# TREETALK: Composition and Compression of Trees for Image Descriptions

**Polina Kuznetsova**[†] **Vicente Ordonez**[‡] **Tamara L. Berg**[‡] **Yejin Choi**[††]
† Stony Brook University, Stony Brook, NY, pkuznetsova@cs.stonybrook.edu
‡ UNC Chapel Hill, Chapel Hill, NC, {vicente,tlberg}@cs.unc.edu
††University of Washington, Seattle, WA, yejin@cs.washington.edu

## Abstract

We present a new tree based approach to composing expressive image descriptions that makes use of naturally occuring web images with captions. We investigate two related tasks: image caption *generalization* and *generation*, where the former is an optional subtask of the latter. The high-level idea of our approach is to harvest expressive phrases (as tree fragments) from existing image descriptions, then to compose a new description by selectively combining the extracted (and optionally pruned) tree fragments. Key algorithmic components are *tree composition* and *compression*, both integrating tree structure with sequence structure. Our proposed system attains significantly better performance than previous approaches for both image caption *generalization* and *generation*. In addition, our work is the first to show the empirical benefit of automatically generalized captions for composing natural image descriptions.

## 1 Introduction

The web is increasingly visual, with hundreds of billions of user contributed photographs hosted online. A substantial portion of these images have some sort of accompanying text, ranging from keywords, to free text on web pages, to textual descriptions directly describing depicted image content (i.e. captions). We tap into the last kind of text, using naturally occuring pairs of images with natural language descriptions to compose *expressive* descriptions for query images via tree composition and compression.

Such automatic image captioning efforts could potentially be useful for many applications: from automatic organization of photo collections, to facilitating image search with complex natural language queries, to enhancing web accessibility for the visually impaired. On the intellectual side, by learning to describe the visual world from naturally existing web data, our study extends the domains of language grounding to the highly expressive language that people use in their everyday online activities.

There has been a recent spike in efforts to automatically describe visual content in natural language (Yang et al., 2011; Kulkarni et al., 2011; Li et al., 2011; Farhadi et al., 2010; Krishnamoorthy et al., 2013; Elliott and Keller, 2013; Yu and Siskind, 2013; Socher et al., 2014). This reflects the long standing understanding that encoding the complexities and subtleties of image content often requires more expressive language constructs than a set of tags. Now that visual recognition algorithms are beginning to produce reliable estimates of image content (Perronnin et al., 2012; Deng et al., 2012a; Deng et al., 2010; Krizhevsky et al., 2012), the time seems ripe to begin exploring higher level semantic tasks.

There have been two main complementary directions explored for automatic image captioning. The first focuses on describing exactly those items (e.g., objects, attributes) that are detected by vision recognition, which subsequently confines *what* should be described and *how* (Yao et al., 2010; Kulkarni et al., 2011; Kojima et al., 2002). Approaches in this direction could be ideal for various practical applications such as image description for the visually impaired. However, it is not clear whether the semantic expressiveness of these approaches can eventually scale up to the casual, but highly expressive language peo-



Transactions of the Association for Computational Linguistics, 2 (2014) 351–362. Action Editor: Hal Daume III.
Submitted 2/2014; Revised 5/2014; Published 10/2014. 

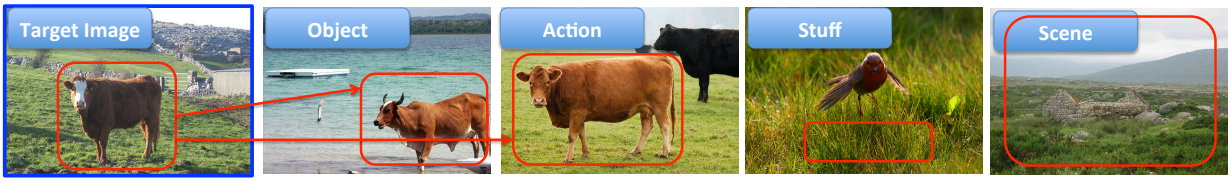


Figure 1: Harvesting phrases (as tree fragments) for the target image based on (partial) visual match.

ple naturally use in their online activities. In Figure 1, for example, it would be hard to compose *"I noticed that this funny cow was staring at me"* or *"You can see these beautiful hills only in the countryside"* in a purely bottom-up manner based on the exact content detected. The key technical bottleneck is that the range of describable content (i.e., objects, attributes, actions) is ultimately confined by the set of items that can be reliably recognized by state-of-the-art vision techniques.

The second direction, in a complementary avenue to the first, has explored ways to make use of the rich spectrum of visual descriptions contributed by online citizens (Kuznetsova et al., 2012; Feng and Lapata, 2013; Mason, 2013; Ordonez et al., 2011). In these approaches, *the set of what can be described* can be substantially larger than *the set of what can be recognized*, where the former is shaped and defined by the data, rather than by humans. This allows the resulting descriptions to be substantially more expressive, elaborate, and interesting than what would be possible in a purely bottom-up manner. Our work contributes to this second line of research.

One challenge in utilizing naturally existing multimodal data, however, is the noisy semantic alignment between images and text (Dodge et al., 2012; Berg et al., 2010). Therefore, we also investigate a related task of image caption *generalization* (Kuznetsova et al., 2013), which aims to improve the semantic image-text alignment by removing bits of text from existing captions that are less likely to be transferable to other images.

The high-level idea of our system is to harvest useful bits of text (as tree fragments) from existing image descriptions using detected visual content similarity, and then to compose a new description by selectively combining these extracted (and optionally pruned) tree fragments. This overall idea of *composition based on extracted phrases* is not new in itself (Kuznetsova et al., 2012), however, we make several technical and empirical contributions.

First, we propose a novel stochastic *tree composition* algorithm based on extracted tree fragments that integrates both tree structure and sequence cohesion into structural inference. Our algorithm permits a substantially higher level of linguistic expressiveness, flexibility, and creativity than those based on rules or templates (Kulkarni et al., 2011; Yang et al., 2011; Mitchell et al., 2012), while also addressing long-distance grammatical relations in a more principled way than those based on hand-coded constraints (Kuznetsova et al., 2012).

Second, we address image caption *generalization* as an optional subtask of image caption *generation*, and propose a *tree compression* algorithm that performs a light-weight parsing to search for the optimal set of tree branches to prune. Our work is the first to report empirical benefits of automatically compressed captions for image captioning.

The proposed approaches attain significantly better performance for both image caption *generalization* and *generation* tasks over competitive baselines and previous approaches. Our work results in an improved image caption corpus with automatic generalization, which is publicly available.[1]

## 2 Harvesting Tree Fragments

Given a query image, we retrieve images that are visually similar to the query image, then extract potentially useful segments (i.e., phrases) from their corresponding image descriptions. We then compose a new image description using these retrieved text fragments (§3). Extraction of useful phrases is guided by both visual similarity and the syntactic parse of the corresponding textual descrip-

[1] http://ilp-cky.appspot.com/

tion. This extraction strategy, originally proposed by Kuznetsova et al. (2012), attempts to make the best use of linguistic regularities with respect to objects, actions, and scenes, making it possible to obtain richer textual descriptions than what current state-of-the-art vision techniques can provide in isolation. In all of our experiments we use the captioned image corpus of Ordonez et al. (2011), first pre-processing the corpus for relevant content by running deformable part model object detectors (Felzenszwalb et al., 2010). For our study, we run detectors for 89 object classes set a high confidence threshold for detection.

As illustrated in Figure 1, for a query image detection, we extract four types of phrases (as tree fragments). First, we retrieve relevant noun phrases from images with visually similar object detections. We use color, texture (Leung and Malik, 1999), and shape (Dalal and Triggs, 2005; Lowe, 2004) based features encoded in a histogram of vector quantized responses to measure visual similarity. Second, we extract verb phrases for which the corresponding noun phrase takes the subject role. Third, from those images with *"stuff"* detections, e.g.*"water"*, or *"sky"* (typically mass nouns), we extract prepositional phrases based on similarity of both visual appearance and relative spatial relationships between detected objects and "stuff". Finally, we use global *"scene"* similarity[2] to extract prepositional phrases referring to the overall scene, e.g., *"at the conference,"* or *"in the market"*.

We perform this phrase retrieval process for each detected object in the query image and generate one sentence for each object. All sentences are then combined together to produce the final description. Optionally, we apply image caption generalization (via compression) (§4) to all captions in the corpus prior to the phrase extraction and composition.

## 3 Tree Composition

We model tree composition as constraint optimization. The input to our algorithm is the set of retrieved phrases (i.e., tree fragments), as illustrated in §2. Let $P = \{p_0, ..., p_{L-1}\}$ be the set of *all* phrases across the four phrase types (objects, actions, stuff and scene). We assume a mapping function $pt : [0, L) \rightarrow T$, where $T$ is the set of phrase *types*, so that the phrase type of $p_i$ is $pt(i)$. In addition, let $R$ be the set of PCFG production rules and $NT$ be the set of nonterminal symbols of the PCFG. The goal is to find and combine a good sequence of phrases $G$, $|G| \leq |T| = N = 4$, drawn from $P$, into a final sentence. More concretely, we want to select and order a subset of phrases (at most one phrase of each phrase type) while considering both the parse structure and n-gram cohesion across phrasal boundaries.

Figure 2 shows a simplified example of a composed sentence with its corresponding parse structure. For brevity, the figure shows only one phrase for each phrase type, but in actuality there would be a set of candidate phrases for each type. Figure 3 shows the CKY-style representation of the internal mechanics of constraint optimization for the example composition from Figure 2. Each cell $ij$ of the CKY matrix corresponds to $G_{ij}$, a subsequence of $G$ starting at position $i$ and ending at position $j$. If a cell in the CKY matrix is labeled with a nonterminal symbol $s$, it means that the corresponding tree of $G_{ij}$ has $s$ as its root.

Although we visualize the operation using a CKY-style representation in Figure 3, note that composition requires more complex combinatorial decisions than CKY parsing due to two additional considerations. We are: (1) *selecting* a subset of candidate phrases, and (2) *re-ordering* the selected phrases (hence making the problem NP-hard). Therefore, we encode our problem using Integer Linear Programming (ILP) (Roth and tau Yih, 2004; Clarke and Lapata, 2008) and use the CPLEX (ILOG, Inc, 2006) solver.

### 3.1 ILP Variables

**Variables for Sequence Structure:** Variables $\alpha$ encode phrase selection and ordering:

$$\alpha_{ik} = 1 \quad \text{iff} \quad \begin{array}{l}\text{phrase } i \in P \text{ is selected} \\ \text{for position } k \in [0, N)\end{array} \qquad (1)$$

Where $k$ is one of the N=4 positions in a sentence.[3] Additionally, we define variables for each pair of adjacent phrases to capture sequence cohesion:

[2] L2 distance between classification score vectors (Xiao et al., 2010)

[3] The number of positions is equal to the number of phrase types, since we select *at most* one from each type.

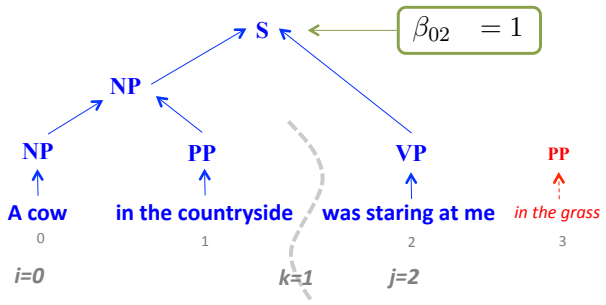


Figure 2: An example scenario of tree composition. Only the first three phrases are chosen for the composition.

$$\alpha_{ijk} = 1 \quad \text{iff} \quad \alpha_{ik} = \alpha_{j(k+1)} = 1 \tag{2}$$

**Variables for Tree Structure:** Variables $\beta$ encode the parse structure:

$$\beta_{ijs} = 1 \quad \text{iff} \quad \text{the phrase sequence } G_{ij} \text{ maps to the nonterminal symbol } s \in NT \tag{3}$$

Where $i \in [0, N)$ and $j \in [i, N)$ index rows and columns of the CKY-style matrix in Figure 3. A corresponding example tree is shown in Figure 2, where the phrase sequence $G_{02}$ corresponds to the cell labeled with $S$. We also define variables to indicate selected PCFG rules in the resulting parse:

$$\beta_{ijkr} = 1 \quad \text{iff} \quad \beta_{ijh} = \beta_{ikp} = \beta_{(k+1)jq} = 1, \tag{4}$$

Where $r = h \rightarrow pq \in R$ and $k \in [i, j)$. Index $k$ points to the boundary of split between two children as shown in Figure 2 for the sequence $G_{02}$.

**Auxiliary Variables:** For notational convenience, we also include:

$$\gamma_{ijk} = 1 \quad \text{iff} \quad \sum_{s \in NT} \beta_{ijs} = \sum_{s \in NT} \beta_{iks} = \sum_{s \in NT} \beta_{(k+1)js} = 1 \tag{5}$$

### 3.2 ILP Objective Function

We model tree composition as maximization of the following objective function:

$$F = \sum_i F_i \times \sum_{k=0}^{N-1} \alpha_{ik} + \sum_{ij} F_{ij} \times \sum_{k=0}^{N-2} \alpha_{ijk} + \sum_{ij} \sum_{k=i}^{j-1} \sum_{r \in R} F_r \times \beta_{ijkr} \tag{6}$$

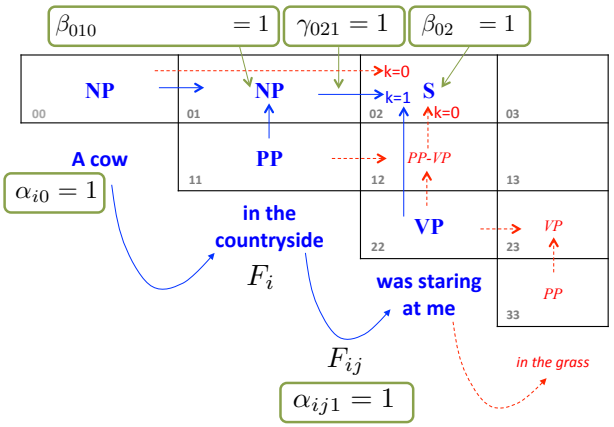


Figure 3: CKY-style representation of decision variables as defined in §3.1 for the tree example in Fig 2. Nonterminal symbols in boldface (in blue) and solid arrows (also in blue) represent the chosen PCFG rules to combine the selected set of phrases. Nonterminal symbols in smaller font (in red) and dotted arrows (also in red) represent possible other choices that are not selected.

This objective is comprised of three types of weights (confidence scores): $F_i, F_{ij}, F_r$.[4] $F_i$ represents the phrase selection score based on visual similarity, described in §2. $F_{ij}$ quantifies the sequence cohesion across phrase boundaries. For this, we use $n$-gram scores ($n \in [2, 5]$) between adjacent phrases computed using the Google Web 1-T corpus (Brants and Franz., 2006). Finally, $F_r$ quantifies PCFG rule scores (log probabilities) estimated from the 1M image caption corpus (Ordonez et al., 2011) parsed using the Stanford parser (Klein and Manning, 2003).

One can view $F_i$ as a *content selection* score, while $F_{ij}$ and $F_r$ correspond to *linguistic fluency* scores capturing sequence and tree structure respectively. If we set positive values for all of these weights, the optimization function would be biased toward verbose production, since selecting an additional phrase will increase the objective function. To control for verbosity, we set scores corresponding to linguistic fluency, i.e., $F_{ij}$ and $F_r$ using negative values (smaller absolute values for higher fluency), to balance dynamics between content selection and linguistic fluency.

### 3.3 ILP Constraints

**Soundness Constraints:** We need constraints to enforce consistency between different types of vari-

[4] All weights are normalized using z-score.

ables (Equations 2, 4, 5). Constraints for a product of two variables have been discussed by Clarke and Lapata (2008). For Equation 2, we add the following constraints (similar constraints are also added for Equations 4,5).

$$\forall_{ijk},\ \alpha_{ijk} \leq \alpha_{ik} \tag{7}$$
$$\alpha_{ijk} \leq \alpha_{j(k+1)}$$
$$\alpha_{ijk} + (1 - \alpha_{ik}) + (1 - \alpha_{j(k+1)}) \geq 1$$

**Consistency between Tree Leafs and Sequences:** The ordering of phrases implied by $\alpha_{ijk}$ must be consistent with the ordering of phrases implied by the $\beta$ variables. This can be achieved by aligning the leaf cells (i.e., $\beta_{kks}$) in the CKY-style matrix with $\alpha$ variables as follows:

$$\forall_{ik}, \alpha_{ik} \leq \sum_{s \in NT^i} \beta_{kks} \tag{8}$$

$$\forall_k, \sum_i \alpha_{ik} = \sum_{s \in NT} \beta_{kks} \tag{9}$$

Where $NT^i$ refers to the set of PCFG nonterminals that are compatible with a phrase type $pt(i)$ of $p_i$. For example, $NT^i = \{\text{NN,NP}, ...\}$ if $p_i$ corresponds to an "object" (noun-phrase). Thus, Equation 8 enforces the correspondence between phrase types and nonterminal symbols at the tree leafs. Equation 9 enforces the constraint that the number of selected phrases and instantiated tree leafs must be the same.

**Tree Congruence Constraints:** To ensure that each CKY cell has at most one symbol we require

$$\forall_{ij},\ \sum_{s \in NT} \beta_{ijs} \leq 1 \tag{10}$$

We also require that

$$\forall_{i,j>i,h},\ \beta_{ijh} = \sum_{k=i}^{j-1} \sum_{r \in R_h} \beta_{ijkr} \tag{11}$$

Where $R_h = \{r \in R : r = h \rightarrow pq\}$. We enforce these constraints only for non-leafs. This constraint forbids instantiations where a nonterminal symbol $h$ is selected for cell $ij$ without selecting a corresponding PCFG rule.

We also ensure that we produce a valid tree structure. For instance, if we select 3 phrases as shown in Figure 3, we must have the root of the tree at the corresponding cell 02.

$$\forall_{k \in [1,N)},\ \sum_{s \in NT} \beta_{kks} \leq \sum_{t=k}^{N-1} \sum_{s \in NT} \beta_{0ts} \tag{12}$$

We also require cells that are not selected for the resulting parse structure to be empty:

$$\forall ij \sum_k \gamma_{ijk} \leq 1 \tag{13}$$

Additionally, we penalize solutions without the $S$ tag at the parse root as a soft-constraint.

**Miscellaneous Constraints:** Finally, we include several constraints to avoid degenerate solutions or to otherwise enhance the composed output. We: (1) enforce that a noun-phrase is selected (to ensure semantic relevance to the image content), (2) allow at most one phrase of each type, (3) do not allow multiple phrases with identical headwords (to avoid redundancy), (4) allow at most one scene phrase for all sentences in the description. We find that handling of sentence boundaries is important if the ILP formulation is based only on sequence structure, but with the integration of tree-based structure, we do not need to specifically handle sentence boundaries.

### 3.4 Discussion

An interesting aspect of description generation explored in this paper is using tree fragments as the building blocks of composition rather than individual words. There are three practical benefits: (1) *syntactic and semantic expressiveness*, (2) *correctness*, and (3) *computational efficiency*. Because we extract phrases from human written captions, we are able to use expressive language, and less likely to make syntactic or semantic errors. Our phrase extraction process can be viewed at a high level as visually-grounded or visually-situated paraphrasing. Also, because the unit of operation is tree fragments, the ILP formulation encoded in this work is computationally lightweight. If the unit of composition was words, the ILP instances would be significantly more computationally intensive, and more likely to suffer from grammatical and semantic errors.

## 4 Tree Compression

As noted by recent studies (Mason and Charniak, 2013; Kuznetsova et al., 2013; Jamieson et al., 2010), naturally existing image captions often include contextual information that does not directly describe visual content, which ultimately hinders their usefulness for describing other images. Therefore, to improve the fidelity of the generated descriptions, we explore image caption generalization as an

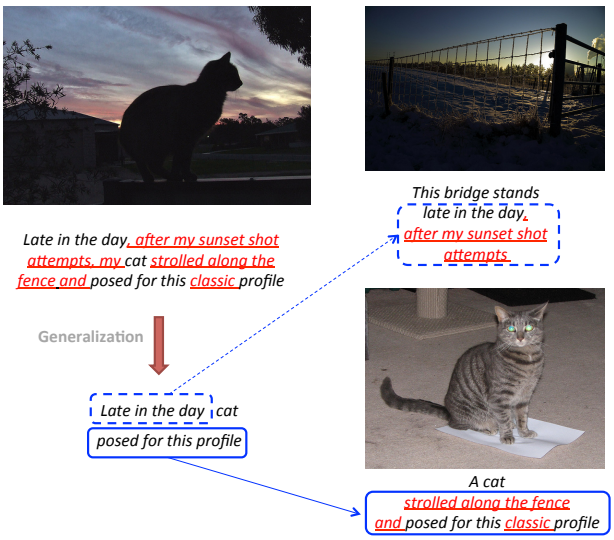


Figure 4: Compressed captions (on the left) are more applicable for describing new images (on the right).

optional pre-processing step. Figure 4 illustrates a concrete example of image caption generalization in the context of image caption generation.

We cast caption generalization as sentence compression. We encode the problem as tree pruning via lightweight CKY parsing, while also incorporating several other considerations such as leaf-level ngram cohesion scores and visually informed content selection. Figure 5 shows an example compression, and Figure 6 shows the corresponding CKY matrix.

At a high level, the compression operation resembles bottom-up CKY parsing, but in addition to parsing, we also consider deletion of parts of the trees. When deleting parts of the original tree, we might need to re-parse the remainder of the tree. Note that we consider re-parsing only with respect to the original parse tree produced by a state-of-the-art parser, hence it is only a *light-weight* parsing.[5]

### 4.1 Dynamic Programming

Input to the algorithm is a sentence, represented as a vector $\mathbf{x} = x_0 ... x_{n-1} = x[0 : n-1]$, and its PCFG parse $\pi(\mathbf{x})$ obtained from the Stanford parser. For simplicity of notation, we assume that both the parse tree and the word sequence are encoded in $\mathbf{x}$. Then, the compression can be formalized as:

[5]Integrating full parsing into the original sentence would be a straightforward extension conceptually, but may not be an empirically better choice when parsing for compression is based on vanilla unlexicalized parsing.

$$\hat{\mathbf{y}} = \arg\max_{\mathbf{y}} \prod_i \phi_i(\mathbf{x}, \mathbf{y}) \tag{14}$$

Where each $\phi_i$ is a potential function, corresponding to a criteria of the desired compression:

$$\phi_i(\mathbf{x}, \mathbf{y}) = \exp(\theta_i \cdot f_i(\mathbf{x}, \mathbf{y})) \tag{15}$$

Where $\theta_i$ is the weight for a particular criteria (described in §4.2), whose scoring function is $f_i$.

We solve the decoding problem (Equation 14) using dynamic programming. For this, we need to solve the compression sub-problems for sequences $x[i : j]$, which can be viewed as branches $\hat{y}[i, j]$ of the final tree $\hat{y}[0 : n-1]$. For example, in Figure 5, the final solution is $\hat{y}[0 : 7]$, while a sub-solution of $x[4 : 7]$ corresponds to a tree branch $PP$. Notice that sub-solution $\hat{y}[3 : 7]$ represents the same branch as $\hat{y}[4 : 7]$ due to branch deletion. Some computed sub-solutions, e.g., $\hat{y}[1 : 4]$, get dropped from the final compressed tree.

We define a matrix of scores $D[i, j, h]$ (Equation 17), where $h$ is one of the nonterminal symbols being considered for a cell indexed by $i, j$, i.e. a candidate for the root symbol of a branch $\hat{y}[i : j]$. When all values $D[i, j, h]$ are computed, we take

$$\hat{h} = \arg\max_{h} D[0, n-1, h] \tag{16}$$

and backtrack to reconstruct the final compression (the exact solution to equation 14).

$$D[i,j,h] = \max_{\substack{k \in [i,j) \\ r \in R_h}} \begin{cases} (1) & D[i,k,p] + D[k+1,j,q] \\ & \quad +\Delta\phi[r, ij] \\ (2) & D[i,k,p] + \Delta\phi[r, ij] \\ (3) & D[k+1,j,p] + \Delta\phi[r, ij] \end{cases} \tag{17}$$

Where $R_h = \{r \in R : r = h \rightarrow pq \vee r = h \rightarrow p\}$. Index $k$ determines a split point for child branches of a subtree $\hat{y}[i : j]$. For example, in the Figure 5 the split point for children of the subtree $\hat{y}[0 : 7]$ is $k = 2$. The three cases ((1) – (3)) of the above equation correspond to the following tree pruning cases:

**Pruning Case (1):** None of the children of the current node is deleted. For example, in Figures 5 and 6, the PCFG rule $PP \rightarrow IN\ PP$, corresponding to the sequence *"in black and white"*, is retained. Another situation that can be encountered is tree re-parsing.

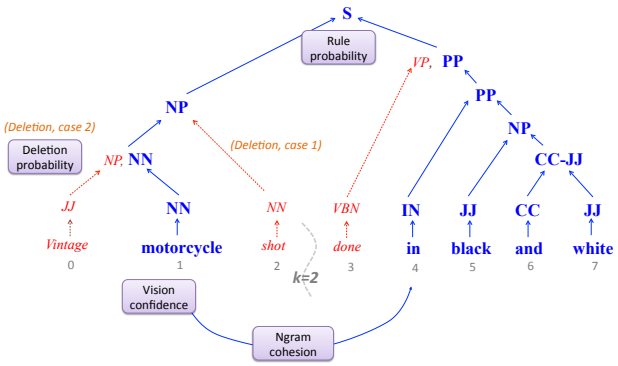


Figure 5: CKY compression. Both the chosen rules and phrases (blue bold font and blue solid arrows) and not chosen rules and phrases (red italic smaller font and red dashed lines) are shown.

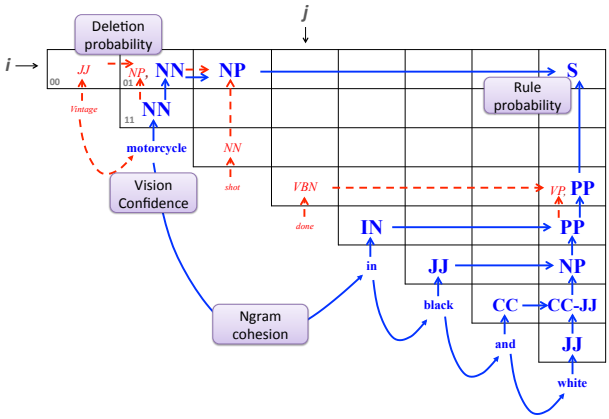


Figure 6: CKY compression. Both the chosen rules and phrases (blue bold font and blue solid arrows) and not chosen rules and phrases (red italic smaller font and red dashed lines) are shown.

**Pruning Case (2)/(3):** Deletion of the left/right child respectively. There are two types of deletion, as illustrated in Figures 5 and 6. The first corresponds to deletion of a child node. For example, the second child $NN$ of rule $NP \rightarrow NP\,NN$ is deleted, which yields deletion of *"shot"*. The second type is a special case of propagating a node to a higher-level of the tree. In Figure 6, this situation occurs when deleting $JJ$ "Vintage", which causes the propagation of $NN$ from cell 11 to cell 01. For this purpose, we expand the set of rules $R$ with additional special rules of the form $h \rightarrow h$, e.g., $NN \rightarrow NN$, which allows propagation of tree nodes to higher levels of the compressed tree.[6]

### 4.2 Modeling Compression Criteria

The $\Delta\phi$ term[7] in Equation 17 denotes the sum of log of potential functions for each criteria $q$:

$$\Delta\phi[r, ij] = \sum_{q} \theta \cdot \Delta f_q(r, ij) \qquad (18)$$

Note that $\Delta\phi$ depends on the current rule $r$, along with the historical information before the current step $ij$, such as the original rule $r_{ij}$, and ngrams on the border between left and right child branches of rule $r_{ij}$. We use the following four criteria $f_q$ in our model, which are demonstrated in Figures 5 and 6.

**I. Tree Structure:** We capture PCFG rule probabilities estimated from the corpus as $\Delta f_{pcfg} = \log P_{pcfg}(r)$.

**II. Sequence Structure:** We incorporate ngram cohesion scores only across the border between two branches of a subtree.

**III. Branch Deletion Probabilities:** We compute probabilities of deletion for children as:

$$\Delta f_{del} = \log P(r_t | r_{ij}) = \log \frac{count(r_t, r_{ij})}{count(r_{ij})} \qquad (19)$$

Where $count(r_t, r_{ij})$ is the frequency in which $r_{ij}$ is transformed to $r_t$ by deletion of one of the children. We estimate this probability from a training corpus, described in §4.3. $count(r_{ij})$ is the count of $r_{ij}$ in uncompressed sentences.

**IV. Vision Detection (Content Selection):** We want to keep words referring to actual objects in the image. Thus, we use $V(x_j)$, a visual similarity score, as our confidence of an object corresponding to word $x_j$. This similarity is obtained from the visual recognition predictions of (Deng et al., 2012b).

Note that some test instances include rules that we have not observed during training. We default to the original caption in those cases. The weights $\theta_i$ are set using a tuning dataset. We control over-compression by setting the weight for $f_{del}$ to a small value relative to the other weights.

### 4.3 Human Compressed Captions

Although we model image caption generalization as sentence compression, in practical applications we may want the outputs of these two tasks to be different. For example, there may be differences in what should be deleted (named entities in newswire summaries could be important to keep, while they may

[6] We assign probabilities of these special propagation rules to 1 so that they will not affect the final parse tree score. Turner and Charniak (2005) handled propagation cases similarly.

[7] We use $\Delta$ to distinguish the potential value for the whole sentence from the gain of the potential during a single step of the algorithm.

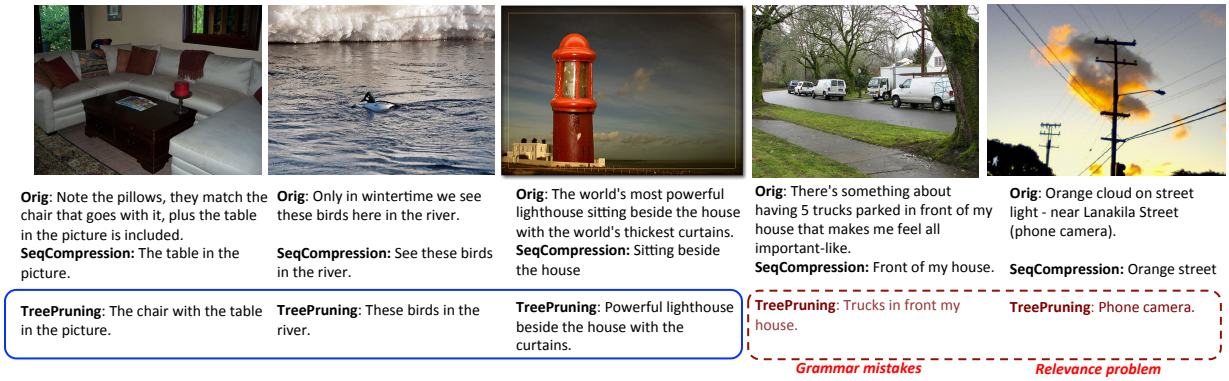


Figure 7: Caption generalization: good/bad examples.

be extraneous for image caption generalization). To learn the syntactic patterns for caption generalization, we collect a small set of example compressed captions (380 in total) using Amazon Mechanical Turk (AMT) (Snow et al., 2008). For each image, we asked 3 turkers to first list all visible objects in an image and then to write a compressed caption by removing not visually verifiable bits of text. We then align the original and compressed captions to measure rule deletion probabilities, excluding misalignments, similar to Knight and Marcu (2000). Note that we remove this dataset from the 1M caption corpus when we perform description generation.

## 5 Experiments

We use the 1M captioned image corpus of Ordonez et al. (2011). We reserve 1K images as a test set, and use the rest of the corpus for phrase extraction. We experiment with the following approaches:

**Proposed Approaches:**

- TREEPRUNING: Our tree compression approach as described in §4.
- SEQ+TREE: Our tree composition approach as described in §3.
- SEQ+TREE+PRUNING: SEQ+TREE using compressed captions of TREEPRUNING as building blocks.

**Baselines for Composition:**

- SEQ+LINGRULE: The most equivalent to the older sequence-driven system (Kuznetsova et al., 2012). Uses a few minor enhancements, such as sentence-boundary statistics, to improve grammaticality.
- SEQ: The §3 system without tree models and mentioned enhancements of SEQ+LINGRULE.

| Method | Bleu w/ (w/o) penalty | Meteor P | Meteor R | Meteor M |
|---|---|---|---|---|
| SEQ+LINGRULE | 0.152 (0.152) | 0.13 | 0.17 | 0.095 |
| SEQ | 0.138 (0.138) | 0.12 | **0.18** | 0.094 |
| SEQ+TREE | 0.149 (0.149) | 0.13 | 0.14 | 0.082 |
| SEQ+PRUNING | **0.177** (0.177) | 0.15 | 0.16 | **0.101** |
| SEQ+TREE+PRUNING | 0.140 (**0.189**) | **0.16** | 0.12 | 0.088 |

Table 1: Automatic Evaluation

- SEQ+PRUNING: SEQ using compressed captions of TREEPRUNING as building blocks.

We also experiment with the compression of human written captions, which are used to generate image descriptions for the new target images.

**Baselines for Compression:**

- SEQCOMPRESSION (Kuznetsova et al., 2013): Inference operates over the sequence structure. Although optimization is subject to constraints derived from dependency parse, parsing is not an explicit part of the inference structure. Example outputs are shown in Figure 7.

### 5.1 Automatic Evaluation

We perform automatic evaluation using two measures widely used in machine translation: BLEU (Papineni et al., 2002)[8] and METEOR (Denkowski and Lavie, 2011).[9] We remove all punctuation and convert captions to lower case. We use 1K test images from the captioned image corpus,[10] and assume the original captions as the gold standard captions to compare against. The results in Table 1

[8] We use the unigram NIST implementation: ftp://jaguar.ncsl.nist.gov/mt/resources/mteval-v13a-20091001.tar.gz

[9] With equal weight between precision and recall in Table 1.

[10] Except for those for which image URLs are broken, or CPLEX did not return a solution.

| Method-1 | Method-2 | Criteria | Method-1 preferred over Method-2 (%) | | |
|---|---|---|---|---|---|
| | | | all turkers | turkers w/ $\kappa > 0.55$ | turkers w/ $\kappa > 0.6$ |
| *Image Description Generation* | | | | | |
| SEQ+TREE | SEQ | Rel | 72 | 72 | 72 |
| SEQ+TREE | SEQ | Gmar | 83 | 83 | 83 |
| SEQ+TREE | SEQ | All | 68 | 69 | 66 |
| SEQ+TREE+PRUNING | SEQ+TREE | Rel | 68 | 72 | 72 |
| SEQ+TREE+PRUNING | SEQ+TREE | Gmar | 41 | 38 | 41 |
| SEQ+TREE+PRUNING | SEQ+TREE | All | 63 | 64 | 66 |
| SEQ+TREE | SEQ+LINGRULE | All | 62 | 64 | 62 |
| SEQ+TREE+PRUNING | SEQ+LINGRULE | All | 67 | 75 | 77 |
| SEQ+TREE+PRUNING | SEQ+PRUNING | All | 73 | 75 | 75 |
| SEQ+TREE+PRUNING | HUMAN | All | 24 | 19 | 19 |
| *Image Caption Generalization* | | | | | |
| TREEPRUNING | SEQCOMPRESSION* | Rel | 65 | 65 | 66 |

Table 2: Human Evaluation: posed as a binary question *"which of the two options is better?"* with respect to *Relevance* (Rel), *Grammar* (Gmar), and *Overall* (All). According to Pearson's $\chi^2$ test, all results are statistically significant.

show that both the integration of the tree structure (+TREE) and the generalization of captions using tree compression (+PRUNING) improve the BLEU score without brevity penalty significantly,[11] while improving METEOR only moderately (due to an improvement on precision with a decrease in recall.)

## 5.2 Human Evaluation

Neither BLEU nor METEOR directly measure grammatical correctness over long distances and may not correspond perfectly to human judgments. Therefore, we supplement automatic evaluation with human evaluation. For human evaluations, we present two options generated from two competing systems, and ask turkers to choose the one that is better with respect to: *relevance, grammar*, and *overall*. Results are shown in Table 2 with 3 turker ratings per image. We filter out turkers based on a control question. We then compute the selection rate (%) of preferring method-1 over method-2. The agreement among turkers is a frequent concern. Therefore, we vary the set of dependable users based on their Cohen's kappa score ($\kappa$) against other users. It turns out, filtering users based on $\kappa$ does not make a big difference in determining the winning method.

As expected, tree-based systems significantly outperform sequence-based counterparts. For example,

[11]While 4-gram BLEU with brevity penalty is found to correlate better with human judges by recent studies (Elliott and Keller, 2014), we found that this is not the case for our task. This may be due to the differences in the gold standard captions. We use naturally existing ones, which include a wider range of content and style than crowd-sourced captions.

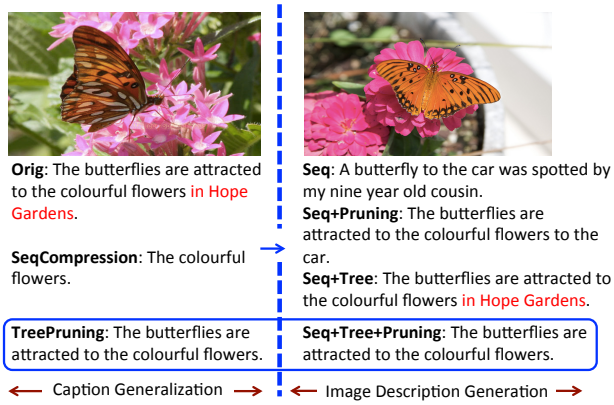


Figure 8: An example of a description preferred over human gold standard. Image description is improved due to caption generalization.

SEQ+TREE is strongly preferred over SEQ, with a selection rate of 83%. Somewhat surprisingly, improved grammaticality also seems to improve relevance scores (72%), possibly because it is harder to appreciate the semantic relevance of automatic captions when they are less comprehensible. Also as expected, compositions based on pruned tree fragments significantly improve relevance (68–72%), while slightly deteriorating grammar (38–41%).

Notably, the captions generated by our system are preferred over the original (owner generated) captions 19–24% of the time. One such example is included in Figure 8: *"The butterflies are attracted to the colorful flowers."*

Additional examples (good and bad) are provided in Figures 9 and 10. Many of these captions are highly expressive while remaining semantically

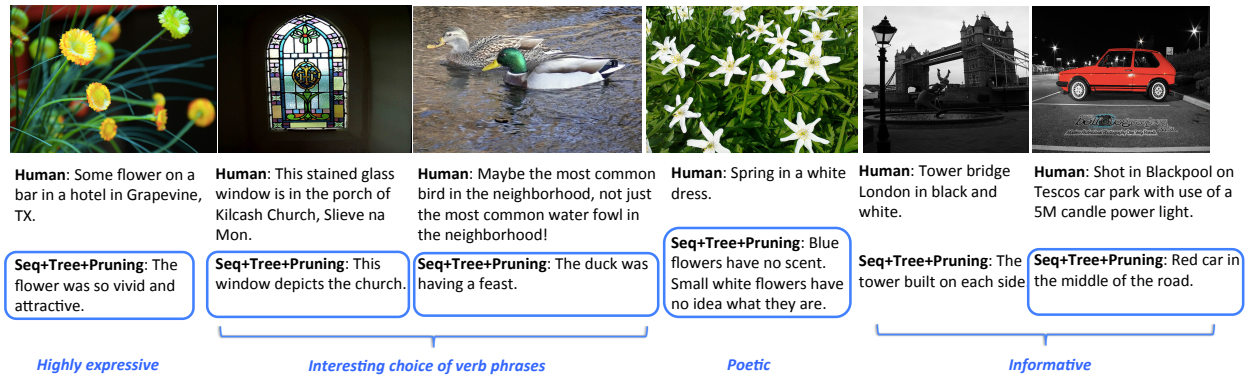


Figure 9: Description generation: good examples. Description preferred over human gold standard are highlighted.


Figure 10: Description generation: bad examples.

plausible, thanks to the expressive, but somewhat predictable descriptions online users write about their photos. Even among the bad examples (Figure 10) one can find highly creative captions with not literal but metaphorical relevance: *"Monarch in her bedroom before the wedding ceremony"*.[12] The complete system captions and the original captions are available at `http://ilp-cky.appspot.com/`

## 6 Related Work

**Sentence Fusion** Sentence fusion has been studied mostly for multi-document summarization (Barzilay and McKeown, 2005), where redundancy across multiple sentences serves as a guideline for syntactic and semantic validity of generation. In contrast, we do not have the natural redundancy to rely upon in our task, therefore requiring the composition algorithm to be intrinsically better constrained for correct sentence structures.

[12] "Monarch" can be a type of butterfly.

**Sentence Compression** At the core of the image caption generalization task is sentence compression. Much work has considered deletion-only edits like ours (Knight and Marcu, 2000; Turner and Charniak, 2005; Cohn and Lapata, 2007; Filippova and Altun, 2013), while recent ones explore more complex edits, such as substitutions, insertions and reordering (Cohn and Lapata, 2008). The latter generally requires a larger training corpus. We leave more expressive compression as a future research work.

## 7 Conclusion

In this paper, we have presented a novel tree composition approach for generating expressive image descriptions. As an optional preprocessing step, we also presented a tree compression approach and reported the empirical benefit of using automatically compressed captions to improve image description generation. By integrating both the tree structure and the sequence structure, we have significantly improved the quality of composed image captions over several competitive baselines.

## References


Regina Barzilay and Kathleen McKeown. 2005. Sentence fusion for multidocument news summarization. *Computational Linguistics*, 31(3):297–328.

Tamara L. Berg, Alexander C. Berg, and Jonathan Shih. 2010. Automatic attribute discovery and characterization from noisy web data. In *Proceedings of the 11th European Conference on Computer Vision: Part I*, ECCV'10, pages 663–676, Berlin, Heidelberg. Springer-Verlag.

Thorsten Brants and Alex Franz. 2006. Web 1t 5-gram version 1. In *Linguistic Data Consortium*.

James Clarke and Mirella Lapata. 2008. Global inference for sentence compression an integer linear programming approach. *Journal of Artificial Intelligence Research*, 31:399–429.

Trevor Cohn and Mirella Lapata. 2007. Large margin synchronous generation and its application to sentence compression. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 73–82, Prague, Czech Republic, June. Association for Computational Linguistics.

Trevor Cohn and Mirella Lapata. 2008. Sentence compression beyond word deletion. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pages 137–144, Manchester, UK, August. Coling 2008 Organizing Committee.

Navneet Dalal and Bill Triggs. 2005. Histograms of oriented gradients for human detection. In *Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05) - Volume 1 - Volume 01*, CVPR '05, pages 886–893, Washington, DC, USA. IEEE Computer Society.

Jia Deng, Alexander C. Berg, Kai Li, and Fei-Fei Li. 2010. What does classifying more than 10,000 image categories tell us? In *ECCV*.

Jia Deng, Alexander C. Berg, Sanjeev Satheesh, Hao Su, Aditya Khosla, and Fei-Fei Li. 2012a. Large scale visual recognition challenge. In *http://www.image-net.org/challenges/LSVRC/2012/index*.

Jia Deng, Jonathan Krause, Alexander C. Berg, and L. Fei-Fei. 2012b. Hedging your bets: Optimizing accuracy-specificity trade-offs in large scale visual recognition. In *Conference on Computer Vision and Pattern Recognition*.

Michael Denkowski and Alon Lavie. 2011. Meteor 1.3: Automatic Metric for Reliable Optimization and Evaluation of Machine Translation Systems. In *Proceedings of the EMNLP 2011 Workshop on Statistical Machine Translation*.

Jesse Dodge, Amit Goyal, Xufeng Han, Alyssa Mensch, Margaret Mitchell, Karl Stratos, Kota Yamaguchi, Yejin Choi, Hal Daume III, Alexander C. Berg, and Tamara L. Berg. 2012. Detecting visual text. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 762–772, Montréal, Canada, June. Association for Computational Linguistics.

Desmond Elliott and Frank Keller. 2013. Image description using visual dependency representations. In *EMNLP*, pages 1292–1302.

Desmond Elliott and Frank Keller. 2014. Comparing automatic evaluation measures for image description. In *ACL (2)*, pages 452–457.

Ali Farhadi, Mohsen Hejrati, Mohammad Amin Sadeghi, Peter Young1, Cyrus Rashtchian, Julia Hockenmaier, and David Forsyth. 2010. Every picture tells a story: generating sentences for images. In *European Conference on Computer Vision*.

Pedro F. Felzenszwalb, Ross B. Girshick, David McAllester, and Deva Ramanan. 2010. Object detection with discriminatively trained part based models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(9):1627–1645.

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

Katja Filippova and Yasemin Altun. 2013. Overcoming the lack of parallel data in sentence compression. In *EMNLP*, pages 1481–1491.

ILOG, Inc. 2006. ILOG CPLEX: High-performance software for mathematical programming and optimization. See `http://www.ilog.com/products/cplex/`.

Michael Jamieson, Afsaneh Fazly, Suzanne Stevenson, Sven J. Dickinson, and Sven Wachsmuth. 2010. Using language to learn structured appearance models for image annotation. *IEEE Trans. Pattern Anal. Mach. Intell.*, 32(1):148–164.

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

Kevin Knight and Daniel Marcu. 2000. Statistics-based summarization - step one: Sentence compression. In *AAAI/IAAI*, pages 703–710.

Atsuhiro Kojima, Takeshi Tamura, and Kunio Fukunaga. 2002. Natural language description of human activities from video images based on concept hierarchy of actions. *IJCV*, 50.

Niveda Krishnamoorthy, Girish Malkarnenkar, Raymond J. Mooney, Kate Saenko, and Sergio Guadarrama. 2013. Generating natural-language video descriptions using text-mined knowledge. In *AAAI*.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *NIPS*.

Girish Kulkarni, Visruth Premraj, Sagnik Dhar, Siming Li, Yejin Choi, Alexander C. Berg, and Tamara L. Berg. 2011. BabyTalk: Understanding and generating simple image descriptions. In *Conference on Computer Vision and Pattern Recognition*.

Polina Kuznetsova, Vicente Ordonez, Alexander C. Berg, Tamara L. 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, July. Association for Computational Linguistics.

Polina Kuznetsova, Vicente Ordonez, Alexander C. Berg, Tamara L. Berg, and Yejin Choi. 2013. Generalizing image captions for image-text parallel corpus. In *The 51st Annual Meeting of the Association for Computational Linguistics - Short Papers*, Sofia, Bulgaria, August. Association for Computational Linguistics.

Thomas K. Leung and Jitendra Malik. 1999. Recognizing surfaces using three-dimensional textons. In *ICCV*, pages 1010–1017.

Siming Li, Girish Kulkarni, Tamara L. Berg, Alexander C. Berg, and Yejin Choi. 2011. Composing simple image descriptions using web-scale n-grams. In *Proceedings of the Fifteenth Conference on Computational Natural Language Learning*, pages 220–228, Portland, Oregon, USA, June. Association for Computational Linguistics.

David G. Lowe. 2004. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60:91–110, November.

Rebecca Mason and Eugene Charniak. 2013. Annotation of online shopping images without labeled training examples. In *Proceedings of Workshop on Vision and Language*, Atlanta, Georgia, June. Association for Computational Linguistics.

Rebecca Mason. 2013. Domain-independent captioning of domain-specific images. In *Proceedings of the 2013 NAACL HLT Student Research Workshop*, pages 69–76, Atlanta, Georgia, June. Association for Computational Linguistics.

Margaret Mitchell, Jesse Dodge, Amit Goyal, Kota Yamaguchi, Karl Stratos, Xufeng Han, Alyssa Mensch, Alexander C. Berg, Tamara L. Berg, and Hal Daumé III. 2012. Midge: Generating image descriptions from computer vision detections. In *EACL*, pages 747–756.

Vicente Ordonez, Girish Kulkarni, and Tamara L. Berg. 2011. Im2text: Describing images using 1 million captioned photographs. In *Neural Information Processing Systems (NIPS)*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *ACL*.

Florent Perronnin, Zeynep Akata, Zaid Harchaoui, and Cordelia Schmid. 2012. Towards good practice in large-scale learning for image classification. In *CVPR*.

Dan Roth and Wen tau Yih. 2004. A linear programming formulation for global inference in natural language tasks. In *Proc. of the Annual Conference on Computational Natural Language Learning (CoNLL)*.

Rion Snow, Brendan O'Connor, Daniel Jurafsky, and Andrew Y. Ng. 2008. Cheap and fast—but is it good?: Evaluating non-expert annotations for natural language tasks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '08, pages 254–263, Stroudsburg, PA, USA. Association for Computational Linguistics.

Richard Socher, Andrej Karpathy, Quoc V. Le, Christopher D. Manning, and Andrew Y. Ng. 2014. Grounded compositional semantics for finding and describing images with sentences. In *Transactions of the Association for Computational Linguistics*, pages 207 – 218, April.

Jenine Turner and Eugene Charniak. 2005. Supervised and unsupervised learning for sentence compression. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 290–297, Ann Arbor, Michigan, June. Association for Computational Linguistics.

Jianxiong Xiao, James Hays, Krista A. Ehinger, Aude Oliva, and Antonio Torralba. 2010. Sun database: Large-scale scene recognition from abbey to zoo. In *CVPR*.

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, UK., July. Association for Computational Linguistics.

Benjamin Z. Yao, Xiong Yang, Liang Lin, Mun Wai Lee, and Song-Chun Zhu. 2010. I2T: Image parsing to text description. *Proc. IEEE*, 98(8).

Haonan Yu and Jeffrey Mark Siskind. 2013. Grounded language learning from video described with sentences. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 53–63, Sofia, Bulgaria, August. Association for Computational Linguistics.