

Common Subspace for Model and Similarity: Phrase Learning for Caption Generation from Images

Yoshitaka Ushiku, Masataka Yamaguchi, Yusuke Mukuta, Tatsuya Harada
 The University of Tokyo
 7-3-1 Hongo Bunkyo-ku, Tokyo, Japan
 {ushiku, yamaguchi, mukuta, harada}@mi.t.u-tokyo.ac.jp

Abstract

Generating captions to describe images is a fundamental problem that combines computer vision and natural language processing. Recent works focus on descriptive phrases, such as “a white dog” to explain the visual composites of an input image. The phrases can not only express objects, attributes, events, and their relations but can also reduce visual complexity. A caption for an input image can be generated by connecting estimated phrases using a grammar model. However, because phrases are combinations of various words, the number of phrases is much larger than the number of single words. Consequently, the accuracy of phrase estimation suffers from too few training samples per phrase.

In this paper, we propose a novel phrase-learning method: Common Subspace for Model and Similarity (CoSMoS). In order to overcome the shortage of training samples, CoSMoS obtains a subspace in which (a) all feature vectors associated with the same phrase are mapped as mutually close, (b) classifiers for each phrase are learned, and (c) training samples are shared among co-occurring phrases. Experimental results demonstrate that our system is more accurate than those in earlier work and that the accuracy increases when the dataset from the web increases.

1. Introduction

Object, event, and attribute recognition from images have been widely investigated. Recently, several works have tackled the sentential description of images to more flexibly explain the contents of images.

In general, collecting a large amount of data from the web is a common means to understand various images. What we can collect automatically are images associated not with semantically clear labels but with surrounding sentences. Hence, the requirements for caption generation from images are: scalability, learning image contents, and caption generation using estimated content.



BabyTalk: This is a picture of three persons, one bottle and one diningtable. The first rusty person is beside the second person. The rusty bottle is near the first rusty person, and within the colorful diningtable. The second person is by the third rusty person. The colorful diningtable is near the first rusty person, and near

the second person, and near the third rusty person.

Corpus-Guided: Three people are showing the bottle on the street.

Midge: People with a bottle at the table.

Ours: Group of people sitting at a table with a dinner.

Figure 1. Qualitative comparison. A common input image is shown in the upper left. We compare our result with Corpus-Guided [48], Midge [28], and BabyTalk [18].

In order to represent image contents, such as objects, events, attributes, and their relations, recent works [7, 10, 20, 22, 28, 35, 41, 43] focus on visual phrases describing image contents and their relations. For example, in order to learn a general class of “dog,” dogs in the following phrases should be considered in the same class: “white dog”, “black dog”, “running dog”, and “sleeping dog.” The semantic gap between image content can be narrowed by learning each phrase independently, not just the single word “dog.” A caption for an input image can then be generated by connecting estimated phrases using a grammar model.

Because phrases are combinations of objects, attributes, and events, a large number of phrases should be learned and recognized. Therefore, the number of training samples per phrase is much less than that for the usual object recognition. Recent large-scale visual classification is tackled using a combination of high-dimensional image features and linear weight vector as classifiers [37] or using a deep convolutional neural network [17]. To adopt these methods to learn phrases, however, learning many parameters using too few training samples per phrase would result in over-fitting.

In order to overcome the shortage of training samples, usage of a subspace is a reasonable way to approximate classifiers for phrases. Traditional multivariable methods, such as linear discriminant analysis, can absorb the shortage of training samples by reducing the dimension of fea-

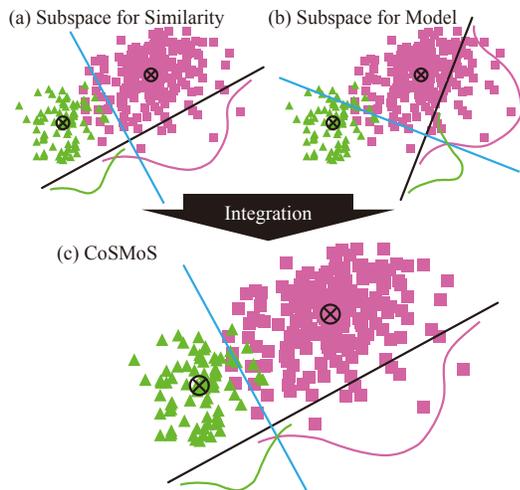


Figure 2. Simple overview of subspace learning. We would like to obtain a one-dimensional subspace (black line) given training samples in a two-dimensional feature space. The blue line orthogonal to each subspace is the decision plane between the green triangle class and the purple rectangle class. The two crossed circles are the mean of each class.

ture vectors. All feature vectors associated with the same phrase are mapped as mutually close in the subspace. Recent works [26,27,42] also propose online learning methods to obtain such subspace. In the subspace obtained via these methods, the Nearest Class Mean (NCM) classifier is employed as a scalable linear classifier. However, the NCM classifier is suboptimal if the distribution for each label is biased, as depicted in Fig. 2 (a). In this paper, we refer to this variation of the subspace learning method as the **similarity**-based method.

Another way to utilize a subspace is to learn linear weight vectors as classifiers in the obtained subspace as proposed in [47]. We refer to this variation of subspace learning as the **model**-based method. However, because there are no constraints for the subspace itself, it is not guaranteed that an obtained subspace is appropriate for classification as depicted in Fig. 2 (b).

Therefore, we propose an integrated form of the subspace learning methods, as shown in Fig. 2 (c). Additionally, semi-supervised learning includes the problem of learning categories using a small number of training samples. To avoid over-fitting, some works [25,36] share training samples among semantically similar classes. We also introduce another function via multimodal NCM in Sec. 3.

To summarize, we propose a novel subspace-embedding method—Common Subspace for Model and Similarity (CoSMoS) for caption generation from images. CoSMoS can obtain a subspace in which (a) all feature vectors associated with the same phrase are mapped as mutually close, (b) classifiers for each phrase are learned, and (c) training samples are shared among co-occurring phrases. Our main

contributions are summarized as follows:

- Proposal of CoSMoS, which reduces the model complexity by integrating both **similarity** and **model** to learn phrases using relatively fewer training samples per phrase. We also provide source codes¹.
- Thorough experiments for caption generation from images. This includes evaluations of two approaches for caption generation: sentence template and combinatorial optimization. The use of an increasing number of web images is also investigated.

The remainder of this paper is organized as follows: Sec. 2 introduces related work for caption generation and for subspace learning. Details of CoSMoS are given in Sec. 3. In Sec. 4, we provide the rest of the pipeline of caption generation. Experimental results are shown in Sec. 5, and we conclude this paper in Sec. 6.

2. Related work

In this paper we focus on subspace learning to overcome the shortage of training samples for each phrase to generate captions. This section introduces related works for caption generation and subspace learning.

Reuse of the entire caption. Natural language generation itself is a challenging task. Therefore, some methods reuse the entire caption associated with the training image to describe the input image. In [6], all images are labeled with a triplet: (object, action, scene). In another method [29], images are labeled according to their objects, stuff, people, and scenes from different datasets. The images with similar labels estimated from an input image are retrieved. In [29], matching local descriptors is also used to search for similar images. [12] adopts Kernel Canonical Correlation Analysis to associate existing captions to input images. However, using a whole caption directly requires a very large number of images related to all combinations of image contents. Moreover, similar images must be retrieved accurately and quickly from such a huge dataset.

Template-based caption generation. Some works generate a new caption using one or more templates. In [18], images are explained sentimentally with respect to the objects' names, number, and their spatial relations by learning objects, stuff, and attributes from different datasets. [48] extended the concepts in [6] to generate a new caption. However, as the authors of [22,28] indicate, the use of a template to generate general captions is suboptimal because templates cannot enable syntactic variability. In [20], integer linear programming is introduced to generate a caption as a combinatorial optimization. However, the contents to be described in a caption are manually defined.

¹http://www.mi.t.u-tokyo.ac.jp/static/projects/mil_cosmos/

Phrase-based caption generation. In [7,35], visual detection with “Visual Phrases” is presented. For example, detecting “person_riding_horse” is performed by decoding the results of object detection for each object, such as person and horse. By connecting these phrases using a grammar model, captions can be generated. Such phrase-based caption generation is common in statistical machine translation. Various works [10, 22, 28, 41, 43] employ such phrases to generate captions from images.

Some works [10, 43] adopt Large Margin Nearest Neighbor (LMNN) classification [45], which is not scalable for the data amount, and templates for caption generation. In [22, 28, 41], phrases are estimated using the object recognition method and connected using the n-gram model. Although new captions can be generated, [22, 28] require an extra dataset for object detection. The closest to this paper is [41] where phrases are learned with **model**-based linear classifiers and combined using a variation of multi-stack beam search with the n-gram model. Although, captions can be generated using only images and corresponding captions, as described in Sec. 1, few training samples per phrase would increase over-fitting for phrase prediction.

Caption generation using neural networks. More recently, [2, 4, 5, 14, 15, 44] introduce a combination of a deep Convolutional Neural Network (CNN) and a Recurrent Neural Network (RNN). CNN is known as a state-of-the-art method to learn image features for object recognition. RNN and Long-Short Term Memory (LSTM) net [11], a kind of RNN, can predict a word given an image feature and currently generated part of a caption for an input image. In general, however, neural networks require a sufficient number of training samples for stable learning. Although our pipeline is simpler, parameter reduction by a subspace does help phrase learning. We experimentally show the competitive performance in Sec. 5.

Subspace learning for linear classification. To generate an accurate caption, phrase estimation for an input image is a crucial step. Hence, subspace learning is necessary to overcome the shortage of training samples per phrase.

Traditional multivariable methods, such as Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), and Canonical Correlation Analysis (CCA), can be considered as methods to satisfy requirement (a) in Sec. 1. In the subspace obtained via these methods, the Nearest Class Mean (NCM) classifier is employed as a scalable linear classifier. These subspace learning methods can be considered as the **similarity**-based methods because the methods can obtain appropriate similarity between an input feature and each NCM. Various methods for nonlinear metric learning such as OASIS [1] and LMNN [45] can also be considered as a **similarity**-based method among train-

ing samples rather than NCM. However, we focus on linear classifiers for scalability.

Recent works [26, 27, 42] propose online learning methods to obtain a subspace with an NCM classifier. Although an NCM classifier is a kind of linear classifier, there is no guarantee that the decision planes according to the NCM classifier are optimal, as illustrated in Fig. 2 (a) in Sec. 1.

WSABIE [47] obtains a subspace and linear weight vectors in that subspace. This approach can be considered as an approximation of linear classifiers (**models**) in the feature space. Such linear weight vectors are also employed in combination with high-dimensional image feature [37], and in the full connected layers in CNN [17]. We refer to this variation of subspace learning as **model**-based method. Although requirement (b) is satisfied, an obtained subspace would be inappropriate because there are no constraints, such as requirement (a) as described in Sec. 1.

Therefore, we propose an integrated form of subspace learning methods as shown in Fig. 2 (c). To the best of our knowledge, this is the first method integrating **similarity** and **model** for a linear classifier.

3. Common Subspace for Model and Similarity

This section describes the proposed method, Commons for Similarity and Model (CoSMoS), to train the relations between an image and extracted phrases. We define the classification rule as a Multimodal NCM classifier.

3.1. Multimodal Nearest Class Mean classifier

We define the number of pairs of images and captions as N . Given the i -th image and a set of captions C_i , we define image feature $\mathbf{x}_i \in \mathbb{R}^d$ and extract phrases $P_i \subset \mathcal{P}$. The detail to extract phrases P_i is described in Sec. 4. We also define the phrase feature vector $\mathbf{y} \in \mathbb{R}^{n_p}$ as a Bag-of-Phrases vector, the j -th element of which is one if the image is associated with p_j , and zero otherwise.

3.1.1 Classification based on similarity and model

In order to learn phrases using large-scale datasets, the linear classifier is preferable in terms of scalability. First, we consider the well-known NCM classifier in a subspace to realize the first requirement: (a) all feature vectors associated with the same phrase are mapped as mutually close. We introduce a **similarity**-based classification,

$$\hat{p} = \arg \max_p \theta_s^p \equiv \arg \max_p \tilde{\mathbf{x}}^{p\top} S^\top S \mathbf{x}_i - b_s^p, \quad (1)$$

using score θ_s^p , Class Mean $\tilde{\mathbf{x}}^p$, and bias b_s^p for phrase p . S is a $r \times d$ matrix to map \mathbf{x} into an r -dimensional subspace.

In order to realize the second requirement, (b) classifiers for each phrase are learned, we introduce a linear weight

vector $\mathbf{m}^p \in \mathbb{R}^r$ as a **model** for each phrase p . The **model**-based classification rule is defined as,

$$\hat{p} = \arg \max_p \theta_m^p \equiv \arg \max_p \mathbf{m}^{p\top} S \mathbf{x}_i - b_m^p, \quad (2)$$

where θ_m^p and b_m^p are, respectively, a score and a bias for phrase p . Thus, integration of **similarity** and **model** for classification becomes,

$$\begin{aligned} \hat{p} &= \arg \max_p \theta_s^p + \alpha \theta_m^p \\ &\equiv \arg \max_p \tilde{\mathbf{x}}^{p\top} S^\top S \mathbf{x}_i + \alpha \mathbf{m}^{p\top} S \mathbf{x}_i - b^p, \end{aligned} \quad (3)$$

where α is a parameter balancing between the **similarity**-based score θ_s^p and the **model**-based score θ_m^p . b^p is an integrated bias defined as $b_s^p + \alpha b_m^p$. Note that we can replace $\alpha \mathbf{m}^p$ with \mathbf{m}^p because α can be implicitly learned by \mathbf{m}^p .

3.1.2 Sharing training samples among related phrases

In this subsection we (c) share training samples among co-occurring phrases. For example, if a phrase “white dog” often co-occurs with another phrase “dog running”, we can combine the **model** for “white dog” $\mathbf{m}^{\text{white dog}} S \mathbf{x}_i - b^{\text{white dog}}$ and the **model** for “dog running” $\mathbf{m}^{\text{dog running}} S \mathbf{x}_i - b^{\text{dog running}}$ during both training and estimating. Therefore, we consider the co-occurrence among phrases using another Class Mean $\tilde{\mathbf{y}}^p$ and the averaged pairwise loss to share training samples.

In order to recognize “white dog”, we can use linear weight vectors for frequently corresponding phrases such as “dog running.” Hence, in order to utilize co-occurrence among phrases, we calculate Class Mean $\tilde{\mathbf{y}}^p$, the average of Bag-of-Phrases vectors that are associated with a phrase p . Then the element corresponding to “white dog” of $\tilde{\mathbf{y}}^{\text{dog running}}$ becomes close to one. Therefore, the **model** \mathbf{m}^p for phrase p becomes $M \tilde{\mathbf{y}}^p$, where M is an $r \times n_p$ matrix, the j -th column vector of which is a linear weight vector for the j -th phrase.

Thus, the classification rule for CoSMoS is defined as a multimodal NCM classification,

$$\hat{p} = \arg \max_p \tilde{\mathbf{x}}^{p\top} S^\top S \mathbf{x}_i + \tilde{\mathbf{y}}^{p\top} M^\top S \mathbf{x}_i - b^p. \quad (4)$$

By introducing a matrix $U \equiv (S \ M)$ and a classification score θ^p for phrase p , this rule becomes,

$$\hat{p} = \arg \max_p \theta^p \equiv \arg \max_p \left(\tilde{\mathbf{x}}^{p\top} \right) U^\top U \begin{pmatrix} \mathbf{x}_i \\ \mathbf{0} \end{pmatrix} - b^p. \quad (5)$$

Additional use of the co-occurrence among phrases is a loss function with multiple pairs of positive and negative phrases, as referred to in [41, 47].

$$\ell_i \equiv 1 - \frac{1}{|S_i|} \sum_{p \in S_i} \theta^p + \frac{1}{|S'_i|} \sum_{p \in S'_i} \theta^p. \quad (6)$$

This loss function can be considered as an averaged hinge loss. We can also employ a rank loss as presented in [47]. However, we use this simple loss function because we cannot obtain significant improvements with rank loss. The members of $S_i \subset P_i$ and $S'_i \subset \mathcal{P}/P_i$ are chosen by selecting a pair of phrases that has a gap of scores is larger than one. Particularly, we design the following procedure:

1. Pick up $p \in P_i$ with the minimum score θ^p .
2. Pick up $p' \notin P_i$ with the maximum score $\theta^{p'}$.
3. If $\theta^p < \theta^{p'} + 1$, add p and p' to S_i and S'_i respectively. Then, (i) remove p and p' respectively from P_i and Y_i , and (ii) go back to the first step if $P_i \neq \emptyset$.

For a simple formulation, we use $\mathbf{g}_i \in \mathbb{R}^{n_p}$ and define the j -th element $g_{i,j} = 1/|S_i|$ if $p_j \in S_i$, $-1/|S'_i|$ if $p_j \in S'_i$, and zero otherwise. Now Eq. (6) can be rewritten as,

$$\ell_i = 1 - \sum_{j=1}^{n_p} g_{i,j} \theta^{p_j} = 1 - \mathbf{g}_i^\top \left(\begin{pmatrix} \tilde{X} \\ \tilde{Y} \end{pmatrix}^\top U^\top U \begin{pmatrix} \mathbf{x} \\ \mathbf{0} \end{pmatrix} - \mathbf{b} \right), \quad (7)$$

where \tilde{X} and \tilde{Y} are matrices consisting of Class Means, the j -th column vector of which corresponds to the Class Mean $\tilde{\mathbf{x}}^{p_j}$ and $\tilde{\mathbf{y}}^{p_j}$ for the phrase p_j . $\mathbf{b} \in \mathbb{R}^{|\mathcal{P}|}$ is a vector of bias, the j -th element of which is the bias of the phrase p_j .

3.2. Learning algorithm using the averaged stochastic gradient descent

The objective function \mathcal{L} is defined as the following cumulative loss for all N training samples:

$$\mathcal{L} = \sum_{i=1}^N \ell_i = \sum_{i=1}^N \left(1 - \mathbf{g}_i^\top \left(\begin{pmatrix} \tilde{X} \\ \tilde{Y} \end{pmatrix}^\top U^\top U \begin{pmatrix} \mathbf{x}_i \\ \mathbf{0} \end{pmatrix} - \mathbf{b} \right) \right). \quad (8)$$

In order to minimize an objective function, we adopt the averaged Stochastic Gradient Descent (SGD) for faster convergence [32]. Given the t -th training sample, SGD, an on-line learning scheme, investigates if the current parameters U_t and \mathbf{b}_t can recognize phrases and update these parameters to U_{t+1} and \mathbf{b}_{t+1} .

As described in [46, 47], U is initialized randomly with mean 0 and standard deviation $1/\sqrt{d+n_p}$. Update rules for U and \mathbf{b} using SGD with learning rate η are,

$$U_{t+1} = U_t + \eta U_t \left(\begin{pmatrix} \mathbf{x}_t \\ \mathbf{0} \end{pmatrix} \mathbf{g}_t^\top \begin{pmatrix} \tilde{X} \\ \tilde{Y} \end{pmatrix}^\top + \begin{pmatrix} \tilde{X} \\ \tilde{Y} \end{pmatrix} \mathbf{g}_t \begin{pmatrix} \mathbf{x}_t \\ \mathbf{0} \end{pmatrix}^\top \right), \quad (9)$$

$$\mathbf{b}_{t+1} = \mathbf{b}_t - \eta \mathbf{g}_t. \quad (10)$$

Given U_T and \mathbf{b}_T after T -times training, the averaged SGD uses the following averaged parameters \tilde{U} and $\tilde{\mathbf{b}}$ instead of U_T and \mathbf{b}_T to estimate phrases of a test sample:

$$\tilde{U} = \frac{1}{T} \sum_{t=1}^T U_t, \quad \tilde{\mathbf{b}} = \frac{1}{T} \sum_{t=1}^T \mathbf{b}_t. \quad (11)$$

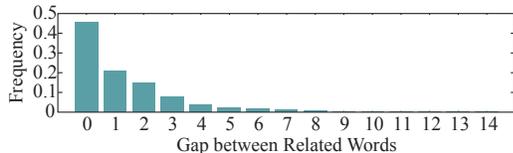


Figure 3. Frequencies of the word gaps between related words.

4. Phrase-based caption generation

Phrase-based caption generation is divided into three steps: phrase extraction from image descriptions, learning phrases using images and extracted phrases, and caption generation from estimated phrases. This section discusses the first and third steps. The second step is achieved by CoSMoS, as described in Sec. 3.

4.1. Phrase extraction from image descriptions

Given the t -th training image and ground truth captions, we would like to extract a set of phrases $P_t \subset \mathcal{P}$, where \mathcal{P} is a set of all phrases and n_p is the number of all phrases.

Some researchers [10, 28, 43] use a parser to extract phrases from the ground truth. However, this method suffers from the parse error.

In another work [41], the authors use continuous words as phrases. To justify the use of continuous words as phrases, we investigate whether most relations in the captions are included in two continuous words. We parse all captions in PASCAL Sentence using Stanford Parser [16]. The frequencies of the word gaps between grammatically related words are shown in Fig. 3. About half of the relations between two words are extracted from two continuous words. The frequent grammatical relations are shown in the Supplemental Materials. We also find that the relations extracted from two discontinuous words include many “prep_*” relations. For example, relation “prep_in” is found from “airplane in flight.” Although “airplane” and “flight” are distant from each other, these three words can be restored by estimating two phrases, “airplane in” and “in flight.” Consequently, “prep_*” relations can also be represented by two continuous words.

In [41], phrases are filtered based on frequencies only. The objective is to eliminate meaningless phrases, such as “is-a”. However, this is not because “is-a” is overly frequent but because “is-a” consists of an auxiliary verb and an article. Therefore, meaningless phrases are apparently found by considering whether each word in the phrase is meaningless. Although new words are made up every year, the meaningless words determined once will be so for a long time. In the literature on Information Retrieval, such meaningless words are called stop words.

Therefore, we present another filter, Stop Word Filter (SWF), based on the rate of stop words. In particular, phrases including not more than one stop word are extracted. In Sec. 5, we evaluate the effect of this filter.

4.2. Caption generation from estimated phrases

Generating captions should (1) use estimated phrases, (2) be grammatically correct, and (3) keep the target length.

We have two options: usage of template [48] and combinatorial optimization [20, 28, 41]. Because usage of template is suboptimal, we would like to adopt a combinatorial optimization. In particular, we adopt multi-stack beam search [41] rather than complicated integer linear programming [20] and tree-generating process [28]. Beam search is a well-known algorithm for statistical machine translation to generate a sentence by minimizing the sum of costs. In this study, we define phrase cost $\phi_p(w_{i-1}, w_i)$ and length cost $\phi_l(l)$. The optimization problem is,

$$\{w_1, \dots, w_l\} = \arg \min_{w_1, \dots, w_l} \phi_l(l) + \lambda_p \sum_i \phi_p(w_{i-1}, w_i), \quad (12)$$

where λ_p is weight parameter for phrase cost. If $\{w_{i-1}, w_i\}$ is one of the estimated phrases, $\phi_p(w_{i-1}, w_i) = 0$. Otherwise, the cost is calculated using the negative log of bigram (and trigram if possible). For the third requirement, a length cost with the target length l (ten words in this paper) is defined as $\phi_l(l) = -\log \mathcal{N}(l, \sigma_0)$. Because it is unclear if the accuracy of captions with a combinatorial optimization method is better than the methods with templates [9, 10, 43], we experimentally compare them in Sec. 5.1 later.

5. Experimental results

We evaluated our method using four datasets: PASCAL Sentence [33], Microsoft COCO released in 2014 (MS COCO) [23], IAPR-TC12 [8], and SBU [29]. PASCAL Sentence and MS COCO respectively consists of 1000 pairs and 164,062 pairs of images and five captions. IAPR-TC12 consists of 19,963 pairs of images and around 1.8 captions for each image. These datasets are organized manually. SBU consists of 1M pairs of images and captions collected from Flickr. *Note that we evaluate not only the accuracy of the generated captions but also the classification performance of CoSMoS. The evaluation of CoSMoS is reported in the Supplemental Materials due to limitations of space.*

For image feature, we used the Fisher Vector (FV) [31] with SIFT [24] and CNN [17, 39] pretrained with 1.2M images from ILSVRC [34] using Caffe [13]. We extracted dense SIFTs with five scales and reduced their dimensions to 64 using PCA. For PASCAL Sentence and IAPR-TC12, we obtained a Gaussian Mixture Model (GMM) with 256 components. The FV was then calculated over 1×1 , 2×2 , and 3×1 cells. For SBU, the FV was calculated over the whole image using a GMM with 16 components to make all FVs fit for the memory space by reducing the dimension. For the CNN feature, we experimentally found that the output of the seventh layer was optimal.

Phrases consisting of two continuous words were extracted using SWF. For PASCAL Sentence, phrases associated with 10 or more images were extracted. We extracted phrases occurring more than the same ratio from IAPR-TC12 and SBU. For phrase learning, we trained CoSMoS with a 128-dimensional subspace in 10 iterations. The best learning rate η was selected from $\{2^{-3}, 2^{-4}, 2^{-5}, 2^{-6}\}$ according to the evaluation score with each test set.

We followed the same experimental setup used in most of the previous work in this area. PASCAL Sentence and IAPR-TC12 were divided into 90% training samples and 10% testing samples. For SBU, 500 testing images were extracted randomly. For each dataset, we repeated the division five times. MS COCO already has training set, validation set, and testing set.

For automatic evaluation of generated captions, we employed BLEU [30] and NIST [3]. “BLEU x” is the cumulative product of the n-gram match rate from unigram to x-gram. “NIST x” is the cumulative sum of the n-gram match rate from unigram to x-gram. Both BLEU and NIST have length penalties to allow for fair evaluation of all captions, including overly short captions. The ceiling on BLEU is one because this score is a variation of the match rate. Because NIST weighs rear expressions, the ceiling is unclear.

5.1. Discussion of the phrase approach

This subsection justifies the use of a phrase-based approach comparing it to a template-based approach. Whereas most work attempts to generate captions from input images, [9] presents a slightly different problem: generating a caption from an already annotated input image. Given labels corresponding to objects and attributes, a proper caption is generated using multiple templates. In [9], captions are generated from pairs of images and words extracted from the ground truth captions. In order to investigate whether our phrase-based approach is preferable to a template-based approach [9], we used “oracle” phrases. In the usual problem setting of caption generation for images, a caption is generated from the input image on the left side. Here, we evaluated our caption generation system by generating a caption from correct phrases existing in the ground truth.

Table 1 presents the results of caption generation from oracle phrases. This table shows that our captions generated from oracle phrases and reference captions are more similar than captions generated using templates.

5.2. Comparison using PASCAL Sentence

In this section, we show the results using the PASCAL Sentence dataset. Typical examples of generated captions are presented in Fig. 4. As shown, estimated phrases contribute to the generation of appropriate captions. However, as the bottom example in Fig. 4 illustrates, even if a few phrases (“decker bus” and “a bus”) are incorrect, our

Table 1. Comparison of output captions using oracle phrases (“ours”) versus a template-based approach [9].

Dataset	Method	BLEU				NIST
		1	2	3	4	5
PASCAL Sentence	template	0.74	0.55	0.35	-	-
	ours	0.82	0.71	0.56	0.42	7.64
IAPR-TC12	template	0.33	0.18	0.07	-	-
	ours	0.74	0.61	0.48	0.37	6.26

Input Image	Estimated Keyphrases	Generated Sentence & Ground Truth
	a group group of of people people sitting a living	A living room with a group of people sitting. Four Asian young people sitting in a den of living room.
	airplane is a grassy grass in a field grassy field	A grassy field in front of a body of water. A beautiful lake surrounded by trees with two small boats on the beach.
	a group of people group of people sitting the table	Group of people sitting at a table with a dinner. three people sitting at a table with food and wine
	a desk desk with computer and a computer table with	Table with a desk with a computer and a chair. An office cube has a desktop computer, a cluttered desk, and a blue office chair.
	room with decker bus a room desk with a bus	A living room with a view of a television. The entertainment center of the living room includes a TV, plants, and several baskets.

Figure 4. Representative examples of estimated phrases and generated captions for PASCAL Sentence. The first column shows input images. The second column shows estimated phrases for each input image. The third column shows the generated caption at the top and the ground truth in the dataset at the bottom. Red-colored words in the generated captions derive from estimated phrases.

method using modified multi-stack beam search automatically selects appropriate phrases to the greatest extent possible. As a result, such incorrect phrases are ignored and an accurate caption (“A living room with a view of a television.”) is generated. Qualitative comparison to previous work was already presented in Fig. 1. Table 2 shows a quantitative comparison. Scores in parentheses were computed by matching synonyms. In general, matching not only the same word but also its synonyms increases the score. The table shows that our framework can generate more accurate captions. We also investigated a naive filter depending on frequency only, as in [41]. As shown in Table 2, CoSMoS leads to better scores than the linear model in [41]. Additionally, usage of SWF contributes to BLEU 1/2 and NIST 5, which is to be expected because eliminating meaningless phrases contributes directly to the matching rate of unigrams and bigrams. Therefore, we used SWF for phrase extraction in the experiments described next.

Table 2. Evaluation of output captions from PASCAL Sentence. Scores in parentheses were computed by matching synonyms.

Method	BLEU				NIST
	1	2	3	4	5
Baby talk [18]	0.25 (0.30)	-	-	-	-
Corpus-guided [48]	(0.41)	(0.13)	(0.03)	-	-
Verma et al. [43]	0.36 (0.43)	-	-	-	-
Gupta et al. [10]	(0.54)	(0.23)	(0.07)	-	-
FV + Linear model +naive filter [41]	-	-	-	0.07	2.65
FV + CoSMoS +naive filter	0.53	0.32	0.19	0.11	3.37
FV+CoSMoS+SWF	0.56	0.33	0.19	0.11	3.45

Table 3. Evaluation of output captions from MS COCO.

Method	BLEU				METEOR
	1	2	3	4	-
Multimodal RNN [14]	0.63	0.45	0.32	0.23	0.20
Mind’s Eye [2]	-	-	-	0.22	0.25
LRCN [4]	0.67	0.49	0.35	0.25	-
[5]	0.70	-	-	0.29	0.25
VGG net + CoSMoS	0.65	0.49	0.32	0.20	0.20

Table 4. Evaluation of output captions from PASCAL Sentence. We used MS COCO as a training dataset, as described in [44].

Method	BLEU				NIST
	1	2	3	4	5
Google NIC [44]	0.59	-	-	-	-
AlexNet+CoSMoS	0.62	0.41	0.25	0.15	4.43

5.3. Comparison to neural networks + MS COCO

After the first submission of this paper, various works based on neural networks such as CNN and RNN are evaluated using larger dataset than PASCAL Sentence. This would be mainly because sufficient number of training samples are required to train neural networks. Since MS COCO, one of the largest manually-organized datasets, is common to RNN-based works for evaluation, we also evaluated our method using this dataset. In this subsection, we employed two kinds of CNN features: AlexNet [17] and VGG net [39]. Because the captions for the test set are not available, we report the BLEU and METEOR [21] scores computed with the coco-caption code².

As shown in Table 3, we achieved performance competitive with RNN-based approaches. Multimodal RNN [14] is a combination of AlexNet and Bidirectional RNN [38]. LRCN [4] integrates VGG net and LSTM net [11], trained for feature learning and caption generation, respectively.

²<http://github.com/tylin/coco-caption>

Table 5. Evaluation of output captions from IAPR-TC12. Scores in parentheses were computed by matching synonyms.

Method	BLEU				NIST
	1	2	3	4	5
Gupta et al. [10]	0.15 (0.21)	0.06 (0.07)	0.01 (0.01)	-	-
FV+CoSMoS	0.60	0.40	0.28	0.20	3.73

Table 6. Evaluation of output captions from the SBU dataset. Scores in parentheses were computed by matching synonyms.

Method	BLEU				NIST
	1	2	3	4	5
Im2text [29]	0.13	-	-	-	-
Kuznetsova et al. [20]	0.11	(0.11)	-	-	-
FV+CoSMoS	0.20	0.09	0.04	0.02	1.15

Google NIC [44] combines LSTM net and GoogLeNet [40], a deeper CNN than VGG net. LSTM net can accurately generate captions by learning the probability of skip-gram, which is a kind of n-gram with word gaps. RNN-based methods were slightly better than our results in terms of BLEU-4 because of the ability of learning skip-gram. Therefore, the competitive performance reported in this section in terms of BLEU-1/2 would be attributable mainly to CoSMoS, which can learn the classifiers for phrases with few training samples. Obviously, we can introduce CoSMoS between CNN and LSTM because CoSMoS can propagate the gradient of loss backward, although a proposal of that combination is beyond the scope of this paper.

For evaluation of generality, we evaluated captions generated from PASCAL Sentence by training MS COCO, as reported in [44]. First, images and captions in MS COCO were trained. All images in PASCAL Sentence dataset were then used as testing samples. Table 4 shows that we slightly outperform the RNN-based approach. For more evaluations about generality, we use SBU dataset in Sec. 5.5.

5.4. Comparison with IAPR-TC12 and SBU dataset

Typical examples of generated captions are shown in Fig. 5 for the IAPR-TC12 and SBU datasets. Although there are many noisy descriptions [19], the results show that our methodology can generate captions for images from a large-scale dataset collected from the web.

Comparisons using the IAPR-TC12 and SBU are shown in Table 5 and Table 6, respectively. These tables also show that we achieve state-of-the-art performance using these datasets. [10] employs LMNN [45] to predict phrases. Although Table 2 shows that the small-scale dataset can be utilized by [10], Table 5 may indicate that such nonlinear metric learning fail to treat a middle-scale dataset.

Input Image	Estimated Keyphrases	Generated Sentence & Ground Truth
	blue sky clock tower blue in sky EOS tower in	Clock tower in the city of the blue sky. The clock tower at Sydney Uni against a perfect blue sky.
	stained glass glass window window in in St. the church	Stained glass window in the church in St. Vitus Cathedral. Stained glass window in Notre Dame.
	the background mountain range mountains in range in snow-covered summit	A green trees and brown mountain range in the background. A steep, grey canyon in the middle of a green valley with trees and houses, and a brownish, bald mountain in the background.
	the background tourists are are standing are sitting the middle	Tourists are standing on the middle of a flat desert. People are looking at rocks in the middle of a desert landscape.

Figure 5. Examples of estimated phrases and captions for the IAPR-TC12 (top two) and SBU (bottom two). Each column has the same information as Fig. 4.

Input Image	# of Images	Generated Sentence
	1K	Is a train station in the lake in the small.
	10K	All the lake in the water is a shot.
	100K	View of the lake in the water in a boat.
	1M	It is a picture of the boat in the water.
	1K	Building a 5D2 from a bar in the evening sky.
	10K	To my desk in the box in the little girl.
	100K	Fienile master bedroom window in the house in my office.
	1M	Desk in the kitchen table in the wall.
	1K	On the 13h floor in the beach at the park.
	10K	Hat in the roof of the castle in the tower.
	100K	Like the backfgnd of our house in the bottom.
	1M	An office building near the roof of the building.
	1K	Stained glass window in aanbouw cofferdam for a field.
	10K	Window in the ossuary glass windows in St. Louis Missouri.
	100K	Stained glass in the tower of the church in St.
	1M	Stained glass window in the church in St. Vitus Cathedral.

Figure 6. Examples of captions generated for PASCAL Sentence (top two) and SBU (bottom two) images using a varying number of SBU datasets.

5.5. Justification for collecting larger web images

We generated captions for images from PASCAL Sentence after learning pairs of images and captions from SBU. The objective of this experiment was to investigate (1) if we could generate captions with automatically collected web data rather than manually labeled data, and (2) the impact of the size of SBU.

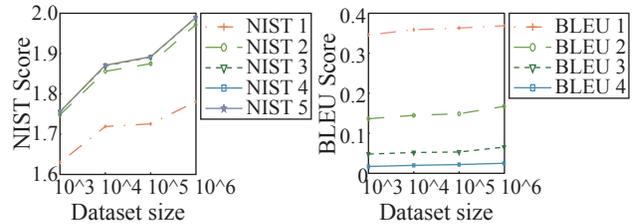


Figure 7. Impact of different dataset sizes within SBU.

We trained our system using 1M images from SBU and generated captions for both PASCAL Sentence and SBU images. We reduced the number of images before generating captions. Fig. 6 presents examples of captions generated using a varying number of SBU datasets for PASCAL Sentence images and SBU images. All captions were generated using the same grammar model extracted from 1M captions in SBU. These examples indicate that, even if the dataset is not manually organized, we can generate a caption using numerous images collected from the web. The improvement of these captions demonstrates that phrase prediction can be improved when the number of images is increased. The NIST and BLEU scores for all dataset sizes up to the full-size dataset are shown in Fig. 7. As this figure shows, increasing the size of the dataset improves these scores, especially the NIST score. Because NIST emphasizes less frequent n-grams, this score improvement means that our system is able to learn less frequent phrases when the number of images is increased. Although the BLEU scores obtained with training samples from SBU are lower than those obtained with training samples from PASCAL Sentence shown in Table 2, they are still comparable to those of several existing works [18, 48].

6. Conclusion

In this paper, we address caption generation of images. We propose a novel subspace embedding method, Common Subspace for Model and Similarity (CoSMoS), for phrase learning using few training samples per phrase. CoSMoS obtains a subspace in which (a) all feature vectors associated with the same phrase are mapped as mutually close, (b) classifiers for each phrase are learned, and (c) training samples are shared among co-occurring phrases. We also propose a simple but effective phrase extraction method.

Our experimental results demonstrate that our method achieves state-of-the-art accuracy although RNN is not employed for caption generation. Captions can be generated for images even with automatically collected web data, and the accuracy of captions increases when the size of the dataset increases. This paper utilizes CNN only for feature extraction. In a future work, we can integrate CoSMoS into combined network of CNN and RNN [4, 14, 44].

References

- [1] G. Chechik, U. Shalit, V. Sharma, and S. Bengio. An online algorithm for large scale image similarity learning. In *NIPS*, 2009. 3
- [2] X. Chen and C. L. Zitnick. Mind’s eye: A recurrent visual representation for image caption generation. In *CVPR*, 2015. 3, 7
- [3] G. Doddington. Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In *HLT*, 2002. 6
- [4] J. Donahue, L. A. Hendricks, S. Guadarrama, M. Rohrbach, S. Venugopalan, K. Saenko, and T. Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *CVPR*, 2015. 3, 7, 8
- [5] H. Fang, S. Gupta, F. N. Iandola, R. Srivastava, L. Deng, P. Dollár, J. Gao, X. He, M. Mitchell, J. C. Platt, C. L. Zitnick, and G. Zweig. From captions to visual concepts and back. In *CVPR*, 2015. 3, 7
- [6] A. Farhadi, M. Hejrati, M. A. Sadeghi, P. Young, C. Rashtchian, J. Hockenmaier, and D. Forsyth. Every picture tells a story: Generating sentences from images. In *ECCV*, 2010. 2
- [7] A. Farhadi and M. A. Sadeghi. Phrasal recognition. *PAMI*, 35(12):2854–65, 2013. 1, 3
- [8] M. Grubinger, P. Clough, H. Müller, and T. Deselaers. The iapr tc-12 benchmark: A new evaluation resource for visual information systems. In *International Workshop OntoImage*, 2006. 5, 10
- [9] A. Gupta and P. Mannem. From image annotation to image description. In *ICONIP*, 2012. 5, 6, 13
- [10] A. Gupta, Y. Verma, and C. V. Jawahar. Choosing linguistics over vision to describe images. In *AAAI*, 2012. 1, 3, 5, 7
- [11] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997. 3, 7
- [12] M. Hodosh, P. Young, and J. Hockenmaier. Framing image description as a ranking task: Data, models and evaluation metrics. *JAIR*, 47:853–899, 2013. 2
- [13] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint*, (1408.5093), 2014. 5
- [14] A. Karpathy and L. Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *CVPR*, 2015. 3, 7, 8
- [15] R. Kiros, R. Salakhutdinov, and R. S. Zemel. Unifying visual-semantic embeddings with multimodal neural language models. In *NIPS*, 2014. 3
- [16] D. Klein and C. D. Manning. Accurate unlexicalized parsing. In *ACL*, 2003. 5
- [17] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012. 1, 3, 5, 7
- [18] G. Kulkarni, V. Premraj, S. Dhar, S. Li, Y. Choi, A. C. Berg, and T. L. Berg. Baby talk: Understanding and generating image descriptions. In *CVPR*, 2011. 1, 2, 7, 8
- [19] P. Kuznetsova, V. Ordonez, A. Berg, T. Berg, Y. Choi, and S. Brook. Generalizing image captions for image-text parallel corpus. In *ACL*, 2013. 7
- [20] P. Kuznetsova, V. Ordonez, A. C. Berg, T. L. Berg, and Y. Choi. Collective generation of natural image descriptions. In *ACL*, 2012. 1, 2, 5, 7
- [21] A. Lavie and A. Agarwal. Meteor: An automatic metric for mt evaluation with high levels of correlation with human judgments. In *ACL WMT*, 2007. 7
- [22] S. Li, G. Kulkarni, T. L. Berg, A. C. Berg, and Y. Choi. Composing simple image descriptions using web-scale n-grams. In *CoNLL*, 2011. 1, 2, 3
- [23] T.-y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollar, and C. L. Zitnick. Microsoft coco: Common objects in context. *arXiv preprint*, 1405.0312, 2014. 5
- [24] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, (2):91–110, 2004. 5, 10
- [25] M. Marszalek and C. Schmid. Semantic hierarchies for visual object recognition. In *CVPR*, 2007. 2
- [26] T. Mensink, J. Verbeek, F. Perronnin, and G. Csurka. Large scale metric learning for distance-based image classification. Technical report, LEAR - INRIA and TVPA - XRCE, 2012. 2, 3, 11
- [27] T. Mensink, J. Verbeek, F. Perronnin, and G. Csurka. Metric learning for large scale image classification: Generalizing to new classes at near-zero cost. In *ECCV*, 2012. 2, 3, 11
- [28] M. Mitchell, J. Dodge, A. Goyal, K. Yamaguchi, K. Stratos, X. Han, A. Mensch, A. Berg, T. Berg, and H. Daumé III. Midge: Generating image descriptions from computer vision detections. In *EACL*, 2012. 1, 2, 3, 5
- [29] V. Ordonez, G. Kulkarni, and T. L. Berg. Im2text: Describing images using 1 million captioned photographs. In *NIPS*, 2011. 2, 5, 7
- [30] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. Bleu: a method for automatic evaluation of machine translation. In *ACL*, 2002. 6
- [31] F. Perronnin, J. Sánchez, and T. Mensink. Improving the fisher kernel for large-scale image classification. In *ECCV*, 2010. 5
- [32] B. T. Polyak and A. B. Juditsky. Acceleration of stochastic approximation by averaging. *SIAM Journal on Control and Optimization*, 30(4):838–855, 1992. 4
- [33] C. Rashtchian, P. Young, M. Hodosh, and J. Hockenmaier. Collecting image annotations using amazon’s mechanical turk. In *Proceedings of NAACL HLT Workshop on Creating Speech and Language Data with Amazon’s Mechanical Turk*, 2010. 5
- [34] 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. *IJCV*, 2015. 5
- [35] M. A. Sadeghi and A. Farhadi. Recognition using visual phrases. In *CVPR*, 2011. 1, 3
- [36] R. Salakhutdinov, A. Torralba, and J. Tenenbaum. Learning to share visual appearance for multiclass object detection. In *CVPR*, 2011. 2
- [37] J. Sánchez and F. Perronnin. High-dimensional signature compression for large-scale image classification. In *CVPR*, 2011. 1, 3
- [38] M. Schuster and K. K. Paliwal. Bidirectional recurrent neural networks. *TSP*, 45(11):2673–2681, 1997. 7
- [39] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *CVPR*, 2015. 5, 7
- [40] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *CVPR*, 2015. 7
- [41] Y. Ushiku, T. Harada, and Y. Kuniyoshi. Efficient image annotation for automatic sentence generation. In *ACMMM*, 2012. 1, 3, 4, 5, 6, 7, 11, 12
- [42] C. J. Veenman and D. M. Tax. Less: a model-based classifier for sparse subspaces. *PAMI*, 27(9):1496–500, 2005. 2, 3
- [43] Y. Verma, A. Gupta, P. Mannem, and C. Jawahar. Generating image descriptions using semantic similarities in the output space. In *Proceedings of CVPR Workshop on Language for Vision*, 2013. 1, 3, 5, 7
- [44] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. Show and tell: A neural image caption generator. In *CVPR*, 2015. 3, 7, 8
- [45] K. Q. Weinberger, J. Blitzer, and L. K. Saul. Distance metric learning for large margin nearest neighbor classification. In *NIPS*, 2006. 3, 7
- [46] J. Weston, S. Bengio, and N. Usunier. Large scale image annotation: Learning to rank with joint word-image embeddings. *Machine Learning*, 81:21–35, 2010. 4, 11
- [47] J. Weston, S. Bengio, and N. Usunier. Wsabie: Scaling up to large vocabulary image annotation. In *IJCAI*, 2011. 2, 3, 4, 11
- [48] Y. Yang, C. L. Teo, H. Daumé III, and Y. Aloimonos. Corpus-guided sentence generation of natural images. In *EMNLP*, 2011. 1, 2, 5, 7, 8