the ACM*, vol.38, no.11, pp.39–41, 1995.

[33] L. Jiang, et al., CMU-Informedia at TRECVID Semantic Indexing, *Proc. TRECVID workshop*, 2014.

[34] K. McGuinness, et al., Insight Centre for Data Analytics at TRECVid 2014: Instance Search and Semantic Indexing, *Proc. TRECVID workshop*, 2014.

[35] C.G.M. Snoek, et al., Video Concept Detection by Deep Nets with FLAIR (MediaMill at TRECVID 2014). *Proc. TRECVID workshop*, 2014.