GRAVL-BERT: Graphical Visual-Linguistic Representations for Multimodal Coreference Resolution

Danfeng Guo[1]*, **Arpit Gupta**[2], **Sanchit Agarwal**[2]†, **Jiun-Yu Kao**[2]
Shuyang Gao[2], **Arijit Biswas**[2], **Chien-Wei Lin**[2], **Tagyoung Chung**[2], **Mohit Bansal**[2,3]
[1]University of California, Los Angeles
[2]Amazon Alexa
[3]University of North Carolina, Chapel Hill

Abstract

Learning from multimodal data has become a popular research topic in recent years. Multimodal coreference resolution (MCR) is an important task in this area. MCR involves resolving the references across different modalities, e.g., text and images, which is a crucial capability for building next-generation conversational agents. MCR is challenging as it requires encoding information from different modalities and modeling associations between them. Although significant progress has been made for visual-linguistic tasks such as visual grounding, most of the current works involve single turn utterances and focus on simple coreference resolutions. In this work, we propose an MCR model that resolves coreferences made in multi-turn dialogues with scene images. We present GRAVL-BERT, a unified MCR framework which combines visual relationships between objects, background scenes, dialogue, and metadata by integrating Graph Neural Networks with VL-BERT. We present results on the SIMMC 2.0 multimodal conversational dataset, achieving the rank-1 on the DSTC-10 SIMMC 2.0 MCR challenge with F1 score 0.783. Our code is available at https://github.com/alexa/gravl-bert.

1 Introduction

Powered by advances in machine learning, intelligent agent systems have seen their capacity expand in recent years. Devices with intelligent assistants have become ubiquitous in everyday life. These systems can handle short task-oriented dialogues, but are limited to speech or text inputs and outputs. Motivated by the widespread adoption of such agents and multimodal devices with screens that house them, multimodal (visual-linguistic) understanding has become a promising discipline for researchers. The next-generation of intelligent assistants are expected to jointly understand multimodal data, i.e., text, image, video, and audio together and their associations.

Figure 1: An example of Multimodal coreference resolution (MCR). Based on the image and dialogue context, there can be multiple ways to refer to the highlighted red coat. Each of them require different information to locate the target.

Within multimodal understanding, an important area of research is Multimodal Coreference Resolution (MCR). It is a crucial capability as user references can span across modalities in a multimodal environment. Moreover, MCR is a challenging problem even when compared to text coreference resolution and visual question answering tasks because, in MCR, two participants can simultaneously refer to objects while looking at a scene from a shared perspective. In this dynamic frame of reference, the notion of left/right, first/second is constantly shifting and the model cannot rely on the fixed position of objects in the scene. Different from both VQA and textual anaphora resolution tasks, there are distinct ways to refer to an object in the MCR task. As the scene may contain a large number of similar objects, it is natural to refer to them by relative position with respect to other objects, front and back w.r.t the camera, and w.r.t. accessory objects like shelves and tables, so as to easily indicate the target object(s). Such references are unusual in other related tasks.

Furthermore, as our focus is on the MCR task within task-oriented dialogues, the objects being

*corresponding author: lyleguo@g.ucla.edu
†corresponding author: agsanchi@amazon.com

Proceedings of the 29th International Conference on Computational Linguistics, pages 285–297
October 12–17, 2022.

referenced are typically associated with a back-end database that also provides metadata information such as price, brand, and size for these objects. This adds an additional dimension to the coreferencing task as the users can also refer to objects based on these non-visual metadata attributes. Finally, as conversations involve multi-turn dialogue, MCR also requires reasoning over dialogue context to resolve references like "*show coat next to the shirt you suggested previously*". An example is shown in Figure 1. Resolving these coreference cases requires various multimodal information extraction and reasoning capabilities. For example, to identify "*the red coat next to the black one*", the model must infer the target's visual features (red), as well as its neighborhood information (a black coat). The user might also say "*I'll take the first one*", if they are choosing from one of the coats offered previously.

Notable progress has been made on multimodal frameworks. Recent models have shown excellent performance on various multimodal tasks including Visual Question Answering (VQA) (Huang et al., 2019b; Su et al., 2020), Visual Commonsense Reasoning (Zheng et al., 2020b; Su et al., 2020), Visual Grounding (Zheng et al., 2020a; Deng et al., 2018) and Image Captioning (Huang et al., 2019a). All these models take both images and text as inputs. Both visual and linguistic tokens are sent to an autoencoder to learn shared representation and perform downstream tasks. However, such methods are still not developed in a way to be able to solve complex MCR scenarios. For example, they may fail when a query has dialogue context instead of a single short sentence, or when there are anaphoric references in the query. Finally, most of these models are not designed to handle external knowledge sources or to scale to hundreds of objects in a single scene.

Motivated by these challenges, we propose a new framework, GRAVL-BERT(Graphical Visual-Linguistic BERT), that can simultaneously reason over dialogues, objects and their relationships, scene information, and object metadata. Our major contributions are as follows:

1. We present GRAVL-BERT, a unified BERT-based framework for encoding and reasoning over dialogues grounded in scenes.
2. GRAVL-BERT
   (a) Incorporates additional knowledge sources in the form of object metadata to also support coreferencing based on non-visual features like brand and price.
   (b) Represents scene objects as a graph and encodes them using Graphical Convolutional Network (GCN) to enable reasoning and coreferencing involving complex spatial relationships.
   (c) Adds information about object's surrounding by explicitly sampling from its neighborhood and generating captions describing the object. This enables coreferencing involving surrounding context (e.g., accessory objects like shelves and tables)
3. Finally, we show the importance of pre-training on dialogue dataset for the task of MCR.

We present results of GRAVL-BERT on the SIMMC 2.0 dataset (Kottur et al., 2021) which involves dialogues between a customer and an agent in the shopping domain. We participate in the SIMMC 2.0 challenge for the task of MCR, where the goal is to resolve the references and identify the target object(s) in the scenes. We achieve SOTA performance with 0.76 object-level F1 score on devtest set and 0.78 on test set. We note that this dataset is available for research and non-commercial use.

2 Related Works

Visual Grounding. Visual Grounding (VG) is an area very close to MCR. Given a query, it aims to find the most relevant target in an image. Some widely-used datasets for VG are RefCOCO, RefCOCO+, and RefCOCOg (Yu et al., 2016). Their queries are usually short and simple, e.g., the cat jumping over the fence. JR-Net (Jain and Gandhi, 2021) which achieves SOTA on this task encodes images and queries separately and then uses a joint-reasoning and a multi-level fusion module to merge the features and generate the results. VLT (Ding et al., 2021) converts image features into the same format as language token embeddings and uses BERT followed by a masked decoder to locate the target. A-ATT (Deng et al., 2018) concatenates visual and linguistic features together and uses accumulative attention layers to focus on the key targets.

Visual-Linguistic Frameworks. Multimodal frameworks that support visual (i.e., image, video) and linguistic inputs (i.e., caption, dialogue) can be fine-tuned for various tasks including MCR. In early works, most models encoded visual and

linguistic features separately and combined them only at a later stage. For example, both MTN (Le et al., 2019) and LXMERT (Tan and Bansal, 2019) have two separate encoders for visual and linguistic inputs and then use a query-aware encoder and cross-modality encoder respectively to extract visual features related to query text. In recent works, increasing number of high-performing models adopt an early-fusion strategy. They first extract the regions of interest (ROI) features from visual inputs, convert them into token embeddings, and concatenate with text embeddings. Then they employ BERT (Devlin et al., 2019) to learn the cross-modal associations and perform different tasks. For instance, ViL-BERT (Lu et al., 2019) concatenates text embeddings and image embeddings together and sends them to BERT. Oscar (Li et al., 2020b), VinVL (Zhang et al., 2021), 12-in-1 (Lu et al., 2020), Unicoder-VL (Li et al., 2020a), and Unified VLP (Zhou et al., 2020) all improve upon ViL-BERT by adding better image features, new pretraining strategies, or multiple datasets for pretraining. VL-BERT (Su et al., 2020) is currently one of the most popular benchmark frameworks for visual-linguistic tasks. It uses Fast R-CNN to extract visual features of objects and scene images, and concatenates them with linguistic token embeddings. The combined features are then fed into a BERT module, which is fine-tuned for various downstream tasks.

Graphical Models. Aside from the aforementioned works, another approach is to represent all ROIs in the scene as one graph. Graph R-CNN(Yang et al., 2018) and GCN-LSTM (Yao et al., 2018) encode images as graphs whose nodes represent objects and edges represent the relationships between objects. The generated graph representations can be used for downstream tasks like VQA and Image Captioning. (Damodaran et al., 2021; Yang et al., 2019) show that scene graphs improve model performance on these tasks.

3 Methodology

Problem Formulation. Given inputs (D, I, N, M, Q) where $D = \{D_1, D_2, ..., D_k\}$ is the dialogue text split into k turns, $I = \{I_1, I_2, ..., I_k\}$ are the scene images for the dialogue turns, $N_j = \{N_{j1}, N_{j2}, ...\}$ is the set of objects inside each scene $I_j \in I$, $M_{ij} = \{M_1, M_2, ...\}$ are the metadata attributes of each object N_{ij} and Q is the user query referring to one or more objects, our task is to predict a label $y_n \in \{0, 1\}$ for each object $n \in N$ that indicates whether the object n is being referenced by the query. The user query can involve spatial references, visual references, metadata based references, or any combination of these.

Model Details. Our model builds upon VL-BERT. We extend the framework from single utterance input to multi-turn dialogue input, with each dialogue turn associated with its respective scene image. In addition, the scene objects can have external knowledge base (e.g., metadata) associated with them. The architecture for our model is shown in Figure 2. The model takes 4 different streams of input: linguistic, visual, segment, and position. Input from different streams are combined via feature-wise addition.

The visual stream consists of the visual features from the whole scene, the candidate object and its surroundings. For each of the visual component, we use Fast-RCNN (Girshick, 2015) to extract the features and then augment it with bounding box location to add spatial information. We further add a GCN layer to explicitly capture the relative position of each object with respect to others.

The linguistic stream includes the dialogue context, user query and candidate object's metadata. We flatten the structure of metadata and convert it into a string of the form *"key1 value1 key2 value2 ..."*. We also add two special tokens, an integer feature T which denotes the distance from the turn when the object was last offered by agent, and a string S, which indicates that the corresponding visual features (in the visual stream) are from the object's surroundings and not the object itself.

The segment stream is used to distinguish the dialogue context, user query and object metadata inputs. It has three different values, corresponding to these input types. The position stream contains token positional embeddings, which is same as the one used in the original BERT model.

Note that, we feed only one object into the model at a time. This allows our system to scale well for scenes that may contain large number of objects. For instance, in the SIMMC 2.0 dataset, many scenes contain more than 100 objects with metadata sequence length larger than 50 for each object, making it impractical to feed all 5K object instances into the system. We supplement our model with GCN based structure to mitigate strong independence assumptions

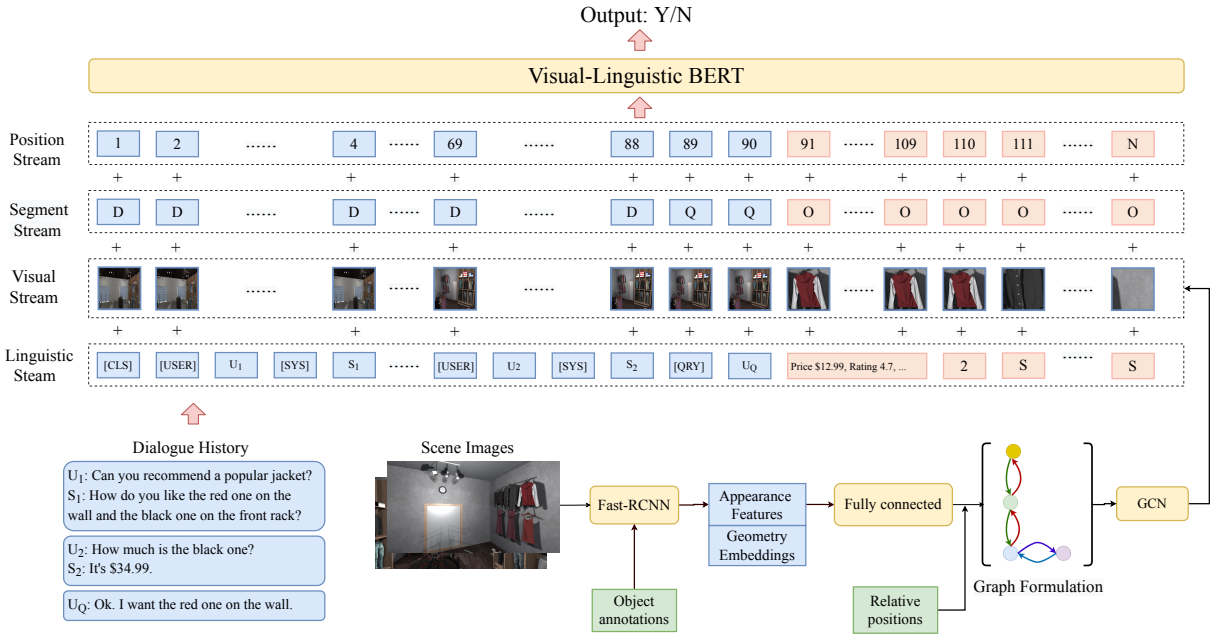


Figure 2: A visualization of our MCR model. It has four streams. The linguistic stream consists of dialogue, query, metadata text, coreference distance and an object surrounding indicator token S. In this example, the coreference distance is 2 because the candidate object is mentioned 2 turns earlier in the dialogue history. The visual stream consists of visual features of the whole scene, the candidate object and its surrounding area. The scene features and object features are repeated such that the visual stream has the same length as the linguistic stream. Image features are extracted using Fast-RCNN backbone and processed by a GCN module. The segment stream is to distinguish the dialogue, query and other tokens. The position stream indicates token positions.

implied by considering only one object at each inference step. Furthermore, feeding more objects to GCN is less memory-intensive than VL-BERT. This is because, for VL-BERT, the sequence length consists of $len(dialogue\_history) + len(metadata) + num(objects)$, while for GCN it is just $num(objects)$ and the dialogue history can be arbitrarily long (e.g., >200 tokens for SIMMC).

We next describe the three major contributions in our model: GCN Structure, Reference Distance, and Environmental Information Encoding.

3.1 GCN Structure

In a scene with multiple objects, it is natural to refer to an object using its attributes combined with spatial information relative to other objects in the scene. For instance, in the referring expression *"the black cat on the yellow sofa"*, object attributes are *"black cat"* and *"yellow sofa"* and the spatial relationship is *"on"*. Graphical approaches can effectively capture such spatial relations by creating edges between neighboring objects. We, therefore, introduce a GCN layer in our model to augment the raw visual features of an object (extracted from Fast-RCNN) with information of its neighbors by adding the spatial relationships between them.

Graph Formulation: We represent all the objects in a scene as one graph. The nodes represent the visual features of the objects and the edges represent the positional relationships between them. An example is shown in Figure 3. There are four basic edges: *top*, *bottom*, *left*, and *right*. We add an additional node that represents the features of the whole scene. It is connected to all the other nodes with a fifth edge type - *inside*, that indicates an object lies inside the scene. We expect this setup to capture the global information of the full scene into the object representations.

GCN Layer: For a node v with feature h_v^0, let its neighbors be $\mu \in \varepsilon$ whose features are h_μ^0. Each $\mu - v$ edge has a type $l \in L$. The purpose of GCN is to update h_v^0 using all h_μ^0.

We use the FiLM-GCN (Brockschmidt, 2020) model. Proposed in 2019, it is a GCN specially designed to support multiple edge types. Its equation is represented as

$$\beta_{l,v}^t, \gamma_{l,v}^t = g(h_v^t; \theta_{g,l})$$
$$h_v^{t+1} = l\Big(\sum_{u \xrightarrow{l} v \in \varepsilon} (\sigma(\gamma_{l,v}^t \odot W_l h_u^t + \beta_{l,v}^t); \theta_t)\Big)$$

For a node v, its current representation h_v^t is first

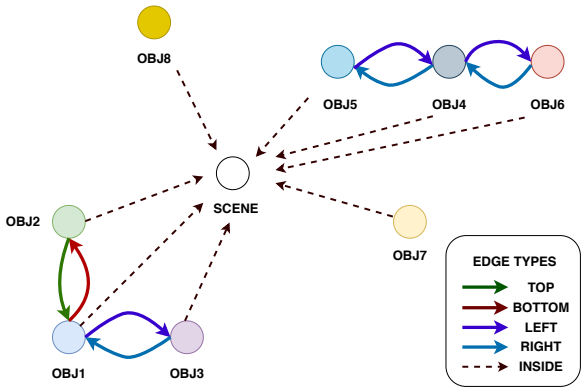


Figure 3: An example of graph formulation. It consists of five types of edges which indicate five positional relations between objects. The center *scene* node represents the features of the whole image.

Figure 4: A sampling of object's neighborhood. The center bounding box does not provide enough features for a model to recognize the ground cabinet. We supplement this input by adding features from its left, right, top and bottom directions.

passed to a function $g(\cdot)$ to compute two variables: the encoded representation $\beta^t_{l,v}$ and the element-wise weight factor $\gamma^t_{l,v}$ associated with each edge type l. The message passed from neighbors to v is represented as the element-wise product of $\gamma^t_{l,v}$ and $W_l h^t_\mu$. It is added to $\beta^t_{l,v}$ and passed to an activation function $\sigma(\cdot)$. The outputs are summed over all edges of v and finally sent to a linear function $l(\cdot)$ to become the new representation h^{t+1}_v.

3.2 Reference Distance

During conversation with a multimodal agent, users can refer to objects mentioned in an earlier turn of the dialogue. For instance, as shown in Figure 1, the user can refer to the red coat offered earlier by the system by saying *"the one you recommended before"*. To aid our model to *look back* in the dialogue history for resolving such references, we add a feature that indicates the distance from the query to the most recent system-mention of the candidate object in the dialogue history.

3.3 Environment Information Encoding

Scene images can contain visual entities that are not direct target objects, e.g., wall, table, shelves etc. As these entities are usually present in a small region of the image, their features may be attenuated during downsampling and not easily available to VL-BERT. To address this issue, we employ the following two approaches.

Object Surrounding: Sometimes the object region does not have all the information to allow for reference resolution, e.g., in Fig 4, given only central bounding box, it would be difficult to identify whether the jacket is on a table or cabinet. Therefore, we sample regions of fixed size around the object and feed them as supplementary information in the visual stream. Sampling only from the object's immediate neighbourhood is based on the intuition that people typically use items in the target object's vicinity to refer to it. The sampled region size is a tunable hyperparameter. Specifically, in this work, we consider eight surroundings regions from 8 directions as shown in Figure 4.

Image Captioning: Image captioning models generate descriptions that include the surrounding context in which an object is situated. For our task, these models generate captions that contain references to non-target objects, see Fig5. We use an off-the-shelf captioning model and generate captions for each object in the scene. We then augment our training dataset with captions as additional metadata. The captioning model that we use, is composed of an Alexnet (Krizhevsky et al., 2017) image feature encoder and a LSTM (Hochreiter and Schmidhuber, 1997) decoder. ROIs are extracted from scene images, resized and used as model inputs. We also perform a cleanup to remove redundancy from the generated captions before adding them to our training set.

4 Experiments and Results

4.1 Dataset

We evaluate our approach on Situated and Interactive Multimodal Conversations (SIMMC) 2.0 dataset(Kottur et al., 2021) released as part of DSTC10 Challenge 2021. It contains 11k task-oriented dialogues between a user and an agent, grounded in photo-realistic virtual reality (VR)

(a) (b)

Figure 5: Examples showing additional metadata extracted from captions generated for scene objects. We augment our training set with these generated attributes to help with resolving coreferences involving surrounding context.

		Precision	Recall	F1
Dev	Ours	0.74	0.83	0.78
Devtest	Ours	0.74	0.78	0.76
Test	Ours	N/A	N/A	**0.78**
	BART-based	N/A	N/A	0.76
	Huang et al. (2021)	N/A	N/A	0.73

Table 1: Our model GRAVL-BERT's performance w.r.t object-level precision, recall and F1 scores on SIMMC 2.0 evaluation sets. Our model is compared with other systems on test set. Only the F1 score is provided by the DSTC challenge officials.

scenes from fashion and furniture stores. Each example contains four elements: dialogue between user and the agent, scene images associated with each dialogue turn, object annotations and metadata for all objects within the scene, and the referring query. The dataset also provides spatial relationship between objects (left, right, top, bottom), which we use to construct graphs. The data is split into train (65%), dev (10%), dev-test (10%), and test-std (15%).

4.2 Experimental Setup

We continue to pre-train our model on in-domain dialogues from the SIMMC Dataset using masked language modeling (MLM) objective. We mask 30% dialogue tokens and train the model to predict these tokens. The pre-trained model was then fine-tuned for the MCR task by progressively building on techniques described in Section 3. We limit the dialogue context length to 3. This was done for two reasons. First, from a practical standpoint, 512 tokens is the maximum sequence length that the transformer module can consume. Secondly, it is reasonable to assume that most users will refer to objects seen recently in the context as opposed to far back in the conversation. Further, we downsample negative examples to maintain positive-negative ratio to $1:5$ for training. We use object-level binary cross entropy for loss. All models are trained on 4 Tesla V100-SXM2 GPUs.

As explained in Sec3.3, we use Alexnet-LSTM captioning model to capture each object's surrounding context by generating captions describing it. In order to fine-tune the captioning model, we mine queries from the SIMMC training set (using keyword-matching heuristics) that involve references based on surrounding objects like tables, racks, stands etc. The captioning model is then trained to generate these queries given enlarged bounding boxes enclosing the corresponding target objects as inputs. For this task, there are 1190 training examples. After training, we use the model to generate captions for all scene objects, and then perform a basic procedure to extract descriptions of surrounding context from the captions. Specifically, we create a pool of non-target objects, search the generated captions for these objects and construct phrases on them. Finally, these phrases such as "on the table" and "in the closet" are added to the dataset as additional metadata text. An example is presented in Figure 5.

For inference, we make predictions over all objects inside the scene. Objects with score ≥ 0.5 are marked as referred objects. We report the model performance with object-level precision, recall and F1 score. DSTC10 uses object-level F1 score as official metric. We report our results on devtest and test set.

4.3 Primary Results

The results are shown in Table 1. The test precision and recall are missing as the ground truth labels have not been released by the challenge officials and only F1 score is reported. Our model has the highest performance among 16 participating teams. It outperforms the second best system[1] (model based on BART (Lewis et al., 2020)) by $\sim 2.5\%$ and the third best system (model based on UNITER (Chen et al., 2020)) by $\sim 5\%$.

[1] https://github.com/KAIST-AILab/DSTC10-SIMMC

Experiments	Precision	Recall	F1
Vanilla VL-BERT	0.46	0.49	0.47
Mask-out Metadata	0.26	0.57	0.36
Mask-out Visual Feats	0.37	0.01	0.02

Table 2: Object-level precision, recall and F1 score of models trained using different kinds of inputs. Measured on validation set of SIMMC 2.0.

Experiments	Precision	Recall	F1
Vanilla VL-BERT	0.46	0.49	0.47
Dialogue Pre-trained VL-BERT	0.57	0.83	0.68
VD-BERT	0.53	0.89	0.66

Table 3: Comparison of VD-BERT and VL-BERT. Both pretrained on dialogue dataset using mask language modeling. Measured on validation set of SIMMC 2.0.

4.4 Ablation Study

Contribution of Metadata and Visual Features. To understand the impact of metadata information and visual features in the model's performance, we start with vanilla VL-BERT and mask-out i.e. zero-out either of the two features and then train and evaluate the system with all the other inputs. As seen in Table. 2, both metadata and visual features provide complementary information and contribute to the model's final performance. When trained without metadata features, the model cannot resolve coreferences based on attributes like brand and price. At the same time, visual features are essential for referencing based on visual characteristics like color and pattern. The F1 score drops close to zero when visual features are masked out because, in this dataset, most queries involving reference by metadata attributes also include visual characteristics, e.g., *the blue Nike one*.

Impact of Dialogue-Oriented Pre-training. To quantify the importance of pre-training on dialogue datasets, we train our system with original VL-BERT (pre-trained on Conceptual Captions (Sharma et al., 2018), Book Corpus (Zhu et al., 2015) and English Wikipedia datasets) and compare it with VL-BERT further pre-trained on SIMMC 2.0 dialogue dataset. As shown in Table 3, dialogue-specific pre-training provides significant performance gain on the multimodal coreference task. We also fine-tune a VD-BERT (Wang et al., 2020) model (pre-trained on VisDial (Das et al., 2019) visual dialogues dataset). VD-BERT has similar structure as VL-BERT but is trained on dialogues instead of captions. It provides similar gains as dialogue pre-trained VL-BERT reinforcing our hypothesis that dialogue pretraining is important for this task. We use VL-BERT as our base architecture because it has been pre-trained on much larger datasets compared to VD-BERT and thus can provide better generalization.

Experiments	Precision	Recall	F1
Pretrained VL-BERT	0.5734	0.8293	0.6780
+GCN	0.6432 (+0.0698)	0.8122 (−0.0175)	0.7179 (+0.0399)
+Reference Distance	0.7249 (+0.0822)	0.8238 (+0.0118)	0.7712 (+0.0538)
+Neighbour Features	0.7316 (+0.0067)	0.8248 (+0.0010)	0.7753 (+0.0041)
+Captions	0.7410 (+0.0093)	0.8306 (+0.0058)	0.7833 (+0.0080)

Table 4: Contribution of different modules. The experiments are cumulative. "+" means the current experiment is based on the above row with the indicated module added. The change of metrics corresponding to above row is shown in bracket. Performance is measured on validation set of SIMMC 2.0.

	Spatial	Non-tgt Objects	Dialogue History	Meta-data	Pure Visual
Pretrained VLBERT	5.0%	4.5%	3.7%	7.4%	4.0%
GRAVL-BERT	1.8% (−64%)	1.6% (−64%)	1.1% (−70%)	2.0% (−73%)	1.7% (−58%)

Table 5: Object-level error rate per coreference type.

Contribution of Various Model Components. To study the contribution of each component of our model architecture (GCN, reference distance, surrounding features, and captions), we perform ablation experiments. Starting from the dialogue pre-trained VL-BERT, we add our modifications incrementally and measure the gain in F1 score. The results are in Table 4. Adding GCN and reference distance provides significant gains. Adding surrounding features and captions further improves the performance by 0.41% and 0.80% respectively.

Note that, most queries refer to an object using multiple attributes. For example, *"the red Nike*

shirt to the left of the blue one in the cabinet.", which includes references by color (red), brand (Nike), relative position (left of the blue one) and absolute position (in the cabinet). Some of these information may be redundant and not jointly required to uniquely locate the target. This may explain only minor improvements in the last two rows in Table 4.

4.5 Qualitative Analysis

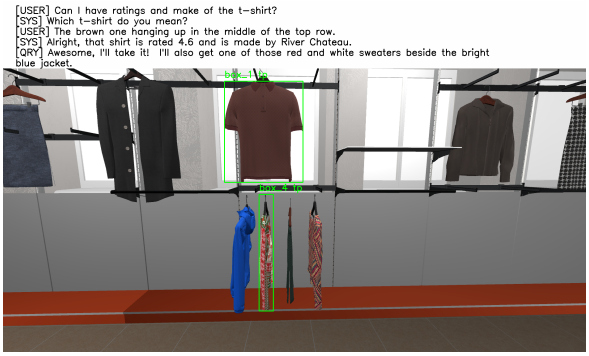
We examine our model's predictions to get a sense of its strengths and weaknesses. We provide a breakdown of the performance on different co-reference types in Table 5. For this analysis, we assigned examples to exactly one co-reference category based on heuristics. Compared with baseline, GRAVL-BERT provides significant improvement on all types of coreferences. We notice that the model is able to resolve complicated references such as in Figure 6a. This example involves pointing to multiple objects and requires visual understanding and spatial reasoning (absolute and relative). In Figure 6b, the model is able to successfully utilize long dialogue context to resolve coreferences to two objects. We also looked at several failure cases. To resolve the coreferences in these cases, a deeper scene understanding is required. For example, in Figure 6c, the model needs to infer the number of cabinets, and then locate the second one from the left. We show more examples in the Appendix.

5 Conclusion and Discussion

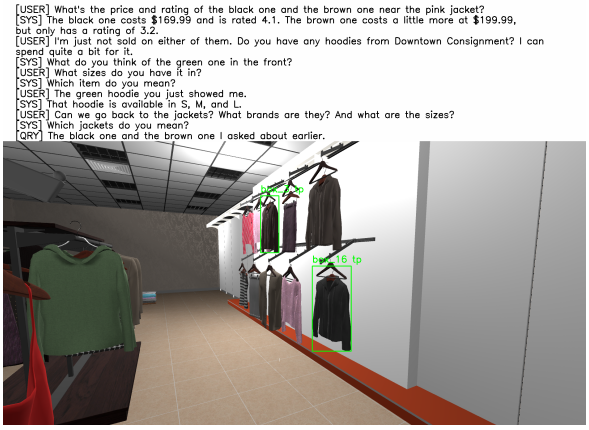
In this work, we proposed a multimodal framework GRAVL-BERT for MCR task. Our contribution lies in systematically combining relevant techniques such as utilizing external knowledge sources (metadata, generated captions), GCN, sampling object's neighborhood, and dialogue-oriented pre-training using a simple BERT-based architecture to perform MCR within dialogues grounded in scenes. We improved over the GPT-2 (Kottur et al., 2021) based baseline by 33.9% absolute and 2.5% over other concurrent work.

For future research in the topic we suggest several avenues. As mentioned in Section 4.5, to identify objects with complex references, global information is required e.g. number of cabinets in the scene, price ranking among objects. Currently, our model is unable to handle this complexity. We believe adding specific encoders to extract these global features will be helpful.

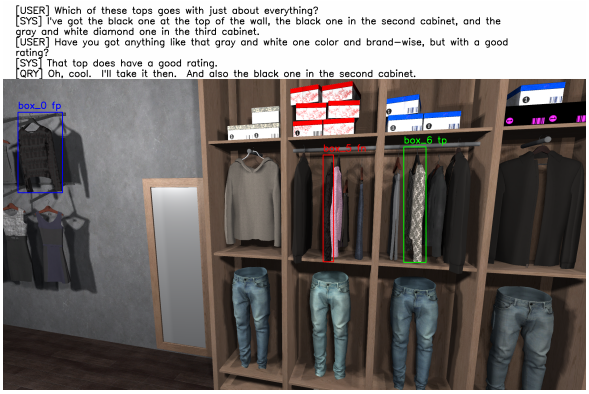
In the pre-training stage, we trained our model on dialogues using MLM. We expect that applying prompt in this stage may have a promising performance. Instead of training the model to learn to



(a) An example with correct predictions. There are two target objects (highlighted with boxes), both of which our model gets right. Coreferencing requires both visual (*"brown one"*, *"red and white sweaters"*) and spatial understanding (*"middle of the top row"*, *"besides the bright blue jacket"*).



(b) A successful example where the model is able to utilize full dialogue context for coreferencing. In this case, the model needs to obtain context from the first turn to point to the two objects being referred in query.



(c) An example of incorrect case. Our model is not able to locate "the second cabinet."

Figure 6: Model predictions on few examples from SIMMC 2.0 devtest set.

predict the randomly masked words, using a carefully designed prompt can teach model to focus on the information that is helpful to our main task.

Lastly, there are cases where the user provides very general descriptions and the information to resolve the coreference is insufficient. For example, the user refers to "the red sweater" while there are multiple red sweaters in the scene. In this situation, instead of trying to resolve the coreference, the system may attempt to disambiguate. (e.g., *"Which sweater do you mean?"*) We expect future work to distinguish these kinds of situations.

References

Marc Brockschmidt. 2020. GNN-FiLM: Graph neural networks with feature-wise linear modulation. In *ICML*, volume 119 of *PMLR*, pages 1144–1152. PMLR.

Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. 2020. Uniter: Universal image-text representation learning. In *ECCV*.

Vinay Damodaran, Sharanya Chakravarthy, Akshay Kumar, et al. 2021. Understanding the role of scene graphs in visual question answering. *CoRR*, abs/2101.05479.

Abhishek Das, Satwik Kottur, Khushi Gupta, Avi Singh, Deshraj Yadav, Stefan Lee, José M. F. Moura, Devi Parikh, and Dhruv Batra. 2019. Visual dialog. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41:1242–1256.

Chaorui Deng, Qi Wu, Qingyao Wu, et al. 2018. Visual grounding via accumulated attention. In *CVPR*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *ACL*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Henghui Ding, Chang Liu, Suchen Wang, and Xudong Jiang. 2021. Vision-language transformer and query generation for referring segmentation. In *ICCV*.

Ross Girshick. 2015. Fast r-cnn. In *International Conference on Computer Vision (ICCV)*.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780.

L Huang, W Wang, Y Xia, and J Chen. 2019a. Adaptively aligned image captioning via adaptive attention time. In *NIPS*, pages 8942–8951.

Pingping Huang, Jianhui Huang, Yuqing Guo, Min Qiao, and Yong Zhu. 2019b. Multi-grained attention with object-level grounding for visual question answering. In *ACL*, pages 3595–3600, Florence, Italy. Association for Computational Linguistics.

Yichen Huang, Yuchen Wang, and Yik-Cheung Tam. 2021. Uniter-based situated coreference resolution with rich multimodal input. *arXiv preprint arXiv:2112.03521*.

Kanishk Jain and Vineet Gandhi. 2021. Comprehensive multi-modal interactions for referring image segmentation. *arXiv preprint arXiv:abs/2104.10412*.

Satwik Kottur, Seungwhan Moon, Alborz Geramifard, and Babak Damavandi. 2021. Simmc 2.0: A task-oriented dialog dataset for immersive multimodal conversations. *arXiv preprint arXiv:2104.08667*.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. 2017. Imagenet classification with deep convolutional neural networks. *ACM*, 60(6):84–90.

Hung Le, Doyen Sahoo, Nancy F. Chen, and S. Hoi. 2019. Multimodal transformer networks for end-to-end video-grounded dialogue systems. In *ACL*.

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 *acl*, pages 7871–7880.

Gen Li, Nan Duan, Yuejian Fang, et al. 2020a. Unicoder-vl: A universal encoder for vision and language by cross-modal pre-training. *AAAI*, 34(07):11336–11344.

Xiujun Li, Xi Yin, Chunyuan Li, et al. 2020b. Oscar: Object-semantics aligned pre-training for vision-language tasks. In *ECCV*.

J. Lu, V. Goswami, M. Rohrbach, et al. 2020. 12-in-1: Multi-task vision and language representation learning. In *CVPR*, pages 10434–10443, Los Alamitos, CA, USA. IEEE Computer Society.

Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *NIPS*.

Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. 2018. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2556–2565, Melbourne, Australia. Association for Computational Linguistics.

Weijie Su, Xizhou Zhu, Yue Cao, et al. 2020. Vl-bert: Pre-training of generic visual-linguistic representations. In *ICLR*.

Hao Tan and Mohit Bansal. 2019. LXMERT: learning cross-modality encoder representations from transformers. In *EMNLP*.

Yue Wang, Shafiq Joty, Michael Lyu, et al. 2020. VD-BERT: A Unified Vision and Dialog Transformer with BERT. In *EMNLP*, pages 3325–3338, Online. Association for Computational Linguistics.

Jianwei Yang, Jiasen Lu, Stefan Lee, et al. 2018. Graph r-cnn for scene graph generation. In *ECCV*, pages 690–706, Cham. Springer International Publishing.

Xu Yang, Kaihua Tang, Hanwang Zhang, and Jianfei Cai. 2019. Auto-encoding scene graphs for image captioning. In *CVPR*.

Ting Yao, Yingwei Pan, Yehao Li, and Tao Mei. 2018. Exploring visual relationship for image captioning. In *ECCV*.

Licheng Yu, Patrick Poirson, Shan Yang, et al. 2016. Modeling context in referring expressions. In *Computer Vision – ECCV 2016*, pages 69–85, Cham. Springer International Publishing.

Pengchuan Zhang, Xiujun Li, Xiaowei Hu, et al. 2021. Vinvl: Revisiting visual representations in vision-language models. In *CVPR*.

Wenbo Zheng, Lan Yan, Chao Gou, and Fei-Yue Wang. 2020a. Webly supervised knowledge embedding model for visual reasoning. In *CVPR*, pages 12442–12451.

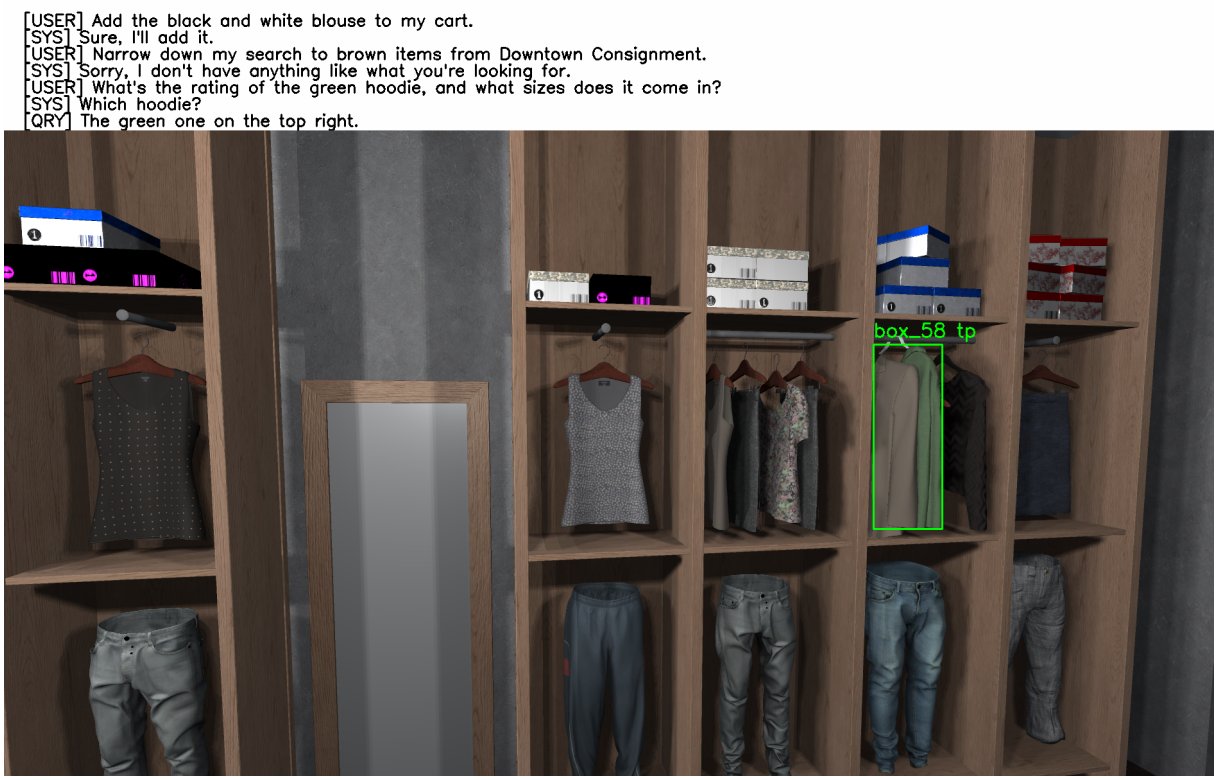
Wenbo Zheng, Lan Yan, et al. 2020b. Webly supervised knowledge embedding model for visual reasoning. In *CVPR*.

Luowei Zhou, Hamid Palangi, et al. 2020. Unified vision-language pre-training for image captioning and vqa. *AAAI*, 34(07):13041–13049.

Yukun Zhu, Ryan Kiros, Richard S. Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 19–27.

A Appendix

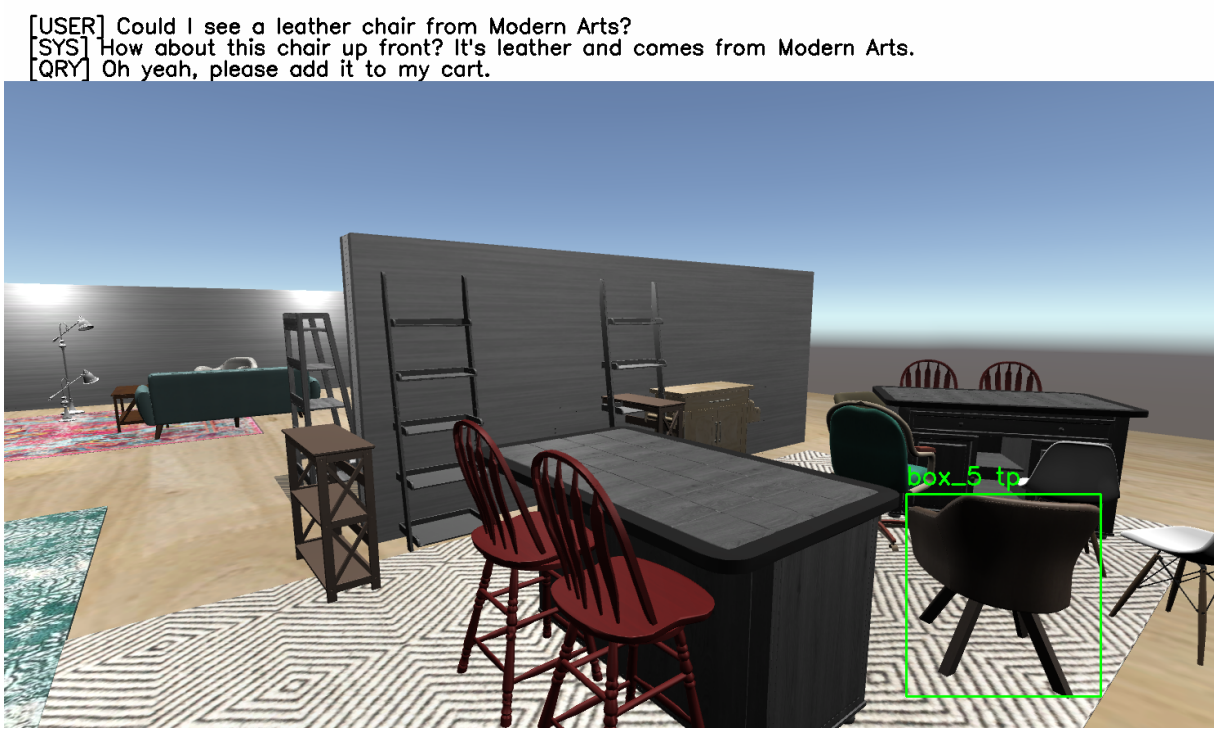
In this appendix, we showcase more examples of our model's predictions. The figures below show 5 examples, out of which the first 4 are correctly predicted by our model. In Figure 7a, the target is referred by its color (*green*) and location (*top right*). In Figure 7b and 8a, the target is referred by non-visible attribute, object brand (*Ocean Wears* and *Modern Arts*). In Figure 8b the target is referred using previous dialogue context (*..that grey sofa you pointed out*). The prediction in Figure 9a is wrong. In this case, the targets are referred as the ones *with good ratings*. Our model, currently, does not have a mechanism to compare the ratings of all objects and select the top-rated ones. This can be improved in the future.



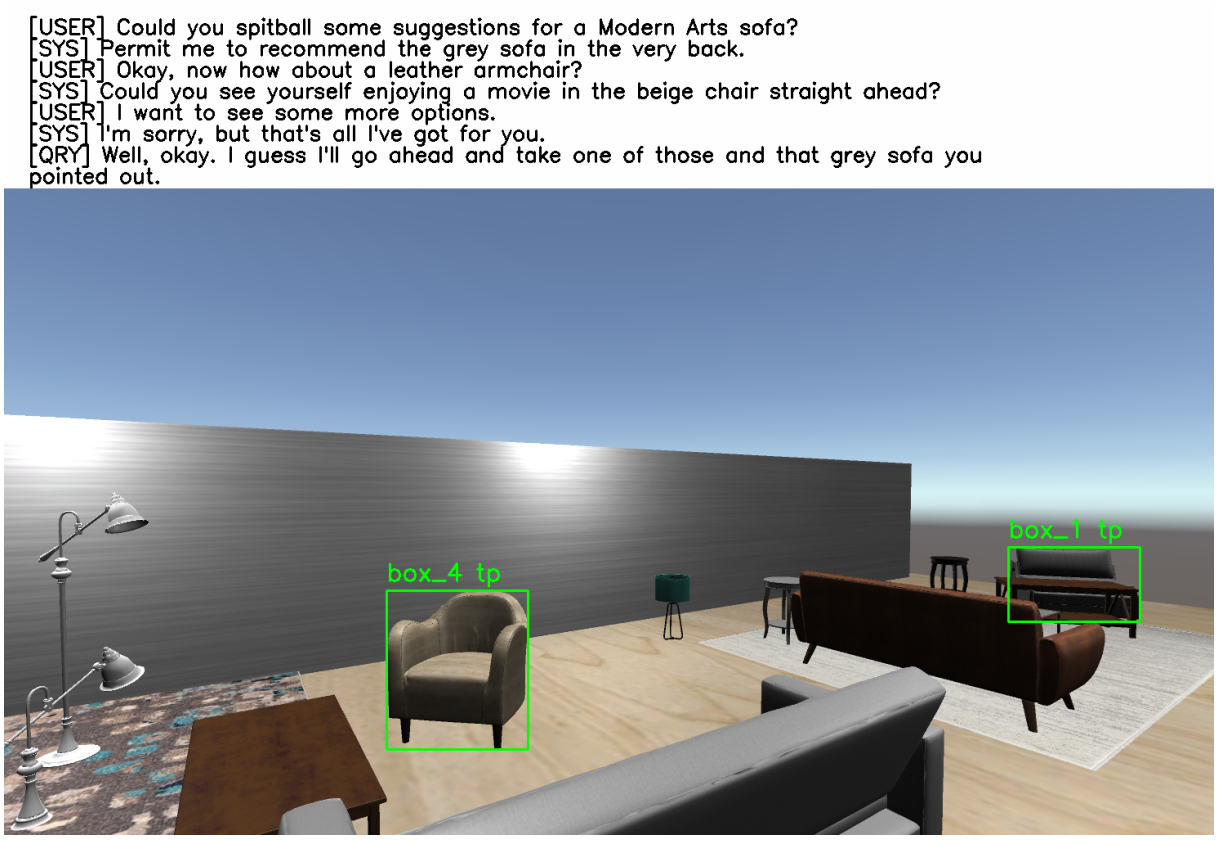
(a) An example of a correct case. The object in query is referred by its color and position.

(b) An example of a correct case. The object in query is referred by its brand.

Figure 7: More prediction results on devtest set.



(a) An example of a correct case. The object in query is referred by its brand.



(b) An example of a correct case. The query refers to an object mentioned in early turns.

Figure 8: More prediction results on devtest set.

(a) An example of an incorrect case. Our model is unable to figure out which ones are "affordable" and "good" without comparing the price and rating of all objects.

Figure 9: More prediction results on devtest set.