Project Report: Scan2Cap

Felix Wimbauer
Technical University of Munich
felix.wimbauer@tum.de

Nicolas Seppich
Technical University of Munich
nicolas.seppich@tum.De

Supervisor: Dave Zhenyu Chen
Technical University of Munich
zhenyu.chen@tum.de

Abstract

In this work, we investigate the task of generating a description for a target object in context to its environment in the 3D domain. To this end, we propose a pipeline which combines concepts from 3D object detection and visual attention-based captioning. The proposed pipeline first uses VoteNet to extract feature vectors of the scene. It then combines this information with the features of the object of interest, which are extracted by PointNet++, and feeds this data into an LSTM captioning mechanism, that generates a caption of the object in context of the scene.

1. Problem Statement & Motivation

Extracting a detailed and correct description of a 3D scene and included objects is crucial for many tasks, e.g. in robotics for navigation and interaction with objects. However, to the best of our knowledge, there has been no work so far to generate a description of an object representation in point clouds or 3D data.

Therefore, we are interested in implementing a pipeline to obtain a description for a given object in a 3D scan, using state-of-the-art point cloud feature extractors, object detectors, and a captioning mechanism. This allows the object to be placed in a global semantic context within its environment. Furthermore, our project lays down the groundwork for dense captioning, in which you obtain a semantic representation of the scene through computing a caption for all objects in that scene.

2. Related Work

Our work is based on the ScanRefer dataset [2]. This dataset consists of 1513 RGB-D scans of ScanNet [3] and contains approximately 5 unique object descriptions for each object in each scene. The work of [2] will also be used as guideline in this project.

The extraction of features on point clouds is presented by [10], who apply the feature extraction directly on the point cloud on a hierarchical level, allowing the extraction of local features in a global context. The task of object detection on point clouds is studied by [9].

Methods for image captioning using visual attention are described by [13], [6] and [1]. These methods have in common, that they generate a caption for the entire image.

3. Architecture

Given a point cloud $p \in R^{N \times (d+C)}$ and an object in that scene, which is described by a target bounding box $b_{target} \in R^6$, our goal is to generate a meaningful caption for the object embedded in the context of the scene. To this end, we used three different pipelines, which are described in the following.

3.1. Baseline

To extract information from the point cloud, we use a PointNet++ [10] model. To give the network information about which object we are interested in, we add a new feature channel to each point that masks all points that lay within the bounding box of the object. To ensure that we receive meaningful features, we use weights pretrained for classification of the masked object. The idea behind this is that classification will not only use information from the masked objects, but also global features, for example from close-by objects.

To generate the caption, we use a classical LSTM with an appended fully-connected layer. The fully-connected layer acts as a word classifier and maps the hidden state h_t of the LSTM to our vocabulary. As input, the LSTM receives the feature vector extracted from the point cloud and the word embedding vector of the previously generated word. The word embedding is taken from a pre-computed GloVe [8] word embedding matrix. This structure of the iterative caption generation is similar to [13].

3.2. Better Feature Extraction with VoteNet

Because PointNet++ is pretrained to classify the object of interest there is no guarantee that the feature vector will give high-quality information about the global context, thus limiting the baseline approach. In the second architecture iteration, we therefore employ a VoteNet [9] network, which computes a fixed number of object proposals and according feature vectors for our point cloud. Those feature vectors nicely describe the context of our scan and help the network to understand the surroundings of the object we want to describe. Because the number of proposals may vary and the proposals are not in a fixed order, we average pool them to obtain a concise representation of the information. This pooled feature vector is finally concatenated with the feature vector from PointNet++ and the embedding vector of the previously predicted word to then be passed into the LSTM.

3.3. Better Captioning with Attention

Average pooling the feature vectors from VoteNet is not ideal as often only a small number of the object proposals is relevant for the final caption. Therefore, in the third model iteration we replace the average pooling step from before with an attention mechanism, as it is described in [13]. The attention mechanism receives the hidden state of the captioning LSTM from the previous iteration and the feature vectors of the object proposals. It then uses a series of fully-connected layers to predict relevancy scores for the different object proposals that are turned into probabilities using the softmax function. Instead of average pooling, we can now multiply all feature vectors with their respective probability and sum over all of them. This approach allows our model to select the most relevant object for each token in the caption and make word predictions that better match the the context of the scene.

Figure 1 summarizes the project architectures.

4. Experiments & Results

To allow a quantitative comparison between the proposed architectures, we have constructed the same training pipeline for the three models. As mentioned above, the models receive the point cloud of a scene, the ground truth bounding box of the target object and the tokenized description as input. Similar to [13], all architectures make use of teacher-forcing during training: instead of inserting the previous predicted word, the LSTM receives the previous ground truth word as input for captioning along with the coded feature vectors of the scene. This allows for faster convergence by minimizing drift between the ground truth token and predicted word in training time. In evaluation time, the network takes the previous predicted word as input.

As metrics for performance evaluation we rely on BLEU [7], ROUGE-L [5], METEOR [4] and CIDEr [12]. In this paper we focus on the BLEU score to determine the best models as in [13].

For our models based on VoteNet we filter out all objects that have an object frequency value below 0.75. We also consider only the 8 closest objects, because these thresholds lead to better results representing the local scene context.

4.1. Quantitative Improvements

In this experiment we want to investigate the influence of the architectural improvements we propose. All models include the same pre-trained masked PointNet++ feature extractor. Table 1 shows the quantitative results.

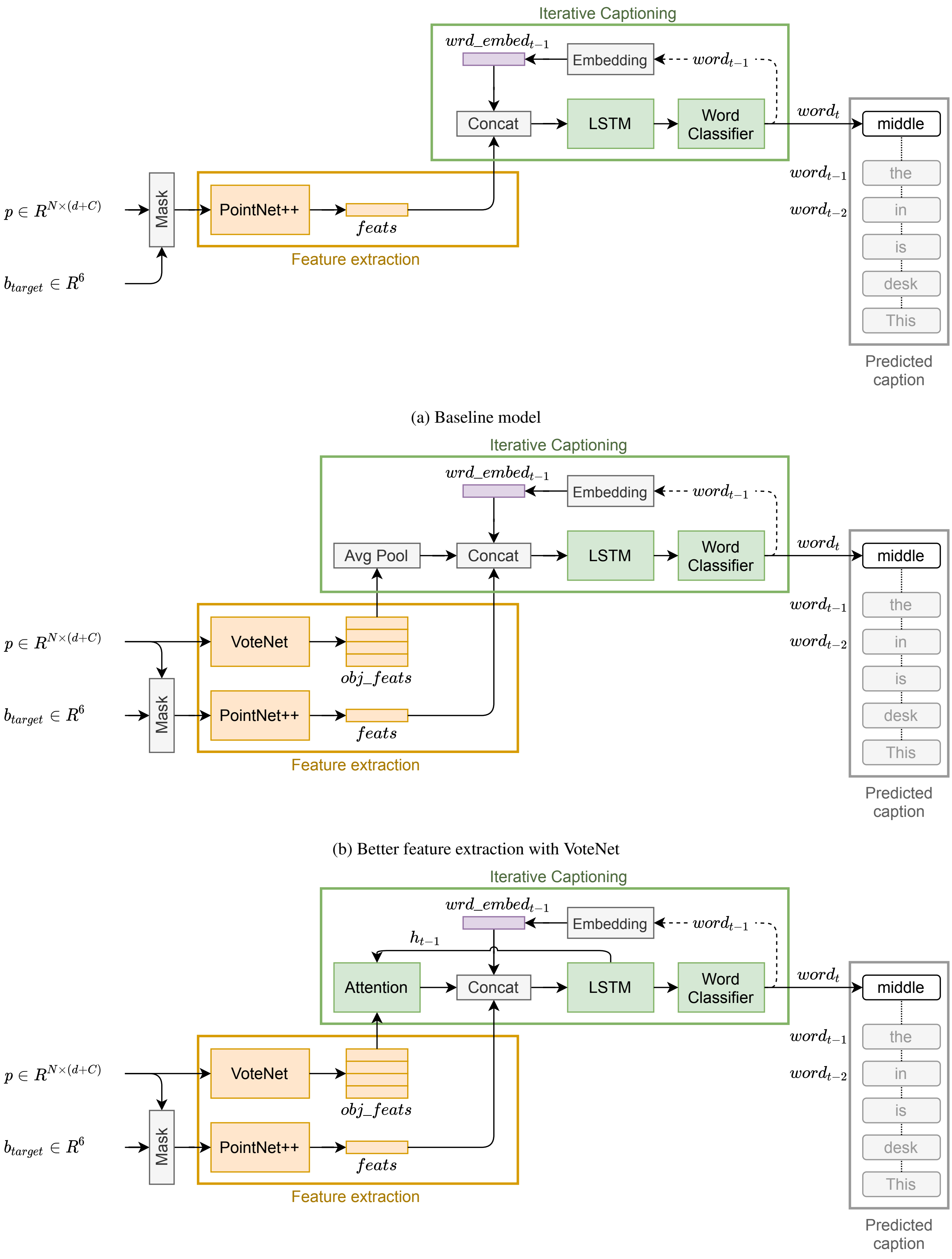
As a first result, we can conclude that the baseline model achieves reasonably good results that are not far from the improved architectures. Comparing the BLEU-4 score to the results of the COCO Image Captioning Challenge 2015 ”quote”, our models outperform the best-performing architectures. Of course, the question always arises whether our task and the tasks of COCO Image Captioning are comparable. We argue that our vocabulary and gt sentence structure is much simpler and therefore leads to better results.

Secondly, the results show that improving the feature extractor (improved architecture) and improving the captioning meachnism (attention) lead to higher scores. The best results are obtained by the attention model, except for the BLEU-4 score where the improved architecture is slightly better.

4.2. Qualitative Analysis

In this experiment we want to analyze the quality of the predicted descriptions. For this purpose, we compare two exemplary ground truth and predicted description pairs shown in Figure X. The first pair shows the comparison between the ground truth and the predictions of our three architectures. The result shows that the three models predict a correct description that puts the object into the context of its environment. There are slight differences in the context the models make use of, especially for the attention model. This indicates that the attention mechanism is capable of producing a description that is rather unusual. The baseline and improved architecture were able to predict a chair near a table, which is a fairly common scenario. The second pair, which illustrates a long ground truth description and the prediction of the attention model, shows the accuracy of the prediction, even if it is quite short.

For all predicted descriptions we can observe a simple structure with two sentences. As an explanation we claim that the statistics of the ScanRefer data set influences this behavior. Therefore we have analyzed the number of tokens per description and the number of records in each description. Figures Z and Y support this assertion by showing the hsitograms of the number of tokens and record lengths for our predictions and for the ScanRefer data set, respectively.



(a) Baseline model

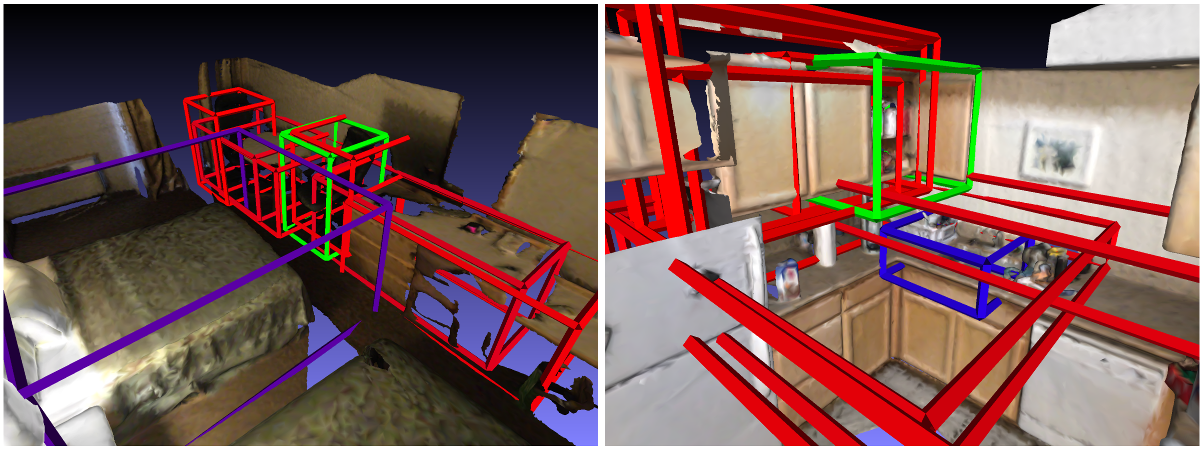
(b) Better feature extraction with VoteNet

(c) Better captioning with attention

Figure 1: Model architectures for project

Model	**BLEU-4**	**ROUGLE-L**	**METEOR**	**CIDEr**
Baseline	0.434	0.569	0.617	0.692
With VoteNet	**0.447**	0.580	0.625	0.757
With Attention	0.447	**0.584**	**0.626**	**0.792**

Table 1: Quantitative results



(a) **GT**: there is a black chair. it is next to a cabinet on the side of the room .
BL: this is a black chair . it is to the left of the table .
VN: this is a black chair . it is to the left of the desk .
Att: this is a black chair . it is at the end of the bed .

(b) **GT**: the open kitchen cabinet is directly above the sink and the water container. the open kitchen cabinet is a brown box with one side hanging out .
Att: this is a white kitchen cabinet . it is above the sink .

Legend: **GT**-Groundtruth, **BL**-Baseline, **VN**-With VoteNet, **Att**-With Attention

Bounding boxes: **Green**-Target object, **Red**-**Blue**-Lowest to highest attention for marked token

Figure 2: Examples from the ScanRefer validation set

4.3. Inference without GT

5. Coclusion

References

[1] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6077–6086, 2018.

[2] Dave Zhenyu Chen, Angel X Chang, and Matthias Nießner. Scanrefer: 3d object localization in rgb-d scans using natural language. *arXiv preprint arXiv:1912.08830*, 2019.

[3] 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.

[4] Michael J. Denkowski and Alon Lavie. Meteor universal: Language specific translation evaluation for any target language. In *WMT@ACL*, 2014.

[5] Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *ACL 2004*, 2004.

[6] Jiasen Lu, Caiming Xiong, Devi Parikh, and Richard Socher. Knowing when to look: Adaptive attention via a visual sentinel for image captioning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 375–383, 2017.

[7] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *ACL*, 2002.

[8] 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.

[9] 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 International Conference on Computer Vision*, pages 9277–9286, 2019.

[10] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *Advances in neural information processing systems*, pages 5099–5108, 2017.

[11] Anna Rohrbach, Marcus Rohrbach, Ronghang Hu, Trevor Darrell, and Bernt Schiele. Grounding of textual phrases in images by reconstruction. In *European Conference on Computer Vision*, pages 817–834. Springer, 2016.

[12] Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. *CoRR*, abs/1411.5726, 2014.

[13] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*, pages 2048–2057, 2015.