Multimodal Logical Inference System for Visual-Textual Entailment

Riko Suzuki[1]
suzuki.riko@is.ocha.ac.jp

Hitomi Yanaka[1,2]
hitomi.yanaka@riken.jp

Masashi Yoshikawa[3]
yoshikawa.masashi.yh8@is.naist.jp

Koji Mineshima[1]
mineshima.koji@ocha.ac.jp

Daisuke Bekki[1]
bekki@is.ocha.ac.jp

[1]Ochanomizu University, Tokyo, Japan
[2]RIKEN Center for Advanced Intelligence Project, Tokyo, Japan
[3]Nara Institute of Science and Technology, Nara, Japan

Abstract

A large amount of research about multimodal inference across text and vision has been recently developed to obtain visually grounded word and sentence representations. In this paper, we use logic-based representations as unified meaning representations for texts and images and present an unsupervised multimodal logical inference system that can effectively prove entailment relations between them. We show that by combining semantic parsing and theorem proving, the system can handle semantically complex sentences for visual-textual inference.

1 Introduction

Multimodal inference across image data and text has the potential to improve understanding information of different modalities and acquiring new knowledge. Recent studies of multimodal inference provide challenging tasks such as visual question answering (Antol et al., 2015; Hudson and Manning, 2019; Acharya et al., 2019) and visual reasoning (Suhr et al., 2017; Vu et al., 2018; Xie et al., 2018).

Grounded representations from image-text pairs are useful to solve such inference tasks. With the development of large-scale corpora such as Visual Genome (Krishna et al., 2017) and methods of automatic graph generation from an image (Xu et al., 2017; Qi et al., 2019), we can obtain structured representations for images and sentences such as scene graph (Johnson et al., 2015), a visually-grounded graph over object instances in an image.

While graph representations provide more interpretable representations for text and image than embedding them into high-dimensional vector spaces (Frome et al., 2013; Norouzi et al., 2014), there remain two challenges: (i) to capture complex logical meanings such as negation and quantification, and (ii) to perform logical inferences on them.

✗ **No** cat is next to a pumpkin. (1)
✗ There are **at least two** cats. (2)
✓**All** pumpkins are orange. (3)

Figure 1: An example of visual-textual entailment. An image paired with logically complex statements, namely, negation (1), numeral (2), and quantification (3), leads to a true (✓) or false (✗) judgement.

For example, consider the task of checking if each statement in Figure 1 is true or false under the situation described in the image. The statements (1) and (2) are false, while (3) is true. To perform this task, it is necessary to handle semantically complex phenomena such as negation, numeral, and quantification.

To enable such advanced visual-textual inferences, it is desirable to build a framework for representing richer semantic contents of texts and images and handling inference between them. We use logic-based representations as unified meaning representations for texts and images and present an unsupervised inference system that can prove entailment relations between them. Our visual-textual inference system combines semantic parsing via Combinatory Categorial Grammar (CCG; Steedman (2000)) and first-order theorem proving (Blackburn and Bos, 2005). To describe information in images as logical formulas, we propose a method of transforming graph representations into logical formulas, using the idea of predicate circumscription (McCarthy, 1986), which complements information implicit in images using the closed world assumption. Experiments show that our system can perform visual-textual inference with semantically complex sentences.

Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop, pages 386–392
Florence, Italy, July 28 - August 2, 2019.

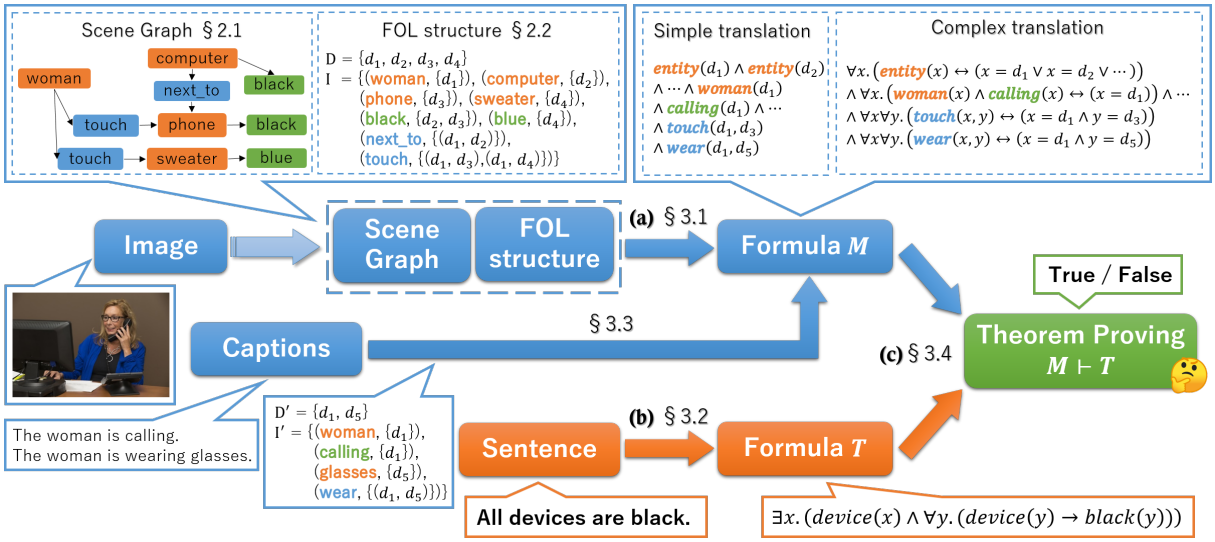

Figure 2: Overview of the proposed system. In this work, we assume the input image is processed into an FOL structure or scene graph a priori. The system consists of three parts: (a) **Graph Translator** converts an image annotated with a scene graph/FOL structure to formula M; (b) **Semantic parser** maps a sentence to formula T via CCG parsing; (c) **Inference Engine** checks whether M entails T by FOL theorem proving.

2 Background

There are two types of grounded meaning representations for images: scene graphs and first-order logic (FOL) structures. Both characterize objects and their semantic relationships in images.

2.1 Scene Graph

A *scene graph*, as proposed in Johnson et al. (2015), is a graphical representation that depicts objects, their attributes, and relations among them occurring in an image. An example is given in Figure 2. Nodes in a scene graph correspond to objects with their categories (e.g. *woman*) and edges correspond to the relationships between objects (e.g. *touch*). Such a graphical representation has been shown to be useful in high-level tasks such as image retrieval (Johnson et al., 2015; Schuster et al., 2015) and visual question answering (Teney et al., 2017). Our proposed method builds on the idea that these graph representations can be translated into logical formulas and be used in complex logical reasoning.

2.2 FOL Structure

In logic-based approaches to semantic representations, *FOL structures* (also called *FOL models*) are used to represent semantic information in images (Hürlimann and Bos, 2016), An FOL structure is a pair (D, I) where D is a domain (also called *universe*) consisting of all the entities in an image and I is an interpretation function that maps a 1-place predicate to a set of entities and a 2-place predicate to a set of pairs of entities, and so on; for instance, we write $I(\mathsf{man}) = \{d_1\}$ if the entity d_1 is a man, and $I(\mathsf{next\_to}) = \{(d_1, d_2)\}$ if d_1 is next to d_2. FOL structures have clear correspondence with the graph representations of images in that they both capture the categories, attributes and relations holding of the entities in an image. For instance, the FOL structure and scene graph in the upper left of Figure 2 have exactly the same information. Thus, the translation from graphs to formulas can also work for FOL structures (see §3.1).

3 Multimodal Logical Inference System

Figure 2 shows the overall picture of the proposed system. We use formulas of FOL with equality as unified semantic representations for text and image information. We use 1-place and 2-place predicates for representing attributes and relations, respectively. The language of FOL consists of (i) a set of atomic formulas, (ii) equations of the form $t = u$, and (iii) complex formulas composed of negation ($\neg$), conjunction ($\wedge$), disjunction ($\vee$), implication ($\rightarrow$), and universal and existential quantification ($\forall$ and $\exists$). The expressive power of the FOL language provides a structured representation that captures not only objects and their semantic relationships but also those complex expressions including negation, quantification and numerals.

$$\mathrm{Tr}_s(D) = \mathsf{entity}(d_1) \wedge \ldots \wedge \mathsf{entity}(d_n)$$

$$\mathrm{Tr}_s(P) = P(d_1) \wedge \ldots \wedge P(d_{n'})$$

$$\mathrm{Tr}_s(R) = R(d_{i_1}, d_{j_1}) \wedge \ldots \wedge R(d_{i_n}, d_{j_n})$$

$$\mathrm{Tr}_c(D) = \forall x.(\mathsf{entity}(x) \leftrightarrow (x = d_1 \vee \ldots \vee x = d_n))$$

$$\mathrm{Tr}_c(P) = \forall x.(P(x) \leftrightarrow (x = d_1 \vee \ldots \vee x = d_{n'}))$$

$$\mathrm{Tr}_c(R) = \forall x \forall y.(R(x,y) \leftrightarrow ((x = d_{i_1} \wedge y = d_{j_1}) \vee \ldots \vee (x = d_{i_m} \wedge y = d_{j_m})))$$

Table 1: Definition of two types of translation, TR_s and TR_c. Here we assume that $D = \{d_1, \ldots, d_n\}$, $P = \{d_1, \ldots, d_{n'}\}$, and $R = \{(d_{i_1}, d_{j_1}), \ldots, (d_{i_m}, d_{j_m})\}$.

The system takes as input an image I and a sentence S and determines whether I entails S, in other words, S is true with respect to the situation described in I. In this work, we assume the input image I is processed into a scene graph/FOL structure G_I using an off-the-shelf converter (Xu et al., 2017; Qi et al., 2019).

To determine entailment relations between sentences and images, we proceed in three steps. First, **graph translator** maps a graph G_I to a formula M. We develop two ways of translating graphs to FOL formulas (§3.1). Second, **semantic parser** takes a sentence S as input and return a formula T via CCG parsing. We improve a semantic parser in CCG for handling numerals and quantification (§3.2). Additionally, we develop a method for utilizing image captions to extend G_I with information obtainable from their logical formulas (§3.3). Third, **inference engine** checks whether M entails T, written $M \vdash T$, using FOL theorem prover (§3.4). Note that FOL theorem provers can accept multiple premises, $M_1, \ldots, M_n$, converted from images and/or sentences and check if $M_1, \ldots, M_n \vdash T$ holds or not. Here we focus on single-premise visual inference.

3.1 Graph Translator

We present two ways of translating graphs (or equivalently, FOL structures) to formulas: a simple translation (Tr_s) and a complex translation (Tr_c). These translations are defined in Table 1. For example, consider a graph consisting of the domain $D = \{d_1, d_2\}$, where we have $\mathsf{man}(d_1), \mathsf{hat}(d_2), \mathsf{red}(d_2)$ as properties and $\mathsf{wear}(d_1, d_2)$ as relations. The simple translation TR_s gives the formula (S) below, which simply conjoins all the atomic information.

(S) $\mathsf{man}(d_1) \wedge \mathsf{hat}(d_2) \wedge \mathsf{red}(d_2) \wedge \mathsf{wear}(d_1, d_2)$

1. $A \in \mathcal{P}$, $\neg A \in \mathcal{N}$, if A is an atomic formula.
2. A, $\neg A \in \mathcal{P}$, if A is an equation of the form $t = u$.
3. $A \wedge B$, $A \vee B \in \mathcal{P}$, if $A \in \mathcal{P}$ and $B \in \mathcal{P}$.
4. $A \wedge B$, $A \vee B \in \mathcal{N}$, if $A \in \mathcal{N}$ or $B \in \mathcal{N}$.
5. $A \to B \in \mathcal{P}$, if $A \in \mathcal{N}$ and $B \in \mathcal{P}$.
6. $A \to B \in \mathcal{N}$, if $A \in \mathcal{P}$ or $B \in \mathcal{N}$.
7. $\forall x.A$, $\exists x.A \in \mathcal{P}$, if $A \in \mathcal{P}$.
8. $\forall x.A$, $\exists x.A \in \mathcal{N}$, if $A \in \mathcal{N}$.

Table 2: Positive ($\mathcal{P}$) and negative ($\mathcal{N}$) formulas

However, this does not capture the *negative* information that d_1 is the only entity that has the property man; similarly for the other predicates. To capture it, we use the complex translation Tr_c, which gives the following formula:

(C) $\forall x.(\mathsf{man}(x) \leftrightarrow x = d_1) \wedge$
$\forall y.(\mathsf{hat}(y) \leftrightarrow y = d_2) \wedge$
$\forall z.(\mathsf{red}(z) \leftrightarrow z = d_2) \wedge$
$\forall x \forall y.(\mathsf{wear}(x,y) \leftrightarrow (x = d_1 \wedge y = d_2))$

This formula says that d_1 is the only man in the domain, d_2 is the only hat in the domain, and so on. This way of translation can be regarded as an instance of Predicate Circumscription (McCarthy, 1986), which complement negative information using the closed world assumption. The translation Tr_c is useful for handling formulas with negation and universal quantification.

One drawback here is that since (C) involves complex formulas, it increases the computational cost in theorem proving. To remedy this problem, we use two types of translation selectively, depending on the polarity of the formula to be proved. Table 2 shows the definition to classify each FOL formula $A \in \mathcal{L}$ into positive ($\mathcal{P}$) and negative ($\mathcal{N}$) one. For instance, the formulas $\exists x \exists y.(\mathsf{cat}(x) \wedge \mathsf{dog} \wedge \mathsf{touch}(x,y))$, which correspond to *A cat touches a dog*, is a positive formula, while $\neg \exists x.(\mathsf{cat}(x) \wedge \mathsf{white}(x))$, which corresponds to *No cats are white*, is a negative formula.

3.2 Semantic Parser

We use ccg2lambda (Mineshima et al., 2015), a semantic parsing system based on CCG to convert sentences to formulas, and extend it to handle numerals and quantificational sentences. In our system, a sentence with numerals, e.g., *There are (at least) two cats*, is compositionally mapped to the following FOL formula:

(Num) $\exists x \exists y.(\mathsf{cat}(x) \wedge \mathsf{cat}(y) \wedge (x \neq y))$

Also, to capture the existential import of universal sentences, the system maps the sentence *All cats are white* to the following one:

$$(\mathbf{Q}) \quad \exists x.\mathsf{cat}(x) \wedge \forall y.(\mathsf{cat}(y) \rightarrow \mathsf{white}(y))$$

3.3 Extending Graphs with Captions

Compared with images, captions can describe a variety of properties and relations other than spatial and visual ones. By integrating caption information into FOL structures, we can obtain semantic representations reflecting relations that can be described only in the caption.

We convert captions into FOL structures (= graphs) using our semantic parser. We only consider the cases where the formulas obtained are composed of existential quantifiers and conjunctions. For extending FOL structures with caption information, it is necessary to analyze co-reference between the entities occurring in sentences and images. We add a new predicate to an FOL structure if the co-reference is uniquely determined.

As an illustration, consider the captions and the FOL structure (D, I) which represents the image shown in Figure 2.[1] The captions, (1a) and (2a), are mapped to the formulas (1a) and (2b), respectively, via semantic parsing.

(1) a. The woman is calling.
b. $\exists x.(\mathsf{woman}(x) \wedge \mathsf{calling}(x))$

(2) a. The woman is wearing glasses.
b. $\exists x \exists y.(\mathsf{woman}(x) \wedge \mathsf{glasses}(y) \wedge \mathsf{wear}(x, y))$

Then, the information in (1b) and (2b) can be added to (D, I), because there is only one woman d_1 in (D, I) and thus the co-reference between *the woman* in the caption and the entity d_1 is uniquely determined. Also, a new entity d_5 for glasses is added because there are no such entities in the structure (D, I). Thus we obtain the following new structure (D^*, I^*) extended with the information in the captions.

$$D^* := D \cup \{d_5\}$$
$$I^* := I \cup \{(\mathsf{glasses}, \{d_5\}), (\mathsf{calling}, \{d_1\}), (\mathsf{wear}, \{(d_1, d_5)\})\}$$

[1]Note that there is a unique correspondence between FOL structures and scene graphs. For the sake of illustration, we use FOL structures in this subsection.

Pattern	Phenomena
There is a $\langle attr \rangle$ $\langle attr \rangle$ $\langle obj \rangle$.	Con
There are at least $\langle number \rangle$ $\langle obj \rangle$.	Num
All $\langle obj \rangle$ are $\langle attr \rangle$.	Q
$\langle obj \rangle$ $\langle rel \rangle$ $\langle obj \rangle$.	Rel
No $\langle obj \rangle$ is $\langle attr \rangle$.	Neg
All $\langle obj \rangle$ $\langle attr \rangle$ or $\langle attr \rangle$.	Con, Q
Every $\langle obj \rangle$ is not $\langle rel \rangle$ $\langle obj \rangle$.	Num, Rel, Neg

Table 3: Examples of sentence templates. $\langle obj \rangle$: objects, $\langle attr \rangle$: attributes, $\langle rel \rangle$: relations.

3.4 Inference Engine

Theorem prover is a method for judging whether a formula M entails a formula T. We use Prover9[2] as an FOL prover for inference. We set timeout (10 sec) to judge that M does not entail T.

4 Experiment

We evaluate the performance of the proposed visual-textual inference system. Concretely, we formulate our task as image retrieval using query sentences and evaluate the performance in terms of the number of correctly returned images. In particular, we focus on semantically complex sentences containing numerals, quantifiers, and negation, which are difficult for existing graph representations to handle.

Dataset: We use two datasets: Visual Genome (Krishna et al., 2017), which contains pairs of scene graphs and images, and GRIM dataset (Hürlimann and Bos, 2016), which annotates an FOL structure of an image and two types of captions (true and false sentences with respect to the image). Note that our system is fully unsupervised and does not require any training data; in the following, we describe only test set creation procedure.

For the experiment using Visual Genome, we randomly extracted 200 images as test data, and a separate set of 4,000 scene graphs for creating query sentences; we made queries by the following steps. First, we prepared sentence templates focusing on five types of linguistic phenomena: logical connective (**Con**), numeral (**Num**), quantifier (**Q**), relation (**Rel**) and negation (**Neg**). See Table 3 for the templates. Then, we manually extracted object, attribute and relation types from the frequent ones (appearing more than 30 times) in the extracted 4,000 graphs, and created queries by

[2]http://www.cs.unm.edu/ mccune/prover9/

Sentences	Phenomena	Count
There is a long red bus.	Con	3
There are at least three men.	Num	32
All windows are closed.	Q	53
Every green tree is tall.	Q	18
A man is wearing a hat.	Rel	12
No umbrella is colorful.	Neg	197
There is a train which is not red.	Neg	6
There are two cups or three cups.	Con, Num	5
All hairs are black or brown.	Con, Q	46
A gray or black pole has two signs.	Con, Num, Rel	6
Three cars are not red.	Num, Neg	28
All women wear a hat.	Q, Rel	2
A man is not walking on a street.	Rel, Neg	76
A clock on a tower is not black.	Rel, Neg	7
Two women aren't having black hair.	Num, Rel, Neg	10
Every man isn't eating anything.	Q, Rel, Neg	67

Table 4: Examples of query sentences In §4.1; Count shows the number of images describing situations under which each sentence is true.

replacing $\langle obj \rangle$, $\langle attr \rangle$ and $\langle rel \rangle$ in the templates with them. As a result, we obtained 37 semantically complex queries as shown in Table 4. To assign correct images to each query, two annotators judged whether each of the test images entails the query sentence. If the two judgments disagreed, the first author decided the correct label.

In the experiment using GRIM, we adopted the same procedure to create a test dataset and obtained 19 query sentences and 194 images.

One of the issues in this dataset is that annotated FOL structures contain only spatial relations such as *next_to* and *near*; to handle queries containing general relations such as *play* and *sing*, our system needs to utilize annotated captions (§3.3). To evaluate if our system can effectively extract information from captions, we split **Rel** of above linguistic phenomena into spatial relation (**Spa-Rel**; relations about spatial information) and general relation (**Gen-Rel**; other relations), and report the scores separately in terms of these categories.

4.1 Experimental Results on Visual Genome

Firstly, we evaluate the performance in terms of our **Graph translator**'s conversion algorithm. As described in §3.1, there are two translation algorithms; simple one that conjunctively enumerates all relation in a graph (**SIMPLE** in the following), and one that selectively employs translation based on Predicate Circumscription (**HYBRID**).

Table 5 shows image retrieval scores per linguistic phenomenon, macro averages of F1 scores of queries labeled with the respective phenomena.

Phenomena (#)		SIMPLE	HYBRID
Con	(17)	36.40	41.66
Num	(9)	43.07	45.45
Q	(9)	8.59	**28.18**
Rel	(11)	25.13	**35.10**
Neg	(11)	66.38	**73.39**

Table 5: Experimental results on Visual Genome (F1). "#" stands for the number of query sentences categorized into that phenomenon.

HYBRID shows better performance for all phenomena than SIMPLE one, improving by 19.59% on **Q**, 9.97% on **Rel** and 7.01% on **Neg**, over SIMPLE, suggesting that the proposed complex translation is useful for inference using semantically complex sentences including quantifier and negation. Figure 3 shows retrieved results for a query (a) *Every green tree is tall* and (b) *No umbrella is colorful*, each containing universal quantifier and negation, respectively. Our system successfully performs inference on these queries, returning the correct images, while excluding wrong ones (note that the third picture in (a) contains short trees).

(a) *Every green tree is tall.*

(b) *No umbrella is colorful.*

Figure 3: Predicted images of our system; Images in green entail the queries, while those in red do not.

Error Analysis: One of the reasons for the lower F1 of **Q** is the gap of annotation rule between Visual Genome and our test set. Quantifiers in natural language often involve vagueness (Pezzelle et al., 2018). for example, the interpretation of *everyone* depends on what counts as an entity in the domain. Difficulties in fixing the interpretation of quantifiers caused the lower performance.

The low F1 in **Rel** is primarily due to lexical gaps between formulas of a query and an image. For example, sentences *All women wear a hat* and *All women have a hat* are the same in their meaning. However, if a scene graph contains only *wear*

relation, our system can handle the former query, while not the other. In future work, we will extend our system with a knowledge insertion mechanism (Martínez-Gómez et al., 2017).

4.2 Experimental Results on GRIM

We test our system on GRIM dataset. As noted above, the main issue on this dataset is the lack of relations other than spatial ones. We evaluate if our system can be enhanced using the information contained in captions. The F1 scores of the Hybrid system with captions are the same with the one without captions on the sets except for **Gen-Rel**;[3] on the subset, the F1 score of the former improves by 60% compared to the latter, which suggests that captions can be integrated into FOL structures for the improved performance.

5 Conclusion

We have proposed a logic-based system to achieve advanced visual-textual inference, demonstrating the importance of building a framework for representing the richer semantic content of texts and images. In the experiment, we have shown that our CCG-based pipeline system, consisting of graph translator, semantic parser and inference engine, can perform visual-textual inference with semantically complex sentences, without requiring any supervised data.

Acknowledgement

We thank the two anonymous reviewers for their encouragement and insightful comments. This work was partially supported by JST CREST Grant Number JPMJCR1301, Japan.

References

Manoj Acharya, Kushal Kafle, and Christopher Kanan. 2019. TallyQA: Answering complex counting questions. In *The Association for the Advancement of Artificial Intelligence (AAAI2019)*.

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. 2015. VQA: Visual Question Answering. In *International Conference on Computer Vision*.

Patrick Blackburn and Johan Bos. 2005. *Representation and Inference for Natural Language: A First Course in Computational Semantics*. Center for the Study of Language and Information, Stanford, CA, USA.

[3] **Con**: 91.41%, **Num**: 95.24%, **Q**: 78.84%, **Spa-Rel**: 88.57%, **Neg**: 62.57%.

Andrea Frome, Greg S Corrado, Jon Shlens, Samy Bengio, Jeff Dean, Marc Aurelio Ranzato, and Tomas Mikolov. 2013. DeViSE: A Deep Visual-Semantic Embedding Model. In *Neural Information Processing Systems conference*, pages 2121–2129.

Drew A. Hudson and Christopher D. Manning. 2019. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Manuela Hürlimann and Johan Bos. 2016. Combining lexical and spatial knowledge to predict spatial relations between objects in images. In *Proceedings of the 5th Workshop on Vision and Language*, pages 10–18. Association for Computational Linguistics.

Justin Johnson, Ranjay Krishna, Michael Stark, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Li Fei-Fei. 2015. Image retrieval using scene graphs. In *IEEE/ CVF International Conference on Computer Vision and Pattern Recognition*, pages 3668–3678. IEEE Computer Society.

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, Michael Bernstein, and Li Fei-Fei. 2017. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123(1):32–73.

Pascual Martínez-Gómez, Koji Mineshima, Yusuke Miyao, and Daisuke Bekki. 2017. On-demand Injection of Lexical Knowledge for Recognising Textual Entailment. In *Proceedings of The European Chapter of the Association for Computational Linguistics*, pages 710–720.

John McCarthy. 1986. Applications of circumscription to formalizing common-sense knowledge. *Artificial Intelligence*, 28(1):89–116.

Koji Mineshima, Pascual Martínez-Gómez, Yusuke Miyao, and Daisuke Bekki. 2015. Higher-order logical inference with compositional semantics. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2055–2061. Association for Computational Linguistics.

Mohammad Norouzi, Tomas Mikolov, Samy Bengio, Yoram Singer, Jonathon Shlens, Andrea Frome, Greg Corrado, and Jeffrey Dean. 2014. Zero-Shot Learning by Convex Combination of Semantic Embeddings. In *International Conference on Learning Representations*.

Sandro Pezzelle, Ionut-Teodor Sorodoc, and Raffaella Bernardi. 2018. Comparatives, quantifiers, proportions: a multi-task model for the learning of quantities from vision. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 419–430. Association for Computational Linguistics.

Mengshi Qi, Weijian Li, Zhengyuan Yang, Yunhong Wang, and Jiebo Luo. 2019. Attentive relational networks for mapping images to scene graphs. In *The IEEE Conference on Computer Vision and Pattern Recognition*.

Sebastian Schuster, Ranjay Krishna, Angel Chang, Li Fei-Fei, and Christopher D. Manning. 2015. Generating semantically precise scene graphs from textual descriptions for improved image retrieval. In *Proceedings of the Fourth Workshop on Vision and Language*, pages 70–80. Association for Computational Linguistics.

Mark Steedman. 2000. *The Syntactic Process*. MIT Press, Cambridge, MA, USA.

Alane Suhr, Mike Lewis, James Yeh, and Yoav Artzi. 2017. A corpus of natural language for visual reasoning. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 217–223, Vancouver, Canada. Association for Computational Linguistics.

Damien Teney, Lingqiao Liu, and Anton van den Hengel. 2017. Graph-structured representations for visual question answering. In *The IEEE Conference on Computer Vision and Pattern Recognition*, pages 3233–3241.

Hoa Trong Vu, Claudio Greco, Aliia Erofeeva, Somayeh Jafaritazehjan, Guido Linders, Marc Tanti, Alberto Testoni, Raffaella Bernardi, and Albert Gatt. 2018. Grounded textual entailment. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2354–2368.

Ning Xie, Farley Lai, Derek Doran, and Asim Kadav. 2018. Visual entailment task for visually-grounded language learning. *arXiv preprint arXiv:1811.10582*.

Danfei Xu, Yuke Zhu, Christopher B. Choy, and Li Fei-Fei. 2017. Scene Graph Generation by Iterative Message Passing. In *The IEEE Conference on Computer Vision and Pattern Recognition*.