Variational Convolutional Networks for Human-Centric Annotations

Tsung-Wei Ke[1], Che-Wei Lin[1], Tyng-Luh Liu[1(✉)], and Davi Geiger[2]

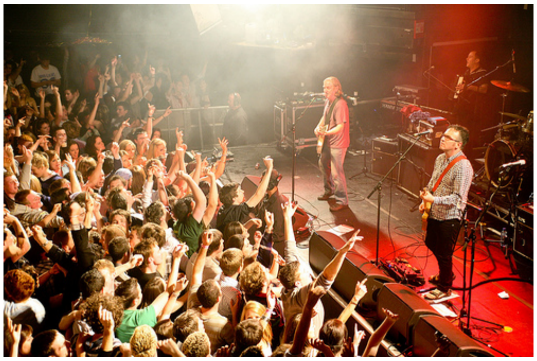
[1] Institute of Information Science, Academia Sinica, Taiwan
[2] Courant Institute of Mathematical Sciences, New York University, USA
liutyng@iis.sinica.edu.tw

Abstract. To model how a human would annotate an image is an important and interesting task relevant to image captioning. Its main challenge is that a same visual concept may be important in some images but becomes less salient in other situations. Further, the subjective viewpoints of a human annotator also play a crucial role in finalizing the annotations. To deal with such high variability, we introduce a new deep net model that integrates a CNN with a variational autoencoder (VAE). With the latent features embedded in a VAE, it becomes more flexible to tackle the uncertainly of human-centric annotations. On the other hand, the supervised generalization further enables the discriminative power of the generative VAE model. The resulting model can be end-to-end fine-tuned to further improve the performance on predicting visual concepts. The provided experimental results show that our method is state-of-the-art over two benchmark datasets: MS COCO and Flickr30K, producing mAP of 36.6 and 23.49, and PHR (Precision at Human Recall) of 49.9 and 32.04, respectively.

1 Introduction

Exploring the intriguing relationships between language and vision models has recently become an active research topic in computer vision community. Notable efforts include generating text descriptions for images, *e.g.*, [1–4] or videos [5, 6], while their main idea is to discover important spatial or spatial-temporal visual information and express it with appropriate wording. Another interesting development has been centered on the problem of *image question answering* [7]. The task often results in a more complex and challenging vision-language computational model, which would require learning different levels/types of semantics to address the various combinations of questions and underlying scenes. Yet, in contrast to dealing with *image captioning*, there are also techniques, *e.g.*, [8], aiming at solving language-to-image problems to generate images according to the given descriptions.

We instead focus on the problem of *human-centric annotations* [9] for images, which can be considered a subtask of image captioning. From popular image caption collections such as MS COCO [10] and Flickr30K [11], one can conclude that it is inappropriate and also impossible to use a caption to name every content in the image. For example, when describing a basketball in a scene, a sensible caption would not state "a round basketball" but simply "a basketball" instead. However, the same concept of "round" would become meaningful if the shape of a target object such as a building or



Visual Concepts

air
band
concert
crowd
fans
hands
musical
performing
performs
plays
put
stage

Fig. 1. An image example with 12 visual concepts as the ground truth.

a church is to be emphasized. The example pinpoints that image annotations are highly correlated to *important* properties of the image, and are inherently linked to the annotator's viewpoints. Following [12], we consider the image annotations termed as *visual concepts*, whose labeling depends on the *subjective* judgment of a human annotator. To construct the set of visual concepts from an image caption dataset, we single out those words with the top most appearances in the captions. The ground truth of visual concepts of an image can then be formed by intersecting all its captions with the set of visual concepts. Figure 1 shows an image and the corresponding visual concepts, with which the task of human-centric annotations aims to predict.

Recent studies have shown that learning to predict human-centric annotations could improve the performances of image captioning [13] and image question answering [4]. Misra *et al.* [12] consider human-centric annotations as visual concepts. Their method can predict both the visual concepts and their presence in an image whether a human would annotate the concept or not. Motivated by the promising progress, we aim to more satisfactorily address the problem of human-centric annotations. In particular, to model the subtlety of how annotations are achieved, we decompose the process into two stages. We first predict the presences of all the available concepts in an image, and then *simulate* how a human would decide their relevance in the final annotations. The reasoning can be realized by fusing a Convolutional Neural Network (CNN) with a Variational Auto-Encoder (VAE) [14], where the resulting network architecture will be termed as a Variational CNN (VCNN). The annotation process by the proposed VCNN proceeds as follows. It starts by using a deep CNN to output the probabilities of all the concepts, and then passes the visual features and the information (or more precisely, the probabilities) of concept presence to a (stacked) VAE model to generate the annotation predictions. The proposed two-stage processing can be seamlessly coupled to form an end-to-end VCNN model, as illustrated in Figure 2. One crucial difference between our method and [12] is that with the proposed VCNN model, the probability of annotating a particular visual concept is conditioned on the presence information of all the concepts, rather than the concept alone.

2 Related Work

Methods dealing with image captioning can be divided into two categories, namely, *caption retrieval* and *caption generation*. For caption retrieval, Devlin *et al.* [15] propose to search for a set of the nearest neighbor images, and gather from them the candidate captions. The description that is most similar to the other candidates is chosen from the set to represent the query image. In [16], Klein *et al.* exploit the alignment between linguistic descriptors, derived from the Gaussian-Laplacian Mixture Model, with CNN-based visual features for caption retrieval. For caption generation, most techniques rely on using deep net models. A popular formulation is to use two subnetworks, which typically consists of a CNN as the vision model and a Recurrent Neural Network (RNN) as the language model [1–4, 17]. And the variants of RNN include the Long Short-Term Memory (LSTM) network [2, 17], the bidirectional RNN [1], etc. Furthermore, in [17], Jia *et al.* extend the input to the LSTM with the extracted semantic information to improve the performance of image caption generation. Xu *et al.* [3] introduce an attention model that aims to help LSTM to emphasize salient objects while generating descriptions. In [4, 13], the CNN module is fine-tuned to detect possible attributes/words in the image, and the resulting prediction is then taken as the input to the language model.

Apart from dealing with a single image, video description generation has also gained increasing attention and interest. Rohrbach *et al.* [18] formulate the task as a machine translation problem by learning a CRF to yield the semantic representation and translating it into the video description. In [19], a factor graph is constructed to combine visual detections on subject, verb, object and scene elements with linguistic statistics to infer the most likely tuple for sentence generation. Yao *et al.* [5] propose to capture spatio-temporal dynamics and build an attention model. With the temporal attention, the most relevant video subsequences are selected for RNN to describe. Venugopalan *et al.* [6] divide text generation into two subtasks: a stacked LSTM network is used to first encode a video sequence and then decode it into a sentence.

Understating the underlying factors behind human-centric annotations has been an interesting topic in computer vision. The analysis conducted by Berg *et al.* [9] investigates three types of factors, including composition, semantics, and context, which are all closely related to how people evaluate the importance of a content in the image. In [20], Turakhia *et al.* model the attribute dominance and argue that more dominant attributes would be described first when seeing an image. Yun *et al.* [21] explore the relationships among images, eye movements and descriptions, and use a gaze-enabled model for detection and annotation. In addition, there are several techniques aiming at directly predicting user-supplied tags. Chen *et al.* [22] propose to pre-train a CNN on easy images to learn an initial visual representation. The weights are then transferred and fine-tuned on realistic images. When testing with image-tag pairs, the resulting two-stage learning approach is shown to outperform schemes with only fine-tuning. In [23], Izadinia *et al.* have focused on predicting 5400 tags over a dataset with 5M Flickr images. Besides recognizing the user-supplied tags, [12, 13, 24] are to predict words filtered from the image captions. Taking these words as noisy labels, Misra *et al.* [12] propose a factor-decoupling model to implicitly predict visual labels, where the classifier is trained essentially with the human-centric annotations. In [24], Joulin *et al.* have attempted to predict 100,000 words over an extremely large-scale dataset with approximately 100M images.

The VAE model by Kingma *et al.* [14] is established by integrating a top-down deep generative network with a bottom-up recognition network. The recognition model is optimized with respect to a variational lower bound to achieve approximate posterior inference. Its extension to semi-supervised applications is proposed in [25]. Another generalization can be found in the so-called Importance Weighted Auto-Encoder (IWAE) [26], which employs a similar network as the VAE, but is learned with a tighter log-likelihood lower bound. Besides these efforts, a popular application of VAE is to include the model to enable variational inference with an RNN, *e.g.*, [27–29]. In [27], Fabius *et al.* generalize the encoding-decoding procedure to the temporal domain. While the distribution over the latent variable is decided from the last state of the recurrent recognition model, the recurrent generative model outputs data with the initial state computed from the updated latent representation. Recently, Chung *et al.* [29] introduce a high-level latent variable into an RNN to model the variability in rich-structured sequential data. The VAE-based models are also used in tackling image generation [28, 30].

3 Our Method

We begin by casting the problem of how a human would annotate an image as follows. Let $\mathcal{V} = \{v_k\}_{k=1}^K$ be a set of K visual concepts. Then, the human-centric annotations for a given image $\mathbf{x}$ form a subset of $\mathcal{V}$, denoted as

$$\mathcal{A}_{\mathbf{x}} = \{v_k \,|\, y_k = 1,\ 1 \leq k \leq K\} \subseteq \mathcal{V} \tag{1}$$

where $y_k \in \{0, 1\}$ is a binary random variable specifying whether visual concept v_k is mentioned in the annotations. Analogous to the formulation in [12], we define a latent random variable $c_k \in \{0, 1\}$ as the *visual label* of v_k and use it to indicate whether the visual concept v_k is present in the image. For convenience, we write $\mathbf{c} = (c_1, \ldots, c_K)^\top$ and marginalizing over $\mathbf{c}$ would yield

$$p(y_k|\mathbf{x}) = \sum_{\mathbf{c}\in\{0,1\}^K} p(y_k|\mathbf{c}, \mathbf{x})\, p(\mathbf{c}|\mathbf{x}) \approx p(y_k|\mathbf{c}^*, \mathbf{x})\, p(\mathbf{c}^*|\mathbf{x}) \tag{2}$$

where the approximation is the result of assuming that the probability distribution $p(\mathbf{c}|\mathbf{x})$ peaks very sharply at $\mathbf{c}^*$. Indeed, the approximation in (2) is *exact* if we do have the factual information about the presence of each concept v_k. That is, the closer $\mathbf{c}^*$ is to the (unavailable) ground truth of visual labels, the more valid the approximation will be. With (2), we carry out our method in two sequential stages.

1. Construct a convolutional neural network (CNN) to yield $p(\mathbf{c}^*|\mathbf{x})$.
2. Learn a variational auto-encoder (VAE) to output $p(y_k|\mathbf{c}^*, \mathbf{x})$ for each concept v_k.

Details about how we sequentially learn the two types of neural networks and fine-tune them as an end-to-end system will be described in the next two subsections. We now remark that unlike the formulation in [12], we estimate $p(y_k|\mathbf{x})$ by marginalizing over $\mathbf{c}$ rather than just c_k. The distinction is crucial, as in many practical situations, the mentioning of a visual concept v_k depends on not only c_k but also the presence of other relevant visual concepts.

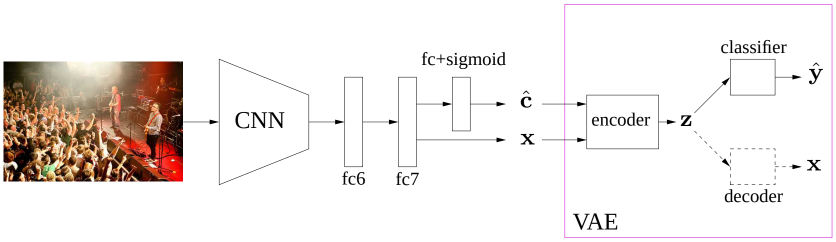


Fig. 2. We couple CNN and VAE to form a variational CNN for human-centric annotations.

3.1 On $p(\mathbf{c}^*|\mathbf{x})$

To model the multi-label learning for $p(\mathbf{c}^*|\mathbf{x})$, we assume the independence of visual labels in an image. That is,

$$p(\mathbf{c}^*|\mathbf{x}) = \prod_{k=1}^{K} p(c_k^*|\mathbf{x}). \tag{3}$$

We employ the VGG net [31] pre-trained on ImageNet as the adopted CNN, and modify the network by adding on top of the fc7 a discriminative classifier composed of a fully-connected layer and a sigmoid function. (See Figure 2.) Due to the lack of visual-label ground truth in the training dataset, we use the information of visual concepts as the *noisy* ground truth and fine-tune the VGG net with the human-centric annotations to yield the probabilities of visual labels.

3.2 On $p(y_k|\mathbf{c}^*, \mathbf{x})$

With the CNN learned in the first stage, we extract features from fc7 and represent each image with $\mathbf{x} \in \mathbb{R}^L$. ($L = 4096$ for VGG.) On the other hand, simply using $\mathbf{c}^* \in \{0,1\}^K$ does not fully utilize the visual-label information. We instead consider their probabilities, and denote them by $\hat{\mathbf{c}} \in \mathbb{R}^K$, whose kth component is the probability $p(c_k^*|\mathbf{x})$ yielded by the CNN. To simplify the notation, we write $\mathbf{w} = \hat{\mathbf{c}} \oplus \mathbf{x} \in \mathbb{R}^{K+L}$ where $\oplus$ denotes vector concatenation. Further, we use $\hat{y}_k$ to denote $p(y_k|\mathbf{w})$, the probability of mentioning concept v_k in the annotations and let $\hat{\mathbf{y}} = (\hat{y}_1, \ldots, \hat{y}_K)^\top \in \mathbb{R}^K$. Before we explain the proposed VAE formulation, we first describe a naïve approach to predicting the probabilities of visual concepts. Assume that the training dataset has N images, represented by $\{(\mathbf{x}_i, \mathbf{y}_i^*)\}_{i=1}^N$, where $\mathbf{y}_i^* \in \{0,1\}^K$ is the visual-concept ground truth of image $\mathbf{x}_i$. We can construct a neural network (detailed in subsection 4.4) to directly model $p(y_k|\hat{\mathbf{c}}, \mathbf{x}) = p(y_k|\mathbf{w})$ with a cross-entropy objective function:

$$\mathcal{E}_{\text{naive}} = -\sum_{i=1}^{N}\sum_{k=1}^{K} \mathbb{I}(\mathbf{y}_i^*(k) = 1)\, \log p(y_k|\mathbf{w}_i) \tag{4}$$

where $\mathbb{I}(\cdot)$ is the indicator function and $\mathbb{I}(\mathbf{y}_i^*(k) = 1)$ verifies that visual concept v_k is mentioned in the ground truth $\mathbf{y}_i^*$.

We next describe the proposed VAE model. Our method is inspired by [25], but we extend it to a combined generative and supervised learning. To begin with, we hypothesize the following data generative process:

$$p_{\boldsymbol{\theta}}(\mathbf{z}) = \mathcal{N}(\mathbf{z}|\mathbf{0}, \mathbf{I}) \quad \text{and} \quad p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z}) = f(\mathbf{x}; \mathbf{z}, \boldsymbol{\theta}) \tag{5}$$

where the prior of the latent variable $\mathbf{z} \in \mathbb{R}^D$ is assumed to be the centered isotropic multivariate Gaussian and $f(\cdot)$ is a suitable likelihood function, while $\boldsymbol{\theta}$ are VAE generative parameters. We then introduce a distribution $q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{w})$ to approximate the true posterior distribution $p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{w})$ where $\boldsymbol{\phi}$ are variational parameters. More specifically, we have

$$q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{w}) = \mathcal{N}(\mathbf{z}|\boldsymbol{\mu}_{\boldsymbol{\phi}}(\mathbf{w}), \mathrm{diag}(\boldsymbol{\sigma}^2_{\boldsymbol{\phi}}(\mathbf{w}))) \tag{6}$$

where $\boldsymbol{\mu}_{\boldsymbol{\phi}}(\mathbf{w})$ and $\boldsymbol{\sigma}_{\boldsymbol{\phi}}(\mathbf{w})$ respectively denote a vector of means and a vector of standard deviations. In our formulation, both are represented by the neural network. Then we can derive the *variational lower bound*, $\mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{w})$:

$$\begin{aligned} \log p_{\boldsymbol{\theta}}(\mathbf{w}) \geq \mathcal{L}(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{w}) &= \mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{w})}[\log p_{\boldsymbol{\theta}}(\mathbf{w}|\mathbf{z}) + \log p_{\boldsymbol{\theta}}(\mathbf{z}) - \log q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{w})] \\ &= -D_{KL}(q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{w}) \| p_{\boldsymbol{\theta}}(\mathbf{z})) + \mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{w})}[\log p_{\boldsymbol{\theta}}(\mathbf{w}|\mathbf{z})]. \end{aligned} \tag{7}$$

The derivation so far follows the standard analysis of variational approximation. To incorporate the ground-truth information of visual concepts and to boost the discriminative power to our model, the last term in (7) is approximated by

$$\mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{w})}[\log p_{\boldsymbol{\theta}}(\mathbf{w}|\mathbf{z})] \approx \mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{w})}[\log p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})] + \mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{w})}[\log p_{\boldsymbol{\theta}}(\mathbf{y}|\mathbf{z})] \tag{8}$$

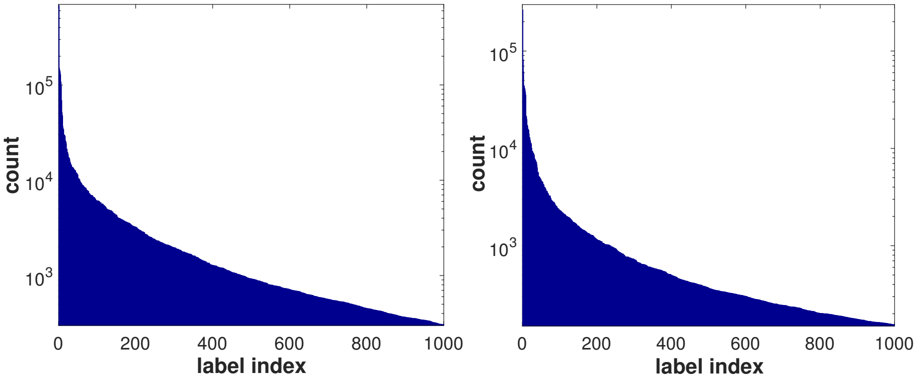
where the approximation decouples the joint generative process into unsupervised decoding and classification, respectively. (See Figure 2.) Thus, the objective function to be minimized in learning the supervised VAE is defined by

$$\begin{aligned} \mathcal{E}_{\mathrm{VAE}}(\boldsymbol{\theta}, \boldsymbol{\phi}, \mathbf{w}) = \sum_{i=1}^{N} D_{KL}(q_{\boldsymbol{\phi}}(\mathbf{z}_i|\mathbf{w}_i) \| p_{\boldsymbol{\theta}}(\mathbf{z}_i)) - \mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{w})}[\log p_{\boldsymbol{\theta}}(\mathbf{x}_i|\mathbf{z}_i)] \\ - \sum_{i=1}^{N} \sum_{k=1}^{K} \mathbb{I}(\mathbf{y}_i^*(k) = 1) \, \log p_{\boldsymbol{\theta}}(y_k|\mathbf{z}_i). \end{aligned} \tag{9}$$

Using the reparameterization trick for $\mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{w})}[\log p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z})]$ and the KL divergence closed-form:

$$D_{KL}(q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{w}) \| p_{\boldsymbol{\theta}}(\mathbf{z})) = -\frac{1}{2} \sum_{j=1}^{D} (1 + \log(\sigma^2_{\phi,j}) - \mu^2_{\phi,j} - \sigma^2_{\phi,j}) \tag{10}$$

where $\sigma^2_{\phi,j}$, $\mu^2_{\phi,j}$ are respectively the jth elements of $\boldsymbol{\sigma}^2_{\phi}(\mathbf{w})$ and $\boldsymbol{\mu}^2_{\phi}(\mathbf{w})$, the supervised VAE can be learned with the Stochastic Gradient Variational Bayes (SGVB) [14]. Having sequentially trained the CNN and the VAE, we link the two models and remove the decoder module (shown as the dotted rectangle in Figure 2) from the architecture. This way we can enhance the discriminative power of the VCNN by end-to-end fine-tuning with only the classification loss function $\mathcal{E}_{naive}$ defined in (4).



(a) MS COCO

(b) Flickr30K

Fig. 3. Total number of presences for each *visual concept* in a dataset.

4 Experimental Results

We evaluate the proposed VCNN model on two image caption datasets: MS COCO [10] and Flickr30K [11]. Numbers, punctuation symbols, accents and special characters are removed from the captions. Every caption is then lower-cased and tokenized into words. For each dataset, we select $K = 1000$ most common words, including nouns, verbs, adjectives and other parts of speech, to form the set of visual concepts for human-centric annotations.

4.1 Datasets

MS COCO [10] includes 82,783 training images and 40,504 validation images. Each image is provided with five human-annotated captions. Following [12], we split the collection of validation images into equally-sized validation and test set, where the split is the same as that in [12]. Flickr30K [11] is composed of 158,915 crowd-sourced captions describing 31,783 images. As in [1], we divide them into training, validation and test sets, each of which contains 29,783, 1000, and 1000 images, respectively.

To generate the ground-truth annotations of visual concepts for each training image, we use a 1000-dimensional binary vector to indicate which of the selected 1000 most common words appeared in any of the 5 corresponding captions. Based on these binary vectors of ground truth, our models and [12, 13] are all learned with the same setting in the experiments. Unless otherwise mentioned, we report the results on the test sets of MS COCO and Flickr30K.

4.2 Cost-Sensitive Criterion

Because the visual concepts are derived from image captions annotated by humans, some words are mentioned much more frequently than the others. For example, in the two datasets, `boy`, `girl` would be used more often than `lion` or `elephant`. We have counted the total number of each visual concept present in the images over MS COCO and Flickr30K. The results are plotted in Figure 3. Such an imbalanced distribution of word labels could cause biases on learning the VAE model. To address this issue, we separate the set of visual concepts into a common set and a rare set, denoted by $\mathcal{V} = \mathcal{V}_c \sqcup \mathcal{V}_r$. We extend the classification loss term in (9) into a cost-sensitive one by

$$\mathcal{E}_{\text{cs}}(\mathbf{y}) = \left(\sum_{\mathbf{x}_i \in \mathcal{V}_c} \lambda_c + \sum_{\mathbf{x}_i \in \mathcal{V}_r} \lambda_r \right) \sum_{k=1}^{K} \mathbb{I}(\mathbf{y}_i^*(k) = 1) \, \log p_\theta(y_k|\mathbf{z}_i)) \tag{11}$$

where λ_c, λ_r are the cost-sensitive weighting parameters. In the experiments, we set $\lambda_r > \lambda_c$ to avoid the penalty dominance from misclassifying common words.

4.3 Stacked VAE

We also try stacking two latent variables to discover more effective architecture of the supervised VAE. The architecture of our stacked VAE is shown in Figure 4. Specifically, we first learn a latent variable $\mathbf{z}_1$ based on Section 3.2 and subsequently learn $\mathbf{z}_2$ using $\mathbf{z}_1$. The deep generative model can be described by

$$p(\mathbf{w}, \mathbf{z}_1, \mathbf{z}_2) = p(\mathbf{x}, \hat{\mathbf{c}}, \mathbf{z}_1, \mathbf{z}_2) = p(\hat{\mathbf{c}})p(\mathbf{z}_1)p(\mathbf{z}_2|\mathbf{z}_1)p(\mathbf{x}|\mathbf{z}_2). \tag{12}$$

Analogously, we can derive the variational lower bound as

$$\begin{aligned} \mathcal{L}_{\text{stacked}}(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{w}) \approx & -D_{KL}(q_{\boldsymbol{\phi}}(\mathbf{z}_1|\mathbf{w}) \| p_{\boldsymbol{\theta}}(\mathbf{z}_1)) - D_{KL}(q_{\boldsymbol{\phi}}(\mathbf{z}_2|\mathbf{z}_1) \| p_{\boldsymbol{\theta}}(\mathbf{z}_2|\mathbf{z}_1)) \\ & + \mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}_2|\mathbf{w})}[\log p_{\boldsymbol{\theta}}(\mathbf{x}|\mathbf{z}_2) + \log p_{\boldsymbol{\theta}}(\hat{\mathbf{c}})]. \end{aligned} \tag{13}$$

Using the holistically-nested structure proposed by Xie [32], we add two side-output classifiers to the latent variables. The summation (before applying the activation) and the probability of the side-output of latent variable $\mathbf{z}_k, k \in \{1, 2\}$ are denoted as $\mathbf{S}^{(k)}, p^{(k)}(\mathbf{y})$, where $p^{(k)}(\mathbf{y}) = \psi(\mathbf{S}^{(k)})$, and ψ is the nonlinear activation function of the classifiers. Then we can construct another classifier by fusing these side-output layers:

$$\mathbf{S}^{(3)} = \boldsymbol{\alpha}_1 \mathbf{S}^{(1)} + \boldsymbol{\alpha}_2 \mathbf{S}^{(2)} \quad \text{and} \quad \mathbf{y}^{(3)} = \psi(\mathbf{S}^{(3)}) \tag{14}$$

where $\boldsymbol{\alpha}_1$ and $\boldsymbol{\alpha}_2$ are learnable weights. The objective function of the stacked VAE can be obtained by replacing the classification loss term in (9) with $\sum_{k=1}^{3} \mathcal{E}_{\text{cs}}(\mathbf{y}^{(k)})$.

4.4 Implementation Details

As the scales of the two datasets are significantly different, the adopted VAE architecture differs in depth for MS COCO and Flickr30K datasets. In testing with MS COCO, we set the layer-wise number of neurons in the encoder as 5096-2500-2500/2500. We also construct the generative decoder with the size of 2500-2000-4096 and the label classifier with the size of 2500-2000-1000. For Flickr30K, the sizes of encoder, decoder and label classifier are set as 5096-2500/2500, 2500-4096 and 2500-1000, respectively. The VAE model is first pre-trained to optimize (9), learn to generate visual features and

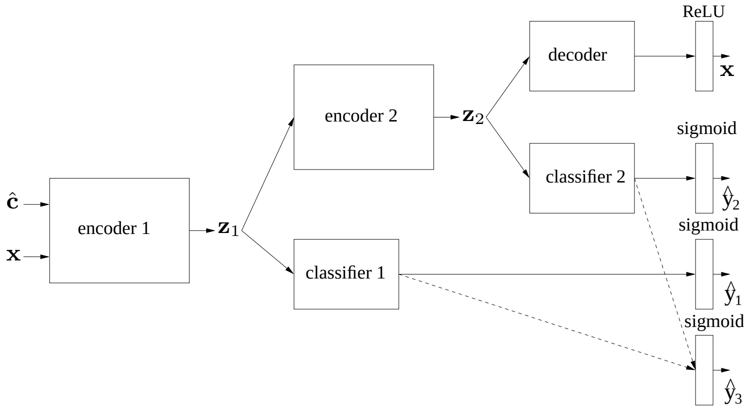


Fig. 4. The architecture of the stacked VAE network.

predict the visual concepts. We use the sigmoid function and ReLU [33] as the respective activation function for label classifier and for the generative decoder. ReLU is also used as the activation function of every hidden unit in the VAE. The values of hyperparameters used in training are stated as follows: batch size of 256, learning rate of 0.001 and weight decay of 0.0005. The network is trained for 20 epoches. For cost-sensitive learning, we separate $\mathcal{V}$ into two subsets, $\mathcal{V}_c$ and $\mathcal{V}_r$, that $\mathcal{V}_c$ is composed of the 100 most common words and the rest of the words belong to $\mathcal{V}_r$. We set the cost-sensitive weight λ_c to 0.001 and λ_r to 1 for balancing penalty.

To construct the stacked-VAE, we first remove the generative decoder from the pretrained network and keep the encoder and the classifier, denoted as enc_1 and $class_1$. We initialize a new encoder (enc_2), generative decoder (rec_2) and label classifier ($class_2$) that are later attached on top of enc_1. enc_2 shares the same latent variable $\mathbf{z}_1$ with $class_1$. In MS COCO, we set the enc_2 to the size of 2500-2500-2500, rec_2 to 2500-2000-4096 and $class_2$ to 2500-2000-1000. For Flickr30K, these layer-wise sizes are set to 2500-2500, 2500-4096 and 2500-1000. We follow the same learning procedure and strategy to train the stacked-VAE and start learning the value fusion weights $\boldsymbol{\alpha}_1$ and $\boldsymbol{\alpha}_2$ from 0.5. For end-to-end fine-tuning the proposed VCNN, we set batch size to 1, learning rate to 0.000015 and weight decay to 0.0005. Stochastic Gradient Descent (SGD) is used to optimize the objective function and update parameter weights in our model. It takes seven days to train the complete model on one Titan-X GPU.

4.5 SGVB Estimator Evaluation

The VAE is learned to optimize the objective function (9), consisting of a SGVB estimator and a supervised term . We argue that such a criterion can help the recognition model (encoder) to capture global information of the contents in an image. To demonstrate the advantage of the proposed scheme, we compare our model with the naïve model mentioned in Section 3.2. We construct a network which has the same architecture with the VAE, except that the generative decoder is removed. Then, the network is optimized with (4) by adopting the same learning strategy described in Section 4.4 and using SGD to update neuron weights. We report the results on Mean Average Precision of the visual concepts over MS COCO and Flickr30K with 224×224 input images. Table 1 shows that owing to the use of supervised VAE, our model can improve the mAP from 32.90 to 33.07, 22.67 to 22.90 for MS COCO and Flickr30K, respectively. We also perform similar comparisons for the stacked-VAE. The mAP performance has increased by 0.08 and 0.09 for the respective datasets.

Table 1. Ablation evaluations on SGVB (mAP).

Method	MS COCO	Flickr30K
VCNN w/o SGVB	32.90	22.67
VCNN with SGVB	33.07	22.90
stacked-VCNN w/o SGVB	33.67	22.95
stacked-VCNN with SGVB	33.75	23.04

4.6 Cost-Sensitive Evaluation

As discussed in Section 4.2, the imbalanced distribution of various labels of visual concepts could cause biases on learning. Fitting common labels will be too dominant such that the penalty on error prediction of uncommon words may be ignored. We have proposed a cost-sensitive criterion to address this issue and help learn the model more efficiently. Table 2 reports that mAP of VCNN and that of stacked-VCNN are originally 33.07 and 33.75, and are boosted to 33.37 and 34.32 if the cost-sensitive criterion is considered in testing with the MS COCO dataset. As for Flickr30K, the criterion also improves mAP from 22.90 to 23.01 and 23.04 to 23.49 for both networks.

Table 2. Cost-sensitive learning (mAP).

Method	MS COCO	Flickr30K
VCNN w/o c.s	33.07	22.90
VCNN with c.s	33.37	23.01
stacked-VCNN w/o c.s	33.75	23.04
stacked-VCNN with c.s	34.32	23.49

4.7 Stacked-VAE Evaluation

The proposed stacked architecture can help VAE to learn richer representations in both low-level and high-level latent variables, namely, $\mathbf{z}_1$ and $\mathbf{z}_2$. We compare the performance of non-stacked and stacked architecture on both the naïve model and VAE. In

Table 1, we show that stacked VAE can improve the performance and increase mAP by 0.68 in MS COCO and 0.14 in Flickr30K. Even with the naïve network (*i.e.*, without SGVB estimator), the mAP has been boosted from 32.90 to 33.67 in MS COCO, and 22.67 to 22.95 in Flickr30K. We also evaluate such architecture jointly learned with the cost-sensitive criterion. In Table 2, the resulting models indeed benefit from stacked structure that mAP is respectively improved from 33.37 to 34.32 and 23.01 to 23.49 that both are state-of-the-art for the MS COCO and Flickr30K datasets.

4.8 More on MS COCO and Flickr30K

Besides focusing on mAP, we follow [12, 13] and conduct further evaluations based on Precision at Human Recall (PHR). Chen *et al.* [34] propose an evaluation metric which takes human agreement into consideration for the task of word prediction. A "human recall" value is an estimated probability of human using the word when viewing the image. The metric computes human recall given multiple references per image and retrieves precision at this human recall value as PHR. As pointed out in [34], it is more stable to evaluate with PHR than with mAP for the prediction of human annotations.

Two training schemes are both considered in [12, 13] to learn the network models, including (1) using only fine-tuning and (2) employing weakly-supervised approach of Multiple Instance Learning (MIL) [35]. With the first scheme, the models are end-to-end fine-tuned with 224×224 input images. In the MIL formulation, a noisy-OR version of MIL [35] is adopted. That is, the probability of a word is computed by scrutinizing the instance probabilities from individual patches of the image:

$$1 - \prod_{u,v}(1 - p^{w_i}_{u,v}) \tag{15}$$

where $p^{w_i}_{u,v}$ denotes the probability of word w_i at region (u, v). For the noisy-OR MIL learning, we transform the fully-connected layers in our model to 1×1 convolutional layers and also resize the input image to 565×565. The convolutional network then performs sliding over the image with a 224×224 window and a stride of 32, which would produce a 12×12 map at both fc7 and fc8. The probability for each label is then computed ranging over this 12×12 spatial grid. Unless otherwise stated, models learned with noisy-OR MIL are marked with MIL in our reported results.

To evaluate the various models by their predictions of visual concepts, we sort the annotations into the following categories of part of speech (`POS`): Nouns (`NN`), Verbs (`VB`), Adjectives (`JJ`), Pronouns (`PRP`) and Prepositions (IN). We report results based on these POS tags and also compute overall mAP and PHR, which are respectively denoted as `All` in Tables 3-6. We take VCNN with stacked-VAE, which is learned with the cost-sensitive criterion, as our final model and compare it with [12] and [13]. The experimental results we obtained are state-of-the-art in both MS COCO and Flicrk30K. Table 3 shows that our model yields better mAP results than the other two. We achieve at mAP of 34.3 (direct classification) and 36.6 (MIL). When the performance is evaluated with the PHR criterion, the overall precisions by our method are respectively 47.1 and 49.9, as in Table 4. For the Flickr30K dataset, we only conduct experiments with 224×224 input images in that [12] has not yet released the code and also does not report

Table 3. Mean Average Precision of MS COCO (mAP).

Method	`NN`	`VB`	`JJ`	`DT`	`PRP`	`IN`	`Others`	`All`
Classification [13]	34.9	18.1	20.5	32.8	19.2	21.8	16.3	29.0
Classification+Latent [12]	38.7	20.1	22.6	33.8	21.2	23.0	17.5	32.0
VCNN (Ours)	**41.6**	**21.5**	**24.3**	33.6	**22.2**	**23.5**	17.3	**34.3**
MILVC [13]	41.6	20.7	23.9	33.4	20.4	22.5	16.3	34.0
MILVC+Latent [12]	44.3	22.3	25.8	34.4	21.8	23.6	17.3	36.3
MIL-VCNN (Ours)	**44.6**	**22.7**	**26.1**	33.9	**22.5**	23.4	17.2	**36.6**

Table 4. Precision at Human Recall of MS COCO (PHR).

Method	`NN`	`VB`	`JJ`	`DT`	`PRP`	`IN`	`Others`	`All`
Classification [13]	42.5	30.4	33.9	40.5	30.4	30.7	23.8	38.2
Classification+Latent [12]	47.8	33.7	37.9	42.5	34.2	34.4	29.0	42.9
VCNN (Ours)	**52.7**	**36.3**	**44.1**	41.0	**36.8**	**35.9**	26.9	**47.1**
MILVC [13]	52.7	32.8	40.5	40.3	32.2	33.0	24.6	45.8
MILVC+Latent [12]	55.5	36.3	44.7	42.9	32.1	37.3	26.4	48.9
MIL-VCNN (Ours)	**56.8**	**37.2**	**44.9**	**43.1**	**36.3**	**37.4**	**26.7**	**49.9**

results on Flickr30K. We implement the method of [12] on our own and follow their training strategy to obtain the experimental results. It can be inferred from both Table 5 and Table 6 that the proposed VCNN still yields better results, 23.49 for mAP and 32.04 for PHR, while the techniques of [12] and [13] achieve similar performances.

4.9 Qualitative Results

Figure 5 shows six examples of how VCNN correctly predicts *visual concepts* by inferring from the distribution of relevant *visual labels*. In the left image of the top row, our model predicts *shovel* should be mentioned with the knowledge of presence of `boy`, `playing`, `holding`, `yellow` and `sand`. In most situations, humans tend not to mention the typical color of the object. For example, Rocks are commonly `gray`. Likewise, our VCNN is able to lower the probability of mentioning the specific visual concept in such a condition. In the left image of the second row, the visual concept `gray` is removed when `mountains`, `climbing`, `hill` and `rocky` are already detected.

5 Discussions

We have proposed a new deep net model to address the problem of human-centric annotations. Our method relies on decomposing the annotation probability that results in two relevant subtasks, where we have used a CNN and a VAE to tackle them, respectively. The integrated architecture is a variational convolutional network that can be end-to-end

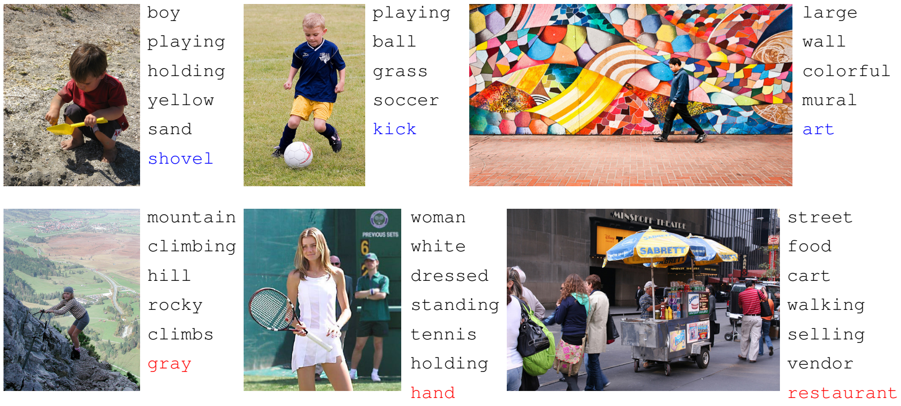


Fig. 5. Visualization results. The proposed VAE predicts that the visual concepts marked in blue should be additionally mentioned, while those marked in red should be removed, given the information of the presence of the visual labels (marked in black).

Table 5. Mean Average Precision of Flickr30K (mAP).

Method	NN	VB	JJ	DT	PRP	IN	Others	All
Classification [13]	24.80	17.20	17.38	28.50	20.38	23.40	15.72	21.75
Classification+Latent [12]	24.61	16.52	16.79	28.42	20.30	23.40	16.43	21.44
VCNN (Ours)	**26.99**	**18.66**	**18.58**	**28.81**	**20.55**	**24.36**	15.84	**23.49**

Table 6. Precision at Human Recall of Flickr30K (PHR).

Method	NN	VB	JJ	DT	PRP	IN	Others	All
Classification [13]	31.12	27.61	27.33	32.43	31.24	32.80	18.47	29.49
Classification+Latent [12]	31.80	25.73	25.64	35.98	30.15	32.67	21.89	29.36
VCNN (Ours)	**33.82**	**28.41**	**31.72**	34.61	**35.90**	**37.78**	21.18	**32.04**

fine-tuned to improve predicting visual labels. Our main contribution is to introduce an effective supervised learning formulation to enable the discriminative power of a VAE, while maintaining its generative property. The experimental results we have obtained are state-of-the-art over two benchmark datasets: MS COCO and Flickr30K, under two different evaluation metrics. Two promising directions for future work are to include attention mechanisms to our model to help capture salient patches in the image, and to integrate techniques in natural language processing to better address the linguistic issues in human-centric annotations.

Acknowledgement. We thank the reviewers for their valuable comments. This work was supported in part by MOST grants 102-2221-E-001-021-MY3, 105-2221-E-001-027-MY2 and an NSF grant 1422021.

References

1. Karpathy, A., Fei-Fei, L.: Deep visual-semantic alignments for generating image descriptions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2015) 3128–3137
2. Donahue, J., Anne Hendricks, L., Guadarrama, S., Rohrbach, M., Venugopalan, S., Saenko, K., Darrell, T.: Long-term recurrent convolutional networks for visual recognition and description. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2015) 2625–2634
3. Xu, K., Ba, J., Kiros, R., Courville, A., Salakhutdinov, R., Zemel, R., Bengio, Y.: Show, attend and tell: Neural image caption generation with visual attention. arXiv preprint arXiv:1502.03044 (2015)
4. Wu, Q., Shen, C., Liu, L., Dick, A., van den Hengel, A.: What value do explicit high level concepts have in vision to language problems? In: Proc. IEEE Conf. Comp. Vis. Patt. Recogn. Volume 2. (2016) 4
5. Yao, L., Torabi, A., Cho, K., Ballas, N., Pal, C., Larochelle, H., Courville, A.: Describing videos by exploiting temporal structure. In: Proceedings of the IEEE International Conference on Computer Vision. (2015) 4507–4515
6. Venugopalan, S., Rohrbach, M., Donahue, J., Mooney, R., Darrell, T., Saenko, K.: Sequence to sequence-video to text. In: Proceedings of the IEEE International Conference on Computer Vision. (2015) 4534–4542
7. Antol, S., Agrawal, A., Lu, J., Mitchell, M., Batra, D., Zitnick, C.L., Parikh, D.: Vqa: Visual question answering. In: International Conference on Computer Vision (ICCV). (2015)
8. Mansimov, E., Parisotto, E., Ba, J., Salakhutdinov, R.: Generating images from captions with attention. In: ICLR. (2016)
9. Berg, A.C., Berg, T.L., Daume III, H., Dodge, J., Goyal, A., Han, X., Mensch, A., Mitchell, M., Sood, A., Stratos, K., et al.: Understanding and predicting importance in images. In: Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, IEEE (2012) 3562–3569
10. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft COCO: Common objects in context. In: Computer Vision–ECCV 2014. Springer (2014) 740–755
11. Plummer, B.A., Wang, L., Cervantes, C.M., Caicedo, J.C., Hockenmaier, J., Lazebnik, S.: Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In: Proceedings of the IEEE International Conference on Computer Vision. (2015) 2641–2649
12. Misra, I., Zitnick, C.L., Mitchell, M., Girshick, R.: Seeing through the Human Reporting Bias: Visual Classifiers from Noisy Human-Centric Labels. In: CVPR. (2016)
13. Fang, H., Gupta, S., Iandola, F., Srivastava, R.K., Deng, L., Dollár, P., Gao, J., He, X., Mitchell, M., Platt, J.C., et al.: From captions to visual concepts and back. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2015) 1473–1482
14. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. In: Proceedings of the 2nd International Conference on Learning Representations (ICLR). Number 2014 (2013)
15. Devlin, J., Gupta, S., Girshick, R., Mitchell, M., Zitnick, C.L.: Exploring nearest neighbor approaches for image captioning. arXiv preprint arXiv:1505.04467 (2015)

16. Klein, B., Lev, G., Sadeh, G., Wolf, L.: Associating neural word embeddings with deep image representations using fisher vectors. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2015) 4437–4446
17. Jia, X., Gavves, E., Fernando, B., Tuytelaars, T.: Guiding long-short term memory for image caption generation. arXiv preprint arXiv:1509.04942 (2015)
18. Rohrbach, M., Qiu, W., Titov, I., Thater, S., Pinkal, M., Schiele, B.: Translating video content to natural language descriptions. In: Proceedings of the IEEE International Conference on Computer Vision. (2013) 433–440
19. Thomason, J., Venugopalan, S., Guadarrama, S., Saenko, K., Mooney, R.J.: Integrating language and vision to generate natural language descriptions of videos in the wild. In: COLING. (2014) 9
20. Turakhia, N., Parikh, D.: Attribute dominance: What pops out? In: Proceedings of the IEEE International Conference on Computer Vision. (2013) 1225–1232
21. Yun, K., Peng, Y., Samaras, D., Zelinsky, G., Berg, T.: Studying relationships between human gaze, description, and computer vision. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. (2013) 739–746
22. Chen, X., Gupta, A.: Webly supervised learning of convolutional networks. In: Proceedings of the IEEE International Conference on Computer Vision. (2015) 1431–1439
23. Izadinia, H., Russell, B.C., Farhadi, A., Hoffman, M.D., Hertzmann, A.: Deep classifiers from image tags in the wild. In: Proceedings of the 2015 Workshop on Community-Organized Multimodal Mining: Opportunities for Novel Solutions, ACM (2015) 13–18
24. Joulin, A., van der Maaten, L., Jabri, A., Vasilache, N.: Learning visual features from large weakly supervised data. arXiv preprint arXiv:1511.02251 (2015)
25. Kingma, D.P., Mohamed, S., Rezende, D.J., Welling, M.: Semi-supervised learning with deep generative models. In: Advances in Neural Information Processing Systems. (2014) 3581–3589
26. Burda, Y., Grosse, R., Salakhutdinov, R.: Importance weighted autoencoders. arXiv preprint arXiv:1509.00519 (2015)
27. Fabius, O., van Amersfoort, J.R.: Variational recurrent auto-encoders. arXiv preprint arXiv:1412.6581 (2014)
28. Gregor, K., Danihelka, I., Graves, A., Wierstra, D.: Draw: A recurrent neural network for image generation. arXiv preprint arXiv:1502.04623 (2015)
29. Chung, J., Kastner, K., Dinh, L., Goel, K., Courville, A.C., Bengio, Y.: A recurrent latent variable model for sequential data. In: Advances in neural information processing systems. (2015) 2962–2970
30. Kulkarni, T.D., Whitney, W.F., Kohli, P., Tenenbaum, J.: Deep convolutional inverse graphics network. In Cortes, C., Lawrence, N.D., Lee, D.D., Sugiyama, M., Garnett, R., eds.: Advances in Neural Information Processing Systems 28. Curran Associates, Inc. (2015) 2539–2547
31. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: International Conference on Learning Representations. (2015)
32. Xie, S., Tu, Z.: Holistically-nested edge detection. In: Proceedings of the IEEE International Conference on Computer Vision. (2015) 1395–1403
33. Nair, V., Hinton, G.E.: Rectified linear units improve restricted boltzmann machines. In: Proceedings of the 27th International Conference on Machine Learning (ICML-10). (2010) 807–814
34. Chen, X., Fang, H., Lin, T.Y., Vedantam, R., Gupta, S., Dollár, P., Zitnick, C.L.: Microsoft COCO captions: Data collection and evaluation server. arXiv preprint arXiv:1504.00325 (2015)
35. Zhang, C., Platt, J.C., Viola, P.A.: Multiple instance boosting for object detection. In: Advances in neural information processing systems. (2005) 1417–1424