Learning to Interpret and Describe Abstract Scenes

Luis Gilberto Mateos Ortiz, Clemens Wolff and **Mirella Lapata**
School of Informatics, University of Edinburgh
10 Crichton Street, Edinburgh EH8 9AB
{clemens.wolff,luismattor}@gmail.com, mlap@inf.ed.ac.uk

Abstract

Given a (static) scene, a human can effortlessly describe what is going on (who is doing what to whom, how, and why). The process requires knowledge about the world, how it is perceived, and described. In this paper we study the problem of interpreting and verbalizing visual information using abstract scenes created from collections of clip art images. We propose a model inspired by machine translation operating over a large parallel corpus of visual relations and linguistic descriptions. We demonstrate that this approach produces human-like scene descriptions which are both fluent and relevant, outperforming a number of competitive alternatives based on templates, sentence-based retrieval, and a multimodal neural language model.

1 Introduction

What is going on in the scene in Figure 1? Is the boy trying to feed the dog or play with it? Why is the girl upset? Is it because the dog is wearing her glasses? Or perhaps she is just scared of the dog? Scene interpretation is effortless for humans, almost everyone can summarize Figure 1 in a few words, without probably paying too much attention to the fact the girl is wearing a pink dress, the sun is yellow or that there is a plane in the sky.

Discovering what an image means and relaying it in words is of theoretical importance raising questions about language and its grounding in the perceptual world but also has practical applications. Examples include sentence-based image search and tools that enhance the accessibility of the web for visually impaired (blind and partially sighted) individuals. Indeed, there has been a recent surge of interest in the development of models that automatically describe image content in natural language (see references in Section 2). Due to the complex nature of the problem, existing approaches resort to modeling simplifications, on the generation side (e.g., through the use of templates and sentence-based retrieval methods), or the image processing side (e.g., by avoiding object-detection), or both.

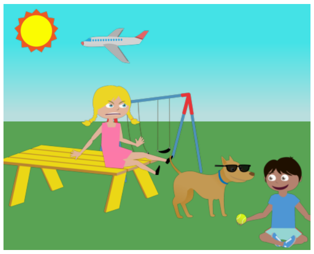


Figure 1: Given an image, humans do not simply see an arrangement of objects, they understand how they relate to each other as well as their attributes and the activities they are involved in.

In this paper we study the problem of interpreting visual scenes and rendering their content using natural language. We approach this problem within the methodology of Zitnick and Parikh (2013), who proposed the use of abstract scenes generated from clip art to model scene understanding (see Figure 1). The use of abstract scenes offers several advantages over real images. Firstly, it allows us to study the scene description problem in isolation, without the noise introduced by automatic object and attribute detectors in real images. Secondly, it is relatively easy to gather large amounts of data, allowing us to compare multiple models on an equal footing, study in more detail the problem of language grounding, and how to identify what is important in an image. Thirdly, information learned from abstract scenes will lead to better understanding of the challenges and data requirements arising when using real images.

We propose a model inspired by machine trans-

Human Language Technologies: The 2015 Annual Conference of the North American Chapter of the ACL, pages 1505–1515,
Denver, Colorado, May 31 – June 5, 2015. ©2015 Association for Computational Linguistics

lation, where the task is to transform a source sentence E into its target translation F. We argue that generating descriptions for scenes is quite similar, but with a twist: the translation process is very loose and selective; there will always be objects in a scene not worth mentioning, and words in a description that will have no visual counterpart. Our key insight is to represent scenes via visual dependency relations (Elliott and Keller, 2013) corresponding to sentential descriptions. This allows us to create a large parallel corpus for training a statistical machine translation system, which we interface with a content selection component guiding the translation toward interesting or important scene content. Advantageously, our model can be used in the reverse direction, i.e., to generate scenes, without additional engineering effort. Our approach outperforms a number of competitive alternatives, when evaluated both automatically and by humans.

2 Related Work

The task of image description generation has recently gained popularity in the natural language processing and computer vision communities. Several methods leverage recent advances in computer vision and generate novel sentences relying on object detectors, attribute predictors, action detectors, and pose estimators. Generation is performed using templates or syntactic rules which piece the description together while leveraging word-co-occurrence statistics (Kulkarni et al., 2011; Yang et al., 2011; Elliott and Keller, 2013; Mitchell et al., 2012). Recent advances in neural language models have led to approaches which generate captions by conditioning on feature vectors from the output of a deep convolutional neural network without the use of templates or syntactic trees (Kiros et al., 2014; Vinyals et al., 2014). Most methods assume no structural information on the image side either (images are represented as unstructured bags of regions or as feature vectors). A notable exception are Elliott and Keller (2013) who introduce visual dependency relations between objects and argue that such structured representations are beneficial for image description.

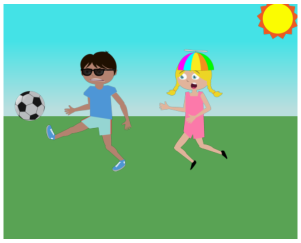
A large body of work has focused on the complementary problem of matching sentences (Ordonez et al., 2011; Farhadi et al., 2010; Hodosh et al., 2013; Feng and Lapata, 2013; Mason and Charniak, 2014) or phrases (Kuznetsova et al., 2012; Kuznetsova et al., 2014) to an image from existing human authored descriptions. Sentence-based approaches embed images and descriptions into the same multidimensional space, and retrieve descriptions from images most similar to a query image. Phrase-based approaches are more involved in that phrases need to be composed into a description and extraneous information optionally removed. A common modeling choice is the use of Integer Linear Programming (ILP) which naturally allows to encode various well-formedness constraints (e.g., grammaticality).

We are not aware of any previous work generating descriptions for abstract scenes, although the same dataset has been used to model sentence-to-scene generation (Zitnick et al., 2013) and predict object dynamics in scenes (Fouhey and Zitnick, 2014). Using the visual relations put forward in Elliott and Keller (2013), we convert the abstract scenes dataset into a parallel corpus of visual and linguistic descriptions, which allows us to train a statistical machine translation (SMT) model. In contrast to earlier work (Kuznetsova et al., 2014; Kuznetsova et al., 2012), which models the task as an optimization problem end-to-end, we employ ILP for content selection only, deferring the surface realization process entirely to an SMT engine.

3 The Abstract Scenes Dataset

The abstract scenes dataset[1] was created with the intent to represent real-world scenes that depict a diverse set of subtle relations. It contains 10,020 images of children playing outside and 60,396 descriptions (on average six per image). The data was collected in three stages. First, Amazon Mechanical Turk (AMT) workers were asked to created scenes for a collection of 80 pieces of clip art depicting a boy and a girl (in different poses and with different facial expressions), and several objects including trees, toys, hats, animals, and so on. Next, a new set of subjects were asked to describe the scenes using a one or two sentence description, finally, semantically similar scenes were generated by asking multiple subjects to create scenes depicting the same writ-

[1] http://research.microsoft.com/en-us/um/people/larryz/clipart/abstract_scenes.html



0 6						
s_3s.png	0	3	467	24	2	1
hb0_10s.png	2	10	145	182	0	1
hb1_19s_png	3	19	323	188	0	1
c_9s.png	5	9	161	116	0	1
c_5s.png	7	4	43	172	0	0
t_4s.png	7	4	43	173	0	0

"Jenny is upset because Mike isn't sharing the soccer ball.", "Mike is wearing sunglasses.", "Jenny is wearing a silly hat.", "Mike is kicking the soccer ball away from Jenny.", "Jenny is chasing Mike to get the ball.", "Jenny is wearing a silly hat."

Figure 2: Example of a scene, its rendering information (right), and human-written descriptions (bottom).

ten description. By construction, the dataset encodes the objects in each scene, and their position.

An example is shown in Figure 2. The table on the right-hand side specifies how the image was rendered. The top row contains the scene identifier (i.e., 0) and the number of pieces of clip art in the image (i.e., 6). The remaining rows encode rendering information for each individual piece of clipart, i.e., its name (column 1), type (column 2), attribute (column 3), position (columns 4–6), and whether or not it is horizontally flipped (column 7). Six human authored descriptions are shown the bottom. AMT participants were instructed to write simple descriptions using basic words that would appear in a book for young children ages 4–6. Participants who wished to use proper names in their descriptions were provided with names "Mike" and "Jenny" for the boy and girl. The vocabulary consists of 2,705 words, and the average sentence length is 6.3 words. As can be seen in Figure 2, subjects choose to focus on different aspects of the image (e.g., Mike and his sunglasses, the fact that Jenny is chasing Mike). Also notice that even though by design we know which visual objects are present in the image and their spatial relationships (see the right hand-side in Figure 2), the alignment between pieces of clipart and linguistic expressions is hidden. In other words, we do not know which actions the objects depict (e.g., playing, holding) and which words can be used to describe them (e.g., that t_4s.png is called a ball).

4 Problem Formulation

We formulate scene description generation as a translation problem from the visual to the linguistic modality. Our approach follows the general paradigm of SMT with two important differences. Firstly, the source side (i.e., scene) is fundamentally different from the target (i.e., linguistic descriptions) both in terms of representation and structure. Secondly, the scene and its corresponding descriptions constitute a very loose parallel corpus: not all visual objects are verbalized (note that no participant chose to mention the sun in Figure 2) and there are multiple valid descriptions for a single scene focusing on different objects and their relations. In the following we first describe how we create a parallel corpus representing the arrangement of objects in a scene and their linguistic realization and then we move on to present our generation model.

4.1 Parallel Corpus Creation

As mentioned earlier, each scene in the dataset has six descriptions (on average). For each *linguistic* description we create its corresponding *visual* encoding. We initially ground words and phrases by aligning them to pieces of clipart. We parse the descriptions using a dependency parser, and identify expressions that function as arguments (e.g., subject, object). In our experiments we used the Stanford parser (Klein and Manning, 2003) but any parser with similar output could have been used instead. Next, we compute the mutual information (MI) between arguments and clip-art objects defined as:

$$MI(X,Y) = \sum_{x \in X} \sum_{y \in Y} p(x,y) \log \frac{p(x,y)}{p(x)p(y)} \qquad (1)$$

where X is the set of clip-art objects and Y the set of arguments found in the dataset. We assume that the visual rendering of an argument is the clip-art object with which its MI value is highest. Figure 3 shows argument-clipart pairs with high MI values.

Having identified which objects in the scene are talked about, we move on to encode their spatial relations. We adopt the relations outlined in *Visual Dependency Grammar* (VDG; Elliott and Keller (2013)). The latter are defined for pairs of image regions but can also directly apply to clip-art objects. VDR Relations are specified according to

X on Y	More than 50% of X overlaps with Y
X surrounds Y	X overlaps entirely with Y
X above Y	The angle between X and Y is between 225°and 315°
X below Y	The angle between X and Y is between 45°and 135°.
X opposite Y	The angle between X and Y is between 315°and 45°or 135°and 225°. The Euclidean distance between X and Y is greater than $w \cdot 0.72$.
X near Y	Similar to *opposite* but the Euclidean distance between X and Y is greater than $w \cdot 0.36$.
X close Y	Similar to *opposite* but the Euclidean distance between X and Y is less or equal to $w \cdot 0.36$.
X infront Y	X is in front Y in the Z-plane
X behind Y	X is behind Y in the Z-plane
X same Y	X and Y are at the same depth

Table 1: VDG relations between pairs of clip art objects. All relations are considered with respect to the centroid of an object and the angle between those centroids. We follow the definition of the unit circle, in which 0°lies to the right and a turn around the circle is counter-clockwise. All regions are mutually exclusive. Parameter *w* refers to the width of the scene.

three geometric properties: pixel overlap, the angle between regions, and the distance between regions. Table 1 summarizes the relations used in our experiments most of which encode spatial object relations in the *x*-*y* space; the last three encode relative object position along the *z* axis. Our relations are broadly similar to those proposed in Elliott and Keller (2013) with the exception of *beside* which we break down into more fine-grained relations (namely *near* and *close*). We also add the *same* relation in the *z* axis. In cases where object *X* is facing object *Y* we subscript relations *opposite*, *near*, and *close* with the letter $\mathcal{F}$.

The procedure described above will generate a visual description for each linguistic description. It also assumes that visual relations hold between pairs of objects. The assumption is not unwarranted, 73.87% of the descriptions in the dataset involve only two arguments. The parallel sentences corresponding to Figure 2 are illustrated in Table 2. In cases where the the original description involves three objects, ternary relations are decomposed into binary ones. We create as many visual representations as there are linguistic descriptions. If two humans generate identical descriptions, we produce identical visual encodings. In total, we were able to create 46,053 parallel descriptions[2] accounting for 79.5% of the sentences in the dataset.

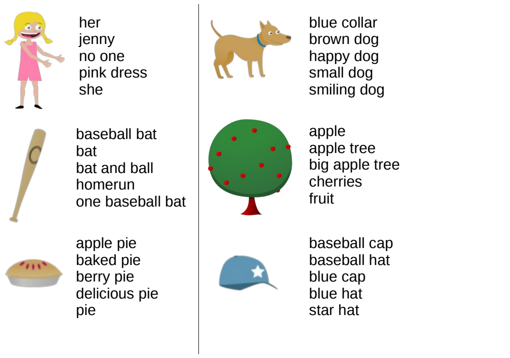


Figure 3: Examples of argument-clipart object pairs with high MI values (shown in descending order).

4.2 Generation Model

It is straightforward to train a phrase-based SMT model on the parallel corpus outlined above. The model would learn to translate a visual description (see the source side in Table 2) into natural language. However, when generating linguistic descriptions for a scene at test time, we must first decide "what to say" (content selection) and then "how to say" it (surface realization). What is the most important content in the scene worth describing? Given that visual relations between objects are assumed to be binary (see the VDG grammar in Table 1), there are $n(n-1)$ combinations of pairs of objects in a scene (where n is the number of clipart pieces available) and as many corresponding visual expressions. However, many of these visual expressions will capture unimportant aspects of the scene, or even express atypical relations unattested in the training data. We develop below a content selection component based on the intuition that frequently

[2] Our parallel corpus can be downloaded from `http://homepages.inf.ed.ac.uk/mlap/index.php?page=resources`.

Image	description
1. hb0.10s.png $close_{\mathcal{F}}$ *same* t.4s.png	Mike isn't sharing the soccer ball
2. hb0.10s.png *surrounds same* c.9s.png	Mike is wearing sunglasses
3. hb1.19s.png *below same* c.5s.png	Jenny is wearing a silly hat
4. hb0.10s.png $close_{\mathcal{F}}$ *same* t.4s.png	Mike is kicking the soccer ball
5. hb1.19s.png $close_{\mathcal{F}}$ *same* hb0.10s.png	Jenny is chasing Mike
6. hb1.19s.png *below same* c.5s.png	Jenny is wearing a silly hat

Table 2: Parallel corpus of visual expressions and linguistic descriptions corresponding to Figure 2.

mentioned object pairs probably express important scene content. In addition, it is reasonable to assume that the selected objects will be in close proximity, especially when actions are involved. One would expect the agent of the action to be near the object or person receiving it (e.g., *Mike is kicking the ball*, *Jenny is holding Mike's hand*). The same is true for instruments, which are typically held by the persons using them (e.g., *Jenny is digging with a shovel*).

Content Selection We cast the problem of finding suitable objects to talk about as an integer linear program (ILP). Our model selects clip art object pairs that best describe the content of a scene and ranks them based on their relevance. Indicator variable y_{stk} denotes whether two objects are being selected and how they are ranked (e.g, whether they should be mentioned first or last):

$$y_{stk} = \begin{cases} 1 & \text{if objects } s \text{ and } t \text{ are selected for rank } k \\ & \text{and } s \text{ is before } t \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

where s and t index two clip art objects and $k = 1, .., S$ encodes their rank (based on relevance). Our objective function is given below:

$$Z = \sum_{s \in S, t \in S} F_{st} \cdot D_{st} \cdot \sum_{k \in S} ((card(S) + 1) - k) \cdot y_{stk} \quad (3)$$

where F_{st} quantifies the normalized co-occurrence frequency of objects s and t (in the training set) and D_{st} specifies their relative distance. The term $((card(S) + 1) - k)$ accounts for the ranking of pairs so that most relevant ones are ranked first. Here, $card(S)$ represents the cardinality of the set S denoting the number of clip art objects in the scene; k ranges over all available ranks (which is limited by the number of clip art objects available). The term $((card(S) + 1) - k)$ is inversely proportional to k, so its highest value is when k is 1. In other words, the value of the term is maximum when the selected objects are ranked first. This way, we ensure that most relevant object pairs are given high ranks.

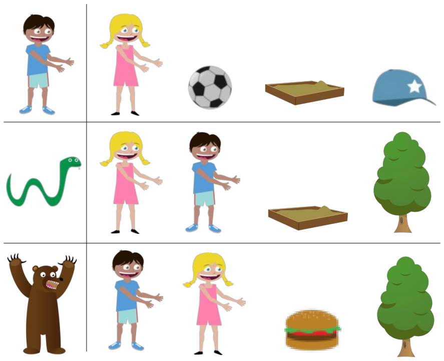


Figure 4: Example of three clip art objects and the most frequent objects they co-occur with.

We compute F_{st} from our parallel corpus (see left-hand side in Table 2), simply by counting the number of times objects s and t co-occur. Figure 4 shows three clip art objects (Mike, a snake, and a bear) and their most frequent co-occurrences. We estimate term D_{st}, the distance between objects s and t, using function $\sqrt{\Delta x^2 + \Delta y^2 + \Delta z^2}$. Coordinate z has only three possible values (see Table 1). To increase the effect of Δz, we use a scaling factor. We normalize and invert D_{st} so that it ranges from 0 to 1. In addition, we transform it with a sigmoid function so as to maximize the effect of near and distant objects (distances of relatively close objects are set to 1 and distances of distant objects are set to 0).

The objective function in Equation (3) is too permissive, allowing repetitions of the same object within a pair and of the object pairs themselves. Constraints (4)–(10) avoid repetitions and ensure that the selected objects are varied with the aim of generating logically consistent descriptions. Constraint (4) avoids empty descriptions, by enforcing that at least one clip art object pair is selected. Constraint (5) ensures that an object cannot appear in the same pair twice, whereas constraint (6) requires that at most one pair can be selected for a given rank k. We also enforce the selection of different pairs of objects (constraint (7)) at contiguous ranks (constraint (8)).

$$\sum_{s \in S, t \in S} y_{st1} = 1 \tag{4}$$

$$\forall_{stk,s==t},\ y_{stk} = 0 \tag{5}$$

$$\forall_{k},\ \sum_{s \in S, t \in S} y_{stk} \leq 1; \tag{6}$$

$$\forall_{st},\ \sum_{k \in S} (y_{stk} + y_{tsk}) \leq 1 \tag{7}$$

$$\forall_{k=1,..,S-1},\ \sum_{s \in S, t \in S} y_{stk+1} \leq \sum_{s \in S, t \in S} y_{stk} \tag{8}$$

Finally, to instill some coherence in the descriptions, while avoiding overly repetitive discourse, we disallow objects to be selected more than four times:

$$\forall_{s},\ sum_s = \sum_{t \in S, k \in S} y_{stk} \tag{9}$$

$$\forall_{t},\ sum_t = \sum_{s \in S, k \in S} y_{stk} \tag{10}$$

$$\forall_{st,s==t},\ sum_s + sum_t \leq 4 \tag{11}$$

Auxiliary variables sum_s and sum_t represent the number of times objects s and t are selected to be the first and second object of a pair.

Given a new unseen scene, we obtain the F_{st} values for all pair-wise combinations of the objects in it and compute their distance D_{st}. We solve the ILP problem defined above and read the value of the variable y_{stk} which contains the selected clip art object pairs ranked by relevance. So, our model can in principle produce multiple descriptions for a given scene, highlighting potentially different aspects of the visually encoded information. We used GLPK[3] to maximize the objective function subject to the constraints introduced above.

Surface realization The ILP selects all description-worthy pairs of clip art objects for a scene. Using the rules presented in Table 1 we create visual encodings for them (see Table 2, source side), and finally translate them into natural language using a Phrase-based SMT engine (Koehn et al., 2003). Specifically, given a source visual expression $\mathbf{f}$, our aim is to find an equivalent target natural language description $\hat{\mathbf{e}}$ that maximizes the posterior probability:

$$\hat{\mathbf{e}} = \underset{\mathbf{e}}{\operatorname{argmax}} P(\mathbf{e}|\mathbf{f}) \tag{12}$$

Most recent SMT work models the posterior $P(\mathbf{e}|\mathbf{f})$ directly (Och and Ney, 2002) using a log-linear combination of several models where:

$$P(\mathbf{e}|\mathbf{f}) = \frac{\exp \sum_{k=1}^{K} \lambda_k h_k(\mathbf{f}, \mathbf{e})}{\sum_{e'} \exp \sum_{k=1}^{K} \lambda_k h_k(\mathbf{f}, \mathbf{e}')} \tag{13}$$

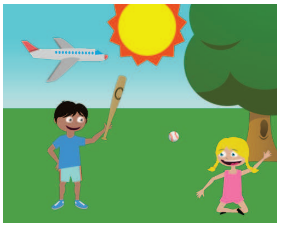
and the decision decision rule is given by:

$$\hat{\mathbf{e}} = \underset{\mathbf{e}}{\operatorname{argmax}} \sum_{k=1}^{K} \lambda_k h_k(\mathbf{f}, \mathbf{e}) \tag{14}$$

where $h_k(\mathbf{f}, \mathbf{e})$ is a scoring function representing important features for the translation of $\mathbf{f}$ into $\mathbf{e}$. Examples include the language model of the target language, a reordering model, or several translation models. K is the number of models (or features) and λ_k are the weights of the log-linear combination. Typically, the weights $\Lambda = [\lambda_1, \ldots, \lambda_K]^T$ are optimized on a development set, by means of Minimum Error Rate Training (MERT; Och (2003)).

One of the most popular instantiations of loglinear models in SMT are phrase-based (PB) models (Zens et al., 2002; Koehn et al., 2003). PB models allow to learn translations for entire phrases instead of individual words. The basic idea behind PB translation is to segment the source sentence into phrases, then to translate each source phrase into a target phrase, and finally reorder the translated target phrases in order to compose the target sentence. For this purpose, phrase-tables are produced, in which a source phrase is listed together with several target phrases and the probability of translating the former into the latter. Throughout our experiments, we obtained translation models using the PB SMT framework implemented in MOSES (Koehn et al., 2007).

[3] https://www.gnu.org/software/glpk/



Mike is kicking the ball	The plane is flying in the sky
`nsubj,aux,verb,det,dobj`	`det,nsubj,aux,verb,prep,det,pobj`

Table 3: Sample scenes with human-written descriptions and corresponding templates.

5 Model Comparison

We evaluated the model described above through comparison to four alternatives, representing different modeling paradigms in the literature. Our first comparison model is based on templates, which are commonly used to produce descriptions for images (Elliott and Keller, 2013; Kulkarni et al., 2011). Rather than manually creating template rules we induce them from dependency-parsed scene descriptions. We represent each description in the data as a sequence of typed dependencies. The scene descriptions are relatively simple, and many sentences have similar structure. Overall, 20 templates represent the syntactic structure of more than 44% of all scene descriptions. Examples of scenes, their descriptions, and corresponding templates are shown in Table 3 (template `nsubj,aux,verb,det,dobj` is the most frequent in the data).

We train a logistic regression classifier (Yu et al., 2011) on scene-template pairs, and learn to assign a template for a new unseen scene. The "template-predictor" uses variety of features based on the alignment between clip-art objects and POS-tags as well as objects and dependency roles. The alignments were computed using MI as described in Section 4.1. We also used visual features based on the absolute and relative distance between objects, their co-occurrence, spatial location, depth ordering, facial expression and poses (see Zitnick et al. (2013) for details). In order to transform the templates into natural language sentences we train a "word-predictor" which fills the most likely word for every grammatical function slot in a given template (again using logistic regression and the same feature space as for the template predictor). The word predictor uses a vocabulary of 70 frequently occurring words (attested no less than 150 times in the corpus). For a new scene, candidate templates are predicted and subsequently expanded to descriptions by predicting words for every function slot in the templates. Candidate descriptions are ranked using a trigram language model to ensure grammatical coherence.

We also implemented two sentence-based retrieval approaches. Our first system is conceptually similar to the model proposed by Farhadi et al. (2010). In their work, images and descriptions seen at training time are mapped into a shared meaning space M using a function f. Given an unseen image λ, the description closest to $f(\lambda)$ in M is retrieved and returned by the model. We used the word-predictor described above as a simple way of annotating an unseen scene λ with the words that most saliently describe it. These keywords then used as a TFIDF search query against the set H of human scene descriptions seen during training:

$$\begin{aligned}\mathrm{TFIDF}(q,d) &= \sum_{w\in q}\mathrm{TF}(w,d)\mathrm{IDF}(w),\\ \mathrm{TF}(w,q) &= \sqrt{\sum_{w'\in q}\mathbb{1}_{w=w'}},\\ \mathrm{IDF}(w) &= 1+\log\frac{\|H\|}{1+\sum\limits_{d\in H}\sum\limits_{w'\in d}\mathbb{1}_{w=w'}}.\end{aligned} \tag{15}$$

where H is the set of all human descriptions seen at training time, $\|\cdot\|$ is the set-norm, q is a search query, d is any description in H and $\mathbb{1}_{w=w'}$ an indicator variable set to 1 if w and w' are the same word and 0 otherwise. The human description maximizing the TFIDF similarity with the predicted keywords is returned as the description for the new scene.

Our third baseline exploits image similarity (Ordonez et al., 2011). Given an unseen scene λ, we retrieve from the training set λ', the scene most similar to it, and return one of λ''s human descriptions selected at random. We used locality sensitivity hashing to find the subset of candidate scenes similar to λ. Scenes were represented with the same visual features used for the word and template predictors and their similarity was computed with the cosine metric.

Finally, we also trained a multimodal log-bilinear model (Kiros et al., 2014) on the abstract scenes

System	BLEU	METEOR
LBL	7.33	17.76
MLBL	12.30	20.40
Image	12.80	21.77
Keyword	14.70	26.60
Template	40.30	30.40
SMT	43.70	35.60

Table 4: Model comparison on scene description task using automatic measures.

System	1^{st}	2^{nd}	3^{rd}	4^{th}	AvgRank
Keyword	0.24	0.13	0.22	0.41	2.22
Template	0.25	0.16	0.13	0.46	2.21
SMT	0.53	0.24	0.12	0.11	3.19
Human	0.57	0.27	0.12	0.04	3.36

Table 5: Rankings (shown as proportions) and mean ratings given to systems by human participants.

dataset. The model essentially implements a feed-forward neural network to predict the next word given the image and previous words.[4] Images were associated with feature representations obtained from the output of a convolutional network, following the feature learning procedure outlined in Kiros et al. (2014).

6 Results

We evaluated system output automatically using (smoothed) BLUE and METEOR as calculated by NIST's MultEval software[5] using the human-written descriptions as reference. Elliott and Keller (2014) find that both metrics correlate well with human judgments. For a fair comparison, we force our model to output one description, i.e., the most relevant one.

Our results are summarized in Table 4. As can be seen, our model (SMT) performs best both in terms of BLEU and METEOR. The template-based generator (Template) obtains competitive performance which is not surprising, it incorporates some of the ingredients of the SMT system such as word-to-clipart alignments, a language model, and is guaranteed to produce grammatical output. The performance of the multimodal log-bilinear model (MLBL) keyword- and image-based retrieval systems is inferior. We conjecture that the image features, and similarity functions used in these models are not fine-grained enough to capture the subtle differences in scenes which humans detect and express in the descriptions. Finally, notice that visual information is critical in doing well on the description generation task. A log-bilinear language model (LBL) trained solely on the descriptions performs poorly (see the top row in Table 4).

Resp	1^{st}	2^{nd}	3^{rd}	4^{th}	5^{th}	6^{th}
YES	75.5	65.8	53.0	44.7	44.0	37.5
NO	18.0	24.8	31.2	31.3	37.5	58.0
MAYBE	6.5	9.50	15.8	15.8	18.5	4.50

Table 6: Proportion of SMT descriptions deemed accurate and relevant. System output evaluated for rank placements 1…6.

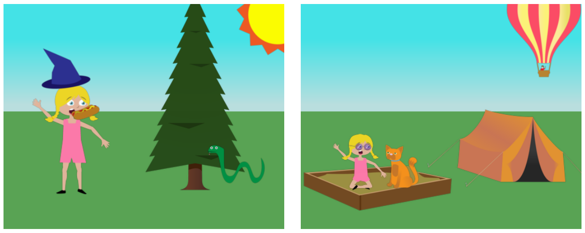
We further evaluated system output eliciting human judgments for 100 randomly selected test scenes. Participants were presented with a scene and descriptions generated by our system, the template-based model, the best-performing sentence retrieval model, and a randomly selected human description. Subjects were asked to rank the four descriptions from best to worst (ties are allowed) in order of informativeness (does the description capture what is shown in the scene?) and fluency (is the description written in well-formed English?). We elicited rankings using Amazon's Mechanical Turk crowdsourcing platform. Participants (self-reported native English speakers) saw 10 scenes per session. We collected 5 responses per item.

The results of our human evaluation study are shown in Table 5. Specifically, we show, proportionally, how often our participants ranked each system 1st, 2nd and so on. Perhaps unsurprisingly, the human-written descriptions were considered best (and ranked 1st 57% of the time). Our model is ranked best 0.53% of the time, followed by the template and keyword-based retrieval systems which are only ranked first 25% of the time.[6] We further

[4] We used the implementation at `http://www.cs.toronto.edu/~rkiros/multimodal.html`.

[5] `ftp://jaguar.ncsl.nist.gov/mt/resources/mteval-v13a-20091001.tar.gz`

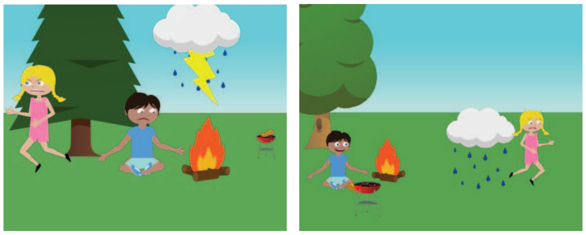
[6] Percentages do not sum to 100% because ties are allowed.



Jenny is holding a hot dog.
Jenny is wearing a witch hat.
Jenny is scared of the snake.
The snake is under the pine tree.

Jenny is sitting in the sandbox.
Jenny is wearing purple sunglasses.
The cat is sitting in the sandbox.
The cat is watching Jenny.

Figure 5: Examples of descriptions generated by the SMT model for two scenes.



“Mike and Jenny decide to make hot dogs on the grill.”, “It's a rainstorm and Jenny runs away to stay dry.”, “Mike stays beside the fire.”, “Jenny is standing next to the tree.”, “Mike is sitting next to the fire.”, “The hot dog is on the pit.”

Figure 6: Right scene is generated by SMT model (left scene is the original) given descriptions (bottom) as input.

converted the ranks to ratings on a scale of 1 to 4 (assigning ratings 4. . . 1 to rank placements 1. . . 4). This allowed us to perform Analysis of Variance (ANOVA) which revealed a reliable effect of system type. Specifically, post-hoc Tukey tests showed that our SMT model is significantly ($p < 0.01$) better than the other two comparison systems but does not differ significantly from the human goldstandard.

We also evaluated more thoroughly our content selection mechanism. Since our system can in principle generate multiple descriptions for a scene, we were interested to see how many of these are indeed relevant. We let the system generate the six best descriptions per scene and asked AMT participants to assess whether they were accurate (are the people, objects and actions mentioned in the description shown in the scene?) and appropriate (is the description relevant for the scene?). Participants answered with “yes”, “no”, or “maybe”. Again we used 100 items from the test set, and elicited 5 responses per item. Table 6 shows the outcome of this study. The majority of first-best descriptions (75.5%) returned by our system are perceived as relevant and scene appropriate. The same is true for 2nd and 3rd best descriptions, whereas the quality of descriptions deteriorates with lower ranks. This suggests that we could generate short discourses describing different viewpoints in a scene.

Figure 5 illustrates the descriptions produced by our model for two scenes, whereas Figure 6 shows example output when the system is run in reverse, i.e., it takes descriptions as input and generates a scene. This can be done straightforwardly, without any additional effort, however note that the model is unaware of the absolute position of objects, it places the cloud next to Jenny.

7 Conclusions

In this paper we presented proof of concept that an SMT-based approach is successful at generating human-like scene descriptions provided that (a) there is a large enough parallel corpus to learn from and (b) a content selection component identifies important scene content. Our results further indicate that instilling some degree of structural information in visual scenes (via the VDG) is beneficial. It allows to describe visual content more accurately and facilitates its rendering in natural language (since the two modalities are structurally similar). The template-based, retrieval, and language modeling systems do not use this structural information, and even though their descriptions are largely grammatical, they are not as felicitous. Our results also point to difficulty of the task. Even when computer vision is taken out of the equation, and the description language is simple, human-written text is still preferable (see Table 5). In the future, we would like to develop better content selection models (e.g., identify surprising aspects in a scene) and more accurate grounding strategies (e.g., via discriminative alignment).

Acknowledgments We are grateful to Lukas Dirzys for his help with the LBL and MLBL models. Special thanks to Frank Keller for his comments on an earlier version of this paper and Larry Zitnick whose talk at the UW MSR Summer Institute 2013 insipred this work.

References

Desmond Elliott and Frank Keller. 2013. Image description using visual dependency representations. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1292–1302, Seattle, Washington.

Desmond Elliott and Frank Keller. 2014. Comparing automatic evaluation measures for image description. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 452–457, Baltimore, Maryland.

Ali Farhadi, Mohsen Hejrati, Amin Sadeghi, Peter Yong, Cyrus Rashtchian, Julia Hockenmaier, and David Forsyth. 2010. Every picture tells a story: Generating sentences from images. In *Proceedings of the 11th European Conference on Computer Vision*, pages 25–29, Heraklion, Greece.

Yansong Feng and Mirella Lapata. 2013. Automatic caption generation for news images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(4):797–812.

David F. Fouhey and C. Lawrence Zitnick. 2014. Predicting object dynamics in scenes. In *Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Reconition*, pages 2027–2034, Columbus, Ohio.

Micah Hodosh, Peter Young, and Julia Hockenmaier. 2013. Framing image description as a ranking task: Data, models and evaluation metrics. *Journal of Artificial Intelligence Research*, 47:853–899.

Ryan Kiros, Ruslan Slakhutdinov, and Richard Zemel. 2014. Multimodal neural language models. In *Proceedings of the 31st International Conference on Machine Learning*, Beijing, China. volume 32.

Dan Klein and Christopher D. Manning. 2003. Accurate unlexicalized parsing. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 423–430, Sapporo, Japan.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Annual Meeting of the North American Chapter of the Association for Computational Linguistics*, pages 48–54, Edmonton, Canada.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondrej Bojar, Alexandra Constantin, and Evan Herbst. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics Companion Volume Proceedings of the Demo and Poster Sessions*, pages 177–180, Prague, Czech Republic.

Girish Kulkarni, Visruth Premraj, Sagnik Dhar, Siming Li, Yejin Choi, Alexander C. Berg, and Tamara L. Berg. 2011. Baby talk: Understanding and generating image descriptions. In *Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1601–1608, Colorado Springs, Colorado.

Polina Kuznetsova, Vicente Ordonez, Alexander Berg, Tamara Berg, and Yejin Choi. 2012. Collective generation of natural image descriptions. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 359–368, Jeju Island, Korea.

Polina Kuznetsova, Vicente Ordonez, Tamara L. Berg, and Yejin Choi. 2014. Treetalk: Composition and compression of trees for image descriptions. *Transactions of the Association for Computational Luingusitics*, 2:351–362.

Rebecca Mason and Eugene Charniak. 2014. Nonparametric method for data-driven image captioning. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 592–598, Baltimore, Maryland.

Margaret Mitchell, Jesse Dodge, Amit Goyal, Kota Yamaguchi, Karl Stratos, Xufeng Han, Alyssa Mensch, Alex Berg, Tamara Berg, and Hal Daume III. 2012. Midge: Generating image descriptions from computer vision detections. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 747–756, Avignon, France.

Franz Josef Och and Hermann Ney. 2002. Discriminative training and maximum entropy models for statistical machine translation. In *Proceedings of 40th Annual Meeting of the Association for Computational Linguistics*, pages 295–302, Philadelphia, Pennsylvania.

Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 160–167, Sapporo, Japan.

Vicente Ordonez, Girish Kulkarni, and Tamara L. Berg. 2011. Im2text: Describing images using 1 million captioned photographs. In J. Shawe-Taylor, R.S. Zemel, P. Bartlett, F.C.N. Pereira, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 24*, pages 1143–1151. Curran Associates, Inc.

Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. 2014. Show and tell: A neural image caption generator. arXiv:1411.4555.

Yezhou Yang, Ching Teo, Hal Daume III, and Yiannis Aloimonos. 2011. Corpus-guided sentence generation of natural images. In *Proceedings of the 2011*

Conference on Empirical Methods in Natural Language Processing, pages 444–454, Edinburgh, Scotland.

Hsiang-Fu Yu, Fang-Lan Huang, and Chih-Jen Lin. 2011. Dual coordinate descent methods for logistic regression and maximum entropy models. *Machine Learning*, 85(1-2):41–75.

R. Zens, F. J. Och, and H. Ney. 2002. Phrase-based statistical machine translation. In Springer Verlag, editor, *German Conference on Artificial Intelligence*, pages 18–32.

C. Lawrence Zitnick and Devi Parikh. 2013. Bringing semantics into focus using visual abstraction. In *Proceedings of the 2013 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3009–3016, Portland, Oregon.

C. Lawrewnce Zitnick, Devi Parikh, and Lucy Vanderwende. 2013. Learning the visual interpretation of sentences. In *Proceedings of the 2013 IEEE International Conference on Computer Vision*, pages 1681–1688, Sydney, Australia.