From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions

Peter Young Alice Lai Micah Hodosh Julia Hockenmaier
Department of Computer Science
University of Illinois at Urbana-Champaign
{pyoung2, aylai2, mhodosh2, juliahmr}@illinois.edu

Abstract

We propose to use the *visual denotations* of linguistic expressions (i.e. the set of images they describe) to define novel *denotational similarity metrics*, which we show to be at least as beneficial as distributional similarities for two tasks that require semantic inference. To compute these denotational similarities, we construct a *denotation graph*, i.e. a subsumption hierarchy over constituents and their denotations, based on a large corpus of 30K images and 150K descriptive captions.

1 Introduction

The ability to draw inferences from text is a prerequisite for language understanding. These inferences are what makes it possible for even brief descriptions of everyday scenes to evoke rich mental images. For example, we would expect an image of *people shopping in a supermarket* to depict aisles of produce or other goods, and we would expect most of these people to be customers who are either standing or walking around. But such inferences require a great deal of commonsense world knowledge. Standard *distributional* approaches to lexical similarity (Section 2.1) are very effective at identifying which words are related to the same topic, and can provide useful features for systems that perform semantic inferences (Mirkin et al., 2009), but are not suited to capture precise entailments between complex expressions. In this paper, we propose a novel approach for the automatic acquisition of *denotational* similarities between descriptions of everyday situations (Section 2). We define the *(visual) denotation* of a linguistic expression as the set of images it describes. We create a corpus of images of everyday activities (each paired with multiple captions; Section 3) to construct a large scale *visual denotation graph* which associates image descriptions with their denotations (Section 4). The algorithm that constructs the denotation graph uses purely syntactic and lexical rules to produce simpler captions (which have a larger denotation). But since each image is originally associated with several captions, the graph can also capture similarities between syntactically and lexically unrelated descriptions. We apply these similarities to two different tasks (Sections 6 and 7): an approximate entailment recognition task for our domain, where the goal is to decide whether the hypothesis (a brief image caption) refers to the same image as the premises (four longer captions), and the recently introduced Semantic Textual Similarity task (Agirre et al., 2012), which can be viewed as a graded (rather than binary) version of paraphrase detection. Both tasks require semantic inference, and our results indicate that denotational similarities are at least as effective as standard approaches to similarity. Our code and data set, as well as the denotation graph itself and the lexical similarities we define over it are available for research purposes at `http://nlp.cs.illinois.edu/Denotation.html`.

2 Towards Denotational Similarities

2.1 Distributional Similarities

The distributional hypothesis posits that linguistic expressions that appear in similar contexts have a

Gray haired man in black suit and yellow tie working in a financial environment.
A graying man in a suit is perplexed at a business meeting.
A businessman in a yellow tie gives a frustrated look.
A man in a yellow tie is rubbing the back of his neck.
A man with a yellow tie looks concerned.

A butcher cutting an animal to sell.
A green-shirted man with a butcher's apron uses a knife to carve out the hanging carcass of a cow.
A man at work, butchering a cow.
A man in a green t-shirt and long tan apron hacks apart the carcass of a cow while another man hoses away the blood.
Two men work in a butcher shop; one cuts the meat from a butchered cow, while the other hoses the floor.

Figure 1: Two images from our data set and their five captions

similar meaning (Harris, 1954). This has led to the definition of vector-based distributional similarities, which represent each word w as a vector $\mathbf{w}$ derived from counts of w's co-occurrence with other words. These vectors can be used directly to compute the *lexical similarities* of words, either via the cosine of the angle between them, or via other, more complex metrics (Lin, 1998). More recently, asymmetric similarities have been proposed as more suitable for semantic inference tasks such as entailment (Weeds and Weir, 2003; Szpektor and Dagan, 2008; Clarke, 2009; Kotlerman et al., 2010). Distributional word vectors can also be used to define the *compositional similarity* of longer strings (Mitchell and Lapata, 2010). To compute the similarity of two strings, the lexical vectors of the words in each string are first combined into a single vector (e.g. by element-wise addition or multiplication), and then an appropriate vector similarity (e.g. cosine) is applied to the resulting pair of vectors.

2.2 Visual Denotations

Our approach is inspired by truth-conditional semantic theories in which the *denotation* of a declarative sentence is assumed to be the set of all situations or possible worlds in which the sentence is true (Montague, 1974; Dowty et al., 1981; Barwise and Perry, 1980). Restricting our attention to visually descriptive sentences, i.e. non-negative, episodic (Carlson, 2005) sentences that can be used to describe an image (Figure 1), we propose to instantiate the abstract notions of possible worlds or situations with concrete sets of images. The interpretation function $[\![\cdot]\!]$ maps sentences to their **visual denotations** $[\![\mathbf{s}]\!]$, which is the set of images $\mathbf{i} \in U_{\mathbf{s}} \subseteq U$ in a 'universe' of images U that $\mathbf{s}$ describes:

$$[\![\mathbf{s}]\!] = \{\mathbf{i} \in U \mid \mathbf{s} \text{ is a truthful description of } \mathbf{i}\} \quad (1)$$

Similarly, we map nouns and noun phrases to the set of images that depict the objects they describe, and verbs and verb phrases to the set of images that depict the events they describe.

2.3 Denotation Graphs

Denotations induce a partial ordering over descriptions: if $\mathbf{s}$ (e.g. *"a poodle runs on the beach"*) entails a description $\mathbf{s}'$ (e.g. *"a dog runs"*), its denotation is a subset of the denotation of $\mathbf{s}'$ ($[\![\mathbf{s}]\!] \subseteq [\![\mathbf{s}']\!]$), and we say that $\mathbf{s}'$ *subsumes* the more specific $\mathbf{s}$ ($\mathbf{s}' \sqsubseteq \mathbf{s}$). In our domain of descriptive sentences, we can obtain more generic descriptions by simple **syntactic and lexical operations** $\omega \in O \subset S \times S$ that preserve upward entailment, so that if $\omega(\mathbf{s}) = \mathbf{s}'$, $[\![\mathbf{s}]\!] \subseteq [\![\mathbf{s}']\!]$. We consider three types of operations: the removal of optional material (e.g PPs like *on the beach*), the extraction of simpler constituents (NPs, VPs, or simple Ss), and lexical substitutions of nouns by their hypernyms (*poodle* $\rightarrow$ *dog*). These operations are akin to the *atomic edits* of MacCartney and Manning (2008)'s NatLog system, and allow us to construct large subsumption hierarchies over image descriptions, which we call **denotation graphs**. Given a set of (upward entailment-preserving) operations $O \subset S \times S$, the denotation graph $DG = \langle E, V \rangle$ of a set of images I and a set of strings S represents a subsumption hierarchy in which each node $V = \langle \mathbf{s}, [\![\mathbf{s}]\!] \rangle$ corresponds to a string $\mathbf{s} \in S$ and its denotation $[\![\mathbf{s}]\!] \subseteq I$. Directed edges $e = (\mathbf{s}, \mathbf{s}') \in E \subseteq V \times V$ indicate a subsumption relation $\mathbf{s} \sqsubseteq \mathbf{s}'$ between a more generic expression $\mathbf{s}$ and its child $\mathbf{s}'$. An edge from $\mathbf{s}$ to $\mathbf{s}'$

exists if there is an operation $\omega \in O$ that *reduces* the string $\mathbf{s}'$ to $\mathbf{s}$ (i.e. $\omega(\mathbf{s}') = \mathbf{s}$) and its inverse ω^{-1} *expands* the string $\mathbf{s}$ to $\mathbf{s}'$ (i.e. $\omega^{-1}(\mathbf{s}) = \mathbf{s}'$).

2.4 Denotational Similarities

Given a denotation graph over N images, we estimate the denotational probability of an expression $\mathbf{s}$ with a denotation of size $|[\![\mathbf{s}]\!]|$ as $P_{[\![]\!]}(\mathbf{s}) = |[\![\mathbf{s}]\!]|/N$, and the joint probability of two expressions analogously as $P_{[\![]\!]}(\mathbf{s}, \mathbf{s}') = |[\![\mathbf{s}]\!] \cap [\![\mathbf{s}']\!]|/N$. The conditional probability $P_{[\![]\!]}(\mathbf{s} \mid \mathbf{s}')$ indicates how likely $\mathbf{s}$ is to be true when $\mathbf{s}'$ holds, and yields a simple directed denotational similarity. The (normalized) pointwise mutual information (PMI) (Church and Hanks, 1990) defines a symmetric similarity:

$$nPMI_{[\![]\!]}(\mathbf{s}, \mathbf{s}') = \frac{\log\left(\frac{P_{[\![]\!]}(\mathbf{s},\mathbf{s}')}{P_{[\![]\!]}(\mathbf{s})P_{[\![]\!]}(\mathbf{s}')}\right)}{-\log(P_{[\![]\!]}(\mathbf{s}, \mathbf{s}'))}$$

We set $P_{[\![]\!]}(\mathbf{s}|\mathbf{s}) = nPMI_{[\![]\!]}(\mathbf{s}, \mathbf{s}) = 1$, and, if $\mathbf{s}$ or $\mathbf{s}'$ are not in the denotation graph, $nPMI_{[\![]\!]}(\mathbf{s}, \mathbf{s}') = P_{[\![]\!]}(\mathbf{s}, \mathbf{s}') = 0$.

3 Our Data Set

Our data set (Figure 1) consists of 31,783 photographs of everyday activities, events and scenes (all harvested from Flickr) and 158,915 captions (obtained via crowdsourcing). It contains and extends Hodosh et al. (2013)'s corpus of 8,092 images. We followed Hodosh et al. (2013)'s approach to collect images. We also use their annotation guidelines, and use similar quality controls to correct spelling mistakes, eliminate ungrammatical or non-descriptive sentences. Almost all of the images that we add to those collected by Hodosh et al. (2013) have been made available under a Creative Commons license. Each image is described independently by five annotators who are not familiar with the specific entities and circumstances depicted in them, resulting in captions such as *"Three people setting up a tent"*, rather than the kind of captions people provide for their own images (*"Our trip to the Olympic Peninsula"*). Moreover, different annotators use different levels of specificity, from describing the overall situation (*performing a musical piece*) to specific actions (*bowing on a violin*). This variety of descriptions associated with the same image is what allows us to induce denotational similarities between expressions that are not trivially related by syntactic rewrite rules.

4 Constructing the Denotation Graph

The construction of the denotation graph consists of the following steps: preprocessing and linguistic analysis of the captions, identification of applicable transformations, and generation of the graph itself.

Preprocessing and Linguistic Analysis We use the Linux spell checker, the OpenNLP tokenizer, POS tagger and chunker (`http://opennlp.apache.org`), and the Malt parser (Nivre et al., 2006) to analyze the captions. Since the vocabulary of our corpus differs significantly from the data these tools are trained on, we resort to a number of heuristics to improve the analyses they provide. Since some heuristics require us to identify different entity types, we developed a lexicon of the most common entity types in our domain (people, clothing, bodily appearance (e.g. hair or body parts), containers of liquids, food items and vehicles).

After spell-checking, we normalize certain words and compounds with several spelling variations, e.g. *barbecue (barbeque, BBQ)*, *gray (grey)*, *waterski (water ski)*, *brown-haired (brown haired)*, and tokenize the captions using the OpenNLP tokenizer. The OpenNLP POS tagger makes a number of systematic errors on our corpus (e.g. mistagging main verbs as nouns). Since these errors are highly systematic, we are able to correct them automatically by applying deterministic rules (e.g. *climbs* is never a noun in our corpus, *stand* is a noun if it is preceded by *vegetable* but a verb when preceded by a noun that refers to people). These fixes apply to 27,784 (17% of the 158,915 image captions). Next, we use the OpenNLP chunker to create a shallow parse. Fixing its (systematic) errors affects 28,587 captions. We then analyze the structure of each NP chunk to identify heads, determiners and prenominal modifiers. The head may include more than a single token if WordNet (or our hypernym lexicon, described below) contains a corresponding entry (e.g. *little girl*). Determiners include phrases such as *a couple* or *a few*. Although we use the Malt parser (Nivre et al., 2006) to identify subject-verb-object dependencies, we have found it more accurate to develop deterministic heuristics and lexi-

cal rules to identify the boundaries of complex (e.g. conjoined) NPs, allowing us to treat *"a man with red shoes and a white hat"* as an NP followed by a single PP, but *"a man with red shoes and a white-haired woman"* as two NPs, and to transform e.g. *"standing by a man and a woman"* into *"standing"* and not *"standing and a woman"* when dropping the PP.

Hypernym Lexicon We use our corpus and WordNet to construct a hypernym lexicon that allows us to replace head nouns with more generic terms. We only consider hypernyms that occur themselves with sufficient frequency in the original captions (replacing *"adult"* with *"person"*, but not with *"organism"*). Since the language in our corpus is very concrete, each noun tends to have a single sense, allowing us to always replace it with the same hypernyms.[1] But since WordNet provides us with multiple senses for most nouns, we first have to identify which sense is used in our corpus. To do this, we use the heuristic cross-caption coreference algorithm of Hodosh et al. (2010) to identify coreferent NP chunks among the original five captions of each image.[2] For each ambiguous head noun, we consider every non-singleton coreference chains it appears in, and reduce its synsets to those that stand in a hypernym-hyponym relation with at least one other head noun in the chain. Finally, we apply a greedy majority voting algorithm to iteratively narrow down each term's senses to a single synset that is compatible with the largest number of coreference chains it occurs in.

Caption Normalization In order to increase the recall of the denotations we capture, we drop all punctuation marks, and lemmatize nouns, verbs, and adjectives that end in *"-ed"* or *"-ing"* before generating the denotation graph. In order to distinguish between frequently occurring homonyms where the noun is unrelated to the verb, we change all forms of the verb *dress* to *dressed*, all forms of the verb *stand* to *standing* and all forms of the verb *park* to *parking*. Finally, we drop sentence-initial *there/here/this is/are* (as in *there is a dog splashing in the water*), and normalize the expressions *in X* and *dressed (up) in X* (where *X* is an article of clothing or a color) to *wear X*. We reduce plural determiners to {*two, three, some*}, and drop singular determiners except for *no*.

[1] Descriptions of people that refer to both age and gender (e.g. *"man"*) can have multiple distinct hypernyms (*"adult"*/*"male"*). Because our annotators never describe young children or babies as *"persons"*, we only allow terms that are likely to describe adults or teenagers (including occupations) to be replaced by the term *"person"*. This means that the term *"girl"* has two senses: a female child (the default) or a younger woman. We distinguish the two senses in a preprocessing step: if the other captions of the same image do not mention children, but refer to teenaged or adult women, we assign *girl* the *woman*-sense. Some nouns that end in *-er* (e.g. *"diner", "pitcher"* also violate our monosemy assumption.

[2] Coreference resolution has also been used for word sense disambiguation by Preiss (2001) and Hu and Liu (2011).

4.1 Rule Templates

The denotation graph contains a directed edge from $\mathbf{s}$ to $\mathbf{s}'$ if there is a rule ω that reduces $\mathbf{s}'$ to $\mathbf{s}$, with an inverse ω^{-1} that expands $\mathbf{s}$ to $\mathbf{s}'$. Reduction rules can drop optional material, extract simpler constituents, or perform lexical substitutions.

Drop Pre-Nominal Modifiers: *"red shirt"* → *"shirt"* In an NP of the form "X Y Z", where X and Y both modify the head Z, we only allow X and Y to be dropped separately if "X Z" and "Y Z" both occur elsewhere in the corpus. Since *"white building"* and *"stone building"* occur elsewhere in the corpus, we generate both *"white building"* and *"stone building"* from the NP *"white stone building"*. But since *"ice player"* is not used, we replace *"ice hockey player"* only with *"hockey player"* (which does occur) and then *"player"*.

Drop Other Modifiers *"run quickly"* → *"run"* We drop ADVP chunks and adverbs in VP chunks. We also allow a prepositional phrase (a preposition followed by a possibly conjoined NP chunk) to be dropped if the preposition is locational (*"in", "on", "above"*, etc.), directional (*"towards", "through", "across"*, etc.), or instrumental (*"by", "for", "with"*). Similarly, we also allow the dropping of all *"wear* NP*"* constructions. Since the distinction between particles and prepositions is often difficult, we also use a predefined list of phrasal verbs that commonly occur in our corpus to identify constructions such as *"climb up a mountain"*, which is transformed into *"climb a mountain"* or *"walk down a street"*, which is transformed into *"walk"*.

Replace Nouns by Hypernyms: *"red shirt"* → *"red clothing"* We iteratively use our hypernym

GENERATEGRAPH():
$\mathrm{Q}, \mathrm{Captions}, \mathrm{Rules} \leftarrow \emptyset$
for all $c \in ImageCorpus$ **do**
  $\mathrm{Rules}(c) \leftarrow GenerateRules(\mathbf{s}_c)$
  $pushAll(\mathrm{Q}, \{c\} \times RootNodes(\mathbf{s}_c, \mathrm{Rules}(c)))$
while $\neg empty(\mathrm{Q})$ **do**
  $(c, \mathbf{s}) \leftarrow pop(\mathrm{Q})$
  $\mathrm{Captions}(\mathbf{s}) \leftarrow \mathrm{Captions}(\mathbf{s}) \cup \{c\}$
  if $|\mathrm{Captions}(\mathbf{s})| = 2$ **then**
    for all $c' \in \mathrm{Captions}(\mathbf{s})$ **do**
      $pushAll(\mathrm{Q}, \{c'\} \times Children(\mathbf{s}, \mathrm{Rules}(c')))$
  else if $|\mathrm{Captions}(\mathbf{s})| > 2$ **then**
    $pushAll(\mathrm{Q}, \{c\} \times Children(\mathbf{s}, \mathrm{Rules}(c)))$

Figure 2: Generating the graph

lexicon to make head nouns more generic. We only allow head nouns to be replaced by their hypernyms if any age based modifiers have already been removed: *"toddler"* can be replaced with *"child"*, but not *"older toddler"* with *"older child"*.

Handle Partitive NPs: *cup of tea* → *"cup", "tea"* In most partitive NP_1-of-NP_2 constructions (*"cup of tea"*, *"a team of football players"*) the corresponding entity can be referred to by both the first or the second NP. Exceptions include the phrase *"body of water"*, and expressions such as *"a kind/type/sort of"*, which we treat similar to determiners.

Handle *VP*$_1$*-to-VP*$_2$ Cases Depending on the first verb, we replace VPs of the form **X to Y** with both **X** and **Y** if **X** is a movement or posture (*jump to catch*, etc.). Otherwise we distinguish between cases we can only replace with **X** (*wait to jump*) and those we can only replace with **Y** (*seem to jump*).

Extract Simpler Constituents Any noun phrase or verb phrase can also be used as a node in the graph and simplified further. We use the Malt dependencies (and the person terms in the entity type lexicon) to identify and extract subject-verb-object chunks which correspond to simpler sentences that we would otherwise not be able to obtain: from *"man laugh(s) while drink(ing)"*, we extract *"man laugh"* and *"man drink"*, and then further split those into *"man"*, *"laugh(s)"*, and *"drink"*.

4.2 Graph Generation

The naive approach to graph generation would be to generate all possible strings for each caption. However, this would produce far more strings than can be processed in a reasonable amount of time, and most of these strings would have uninformative denotations, consisting of only a single image. To make graph generation tractable, we use a top-down algorithm which generates the graph from the most generic (root) nodes, and stops at nodes that have a singleton denotation (Figure 2). We first identify the set of rules that can apply to each original caption (*GenerateRules*). These rules are then used to reduce each caption as much as possible. The resulting (maximally generic) strings are added as root nodes to the graph (*RootNodes*), and added to the queue Q. Q keeps track of all currently possible node expansions. It contains items $\langle c, \mathbf{s} \rangle$, which pair the ID of an original caption and its image (c) with a string ($\mathbf{s}$) that corresponds to an existing node in the graph and can be derived from c's caption. When $\langle c, \mathbf{s} \rangle$ is processed, we check how many captions have generated $\mathbf{s}$ so far (Captions($\mathbf{s}$)). If $\mathbf{s}$ has more than a single caption, we use each of the applicable rewrite rules of c's caption to create new strings $\mathbf{s}'$ that correspond to the children of $\mathbf{s}$ in the graph, and push all resulting $\langle c, \mathbf{s}' \rangle$ onto Q. If $\mathbf{c}$ is the second caption of $\mathbf{s}$, we also use all of the applicable rewrite rules from the first caption c' to create its children.

A post-processing step (not shown in Figure 2) attaches each original caption to all leaf nodes of the graph to which it can be reduced. Finally, we obtain the denotation of each node $\mathbf{s}$ from the set of images whose captions are in Captions($\mathbf{s}$).

5 The Denotation Graph

Size and Coverage On our corpus of 158,439 unique captions and 31,783 images, the denotation graph contains 1,749,097 captions, out of which 230,811 describe more than a single image. Table 1 provides the distribution of the size of denotations. It is perhaps surprising that the 161 captions which describe each over 1,000 images do not just consist of nouns such as *person*, but also contain simple sentences such as *woman standing*, *adult work*, *person walk street*, or *person play instrument*. Since the graph is derived from the original captions by very simple syntactic operations, the denotations it captures are most likely incomplete: $[\![soccer\ player]\!]$ contains 251 images, $[\![play\ soccer]\!]$ contains 234 images, and $[\![soccer\ game]\!]$ contains

Size of denotations	$\|[\![\mathbf{s}]\!]\| \geq 1$	$\|[\![\mathbf{s}]\!]\| \geq 2$	$\|[\![\mathbf{s}]\!]\| \geq 5$	$\|[\![\mathbf{s}]\!]\| \geq 10$	$\|[\![\mathbf{s}]\!]\| \geq 100$	$\|[\![\mathbf{s}]\!]\| \geq 1000$
Nr. of captions	1,749,096	230,811	53,341	22,683	1,921	161

Table 1: Distribution of the size of denotations in our graph

119 images. We have not yet attempted to identify variants in word order (*"stick tongue out"* vs. *"stick out tongue"*) or equivalent choices of preposition (*"look into mirror"* vs. *"look in mirror"*). Despite this brittleness, the current graph already gives us a large number of semantic associations.

Denotational Similarities The following examples of the similarities found by $nPMI_{[\![]\!]}$ and $P_{[\![]\!]}$ show that denotational similarities do not simply find topically related events, but instead find events that are related by entailment:

$P_{[\![]\!]}(x\|y)$	x	y
0.962	*sit*	*eat lunch*
0.846	*play guitar*	*strum*
0.811	*surf*	*catch wave*
0.800	*ride horse*	*rope calf*
0.700	*listen*	*sit in classroom*

If someone is *eating lunch*, it is likely that they are *sitting*, and people who *sit in a classroom* are likely to be *listening* to somebody. These entailments can be very precise: *"walk up stair"* entails *"ascend"*, but not *"descend"*; the reverse is true for *"walk down stair"*:

$P_{[\![]\!]}(x\|y)$	x =*ascend*	x =*descend*
y =*walk up stair*	32.0	0.0
y =*walk down stair*	0.0	30.8

$nPMI_{[\![]\!]}$ captures paraphrases as well as closely related events: people *look in a mirror* when *shaving their face*, and baseball players may *try to tag* someone who is *sliding into base*:

$nPMI_{[\![]\!]}$	x	y
0.835	*open present*	*unwrap*
0.826	*lasso*	*try to rope*
0.791	*get ready to kick*	*run towards ball*
0.785	*try to tag*	*slide into base*
0.777	*shave face*	*look in mirror*

Comparing the expressions that are most similar to *"play baseball"* or *"play football"* according to the denotational $nPMI_{[\![]\!]}$ and the compositional Σ similarities reveals that the denotational similarity finds a number of actions that are part of the particular sport, while the compositional similarity finds events that are similar to *playing baseball* (*football*):

play baseball

$nPMI_{[\![]\!]}$		Σ	
0.674	*tag him*	0.859	*play softball*
0.637	*hold bat*	0.782	*play game*
0.616	*try to tag*	0.768	*play ball*
0.569	*slide into base*	0.741	*play catch*
0.516	*pitch ball*	0.739	*play cricket*

play football

$nPMI_{[\![]\!]}$		Σ	
0.623	*tackle person*	0.826	*play game*
0.597	*hold football*	0.817	*play rugby*
0.545	*run down field*	0.811	*play soccer*
0.519	*wear white jersey*	0.796	*play on field*
0.487	*avoid*	0.773	*play ball*

6 Task 1: Approximate Entailment

A caption never provides a complete description of the depicted scene, but commonsense knowledge often allows us to draw implicit inferences: when somebody mentions a *bride*, it is quite likely that the picture shows a woman in a *wedding dress*; a picture of a *parent* most likely also has a *child* or *baby*, etc. In order to compare the utility of denotational and distributional similarities for drawing these inferences, we apply them to an *approximate entailment task*, which is loosely modeled after the Recognizing Textual Entailment problem (Dagan et al., 2006), and consists of deciding whether a brief caption $\mathbf{h}$ (the hypothesis) can describe the same image as a set of captions $\mathbf{P} = \{\mathbf{p}_1, ..., \mathbf{p}_N\}$ known to describe the same image (the premises).

Data We generate positive and negative items $\langle \mathbf{P}, \mathbf{h}, \pm \rangle$ (Figure 3) as follows: Given an image, any subset of four of its captions form a set of premises. A hypothesis is either a short verb phrase or sentence that corresponds to a node in the denotation graph. By focusing on short hypotheses, we minimize the possibility that they contain extraneous details that cannot be inferred from the premises. Positive examples are generated by choosing a node $\mathbf{h}$ as hypothesis and an image $i \in [\![\mathbf{h}]\!]$ such that exactly one caption of i generates $\mathbf{h}$ and the other four captions of i are not descendants of $\mathbf{h}$ and hence do not trivially entail $\mathbf{h}$, giving an unfair advantage to denotational approaches. Negative examples are generated by choosing a node $\mathbf{h}$ as hypothesis and selecting four of the captions of an image $i \notin [\![\mathbf{h}]\!]$.

Premises:	A woman with dark hair in bending, open mouthed, towards the back of a dark headed toddler's head. A dark-haired woman has her mouth open and is hugging a little girl while sitting on a red blanket. A grown lady is snuggling on the couch with a young girl and the lady has a frightened look. A mom holding her child on a red sofa while they are both having fun.
VP Hypothesis:	make face
Premises:	A man editing a black and white photo at a computer with a pencil in his ear. A man in a white shirt is working at a computer. A guy in white t-shirt on a mac computer. A young main is using an Apple computer.
S Hypothesis:	man sit

Figure 3: Positive examples from the Approximate Entailment tasks.

Since our items are created automatically, a positive hypothesis is not necessarily logically entailed by its premises. We have performed a small-scale human evaluation on 300 items (200 positive, 100 negative), each judged independently by the same three judges (inter-annotator agreement: Fleiss-κ = 0.74). Our results indicate that over half (55%) of the positive hypotheses can be inferred from their premises alone without looking at the original image, while almost none of the negative hypotheses (100% for sentences, 96% for verb phrases) can be inferred from their premises. The training items are generated from the captions of 25,000 images, and the test items are generated from a disjoint set of 3,000 images. The VP data set consists of 290,000 training items and 16,000 test items, while the S data set consists of 400,000 training items and 22,000 test items. Half of the items in each set are positive, and the other half are negative.

Models All of our models are binary MaxEnt classifiers, trained using MALLET (McCallum, 2002). We have two baseline models: a plain bag-of-words model (BOW) and a bag-of-words model where we add all hypernyms in our lexicon to the captions before computing their overlap (BOW-H). This is intended to minimize the advantage the denotational features obtain from the hypernym lexicon used to construct the denotation graph. In both cases, a global BOW feature captures the fraction of tokens in the hypothesis that are contained in the premises. Word-specific BOW features capture the product of the frequencies of each word in $\mathbf{h}$ and $\mathbf{P}$. All other models extend the BOW-H model.

Denotational Similarity Features We compute denotational similarities $nPMI_{[\![]\!]}$ and $P_{[\![]\!]}$ (Section 2.4) over the pairs of nodes in a denotation graph that is restricted to the training images. We only consider pairs of nodes $\mathbf{n}, \mathbf{n}'$ if their denotations contain at least 10 images and their intersection contains at least 2 images.

To map an item $\langle \mathbf{P}, \mathbf{h} \rangle$ to denotational similarity features, we represent the premises as the set of all nodes P that are ancestors of its captions. A sentential hypothesis is represented as the set of nodes $H = \{h_S, h_{sbj}, h_{VP}, h_v, h_{dobj}\}$ that correspond to the sentence ($\mathbf{h}$ itself), its subject, its VP and its direct object. A VP hypothesis has only the nodes $H = \{h_{VP}, hv, h_{dobj}\}$. In both cases, h_{dobj} may be empty. Both of the denotational similarities $nPMI_{[\![]\!]}(h, p)$ and $P_{[\![]\!]}(h|p)$ for $h \in H$, $p \in P$ lead to two constituent-specific features, sum_x and max_x, (e.g. $\mathrm{sum}_{sbj} = \sum_p \mathrm{sim}(h_{sbj}, p)$, $\mathrm{max}_{dobj} = \mathrm{max}_p \mathrm{sim}(h_{dobj}, p)$) and two global features $\mathrm{sum}_{p,h} = \sum_{p,h} \mathrm{sim}(h, p)$ and $\mathrm{max}_{p,h} = \mathrm{max}_{p,h} \mathrm{sim}(h, p)$. Each constituent type also has a set of node-specific $\mathrm{sum}_{x,\mathbf{s}}$ and $\mathrm{max}_{x,\mathbf{s}}$ features that are on when constituent x in $\mathbf{h}$ is equal to the string $\mathbf{s}$ and whose value is equal to the constituent-based feature. For $P_{[\![]\!]}$, each constituent (and each constituent-node pair) has an additional feature $P(h|P) = 1 - \prod_n (1 - P_{[\![]\!]}(h|p_n))$ that estimates the probability that h is generated by some node in the premise.

Lexical Similarity Features We use two symmetric lexical similarities: standard cosine distance (cos), and Lin (1998)'s similarity (Lin):

$$\cos(\mathbf{w}, \mathbf{w}') = \frac{\mathbf{w} \cdot \mathbf{w}'}{\|\mathbf{w}\| \|\mathbf{w}'\|}$$

$$\mathrm{Lin}(\mathbf{w}, \mathbf{w}') = \frac{\sum_{i:\mathbf{w}(i)>0 \wedge \mathbf{w}'(i)>0} \mathbf{w}(i) + \mathbf{w}'(i)}{\sum_i \mathbf{w}(i) + \sum_i \mathbf{w}'(i)}$$

We use two directed lexical similarities: Clarke (2009)'s similarity (Clk), and Szpektor and Dagan (2008)'s balanced precision (Bal), which builds on Lin and on Weeds and Weir (2003)'s similarity ($\mathbf{W}$):

$$\mathrm{Clk}(\mathbf{w} \mid \mathbf{w}') = \frac{\sum_{i:\mathbf{w}(i)>0 \wedge \mathbf{w}'(i)>0} \min(\mathbf{w}(i), \mathbf{w}'(i))}{\sum_i \mathbf{w}(i)}$$

$$\mathrm{Bal}(\mathbf{w} \mid \mathbf{w}') = \sqrt{\mathbf{W}(\mathbf{w} \mid \mathbf{w}') \times \mathrm{Lin}(\mathbf{w}, \mathbf{w}')}$$

$$\mathbf{W}(\mathbf{w} \mid \mathbf{w}') = \frac{\sum_{i:\mathbf{w}(i)>0 \wedge \mathbf{w}'(i)>0} \mathbf{w}(i)}{\sum_i \mathbf{w}(i)}$$

We also use two publicly available resources that provide precomputed similarities, Kotlerman et al. (2010)'s DIRECT noun and verb rules and Chklovski and Pantel (2004)'s VERBOCEAN rules. Both are motivated by the need for numerically quantifiable semantic inferences between predicates. We only use entries that correspond to single tokens (ignoring e.g. phrasal verbs).

Each lexical similarity results in the following features: words in the output are represented by a max-sim$_w$ feature which captures its maximum similarity with any word in the premises (max-sim$_w$ $=$ $\max_{w' \in P} \mathrm{sim}(w, w')$) and by a sum-sim$_w$ feature which captures the sum of its similarities to the words in the premises (sum-sim$_w$ $= \sum_{w' \in P} \mathrm{sim}(w, w')$). Global max sim and sum sim features capture the maximal (resp. total) similarity of any word in the hypothesis to the premise.

We compute distributional and compositional similarities (cos, Lin, Bal, Clk, Σ, Π) on our image captions ("cap"), the BNC and Gigaword. For each corpus C, we map each word w that appears at least 10 times in C to a vector $\mathbf{w}_C$ of the non-negative normalized pointwise mutual information scores (Section 2.4) of w and the 1,000 words (excluding stop words) that occur in the most sentences of C. We generally define $P(w)$ (and $P(w, w')$) as the fraction of sentences in C in which w (and w') occur. To allow a direct comparison between distributional and denotational similarities, we first define $P(w)$ (and $P(w, w')$) over individual captions ("cap"), and then, to level the playing field, we redefine $P(w)$ (and $P(w, w')$) as the fraction of images in whose captions w (and w') occur ("img"), and then we use our lexicon to augment captions with all hypernyms ("+hyp"). Finally, we include BNC and Gigaword similarity features ("all").

	VP task		S task	
Baseline 1: BoW	58.7		71.2	
Baseline 2: BoW-H	59.0		73.6	
External 1: DIRECT	59.2		73.5	
External 2: VerbOcean	60.8		74.0	
	Cap	**All**	**Cap**	**All**
Distributional cos	67.5	**71.9**	76.1	**78.9**
Distributional Lin	62.6	70.2	75.4	77.8
Distributional Bal	62.3	69.6	74.7	75.3
Distributional Clk	62.4	69.2	75.4	77.5
Compositional Π	68.4	70.3	75.3	77.3
Compositional Σ	67.8	71.4	76.9	79.2
Compositional Π, Σ	69.8	**72.7**	77.0	**79.6**
Denotational $nPMI_{[\![]\!]}$	74.9		80.2	
Denotational $P_{[\![]\!]}$	73.8		79.5	
$nPMI_{[\![]\!]}$, $P_{[\![]\!]}$	**75.5**		**81.2**	
Combined cos, Π, Σ	71.1	72.6	77.4	79.2
$nPMI_{[\![]\!]}$, $P_{[\![]\!]}$, Π, Σ	75.6	**75.9**	80.2	80.7
$nPMI_{[\![]\!]}$, $P_{[\![]\!]}$, cos	75.6	75.7	80.2	**81.2**
$nPMI_{[\![]\!]}$, $P_{[\![]\!]}$, cos, Π, Σ	75.8	**75.9**	**81.2**	80.5

Table 2: Test accuracy on Approximate Entailment.

Compositional Similarity Features We use two standard compositional baselines to combine the word vectors of a sentence into a single vector: addition ($\mathbf{s}_{\sum} = \mathbf{w}_1 + ... + \mathbf{w}_n$, which can be interpreted as a disjunctive operation), and element-wise (Hadamard) multiplication ($\mathbf{s}_{\prod} = \mathbf{w}_1 \odot ... \odot \mathbf{w}_n$, which can be seen as a conjunctive operation). In both cases, we represent the premises (which consist of four captions) as a the sum of each caption's vector $\mathbf{p} = \mathbf{p}_1 + ...\mathbf{p}_4$. This gives two compositional similarity features: $\Sigma = \cos(\mathbf{p}_\Sigma, \mathbf{h}_\Sigma)$, and $\Pi = \cos(\mathbf{p}_\Pi, \mathbf{h}_\Pi)$.

6.1 Experimental Results

Table 2 provides the test accuracy of our models on the VP and S tasks. Adding hypernyms (BOW-H) yields a slight improvement over the basic BOW model. Among the external resources, VERBOCEAN is more beneficial than DIRECT, but neither help as much as in-domain distributional similarities (this may be due to sparsity).

Table 2 shows only the simplest ("Cap") and the most complex ("all") distributional and compositional models, but Table 3 provides accuracies of these models as we go from standard sentence-based co-occurrence counts towards more denotation graph-like co-occurrence counts that are based on all captions describing the same image ("Img"),

	VP task				S task			
	Cap	**Img**	**+Hyp**	**All**	**Cap**	**Img**	**+Hyp**	**All**
cos	67.5	69.3	69.8	**71.9**	76.1	76.8	77.5	**78.9**
Lin	62.6	63.4	61.3	**70.0**	75.4	74.8	75.2	**77.8**
Bal	62.3	61.9	62.8	**69.6**	74.7	75.5	75.1	**75.3**
Clk	62.4	67.3	68.0	**69.2**	75.4	75.5	76.0	**77.5**
Π	68.4	**70.5**	**70.5**	70.3	75.3	76.6	77.1	**77.3**
Σ	67.8	71.4	**71.6**	71.4	76.9	78.1	79.1	**79.2**
Π, Σ	69.8	72.7	**72.9**	72.7	77.0	78.6	79.3	**79.6**
$nPMI_{[\![]\!]}$			74.9				80.2	
$P_{[\![]\!]}$			73.8				79.5	
$nPMI_{[\![]\!]}, P_{[\![]\!]}$			**75.5**				**81.2**	

Table 3: Accuracy on hypotheses as various additions are made to the vector corpora. **Cap** is the image corpus with caption co-occurrence. **Img** is the image corpus with image co-occurrence. **+Hyp** augments the image corpus with hypernyms and uses image co-occurrence. **All** adds the BNC and Gigaword corpora to **+Hyp**.

	VP task			S task		
Words in h	**1**	**2**	**3+**	**2**	**3**	**4+**
% of items	**72.8**	**13.9**	**13.3**	**65.3**	**22.8**	**11.9**
BoW-H	52.0	75.0	80.1	69.1	80.8	84.4
cos (All)	68.8	79.4	81.1	75.9	83.9	85.7
$\sum$ (All)	68.1	80.8	79.5	76.5	83.9	85.1
$nPMI_{[\![]\!]}$	**72.0**	**82.9**	**82.2**	**77.3**	**85.4**	**86.2**

Table 4: Accuracy on hypotheses of varying length.

include hypernyms ("+Hyp"), and add information from other corpora ("All"). The "+Hyp" column in Table 3 shows that the denotational metrics clearly outperform any distributional metric when both have access to the same information. Although the distributional models benefit from the BNC and Gigaword-based similarities ("All"), their performance is still below that of the denotational models. Among the distributional model, the simple cos performs better than Lin, or the directed Clk and Bal similarities. In all cases, giving models access to different similarity features improves performance.

Table 4 shows the results by hypothesis length. As the length of **h** increases, classifiers that use similarities between pairs of words (BOW-H and cos) continue to improve in performance relative to the classifiers that use similarities between phrases and sentences (Σ and $nPMI_{[\![]\!]}$). Most likely, this is due to the lexical similarities having a larger set of features to work with for longer **h**. $nPMI_{[\![]\!]}$ does especially well on shorter **h**, likely due to the shorter **h** having larger denotations.

7 Task 2: Semantic Textual Similarity

To assess how the denotational similarities perform on a more established task and domain, we apply them to the 1500 sentence pairs from the MSR Video Description Corpus (Chen and Dolan, 2011) that were annotated for the SemEval 2012 Semantic Textual Similarity (STS) task (Agirre et al., 2012). The goal of this task is to assign scores between 0 and 5 to a pair of sentences, where 5 indicates equivalence, and 0 unrelatedness. Since this is a symmetric task, we do not consider directed similarities. And because the goal of this experiment is not to achieve the best possible performance on this task, but to compare the effectiveness of denotational and more established similarities, we only compare the impact of denotational similarities with compositional similarities computed on our own corpus. Since the MSR Video corpus associates each video with multiple sentences, it is in principle also amenable to a denotational treatment, but the STS task description explicitly forbids its use.

7.1 Models

Baseline and Compositional Features Our starting point is Bär et al. (2013)'s DKPro Similarity, one of the top-performing models from the 2012 STS shared task, which is available and easily modified. It consists of a log-linear regression model trained on multiple text features (word and character n-grams, longest common substring and longest common subsequence, Gabrilovich and Markovitch (2007)'s Explicit Semantic Analysis, and Resnik (1995)'s WordNet-based similarity). We investigate the effects of adding compositional (computed on the vectors obtained from the image-caption training data) and denotational similarity features to this state-of-the-art system.

Denotational Features Since the STS task is symmetric, we only consider $nPMI_{[\![]\!]}$ similarities. We again represent each sentence s by features based on 5 types of constituents: $S = \{s_S, s_{sbj}, s_{VP}, s_v, s_{dobj}\}$. Since sentences might be complex, they might contain multiple constituents of the same type, and we therefore think of each feature as a feature over sets of nodes. For each constituent C we consider two sets of nodes in the denotation graph: C itself (typically leaf nodes),

	DKPro	+Σ, Π (img)	+$nPMI_{[\![]\!]}$	+*both*
Pearson r	0.868	0.880	0.888	0.890

Table 5: Performance on the STS MSRvid task: DKPro (Bär et al., 2013) plus compositional (Σ, Π) and/or denotational similarities ($nPMI_{[\![]\!]}$) from our corpus

and C^{anc}, their parents and grandparents. For each pair of sentences, C-C similarities compute the similarity of the constituents of the same type, while C-all similarities compute the similarity of a C constituent in one sentence against all constituents in the other sentence. For each pair of constituents we consider three similarity features: $\text{sim}(C_1, C_2)$, $\max(\text{sim}(C_1 C_2^{anc}), \text{sim}(C_1^{anc}, C_2))$, $\text{sim}(C_1^{anc}, C_2^{anc})$. The similarity of two sets of nodes is determined by the maximal similarity of any pair of their elements: $\text{sim}(C_1, C_2) = \max_{c_1 \in C_1, c_2 \in C_2} nPMI_{[\![]\!]}(c_1, c_2)$. This gives us 15 C-C features and 15 C-all features.

7.2 Experiments

We use the STS 2012 train/test data, normalized in the same way as the image captions for the denotation graph (i.e. we re-tokenize, lemmatize, and remove determiners). Table 5 shows experimental results for four models: *DKPro* is the off-the-shelf DKProSimilarity model (Bär et al., 2013). From our corpus, we either add additive and multiplicative *compositional* features (Σ, Π) from Section 6 (img), the C-C and C-All *denotational* features based on $nPMI_{[\![]\!]}$, or *both* compositional and denotational features. Systems are evaluated by the Pearson correlation (r) of their predicted similarity scores to the human-annotated ones. We see that the denotational similarities outperform the compositional similarities, and that including compositional similarity features in addition to denotational similarity features has little effect. For additional comparison, the published numbers for the TakeLab Semantic Text Similarity System (Šarić et al., 2012), another top-performing model from the 2012 shared task, are $r = 0.880$ on this dataset.

8 Conclusion

Summary of Contributions We have defined novel *denotational* metrics of linguistic similarity (Section 2), and have shown them to be at least competitive with, if not superior to, distributional similarities for two tasks that require simple semantic inferences (Sections 6, 7), even though our current method of computing them is somewhat brittle (Section 5). We have also introduced two new resources: a large data set of images paired with descriptive captions, and a *denotation graph* that pairs generalized versions of these captions with their *visual denotations*, i.e. the sets of images they describe. Both of these resources are freely available (`http://nlp.cs.illinois.edu/Denotation.html`) Although the aim of this paper is to show their utility for a purely linguistic task, we believe that they should also be of great interest for people who aim to build systems that automatically associate image with sentences that describe them (Farhadi et al., 2010; Kulkarni et al., 2011; Li et al., 2011; Yang et al., 2011; Mitchell et al., 2012; Kuznetsova et al., 2012; Gupta et al., 2012; Hodosh et al., 2013).

Related Work and Resources We believe that the work reported in this paper has the potential to open up promising new research directions. There are other data sets that pair images or video with descriptive language, but we have not yet applied our approach to them. Chen and Dolan (2011)'s MSR Video Description Corpus (of which the STS data is a subset) is most similar to ours, but its curated part is significantly smaller. Instead of several independent captions, Grubinger et al. (2006)'s IAPR TC-12 data set contains longer descriptions. Ordonez et al. (2011) harvested 1 million images and their user-generated captions from Flickr to create the SBU Captioned Photo Dataset. These captions tend to be less descriptive of the image. The denotation graph is similar to Berant et al. (2012)'s 'entailment graph', but differs from it in two ways: first, entailment relations in the denotation graph are defined extensionally in terms of the images described by the expressions at each node, and second, nodes in Berant et al.'s entailment graph correspond to generic propositional templates (*X treats Y*), whereas nodes in our denotation graph correspond to complete propositions (*a dog runs*).

Acknowledgements

We gratefully acknowledge the support of the National Science Foundation under NSF awards 0803603 *"INT2-Medium: Understanding the meaning of images"*, 1053856 *"CAREER: Bayesian Models for Lexicalized Grammars"*, and 1205627 *"CI-P:Collaborative Research: Visual entailment data set and challenge for the Language and Vision Community"*, as well as via an NSF Graduate Research Fellowship to Alice Lai.

References

Eneko Agirre, Mona Diab, Daniel Cer, and Aitor Gonzalez-Agirre. 2012. SemEval-2012 task 6: a pilot on semantic textual similarity. In *Proceedings of the First Joint Conference on Lexical and Computational Semantics - Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation*, SemEval '12, pages 385–393.

Daniel Bär, Torsten Zesch, and Iryna Gurevych. 2013. DKPro Similarity: An Open Source Framework for Text Similarity. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 121–126, Sofia, Bulgaria, August.

Jon Barwise and John Perry. 1980. Situations and attitudes. *Journal of Philosophy*, 78:668–691.

Jonathan Berant, Ido Dagan, and Jacob Goldberger. 2012. Learning entailment relations by global graph structure optimization. *Computational Linguistics*, 38(1):73–111.

Greg Carlson, 2005. *The Encyclopedia of Language and Linguistics*, chapter Generics, Habituals and Iteratives. Elsevier, 2nd edition.

David Chen and William Dolan. 2011. Collecting highly parallel data for paraphrase evaluation. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 190–200, Portland, Oregon, USA, June.

Timothy Chklovski and Patrick Pantel. 2004. Verbocean: Mining the web for fine-grained semantic verb relations. In *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 33–40, Barcelona, Spain, July.

Kenneth Ward Church and Patrick Hanks. 1990. Word association norms, mutual information, and lexicography. *Computational Linguistics*, 16(1):22–29.

Daoud Clarke. 2009. Context-theoretic semantics for natural language: an overview. In *Proceedings of the Workshop on Geometrical Models of Natural Language Semantics*, pages 112–119, Athens, Greece, March.

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2006. The PASCAL Recognising Textual Entailment challenge. In *Machine Learning Challenges*, volume 3944 of *Lecture Notes in Computer Science*, pages 177–190. Springer.

David Dowty, Robert Wall, and Stanley Peters. 1981. *Introduction to Montague Semantics*. Reidel, Dordrecht.

Ali Farhadi, Mohsen Hejrati, Mohammad Amin Sadeghi, Peter Young, Cyrus Rashtchian, Julia Hockenmaier, and David Forsyth. 2010. Every picture tells a story: Generating sentences from images. In *Proceedings of the European Conference on Computer Vision (ECCV), Part IV*, pages 15–29, Heraklion, Greece, September.

Evgeniy Gabrilovich and Shaul Markovitch. 2007. Computing semantic relatedness using wikipedia-based explicit semantic analysis. In *Proceedings of the 20th international joint conference on Artifical intelligence*, IJCAI'07, pages 1606–1611.

Michael Grubinger, Paul Clough, Henning Müller, and Thomas Deselaers. 2006. The IAPR benchmark: A new evaluation resource for visual information systems. In *OntoImage 2006, Workshop on Language Resources for Content-based Image Retrieval during LREC 2006*, pages 13–23, Genoa, Italy, May.

Ankush Gupta, Yashaswi Verma, and C. Jawahar. 2012. Choosing linguistics over vision to describe images. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence*, Toronto, Ontario, Canada, July.

Zellig S Harris. 1954. Distributional structure. *Word*, 10:146–162.

Micah Hodosh, Peter Young, Cyrus Rashtchian, and Julia Hockenmaier. 2010. Cross-caption coreference resolution for automatic image understanding. In *Proceedings of the Fourteenth Conference on Computational Natural Language Learning*, pages 162–171, Uppsala, Sweden, July.

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 (JAIR)*, 47:853–899.

Shangfeng Hu and Chengfei Liu. 2011. Incorporating coreference resolution into word sense disambiguation. In Alexander F. Gelbukh, editor, *Computational Linguistics and Intelligent Text Processing*, volume 6608 of *Lecture Notes in Computer Science*, pages 265–276. Springer Berlin Heidelberg.

Lili Kotlerman, Ido Dagan, Idan Szpektor, and Maayan Zhitomirsky-Geffet. 2010. Directional distributional similarity for lexical inference. *Natural Language Engineering*, 16(4):359–389.

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

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, July.

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 (CoNLL)*, pages 220–228, Portland, OR, USA, June.

Dekang Lin. 1998. An information-theoretic definition of similarity. In *Proceedings of the Fifteenth International Conference on Machine Learning (ICML)*, pages 296–304, Madison, WI, USA, July.

Bill MacCartney and Christopher D. Manning. 2008. Modeling semantic containment and exclusion in natural language inference. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pages 521–528, Manchester, UK, August.

Andrew Kachites McCallum. 2002. Mallet: A machine learning for language toolkit. http://www.cs.umass.edu/ mccallum/mallet.

Shachar Mirkin, Ido Dagan, and Eyal Shnarch. 2009. Evaluating the inferential utility of lexical-semantic resources. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pages 558–566, Athens, Greece, March.

Jeff Mitchell and Mirella Lapata. 2010. Composition in distributional models of semantics. *Cognitive Science*, 34(8):1388–1429.

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 (EACL)*, pages 747–756, Avignon, France, April.

Richard Montague. 1974. *Formal philosophy: papers of Richard Montague*. Yale University Press, New Haven. Edited by Richmond H. Thomason.

Joakim Nivre, Johan Hall, and Jens Nilsson. 2006. Maltparser: A data-driven parser-generator for dependency parsing. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*, pages 2216–2219.

Vicente Ordonez, Girish Kulkarni, and Tamara L. Berg. 2011. Im2text: Describing images using 1 million captioned photographs. In *Advances in Neural Information Processing Systems 24*, pages 1143–1151.

Judita Preiss. 2001. Anaphora resolution with word sense disambiguation. In *Proceedings of SENSEVAL-2 Second International Workshop on Evaluating Word Sense Disambiguation Systems*, pages 143–146, Toulouse, France, July.

Philip Resnik. 1995. Using information content to evaluate semantic similarity in a taxonomy. In *Proceedings of the 14th international joint conference on Artificial intelligence - Volume 1*, IJCAI'95, pages 448–453.

Idan Szpektor and Ido Dagan. 2008. Learning entailment rules for unary templates. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pages 849–856, Manchester, UK, August. Coling 2008 Organizing Committee.

Frane Šarić, Goran Glavaš, Mladen Karan, Jan Šnajder, and Bojana Dalbelo Bašić. 2012. Takelab: Systems for measuring semantic text similarity. In *Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, pages 441–448, Montréal, Canada, 7-8 June.

Julie Weeds and David Weir. 2003. A general framework for distributional similarity. In *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 81–88.

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 (EMNLP)*, pages 444–454, Edinburgh, UK, July.