CLEVR Parser: A Graph Parser Library for Geometric Learning on Language Grounded Image Scenes

Raeid Saqur[1,2]
[1]University of Toronto Computer Science
[2]Vector Institute for Artificial Intelligence
raeidsaqur@cs.toronto.edu

Ameet Deshpande
Department of Computer Science
Princeton University
asd@cs.princeton.edu

Abstract

The CLEVR dataset has been used extensively in language grounded visual reasoning in Machine Learning (ML) and Natural Language Processing (NLP) domains. We present a **graph parser library** for CLEVR, that provides functionalities for object-centric attributes and relationships extraction, and construction of structural graph representations for dual modalities. Structural order-invariant representations enable geometric learning and can aid in downstream tasks like language grounding to vision, robotics, compositionality, interpretability, and computational grammar construction. We provide three extensible main components – **parser, embedder, and visualizer** that can be tailored to suit specific learning setups. We also provide out-of-the-box functionality for seamless integration with popular deep graph neural network (GNN) libraries. Additionally, we discuss downstream usage and applications of the library, and how it accelerates research for the NLP research community[1].

1 Introduction

The CLEVR dataset (Johnson et al., 2017a) is a modern 3D incarnation of historically significant shapes-based datasets like SHRDLU (Winograd, 1970), used for demonstrating AI efficacy on language understanding (Ontanon, 2018; Winograd, 1980; Hudson and Manning, 2018). Although originally aimed at the visual question answering (VQA) problem (Santoro et al., 2017; Hu et al., 2018), its versatility has seen its use in diverse ML domains, including extensions to physics simulation engines for language augmented hierarchical reinforcement learning (Jiang et al., 2019) and causal reasoning (Yi et al., 2019).

---
[1]Code is available at - https://github.com/raeidsaqur/clevr-parser

(a) Question on image (Figure 2): 'Is the color of the *metal block* that is *right* of the *yellow rubber object* the same as the *large metal cylinder*?'
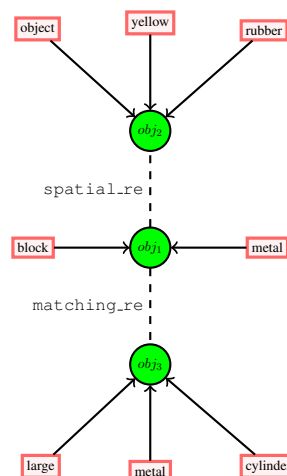
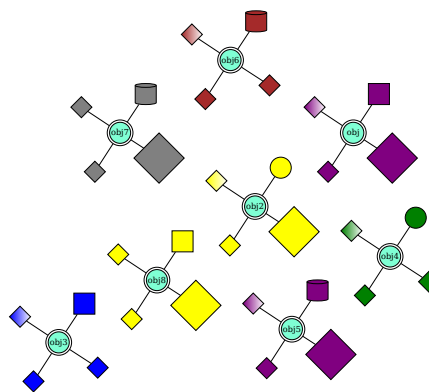(b) Image (Figure 2) scene graph parsed representation

Figure 1: A question about a CLEVR image visualized as multimodal parsed graphs

Parallelly, research interest in geometric learning and GNN (Kipf and Welling, 2016; Schlichtkrull et al., 2018; Hamilton et al., 2017) based techniques have seen a dramatic surge in recent deep learning zeitgeist. In this focused paper, we present a library that allows easy integration and application of geometric representation learning on CLEVR dataset

tasks - enabling the NLP research community to apply GNN based techniques to their research (see 4).

The library has three main (extensible) components: 1. **Parser**: allows extraction of graph structured relationships among objects of the environment – both for textual questions, and semantic image scene graphs, 2. **Embedder**: allows generation of latent embeddings using any models or desired backend of choice (like PyTorch[2]), 3. **Visualizer**: provides tools for visualizing structural graphs and latent embeddings.

2 Background

CLEVR Environment The dataset consists of images with rendered 3D objects of various shapes, colors, materials, and sizes, along with corresponding image scene graphs containing visual semantic information. Templated question generation on the images allows the creation of complex questions that test various aspects of scene understanding. The original dataset contains $\approx$1M questions generated from $\approx$100k questions with 90 question template families that can be broadly categorized into five question types: count, exist, numerical comparison, attribute comparison, and query.
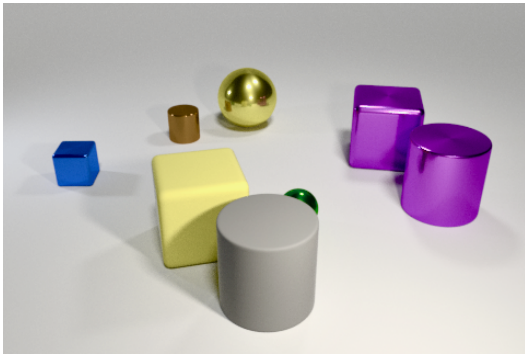

Figure 2: A CLEVR image

The dataset also comes with a defined domain-specific-language (DSL) function library $\mathcal{F}$, containing primitive functions that can be composed together to answer questions on CLEVR images (Johnson et al., 2017b). We delegate further details of this dataset to (Johnson et al., 2017a) and the appendix A.

3 CLEVR-PARSER

Here we describe each of the main library components in detail.

3.1 Parser

Text The parser takes a language utterance, which can be a question, caption or command, that is valid in the CLEVR environment, and outputs a structural graph representation – G_s, capturing object attributes, spatial relationships (spatial_re), and attribute similarity based matching predicates (matching_re) in the textual input. This is implemented by adding a CLEVR object entity recognizer (NER) in the NLP parse pipeline as depicted by Figure 3. Note that the NER is permutationally equivariant to the object attributes – i.e. a 'large red rubber ball' will be detected as an object by any of these spans: 'red large rubber ball', 'large ball', 'ball' etc.

Figure 3: Entity visualization

Images The parser takes image scene graphs as input and outputs a structural graph – G_t. The synthesized image scenes accompanying the original dataset can be used as input. Alternatively, parsed image scenes generated using any modern semantic image segmentation method (for e.g. 'Mask-RCNN' (He et al., 2017)) can also be used as input (Yi et al., 2018). A visualized example of a parsed image is shown in figure 4a. For the ease of reproducibility, we also include a curated dataset '1obj' with parsed image scenes using Mask-RCNN semantic segmentation (AppendixA).

While we provide a concrete implementation using the SpaCy[3] NLP library, any other library like the Stanford Parser[4], or NLTK[5] could be used in its place. The output of the parser from a question and image is depicted in Figure 1.

3.2 Embedder

The embedder provides 'word-embedding' (Mikolov et al., 2017) based representation of input text utterances and image scenes using a pre-trained language model (LM). The end-user can instantiate the embedder with a preferred LM, which could be a simple one-hot representation of the CLEVR environment vocabulary, or a large transformer based SotA LMs like BERT, GPT-2, XLNet (Peters et al., 2018; Devlin et al.,

2018; Radford et al., 2019; Yang et al., 2019). The embedder uses the parser (see section 3.1) generated graphs $\mathcal{G}_s, \mathcal{G}_t$ – where graph $\mathcal{G}_s$ and $\mathcal{G}_t$ are defined as generic graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{A})$, where $\mathcal{V}$ is the set of nodes $\{1,2,..\}$, $\mathcal{E}$ is the set of edges, and $\mathcal{A}$ is the adjacency matrix – and returns $\mathcal{X}$, E, the feature matrices of the nodes and edges respectively:

$$\begin{aligned} \mathcal{X}_s, A_s, E_s &\leftarrow \text{EMBED}(S) \\ \mathcal{X}_t, A_t, E_t &\leftarrow \text{EMBED}(T), \end{aligned} \quad (1)$$

The output signature of the embedder is a tuple: ($\mathcal{X}$, A, E), which matches the fundamental data-structure of popular geometric learning libraries like PyTorch Geometric (Fey and Lenssen, 2019), thus allowing seamless integration. We show a concrete implementation of this use case using **PyTorch Geometric** (Fey and Lenssen, 2019) and Pytorch in 3.3.2.

3.3 Visualizer

We provide multiple visualization tools for analyzing images, text, and latent embeddings.

3.3.1 Visualizing Structural Graphs

This visualizer sub-component enables visualization of the multimodal structural graph outputs – G_s, G_t – by the parser (see 3.1) using `Graphviz` and `matplotlib`.

Visualizing Images   Image graphs (G_t) can have a large number of objects and attributes. For ease of viewing, attributes like size, shape (e.g. cylinder), color (e.g. yellow), and material (e.g. metallic) are displayed as nodes of the graph (Figure 4a). We explain elements of Figure 4a to describe the **legend** in greater detail. The double circles represent the objects, and the adjacent nodes are their attributes. The *shape* is depicted using the actual shape (e.g. the cyan cylinder – $obj2$), and the other attributes are depicted as diamonds. The *size* of one of the diamonds depicts if the object is small or large, e.g. the large cyan diamond attached to $obj2$ means that it is large. The *color* of all the attribute nodes depicts the color of the object (e.g. the cyan color of $obj2$). The presence of a gradient in the remaining diamond depicts the *material* of the object. For example, the gradient in the diamond attached to $obj4$ means that it is *metallic*, and the solid fill for $obj2$ means that it is *rubber*. While this legend is a little lengthy, we found that it makes visualiza-

tion easier, but the user can choose to revert to the simpler setting of using text to depict the attributes.

Visualizing Text   Text corresponding to an image is a partially observable subset of objects, their relationships, and attributes. The dependency graph of the text is visualized just like the images, with only the observable information being depicted (Figure 4b).

Composing image and text   We also provide an option to view an image and the text in the same graph. By connecting corresponding object nodes from the image and text, we create a bipartite graph that allows us to visualize all the information that an image-text pair contains (Figure 4c). Additional examples from the visualizer are presented in appendix A.4.

3.3.2 Visualizer - Embeddings

We also provide a visualizer to analyze the embeddings produced by using methods in section 3.2. We use t-SNE (Maaten and Hinton, 2008), which is a method used to visualize high-dimensional data on 2 or 3 dimensions. We also offer clustering support to allow grouping of similar embeddings together. Both image (Frome et al., 2013) and word embeddings (Mikolov et al., 2013) from learned models have the nice property of capturing semantic information, and our visualizers capture this semantic similarity information in the form of clusters.

Figure 5 plots the embeddings for questions drawn from two different distributions *train* and *test*, which represent semantically different sequences, and they separate out into distinct clusters.
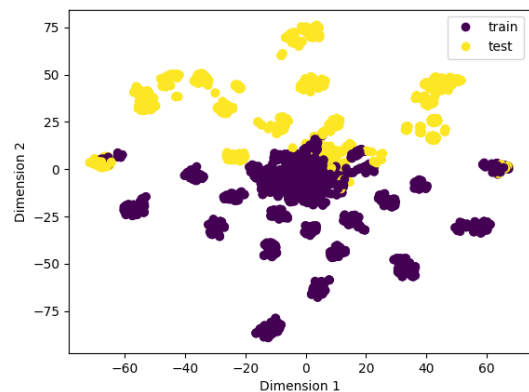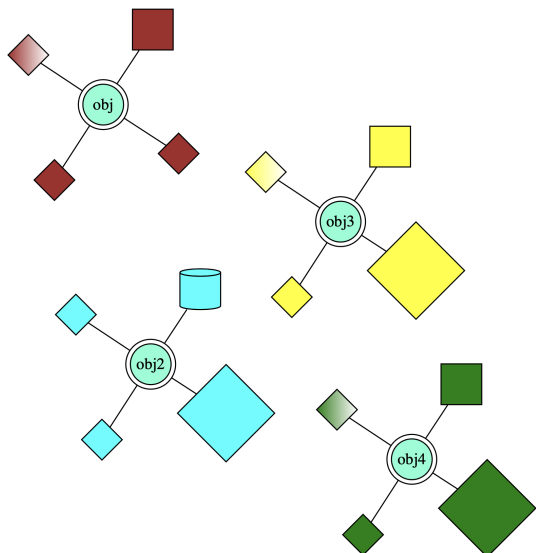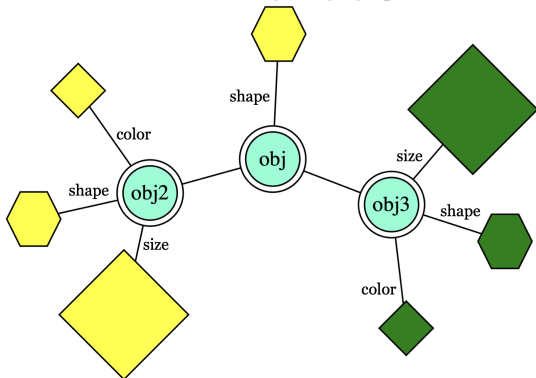

Figure 5: Questions from two different distributions which form separate clusters
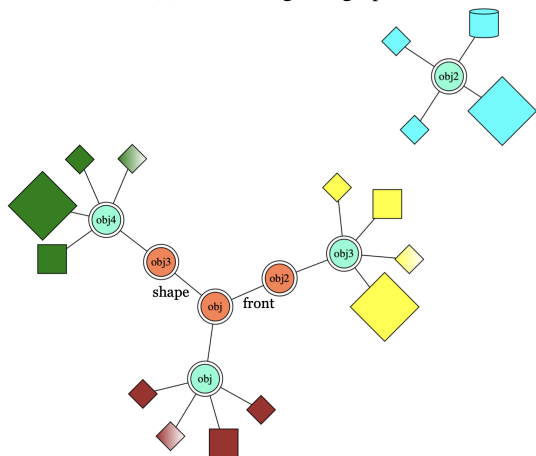
Similarly, Figure 6 analyzes embeddings drawn from 7 different templates. Questions that corre-

(a) Visualizing image graph – G_t

What is the size of the thing that is in front of the big yellow object and is the same shape as the big green thing?

(b) Visualizing text graph – G_s

(c) Visualizing joint (image and text) graph – G_u for the above two figures

Figure 4: Visualizing G_s, G_t, G_u

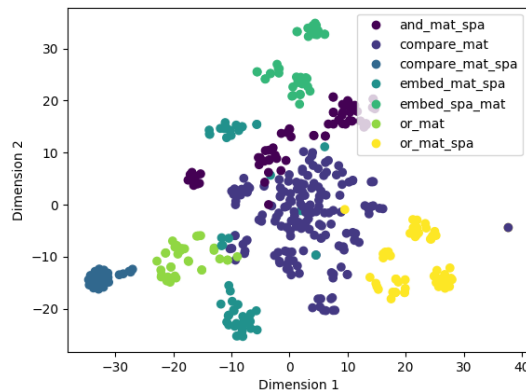spond to the same templates form tight clusters while being far away from other questions.

Figure 6: Questions from 7 different templates forming tight clusters

4 Related Work and Applications

Some lines of work attempt to generate scene graphs for images. The Visual Genome library (Krishna et al., 2017), in a real-world image setting, is a collection of annotated images (from Flickr, COCO) and corresponding knowledge graph associations. The work of (Schuster et al., 2015) and the corresponding library which is a part of the Stanford NLP library[6], allows scene graph generation from text (image caption) as input.

Our work is orthogonal to these in that our target dataset is synthetic, which allows full control over the generation of images, questions, and ground truth semantic program chains. Thus, coalesced with our library's functionalities, it allows end-to-end (e2e) control over experimenting on every modular aspect of research hypotheses (see 4.1). Further, our work premises on providing multimodal representations – including ground-truth paired graph (joint graph $G_u \leftarrow (G_s, G_t)$) – which has interesting downstream research applications.

4.1 Usages and Applications

Applications of language grounding in ML/NLP research are quite broad. To avoid sounding overly grandiose, we exemplify possible applications citing work that pertains to the CLEVR dataset.

Recent work by (Bahdanau et al., 2019) has shown lack of distributional robustness and compositional generalization (Fodor et al., 1988) in NLP. Permutation equivariance within local linguistic component groups has been shown to help with language compositionality (Gordon et al., 2020). Graph-based representations are intrinsically or-

---

[6]https://nlp.stanford.edu/software/scenegraph-parser.shtml

der invariant – thus, may help with language compositionality research. Language augmented reward mechanisms are a dense topic in concurrent (human-in-the-loop) reinforcement learning (Knox and Stone, 2012; Griffith et al., 2013), robotics (Knox et al., 2013; Kuhlmann et al., 2004), long-horizon, hierarchical POMDP problems in general (Kaplan et al., 2017) – like command completion in physics simulators (Jiang et al., 2019). Other applications could be in program synthesis and interpretability (Mascharka et al., 2018), causal reasoning (Yao, 2010), and general visually grounded language understanding (Yu et al., 2016).

In general, we expect and hope that any existing line or domain of work in NLP using the CLEVR dataset (hundreds, based on citations), will benefit from having graph-based representational learning aided by our proposed library.

References

Dzmitry Bahdanau, Harm de Vries, Timothy J O'Donnell, Shikhar Murty, Philippe Beaudoin, Yoshua Bengio, and Aaron Courville. 2019. Closure: Assessing systematic generalization of clevr models. *arXiv preprint arXiv:1912.05783*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Matthias Fey and Jan Eric Lenssen. 2019. Fast graph representation learning with pytorch geometric. *arXiv preprint arXiv:1903.02428*.

Jerry A Fodor, Zenon W Pylyshyn, et al. 1988. Connectionism and cognitive architecture: A critical analysis. *Cognition*, 28(1-2):3–71.

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 *Advances in neural information processing systems*, pages 2121–2129.

Jonathan Gordon, David Lopez-Paz, Marco Baroni, and Diane Bouchacourt. 2020. Permutation equivariant models for compositional generalization in language. In *International Conference on Learning Representations*.

Shane Griffith, Kaushik Subramanian, Jonathan Scholz, Charles L Isbell, and Andrea L Thomaz. 2013. Policy shaping: Integrating human feedback with reinforcement learning. In *Advances in neural information processing systems*, pages 2625–2633.

William L Hamilton, Rex Ying, and Jure Leskovec. 2017. Representation learning on graphs: Methods and applications. *arXiv preprint arXiv:1709.05584*.

Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. 2017. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969.

Han Hu, Jiayuan Gu, Zheng Zhang, Jifeng Dai, and Yichen Wei. 2018. Relation Networks for Object Detection. Technical report.

Drew A. Hudson and Christopher D. Manning. 2018. Compositional attention networks for machine reasoning. Technical report.

Yiding Jiang, Shixiang Shane Gu, Kevin P Murphy, and Chelsea Finn. 2019. Language as an abstraction for hierarchical deep reinforcement learning. In *Advances in Neural Information Processing Systems*, pages 9414–9426.

Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. 2017a. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2901–2910.

Justin Johnson, Bharath Hariharan, Laurens Van Der Maaten, Judy Hoffman, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. 2017b. Inferring and executing programs for visual reasoning. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2989–2998.

Russell Kaplan, Christopher Sauer, and Alexander Sosa. 2017. Beating atari with natural language guided reinforcement learning. *arXiv preprint arXiv:1704.05539*.

Thomas N Kipf and Max Welling. 2016. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*.

W Bradley Knox and Peter Stone. 2012. Reinforcement learning from simultaneous human and mdp reward. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems-Volume 1*, pages 475–482. International Foundation for Autonomous Agents and Multiagent Systems.

W Bradley Knox, Peter Stone, and Cynthia Breazeal. 2013. Training a robot via human feedback: A case study. In *International Conference on Social Robotics*, pages 460–470. Springer.

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

Gregory Kuhlmann, Peter Stone, Raymond Mooney, and Jude Shavlik. 2004. Guiding a reinforcement learner with natural language advice: Initial results in robocup soccer. In *The AAAI-2004 workshop on supervisory control of learning and adaptive systems*. San Jose, CA.

Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605.

David Mascharka, Philip Tran, Ryan Soklaski, and Arjun Majumdar. 2018. Transparency by design: Closing the gap between performance and interpretability in visual reasoning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4942–4950.

Tomas Mikolov, Edouard Grave, Piotr Bojanowski, Christian Puhrsch, and Armand Joulin. 2017. Advances in pre-training distributed word representations. *arXiv preprint arXiv:1712.09405*.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119.

Santiago Ontanon. 2018. Shrdlu: A game prototype inspired by winograd's natural language understanding work. In *Fourteenth Artificial Intelligence and Interactive Digital Entertainment Conference*.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8):9.

Adam Santoro, David Raposo, David G.T. Barrett, Mateusz Malinowski, Razvan Pascanu, Peter Battaglia, and Timothy Lillicrap. 2017. A simple neural network module for relational reasoning. *Advances in Neural Information Processing Systems*, 2017-Decem(Nips):4968–4977.

Michael Schlichtkrull, Thomas N Kipf, Peter Bloem, Rianne Van Den Berg, Ivan Titov, and Max Welling. 2018. Modeling relational data with graph convolutional networks. In *European Semantic Web Conference*, pages 593–607. Springer.

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.

Terry Winograd. 1970. Shrdlu.

Terry Winograd. 1980. What does it mean to understand language? *Cognitive science*, 4(3):209–241.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in neural information processing systems*, pages 5754–5764.

Shuiying Yao. 2010. Stage/individual-level predicates, topics and indefinite subjects. In *Proceedings of the 24th Pacific Asia Conference on Language, Information and Computation*, pages 573–582, Tohoku University, Sendai, Japan. Institute of Digital Enhancement of Cognitive Processing, Waseda University.

Kexin Yi, Chuang Gan, Yunzhu Li, Pushmeet Kohli, Jiajun Wu, Antonio Torralba, and Joshua B Tenenbaum. 2019. Clevrer: Collision events for video representation and reasoning. *arXiv preprint arXiv:1910.01442*.

Kexin Yi, Antonio Torralba, Jiajun Wu, Pushmeet Kohli, Chuang Gan, and Joshua B. Tenenbaum. 2018. Neural-symbolic VQA: Disentangling reasoning from vision and language understanding. *Advances in Neural Information Processing Systems*, 2018-Decem(NeurIPS):1031–1042.

Yanchao Yu, Arash Eshghi, and Oliver Lemon. 2016. Training an adaptive dialogue policy for interactive learning of visually grounded word meanings. In *Proceedings of the 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 339–349, Los Angeles. Association for Computational Linguistics.