**Abstract**

We propose **VD-GR** – a novel visual dialog model that combines pre-trained language models (LMs) with graph neural networks (GNNs). Prior works mainly focused on one class of models at the expense of the other, thus missing out on the opportunity of combining their respective benefits. At the core of **VD-GR** is a novel integration mechanism that alternates between spatial-temporal multi-modal GNNs and BERT layers, and that covers three distinct contributions: First, we use multi-modal GNNs to process the features of each modality (image, question, and dialog history) and exploit their local structures before performing BERT global attention. Second, we propose hub-nodes that link to all other nodes within one modality graph, allowing the model to propagate information from one GNN (modality) to the other in a cascaded manner. Third, we augment the BERT hidden states with fine-grained multi-modal GNN features before passing them to the next **VD-GR** layer. Evaluations on VisDial v1.0, VisDial v0.9, VisDialConv, and VisPro show that **VD-GR** achieves new state-of-the-art results across all four datasets.

**1. Introduction**

Visual dialog is a multi-modal task to assess how well an artificial agent can hold a conversation with a human on a visual content using natural language [8]. Visual dialog differs from other tasks, such as visual [3] or video question answering [49], in that it requires the agent to answer a series of temporally dependent questions. That is, the agent not only has to reason about the visual input but also has to leverage the context of previous rounds to be able to answer the current question correctly. Although other datasets have been proposed for this task [22, 39], VisDial [8] has established itself as the de-facto standard because of its challenging, open-ended, and real-world nature.

Early visual dialog models on this dataset were based on recurrent networks [15] within deep neural architectures ranging from vanilla LSTMs [14] over memory nets [48] to hierarchical structures [40]. More recently, graph neural networks (GNNs) have been proposed and have been shown to produce more fine-grained features based on the local structures of each modality [6, 12, 16, 17, 25]. Other works have focused on attention models and pre-trained language models (LM) [1, 8, 33, 38], fine-tuned for the visual dialog task [27, 32, 46, 55]. However, both methodological approaches have so far remained separate despite their com-
plementary strengths and weaknesses: While GNNs are effective at exploiting local structure, they struggle to capture the global inter-modal context, especially in a rich multi-modal task like visual dialog. In contrast, transformer-based models [44] (e.g., BERT [9]) excel at learning the global context using self and global attention but often fail to exploit local intra-modal structures within each modality and suffer from the lack of inductive bias [50].

We posit that it is essential to exploit both local intra-modal structures and the global inter-modal context: Each modality is composed of smaller entities whose relationships have to be separately captured and understood by the model (i.e., the objects in the image, the words that constitute the question, and the rounds that form the history). At the same time, the inter-dependency of the modalities and the global context are equally important for a model to be able to answer the current question efficiently. Implementing this idea, we propose VD-GR – the first visual dialog model to combine transformer-based LMs and GNNs. Each of our proposed layers alternates between GNNs that use hub-nodes to propagate information from one modality graph to another to alleviate the lack of inter-modal context, as well as BERT layers to learn the global intra-model context. The contributions of our work are threefold:

- A novel integration method of GNNs into transformer-based models that alternates between multi-modal graph aggregation and BERT layers. The GNNs exploit the local structure of each modality to augment the BERT hidden states with their fine-grained features before passing them to the next VD-GR layer in an attempt to mitigate their lack of inductive bias.

- A novel feature propagation technique for multi-modal GNNs that relies on hub-nodes that link to all other nodes of the other modality in a cascaded manner, thus alleviating the lack of inter-modal context within the multi-modal graphs.

- Our model achieves new state-of-the-art results on VisDial v1.0, VisDial v0.9, VisDialConv, and VisPro datasets, thereby outperforming strong baselines on all four datasets.

2. Related Work

2.1. GNNs for Visual Dialog

GNN-EM [59] was one of the earliest models to deploy GNNs for visual dialog. It relied on building graphically structured neural networks to approximate the learning and inference processes of graphical models [10, 28, 29, 42]. FGA [38] developed a general attention mechanism that borrowed from the core idea of GNN message passing, and that was able to combine representations of any number of utilities. CAG [12] proposed representing the image as a fully-connected graph of objects whose adjacency matrices could be dynamically updated using a question-based attention mechanism. GOG [6] proposed representing each modality as a graph before processing them by a light-weight fusion module [33] to rank the candidate answers. Similarly to CAG [12], DualVD [18] proposed representing the image as a graph consisting of connected objects. They argued that the visual view helped to capture the appearance-level information, including objects and their relationships, while the semantic view enabled the agent to understand high-level visual semantics from the whole image to the local regions [18].

Our model differs from the aforementioned works in two major aspects: (1) Instead of concatenating the GNN node features, we propose using hub-nodes that link to all remaining nodes within a given modality. We argue that this improves the feature fusion on a local scale given that GNNs can freely learn how to integrate information extracted from previous modalities. (2) We marry GNNs with pre-trained LMs (BERT) in a novel way to benefit from the advantages of both worlds. Contrarily to [51, 53] where simple integration approaches for plain uni-modal tasks were introduced, we propose a sophisticated integration method for the complex and rich multi-modal visual dialog task: We first exploit the local multi-modal structures through GNNs before propagating inter-modal information in a cascaded manner via hub-nodes to finally enhance the hidden states of each BERT layer. To the best of our knowledge, we are the first to propose this for the visual dialog task.

2.2. Language Models for Visual Dialog

ViLBERT [30] and XLMERT [43] were among the first attempts to leverage pre-trained LMs for vision-language tasks (e.g., VQA [3], VCR [58], and image retrieval [54]). Concretely, they used a transformer encoder as a backbone and deployed a two-stream architecture to separately encode text and visual input. VisDiab-BERT [32] built on top of ViLBERT and specifically adapted it to the visual dialog task, achieving new state-of-the-art performance on VisDial, and thus becoming the standard baseline for this dataset. More recently, UTC [5] was introduced as an improvement of VisDial-BERT and used two inter-task contrastive losses to improve training. Another class of models used a single stream to encode the multi-modal input such as B2T2 [2], VisualBERT [26], VL-BERT [41], and UNITER [7]. VD-BERT [46] adapted a single stream VQA model [60] to the visual dialog task and managed to achieve good performance without relying on external datasets.

Contrarily to the aforementioned works, our model relies on the power of cascaded GNNs to exploit local structures within each modality and, thus produce more fine-grained representations for subsequent BERT layers. To the best of our knowledge, this combination of GNNs and pre-trained LMs has not been explored before for visual dialog.
3. Method

As shown in Figure 2, VD-GR consists of four main components: (a) a graph construction and pruning module, (b) a backbone operating on alternating cascaded multi-modal graphs and BERT layers, (c) multiple loss heads (next sentence prediction \(L_{\text{ns}}\)), masked language modelling \(L_{\text{lm}}\), masked region modelling \(L_{\text{rm}}\), and multi-modal graph edge masking \(L_{\text{gem}}\), \(L_{\text{gem}}^2\), \(L_{\text{gem}}^3\), \(L_{\text{gem}}^4\), and (d) a prediction module to rank a set of candidate answers.

3.1. Problem Formulation

Given a question \(Q_t\) grounded on an image \(I\) at \(t\)-th turn, as well as its dialog history \(H_t = \{C, (Q_1, A_1), ..., (Q_{t-1}, A_{t-1})\}\) (where \(C\) denotes the image caption), the model is tasked to predict its answer \(A_t\) by ranking a list of \(N = 100\) answer candidates \(\{A_1, A_2, ..., A_{100}\}\).

3.2. Graph Construction and Pruning

**Image Modality.** The image graph constructor \(G_I(\cdot)\) treats each object in the image as a node and relies on spatial relationships to construct the graph topology, i.e. the adjacency matrix, where each edge represents a relationship between two objects in the image. The objects \(I = \{v_1, ..., v_{N_i}\}\) are obtained using Faster R-CNN [37] pre-trained on Visual Genome [23] where each object feature \(v_i\) is a 2048-dimensional vector and \(N_i = 36\). Similar to [52], we distinguish between 11 relations based on the \((x_1, y_1, x_2, y_2)\) object coordinates. Specifically, the overlapping region and spatial coordinates of two regions are used to judge whether an edge exists between them or not.

**Question Modality.** The question graph constructor \(G_Q(\cdot)\) treats each word in the question as a node and relies on dependency relations to construct the graph topology, i.e. the adjacency matrix, where each edge represents a relationship between two words in the question. To extract these relationships, we use the neural dependency parser of Stanza [36] that yields 47 relations.

**History Modality.** The history graph constructor \(G_H(\cdot)\) treats each dialog round as a node and relies on coreference relations to construct the graph topology, i.e. the adjacency matrix, where each edge represents a relationship between two rounds in the history.

As can be seen in Figure 2, the topology of all multi-modal graphs is computed once during a pre-processing stage and is always kept constant over time. We refer to the supplementary material for additional details.

3.3. Proposed Layer

3.3.1 Transformer Features

Inline with previous works, we use VisDial-BERT [32] layers within our novel GNN-enhanced approach. For a given
image-question pair at round \( t \), we first concatenate the caption \( C \), the previous dialog rounds \( H_t \), the current question \( Q_t \), and a candidate answer \( \hat{A}_t \) to form the textual input

\[
T = \{ [CLS] C [SEP] Q_t [SEP] A_1, \ldots, Q_t [SEP] \hat{A}_t \},
\]

where \([CLS]\) and \([SEP]\) are the special classification and separation tokens, respectively. As in [5, 32, 46, 55], we use a special learnable token \([IMG]\), and initialise it using mean pooling of the object sequence. The hidden features of the \([IMG]\) token are used in conjunction with those of \([CLS]\) by means of element-wise multiplication to produce the final input features of the NSP head.

Finally, the two sequences are used as input for the language and vision streams of the VisDial-BERT layer to obtain the hidden states \( T^{(l)}_{h} \) and \( T^{(l)}_{h} \), where \( l \) is the \( \mathcal{VD-GR} \) layer index.

### 3.3.2 Spatial-Temporal GNN Features

**Node Features.** As illustrated in Figure 2, the multi-modal GNNs of the \( l \)-th \( \mathcal{VD-GR} \) layer get their node features from the hidden states of the previous VisDial-BERT layer (or from the embedding layers in the first step). It is worth noting that the special tokens \([IMG]\) and \([CLS]\) are not included in the graph features. The nodes of the image and question graphs are gathered from the image and question token embeddings of the \((l-1)\)-th layer, i.e. \( \{I^{(l-1)}_{h.i} \}_{i=1}^{N_i} \) and \( \{T^{(l-1)}_{h.i} \}_{t=s_q}^{s_q+N_q} \), where \( s_q \) and \( N_q \) denote the index of the first question token and the length of the question, respectively. We use the special \([SEP]\) tokens to represent each dialog round in the history and gather their hidden states from \( T^{(l-1)}_{h} \) to get the node features of the history graph. The node features of the question and history graphs have to be extracted carefully since the textual input, i.e. where the question starts and ends and where the \([SEP]\) tokens are located, varies within \( T^{(l-1)}_{h} \) for each dialog round.

**Hub-node Features.** To make each modality aware of the other in a cascaded manner, we introduce hub-nodes as illustrated in Figure 2. As a result, the history becomes aware of the image, the question becomes aware of the history, and finally the image becomes aware of the question. The hub-node within each graph links to all other nodes using a special edge feature to propagate information on a local scale from one modality to the other before applying self- and cross-attention. To obtain the hub-node features, we train attention-based graph embeddings for each modality, i.e. \( f_{\theta_{o.c}}(\cdot) \), \( f_{\theta_{o.q}}(\cdot) \), and \( f_{\theta_{q}}(\cdot) \), which take the node features of the corresponding graph and output a single vector representing it. For example, the image hub-node \( I_G \) is computed from the image graph node features \( I_G \) of the \( l \)-th layer as follows:

\[
I = f_{\theta_{I}}(I_G) = \sum_{i} \alpha_{i} v_{i} \text{ for } v_{i} \in I_G, \tag{2}
\]

\[
\alpha = \{ \alpha \}_{i} = \text{MLP}(I_G), \tag{3}
\]

where MLP is a multi-layer perceptron that maps from the node features’ vector space to \( \mathbb{R} \) and \( \theta_{I} \) is the set of learnable parameters of the embedding. The question hub-node \( Q \) and history hub-node \( H \) are obtained in the same manner.

### 3.3.3 Graph Aggregation

Our multi-modal graphs are a variant of graph attention networks [45], although other types of GNNs can be used within each \( \mathcal{VD-GR} \) layer. Each multi-modal graph uses \( K \) layers (purple boxes in Figure 2) to propagate information between the nodes. For example, the \( k \)-th image graph layer receives the node features from the previous graph layer and updates them as follows:

\[
I^{(k)}_G = \{ v^{(k)}_1, \ldots, v^{(k)}_{N_i}, v^{(k)}_{N_i+1} \} = \text{GNN}_2(I^{(k-1)}_G), \tag{4}
\]

\[
v^{(k)}_t = \text{GeLU} \left( \sum_{h=1}^{H} \| v^{(k,h)}_t + v^{(k-1)}_t \| \right), \tag{5}
\]

\[
\tilde{v}^{(k,h)}_t = f(\sum_{v_i \in N_{vt}} \alpha_{s \rightarrow t} m^{(k,h)}_{s \rightarrow t}) \text{ for all } 1 \leq h \leq H, \tag{6}
\]

where \( \| \cdot \| \) and \( H \) are the concatenation operation and the number of GNN attention heads, respectively. \( N_{vt} \) represents the neighbourhood of node \( v_t \), \( \alpha_{s \rightarrow t} \) denotes the attention weight that scales the message \( m^{(k,h)}_{s \rightarrow t} \) between a source node \( v_s \) and a target node \( v_t \), and \( f \) is a linear layer. The messages \( m^{(k,h)}_{s \rightarrow t} \) between the nodes are computed following:

\[
m^{(k,h)}_{s \rightarrow t} = g_h(v^{(k-1)}_s, e_{s \rightarrow t}), \tag{7}
\]

where \( g_h \) is a linear layer and \( e_{s \rightarrow t} \) is the edge feature between the nodes \( v_s \) and \( v_t \). We omitted the \( l \) index of the \( \mathcal{VD-GR} \) layer in the previous equations for brevity. The \( k \)-th question and history graph layers update their node features \( Q^{(k)}_G \) and \( H^{(k)}_G \) in the same manner.

### 3.3.4 Hidden States Enhancement

The outputs of the multi-modal graphs of the last \( K \)-th layer, i.e. \( I^{(l,K)}_G, Q^{(l,K)}_G \) and \( H^{(l,K)}_G \), are used to enhance the hidden states of the following BERT layer. First, the GNN features are scattered back to their corresponding places within the VisDial-BERT hidden states. Then, we apply a fusion operation inspired by the idea of residual connections [13] as illustrated in Figure 2. Our experiments show
that this step is crucial and leads to significant improvements in performance. These operations can be summarised as follows:

\[
I_h^1 = I_h^1 \odot \left( I_G^{(l,K)} \cdot \text{Idx}_v \right), \quad I_h^2 = \lambda I_h + (1 - \lambda) I_h^1, \quad (8)
\]

\[
T_h^1 = \left( T_h^1 \odot \left( Q_G^{(l,K)} \cdot \text{Idx}_v \right) \right) \odot \left( H_G^{(l,K)} \cdot \text{Idx}_h \right), \quad (9)
\]

\[
T_h^1 = \lambda T_h^1 + (1 - \lambda) T_h^1 \quad \text{for} \quad \lambda \in [0, 1], \quad (10)
\]

where \( \odot \) denote the scatter operation and \( \text{Idx}_v \) the indices of the graph nodes features with respect to the BERT hidden states. The final enhanced features \( I_h^1 \) and \( T_h^1 \) are passed to the next BERT layer.

### 3.4. Loss Heads

We complement the traditional losses used for the visual dialog task (masked language modelling \( \mathcal{L}_{\text{mlm}} \), masked region modelling \( \mathcal{L}_{\text{mmr}} \), and next sentence prediction \( \mathcal{L}_{\text{nsp}} \)) with graph edge masking \( \mathcal{L}_{\text{gem}} \) to improve learning of the local structure of each modality, and thus to enhance the feature representation of our multi-modal graphs.

**Masked Language and Region Modelling.** Similar to masked language modelling introduced in [9], we randomly masked 10% of the text tokens and image objects with the special token `[MASK]` and the model had to recover them based on the surrounding tokens and cross-modal clues:

\[
\mathcal{L}_{\text{mlm}} = -\mathbb{E}_{(w,I) \sim S_{tr}} \left[ \log P(w_m|w_m, h_{[CLS]}) \right], \quad (11)
\]

\[
\mathcal{L}_{\text{mmr}} = -\mathbb{E}_{(w,I) \sim S_{tr}} \left[ \log P(v_m|v_m, h_{[CLS]}) \right], \quad (12)
\]

where \( w_m \), \( v_m \), and \( S_{tr} \) denote the masked words, the masked regions, and the training set, respectively.

**Next Sentence Prediction.** The aim of next sentence prediction (NSP) is to identify whether the appended answer \( \hat{a}_c \) is correct or not, i.e.

\[
\mathcal{L}_{\text{nsp}} = -\mathbb{E}_{(w,I) \sim S_{tr}} \left[ \log P(y|y) \right], \quad (13)
\]

where \( y \in \{0, 1\} \) is a ground-truth binary label, and \( \mathbb{N}(_,_) \) is the binary answer prediction head operating on the element-wise product of the `[IMG]` and `[CLS]` token representations.

**Graph Edge Masking.** We introduce the multi-modal graph edge masking to alleviate the potential problem of having well-trained BERT layers but not sufficiently trained GNNs. Given that, by design, our GNNs receive their features from the previous BERT layer, they inherit some masked node representations. In order to make the edge prediction task stable, we only masked 15% of the edges connecting two unmasked node features as illustrated in Figure 3. The representations of these nodes were then used to predict the masked edges. For the example of the image graph, this results in the following loss:

\[
\mathcal{L}_{\text{gem}} = -\mathbb{E}_{(w,I) \sim S_{tr}} \left[ \log P(y^{(i,j)}|v_i, v_j) \right], \quad (14)
\]

where \( y^{(i,j)} \) is the ground-truth edge type between the nodes \( v_i \) and \( v_j \). The question and history edge graph masking losses \( \mathcal{L}_{\text{gem}} \) and \( \mathcal{L}_{\text{gem}} \) are obtained in a similar manner.

**Total Loss.** We adopt a two-stage approach to train our model. First, we train it on a warm-up task of masked token and graph edge prediction, i.e. using the total loss \( \mathcal{L}_{\text{warm}} \):

<table>
<thead>
<tr>
<th>Method</th>
<th>VisPro</th>
<th>VisDialConv</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>NDCG↑</td>
<td>MRR↑</td>
</tr>
<tr>
<td>MCA-I [1]</td>
<td>59.80</td>
<td>57.88</td>
</tr>
<tr>
<td>MCA-I-HConQ [1]</td>
<td>61.08</td>
<td>61.79</td>
</tr>
<tr>
<td>MCA-I-HGuidedQ [1]</td>
<td>61.35</td>
<td>60.13</td>
</tr>
<tr>
<td>MCA-I-VGH [1]</td>
<td>61.68</td>
<td>59.33</td>
</tr>
<tr>
<td>MCA-I-H [1]</td>
<td>61.72</td>
<td>59.62</td>
</tr>
<tr>
<td>Student [19]</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>

\[ \text{VD-GR} \]

67.09 | 66.82 | 54.47 | 81.71 | 91.44 | 3.54 | 60.35 | 69.89 | 57.21 | 85.97 | 92.68 | 3.15

Table 1. Performance comparison on VisPro. The best and second-best results are in **bold** and *underlined*, respectively. ↑ indicates higher is better and ↓ indicates lower is better.

![Figure 3](image-url)
the role of dialog history in answering the current question

VisPro

100

Each image comes from dense annotations for the candidate answers. Further-

tion data and part of the training data of VisDial v1.0 pro-

sion consists of about 38k images for training, validation, and testing, respectively. Each image comes with a caption and 10 question-answer pairs; each question turn is associated with 100 candidate answers. The validation data and part of the training data of VisDial v1.0 provide dense annotations for the candidate answers. Furthermore, we evaluated our model on two additional datasets, i.e., VisPro [56] and VisDialConv [1].

4.2. Quantitative Results

VisDialConv & VisPro. First, we evaluated our model on VisPro \(^1\) and VisDialConv which were introduced to verify the role of dialog history in answering the current question \(Q_c\). We compared \(\text{VD-GR}\) to the baselines introduced in [1] as well as the most recent Student model [19]. As can be seen from Table 1, \(\text{VD-GR}\) significantly outperformed all MCA variants across all metrics on both datasets. Specifically, it increased the performance of the baselines by over 5 absolute points on NDCG and MRR on VisPro. On VisDialConv, \(\text{VD-GR}\) increased the top performance by over 4 absolute points on the same metrics. Table 1 also shows that our model managed to surpass the Student model by over 1

\[
\mathcal{L}_{\text{warm}} = \alpha_1 (\mathcal{L}_{\text{mlm}} + \mathcal{L}_{\text{mem}}) + \alpha_2 \mathcal{L}_{\text{GEM}},
\]

\[
\mathcal{L}_{\text{GEM}} = \mathcal{L}_{\text{GEM}}^L + \mathcal{L}_{\text{GEM}}^Q + \mathcal{L}_{\text{GEM}}^U.
\]

4.1. Datasets

We evaluated \(\text{VD-GR}\) on the challenging VisDial v0.9 and VisDial v1.0 datasets. VisDial v0.9 has circa 83k training and 40k validation dialogs. The more recent v1.0 version consists of about 123k, 2k, and 8k images for training, validation, and testing, respectively. Each image comes with a caption and 10 question-answer pairs; each question turn is associated with 100 candidate answers. The validation data and part of the training data of VisDial v1.0 provide dense annotations for the candidate answers. Furthermore, we evaluated our model on two additional datasets, i.e., VisPro [56] and VisDialConv [1].

Table 2. Performance comparison on the \(\text{val}\) split of VisDial v1.0 dataset. NDCG is not supported in this version of the dataset.

<table>
<thead>
<tr>
<th>Method</th>
<th>NDCG↑</th>
<th>MRR↑</th>
<th>R@1↑</th>
<th>R@5↑</th>
<th>R@10↑</th>
<th>Mean↓</th>
</tr>
</thead>
<tbody>
<tr>
<td>MN [8]</td>
<td>59.65</td>
<td>45.55</td>
<td>76.22</td>
<td>85.37</td>
<td>5.46</td>
<td></td>
</tr>
<tr>
<td>CoAtt [57]</td>
<td>63.98</td>
<td>50.29</td>
<td>80.71</td>
<td>88.81</td>
<td>4.47</td>
<td></td>
</tr>
<tr>
<td>HCIAE [31]</td>
<td>62.22</td>
<td>48.48</td>
<td>78.75</td>
<td>87.59</td>
<td>4.81</td>
<td></td>
</tr>
<tr>
<td>CorefMN [21]</td>
<td>64.10</td>
<td>50.92</td>
<td>80.18</td>
<td>88.81</td>
<td>4.45</td>
<td></td>
</tr>
<tr>
<td>RvA [34]</td>
<td>66.34</td>
<td>52.71</td>
<td>82.97</td>
<td>90.73</td>
<td>3.93</td>
<td></td>
</tr>
<tr>
<td>Student [19]</td>
<td>60.03</td>
<td>50.40</td>
<td>70.74</td>
<td>77.15</td>
<td>12.13</td>
<td></td>
</tr>
<tr>
<td>DvAN [11]</td>
<td>66.67</td>
<td>53.62</td>
<td>82.85</td>
<td>90.72</td>
<td>3.93</td>
<td></td>
</tr>
<tr>
<td>VD-BERT [46]</td>
<td>70.04</td>
<td>57.79</td>
<td>85.34</td>
<td>92.68</td>
<td>4.04</td>
<td></td>
</tr>
<tr>
<td>VisDial-BERT [32]</td>
<td>71.09</td>
<td>59.41</td>
<td>87.92</td>
<td>94.50</td>
<td>2.87</td>
<td></td>
</tr>
</tbody>
</table>

\(\text{VD-GR}\)

4.50 62.10 90.49 96.37 2.45

Table 3. Performance comparison on the \(\text{val}\) split of VisDial v1.0 dataset.

NDCG absolute point although it was trained on circa 13M additional images.

VisDial v0.9. Second, we compared \(\text{VD-GR}\) with the state of the art on the \(\text{val}\) split of VisDial v0.9. As can be seen from Table 2, our model significantly outperformed all previous models and achieved new state-of-the-art results across all metrics. Specifically, it outperformed pre-training methods such as VisDial-BERT and VD-BERT by a large margin. Even more importantly, it managed to also surpass more recent models such as Student. Specifically, \(\text{VD-GR}\) improved the MRR and R@1 scores by over 2.5 absolute points compared to the second best model.

VisDial v1.0 \(\text{val}\). We then compared \(\text{VD-GR}\) with the state of the art on the \(\text{val}\) split of VisDial v1.0. As can be seen from Table 3, our model outperformed all previous models across all five sparse metrics. Specifically, it outperformed pre-training methods such as VisDial-BERT and VD-BERT by a significant margin. Even more importantly, it managed to also surpass more recent models such as UniMM-UL, VD-PCR, UTC, and Student. \(\text{VD-GR}\) improved MRR, R@1, and R@5 by over 2 absolute points compared to the second best VD-PCR model. Furthermore, we compared their performance on individual dialog rounds using the sparse metrics (MRR, R@1, R@5, and R@10). As can be seen from Figure 4, \(\text{VD-GR}\) managed to outperform VD-PCR on almost all rounds of the dataset.

VisDial v1.0 \(\text{test-std}\). Finally, we compared our model with state-of-the-art published baselines on the \(\text{test-std}\) split of the VisDial v1.0 dataset.

• State-of-the-art Results on Sparse Metrics: As can be seen from the first section of Table 4, \(\text{VD-GR}\) lifted the state-of-the-art R@5, R@10, and Mean scores from 85.38, 93.53, and 3.21 achieved by VD-PCR to 85.58, 93.85, and 3.20, respectively. On the remaining metrics, our models performed on par with the state of the art. Specifically, it reached respective scores of 68.65 and 55.33 on MRR and R@1, only third to UTC and VD-PCR.

1Same subset as in [1].
To this end, we increased the last section of Table 4, our ensemble model reached an Net [4] ranking optimisation method. As can be seen in varying number for GNN layers $K$ models to further improve the NDCG score. One set of four other baselines, and Mean). This well-known phenomenon of the dataset performance on the sparse metrics (MRR, R@1, R@5, R@10, fine-tuning on dense annotations, we decreased the performance in the single-model setting. However, by sharply increased from 57.00 56.95 56.90 86.14 86.12 86.10 93.73 93.71 93.69 performance. As illustrated in the second section of Table 4, the NDCG score released by [32] in order to improve the NDCG score. As shown in Figure 5, the performance of our model: If $K$ is too small, then the expressive power of the GNNs will be hampered. Contrarily, if $K$ is too large, the GNNs will suffer from over-smoothing [24]. To this end, we increased $K$ incrementally from one to four.

- **Fine-tuning on Dense Annotations:** As in previous works, we fine-tuned our model on the dense annotations released by [32] in order to improve the NDCG score. As illustrated in the second section of Table 4, the NDCG score sharply increased from 63.49 to 75.95, outperforming all previous models in the single-model setting. However, by fine-tuning on dense annotations, we decreased the performance on the sparse metrics (MRR, R@1, R@5, R@10, and Mean). This well-known phenomenon of the dataset is due to the misalignment of the dense and sparse annotations as previously observed by [32, 46]. In contrast to other baselines, VD-GR managed to keep relatively high sparse scores (4/5 metrics are the new state of the art) after fine-tuning.

- **Ensemble Setting:** As it is common practice, we fine-tuned an ensemble comprised of eight pre-trained VD-GR models to further improve the NDCG score. One set of four models was fine-tuned with Cross Entropy (CE) and with a varying number for GNN layers $K$, i.e. $K = \{1, 2, 3, 4\}$. Each model of the second set was trained using the ListNet [4] ranking optimisation method. As can be seen in the last section of Table 4, our ensemble model reached an NDCG score of 76.43 outperforming the closest competitor UniMM-UL with only 76.17 and, thus setting a new state of the art in the ensemble setting.

### 4.3. Ablation Study

#### Number of GNN layers $K$. This is an important hyperparameter of our model: If $K$ is too small, then the expressive power of the GNNs will be hampered. Contrarily, if $K$ is too large, the GNNs will suffer from over-smoothing [24]. To this end, we increased $K$ incrementally from one to four.

**Table 4.** Performance comparison on the test-std split of VisDial v1.0 dataset. $\blacktriangle$ indicates fine-tuning on dense annotations and $\blacklozenge$ denotes ensemble model. $\dagger$ denotes the use of extra large datasets for training.

<table>
<thead>
<tr>
<th>Ablation</th>
<th>NDCG</th>
<th>MRR</th>
<th>R@1</th>
<th>R@5</th>
<th>R@10</th>
<th>Mean</th>
</tr>
</thead>
<tbody>
<tr>
<td>VD-GR w/ $\lambda = 0$</td>
<td>65.69</td>
<td>67.71</td>
<td>54.07</td>
<td>85.03</td>
<td>92.84</td>
<td>3.33</td>
</tr>
<tr>
<td>VD-GR w/o $L_{warm}$</td>
<td>63.76</td>
<td>68.53</td>
<td>56.84</td>
<td>86.05</td>
<td>93.70</td>
<td>3.15</td>
</tr>
<tr>
<td>VD-GR w/o sharing</td>
<td>64.15</td>
<td>69.79</td>
<td>56.73</td>
<td>86.02</td>
<td>93.68</td>
<td>3.15</td>
</tr>
<tr>
<td>VD-GR w/o HN</td>
<td>64.11</td>
<td>69.86</td>
<td>56.88</td>
<td>85.97</td>
<td>93.67</td>
<td>3.14</td>
</tr>
<tr>
<td>VD-GR (Full)</td>
<td>64.32</td>
<td>69.91</td>
<td>57.01</td>
<td>86.14</td>
<td>93.74</td>
<td>3.13</td>
</tr>
</tbody>
</table>

**Table 5.** Performance comparison of ablated versions of our model on the val split of VisDial v1.0.

and used the sparse metrics of the task (MRR, R@1, R@5, R@10) for validation. As illustrated in Figure 5, the performance of our model peaked at $K = 2$ for all four metrics. Thus, we kept this value fixed in all previous experiments unless explicitly stated otherwise.

**Model Ablations.** In addition to the full model, we evaluated the following ablated versions:

- **VD-GR w/ $\lambda = 0$:** This variant did not apply the proposed residual connection of Equation 10 while augmenting the BERT hidden states.
- **VD-GR w/o $L_{warm}$:** This variant was not trained on the warm-up task of edge prediction as discussed in Sec. 3.4.
- **VD-GR w/o sharing:** This variant did not share the weights of the GNNs in each layer of our model.
- **VD-GR w/o HN:** This variant did not use hub-nodes to
propagate the information between the multi-modal GNNs. As can be seen from Table 5, the residual connections are essential for high performance. Without them, VD-GR achieved the lowest performance across all metrics. The same applies to the warm-up training stage: Although the performance of this ablated version improved over the previous one, it still performed significantly worse than our best model. The results also underline the importance of sharing the GNN weights within the different VD-GR layers: Although this version has more weights, it still performed worse than our best model on all metrics. This finding was to be expected given that the local structure of each modality does not change from one VD-GR layer to another. Finally, the importance of the inter-modal feature propagation using hub-nodes is highlighted by the two last rows of Table 5: The hub-nodes enabled our model to achieve the best performance across all metrics of the val split of the VisDial v1.0 dataset.

4.4. Qualitative Results

Finally, in Figure 6 we show selected qualitative samples (with more in the supplementary material) from the val split of VisDial v1.0 alongside the ground truth answers, as well as the top-1 predictions of our VD-GR model and VD-PCR for comparison since it achieved the second best results on this split. We make two interesting observations: (1) Our model deals better with questions that require exploiting local structure within modalities. For example, it managed to correctly answer \( Q_2 = \text{How many skiers can you see?} \) in the first dialog sample (see Figure 6a) by predicting 11 whereas VD-PCR predicted \( \text{I can only see two} \). We hypothesise that this is due to the fact that our model exploits the spatial structure of the visual input more effectively using the image graph compared to VD-PCR, although the latter has access to the same visual features. (2) VD-GR has more accurate semantic understanding of the question. This is highlighted in answering \( Q_3 = \text{How old does she seem to be?} \) (referring to the woman) in the second dialog sample (see Figure 6b). Whereas our model correctly predicted Early 20’s, VD-PCR failed by answering Grey, which is not a semantically-sound response. We posit that this advantage of our model is related to the fine-grained features of the question graph.

5. Conclusion

In this work we proposed VD-GR – a novel visual dialog model that combines pre-trained language models and GNNs. Specifically, VD-GR alternates between multi-modal graphs and BERT layers, and augments the hidden states of the latter with the fine-grained features obtained by the former. VD-GR propagates information from one modality graph to another in a cascaded manner using hub-nodes that link to all other nodes within each modality, thereby effectively alleviating the lack of inter-modal context. Extensive analyses underlined its effectiveness, while experiments on four challenging visual dialog datasets (VisDial v1.0, VisDial v0.9, VisDialConv, and VisPro) demonstrated its superior performance over existing methods.

Acknowledgments. A. Bulling was funded by the European Research Council (ERC; grant agreement 801708) and L. Shi was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany’s Excellence Strategy - EXC 2075–390740016. We acknowledge the support by the Stuttgart Center for Simulation Science (SimTech).
References


[31] Van Quang Nguyen, Masanori Suganuma, and Takayuki Okatani. Efficient Attention Mechanism for Visual Dialog that Can Handle All the Interactions Between Multiple Inputs. In ECCV, 2020. 1, 2, 6, 7
[38] Idan Schwartz, Seunghak Yu, Tamir Hazan, and Alexander G. Schwing. Factor Graph Attention. In CVPR, 2019. 1, 2, 7
[49] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. MSR-VTT: A Large Video Description Dataset for Bridging Video and Language. In CVPR, 2016. 1
[50] Yutong Yu, Hongming Zhang, Ruixin Hong, Yangqiu Song, and Changshui Zhang. VD-PCR: Improving visual dialog with pronoun coreference resolution. In EMNLP-IJCNLP, 2019. 6
[57] Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. From image descriptions to visual setonations: New similarity metrics for semantic inference over event descriptions. In TACL, 2014. 2