Scene Graph Parsing via Abstract Meaning Representation in Pre-trained Language Models

Woo Suk Choi[1], **Yu-Jung Heo**[1,3], **Dharani Punitan**[1], **and Byoung-Tak Zhang**[1,2]
[1] Seoul National University [2] AI Institute (AIIS), Seoul National University [3] Surromind
{wschoi, yjheo}@bi.snu.ac.kr, punithan.dharani@gmail.com, btzhang@bi.snu.ac.kr

Abstract

In this work, we propose the application of abstract meaning representation (AMR) based semantic parsing models to parse textual descriptions of a visual scene into scene graphs, which is the first work to the best of our knowledge. Previous works examined scene graph parsing from textual descriptions using dependency parsing and left the AMR parsing approach as future work since sophisticated methods are required to apply AMR. Hence, we use pre-trained AMR parsing models to parse the region descriptions of visual scenes (i.e. images) into AMR graphs and pre-trained language models (PLM), BART and T5, to parse AMR graphs into scene graphs. The experimental results show that our approach explicitly captures high-level semantics from textual descriptions of visual scenes, such as objects, attributes of objects, and relationships between objects. Our textual scene graph parsing approach outperforms the previous state-of-the-art results by 9.3% in the SPICE metric score.

1 Introduction

Understanding and representing a scene is straightforward for humans, but an AI system requires various techniques to implement it. One such technique is scene graph proposed by (Johnson et al., 2015). Scene graph is a graph-structured representation that captures high-level semantics of visual scenes (i.e. images) by explicitly modeling objects along with their attributes and relationships with other objects. Scene graph is demonstrated effective in various tasks including semantic image retrieval (Wang et al., 2020; Schroeder and Tripathi, 2020), image captioning (Yang et al., 2019; Zhong et al., 2020), and visual question answering (Hildebrandt et al., 2020; Damodaran et al., 2021).

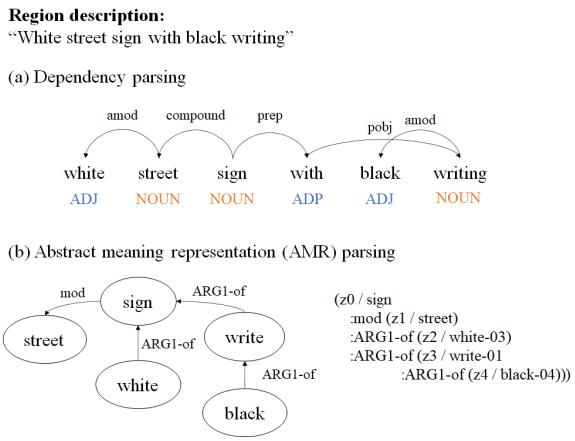

Figure 1: An example of (a) dependency parsing and (b) abstract meaning representation (AMR) parsing from textual description (i.e. region description) of "White street sign with black writing".

Approaches for scene graph generation are classified into two categories: 1) scene graph generation based on image as input and 2) scene graph generation based on text (i.e. image caption) as input. Various approaches (Xu et al., 2017; Zellers et al., 2018; Gu et al., 2019; Zhong et al., 2021) are proposed for the former category. On the other hand, only a fewer approaches (Schuster et al., 2015; Anderson et al., 2016; Wang et al., 2018; Andrews et al., 2019) are proposed for the latter. In this paper, we focus on the latter category, which is also called textual scene graph parsing. Textual scene graph parsing has the advantage of being able to capture the high-level meaning of the image scene from the text.

Most of previous works (Schuster et al., 2015; Anderson et al., 2016; Wang et al., 2018) for scene graph parsing generated scene graphs using dependency parsing to acquire the dependency relationships for all words in a text, as shown in Figure 1 (a). Apart from dependency parsing, there is also another approach for parsing semantic graphs from textual descriptions, which is called abstract mean-

ing representation (AMR) proposed by (Banarescu et al., 2013). AMR abstracts semantic concepts from words, and we therefore consider AMR is more suitable for scene graph parsing. However, the use of dependency parsing appeared to be a common theme in the literature rather than AMR, hence scene graph parsing with AMR has been left as future work in (Anderson et al., 2016; Wang et al., 2018).

To this end, we investigate the use of AMR with pre-trained language models (PLM), such as BART (Lewis et al., 2020) and T5 (Raffel et al., 2020), for parsing scene graphs from textual descriptions of visual scenes. We first parse sentences to AMR graphs using a pre-trained AMR parsing model, and then we generate scene graphs from AMR graphs using the PLM.

Our contributions are the following: i) To the best of our knowledge, ours is the first work for parsing scene graphs from texts using abstract meaning representation (AMR) contrary to the previous works (Schuster et al., 2015; Anderson et al., 2016; Wang et al., 2018). ii) We extend pre-trained language models such as BART and T5 to generate scene graphs from texts and AMR graphs. iii) Our approach outperforms the previous state-of-the-art result by 9.3% on SPICE metric for scene graph parsing task on intersection of Visual Genome and MS COCO datasets.

2 Related Works

2.1 Abstract Meaning Representation

Abstract meaning representation (AMR) (Banarescu et al., 2013) is a graph-based semantic representation which captures semantics "*who is doing what to whom*" in a sentence. Each sentence is represented as a rooted, directed, acyclic graph with labels on nodes (e.g. semantic concepts) and edges (e.g. semantic relations). Representative tasks for AMR are *Text-to-AMR*, capturing the meaning of a sentence within a semantic graph, and *AMR-to-Text*, generating text from such a graph. AMR2.0 (LDC2017T10) and AMR3.0 (LDC2020T02) datasets are currently actively used, which contain a semantic treebank of over $39,260$ and $59,255$ English natural language sentences, respectively from broadcast conversations, newswire, weblogs and web discussion forums.

To address these tasks, earlier studies used statistical methods. With the development of deep learning, researchers have proposed neural models such as graph-to-sequence (Zhu et al., 2019), sequence-to-graph (Cai and Lam, 2020), and neural transition-based parser models (Zhou et al., 2021). Recently, with the advent of pre-trained language models (PLM), AMR-based models incorporating the generation capability of PLM have been proposed and shown interesting results for various NLP tasks such as information extraction (Huang et al., 2018; Zhang and Ji, 2021), text summarization (Liu et al., 2015; Dohare and Karnick, 2017), and dialogue systems (Bonial et al., 2020).

(Lam et al., 2021) proposed an efficient heuristic algorithm to approximate the optimal solution by formalizing ensemble graph prediction as mining the largest graph that is the most supported by a collection of graph predictions. (Bevilacqua et al., 2021) proposed symmetric parsing and generation (SPRING), which casts AMR tasks as a symmetric transduction task by devising graph linearization and extending the pre-trained encoder-decoder model, BART. In this paper, we utilize pre-trained AMR parsing (i.e. Text-to-AMR) models from (Bevilacqua et al., 2021) to parse AMR graph from sentences since the SPRING model has the best performance among the publicly available pre-trained AMR parsing models[1].

2.2 Scene Graph Parsing

Scene graph proposed by (Johnson et al., 2015) is a graph-structured representation that represents rich structured semantics of visual scenes (i.e. images). Nodes in the scene graph represent either an object, an attribute for an object, or a relationship between objects. Edges depict the connection between two nodes. In this subsection, we introduce the study of scene graph parsing based on text. Most of the previous studies (Schuster et al., 2015; Anderson et al., 2016; Wang et al., 2018) used dependency parsing as a common theme. (Schuster et al., 2015) proposed a rule-based and a learned classifier with dependency parsing. (Wang et al., 2018) proposed a customized dependency parser with end-to-end training to parse scene graph. (Andrews et al., 2019) proposed a customized attention graph mechanism using the OpenAI Transformer[2] (Radford and Narasimhan, 2018). Unlike these studies, we use the AMR approach to parse scene graphs and demonstrate better quantitative

[1] https://github.com/SapienzaNLP/spring

[2] This model consists of a BPE (Byte-Pair-Encoding) subword embedding layer followed by 12-layers of decoder-only transformer with masked self-attention heads.

performance.

2.3 Pre-trained Language Model

BART (Lewis et al., 2020) is a denoising autoencoder for pretraining sequence-to-sequence (seq2seq) models. It uses the standard Transformer (Vaswani et al., 2017)-based neural machine translation (NMT) architecture. It is constructed based on seq2seq/NMT architecture by combining a bidirectional encoder (Devlin et al., 2019) and a left-to-right decoder (Radford et al., 2019). BART is trained by corrupting text with an arbitrary noising function (i.e. token masking, infilling, deletion, and sentence permutation) and learning a model to reconstruct the original text. We use both BART-base (BART model with 6 encoder and decoder layers and around 140M parameters) and BART-large (BART model with 12 encoder and decoder layers and nearly 400M parameters) models for our investigation.

T5 (Raffel et al., 2020) is an encoder-decoder unified framework that is pre-trained on a multi-task mixture of unsupervised and supervised tasks and for which a wide range of NLP tasks such as translation, classification, and question answering are cast as feeding the model text as input and training it to generate some target text. We use both T5-base (T5 model with 12 encoder and decoder layers and nearly 220M parameters) and T5-large (T5 model with 24 encoder and decoder layers and nearly 770M parameters) models for our examination.

3 Methodology

In this section, we use pre-trained language models (PLM), BART (Lewis et al., 2020) and T5 (Raffel et al., 2020), as baselines to parse scene graph (SG) from text directly (Text-to-SG). We then describe how to generate scene graphs from AMR graphs (AMR-to-SG) using PLM models.

3.1 Text-to-SG Parsing

We train the pre-trained language models to take each region description of an image as input and generate scene graphs. The PLM models take text as input and map it into a task-specific output sequence. For instance, if the region description *"White street sign with black writing"* is an input, the parsed output, {*(street sign, writing), (white-street sign, black-writing), (street sign-with-writing)*} will be in the form of *{(objects), (attribute-object), (object-relationship-object)}*.

3.2 AMR-to-SG Parsing

First, we parse the region descriptions into AMR graphs. Then, we parse the AMR graphs into the scene graphs. For this, we use the two AMR parsing models of SPRING (Bevilacqua et al., 2021), which are pre-trained on AMR2.0 (LDC2017T10) and AMR3.0 (LDC2020T02) datasets.

We linearize the AMR graph into a sequence of symbols which will be the input to pre-trained language models, BART and T5, for training. For the linearization technique, we adopt the depth-first search (DFS) based algorithm used in (Konstas et al., 2017), as it is closely related to the way how natural language syntactic trees are linearized (Bevilacqua et al., 2021). Thus, as shown in Figure 1 (b), the input of BART and T5 will be *"(z0 / sign :mod (z1 / street) :ARG1-of (z2 / white-03) :ARG1-of (z3 / write-01 :ARG1-of (z4 / black-04)))"*, where *zo, z1, z2, z3* and *z4* are special tokens to handle co-referring nodes, and the output will be in the same format as Text-to-SG parsing output.

4 Experiments

4.1 Implementation Details

Datasets For fair comparisons with the existing models, we train and validate our models with the subsets of Visual Genome (VG) (Krishna et al., 2016) and MS COCO (Lin et al., 2014) datasets. The training set is the intersection of the VG and MS COCO train2014 set (34,027 images with 1,070,145 regions). The evaluation set is the intersection of VG and MS COCO val2014 set (17,471 images with 547,795 regions). We follow the same preprocessing steps as in (Wang et al., 2018) for setting the training/test splits.

Evaluation To evaluate parsed scene graphs from region descriptions with the ground truth region scene graphs, we use SPICE metric (Anderson et al., 2016) which calculates a F-score over tuples. As mentioned in (Wang et al., 2018), there is an issue that a node in one graph could be matched to several nodes in the other when SPICE calculates the F-score. Thus, following previous works, we enforce one-to-one matching while calculating the F-score and report the average F-score for all regions.

Scene graph parser		F-score
Text-to-SG		
Stanford (Schuster et al., 2015)		0.3549
SPICE (Anderson et al., 2016)		0.4469
CDP (Wang et al., 2018)		0.4967
AG (Andrews et al., 2019)		0.5221
BART-base		0.5071
BART-large		0.5073
T5-base		0.5093
T5-large		0.5101
AMR-to-SG		
BART-base	AMR2.0	0.6112
	AMR3.0	0.6096
BART-large	AMR2.0	0.6062
	AMR3.0	0.6092
T5-base	AMR2.0	0.6128
	AMR3.0	0.6114
T5-large	AMR2.0	**0.6151**
	AMR3.0	0.6149

Table 1: F-score (i.e. SPICE metric) comparison between pre-trained language models (for Text-to-SG and AMR-to-SG) and existing parsers on the intersection of VG (Krishna et al., 2016) and MS COCO (Lin et al., 2014) validation set. CDP and AG are abbreviations of Customized Dependency Parser and Attention Graph, respectively.

Experimental Settings In our experiments, We set the number of epoch to 5, the batch size to 32, and learning rate to 0.0005 with a weight decay of 0.004. It takes about a day to train BART-base, BART-large, and T5-base models and around four days to train T5-large model using two Tesla V100 with 32 GB graphic memory.

4.2 Results and Analysis

Table 1 shows results of the F-score comparison between pre-trained language models (PLM) with both Text-to-SG and AMR-to-SG and existing parsers on the intersection of VG and MS COCO validation set.

Text-to-SG We observe that the performance of PLM models is relatively higher than dependency parsing based models (i.e. Stanford, SPICE and Customized Dependency Parser) and shows comparable results with the previous state-of-the-art model, Attention Graph (AG), which used customized attention graph with pre-trained transformer model. Furthermore, we find that the larger the model size, the better the performance. We expect to improve the performance of PLM models with hyperparameter tuning, which we perform as our future work.

AMR-to-SG All parsing models using AMR (AMR-to-SG) not only outperform the previous state-of-the-art model, Attention Graph (AG), but also show better performance than Text-to-SG PLM-based models. All of AMR-to-SG models for AMR 2.0 achieves an average of 8.92% performance improvement, and 8.91% for AMR 3.0. In particular, our best model (T5-large for AMR 2.0) outperforms the previous state-of-the-art model by 9.3%. Interestingly, despite the same PLM model, when comparing the case where AMR graph is input instead of text, BART shows an average of 10.32% performance improvement for AMR 2.0 and 10.22% for AMR 3.0, respectively. T5 shows an average of 10.43% performance improvement for AMR 2.0 and 12.87% for AMR 3.0, respectively. In consequence, we find that the AMR based approach captures high-level abstract semantics of text better than dependency parsers and the other baseline models.

5 Conclusion

In this work, we investigate the application of abstract meaning representation (AMR) for parsing scene graph by using pre-trained language models (PLM), BART and T5, with AMR parsing model of SPRING. We conducted two sets of experiments: 1) scene graph parsing using PLM models, directly from region descriptions, and 2) scene graph parsing using PLM models from AMR graphs parsed from region descriptions via AMR parsing pre-trained models. Our results show AMR graphs capture high-level abstract semantics of region descriptions. We evaluate our approach using the SPICE metric score. The results of Text-to AMR are comparable and of AMR-to-Text outperform the existing state-of-the-art models by 9.3%.

In our future work, we will investigate an adapter-based method (Ribeiro et al., 2021) to encode graph structures into PLM models to improve the performance of textual scene graph parsing. Furthermore, we will examine our approach based on the lately published, pre-trained AMR parsing model, AMRBART[3] (Bai et al., 2022). As our scene graph parser performance improves further, we expect to be able to use it to automatically generate either an image scene graph or video scene graph datasets with less biased and more diverse labels.

[3]https://github.com/muyeby/AMRBART

Acknowledgements

This work was partly supported by the Institute of Information Communications Technology Planning Evaluation (2015-0-00310-SW.StarLab/20%, 2019-0-01371-BabyMind/20%, 2021-0-02068-AIHub/10%, 2021-0-01343-GSAI/10%, 2022-0-00951-LBA/20%, 2022-0-00166-PICA/20%) grant funded by the Korean government.

References

Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. 2016. Spice: Semantic propositional image caption evaluation. In *Computer Vision – ECCV 2016*, pages 382–398, Cham. Springer International Publishing.

Martin Andrews, Yew Ken Chia, and Sam Witteveen. 2019. Scene graph parsing by attention graph. *CoRR*, abs/1909.06273.

Xuefeng Bai, Yulong Chen, and Yue Zhang. 2022. Graph pre-training for amr parsing and generation. *ArXiv*, abs/2203.07836.

Laura Banarescu, Claire Bonial, Shu Cai, Madalina Georgescu, Kira Griffitt, Ulf Hermjakob, Kevin Knight, Philipp Koehn, Martha Palmer, and Nathan Schneider. 2013. Abstract Meaning Representation for sembanking. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 178–186, Sofia, Bulgaria. Association for Computational Linguistics.

Michele Bevilacqua, Rexhina Blloshmi, and Roberto Navigli. 2021. One SPRING to rule them both: Symmetric AMR semantic parsing and generation without a complex pipeline. In *Proceedings of AAAI*.

Claire Bonial, Lucia Donatelli, Mitchell Abrams, Stephanie M. Lukin, Stephen Tratz, Matthew Marge, Ron Artstein, David Traum, and Clare Voss. 2020. Dialogue-AMR: Abstract Meaning Representation for dialogue. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 684–695, Marseille, France. European Language Resources Association.

Deng Cai and Wai Lam. 2020. AMR parsing via graph-sequence iterative inference. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1290–1301, Online. Association for Computational Linguistics.

Vinay Damodaran, Sharanya Chakravarthy, Akshay Kumar, Anjana Umapathy, Teruko Mitamura, Yuta Nakashima, Noa García, and Chenhui Chu. 2021. Understanding the role of scene graphs in visual question answering. *ArXiv*, abs/2101.05479.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*.

Shibhansh Dohare and Harish Karnick. 2017. Text summarization using abstract meaning representation. *ArXiv*, abs/1706.01678.

Jiuxiang Gu, Handong Zhao, Zhe Lin, Sheng Li, Jianfei Cai, and Mingyang Ling. 2019. Scene graph generation with external knowledge and image reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Marcel Hildebrandt, Hang Li, Rajat Koner, Volker Tresp, and Stephan Günnemann. 2020. Scene graph reasoning for visual question answering. *ArXiv*, abs/2007.01072.

Lifu Huang, Heng Ji, Kyunghyun Cho, Ido Dagan, Sebastian Riedel, and Clare Voss. 2018. Zero-shot transfer learning for event extraction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2160–2170, Melbourne, Australia. Association for Computational Linguistics.

Justin Johnson, Ranjay Krishna, Michael Stark, Li-Jia Li, David Shamma, Michael Bernstein, and Li Fei-Fei. 2015. Image retrieval using scene graphs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Ioannis Konstas, Srinivasan Iyer, Mark Yatskar, Yejin Choi, and Luke Zettlemoyer. 2017. Neural AMR: Sequence-to-sequence models for parsing and generation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 146–157, Vancouver, Canada. Association for Computational Linguistics.

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. 2016. Visual genome: Connecting language and vision using crowdsourced dense image annotations.

Hoang Thanh Lam, Gabriele Picco, Yufang Hou, Young-Suk Lee, Lam M. Nguyen, Dzung T. Phan, Vanessa López, and Ramon Fernandez Astudillo. 2021. Ensembling graph predictions for amr parsing. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *ECCV*.

Fei Liu, Jeffrey Flanigan, Sam Thomson, Norman M. Sadeh, and Noah A. Smith. 2015. Toward abstractive summarization using semantic representations. In *NAACL*.

Alec Radford and Karthik Narasimhan. 2018. Improving language understanding by generative pre-training.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Leonardo F. R. Ribeiro, Yue Zhang, and Iryna Gurevych. 2021. Structural adapters in pretrained language models for AMR-to-Text generation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4269–4282, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Brigit Schroeder and Subarna Tripathi. 2020. Structured query-based image retrieval using scene graphs. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 680–684.

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, Lisbon, Portugal. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Sijin Wang, Ruiping Wang, Ziwei Yao, Shiguang Shan, and Xilin Chen. 2020. Cross-modal scene graph matching for relationship-aware image-text retrieval. In *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1497–1506.

Yu-Siang Wang, Chenxi Liu, Xiaohui Zeng, and Alan Yuille. 2018. Scene graph parsing as dependency parsing. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 397–407, New Orleans, Louisiana. Association for Computational Linguistics.

Danfei Xu, Yuke Zhu, Christopher Choy, and Li Fei-Fei. 2017. Scene graph generation by iterative message passing. In *Computer Vision and Pattern Recognition (CVPR)*.

Xu Yang, Kaihua Tang, Hanwang Zhang, and Jianfei Cai. 2019. Auto-encoding scene graphs for image captioning. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10677–10686.

Rowan Zellers, Mark Yatskar, Sam Thomson, and Yejin Choi. 2018. Neural motifs: Scene graph parsing with global context. In *Conference on Computer Vision and Pattern Recognition*.

Zixuan Zhang and Heng Ji. 2021. Abstract Meaning Representation guided graph encoding and decoding for joint information extraction. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 39–49, Online. Association for Computational Linguistics.

Yiwu Zhong, Jing Shi, Jianwei Yang, Chenliang Xu, and Yin Li. 2021. Learning to generate scene graph from natural language supervision. In *ICCV*.

Yiwu Zhong, Liwei Wang, Jianshu Chen, Dong Yu, and Yin Li. 2020. Comprehensive image captioning via scene graph decomposition. In *ECCV*.

Jiawei Zhou, Tahira Naseem, Ramón Fernandez Astudillo, and Radu Florian. 2021. AMR parsing with action-pointer transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5585–5598, Online. Association for Computational Linguistics.

Jie Zhu, Junhui Li, Muhua Zhu, Longhua Qian, Min Zhang, and Guodong Zhou. 2019. Modeling graph structure in transformer for better AMR-to-text generation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5459–5468, Hong Kong, China. Association for Computational Linguistics.