

Generating Image Descriptions Using Semantic Similarities in the Output Space

Yashaswi Verma Ankush Gupta Prashanth Mannem C. V. Jawahar
International Institute of Information Technology, Hyderabad, India

Abstract

Automatically generating meaningful descriptions for images has recently emerged as an important area of research. In this direction, a nearest-neighbour based generative phrase prediction model (PPM) proposed by (Gupta et al. 2012) was shown to achieve state-of-the-art results on PASCAL sentence dataset, thanks to the simultaneous use of three different sources of information (i.e. visual clues, corpus statistics and available descriptions). However, they do not utilize semantic similarities among the phrases that might be helpful in relating semantically similar phrases during phrase relevance prediction. In this paper, we extend their model by considering inter-phrase semantic similarities. To compute similarity between two phrases, we consider similarities among their constituent words determined using WordNet. We also re-formulate their objective function for parameter learning by penalizing each pair of phrases unevenly, in a manner similar to that in structured predictions. Various automatic and human evaluations are performed to demonstrate the advantage of our “semantic phrase prediction model” (SPPM) over PPM.

1. Introduction

Along with the outburst of digital photographs on the Internet as well as in personal collections, there has been a parallel growth in the amount of images with relevant and more or less structured captions. This has opened-up new dimensions to deploy machine learning techniques to study available descriptions, and build systems to describe new images automatically. Analysis of available image descriptions would help to figure out possible relationships that exist among different entities within a sentence (e.g. *object*, *action*, *preposition*, etc.). However, even for simple images, automatically generating such descriptions may be quite complex, thus suggesting the hardness of the problem.

Recently, there have been few attempts in this direction [2, 6, 8, 9, 12, 15, 17, 24]. Most of these approaches rely on visual clues (global image features and/or trained detectors and classifiers) and generate descriptions in an independent manner. This makes such methods susceptible

to linguistic errors during the generation step. An attempt towards addressing this was made in [6] using a nearest-neighbour based model. This model utilizes image descriptions at hand to learn different language constructs and constraints practiced by humans, and associates this information with visual properties of an image. It extracts linguistic phrases of different types (e.g. “white aeroplane”, “aeroplane at airport”, etc.) from available sentences, and uses them to describe new images. The underlying hypothesis of this model is that an image inherits the phrases that are present in the ground-truth of its visually similar images. This simple but conceptually coherent hypothesis resulted in state-of-the-art results on PASCAL-sentence dataset [19]¹.

However, this hypothesis has its limitations as well. One such limitation is the ignorance of semantic relationships among the phrases; i.e., presence of one phrase should trigger presence of other phrases that are *semantically similar* to it. E.g., consider a set of three phrases {“kid”, “child”, “building”}, an image J and its neighbouring image I . If the image I has the phrase “kid” in its ground-truth, then according to the model of [6], it will get associated with J with some probability, while (almost) ignoring the remaining phrases. However, if we look at these phrases, then it can be easily noticed that the phrases “kid” and “child” are semantically very similar, whereas the phrases “child” and “building” are semantically very different. Thus, it would not be justifiable to treat the phrases “child” and “building” as *equally* absent. That is to say, presence of “kid” should also indicate the presence of the phrase “child”. From the machine learning perspective, this relates with the notion of predicting structured outputs [21]. Intuitively, it asserts that given a true (or positive) label and a set of false (or negative) labels, each negative label should be penalized unevenly depending on its (dis)similarity with the true label.

In this paper, we try to address this limitation of the phrase prediction model (PPM) of [6]. For this, we propose two extensions to PPM. First, we modify their model for predicting a phrase given an image. This is performed by considering semantic similarities among the phrases. And second, we propose a parameter learning formulation in the nearest-neighbour set-up that takes into account the relation

¹<http://vision.cs.uiuc.edu/pascal-sentences/>

(structure) present in the output space. This is a generic formulation and can be used/extended to other scenarios (such as metric learning in nearest-neighbour based methods [23]) where structured prediction needs to be performed using some nearest-neighbour based model. Both of our extensions utilize semantic similarities among phrases determined using WordNet [3]. Since our model relies on consideration of semantics among phrases during prediction, we call it “*semantic phrase prediction model*” (or SPPM). We perform several automatic and human evaluations to demonstrate the advantage of SPPM over PPM.

2. Related Works

Here we discuss some of the notable contributions in this domain. In [25], a semi-automatic method is proposed where first an image is parsed and converted into a semantic representation, which is then used by a text parse engine to generate image description. The visual knowledge is represented using a parse graph which associates objects with WordNet synsets to acquire categorical relationships. Using this, they are able to compose new rule-based grounded symbols (e.g., “zebra” = “horse” + “stripes”). In [8], they use trained detectors and classifiers to predict the objects and attributes present in an image, and simple heuristics to figure out the preposition between any two objects. These predictions are then combined with corpus statistics (frequency of a term in a large text corpus, e.g. Google) and given as an input to a CRF model. The final output is a set of objects, their attributes and a preposition for each pair of objects, which are then mapped to a sentence using a simple template-based approach. Similar to this, [24] relies on detectors and classifiers to predict upto two objects and the overall scene of an image. Along with preposition, they also predict the action performed by subject; and combine the predictions using an HMM model. In [12], the outputs of object detectors are combined with frequency counts of different n-grams ($n \leq 5$) obtained using the Google-1T data. Their phrase fusion technique specifically infuses some creativity into the output descriptions. Another closely related work with similar motivation is [15].

One of the limitations of most of these methods is that they don’t make use of available descriptions. This may help in avoiding generation of noisy/absurd descriptions (e.g. “person under road”). Two recent methods [6, 9] try to address this issue by making use of higher-level language constructs, called *phrases*. A phrase is a collection of syntactically ordered words that is semantically meaningful and complete on its own (e.g., “person pose”, “cow in field”, etc.)². In [9], phrases are extracted from the dataset proposed in [17]. Then, an integer-programming based formulation is used that fuses visual clues with words and phrases

to generate sentences. In [6], a nearest-neighbour based model is proposed that simultaneously integrates three different sources of information, i.e. visual clues, corpus statistics and available descriptions. They use linguistic phrases extracted from available sentences to construct descriptions for new images. These two models are closely related with the notion of visual phrases [20], which says that it is more meaningful to detect visual phrases (e.g. “person next to car”) than individual objects in an image.

Apart from these, there are few other methods that directly transfer one or more complete sentences from a collection of sentences. E.g., the method proposed in [17] transfers multiple descriptions from some other images to a given image. They discuss two ways to perform this: (i) using global image features to find similar images, and (ii) using detectors to re-rank the descriptions obtained after the first step. Their approach mainly relies on a very large collection of one million captioned images. Similar to [17], in [2] also a complete sentence from the training image descriptions is transferred by mapping a given (test) image and available descriptions into a “meaning space” of the form $(object, action, scene)$. This is done using a retrieval based approach combined with an MRF model.

3. Phrase Prediction Model

In this section, we briefly discuss PPM [6]. Given images and corresponding descriptions, a set of phrases \mathcal{Y} is extracted using all the descriptions. These phrases are restricted to five different types (considering “subject” and “object” as equivalent for practical purposes): $(object)$, $(attribute, object)$, $(object, verb)$, $(verb, prep, object)$, and $(object, prep, object)$. The dataset takes the form $\mathcal{T} = \{(I_i, Y_i)\}$, where I_i is an image and $Y_i \subseteq \mathcal{Y}$ is its set of phrases. Each image I is represented using a set of n features $\{f_{1,I}, \dots, f_{n,I}\}$. Given two images I and J , distance between them is computed using a weighted sum of distances corresponding to each feature as:

$$D_{I,J} = w_1 d_{1,I,J} + \dots + w_n d_{n,I,J} = \mathbf{w} \cdot \mathbf{d}_{I,J}, \quad (1)$$

where $w_i \geq 0$ denotes the weight corresponding to i^{th} feature distance. Using this, for a new image I , its K most similar images $\mathcal{T}_I^K \subseteq \mathcal{T}$ are picked. Then, the joint probability of associating a phrase $y_i \in \mathcal{Y}$ with I is given by:

$$P(y_i, I) = \sum_{J \in \mathcal{T}_I^K} P_{\mathcal{T}}(J) P_{\mathcal{F}}(I|J) P_{\mathcal{Y}}(y_i|J). \quad (2)$$

Here, $P_{\mathcal{T}}(J) = 1/K$ denotes the uniform probability of picking some image J from \mathcal{T}_I^K . $P_{\mathcal{F}}(I|J)$ denotes the likelihood of image I given J , defined as:

$$P_{\mathcal{F}}(I|J) = \frac{\exp(-D_{I,J})}{\sum_{J' \in \mathcal{T}_I^K} \exp(-D_{I,J'})}. \quad (3)$$

²The term ‘phrase’ is used in a more general sense, and is different from the linguistic sense of phrase.

Finally, $P_Y(y_i|J)$ denotes the probability of seeing the phrase y_i given image J , and is defined according to [4]:

$$P_Y(y_i|J) = \frac{\mu_i \delta_{y_i,J} + N_i}{\mu_i + N}. \quad (4)$$

Here, if $y_i \in Y_J$, then $\delta_{y_i,J} = 1$ and 0 otherwise. N_i is the (approximate) Google count of the phrase y_i , N denotes the sum of Google counts of all phrases in \mathcal{Y} that are of the same type as that of y_i , and $\mu_i \geq 0$ is the smoothing parameter. The motivation behind using Google counts of phrases is to smooth their relative frequencies.

In order to learn the two sets of parameters (i.e., the weights w_i 's and smoothing parameters μ_i 's), an objective function analogous to [23] is used. Given an image J along with its true phrases Y_J , the goal is to learn the parameters such that (i) the probability of predicting the phrases in $\mathcal{Y} \setminus Y_J$ should be minimized, and (ii) the probability of predicting each phrase in Y_J should be more than any other phrase. Precisely, we minimize the following function:

$$e = \sum_{J, y_k} P(y_k, J) + \lambda \sum_{(J, y_k, y_j) \in \mathcal{M}} (P(y_k, J) - P(y_j, J)). \quad (5)$$

Here, $y_j \in Y_J$, $y_k \in \mathcal{Y} \setminus Y_J$, \mathcal{M} is the set of triples that violate the second constraint stated above, and $\lambda > 0$ is used to manage the trade-off between the two terms. The objective function is optimized using a gradient descent method, by learning w_i 's and μ_i 's in an alternate manner.

Using equation 2, a ranked list of phrases is obtained, which are then integrated to produce *triples* of the form $\{((attribute1, object1), verb), (verb, prep, (attribute2, object2)), (object1, prep, object2)\}$. These are then mapped to simple sentences using SimpleNLG [5].

4. Semantic Phrase Prediction Model

As discussed before, one of the limitations of PPM is that it treats phrases in a binary manner; i.e., in equation 4, either $\delta_{y_i,J}$ is 1 or 0 depending on presence or absence of y_i in Y_J . This results in penalizing semantically similar phrases (e.g. “person” vs. “man”). Here we extend this model by considering semantic similarities among phrases. To begin with, first we discuss how to compute semantic similarities.

4.1. Computing Semantic Similarities

Let a_1 and a_2 be two words (e.g. “boy” and “man”). We use WordNet based JCN similarity measure [7] to compute semantic similarity between the words a_1 and a_2 ³. WordNet is a large lexical database of English where words are inter-linked in a hierarchy based on their semantic and lexical relationships. Given a pair of words (a_1, a_2) , the JCN similarity measure returns a score $s_{a_1 a_2}$ in the range $[0, \inf)$, with

³Using the code available at <http://search.cpan.org/CPAN/authors/fid/TP/TPEDERSE/WordNet-Similarity-2.05.tar.gz>

higher score corresponding to larger similarity and vice-versa. This similarity score is then mapped into the range $[0, 1]$ using the following non-linear transformation as described in [11] (denoting $s_{a_1 a_2}$ by s in short):

$$\gamma(s) = \begin{cases} 1 & s \geq 0.1 \\ 0.6 - 0.4 \sin(\frac{25\pi}{2}s + \frac{3}{4}\pi) & s \in (0.06, 0.1) \\ 0.6 - 0.6 \sin(\frac{\pi}{2}(1 - \frac{1}{3.471s+0.653})) & s \leq 0.06 \end{cases}$$

Using this, we define a similarity function that takes two words as input and returns the semantic similarity score between them computed using the above equation as:

$$W_{sim}(a_1, a_2) = \gamma(s_{a_1 a_2}) \quad (6)$$

From this, we compute semantic dissimilarity score as:

$$\bar{W}_{sim}(a_1, a_2) = 1 - W_{sim}(a_1, a_2) \quad (7)$$

Based on equation 6, we define semantic similarity between two phrases (of the same type) as V_{sim} , which is an average of the semantic similarity between each of their corresponding constituting terms. E.g., if we have two phrases v_1 = (“person”, “walk”) and v_2 = (“boy”, “run”) of the type (*object, verb*), then their semantic similarity score will be given by $V_{sim}(v_1, v_2) = 0.5 * (W_{sim}(\text{“person”}, \text{“boy”}) + W_{sim}(\text{“walk”}, \text{“run”}))$. It should be noted that we cannot compute semantic similarity between two prepositions using WordNet. So, while computing semantic similarity between two phrases that contain prepositions in them (i.e., of type (*verb, prep, object*) or (*object, prep, object*)), we do not consider the prepositions. Analogous to equation 7, we can compute semantic dissimilarity score between two phrases as $\bar{V}_{sim}(v_1, v_2) = 1 - V_{sim}(v_1, v_2)$. Finally, given a phrase y_i and a set of phrases Y of the same type as that of y_i , we define semantic similarity between them as

$$U_{sim}(y_i, Y) = \max_{y_j \in Y} V_{sim}(y_i, y_j). \quad (8)$$

In practice, if $|Y| = 0$ then we set $U_{sim}(y_i, Y) = 0$. Also, in order to emphasize more on an exact match, we set $U_{sim}(y_i, Y)$ to $\exp(1)$ if $y_i \in Y$ in the above equation.

4.2. SPPM

In order to benefit from semantic similarity between two phrases while predicting relevance of some given phrase y_i with Y_J of image J , we need to modify equation 4 accordingly. Let y_i be of type t , and the set of phrases of type t in Y_J be $Y_J^t \subseteq Y_J$. Then, we re-define $P_Y(y_i|J)$ as:

$$P_Y(y_i|J) = \frac{\mu_i \delta'_{y_i,J} + N_i}{\mu_i + N}, \quad (9)$$

where $\delta'_{y_i,J} = U_{sim}(y_i, Y_J^t)$. This means that when $y_i \notin Y_J^t$, we look for that phrase in Y_J^t that is semantically most similar to y_i and use their similarity score,

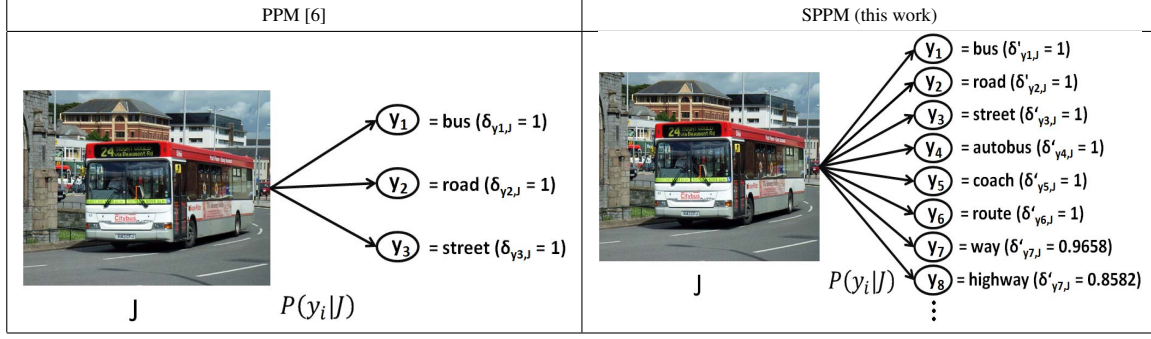


Figure 1. Difference between the two models. In PPM, the conditional probability of a phrase y_i given an image J depends on whether that phrase is present in the ground-truth phrases of J (i.e. Y_J) or not. When the phrase is not present, corresponding $\delta_{y_i,J}$ (equation 4) becomes zero without considering the semantic similarity of y_i with other phrases in Y_J . This limitation of PPM is addressed in SPPM by finding the phrase in Y_J that is semantically most similar to y_i and using their similarity score instead of zero. In the above example, we have $Y_J = \{\text{"bus", "road", "street"}\}$. Given a phrase $y_i = \text{"highway"}$, $\delta_{y_i,J} = 0$ according to PPM. Whereas $\delta'_{y_i,J} = 0.8582$ according to SPPM (equation 9) by considering the similarity of “highway” with “road” (i.e., $V_{sim}(\text{"highway"}, \text{"road"}) = 0.8582$).

rather than putting a zero. Such a definition allows us to take into account the structure/semantic inter-dependence among phrases while predicting the relevance of a phrase.

Since we have modified the conditional probability model for predicting a phrase given an image, we also need to update the objective function of equation 5 accordingly. Given an image J along with its true phrases y_j 's in Y_J , now we additionally need to ensure that the penalty imposed for a higher relevance score of some phrase $y_k \in \mathcal{Y} \setminus Y_J$ than any phrase $y_j \in Y_J$ should also depend on the semantic similarity between y_j and y_k . This is similar to the notion of predicting structured outputs as discussed in [21]. Precisely, we re-define the objective function as:

$$e = \sum_{J, y_k} P(y_k, J) + \lambda \sum_{(J, y_k, y_j) \in \mathcal{M}} \Delta(J, y_k, y_j), \quad (10)$$

$$\Delta(J, y_k, y_j) = \bar{V}_{sim}(y_k, y_j)(P(y_k, J) - P(y_j, J)). \quad (11)$$

The implication of $\Delta(\cdot)$ is that if two phrases are semantically similar (e.g. “kid” and “child”), then penalty should be small and vice-versa. This objective function looks similar to that used in [22] for metric learning in nearest neighbour scenario. The major difference being that there the objective function is defined over samples, and penalty is based on semantic similarity between two samples (proportional to number of labels they share). Whereas, here the objective function is defined over phrases, and penalty is based on semantic similarity between two phrases.

5. Experiments

5.1. Experimental Details

We follow the same experimental set-up as in [6], and use UIUC PASCAL sentence dataset [19] for evaluation. It has 1,000 images and each image is described using 5 independent sentences. These sentences are used to extract

different types of phrases using “collapsed-ccprocessed-dependencies” in the Stanford CoreNLP toolkit [1]⁴, giving 12,865 distinct phrases. In order to consider synonyms, WordNet synsets are used to expand each noun upto 3 hyponym levels resulting in a reduced set of 10,429 phrases.

Similar to [6], we partition the dataset into 90% training and 10% testing for learning the parameters, and repeat this over 10 partitions in order to generate descriptions for all the images. During relevance prediction, we consider $K = 15$ nearest-neighbours from the training data.

For image representation, we use a set of colour (RGB and HSV), texture (Gabor and Haar), scene (GIST [16]) and shape (SIFT [14]) descriptors computed globally. All features other than GIST are also computed over three equal horizontal and vertical partitions [10]. This gives a set of 16 features per image. While computing distance between two images (equation 1), $L1$ distance is used for colour, $L2$ for scene and texture, and χ^2 for shape features.

5.2. Evaluation Measures

In our experiments, we perform both automatic as well as human evaluations for performance analysis.

5.2.1 Automatic Evaluation

For this we use the BLEU [18] and Rouge [13] metrics. These are frequently used for evaluations in the areas of machine translation and automatic summarization respectively.

5.2.2 Human Evaluation

Automatically describing an image is significantly different from machine translation or summary generation. Since an image can be described in several ways, it is not justifiable

⁴<http://nlp.stanford.edu/software/corenlp.shtml>

Approach	BLEU-1 Score	Rouge-1 Score
BabyTalk [8]	0.30	-
CorpusGuided [24]	-	0.44
PPM [6] w/ syn.	0.41	0.28
PPM [6] w/o syn.	0.36	0.21
SPPM w/ syn.	0.43	0.29
SPPM w/o syn.	0.36	0.20

Table 1. Automatic evaluation results for sentence generation. (Higher score means better performance.)

Approach	Readability	Relevance
PPM [6] w/ syn.	2.84	1.49
PPM [6] w/o syn.	2.75	1.32
SPPM w/ syn.	2.93	1.61
SPPM w/o syn.	2.91	1.39

Table 2. Human evaluation results for “Relevance” and “Readability”. (Higher score means better performance.)

to rely just on automatic evaluation, and hence the need for human evaluation arises. We gather judgements from two human evaluators on 100 images randomly picked from the dataset and take their average. The evaluators are asked to verify three aspects on a likert scale of $\{1, 2, 3\}$ [6, 12]:

Readability: To measure grammatical correctness of generated description by giving the following ratings: (1) Terrible, (2) Mostly comprehensible with some errors, (3) Mostly perfect English sentence.

Relevance: To measure the semantic relevance of the generated sentence by giving the following ratings: (1) Totally off, (2) Reasonably relevant, (3) Very relevant.

Relative Relevance: We also try to analyze the relative relevance of descriptions generated using PPM and SPPM. Corresponding to each image, we present the descriptions generated using these two models to the human evaluators (without telling them that they are generated using two different models) and collect judgements based on the following ratings: (1) Description generated by PPM is more relevant, (2) Description generated by SPPM is more relevant, (3) Both descriptions are equally relevant/irrelevant.

5.3. Results and Discussion

5.3.1 Quantitative Results

Table 1 shows the results corresponding to automatic evaluations. It can be noticed that SPPM shows comparable or superior performance than PPM. One important thing that we would like to point out is that it is not fully justifiable to directly compare our results with those of [8] and [24]. This is because the data (i.e., the fixed sets of objects, prepositions, verbs) that they use for composing new sentences is very much different from that of ours. However, in [6], it

	PPM [6] count	SPPM count	Both/None count
w/ syn.	16	28	56
w/o syn.	21	25	54

Table 3. Human evaluation results for “Relative Relevance”. Last column denotes the number of times descriptions generated using the two methods were judged as equally relevant or irrelevant with given image. (Larger count means better performance.)


	PPM: (1) flap (2) csa (3) symbol (4) aircraft (5) slope (6) crag (7) villa (8) biplane (9) distance (10) sky
	SPPM: (1) aeroplane (2) airplane (3) plane (4) sky (5) boat (6) water (7) air (8) aircraft (9) jet (10) gear

Figure 3. Example image from the PASCAL sentence dataset along with the top ten “objects” predicted using the two models.

was shown that when same data is used, PPM performs better than both of these. Since the data that we use in our experiments is exactly the same as that of PPM, and SPPM performs comparable or better than PPM, we believe that under the same experimental set-up, our model would perform better than both [8] and [24]. Also, we are not comparing with other works because since this is an emerging domain, different works have used either different evaluation measures (such as [2]), or experimental set-up (such as [15]), or even datasets (such as [9, 17]). In conclusion, our results are directly comparable only with PPM [6].

5.3.2 Qualitative Results

Human evaluation results corresponding to “Readability” and “Relevance” are shown in Table 2. Here, we can notice that SPPM consistently performs better than PPM on all the evaluation metrics. This is because SPPM takes into account semantic similarities among the phrases, which in turn results in generating more coherent descriptions than PPM. This is also highlighted in Figure 2 that shows example descriptions generated using PPM and SPPM. It can be noticed that the words in descriptions generated using SPPM usually show semantic connectedness; which is not always the case with PPM. E.g., compare the descriptions obtained using PPM (in the second row) with those obtained using SPPM (in the fourth row) for the last three images. In Table 3, results corresponding to “Relative Relevance” are shown. In this case also, SPPM always performs better than PPM. This means that the descriptions generated using SPPM are semantically more relevant than those using PPM.

In Figure 3, we try to get some insight about how the internal functioning of SPPM is different from that of PPM. For this, we show the top ten phrases of the type “object”





			
A groom is posing with a scraggly person.	A sandy field is parked beside a small outpost.	A teal car is sitting atop a white semus.	A decorate room is filling with a snack.
A blond woman is posing in a library.	A sandy field is parked beside a small outpost.	A black ferrari is parked in front of a green tree.	A clothed table is filling with a snack.
A young person is posing with a young person.	A small boat is traveling in a blue water.	A yellow bus is parking on a busy road.	A several person is sitting with a several person.
A gray man is posing with a gray man.	A small boat is floating on a blue water.	A yellow bus is parking on a city street.	A gray man is sitting at a restaurant table.

Figure 2. Example images from PASCAL sentence dataset along with their generated descriptions. The descriptions in second and third rows are generated using PPM [6] with and without considering synonyms respectively.. The descriptions in fourth and fifth rows are generated using SPPM with and without considering synonyms respectively.

predicted using the two models for an example image. From these phrases, it can be noticed that the top phrases obtained using SPPM are all semantically very much related with each other. Whereas, in case of PPM, the phrases are quite diversified. This is because in SPPM, the relevance (or presence) of a phrase also depends on the presence of other phrases that are semantically similar to it. This results in an indirect propagation of relevance among the phrases, thus collectively pushing semantically related phrases towards the top.

6. Conclusion

We have presented an extension to PPM [6] by incorporating semantic similarities among phrases during phrase prediction and parameter learning steps. As the number of phrases increases, inter-phrase relationships start getting prominent. However, due to the phenomenon of “long-tail”, available data alone might not be sufficient to learn such complex relationships, and thus arises the need of bringing-in knowledge from other sources. In this work, we have tried to perform this using WordNet. To the best of our knowledge, this is the attempt of its kind in this domain, and can be integrated with other similar models as well.

References

- [1] M.-C. de Marneffe and C. D. Manning. The stanford typed dependencies representation. In *COLING Workshop on Cross-framework and Cross-domain Parser Evaluation*, 2008.
- [2] A. Farhadi, M. Hejrati, A. Sadeghi, P. Young, C. Rashtchian, J. Hockenmaier, and D. Forsyth. Every picture tells a story: Generating sentences for images. In *ECCV*, 2010.
- [3] C. Fellbaum. Wordnet: An electronic lexical database. *Cambridge, MA: MIT Press*, 1998.
- [4] S. L. Feng, R. Manmatha, and V. Lavrenko. Multiple bernoulli relevance models for image and video annotation. In *CVPR*, 2004.
- [5] A. Gatt and E. Reiter. Simplenlg: A realisation engine for practical applications. In *ENLG*, 2009.
- [6] A. Gupta, Y. Verma, and C. V. Jawahar. Choosing linguistics over vision to describe images. In *AAAI*, 2012.
- [7] J. J. Jiang and D. W. Conrath. Semantic similarity based on corpus statistics and lexical taxonomy. In *ICRCL*, 1997.
- [8] G. Kulkarni, V. Premraj, S. Dhar, S. Li, Y. Choi, A. C. Berg, and T. L. Berg. Baby talk: Understanding and generating simple image descriptions. In *CVPR*, 2011.
- [9] P. Kuznetsova, V. Ordonez, A. C. Berg, T. L. Berg, and Y. Choi. Collective generation of natural image descriptions. In *ACL*, 2012.
- [10] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *CVPR*, 2006.
- [11] J. Li. A mutual semantic endorsement approach to image retrieval and context provision. In *MIR*, 2005.
- [12] 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.
- [13] C.-Y. Lin and E. Hovy. Automatic evaluation of summaries using n-gram co-occurrence statistics. In *NAACLHLT*, 2003.
- [14] D. G. Lowe. Distinctive image features from scale-invariant keypoints. In *IJCV*, 2004.
- [15] M. Mitchell, J. Dodge, A. Goyal, K. Yamaguchi, K. Sratos, X. Han, A. Mensch, A. C. Berg, T. L. Berg, and H. D. III. Midge: Generating image descriptions from computer vision detections. In *EACL*, 2012.
- [16] A. Oliva and A. Torralba. Modelling the shape of the scene: A holistic representation of the spatial envelope. *IJCV*, 2001.
- [17] V. Ordonez, G. Kulkarni, and T. L. Berg. Im2text: Describing images using 1 million captioned photographs. In *NIPS*, 2011.
- [18] K. Papineni, S. Roukos, T. Ward, and W. Zhu. Bleu: A method for automatic evaluation of machine translation. In *ACL*, 2002.
- [19] C. Rashtchian, P. Young, M. Hodosh, and J. Hockenmaier. Collecting image annotation using amazon’s mechanical turk. In *NAACLHLT Workshop on Creating Speech and Language Data with Amazon’s Mechanical Turk*, 2010.
- [20] M. A. Sadeghi and A. Farhadi. Recognition using visual phrases. In *CVPR*, 2011.
- [21] I. Tsoukantaridis, T. Hofmann, T. Joachims, and Y. Altun. Support vector machine learning for interdependent and structured output spaces. In *ICML*, 2004.
- [22] Y. Verma and C. V. Jawahar. Image annotation using metric learning in semantic neighbourhoods. In *ECCV*, 2012.
- [23] K. Q. Weinberger and L. K. Saul. Distance metric learning for large margin nearest neighbor classification. 2009.
- [24] Y. Yang, C. L. Teo, H. D. III, and Y. Aloimonos. Corpus-guided sentence generation of natural images. In *EMNLP*, 2011.
- [25] B. Z. Yao, X. Yang, L. Lin, M. W. Lee, and S.-C. Zhu. I2t: Image parsing to text description. In *Proceedings of the IEEE*, 2008.