A. Zero-Shot Experiments

In this section, we discuss our zero-shot experiments on the 3RScan dataset [35]. First, we discuss the collection of referential sentences. 3RScan is a large-scale, real-world dataset that contains 1482 3D reconstructions. Second, we report the zero-shot listening accuracy of our proposed model MVT-ScanEnts compared to the original MVT model.

A.1. Referential Sentences Collection for 3RScan

We collect referential sentences for the validation scans present in the 3RScan dataset. We follow the data collection approach presented in [5]. The dataset collection pipeline consists of two stages; data collection and data verification. In Figure 6, we show the AMT interface used for data collection along with actual collected data examples. We collect in total 840 referential sentences covering all of the 47 scans of the official validation split.

A.2. Zero-Shot Listening Results

We do zero-shot neural listening tests using a pre-trained MVT-ScanEnts model, which is trained on Nr3D using the rich annotations of ScanEnts3D and our novel proposed losses and using an original MVT model trained on Nr3D as in [34] without ScanEnts3D. We center the input scene point cloud around the origin point and transform the point cloud to become axis-aligned as described in [23]. In Table 8, MVT-ScanEnts outperforms MVT on out-of-domain 3D scenes by 4.17%. The result shows that neural listeners when trained on ScanEnts3D, can exhibit better 3D scene understanding even on unseen scans.

Method	Overall Acc.
MVT [34]	11.80%
MVT-ScanEnts	**15.97%** (+4.17%)

Table 8. Zero-Shot listening performance on our collected referential sentences on the 3RScan dataset. MVT-ScanEnts outperforms the original MVT model on test examples of unseen scans.

B. ScanEnts3D Dataset Analysis

In this section, we provide a more detailed analysis of our proposed ScanEnts3D dataset. In Figure 7, we show a breakdown of the extracted pairwise spatial relationships between the scan entities in ScanEnts3D. In total, we extract using existing spatial relation classifiers [43] 24,028 pairwise spatial relations for the Nr3D dataset and 15,278 pairwise spatial relations for the ScanRefer dataset.

In Figure 8, we show the classes most used as anchor objects for both Nr3D and ScanRefer datasets. We observe that the most used anchor classes are walls, chairs, windows, and doors. We also observe that only 363 fine-grained object classes are used for the anchor objects.

In Figure 10, we show a histogram of the number of scan entities of ScanEnts3D for the Nr3D and ScanRefer datasets. The mean number of scan entities in Nr3D is 2.5, with a standard deviation of 1.17. The mean number of scan entities in ScanRefer is 3.96 with a standard deviation of 1.45.

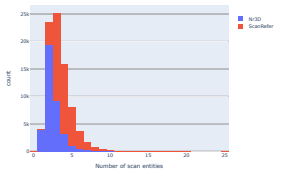


Figure 10. **Histogram of the number of ScanEnts3D scan entities present in Nr3D and ScanRefer datasets**.

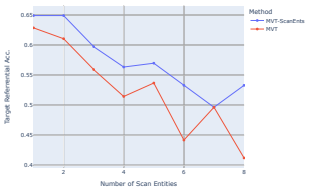


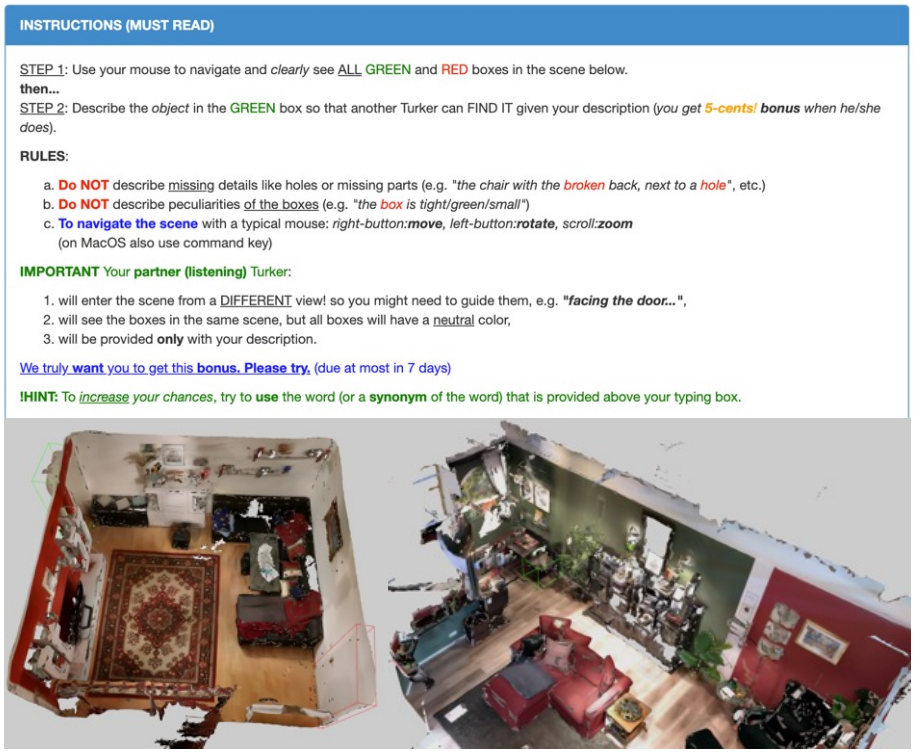
Figure 11. **Comparison between the performance of MVT-ScanEnts and MVT models when increasing the number of scan entities and the number of same-class distractor objects.** The performance generally decreases when increasing the number of the scan entities and the same-class distractor objects.

C. ScanEnts3D Dataset Collection

This section discusses in detail the two phases of our ScanEnts3D curation. Figure 9 shows the user interface we implemented.

Annotation Phase. An annotator is given an utterance and a 3D scene. While the utterance generally describes one specific object in the 3D scene, the annotator is first asked to mark all the nouns (entities) that describe specific objects in the given 3D scene (e.g., chair, table, etc.) in the utterance. Then, for each selected entity in the given utterance, the annotator must highlight the corresponding 3D objects in the given 3D scene. The annotator can zoom, pan or rotate the 3D scene to find the corresponding 3D objects. Each annotator is provided with one random utterance at a time. We assign one annotator for each example.

Review Phase. A reviewer is given one annotated example randomly and is asked to determine whether the example was correctly annotated. If the example was annotated incorrectly, the reviewer is then requested to correct and fix the annotation. The reviewer is shown a similar user interface to the annotator. Each submission is reviewed by one reviewer.



"The door is to the right of the red wall with a television and television stand on it."

"The chair closest to the large green plant."

Figure 6. **User interface for the collection of referential sentences for the 3RScan zero-shot experiment**. On the top, we show the detailed instructions provided to the annotator to ensure the task requirements are clear and straightforward. On the bottom (b), we show two examples of the resulting annotations. The target objects are the ones inside the green bounding boxes, while the same class distractor objects are in the red bounding boxes.

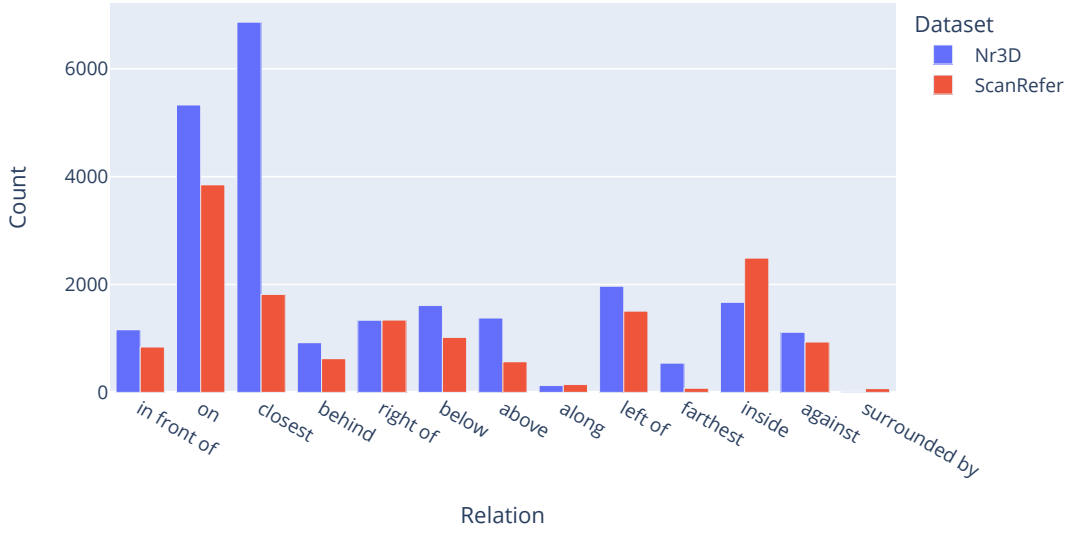
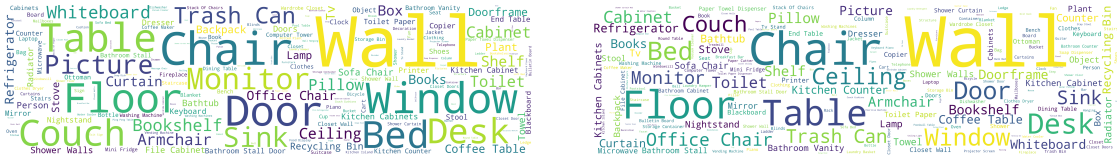


Figure 7. **Breakdown of the extracted pairwise spatial relationships of Nr3D and ScanRefer datasets.** Despite their similar nature, we see that in terms of spatial relations types used to describe objects, there is a noticeable discrepancy among their annotations.

D. Neural Listeners

D.1. SAT-ScanEnts

This section discusses our modifications to the SAT [68] neural listener. For the cross-attention map loss, since the



(a) (b)

Figure 8. **Wordclouds depicting the most common object classes in (a) Nr3D and (b) ScanRefer datasets.** The font size of each printed class name is proportional to its underlying frequency (better seen by zooming in).

(a)

(b)

Figure 9. **User interface for the ScanEnts3D dataset collection**. On the top (a), we show the detailed instructions provided to the annotator to ensure the task requirements are clear and straightforward. On the bottom (b), we show an example of a resulting annotation.

SAT model is using transformer encoder layers, it does not contain an apparent cross-attention operation between the object tokens and the language tokens. To address that, we add one transformer decoder layer as shown in Figure 13, where we apply our proposed cross-attention map loss. The Cross-Attention Map loss encourages the network to attain high relevance values between both the objects and the words representing the same underlying scan entity. The target

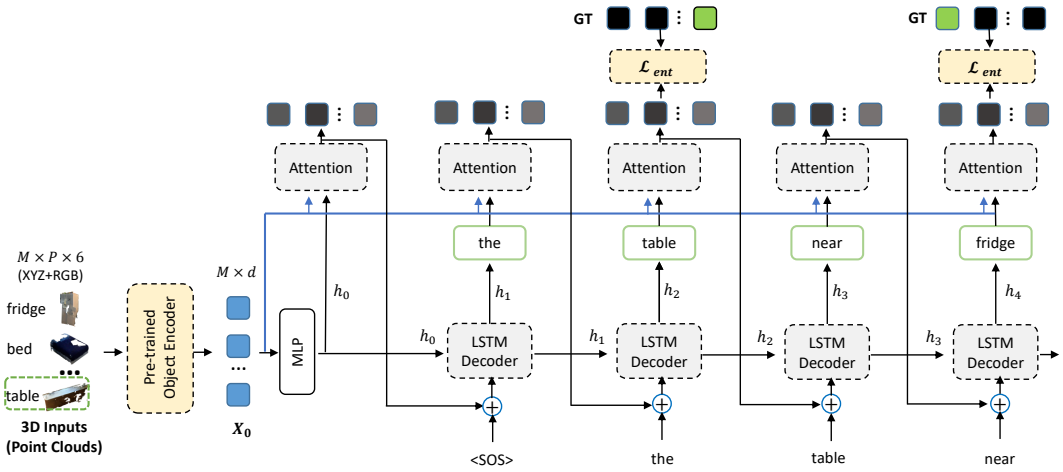


Figure 12. **Our proposed SATCap-ScanEnts model.** SATCap-ScanEnts is based on the "Show, Attend, and Tell" model [56]. We use a pre-trained 3D object encoder for encoding the scene objects. The decoder is an LSTM [27], where we apply our proposed loss $\mathcal{L}_{ent}$ during training. If the word to be predicted by the decoder in the current time-step (like table and fridge) corresponds to a scan entity in the target caption, the attention values to the 3D objects that belong to the scan entity should be higher than that of the objects that do not belong to that scene entity.

matrix Y_{attn} is a binary matrix of shape $M \times N$, where a cell $(y_{i,j})$ has a value of 1 if the ith object and the jth word correspond to one another. To cover the case of the 3D objects that do not belong to any of the scan entities in the given utterance, we add an extra word token called <NM> as shown in Figure 13 and for every object k that does not belong to any of the scan entities, we set the cell $(y_{1,k}$ to the value of 1. The <NM> mention token is always added after the <CLS> token. The anchor prediction loss and the same-class distractor loss are applied to the late context-aware feature.

D.2. 3DJCG-ScanEnts

The 3DJCG [12] model is an object-detection-based model, where the input to the model is a point cloud of a 3D scene and an input utterance. The task of the model is to localize the target object via predicting an axis-aligned 3D bounding box around the target object. We apply the anchor prediction loss as discussed in the main paper in Section 4.1.1. We apply an MLP ϕ on the feature vectors of the detected object proposals to obtain a confidence score $x_i \in [0, 1]$ of whether the object proposal is an anchor object or not. To construct the ground truth for the anchor prediction loss, we follow a similar approach as in [20, 67]. For each object proposal, the ground-truth label is $y_i \in \{0, 1\}$. We set the label $y_j = 1$ for the j^{th} proposal that has the highest IOU with the box of one of the ground truth anchor objects. We apply a binary cross entropy loss between the predicted confidence vector X and the ground truth vector Y as in $\mathcal{L}_{\text{anc}} = \text{BCE}(\text{X}, \text{Y})$. The total loss used in the 3DJCG model would be $\mathcal{L} = \mathcal{L}_{org} + \mathcal{L}_{anc}$, where $\mathcal{L}_{org}$ represents the original losses used.

E. Neural Speakers

E.1. SATCap-ScanEnts

In Figure 12, we show the SATCap-ScanEnts model, which is discussed in Section 4.2.1 in the main paper. The SATCap-ScanEnts model is based on the "Show, Attend, and Tell" model, which is a 2D image captioning model. To make it amenable to purely 3D inputs, we replace the image encoder with the encoder network found in the MVT model [34], which is a point cloud PointNet++ encoder together with 3D object self-attention layers. For the decoder network, we use a unidirectional LSTM cell [27]. The speaker model is trained via teacher-forcing [60]. our proposed entity prediction loss $\mathcal{L}_{ent}$ is applied during the decoding steps in the following manner. At each decoding step, if the current word to be predicted corresponds to a scan entity (table and fridge words in Figure 12), our loss pushes the object(s) corresponding to the underlying scan entity to be the highest scoring among all objects present in the input scene. The entity prediction loss is not applied if the current word to be predicted does not correspond to a scan entity.

F. Implementation Details

For the listening experiments, we used the same hyper-parameters specified in MVT [34] and SAT [68]. For the 3D object localization experiment, we use the same hyper-parameters of 3DJCG [12]. We use one NVidia V100 GPU in each of our experiments. We use the same hyper-parameters found in [64] for the neural speakers.

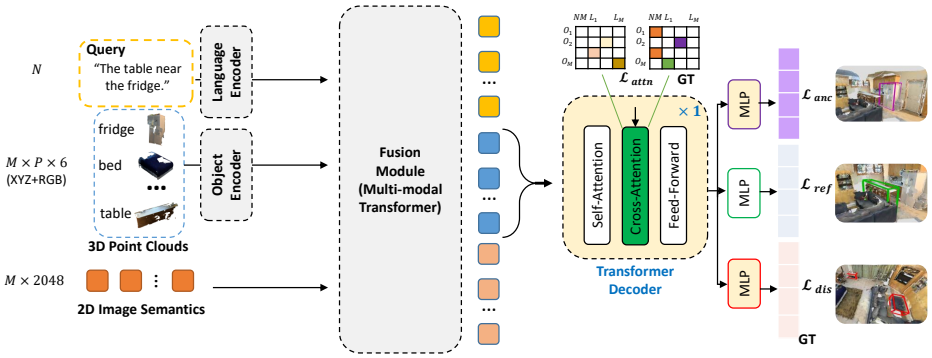


Figure 13. **Our proposed SAT-ScanEnts model.** We add a cross-attention layer operating on the 3D object and language features. Our proposed losses are applied after the added cross-attention layer in a similar manner to the MVT- datasetSuffix model.

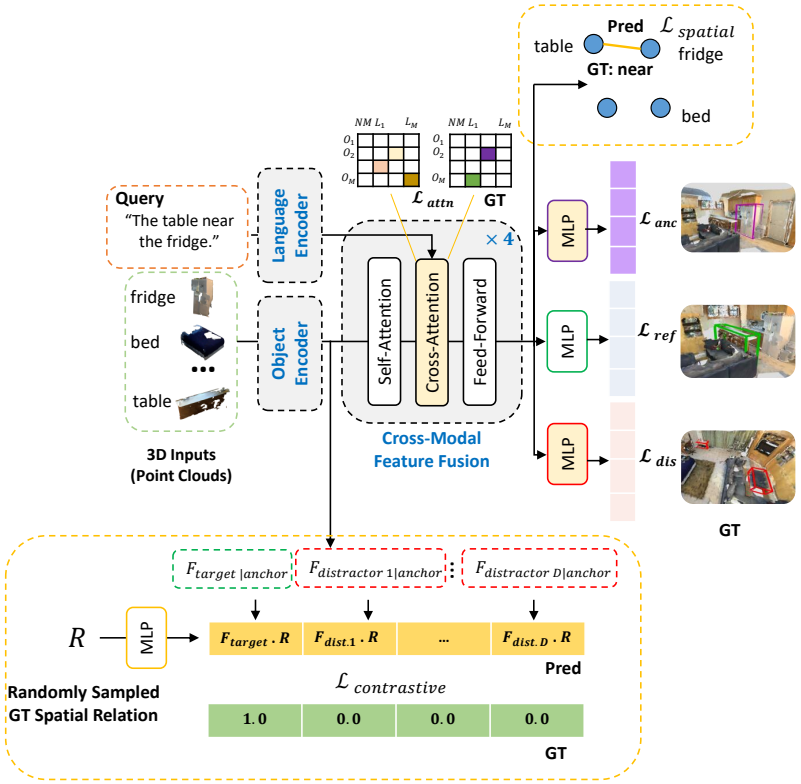


Figure 14. **Our proposed modifications to MVT-ScanEnts for exploiting the pair-wise spatial relationships that improve the listening performance.** We propose two losses; $\mathcal{L}_{contrastive}$ and $\mathcal{L}_{spatial}$. $\mathcal{L}_{contrastive}$ aims at better understanding the spatial relationship between the target object and an anchor object while contrasting the spatial relation between the anchor and the same-class distractor objects. The $\mathcal{L}_{spatial}$ aims at predicting the spatial relationships between the object pairs where their ground truth spatial relationship is known.

G. Ablation Studies and More Quantitative Results

Usefulness of exploiting the pairwise spatial relationships We exploit the extra annotations of the extracted pairwise spatial relationships discussed in Section 3.2 in the main paper. In Figure 14, we show our modifications to MVT-ScanEnts neural listener. We introduce two losses that exploit the pair-wise spatial relations. The first loss $\mathcal{L}_{contrastive}$ is a contrastive loss that operates as follows; for a training example, we randomly sample a ground-truth spatial relationship between the target object and an anchor object (the relationship does not necessarily present in the input utterance). The sampled spatial relationship is valid between the target object and the anchor while it is valid for none of the same-class distractor objects. We embed the spatial relation class into a

vector R with dimension d. We then concatenate the object feature (computed by the PointNet++ encoder [47]) of the anchor object to the target object feature and the feature of each of the same-class distractor objects. The concatenated features are then transformed using an MLP and the generated features are called F each of dimension d as shown in Figure 14. We then apply a cosine similarity between the embedded feature of the spatial relation R and each of the F features. The $\mathcal{L}_{contrastive}$ loss is the cross entropy between the predicted distribution and the ground-truth vector which is a one-hot vector, where the value of one corresponds to the target object. The second loss is called $\mathcal{L}_{spatial}$ and it operates on the context-aware features that are computed after the cross-modal fusion between the 3D objects and the input language and it works in the following manner. For each of the object pairs where the ground-truth spatial relationships are known, we apply a spatial relation classification loss on the concatenated features of the object pairs. To summarize, the spatial relationship losses are defined as $\mathcal{L}_{rel} = \mathcal{L}_{contrastive} + \mathcal{L}_{spatial}$.

As shown in Table 9, we observe an improvement in the listening performance when combining the spatial relationship losses with both the anchor prediction loss and the same-class distractor loss. However, the performance didn't improve when using all four losses together.

$\mathcal{L}_{attn}$	$\mathcal{L}_{anc}$	$\mathcal{L}_{dis}$	$\mathcal{L}_{rel}$	Overall
✓	✓			58.7%
✓	✓	✓		59.3%
✓	✓		✓	**59.7%**
✓	✓	✓	✓	59.3%

Table 9. **Ablation study on using our proposed losses that exploit the extracted spatial relationships in the MVT-ScanEnts model.** Our proposed losses cause an improvement in the listening performance (+1.0%) when being used with the anchor prediction loss and the same-class distractor loss.

Performance of listener with an increasing number of scan entities. In Figure 11, we observe that the listening performance decreases when the difficulty of the input utterances increases where more scan entities and same-class distractor objects are involved. MVT-ScanEnts performs better than the original MVT model.

Effectiveness of the pre-trained encoder in the M2cap-ScanEnts. In Table 11, we show the usefulness of using a pre-trained object encoder (trained on the neural listening task), which is discussed in Section 4.2.2 in the main paper. The usage of the pre-trained encoder improves the performance of the M2Cap-ScanEnts neural listener in all of the four captioning metrics on the Nr3D dataset.

Effectiveness of losses in MVT-ScanEnts. In Table 10, we show an ablation study upon using our proposed losses on the MVT-ScanEnts neural listeners. Following [5], we do testing using five random seeds, and we report the mean and the standard deviation of the accuracy.

H. Limitations

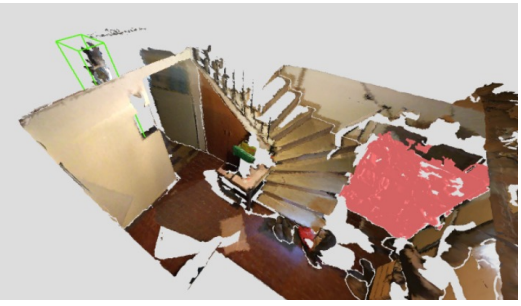
Our extension of Nr3D and ScanRefer with ScanEnts3D is based on the original utterances in these two datasets. Hence, we are constrained in a linguistic corpus where the grounding language used is English. It would be of interest to explore the efficacy of our method and annotation approach to other languages, especially to reduce the possible biases a restrictive set of cultural groups might be introducing. Moreover, despite achieving SoTA results in two popular and essential tasks for 3D-based visio-linguistic grounding tasks, it is clear that our methods are not yet on par with human-level performance (see Fig. 15). More studies around competing methods, the underlying supervision used, and even transfer-learning approaches that can leverage e.g., large-scale 2D-based data, or recent foundational models, are expected to be fruitful in closing the gap between learning-based methods and human efficacy.

$\mathcal{L}_{attn}$	$\mathcal{L}_{anc}$	$\mathcal{L}_{dis}$	Overall	Easy	Hard	View-dep.	View-indep.
			55.1%±0.3%	61.3%±0.4%	49.1%±0.4%	54.3%±0.5%	55.4%±0.3%
✓			56.6%±0.2%	63.0%±0.3%	50.5%±0.3%	55.4%±0.4%	57.2%±0.2%
		✓	56.9%±0.3%	63.5%±0.3%	50.6%±0.3%	55.3%±0.4%	57.8%±0.4%
✓		✓	57.4%±0.3%	64.3%±0.4%	50.8%±0.4%	55.6%±0.6%	58.3%±0.3%
	✓	✓	57.9%±0.2%	63.7%±0.2%	52.3%±0.2%	56.0%±0.2%	58.9%±0.3%
	✓		58.1%±0.3%	63.8%±0.5%	52.6%±0.3%	56.7%±0.3%	58.8%±0.4%
✓	✓		58.7%±0.3%	64.6%±0.4%	53.1%±0.4%	**57.5%±0.3%**	59.3%±0.4%
✓	✓	✓	**59.3%±0.1%**	**65.4%±0.3%**	**53.5%±0.2%**	57.3%±0.3%	**60.4%±0.2%**

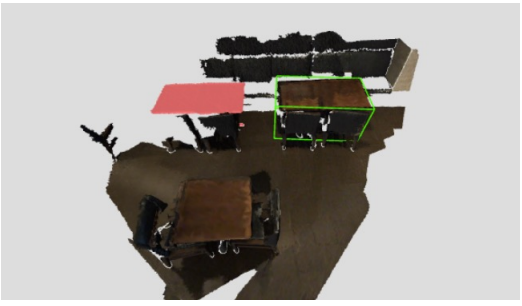
Table 10. **Ablation study on neural listeners.** We ablate different combinations of our proposed auxiliary losses on the MVT neural listener, trained on Nr3D using our proposed ScanEnts3D dataset.

Arch.	Nr3D			
	C	B-4	M	R
M2Cap	86.15	37.03	30.63	67.00
M2Cap-ScanEnts w/o Pre-trained Encoder	88.68	37.29	31.06	67.35
M2Cap-ScanEnts	**93.25**	**39.33**	**31.55**	**68.33**

Table 11. **Effectiveness of using the pre-trained object encoder in M2Cap-ScanEnts**. Using the pre-trained object encoder helps improve the performance of M2-Cap-ScanEnts neural speakers in all of the four metrics.



"if you walk down the stairs and take a right, you will find the shelf you seek."



"So, if you have the two tables in the back and one table in the front, you want the right side"

Figure 15. **Failure examples where the MVT-ScanEnts model struggles to identify the target object (green) because of the complex language descriptions. The incorrect predictions are highlighted in red color.**