Generating Context-Aware Natural Answers for Questions in 3D Scenes

Mohammed Munzer Dwedari
munzer.dwedari@tum.de
Matthias Niessner
niessner@tum.de
Zhenyu Chen
zhenyu.chen@tum.de

Visual Computing Lab
Technical University of Munich
Munich, Germany

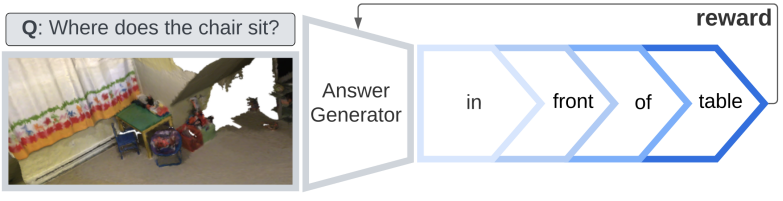

Figure 1: We propose Gen3DQA, an end-to-end transformer-based architecture for generating natural answers for questions in 3D scenes. Our method directly optimizes the global semantics of the generated sentences via the language rewards.

Abstract

3D question answering is a young field in 3D vision-language that is yet to be explored. Previous methods are limited to a pre-defined answer space and cannot generate answers naturally. In this work, we pivot the question answering task to a sequence generation task to generate free-form natural answers for questions in 3D scenes (Gen3DQA). To this end, we optimize our model directly on the language rewards to secure the global sentence semantics. Here, we also adapt a pragmatic language understanding reward to further improve the sentence quality. Our method sets a new SOTA on the ScanQA benchmark (CIDEr score **72.22/66.57** on the test sets). The project code can be found at: https://github.com/MunzerDw/Gen3DQA.git.

1 Introduction

Visual question answering is a fundamental task in vision-language understanding [2, 3, 40, 41, 53]. Unlike dense captioning [11, 21, 23, 53] or visual grounding [7, 9, 15, 18, 19, 22, 57, 58], question answering requires the intelligent system to understand the joint context of the question (language) and scene (vision) to interact with the environment by providing answers. While visual question answering on images has been extensively researched, question answering on 3D scenes is yet to be explored.

The seminal work such as ScanQA [4] relies on a two-branch architecture for encoding both modalities of the input (point cloud and question) before fusing them into a joint vector representation. Then, an answer is predicted among a predefined answer space using such multimodal feature. Along with the answer classification, a bounding box is predicted to localize the object referred to in the question. Another competitive method, CLIP-guided [32], transfers 2D prior knowledge to the 3D domain via a contrastive learning scheme using CLIP features [37]. However, aforementioned baseline methods are limited to a predefined answer space, which consequently hinders the capability of interacting with human users.

To tackle this challenge, we propose a transformer-based architecture to generate, rather than predict, free-from answers for questions in 3D environments. Using reinforcement learning, we directly train our model with a language reward to secure the global semantics of the generated sentences, as shown in Figure 1. To this end, we utilize the policy gradient method [38] to approximate sampled gradients through our end-to-end architecture. To further ensure the correctness of the generated answers, we introduce an additional helper reward that encourages the model to reversely reconstruct the questions from the respective answers. Our method outperforms the state-of-the-art methods for the image captioning metrics on the ScanQA [4] benchmark. We summarize our contributions as follows:

- We propose an end-to-end transformer-based architecture for the task of 3D visual question answering, which deals with ambiguous contexts and generates free-form natural answers.
- We present a reinforcement-learning-based training objective that directly optimizes the global semantics of the generated sentences. We also incorporate a pragmatic helper reward that encourages correct reconstruction of the question from the generated answer, which further improves answer quality.
- We conduct extensive experiments and ablation studies to show the effectiveness of our method. Our method sets a new SOTA performance on the ScanQA benchmark [4] (CIDEr score **72.22/66.57** on the test sets).

2 Related Work

3D Vision-Language. One of the first works to combine 3D scenes and natural language is the ScanRefer [7] benchmark, which introduces the task of visual grounding in ScanNet [14] scenes. The ScanRefer [7] dataset has more object categories than originally set in ScanNet [14]. The authors design a two branch model where one branch encodes the scene with a PointNet++ [36] backbone and the other the reference sentence with a GRU [12]. A fusion module takes in both encoded modalities to predict the final target object. Shortly after, the reverse task of dense captioning in 3D scenes is introduced [11]. The same backbone is used in addition to a graph module and attention mechanism to predict the bounding boxes and generate their descriptions. D3Net [8] combines both tasks into a speaker-listener architecture, where the speaker takes in predicted object proposals from a detector backbone and generates a caption sentence for each one. The generated captions are passed to the listener, which grounds the target objects. The method employs the REINFORCE [50] algorithm for sequence generation [38] to train both modules jointly.

Visual Question Answering. Question answering [41] on images has been extensively researched, where most models approach the problem as a classification task [2, 20, 45, 54, 56].

In addition, several works focus on VQA with video as input [25, 39, 51, 60]. Today, large transformer based pretrained models have shown the ability to achieve superior performance on the VQA task [6, 10, 16, 29, 41, 42, 44, 49]. However, in the 3D domain, it is still unexplored. One of the first works is introduced by Azuma *et al*. [4]. Based on ScanNet [14] scenes and ScanRefer [7] object descriptions, the authors create the ScanQA dataset with the additional task of grounding the target object(s) with a bounding box in the scenes. Their baseline architecture consists of two encoder branches, one of which utilizes a PointNet++ [36] backbone to encode the scene into object proposals. A transformer based fusion module [59] produces a final vector from which the answer is predicted. Furthermore, Prelli *et al*. achieve the current state-of-the-art on the ScanQA [4] benchmark with their method where they apply knowledge from the 2D domain into the 3D domain. They pretrain a CLIP [37] encoder to align the scene features with the image and question embeddings. In the next training step, the CLIP module is used to encode the question sequence. Similarly, the answer is predicted from the final <end> token representation of the question. In contrast to previous works, we solely train on the 3D scenes with SoftGroup [48] as a backbone. Apart from a transformer encoder, we also implement a transformer decoder to generate rather than predict the answer.

Reinforcement Learning for Sequence Generation. Rennie *et al*. [38] introduce reinforcement learning to the task of image captioning where they utilize the REINFORCE algorithm [50] and optimize their model directly on the non differentiable CIDEr [47] metric. Their LSTM [17] model is seen as the "agent" that interacts with the "environment", which is the words and image features. The network acts as the "policy" that determines the "action" taken by the agent, which in this case is the prediction of the next word. Following an action, the model updates its "state", i.e. weights. Once a sentence is generated, the model receives a "reward" in form of the CIDEr score of the generated sentence. Vedantam *et al*. [13] apply the same training method on their transformer based architecture with a small variation, where they generate k sentences with the beam search algorithm and baseline each sentence on the average reward of all sentences. Luo *et al*. [30] introduce a better variant of the original self-critical sequence training (SCST [38]) where they sample k sentences (using random sampling) and calculate for each sentence the average reward of the rest as a baseline.

3 Method

In this section, we explain our model architecture (Figure 2) and training method. The input of our model is a 3D point cloud with RGB and normals features. The second input is the question sequence, and the output is a token sequence of the generated answer. Overall, our model can be divided into 3 segments: SoftGroup, transformer encoder-decoder, and object localization. Our training method consists of 3 stages, which we will explain in detail in Section 3.2.

3.1 Model

SoftGroup. Instead of relying on a pointwise method for the backbone network [35] like in previous works [4, 7, 11, 32], we employ the 3D-sparse-convolution based method SoftGroup [48] to extract denser semantic information of the object proposals. This method has shown better performance and speed on the object detection task [48]. In addition, the instance masks of the generated object proposals offer a unique identity for objects and thus

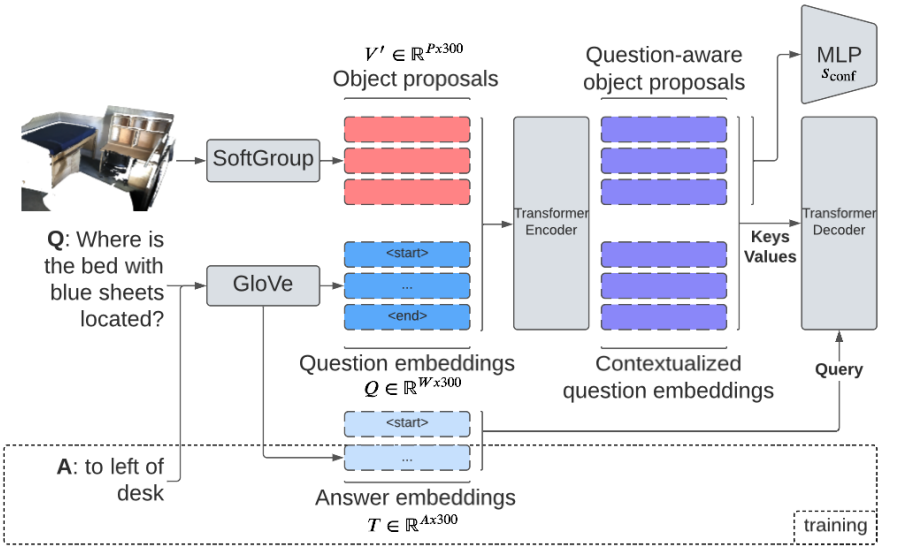

Figure 2: Overview of our model architecture. The input scene is encoded into object proposals V' with SoftGroup [48]. The question and answer tokens are turned into word embeddings (Q and T) with GloVe [34]. The question sequence and object proposals are concatenated as a single sequence and fed into the transformer encoder. The contextualized sequence is then forwarded to a transformer decoder as keys and values, while the embedded answer sequence is passed as query during training with XE loss. During inference, only the <start> token embedding is used as a query in the decoder to begin the sentence. Best viewed in color.

provide better scene understandings for the answers. With that, we generate P semantically rich object proposals $V \in \mathbb{R}^{P\times 32}$ from the point cloud scene. To match the dimension of our question token embeddings, we train a linear layer that expands the object proposals to $V' \in \mathbb{R}^{P\times 300}$.

Transformer Encoder-Decoder. We encode our W question tokens with GloVe [34] embeddings as $Q \in \mathbb{R}^{W\times 300}$. Then, we add positional encodings to both of our modalities representations. For the question embeddings, we follow the original transformer positional encoding [46]. As for the object proposals, we add the normalized center points $Z \in \mathbb{R}^{P\times 3}$ to the last 3 dimensions of each (expanded) object proposal. Transformers have shown great performance when it comes to sequence-to-sequence generation. There are several approaches to encoding two sequences of different modalities with transformers. Following the approaches mentioned in the survey by Xu *et al.* [52], we choose early concatenation, which enables the model to equally encode the scene information into the question and the question information into the scene. This method has shown to well preserve the global multi-modal context [27, 43, 52], which is necessary for both answer generation and object localization tasks. Hence, we concatenate both sequences into one sequence $S \in \mathbb{R}^{L\times 300}$ where $L = P + W$ and feed it into a two-layer transformer encoder. The sequence S acts as the keys, values and query and is encoded into one sequence $S' \in \mathbb{R}^{L\times 300}$ containing the contextualized object proposals and question embeddings. The contextualized sequence is fed into a two-layer transformer decoder as keys and values. The target sequence containing

GloVe word embeddings of the answer $T \in \mathbb{R}^{A\times 300}$ is used as the query.

Object Localization. After encoding the full multi-modal sequence, we feed the question-aware object proposals into an MLP to predict their confidence scores $s_{\text{conf}} \in \mathbb{R}^{P\times 1}$. The object proposal with the highest confidence score is considered as our target object.

3.2 Training

First, we pretrain SoftGroup [48] with the ScanRefer [7] object classes. We experiment with different input features and find that RGB + normals features result in the best overall scores. Our object detection scores from SoftGroup [48] can be found in the appendix. Since the forward pass of SoftGroup [48] is significantly more time-consuming than the forward pass of our language model, we precompute the object proposals from the pretrained SoftGroup [48] model and save them on disk before using them for our visual-language model. During prediction on the test sets, we re-activate SoftGroup [48] and do the full forward pass. SoftGroup [48] is trained end-to-end on a multitask loss $L_{\text{softgroup}}$, which encompasses the total loss for the first training stage.

Next, we train our question answering model on word level cross entropy (XE) loss: $L_{\text{ans}} = -\sum_{t\in T}\sum_{z\in Z} y_{t,z} log(\hat{y}_{t,z})$ where T is the ground truth answer including the <end> token and Z is the training vocabulary. $y_{t,z}$ has the value 1 when the current ground truth token t matches the vocabulary token z and 0 otherwise. $\hat{y}_{t,z}$ is the predicted probability of the token z in the Softmax output for the word at step t. The model is trained with the teacher forcing scheme, where we pass the ground truth previous words as the query to predict the next word at each time step. Simultaneously, we train our object localization branch on cross entropy loss L_{loc}, similar to [4, 7]. Thus, our total loss for the second training stage is $L = L_{\text{ans}} + L_{\text{loc}}$.

After our question-answering model converges on the CIDEr score accuracy, we drop the word level XE loss L_{ans} and switch to reinforcement learning, while keeping the object localization loss L_{loc}. Here, we apply the self-critical sequence training [38] method and train directly on the CIDEr score. We treat our transformer model as the "agent", the question & answer words and object proposals as the "environment", our network parameters as the "policy" p_θ, the prediction of the next word as the "action", and the CIDEr score of the generated answer as the "reward". Instead of sampling the answer sequence like in [38], we generate it using test-time greedy decoding to get w^g where $w^g = (w_1^g, ..., w_T^g)$ and w_t^g is the word with the maximum likelihood at time step t. Our loss can be expressed as the negative expected reward:

$$L_{\text{cider}}(\theta) = -\mathbb{E}_{w^g\sim p_\theta}[r(w^g)] \tag{1}$$

where $r(.)$ is the reward function (CIDEr score). As for the baseline, we generate k answers using beam search decoding. We keep track of the top-k answers and predict the next word until we reach the <end> token for all top k sequences. We take the average reward of the k answers as the baseline reward r^b_{VQA}. In addition to the reward from the generated answer, we also train a Visual Question Generation (VQG) module by simply switching the input (question) and output (answer) of our transformer model. Since we treat the VQA task as a sequence generation problem, our model can be easily switched to the inverse task. Once we generate an answer from the VQA module, we feed it into the frozen VQG module to greedily generate a question q^g. The same thing is also done with the generated baseline answers, which results in k baseline questions. Similar to VQA, we get the CIDEr scores for the generated question r^g_{VQG} and for the baseline questions r^b_{VQG}. With that, we can express

Model	BLEU-1	BLEU-4	ROUGE	METEOR	CIDEr
Test w/ object IDs					
ScanQA [4]	31.56	12.04	34.34	13.55	67.29
CLIP-guided [32]	32.72	**14.64**	35.15	13.94	69.53
Gen3DQA (XE loss)	35.24	10.79	33.50	13.61	64.83
Gen3DQA	**39.30**	12.24	**35.78**	**14.99**	**72.22**
Test w/o object IDs					
ScanQA [4]	30.68	10.75	31.09	12.59	60.24
CLIP-guided [32]	32.70	**11.73**	32.41	13.28	62.83
Gen3DQA (XE loss)	35.08	10.62	30.99	12.87	60.05
Gen3DQA	**38.07**	11.61	**33.03**	**14.28**	**66.57**

Table 1: Image captioning metrics scores of previous methods and ours on the ScanQA [4] test benchmark with and without object IDs. At the time of evaluation on the benchmark website, the SPICE [1] score is not available.

the gradient of our loss as:

$$\nabla_\theta L_{\text{cider}}(\theta) = -((r^g_{\text{VQA}} - r^b_{\text{VQA}}) + (r^g_{\text{VQG}} - r^b_{\text{VQG}}))\nabla_\theta log p_\theta(w^g) \tag{2}$$

where r^g_{VQA} is the reward for the generated answer w^g. During our experiments, we find that using greedy decoding for the VQA baseline, as in [33], does not yield any improvement, since the sampled sentence in our case usually has a worse CIDEr score than the answer generated with greedy decoding. Thus, we experiment with greedy decoding for generating the answer and sampling for the baseline. However, we also find that sentences generated with beam search have worse results than the ones generated greedily. Therefore, we conduct experiments with using beam search as our baseline and see a noticeable improvement. When using a beam size of 2, the reward difference between the generated answer and the average of the baseline answers becomes too small and crashes the accuracies after few epochs. Increasing the beam size to 3 widens the difference in rewards and stabilizes our training. The final total loss for the third training stage is $L = L_{\text{cider}} + L_{\text{loc}}$.

3.3 Inference

During inference, we re-activate SoftGroup [48] to generate object proposals. As for the transformer decoder, we apply greedy decoding to generate the answer sequence beginning with the token <start>. Once we reach the <end> token, our decoder stops. We determine the confidence score for each object proposal with the object localization branch and pick the one with the highest value as our target object. The object class of the target object is determined by the classification branch of SoftGroup [48].

4 Experiments

4.1 Data

We train and test our model on the ScanQA [4] dataset. The 3D scenes are from the ScanNet [14] dataset, while the questions are based on the ScanRefer [7] object descriptions.

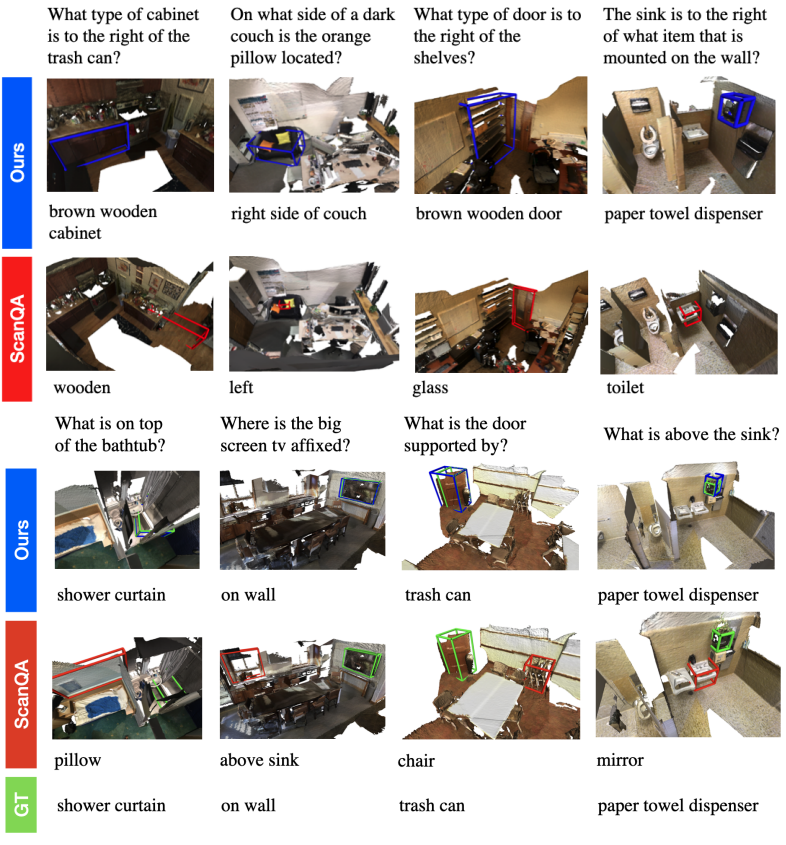

Figure 3: Example questions and answers from the test set with object IDs (top) and the validation set (bottom). We compare the results of our model (blue) to ScanQA [4] (red) and the ground truth (GT) (green). Below every image is the predicted or generated answer. Since we do not axis-align our scenes, the bounding boxes in our model look tilted. We include more examples in the appendix. Best viewed in color.

Hence, the categories of the objects in question are from the ScanRefer [7] classes. Moreover, we treat questions with multiple answers as multiple training samples, where every sample contains the same question and one of the answers. This introduces 952 additional training samples. Furthermore, we evaluate our model on both test sets of ScanQA [4] on the image captioning metrics BLUE-1 [31], BLEU-4 [31], ROUGE [26], METEOR [5] and CIDEr [47] and exclude the EM@1 and EM@10 accuracies since we do not have answer classification in our model. Our scores are calculated by uploading our question-answering results to the ScanQA [4] benchmark server [1], where at the time of writing the SPICE [1] score is returning with the value of 0.0 and is thus not included.

[1] https://eval.ai/web/challenges/challenge-page/1715/overview

	ScanQA [4]	CLIP-guided [32]	Gen3DQA
Acc@0.5	15.42	21.22	**23.79**

Table 2: Object localization accuracy Acc@0.5 of previous methods and ours on the ScanQA [4] validation set.

4.2 Implementation Details

We follow the implementation of MINSU3D [2] for training SoftGroup [48] and change the class mappings in the data preparation phase to fit the ScanRefer [7] object classes. We use Adam [24] optimizer for training our language model with a learning rate of 8e-5 when training on XE loss and 2e-5 when training with reinforcement learning. In both cases, we apply cosine annealing [28] for scheduling. We train with a batch size of 64 on a GeForce RTX 2080 Ti. Our models are implemented in PyTorch [33]. For data augmentation, we randomly replace one random word in the question with the <unk> token. When training on XE loss, our model converges on the CIDEr score after 80,000 iterations. As for training with the REINFORCE algorithm, our best model converges after 100,000 iterations.

4.3 Quantitative Analysis

We show in Table 1 our final results in comparison to ScanQA [4] and CLIP-guided [32]. Our method outperforms previous works on the conditional image captioning metrics and especially on the more challenging CIDEr score. Unlike the CLIP-based method [32], our model only requires 3D point cloud data to train. By training directly on the CIDEr score, our model performance is significantly improved on the rest of the metrics too.

Furthermore, we also look at the object localization task on the validation set in comparison to previous methods (Table 2). Even though the current state-of-the-art trains with additional image data, our model achieves a noticeable improvement on the Acc@0.5 accuracy. We also see that our early concatenation method is superior to the fusion module of ScanQA [4] in multi-modal context understanding. With that, our method presents a stronger understanding of the scene and question and can generate context-aware answers naturally while localizing relevant objects significantly better than previous methods.

4.4 Qualitative Analysis

In Figure 3 we showcase samples from the test set (with object IDs) where our model generates better answers than ScanQA [4] predicts, while localizing a meaningful target object. Overall, we see that our model generates longer answers that contain more information. In fact, compared to ScanQA [4], the average number of words in an answer from our model is 1.87/1.92 (test set with and without object IDs) compared to ScanQA [4] with 1.41/1.47. Furthermore, we show in Figure 3 samples from the validation set where our model performs better object localization than ScanQA [4] while also generating the correct answer. We see in the samples that our model performs well when the question requires spatial awareness and can also extract details about object types and looks.

[2] https://github.com/3dlg-hcvc/minsu3d

Model	BLEU-1	BLEU-4	ROUGE	METEOR	CIDEr
Gen3DQA (single object)	35.4	**10.52**	**33.39**	13.62	**64.91**
Gen3DQA (multiple objects)	**36.02**	10.21	32.84	**13.68**	64.51
Gen3DQA (w/o object localization)	34.29	10.02	31.2	12.99	59.35

Table 3: Scores of our model trained on XE loss once with targeting a single object, once multiple objects, and once without object localization at all. Evaluation is done on the validation set.

Model	BLEU-1	BLEU-4	ROUGE	METEOR	CIDEr
Valid					
Gen3DQA (w/o VQG reward)	39.12	**13.2**	35.48	14.89	71.39
Gen3DQA (w/ VQG reward)	**39.53**	12.7	**35.97**	**15.11**	**71.97**
Test w/ object IDs					
Gen3DQA (w/o VQG reward)	38.89	**12.67**	35.35	14.82	71.09
Gen3DQA (w/ VQG reward)	**39.30**	12.24	**35.78**	**14.99**	**72.22**
Test w/o object IDs					
Gen3DQA (w/o VQG reward)	37.61	**12.00**	32.57	14.09	65.58
Gen3DQA (w/ VQG reward)	**38.07**	11.61	**33.03**	**14.28**	**66.57**

Table 4: Scores of our model trained with reinforcement learning with and without the additional reward of Visual Question Generation (VQG).

4.5 Ablation Studies

Does multi-object localization help? In ScanQA [4] the authors experiment with training the object localization branch on binary cross entropy (BCE) loss. This enables the model to decide for each object whether it should be targeted or not, allowing multiple objects to be selected. Overall, we don't see a clear performance improvement in our model from targeting multiple objects (Table 3). We also conduct an experiment where we train our model without the object localization loss and see that by training our model to localize the target object, it becomes better at generating answers.

Does VQG reward improve answer generation? We hypothesize in the beginning that by generating a better answer, it becomes easier to regenerate the original question from it. Hence, we train a question-generation module (VQG) and use it to generate a question from our generated answer during reinforcement learning. We then add the CIDEr score of the generated question as an additional reward. The results in Table 4 show that training with the additional question generation reward yields better answer generation scores.

5 Conclusion and Future Work

In this work, we propose a new architecture for the task of 3D visual question answering to generate free-form answers. We directly train our model on the CIDEr metric using a version of the REINFORCE algorithm [38, 50]. In addition, we introduce the inverse task of question generation to enhance our question-answering model during reinforcement learning. Our

experiments and results show that our method outperforms the current state-of-the-art on the image captioning metrics of the ScanQA [4] benchmark. For future work, we encourage the research community to further explore the dual tasks of question answering and generation. For instance, we suggest jointly training both tasks without freezing any weights. We also look forward to future works to explore and develop better answer generation models instead of answer classification ones for questions in 3D environments.

6 Acknowledgement

This work was supported by the ERC Starting Grant Scan2CAD (804724) and the German Research Foundation (DFG) Research Unit "Learning and Simulation in Visual Computing".

References

[1] Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. Spice: Semantic propositional image caption evaluation. In *ECCV*, 2016.

[2] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pages 2425–2433, 2015.

[3] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In *Proceedings of the IEEE international conference on computer vision*, pages 2425–2433, 2015.

[4] Daichi Azuma, Taiki Miyanishi, Shuhei Kurita, and Motoaki Kawanabe. Scanqa: 3d question answering for spatial scene understanding. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19129–19139, 2022.

[5] Satanjeev Banerjee and Alon Lavie. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan, June 2005. Association for Computational Linguistics. URL https://aclanthology.org/W05-0909.

[6] Hangbo Bao, Wenhui Wang, Li Dong, Qiang Liu, Owais Khan Mohammed, Kriti Aggarwal, Subhojit Som, Songhao Piao, and Furu Wei. Vlmo: Unified vision-language pre-training with mixture-of-modality-experts. *Advances in Neural Information Processing Systems*, 35:32897–32912, 2022.

[7] Dave Zhenyu Chen, Angel X Chang, and Matthias Nießner. Scanrefer: 3d object localization in rgb-d scans using natural language. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XX*, pages 202–221. Springer, 2020.

[8] Dave Zhenyu Chen, Qirui Wu, Matthias Nießner, and Angel X. Chang. D3net: A speaker-listener architecture for semi-supervised dense captioning and visual grounding in rgb-d scans, 2021.

[9] Dave Zhenyu Chen, Ronghang Hu, Xinlei Chen, Matthias Nießner, and Angel X Chang. Unit3d: A unified transformer for 3d dense captioning and visual grounding. *arXiv preprint arXiv:2212.00836*, 2022.

[10] Xi Chen, Xiao Wang, Soravit Changpinyo, AJ Piergiovanni, Piotr Padlewski, Daniel Salz, Sebastian Goodman, Adam Grycner, Basil Mustafa, Lucas Beyer, et al. Pali: A jointly-scaled multilingual language-image model. *arXiv preprint arXiv:2209.06794*, 2022.

[11] Zhenyu Chen, Ali Gholami, Matthias Nießner, and Angel X Chang. Scan2cap: Context-aware dense captioning in rgb-d scans. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3193–3203, 2021.

[12] Junyoung Chung, Caglar Gulcehre, KyungHyun Cho, and Yoshua Bengio. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555*, 2014.

[13] Marcella Cornia, Matteo Stefanini, Lorenzo Baraldi, and Rita Cucchiara. Meshed-memory transformer for image captioning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10578–10587, 2020.

[14] Angela Dai, Angel X Chang, Manolis Savva, Maciej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5828–5839, 2017.

[15] Chaorui Deng, Qi Wu, Qingyao Wu, Fuyuan Hu, Fan Lyu, and Mingkui Tan. Visual grounding via accumulated attention. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7746–7755, 2018.

[16] Danny Driess, Fei Xia, Mehdi SM Sajjadi, Corey Lynch, Aakanksha Chowdhery, Brian Ichter, Ayzaan Wahid, Jonathan Tompson, Quan Vuong, Tianhe Yu, et al. Palm-e: An embodied multimodal language model. *arXiv preprint arXiv:2303.03378*, 2023.

[17] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9:1735–80, 12 1997. doi: 10.1162/neco.1997.9.8.1735.

[18] Ronghang Hu, Marcus Rohrbach, and Trevor Darrell. Segmentation from natural language expressions. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*, pages 108–124. Springer, 2016.

[19] Ronghang Hu, Huazhe Xu, Marcus Rohrbach, Jiashi Feng, Kate Saenko, and Trevor Darrell. Natural language object retrieval. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4555–4564, 2016.

[20] Yu Jiang, Vivek Natarajan, Xinlei Chen, Marcus Rohrbach, Dhruv Batra, and Devi Parikh. Pythia v0. 1: the winning entry to the vqa challenge 2018. *arXiv preprint arXiv:1807.09956*, 2018.

[21] Justin Johnson, Andrej Karpathy, and Li Fei-Fei. Densecap: Fully convolutional localization networks for dense captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4565–4574, 2016.

[22] Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara Berg. Referitgame: Referring to objects in photographs of natural scenes. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 787–798, 2014.

[23] Dong-Jin Kim, Jinsoo Choi, Tae-Hyun Oh, and In So Kweon. Dense relational captioning: Triple-stream networks for relationship-based captioning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6271–6280, 2019.

[24] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[25] Yicong Li, Xiang Wang, Junbin Xiao, Wei Ji, and Tat-Seng Chua. Invariant grounding for video question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2928–2937, June 2022.

[26] Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain, July 2004. Association for Computational Linguistics. URL https://aclanthology.org/W04-1013.

[27] Junyang Lin, An Yang, Yichang Zhang, Jie Liu, Jingren Zhou, and Hongxia Yang. Interbert: Vision-and-language interaction for multi-modal pretraining. *arXiv preprint arXiv:2003.13198*, 2020.

[28] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016.

[29] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *Advances in neural information processing systems*, 32, 2019.

[30] Ruotian Luo. A better variant of self-critical sequence training. *arXiv preprint arXiv:2003.09971*, 2020.

[31] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, ACL '02, page 311–318, USA, 2002. Association for Computational Linguistics. doi: 10.3115/1073083.1073135. URL https://doi.org/10.3115/1073083.1073135.

[32] Maria Parelli, Alexandros Delitzas, Nikolas Hars, Georgios Vlassis, Sotirios Anagnostidis, Gregor Bachmann, and Thomas Hofmann. Clip-guided vision-language pre-training for question answering in 3d scenes. *arXiv preprint arXiv:2304.06061*, 2023.

[33] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.

[34] Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

[35] Charles R Qi, Or Litany, Kaiming He, and Leonidas J Guibas. Deep hough voting for 3d object detection in point clouds. In *proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9277–9286, 2019.

[36] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30, 2017.

[37] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021.

[38] Steven J Rennie, Etienne Marcheret, Youssef Mroueh, Jerret Ross, and Vaibhava Goel. Self-critical sequence training for image captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7008–7024, 2017.

[39] Arka Sadhu, Kan Chen, and R. Nevatia. Video question answering with phrases via semantic roles. In *NAACL*, 2021.

[40] Naganand Yadati Sanket Shah, Anand Mishra and Partha Pratim Talukdar. Kvqa: Knowledge-aware visual question answering. In *AAAI*, 2019.

[41] Yash Srivastava, Vaishnav Murali, Shiv Ram Dubey, and Snehasis Mukherjee. Visual question answering using deep learning: A survey and performance analysis. In *Computer Vision and Image Processing: 5th International Conference, CVIP 2020, Prayagraj, India, December 4-6, 2020, Revised Selected Papers, Part II 5*, pages 75–86. Springer, 2021.

[42] Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. Vl-bert: Pre-training of generic visual-linguistic representations. *arXiv preprint arXiv:1908.08530*, 2019.

[43] Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. Videobert: A joint model for video and language representation learning. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 7464–7473, 2019.

[44] Hao Tan and Mohit Bansal. Lxmert: Learning cross-modality encoder representations from transformers. *arXiv preprint arXiv:1908.07490*, 2019.

[45] Damien Teney, Peter Anderson, Xiaodong He, and Anton Van Den Hengel. Tips and tricks for visual question answering: Learnings from the 2017 challenge. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4223–4232, 2018.

[46] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[47] Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4566–4575, 2015.

[48] Thang Vu, Kookhoi Kim, Tung M Luu, Thanh Nguyen, and Chang D Yoo. Softgroup for 3d instance segmentation on point clouds. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2708–2717, 2022.

[49] Wenhui Wang, Hangbo Bao, Li Dong, Johan Bjorck, Zhiliang Peng, Qiang Liu, Kriti Aggarwal, Owais Khan Mohammed, Saksham Singhal, Subhojit Som, et al. Image as a foreign language: Beit pretraining for all vision and vision-language tasks. *arXiv preprint arXiv:2208.10442*, 2022.

[50] Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Reinforcement learning*, pages 5–32, 1992.

[51] Junbin Xiao, Angela Yao, Zhiyuan Liu, Yicong Li, Wei Ji, and Tat-Seng Chua. Video as conditional graph hierarchy for multi-granular question answering. In *Proceedings of the 36th AAAI Conference on Artificial Intelligence (AAAI)*, pages 2804–2812, 2022.

[52] Peng Xu, Xiatian Zhu, and David A Clifton. Multimodal learning with transformers: A survey. *arXiv preprint arXiv:2206.06488*, 2022.

[53] Linjie Yang, Kevin Tang, Jianchao Yang, and Li-Jia Li. Dense captioning with joint inference and visual context. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2193–2202, 2017.

[54] Zichao Yang, Xiaodong He, Jianfeng Gao, Li Deng, and Alex Smola. Stacked attention networks for image question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 21–29, 2016.

[55] Shuquan Ye, Dongdong Chen, Songfang Han, and Jing Liao. 3d question answering. *IEEE Transactions on Visualization and Computer Graphics*, 2022.

[56] Kexin Yi, Jiajun Wu, Chuang Gan, Antonio Torralba, Pushmeet Kohli, and Josh Tenenbaum. Neural-symbolic vqa: Disentangling reasoning from vision and language understanding. *Advances in neural information processing systems*, 31, 2018.

[57] Licheng Yu, Patrick Poirson, Shan Yang, Alexander C Berg, and Tamara L Berg. Modeling context in referring expressions. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14*, pages 69–85. Springer, 2016.

[58] Licheng Yu, Hao Tan, Mohit Bansal, and Tamara L Berg. A joint speaker-listener-reinforcer model for referring expressions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7282–7290, 2017.

[59] Zhou Yu, Jun Yu, Yuhao Cui, Dacheng Tao, and Qi Tian. Deep modular co-attention networks for visual question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6281–6290, 2019.

[60] Yaoyao Zhong, Junbin Xiao, Wei Ji, Yicong Li, Weihong Deng, and Tat-Seng Chua. Video question answering: Datasets, algorithms and challenges. In *The 2022 Conference on Empirical Methods in Natural Language Processing*, 2022.