Midge: Generating Image Descriptions From Computer Vision Detections

Margaret Mitchell[†] **Jesse Dodge**[‡‡] **Amit Goyal**[††] **Kota Yamaguchi**[§] **Karl Stratos**[‖]
Xufeng Han[§] **Alyssa Mensch**[**] **Alex Berg**[§] **Tamara Berg**[§] **Hal Daumé III**[††]
[†] U. of Aberdeen and Oregon Health and Science University, m.mitchell@abdn.ac.uk
[§] Stony Brook University, {aberg,tlberg,xufhan,kyamagu}@cs.stonybrook.edu
[††] U. of Maryland, {hal,amit}@umiacs.umd.edu
[‖] Columbia University, stratos@cs.columbia.edu
[‡‡] U. of Washington, dodgejesse@gmail.com, [**] MIT, acmensch@mit.edu

Abstract

This paper introduces a novel generation system that composes humanlike descriptions of images from computer vision detections. By leveraging syntactically informed word co-occurrence statistics, the generator filters and constrains the noisy detections output from a vision system to generate syntactic trees that detail what the computer vision system sees. Results show that the generation system outperforms state-of-the-art systems, automatically generating some of the most natural image descriptions to date.

1 Introduction

It is becoming a real possibility for intelligent systems to talk about the visual world. New ways of mapping computer vision to generated language have emerged in the past few years, with a focus on pairing detections in an image to words (Farhadi et al., 2010; Li et al., 2011; Kulkarni et al., 2011; Yang et al., 2011). The goal in connecting vision to language has varied: systems have started producing language that is descriptive and poetic (Li et al., 2011), summaries that add content where the computer vision system does not (Yang et al., 2011), and captions copied directly from other images that are globally (Farhadi et al., 2010) and locally similar (Ordonez et al., 2011).

A commonality between all of these approaches is that they aim to produce natural-sounding descriptions from computer vision detections. This commonality is our starting point: We aim to design a system capable of producing natural-sounding descriptions from computer vision detections *that are flexible enough* to become more descriptive and poetic, or include likely information from a language model, or to be short and simple, but as true to the image as possible.

The bus by the road with a clear blue sky

Figure 1: Example image with generated description.

Rather than using a fixed template capable of generating one kind of utterance, our approach therefore lies in generating syntactic trees. We use a tree-generating process (Section 4.3) similar to a Tree Substitution Grammar, but preserving some of the idiosyncrasies of the Penn Treebank syntax (Marcus et al., 1995) on which most statistical parsers are developed. This allows us to automatically parse and train on an unlimited amount of text, creating data-driven models that flesh out descriptions around detected objects in a principled way, based on what is both likely and syntactically well-formed.

An example generated description is given in Figure 1, and example vision output/natural language generation (NLG) input is given in Figure 2. The system ("Midge") generates descriptions in present-tense, declarative phrases, as a naïve viewer without prior knowledge of the photograph's content.[1]

Midge is built using the following approach: An image processed by computer vision algorithms can be characterized as a triple $<A_i, B_i, C_i>$, where:

[1] Midge is available to try online at: http://recognition.cs.stonybrook.edu:8080/~mitchema/midge/.

Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics, pages 747–756, Avignon, France, April 23 - 27 2012.

stuff:	*sky*	.999	
	id:	1	
	atts:	clear:0.432,	blue:0.945
		grey:0.853,	white:0.501 ...
	b. box:	(1,1 440,141)	
stuff:	*road*	.908	
	id:	2	
	atts:	wooden:0.722	clear:0.020 ...
	b. box:	(1,236 188,94)	
object:	*bus*	.307	
	id:	3	
	atts:	black:0.872,	red:0.244 ...
	b. box:	(38,38 366,293)	
preps:	id 1, id 2: by	id 1, id 3: by	id 2, id 3: below

Figure 2: Example computer vision output and natural language generation input. Values correspond to scores from the vision detections.

- A_i is the set of object/stuff detections with bounding boxes and associated "attribute" detections within those bounding boxes.
- B_i is the set of action or pose detections associated to each $a_i \in A_i$.
- C_i is the set of spatial relationships that hold between the bounding boxes of each pair $a_i, a_j \in A_i$.

Similarly, a description of an image can be characterized as a triple $<A_d, B_d, C_d>$ where:

- A_d is the set of nouns in the description with associated modifiers.
- B_d is the set of verbs associated to each $a_d \in A_d$.
- C_d is the set of prepositions that hold between each pair of $a_d, a_e \in A_d$.

With this representation, mapping $<A_i, B_i, C_i>$ to $<A_d, B_d, C_d>$ is trivial. The problem then becomes: (1) How to filter out detections that are wrong; (2) how to order the objects so that they are mentioned in a natural way; (3) how to connect these ordered objects within a syntactically/semantically well-formed tree; and (4) how to add further descriptive information from language modeling alone, if required.

Our solution lies in using A_i and A_d as description anchors. In computer vision, object detections form the basis of action/pose, attribute, and spatial relationship detections; therefore, in our approach to language generation, nouns for the object detections are used as the basis for the description. Likelihood estimates of syntactic structure and word co-occurrence are conditioned on object nouns, and this enables each noun head in a description to select for the kinds of structures it tends to appear in (syntactic constraints) and the other words it tends to occur with (semantic constraints). This is a data-driven way to generate likely adjectives, prepositions, determiners, etc., taking the intersection of what the vision system predicts and how the object noun tends to be described.

2 Background

Our approach to describing images starts with a system from Kulkarni et al. (2011) that composes novel captions for images in the PASCAL sentence data set,[2] introduced in Rashtchian et al. (2010). This provides multiple object detections based on Felzenszwalb's mixtures of multi-scale deformable parts models (Felzenszwalb et al., 2008), and stuff detections (roughly, mass nouns, things like sky and grass) based on linear SVMs for low level region features.

Appearance characteristics are predicted using trained detectors for colors, shapes, textures, and materials, an idea originally introduced in Farhadi et al. (2009). Local texture, Histograms of Oriented Gradients (HOG) (Dalal and Triggs, 2005), edge, and color descriptors inside the bounding box of a recognized object are binned into histograms for a vision system to learn to recognize when an object is rectangular, wooden, metal, etc. Finally, simple preposition functions are used to compute the spatial relations between objects based on their bounding boxes.

The original Kulkarni et al. (2011) system generates descriptions with a template, filling in slots by combining computer vision outputs with text based statistics in a conditional random field to predict the most likely image labeling. Template-based generation is also used in the recent Yang et al. (2011) system, which fills in likely verbs and prepositions by dependency parsing the human-written UIUC Pascal-VOC dataset (Farhadi et al., 2010) and selecting the dependent/head relation with the highest log likelihood ratio.

Template-based generation is useful for automatically generating consistent sentences, however, if the goal is to vary or add to the text produced, it may be suboptimal (cf. Reiter and Dale (1997)). Work that does not use template-based generation includes Yao et al. (2010), who generate syntactic trees, similar to the approach in this

[2] http://vision.cs.uiuc.edu/pascal-sentences/

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

Yang et al.: Three people are showing the bottle on the street

Midge: people with a bottle at the table

Kulkarni et al.: This is a picture of two pottedplants, one dog and one person. The black dog is by the black person, and near the second feathered pottedplant.

Yang et al.: The person is sitting in the chair in the room

Midge: a person in black with a black dog by potted plants

Figure 3: Descriptions generated by Midge, Kulkarni et al. (2011) and Yang et al. (2011) on the same images. Midge uses the Kulkarni et al. (2011) front-end, and so outputs are directly comparable.

paper. However, their system is not automatic, requiring extensive hand-coded semantic and syntactic details. Another approach is provided in Li et al. (2011), who use image detections to select and combine web-scale n-grams (Brants and Franz, 2006). This automatically generates descriptions that are either poetic or strange (e.g., "tree snowing black train").

A different line of work transfers captions of similar images directly to a query image. Farhadi et al. (2010) use <object,action,scene> triples predicted from the visual characteristics of the image to find potential captions. Ordonez et al. (2011) use global image matching with local reordering from a much larger set of captioned photographs. These transfer-based approaches result in natural captions (they are written by humans) that may not actually be true of the image.

This work learns and builds from these approaches. Following Kulkarni et al. and Li et al., the system uses large-scale text corpora to estimate likely words around object detections. Following Yang et al., the system can hallucinate likely words using word co-occurrence statistics alone. And following Yao et al., the system aims

black, blue, brown, colorful, golden, gray, green, orange, pink, red, silver, white, yellow, bare, clear, cute, dirty, feathered, flying, furry, pine, plastic, rectangular, rusty, shiny, spotted, striped, wooden

Table 1: Modifiers used to extract training corpus.

for naturally varied but well-formed text, generating syntactic trees rather than filling in a template.

In addition to these tasks, Midge automatically decides what the subject and objects of the description will be, leverages the collected word co-occurrence statistics to filter possible incorrect detections, and offers the flexibility to be as descriptive or as terse as possible, specified by the user at run-time. The end result is a fully automatic vision-to-language system that is beginning to generate syntactically and semantically well-formed descriptions with naturalistic variation. Example descriptions are given in Figures 4 and 5, and descriptions from other recent systems are given in Figure 3.

The results are promising, but it is important to note that Midge is a first-pass system through the steps necessary to connect vision to language at a deep syntactic/semantic level. As such, it uses basic solutions at each stage of the process, which may be improved: Midge serves as an illustration of the types of issues that should be handled to automatically generate syntactic trees from vision detections, and offers some possible solutions. It is evaluated against the Kulkarni et al. system, the Yang et al. system, and human-written descriptions on the same set of images in Section 5, and is found to significantly outperform the automatic systems.

3 Learning from Descriptive Text

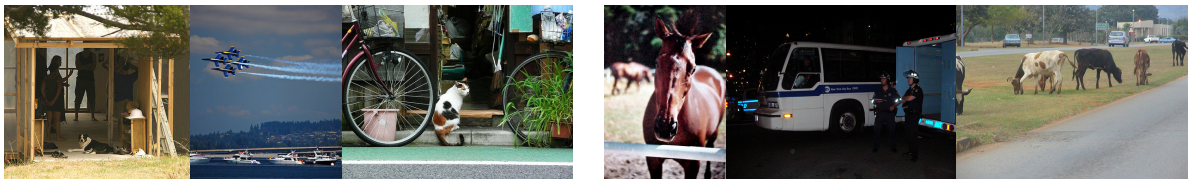
To train our system on how people describe images, we use 700,000 (Flickr, 2011) images with associated descriptions from the dataset in Ordonez et al. (2011). This is separate from our evaluation image set, consisting of 840 PASCAL images. The Flickr data is messier than datasets created specifically for vision training, but provides the largest corpus of natural descriptions of images to date.

We normalize the text by removing emoticons and mark-up language, and parse each caption using the Berkeley parser (Petrov, 2010). Once parsed, we can extract syntactic information for individual (word, tag) pairs.

a cow with green grass | sheep with a gray sky by the road | people with boats | a brown cow | people at a wooden table

Figure 4: Example generated outputs.

Awkward Prepositions | **Incorrect Detections**



a person *on* the dog | boats *under* the sky | a black bicycle *at* a green potted plant | *the sky* | a *yellow* bus with people | cows by black *sheep* by the road

Figure 5: Example generated outputs: Not quite right

We compute the probabilities for different prenominal modifiers (*shiny*, *clear*, *glowing*, ...) and determiners (*a/an*, *the*, `None`, ...) given a head noun in a noun phrase (NP), as well as the probabilities for each head noun in larger constructions, listed in Section 4.3. Probabilities are conditioned only on open-class words, specifically, nouns and verbs. This means that a closed-class word (such as a preposition) is never used to generate an open-class word.

In addition to co-occurrence statistics, the parsed Flickr data adds to our understanding of the basic characteristics of visually descriptive text. Using WordNet (Miller, 1995) to automatically determine whether a head noun is a physical object or not, we find that 92% of the sentences have no more than 3 physical objects. This informs generation by placing a cap on how many objects are mentioned in each descriptive sentence: When more than 3 objects are detected, the system splits the description over several sentences. We also find that many of the descriptions are not sentences as well (tagged as S, 58% of the data), but quite commonly noun phrases (tagged as NP, 28% of the data), and expect that the number of noun phrases that form descriptions will be much higher with domain adaptation. This also informs generation, and the system is capable of generating both sentences (contains a main verb) and noun phrases (no main verb) in the final image description. We use the term 'sentence' in the rest of this paper to refer to both kinds of complex phrases.

4 Generation

Following Penn Treebank parsing guidelines (Marcus et al., 1995), the relationship between two head nouns in a sentence can usually be characterized among the following:

1. prepositional (a boy *on* the table)
2. verbal (a boy *cleans* the table)
3. verb with preposition (a boy *sits on* the table)
4. verb with particle (a boy *cleans up* the table)
5. verb with S or SBAR complement (a boy sees *that* the table is clean)

The generation system focuses on the first three kinds of relationships, which capture a wide range of utterances. The process of generation is approached as a problem of generating a semantically and syntactically well-formed tree based on object nouns. These serve as head noun anchors in a lexicalized syntactic derivation process that we call *tree growth*.

Vision detections are associated to a {tag word} pair, and the model fleshes out the tree details around head noun anchors by utilizing syntactic dependencies between words learned from the Flickr data discussed in Section 3. The analogy of growing a tree is quite appropriate here, where nouns are bundles of constraints akin to seeds, giving rise to the rest of the tree based on the lexicalized subtrees in which the nouns are likely to occur. An example generated tree structure is shown in Figure 6, with noun anchors in bold.

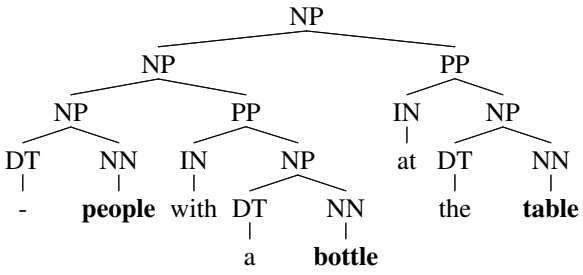


Figure 6: Tree generated from tree growth process.

Midge was developed using detections run on Flickr images, incorporating action/pose detections for verbs as well as object detections for nouns. In testing, we generate descriptions for the PASCAL images, which have been used in earlier work on the vision-to-language connection (Kulkarni et al., 2011; Yang et al., 2011), and allows us to compare systems directly. Action and pose detection for this data set still does not work well, and so the system does not receive these detections from the vision front-end. However, the system can still generate verbs when action and pose detectors have been run, and this framework allows the system to "hallucinate" likely verbal constructions between objects if specified at runtime. A similar approach was taken in Yang et al. (2011). Some examples are given in Figure 7.

We follow a three-tiered generation process (Reiter and Dale, 2000), utilizing *content determination* to first cluster and order the object nouns, create their local subtrees, and filter incorrect detections; *microplanning* to construct full syntactic trees around the noun clusters, and *surface realization* to order selected modifiers, realize them as postnominal or prenominal, and select final outputs. The system follows an overgenerate-and-select approach (Langkilde and Knight, 1998), which allows different final trees to be selected with different settings.

4.1 Knowledge Base

Midge uses a knowledge base that stores models for different tasks during generation. These models are primarily data-driven, but we also include a hand-built component to handle a small set of rules. The data-driven component provides the syntactically informed word co-occurrence statistics learned from the Flickr data, a model for ordering the selected nouns in a sentence, and a model to change computer vision attributes to attribute:value pairs. Below, we discuss the three main data-driven models within the generation pipeline. The hand-built component contains plural forms of singular nouns, the list of possible spatial relations shown in Table 3, and a mapping between attribute values and modifier surface forms (e.g., a *green* detection for *person* is to be realized as the postnominal modifier *in green*).

Unordered		Ordered
bottle, *table*, *person*	→	*person*, *bottle*, *table*
road, *sky*, *cow*	→	*cow*, *road*, *sky*

Figure 8: Example nominal orderings.

4.2 Content Determination

4.2.1 Step 1: Group the Nouns

An initial set of object detections must first be split into clusters that give rise to different sentences. If more than 3 objects are detected in the image, the system begins splitting these into different noun groups. In future work, we aim to compare principled approaches to this task, e.g., using mutual information to cluster similar nouns together. The current system randomizes which nouns appear in the same group.

4.2.2 Step 2: Order the Nouns

Each group of nouns are then ordered to determine when they are mentioned in a sentence. Because the system generates declarative sentences, this automatically determines the subject and objects. This is a novel contribution for a general problem in NLG, and initial evaluation (Section 5) suggests it works reasonably well.

To build the nominal ordering model, we use WordNet to associate all head nouns in the Flickr data to all of their hypernyms. A description is represented as an ordered set $[a_1...a_n]$ where each a_p is a noun with position p in the set of head nouns in the sentence. For the position p_i of each hypernym h_a in each sentence with n head nouns, we estimate $\mathrm{p}(p_i|n, h_a)$.

During generation, the system greedily maximizes $\mathrm{p}(p_i|n, h_a)$ until all nouns have been ordered. Example orderings are shown in Figure 8. This model automatically places animate objects near the beginning of a sentence, which follows psycholinguistic work in object naming (Branigan et al., 2007).

4.2.3 Step 3: Filter Incorrect Attributes

For the system to be able to extend coverage as new computer vision attribute detections become available, we develop a method to automatically


A person sitting on a sofa


Cows grazing


Airplanes flying


A person walking a dog

Figure 7: Hallucinating: Creating likely actions. Straightforward to do, but can often be wrong.

COLOR	purple blue green red white ...
MATERIAL	plastic wooden silver ...
SURFACE	furry fluffy hard soft ...
QUALITY	shiny rust dirty broken ...

Table 2: Example attribute classes and values.

group adjectives into broader attribute classes,[3] and the generation system uses these classes when deciding how to describe objects. To group adjectives, we use a bootstrapping technique (Kozareva et al., 2008) that learns which adjectives tend to co-occur, and groups these together to form an attribute class. Co-occurrence is computed using cosine (distributional) similarity between adjectives, considering adjacent nouns as context (i.e., JJ NN constructions). Contexts (nouns) for adjectives are weighted using Pointwise Mutual Information and only the top 1000 nouns are selected for every adjective. Some of the learned attribute classes are given in Table 2.

In the Flickr corpus, we find that each attribute (COLOR, SIZE, etc.), rarely has more than a single value in the final description, with the most common (COLOR) co-occurring less than 2% of the time. Midge enforces this idea to select the most likely word v for each attribute from the detections. In a noun phrase headed by an object noun, NP{NN noun}, the prenominal adjective (JJ v) for each attribute is selected using maximum likelihood.

4.2.4 Step 4: Group Plurals

How to generate natural-sounding spatial relations and modifiers for a set of objects, as opposed to a single object, is still an open problem (Funakoshi et al., 2004; Gatt, 2006). In this work, we use a simple method to group all same-type objects together, associate them to the plural form listed in the KB, discard the modifiers, and return spatial relations based on the first recognized member of the group.

4.2.5 Step 5: Gather Local Subtrees Around Object Nouns

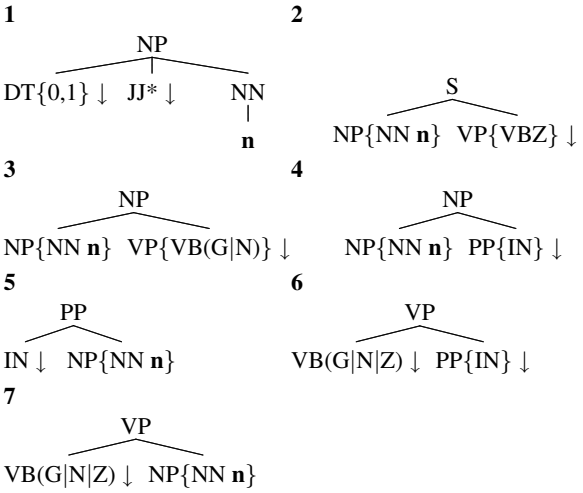


Figure 9: Initial subtree frames for generation, present-tense declarative phrases. ↓ marks a substitution site, * marks ≥ 0 sister nodes of this type permitted, {0,1} marks that this node can be included of excluded. **Input:** set of ordered nouns, **Output:** trees preserving nominal ordering.

Possible actions/poses and spatial relationships between objects nouns, represented by verbs and prepositions, are selected using the subtree frames listed in Figure 9. Each head noun selects for its likely local subtrees, some of which are not fully formed until the Microplanning stage. As an example of how this process works, see Figure 10, which illustrates the combination of Trees 4 and 5. For simplicity, we do not include the selection of further subtrees. The subject noun *duck* selects for prepositional phrases headed by different prepositions, and the object noun *grass* selects for prepositions that head the prepositional phrase in which it is embedded. Full PP subtrees are created during Microplanning by taking the intersection of both.

The leftmost noun in the sequence is given a rightward directionality constraint, placing it as the subject of the sentence, and so it will only se-

[3] What in computer vision are called *attributes* are called *values* in NLG. A value like *red* belongs to a COLOR attribute, and we use this distinction in the system.

a over b	a above b b underneath a	b below a a upon b	b beneath a a over b	a by b	b by a	a on b	b under a
a by b	a against b b beside a	b against a a by b	b around a b by a	a around b a near b	a at b b near a	b at a b with a	a beside b a with b
a in b	a in b	b outside a	a within b	a by b	b by a		

Table 3: Possible prepositions from bounding boxes.

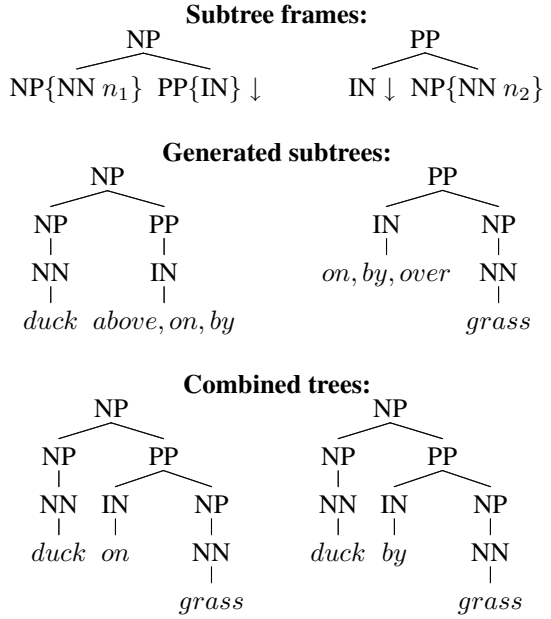


Figure 10: Example derivation.

lect for trees that expand to the right. The rightmost noun is given a leftward directionality constraint, placing it as an object, and so it will only select for trees that expand to its left. The noun in the middle, if there is one, selects for all its local subtrees, combining first with a noun to its right or to its left. We now walk through the derivation process for each of the listed subtree frames. Because we are following an overgenerate-and-select approach, all combinations above a probability threshold α and an observation cutoff γ are created.

Tree 1:
Collect all NP $\rightarrow$ (DT det) (JJ adj)* (NN $noun$) and NP $\rightarrow$ (JJ adj)* (NN $noun$) subtrees, where:

- p((JJ adj)|(NN $noun$)) $> \alpha$ for each adj
- p((DT det)|JJ, (NN $noun$)) $> \alpha$, and the probability of a determiner for the head noun is higher than the probability of no determiner.

Any number of adjectives (including none) may be generated, and we include the presence or absence of an adjective when calculating which determiner to include.

The reasoning behind the generation of these subtrees is to automatically learn whether to treat a given noun as a mass or count noun (not taking a determiner or taking a determiner, respectively) or as a given or new noun (phrases like *a sky* sound unnatural because *sky* is given knowledge, requiring the definite article *the*). The selection of determiner is not independent of the selection of adjective; *a sky* may sound unnatural, but *a blue sky* is fine. These trees take the dependency between determiner and adjective into account.

Trees 2 and 3:
Collect beginnings of VP subtrees headed by (VBZ *verb*), (VBG *verb*), and (VBN *verb*), notated here as VP{VBX *verb*}, where:

- p(VP{VBX $verb$}|NP{NN $noun$}=SUBJ) $> \alpha$

Tree 4:
Collect beginnings of PP subtrees headed by (IN $prep$), where:

- p(PP{IN $prep$}|NP{NN $noun$}=SUBJ) $> \alpha$

Tree 5:
Collect PP subtrees headed by (IN $prep$) with NP complements (OBJ) headed by (NN $noun$), where:

- p(PP{IN $prep$}|NP{NN $noun$}=OBJ) $> \alpha$

Tree 6:
Collect VP subtrees headed by (VBX $verb$) with embedded PP complements, where:

- p(PP{IN $prep$}|VP{VBX $verb$}=SUBJ) $> \alpha$

Tree 7:
Collect VP subtrees headed by (VBX $verb$) with embedded NP objects, where:

- p(VP{VBX $verb$}|NP{NN $noun$}=OBJ) $> \alpha$

4.3 Microplanning

4.3.1 Step 6: Create Full Trees

In Microplanning, full trees are created by taking the intersection of the subtrees created in Content Determination. Because the nouns are ordered, it is straightforward to combine the subtrees surrounding a noun in position 1 with subtrees surrounding a noun in position 2. Two

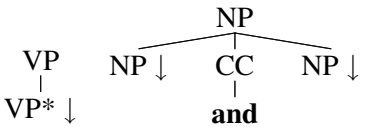


Figure 11: Auxiliary trees for generation.

further trees are necessary to allow the subtrees gathered to combine within the Penn Treebank syntax. These are given in Figure 11. If two nouns in a proposed sentence cannot be combined with prepositions or verbs, we backoff to combine them using (CC and).

Stepping through this process, all nouns will have a set of subtrees selected by Tree 1. Prepositional relationships between nouns are created by substituting Tree 1 subtrees into the NP nodes of Trees 4 and 5, as shown in Figure 10. Verbal relationships between nouns are created by substituting Tree 1 subtrees into Trees 2, 3, and 7. Verb with preposition relationships are created between nouns by substituting the VBX node in Tree 6 with the corresponding node in Trees 2 and 3 to grow the tree to the right, and the PP node in Tree 6 with the corresponding node in Tree 5 to grow the tree to the left. Generation of a full tree stops when all nouns in a group are dominated by the same node, either an S or NP.

4.4 Surface Realization

In the surface realization stage, the system selects a single tree from the generated set of possible trees and removes mark-up to produce a final string. This is also the stage where punctuation may be added. Different strings may be generated depending on different specifications from the user, as discussed at the beginning of Section 4 and shown in the online demo. To evaluate the system against other systems, we specify that the system should (1) not hallucinate likely verbs; and (2) return the longest string possible.

4.4.1 Step 7: Get Final Tree, Clear Mark-Up

We explored two methods for selecting a final string. In one method, a trigram language model built using the Europarl (Koehn, 2005) data with start/end symbols returns the highest-scoring description (normalizing for length). In the second method, we limit the generation system to select the most likely closed-class words (determiners, prepositions) while building the subtrees, overgenerating all possible adjective combinations. The final string is then the one with the most words. We find that the second method produces descriptions that seem more natural and varied than the n-gram ranking method for our development set, and so use the longest string method in evaluation.

4.4.2 Step 8: Prenominal Modifier Ordering

To order sets of selected adjectives, we use the top-scoring prenominal modifier ordering model discussed in Mitchell et al. (2011). This is an n-gram model constructed over noun phrases that were extracted from an automatically parsed version of the New York Times portion of the Gigaword corpus (Graff and Cieri, 2003). With this in place, *blue clear* sky becomes *clear blue* sky, *wooden brown* table becomes *brown wooden* table, etc.

5 Evaluation

Each set of sentences is generated with α (likelihood cutoff) set to .01 and γ (observation count cutoff) set to 3. We compare the system against human-written descriptions and two state-of-the-art vision-to-language systems, the Kulkarni et al. (2011) and Yang et al. (2011) systems.

Human judgments were collected using Amazon's Mechanical Turk (Amazon, 2011). We follow recommended practices for evaluating an NLG system (Reiter and Belz, 2009) and for running a study on Mechanical Turk (Callison-Burch and Dredze, 2010), using a balanced design with each subject rating 3 descriptions from each system. Subjects rated their level of agreement on a 5-point Likert scale including a neutral middle position, and since quality ratings are ordinal (points are not necessarily equidistant), we evaluate responses using a non-parametric test. Participants that took less than 3 minutes to answer all 60 questions and did not include a humanlike rating for at least 1 of the 3 human-written descriptions were removed and replaced. It is important to note that this evaluation compares full generation systems; many factors are at play in each system that may also influence participants' perception, e.g., sentence length (Napoles et al., 2011) and punctuation decisions.

The systems are evaluated on a set of 840 images evaluated in the original Kulkarni et al. (2011) system. Participants were asked to judge the statements given in Figure 12, from Strongly Disagree to Strongly Agree.

	Grammaticality	Main Aspects	Correctness	Order	Humanlikeness
Human	4 (3.77, 1.19)	4 (4.09, 0.97)	4 (3.81, 1.11)	4 (3.88, 1.05)	4 (3.88, 0.96)
Midge	3 (2.95, 1.42)	3 (2.86, 1.35)	3 (2.95, 1.34)	3 (2.92, 1.25)	3 (3.16, 1.17)
Kulkarni et al. 2011	3 (2.83, 1.37)	3 (2.84, 1.33)	3 (2.76, 1.34)	3 (2.78, 1.23)	3 (3.13, 1.23)
Yang et al. 2011	3 (2.95, 1.49)	2 (2.31, 1.30)	2 (2.46, 1.36)	2 (2.53, 1.26)	3 (2.97, 1.23)

Table 4: Median scores for systems, mean and standard deviation in parentheses. Distance between points on the rating scale cannot be assumed to be equidistant, and so we analyze results using a non-parametric test.

GRAMMATICALITY:
This description is **grammatically correct**.
MAIN ASPECTS:
This description **describes the main aspects** of this image.
CORRECTNESS:
This description **does not include extraneous** or incorrect information.
ORDER:
The objects described are mentioned in **a reasonable order**.
HUMANLIKENESS:
It sounds like a **person wrote** this description.

Figure 12: Mechanical Turk prompts.

We report the scores for the systems in Table 4. Results are analyzed using the non-parametric Wilcoxon Signed-Rank test, which uses median values to compare the different systems. Midge outperforms all recent automatic approaches on CORRECTNESS and ORDER, and Yang et al. additionally on HUMANLIKENESS and MAIN ASPECTS. Differences between Midge and Kulkarni et al. are significant at $p < .01$; Midge and Yang et al. at $p < .001$. For all metrics, human-written descriptions still outperform automatic approaches ($p < .001$).

These findings are striking, particularly because Midge uses the same input as the Kulkarni et al. system. Using syntactically informed word co-occurrence statistics from a large corpus of descriptive text improves over state-of-the-art, allowing syntactic trees to be generated that capture the variation of natural language.

6 Discussion

Midge automatically generates language that is as good as or better than template-based systems, tying vision to language at a syntactic/semantic level to produce natural language descriptions. Results are promising, but, there is more work to be done: Evaluators can still tell a difference between human-written descriptions and automatically generated descriptions.

Improvements to the generated language are possible at both the vision side and the language side. On the computer vision side, incorrect objects are often detected and salient objects are often missed. Midge does not yet screen out unlikely objects or add likely objects, and so provides no filter for this. On the language side, likelihood is estimated directly, and the system primarily uses simple maximum likelihood estimations to combine subtrees. The descriptive corpus that informs the system is not parsed with a domain-adapted parser; with this in place, the syntactic constructions that Midge learns will better reflect the constructions that people use.

In future work, we hope to address these issues as well as advance the syntactic derivation process, providing an adjunction operation (for example, to add likely adjectives or adverbs based on language alone). We would also like to incorporate meta-data – even when no vision detection fires for an image, the system may be able to generate descriptions of the time and place where an image was taken based on the image file alone.

7 Conclusion

We have introduced a generation system that uses a new approach to generating language, tying a syntactic model to computer vision detections. Midge generates a well-formed description of an image by filtering attribute detections that are unlikely and placing objects into an ordered syntactic structure. Humans judge Midge's output to be the most natural descriptions of images generated thus far. The methods described here are promising for generating natural language descriptions of the visual world, and we hope to expand and refine the system to capture further linguistic phenomena.

8 Acknowledgements

Thanks to the Johns Hopkins CLSP summer workshop 2011 for making this system possible, and to reviewers for helpful comments. This work is supported in part by Michael Collins and by NSF Faculty Early Career Development (CAREER) Award #1054133.

References

Amazon. 2011. Amazon mechanical turk: Artificial artificial intelligence.

Holly P. Branigan, Martin J. Pickering, and Mikihiro Tanaka. 2007. Contributions of animacy to grammatical function assignment and word order during production. *Lingua*, 118(2):172–189.

Thorsten Brants and Alex Franz. 2006. Web 1T 5-gram version 1.

Chris Callison-Burch and Mark Dredze. 2010. Creating speech and language data with Amazon's Mechanical Turk. *NAACL 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*.

Navneet Dalal and Bill Triggs. 2005. Histograms of oriented gradients for human detections. *Proceedings of CVPR 2005*.

Ali Farhadi, Ian Endres, Derek Hoiem, and David Forsyth. 2009. Describing objects by their attributes. *Proceedings of CVPR 2009*.

Ali Farhadi, Mohsen Hejrati, Mohammad Amin Sadeghi, Peter Young, Cyrus Rashtchian, Julia Hockenmaier, and David Forsyth. 2010. Every picture tells a story: generating sentences for images. *Proceedings of ECCV 2010*.

Pedro Felzenszwalb, David McAllester, and Deva Ramaman. 2008. A discriminatively trained, multiscale, deformable part model. *Proceedings of CVPR 2008*.

Flickr. 2011. http://www.flickr.com. Accessed 1.Sep.11.

Kotaro Funakoshi, Satoru Watanabe, Naoko Kuriyama, and Takenobu Tokunaga. 2004. Generating referring expressions using perceptual groups. *Proceedings of the 3rd INLG*.

Albert Gatt. 2006. Generating collective spatial references. *Proceedings of the 28th CogSci*.

David Graff and Christopher Cieri. 2003. *English Gigaword*. Linguistic Data Consortium, Philadelphia, PA. LDC Catalog No. LDC2003T05.

Philipp Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. *MT Summit*. http://www.statmt.org/europarl/.

Zornitsa Kozareva, Ellen Riloff, and Eduard Hovy. 2008. Semantic class learning from the web with hyponym pattern linkage graphs. *Proceedings of ACL-08: HLT*.

Girish Kulkarni, Visruth Premraj, Sagnik Dhar, Siming Li, Yejin Choi, Alexander C. Berg, and Tamara Berg. 2011. Baby talk: Understanding and generating image descriptions. *Proceedings of the 24th CVPR*.

Irene Langkilde and Kevin Knight. 1998. Generation that exploits corpus-based statistical knowledge. *Proceedings of the 36th ACL*.

Siming Li, Girish Kulkarni, Tamara L. Berg, Alexander C. Berg, and Yejin Choi. 2011. Composing simple image descriptions using web-scale n-grams. *Proceedings of CoNLL 2011*.

Mitchell Marcus, Ann Bies, Constance Cooper, Mark Ferguson, and Alyson Littman. 1995. Treebank II bracketing guide.

George A. Miller. 1995. WordNet: A lexical database for english. *Communications of the ACM*, 38(11):39–41.

Margaret Mitchell, Aaron Dunlop, and Brian Roark. 2011. Semi-supervised modeling for prenominal modifier ordering. *Proceedings of the 49th ACL:HLT*.

Courtney Napoles, Benjamin Van Durme, and Chris Callison-Burch. 2011. Evaluating sentence compression: Pitfalls and suggested remedies. *ACL-HLT Workshop on Monolingual Text-To-Text Generation*.

Vicente Ordonez, Girish Kulkarni, and Tamara L Berg. 2011. Im2text: Describing images using 1 million captioned photographs. *Proceedings of NIPS 2011*.

Slav Petrov. 2010. Berkeley parser. GNU General Public License v.2.

Cyrus Rashtchian, Peter Young, Micah Hodosh, and Julia Hockenmaier. 2010. Collecting image annotations using amazon's mechanical turk. *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk*.

Ehud Reiter and Anja Belz. 2009. An investigation into the validity of some metrics for automatically evaluating natural language generation systems. *Computational Linguistics*, 35(4):529–558.

Ehud Reiter and Robert Dale. 1997. Building applied natural language generation systems. *Journal of Natural Language Engineering*, pages 57–87.

Ehud Reiter and Robert Dale. 2000. *Building Natural Language Generation Systems*. Cambridge University Press.

Yezhou Yang, Ching Lik Teo, Hal Daumé III, and Yiannis Aloimonos. 2011. Corpus-guided sentence generation of natural images. *Proceedings of EMNLP 2011*.

Benjamin Z. Yao, Xiong Yang, Liang Lin, Mun Wai Lee, and Song-Chun Zhu. 2010. I2T: Image parsing to text description. *Proceedings of IEEE 2010*, 98(8):1485–1508.