7-2023

Contrastive video question answering via video graph transformer

Junbin Xiao XIAO

Pan ZHOU
Singapore Management University, panzhou@smu.edu.sg

Angela YAO

Yicong LI

Richang HONG

See next page for additional authors

Citation

XIAO, Junbin Xiao; ZHOU, Pan; YAO, Angela; LI, Yicong; HONG, Richang; YAN, Shuicheng; and CHUA, Tat-Seng. Contrastive video question answering via video graph transformer. (2023). *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 45, (11), 13265-13280.
Available at: https://ink.library.smu.edu.sg/sis_research/9053

Author

Junbin Xiao XIAO, Pan ZHOU, Angela YAO, Yicong LI, Richang HONG, Shuicheng YAN, and Tat-Seng CHUA

Contrastive Video Question Answering via Video Graph Transformer

Junbin Xiao, Pan Zhou, Angela Yao, Yicong Li, Richang Hong, Shuicheng Yan and Tat-Seng Chua

Abstract—We propose to perform video question answering (VideoQA) in a **Co**ntrastive manner via a **V**ideo **G**raph **T**ransformer model (CoVGT). CoVGT's uniqueness and superiority are three-fold: 1) It proposes a dynamic graph transformer module which encodes video by explicitly capturing the visual objects, their relations and dynamics, for complex spatio-temporal reasoning. 2) It designs separate video and text transformers for contrastive learning between the video and text to perform QA, instead of multi-modal transformer for answer classification. Fine-grained video-text communication is done by additional cross-modal interaction modules. 3) It is optimized by the joint fully- and self-supervised contrastive objectives between the correct and incorrect answers, as well as the relevant and irrelevant questions respectively. With superior video encoding and QA solution, we show that CoVGT can achieve much better performances than previous arts on video reasoning tasks. Its performances even surpass those models that are pretrained with millions of external data. We further show that CoVGT can also benefit from cross-modal pretraining, yet with orders of magnitude smaller data. The results demonstrate the effectiveness and superiority of CoVGT, and additionally reveal its potential for more data-efficient pretraining. We hope our success can advance VideoQA beyond coarse recognition/description towards fine-grained relation reasoning of video contents. Our code is available at https://github.com/doc-doc/CoVGT.

Index Terms—VideoQA, Cross-Modal Visual Reasoning, Video-Language, Dynamic Visual Graphs, Contrastive Learning, Transformer

1 Introduction

Since 1960s, the very beginning of Artificial Intelligence (AI), continuous efforts and progress have been made towards developing AI systems that can demonstrate their understanding of the dynamic visual world by responding to the natural language queries in the context of videos which directly reflect our physical surroundings. In particular, from 2019 [1], [2] we have been witnessing a drastic advancement in such multi-disciplinary AI, where computer vision, natural language processing, and knowledge reasoning are coordinated for accurate decision-making. The advancement stems, in part, from the success of *cross-modal learning* on large-scale vision-text data [3], [4], [5], [6], [7], and in part from a unified deep neural network for modeling of vision and natural languages, *i.e.*, *transformer* [8]. As a typical multi-disciplinary AI task, Video Question Answering (VideoQA) has benefited a lot from these developments which helps to propel the field steadily forward over the use of purely conventional techniques [9], [10], [11], [12], [13], [14], [15].

Despite the excitement, we find that the advances made by such cross-modal learned *transformer*-style models mostly lie in answering questions that demand the coarse recognition or description of video contents [16], [17], [18], [19], [20], [21]. The problem of answering questions that challenge real-world visual relation reasoning [22], especially the causal and temporal relations that feature video dynamics at the action- and event-level [23], [24], [25], is largely under-explored. Cross-modal pretraining seems promising [26], [27], [28]. Yet, it requires the handling of prohibitively large-scale *video*-text data [28], [29], or otherwise the performances are still inferior to the state-of-the-art (SOTA) conventional techniques in handling temporal dynamics [26], [27], [30]. In this work, we attribute the failure to the following causes:

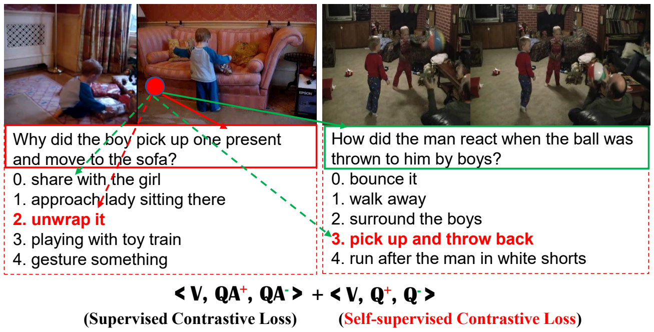

Why did the boy pick up one present and move to the sofa?
0. share with the girl
1. approach lady sitting there
2. unwrap it
3. playing with toy train
4. gesture something
How did the man react when the ball was thrown to him by boys?
0. bounce it
1. walk away
2. surround the boys
3. pick up and throw back
4. run after the man in white shorts
< V, QA+, QA- > + < V, Q+, Q- >
(Supervised Contrastive Loss)
(Self-supervised Contrastive Loss)

Fig. 1: Illustration of our contrastive learning strategy. For fully-supervised contrastive learning (dash arrows), we pull closer the video with its positive QA pair (red) in embedding space, while simultaneously pushing apart the video with the negative QA pairs (green). At the same time, we treat each video's question as its relevant description and collect irrelevant questions from other samples for self-supervised contrastive learning (solid arrows).

Video encoders are overly simplistic. Current video encoders are either 2D or 3D neural networks operated over sparse frames [31], [32], [33] or short video segments [34], [35], [36]. Such networks encode the video scene holistically, but fail to model the spatio-temporal interactions between visual objects and reason about their compositionality [15]. Resulting models are therefore weak in reasoning, or demand learning on large-scale video data to compensate for such weak form of inputs.

Formulation of VideoQA problem is sub-optimal. Existing

- *Junbin Xiao, Angela Yao, Yicong Li and Tat-Seng Chua are with National University of Singapore. Emails: {junbin,ayao,chuats}@comp.nus.edu.sg, liyicong@u.nus.edu.*
Pan Zhou and Shuicheng Yan are with Sea AI Lab, Singapore. Emails: {zhoupan, yansc}@sea.com
Richang Hong is with Hefei University of Technology, China. Email: hongrc.hfut@gmail.com
Corresponding to: {junbin, chuats}@comp.nus.edu.sg.

work solves VideoQA by classification, either cross-modal matching for multi-choice QA or multi-class classification for open-ended QA. The classification setting essentially learns a global representation (or classification layer) to predict the answer. Such a global representation is weak in disambiguating the correct versus incorrect answers. Because in multi-choice QA, the inputs of the video and question parts of a sample are the same and large, which may overwhelm the short candidate answers and dominate the overall representation. In open-ended QA, answers are treated as class indexes and their word semantics (which are helpful for QA) are not modelled. Neglecting answer semantics exacerbates the need for data and leads to sub-optimal performance as well.

The cross-modal correspondence between video and language is insufficiently mined. Existing models are learned on training samples with pure supervision oriented for question answering. Such models tend to learn spurious correlation between the inputs and the target labels. They fail to effectively capture the cross-modal correspondence (between video and language) which is essential for model to generalize [37], [38], [39], [40]. Consequently, the models are prone to overfitting. This problem manifests especially in inference-type QA [23], because the questions may involve multiple visual elements for multi-hop reasoning in space-time (refer to the examples in Fig. 1).

In light of this, we propose a **Co**ntrastively learned **V**ideo **G**raph **T**ransformer model (CoVGT) that tackles the aforementioned problems and advances previous *transformer*-style VideoQA models in several ways. First, for the video encoder, we design a dynamic graph transformer module that explicitly captures the objects and relations as well as their dynamics to improve visual reasoning in dynamic scenarios. Second, for the problem formulation, we leverage dual-transformer architecture which maintains *separate* vision and text transformers to encode video and text respectively for contrastive learning, instead of using cross-modal transformers to fuse the vision and text information for answer classification. Vision-text communication is done by additional cross-modal interaction modules. Importantly, to more effectively learn the cross-modal correspondence, we propose a joint fully- and self-supervised contrastive objective for model optimization (as illustrated in Fig. 1). The fully-supervised contrastive objective utilizes the ground-truth answer information and enables the model to be directly optimized to distinguish the correct and incorrect answers. At the same time, the self-supervised contrastive objective captures the insight that the paired questions of a given video should be relevant to the corresponding video contents while the unpaired ones should be irrelevant. It helps to suppress common question words and enhance the contribution of the visually-related linguistic concepts. This in turn should reduce spurious correlations between inputs and target labels during training and improve test performance.

We experiment on different VideoQA datasets that challenge the various aspects of cross-modal video understanding. CoVGT achieves SOTA results on VideoQA benchmarks that challenge the reasoning of complex spatio-temporal dynamics as well as causal and commonsense knowledge (*e.g.*, TGIF-QA [10], TGIF-QA-R [41], NExT-QA [23], STAR [24], and Causal-VidQA [25]). CoVGT also performs competitively on descriptive VideoQA datasets (*e.g.*, TGIF-FrameQA [10] and MSRVTT-QA [17]) . Notably, CoVGT's strong performance does *not* require external data to pretrain, though pretraining with a small amount of data additionally increases accuracy. The results clearly demonstrate CoVGT's effectiveness and superiority.

This paper extends our preliminary work VGT [42] in four major aspects: 1) We propose to learn the VGT model in a joint supervised and self-supervised contrastive manner rather than merely supervised contrastive learning. The enhanced objective brings steady performance improvements by making full use of cross-modal information that are available in the VQA datasets. 2) We explore superior model implementations to realize the architecture of video graph transformer (VGT). For instance, we find that by substituting BERT [1] with RoBERTa [43] for language encoding can improve the performances in most cases though with small sacrifice of efficiency. Moreover, we find that the multi-choice QA performances can be improved by adding some randomness to the negative answers. 3) We substantially extend our experiments to more datasets that target at cross-modal reasoning the various aspects of video contents, *e.g.*, STAR [24] for real-world situation reasoning and Causal-VidQA [25] for both evidence and commonsense reasoning. 4) We comprehensively analyze our model's strength and the contribution of each component. Furthermore, we share some heuristic observations about the performances of pretraining of visual graph transformer on cross-modal video reasoning tasks. For example, we find that existing cross-modal pretraining can hardly improve action- and event-level temporal relation reasoning in videos, which calls for more future efforts towards this direction.

Our contributions are summarized as follows:

1) We propose a contrastively learned video graph transformer model (CoVGT) to advance VideoQA from coarse recognition (or description) to fine-grained visual relation reasoning in dynamic scenarios. The model achieves SOTA results on a wide range of video reasoning tasks.
2) We propose a dynamic graph transformer module which jointly models the objects, their relations and dynamics, for visual reasoning. The module is task-agnostic and can be applied to other video-language tasks.
3) We propose to solve VideoQA in a joint fully-supervised and self-supervised contrastive manner by mining the distinction between correct and incorrect answers, as well as the relevant and irrelevant questions of a given video. Such a strategy is the first of its kind in VideoQA and shows steady advantages across different benchmarks.
4) To our best knowledge, we are the 1st to perform pretraining on visual graph transformer for video-language understanding (VLU). We hope that our success could promote VLU towards a more fine-grained and data-efficient direction.

2 Related Work

2.1 Conventional Techniques for VideoQA

Prior to Transformer's success for vision-language tasks, various techniques such as cross-modal attention [10], [44], [45], motion-appearance memory [11], [12], [46] and graphs [13], [47] have been applied to model informative contents from video for answering questions. Nonetheless. most of these techniques are built upon a sequence of frame-level or clip-level video representations which are insufficient for fine-grained object relation reasoning. More recently, graphs that exploit object-level representations [14], [15], [41], [46], [48], [49], [50], [51] have demonstrated superior performance, especially on benchmarks that emphasize relation reasoning between objects and actions [10], [23]. However, these graph-based methods build either monolithic graphs [14] that do not disambiguate between relations in 1) space and time and 2) local and global, or static graphs at frame-level without explicitly capturing the temporal dynamics [15], [41], [46]. The monolithic graph is cumbersome to adapt to long videos where multiple objects interact in space-time, while the static graph may lead to incorrect relations (*e.g.*, `hug` *vs.* `fight`) or fail to capture dynamic relations (*e.g.*, `take away`).

In contrast to monolithic and static graphs, we maintain a local to global hierarchical graph architecture which reflects the intrinsic structure of video contents, and meanwhile we design temporal graph transformer to explicitly capture the graph dynamics over time. Moreover, we integrate strong language models and explore both fully-supervised and self-supervised contrastive learning strategy rather than a simple classification, to learn the structured video representations.

2.2 Transformers for VideoQA

Transformer is a nascent technique for VideoQA. Several pioneering works [16], [21], [52] learn QA-favoured *transformer*-style models from HowTo100M [53] via designing proxy tasks (*e.g.*, masked language modeling [1] and cross-model matching [3], [21]), or curating dedicated supervisions (*e.g.*, future utterance [16] and QA pairs [52]). Despite their stronger performance over conventional models [11], [12], [13], [46], [47], these works focus on answering questions that require only the recognition [17] or shallow description [20] of the video scenes; their performances on visual relation reasoning [10], [23], [24] remains unknown. In addition, due to the heavy noise [54], [55] and limited data scope (instructional videos) of HowTo100M, models trained on it are weak in handling open-domain texts [28], [56].

Recent efforts tend to leverage open-domain vision-text data for representation learning. ClipBERT [26] takes advantage of human-annotated clean (to be distinguished from user-generated which is noisy) image-caption data [57], [58] for pretraining. The pretraining may benefit spatial object recognition and visual-content related language understanding. Yet, gains in temporal reasoning [10] are limited, because temporal relations are hard to learn from static images. Furthermore, the pretraining relies on clean annotations which are expensive to obtain and hard to scale up. Yet, ClipBERT has inspired works [28], [29] to take advantage of user-generated vision-description data (vastly available on the Web) [28], [56], [59] for end-to-end learning. While promising, it is computationally expensive to end-to-end learn from raw videos on such large-scale datasets. Corresponding models, if not pretrained, are prone to over-fit the target datasets, given the complex reasoning tasks defined over videos [23], [24] and the scarcity of annotated training data.

Overall, the *poor dynamic reasoning* and *data-hungry* problems in existing *transformer*-style video-language models largely motivate this work. To alleviate these problems, we explicitly model the objects and relations for dynamic visual reasoning and incorporate structure priors (or relational inductive bias [60]) into transformer architectures to reduce the demand for data.

2.3 Transformer Over Visual Graph

While graph and transformer techniques have gained increasing attention [61], [62], [63], [64] in modelling natural graph data (*e.g.* social connections), their combination in the video domain is still sparse. Two recent works [65], [66] have explored graph transformers for video-language tasks. [66] focuses on video dialogues and simply applying a global transformer over pooled graph representations built from static frames to represent a video. [65] proposes a tailored-made similarity-kernel in the self-attention blocks to capture the proximity of nodes in a pseudo 3D space. Both works do not explicitly take advantage of the objects and relations in adjacent frames to regulate the scene graph obtained at a static frame. Moreover, they neglect the local and global nature of video contents. In our work, we handle these problems by applying transformers global-locally at different-levels of graph granularity (*e.g.*, nodes, edges and graphs) without disturbing the original structure of transformer, which brings better performance for video question answering.

2.4 Contrastive Video-Language Understanding

Contrastive learning aims to automatically learn generalizable data representations by contrasting the similar data against the dissimilar ones in the embedding space. The idea has recently shown great success for cross-modal pretraining [3], [18], [52], [55], [67]. A handful of recent works have specially studied contrastive learning for VQA. They either learn to contrast the original and perturbed samples (obtained by masking [68] or paraphrasing [69]) for robust *image* question answering, or aim to *pretraining* a good model initialization to improve QA performance [67], [70]. Our use of contrastive learning differs from previous arts in two major aspects. First, we harmoniously integrate the supervised and self-supervised contrastive objectives into one learning framework, in which the supervised contrastive objective is directly oriented for question answering and the self-supervised objective aims to enhance the cross-modal correspondence learning. Second, our method does not require to additionally curate contrastive data inputs; instead, we sample hard negatives from existing samples which is more convenient and shows effectiveness as well.

3 Methodology

3.1 Overview of Contrastive Solution

Recall that our goal is to contrastively learn a video graph transformer model for VideoQA. To begin with, we formally define the VideoQA task as follows: given a video v and a question q, VideoQA aims to combine the two stream information v and q to predict the answer a. Depending on the task setting, a can be given as multiple choices along with each question for multi-choice QA, or it is defined in a global answer set for open-ended QA.

In this work, we handle both types of VideoQA by jointly optimizing the supervised and self-supervised contrastive objectives, *i.e.*, for multi-choice QA:

$$\mathcal{L} = \underbrace{\mathcal{L}_{vqa}(v, qa^{+}, qa^{-})}_{\text{supervised}} + \lambda \underbrace{\mathcal{L}_{vq}(v, q^{+}, q^{-})}_{\text{self-supervised}}, \quad (1)$$

and for open-ended QA:

$$\mathcal{L} = \underbrace{\mathcal{L}_{vqa}(vq, a^{+}, a^{-})}_{\text{supervised}} + \lambda \underbrace{\mathcal{L}_{vq}(v, q^{+}, q^{-})}_{\text{self-supervised}}, \quad (2)$$

where $\mathcal{L}_{vqa}$ is the supervised contrastive objective oriented for question answering, and $\mathcal{L}_{vq}$ is the self-supervised contrastive objective designed to enhance the cross-modal correspondence learning between the video and its related questions. Note that for pretraining with weakly-paired video-text data (*e.g.*, on WebVid [56]), only the self-supervised contrastive loss function is reserved. In both Eqn. (1) and Eqn. (2), vq and qa denote the combined (by concatenation) inputs of the video and question, question and

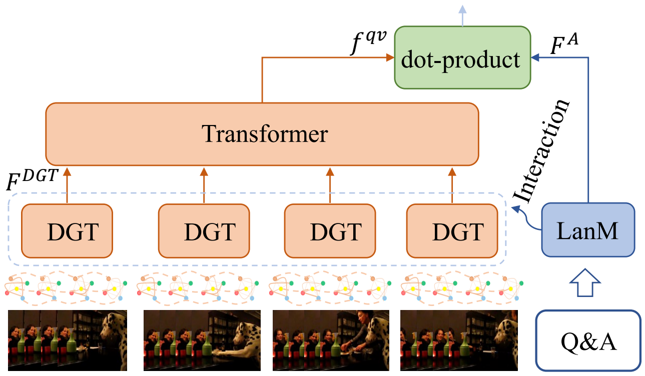


Fig. 2: Framework of video graph transformer (VGT). It projects the videos and texts into embedding space via a video encoder (orange) and a language model (blue) respectively, and computes their dot-product similarity for decision making. The video encoder maintains a local to global hierarchical architecture which includes a dynamic graph transformer (DGT) module and a global transformer. Additionally, a cross-modal interaction module is integrated to pinpoint the informative visual contents from videos.

answer respectively. $\mathcal{L}$ is the loss value, λ is a trade-off parameter and $\mathcal{L}_*(x, x^+, x^-)$ is given by the InfoNCE loss [71]:

$$-\mathbb{E}_i[\log(\frac{e^{s_{\text{VGT}}(x_i,x_i^+)}}{e^{s_{\text{VGT}}(x_i,x_i^+)}+\sum\limits_{(x_i,x_j^-)\in\mathcal{N}_i} e^{s_{\text{VGT}}(x_i,x_j^-)}})], \qquad (3)$$

where x, x^+, x^- denote the placeholders for the inputs of the anchor, positive and negative samples respectively. s_{VGT} denotes the dot-product value of the two inputs in the embedding space. The embeddings are computed by our video graph transformer (VGT) model illustrated in Fig. 2. Specifically:

$$s_{\text{VGT}}(x, x^+) = \mathcal{F}_{cm}(\mathcal{F}_{\text{vid}}(x), \mathcal{F}_{\text{lang}}(x^+))^\intercal \cdot \mathcal{F}_{\text{lang}}(x^+), \qquad (4)$$

where $\mathcal{F}_{cm}$, $\mathcal{F}_{\text{vid}}$ and $\mathcal{F}_{\text{lang}}$ denote the cross-modal interaction module, the video encoder and language encoder respectively. The parameters to be optimized are contained in these three modules.

Our solution differs from the vast majority of previous works that formulate and solve VQA as a classification problem [10], [15], [26]. Instead, we formulate VQA as a cross-modal contrastive learning problem and explicitly optimize the distinction between the correct and incorrect answers, as well as the relevant and irrelevant questions with respect to a given video. Our solution enjoys several major advantages: For multi-choice QA, it encodes video and QAs separately with different transformers (*e.g.*, $\mathcal{F}_{\text{vid}}$ and $\mathcal{F}_{\text{lang}}$), and thus circumvents the over-fitting and hard-negative answer issues resulted from representing each $\langle \text{video, question, candidate answer} \rangle$ triplet with a single feature from cross-modal transformer for classification. As a result, it can achieve much better performances. For open-ended QA, it encodes the semantics of the answer words which can benefit question answering. Last but not least, our formulation makes it convenience to adapt the model architecture to different datasets, since it dispenses with the classification layer which are usually dataset-specific. As such, it can fully enjoy the pretrained weights in the *pretrain-and-finetune* experiments.

We next elaborate the details of our model by first introducing the video encoder which comprises a video graph representation stage in Sec. 3.2 and a dynamic graph transformer module in Sec. 3.3, and finally a global transformer in Sec. 3.4. Cross-modal interaction along with text encoder is presented in Sec. 3.5. Finally, we introduce the answer decoder in Sec. 3.6.

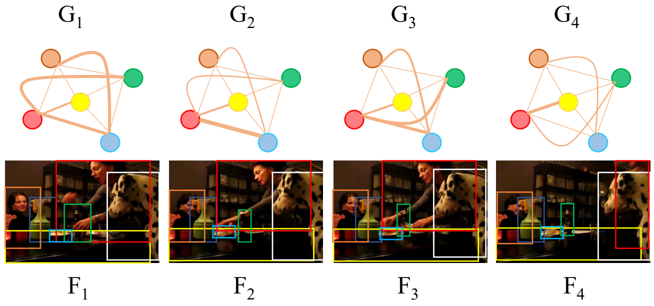


Fig. 3: Illustration of graph representation for a short video clip with 4 frames. The nodes of the same color represent the same object. Self-loops are omitted for brevity.

3.2 Video Graph Representation

To explicitly model the visual objects and their relationships for visual reasoning, it is necessary to represent a video as visual graphs whose nodes and edges correspond to visual objects and their relationships respectively [15], [72]. In this section, we introduce in detail how to convert a video clip into visual object graphs. Given a raw video, we sparsely sample l_v frames in a way analogous to [15]. The l_v frames are evenly distributed into k clips of length $l_c = \frac{l_v}{k}$. For each sampled frame (see Fig. 3), we extract n RoI-aligned features as object appearance representations $F_r = \{f_{r_i}\}_{i=1}^n$ along with their spatial locations $B = \{b_{r_i}\}_{i=1}^n$ with a pretrained object detector [32], [73], where r_i represents the i-th object region in a frame. Additionally, we obtain an image-level feature $F_I = \{f_{I_t}\}_{t=1}^{l_v}$ for all the sampled frames with a pretrained image classification model [31]. F_I will serve as the global contexts to augment the graph representations aggregated from the local objects.

To find the same object across different frames within a clip, we define a linking score s by considering the appearance and spatial location:

$$s_{i,j} = \psi(f^t_{r_i}, f^{t+1}_{r_j}) + \gamma * \text{IoU}(b^t_i, b^{t+1}_j), \qquad (5)$$

where ψ computes the appearance similarity between detected objects i and j in two adjacent frames. In this work, we use the cosine similarity for ψ, while IoU denotes intersection-over-union between the bounding boxes of objects i and j. The trade-off hyper-parameter γ is set to 1 in our experiments. The n detected objects in the first frame of each clip are designated as anchor objects. Detected objects in consecutive frames are then linked to the anchor objects by greedily maximizing s frame by frame[1]. By aligning the objects within a clip, we ensure the consistency of the node and edge representations for the graphs constructed at different frames (we construct one graph per frame at this stage).

After the alignment, we concatenate for each object its appearance f_r and location representation f_{loc}, and project the combined feature into the d-dimensional space via

$$f_o = \text{ELU}(\phi_{W_o}([f_r; f_{loc}])), \qquad (6)$$

where $[;]$ denotes concatenation and f_{loc} is obtained by applying a 1×1 convolution over the relative coordinates as in [15]. The function ϕ_{W_o} denotes linear transformation with parameters W_o.

[1] We assume that the group of objects do not change in a short video clip.

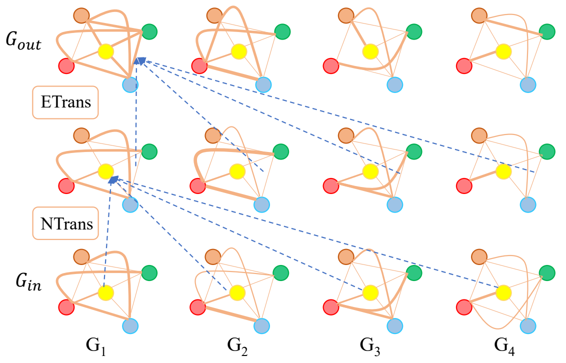


Fig. 4: Illustration of temporal graph transformer in a short video clip. It contains a node transformer (NTrans) and an edge transformer (ETrans) that are operated sequentially.

After obtaining the object representations $F_o = \{f_{o_i}\}_{i=1}^{n}$, their relationships in the t-th frame can be initialized as pairwise similarities in adjacency matrix R_t as follows

$$R_t = \sigma(\phi_{W_{ak}}(F_{o_t})\phi_{W_{av}}(F_{o_t})^\intercal), \quad t \in \{1, 2, \dots, l_v\}, \tag{7}$$

where $\phi_{W_{ak}}$ and $\phi_{W_{av}}$ denote linear transformations with parameters W_{ak} and $W_{av} \in \mathbb{R}^{d \times \frac{d}{2}}$ respectively. We use different transformations to make the relation asymmetric, which reflects the nature of real-world subject-object interaction [50], [74] (*e.g.*, `hit` *vs.* `being hit`). As for symmetric relations, we expect that their transformed representations are quite similar. Here $\intercal$ indicates matrix transpose, and σ is the Softmax operation that normalizes each row. After initialization, the values of the i-th row in the adjacency matrix R_t denote the relations of object i with regard to all of the other objects in the t-th frame.

The adjacency matrix R is obtained independently for each frame; hence there are l_c such adjacency matrices for each video clip. However, as the objects are aligned in the video clip, each entry of the adjacency matrices always corresponds to the relation for the same pair of objects. For brevity, we use $G_t = (F_{o_t}, R_t)$ to denote the graph representation of the t-th frame where F_o are node representations and R are edge representations of the graph.

3.3 Dynamic Graph Transformer

Previous efforts construct visual graphs at static frame level [15], [41], [46], thus failing to explicitly capture the object dynamics. Therefore, at the heart of our video encoder is the dynamic graph transformer (DGT) module. This module takes as inputs a set of visual graphs $\{G_t\}_{t=1}^{L_v}$ clip-wisely, and outputs a sequence of representations $F^{DGT} \in \mathbb{R}^{d \times k}$ by mining the temporal dynamics of objects and their spatial interactions. To achieve this, we sequentially operate a temporal graph transformer unit, a spatial graph convolution unit and a hierarchical aggregation unit.

3.3.1 Temporal Graph Transformer

As illustrated in Fig. 4, the temporal graph transformer unit operates over a single video clip. It takes as input the set of graphs $G_{in} = \{G_t\}_{t=1}^{l}$ from that clip and outputs a new set of graphs G_{out} by mining the temporal dynamics of the objects and their relations (*i.e.* interactions). Specifically, there is a node transformer (NTrans) and an edge transformer (ETrans) that operate over the graph nodes and edges respectively. The two types of transformers are designed to exploit the objects and their relations in adjacent frames to enhance the object representations and calibrate the relationships captured at static frames. For completeness, we briefly recap the self-attention mechanism in transformer [8]. Given a sequence of inputs $X_{in} = \{x_{in}^{t}\}_{t=1}^{l}$, the transformer module mix token information by employing a multi-head self-attention (MHSA):

$$X_{out} = \text{MHSA}(X_{in}) = \phi_{W_c}([h_1; h_2; \dots, h_e]), \tag{8}$$

where ϕ_{W_c} is a linear transformation with parameters W_c, and

$$h_i = \text{SA}(\phi_{W_{i_q}}(X_{\text{in}}), \phi_{W_{i_k}}(X_{\text{in}}), \phi_{W_{i_v}}(X_{\text{in}})), \tag{9}$$

where $\phi_{W_{i_q}}$, $\phi_{W_{i_k}}$ and $\phi_{W_{i_v}}$ denote the linear transformations of the query, key, and value vectors of the i-th self-attention (SA) head respectively. e denotes the number of self-attention heads, and SA is defined as:

$$\text{SA}(X_q, X_k, X_v) = \sigma\left(X_k X_q^\intercal / \sqrt{d_k}\right) X_v, \tag{10}$$

in which d_k is the dimension of the key vector. Finally, a skip-connection with layer normalization (LN) is applied to the output sequence $X = LN(X_{out} + X_{in})$. The final X may undergo further MHSAs depending on the number of transformer layers.

In our temporal graph transformer, we first apply H number of MHSA blocks to enhance the node (or object) representations by aggregating information from other nodes of the same object from all l adjacent frames within a clip:

$$F'_{o_i} = \text{NTrans}(F_{o_i}) = \text{MHSA}^{(H)}(F_{o_i}), \tag{11}$$

in which $F_{o_i} \in \mathbb{R}^{l \times d}$ denotes a sequence of feature representations corresponding to object i in a video clip of length l. Our motivation behind the node transformer is that it helps model the change of single object behaviours and thus infer the dynamic actions (*e.g.* `bend down`). Also, it is helpful in improving the appearance features in the cases where the object at certain frames suffer from motion blur or partial occlusion.

Based on the new node representations $F'_o = \{F'_{o_i}\}_{i=1}^{n}$, we can update the relation matrix R via Eqn. (7). Then, to explicitly model the temporal dynamics of the relations, we further apply a edge transformer (of H-MHSA layers) on all the updated relation matrices:

$$\mathcal{R}' = \text{ETrans}(\mathcal{R}) = \text{MHSA}^{(H)}(\mathcal{R}), \tag{12}$$

where $\mathcal{R} = \{R_t\}_{t=1}^{l} \in \mathbb{R}^{l \times d_n}$ ($d_n = n^2$) is the l adjacency matrices that are row-wisely expanded. ffHere, our motivation is that the relations captured at the static frames may be incorrect (*e.g.*, `hug` *vs.* `fight`), trivial (*e.g.*, `tough` *vs.* `wipe`) or incomplete (*i.e.*, unable to identify dynamic relations like `put down`), and the edge transformer can help to calibrate the wrong relations and recall the missing ones. Note that in our implementation, the temporal positions for both node- and edge- transformers are not used since the transformers are applied within a short video clip, and we empirically find that the temporal positions do not help the performance. For brevity, we refer to the resultant graph at the t-th frame as $G_{out_t} = (F'_{o_t}, R'_t)$.

3.3.2 Spatial Convolution and Hierarchical Aggregation

Spatial Graph Convolution. The temporal graph transformer focuses on temporal relation capturing. To reason over the object spatial interactions, we apply a U-layer graph attention convolution [75] (as illustrated in Fig. 5) on all the l_v graphs:

$$F_o^{'(u)} = \text{ReLU}((R' + I)F_o^{'(u-1)} W^{(u)}), \tag{13}$$

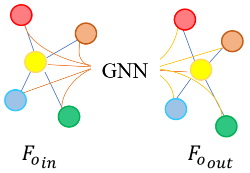


Fig. 5: Illustration of spatial graph convolution. Given a set of input nodes $F_{o_{in}}$, this unit applies GNN to enhance the node representations according to their relations with regard to their neighbors. The nodes denote the visual objects.

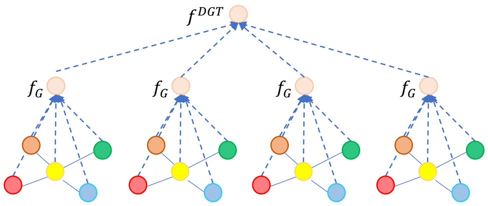


Fig. 6: Illustration of hierarchical aggregation.

in which $W^{(u)}$ denote the graph parameters at the u-th layer. I is the identity matrix for skip connections. $F_o^{'(u)}$ are initialized with F_o' which are the output node representations of the temporal graph transformer unit. The index t is omitted for brevity. Afterwards, a last skip-connection: $F_{o_{out}} = F_o' + F_o^{'(U)}$ is used to obtain the final node representations.

Hierarchical Aggregation. The node representations so far have explicitly taken into account the objects' temporal dynamics and spatial interactions. Nonetheless, such interactions are mostly atomic. To aggregate the atomic interactions into higher level video elements and to narrow the semantic gap between the visual and textual representations, we design a hierarchical aggregation strategy as illustrated in Fig. 6. First, we aggregate the graph nodes at each frame by a simple self-attention:

$$f_G = \sum\nolimits_{i=1}^{N} \alpha_i F_{o_{out_i}}, \quad \text{where} \quad \alpha = \sigma(\phi_{W_G}(F_{o_{out}})), \quad (14)$$

and ϕ_{W_G} is a linear transformation with parameters $W_G \in \mathbb{R}^{d \times 1}$. The graph representation usually captures local object interactions. However, it may lose sight of the global picture of a frame, especially since we only detect n objects and cannot guarantee that the detected objects will contain all the objects of interest in that frame. To enhance graph representation, we concatenate F_G with the frame-level appearance features F_a:

$$f_G = \mathrm{ELU}(\phi_{W_m}([\phi_{W_f}(f_a); f_G])) \quad (15)$$

in which ϕ_{W_m} and ϕ_{W_f} are linear transformations with parameters $W_m \in \mathbb{R}^{2d \times d}$ and $W_f \in \mathbb{R}^{2048 \times d}$ respectively.

We next aggregate the local interactions to obtain a sequence of clip-level feature representations via a simple mean-pooling:

$$f^{\mathrm{DGT}} = \mathrm{MPool}(F_G) = \frac{1}{l_c} \sum_{t=1}^{l} f_{G_t} \quad (16)$$

The set of k clips are finally represented by $F^{\mathrm{DGT}} = \{f_c^{\mathrm{DGT}}\}_{c=1}^{k}$.

3.4 Global Transformer

The aforementioned DGT module pays attention to derive informative visual clues from the local video contents. To capture the temporal and causal relations between these local video contents, we employ another H-layer transformer over the outputs of DGT (*i.e.* F^{DGT}), and add learnable sinusoidal temporal position embeddings [1]. Finally, the transformer's outputs are mean-pooled to obtain the global representation $f^v \in \mathbb{R}^d$ for the entire video. This process is formally represented by

$$f^v = \mathrm{MPool}(\mathrm{MHSA}^{(H)}(F^{\mathrm{DGT}})). \quad (17)$$

The global transformer has two major advantages. First, it retains the overall hierarchical structure which progressively drives the video elements at different granularity as in [15]. Second, it enhances the compatibility of the visual and text representations which may benefit cross-modal comparison.

3.5 Cross-modal Interaction

To find the informative visual contents with respect to a particular text query, a cross-model interaction between the visual and textual nodes is designed. Given a set of visual nodes represented by X^v, we incorporate textual information represented by $X^q = \{x_m^q\}_{m=1}^{M}$ (M denotes number of tokens in the text query) into the visual nodes via

$$x^{qv} = \mathcal{F}_{cm}(x^v, X^q) = x^v + \sum\nolimits_{m=1}^{M} \beta_m x_m^q, \quad (18)$$

where X^v and X^q denote the placeholders for the visual and text representations for interaction module. $\beta = \sigma((x^v)^\intercal X^q)$. The query-aware nodes X^{qv} then substitute the original visual nodes and proceed to subsequent operations.

In principle, this cross-modal interaction module can be applied at different level of visual abstractions (object-level F_O in Eqn. (6), frame-level F_G in Eqn. (15) or clip-level F^{DGT} in Eqn. (16)) to be in line with the multi-granularity of video elements and linguistic concepts [15]. In our implementation, we investigate several variants and find that performances vary among different datasets. As a default, we simply plug it at the outputs of the DGT module (*i.e.* $X^v := F^{\mathrm{DGT}}$) for efficiency consideration as the number of visual representations at this stage is much smaller. Empirically, this implementation generalizes better across different datasets, because the visual nodes at this level have already absorbed the information of both static and dynamic from preceding layers. For the textual node X^q, we obtain them by a simple linear projection on the token outputs of a language model (*e.g.*, $\mathcal{F}_{\text{lang}}$ can be BERT [1] or RoBERTa [43]):

$$X^q = \phi_{W_Q}(\mathcal{F}_{\text{lang}}(Q)), \quad (19)$$

where $W_Q \in \mathbb{R}^{768 \times d}$, and Q denotes question words in open-end QA or words of QA pairs in multi-choice QA. Particularly, in multi-choice QA, the candidate answers of a question will be appended to the question to form multiple textual queries. In this scenario, we max-pool the obtained query-aware visual representations with respect to different QA queries to find the one that is mostly relevant to the video. Our intuition is that in most cases, the QA queries corresponding to the correct answers are mostly relevant to the video.

3.6 Answer Prediction

To get a global representation for a particular answer candidate, we mean-pool its token representations from the language model:

$$f^A = \mathrm{MPool}(X^A), \quad (20)$$

where X^A denotes a candidate answer's token representations, and is obtained by feeding the text answer to the language model in

TABLE 1: Dataset statistics. OE-1450: open-ended QA with 1450 global answer candidates. MC-5: multi-choice QA with 5 options and only one of them is correct. Note that TGIF-QA-R [41] shares the same statistics with TGIF-QA.

Datasets	Main Challenges	#Videos/#QAs	Train	Val	Test	Video Length (s)	QA
NExT-QA [23]	Causal & Temporal Reasoning	5.4K/48K	3.8K/34K	0.6K/5K	1K/9K	44	MC-5
TGIF-QA [10]	Repetition Action	22.8K/22.7K	20.5K/20.5K	-	2.3K/2.3K	3	MC-5
	State Transition	29.5K/58.9K	26.4K/52.7K	-	3.1K/6.2K	3	MC-5
	Frame QA	39.5K/53.1K	32.3K/39.4K	-	7.1K/13.7K	3	OE-1450
STAR-QA [24]	Situated Reasoning	5K/ 60K	3K/46K	1K/7K	1K/7K	30	MC-4
Causal-VidQA [25]	Evidence & Commonsense Reasoning	26.9K/ 161.4K	18.8K/112.7K	2.7K/16.0K	5.4K/32.6K	9	MC-5
MSRVTT-QA [17]	Visual Recognition	10K/ 244K	6.5K/159K	0.5K/12K	3K/73K	15	OE-4000

a way analogous to Eqn. (19). Its similarity with the query-aware video representation f^{qv} (obatined via Eqn. (17)) is thus obtained through a dot-product between the two vectors. Consequently, the candidate of maximal similarity is returned as a prediction:

$$a^* = \arg\max((f^{qv})^\intercal F^A), \tag{21}$$

in which $F^A = \{f^A_a\}_{a=1}^{|\mathcal{A}|} \in \mathbb{R}^{d \times |\mathcal{A}|}$, and $|\mathcal{A}|$ means the number of answer candidate. Additionally, for open-ended QA, we follow previous works [15] and enable a video-absent QA by directly computing the similarities between the question representation f^q (obtained in a way similar to f^A.) and the answer representations F^A. As a result, the final answer can be a joint decision:

$$a^* = \arg\max((f^{qv})^\intercal F^A \odot (f^q)^\intercal F^A), \tag{22}$$

in which $\odot$ is element-wise product.

4 Experiments

4.1 Datasets

We experiment on different datasets. Seven datasets (NExT-QA [23], TGIF-Action and Transition [10], TGIF-QA-R Action and Transition [41], STAR-QA [24] and Causal-VidQA [25]) challenge the complex temporal and causal relation as well as commonsense reasoning in videos (feature temporal dynamics). Two additional datasets (TGIF FrameQA [10] and MSRVTT-QA [17]) challenge the recognition of the video objects, their attributes and actions as well as activities (feature frame statics). The related statistics of the datasets are presented in Tab. 1. Other details are given in the Appendix A. For all experiments, we follow standard protocol and report accuracy (percentage of correctly answered questions) for evaluation metric.

4.2 Implementation Details

We decode the video into frames following [15] and sparsely sample $l_v = 32$ frames for each video. The frames are distributed into $k = 8$ clips whose length $l_c = 4$. For each frame, we detect and select the top $n = 10$ regions of high confidence by default (20 for NExT-QA following [15] and 5 regions are used in the pretraining-free experiments (see study in Appendix C.5)), using the object detection model from [73] which is Faster R-CNN with ResNet-101 backbone pretrained on the Visual Genome dataset [58]. The frame appearance feature F_I is extracted from ResNet-101 pretrained on ImageNet [76]. The dimension of the models' hidden states is set to $d = 512$, and the default number of graph layers is $U = 2$. Besides, the default number of layers and self-attention heads in transformer are $H = 1$ and $e = 8$ ($e = 5$ for edge transformer in DGT) respectively.

For fully-supervised contrastive learning, the negative answers are from two sources comprising the original multiple choices and those sampled from the other questions' correct answers at a probability of 0.3. In open-ended QA, all the other answers in the answer set are treated as the negatives for a given question. For self-supervised contrastive learning, we sample questions from the other samples and treat them as the negative descriptions of the anchor video. In particular, the sampled negatives are from the questions of the same category as the positive question, so as to ensure the hard negatives. The question types are obtained by simple question parsing (see Appendix B for details). The trade-off parameter λ is set to 1. We employ Adam [77] optimizer with an initial learning rate of 1×10^{-5} or 5×10^{-5}. The learning rate will degenerate following a cosine annealing schedule with respect to the total iterations. The batch size is set to 64, and the maximum epoch varies from 10 to 30 among different datasets. For the pretraining experiment, we download about 0.18M video-text data (less than 10%) from WebVid2M [56] and pretrain 2 epochs. More details are presented in the Appendix B.

4.3 State-of-the-Art Comparison

4.3.1 Results on NExT-QA

In Tab. 2, we compare CoVGT with some of the lastest graph-based and transformer-based methods on NExT-QA (results per question type are found in Appendix C.1). The results show that CoVGT significantly surpasses the previous SOTAs on all tasks defined in NExT-QA, especially on the causal (Acc@C) and temporal (Acc@T) reasoning tasks, improving the accuracy on the val and test sets by 5.7% (*vs.* ATP [30]) and 7.7% (*vs.* HQGA [15]) respectively. Notably, such strong performance does not use external vision-text data for pretraining. The pretrained variants (with 0.18M data) can further increase the accuracy by about 2.0% for VGT on both the validation and test sets, and 0.7% for CoVGT on the validation set. The relatively smaller improvement for CoVGT is because pretraining has lost its dominated superiority in answering the descriptive questions (Acc@D). A detailed analysis of the effectiveness of pretraining is presented in Sec. 4.5.

CoVGT *vs.* VQA-T: Compared with VQA-T [52] which also solves VideoQA in a contrastive manner but in a supervised fashion analogous to the left term of our Eqn. (2)), our CoVGT wins on several aspects. First, we design DGT for video encoding while VQA-T uses S3D [36], [55] (see the benefits of S3D → DGT). Second, we design both supervised and self-supervised contrastive objectives which steadily benefit model optimization (see Sec. 4.4). Third, we encode the question and answer with a single language model, whereas VQA-T encodes question and answer independently with two language models. Our method benefits answer encoding with question as context and reduces model parameters as well (see VGT(DistilBERT)). In addition, such implementation permits direct pretraining on user-generated video-text data without the need to generate QA pairs. Finally, VQA-T applies cross-modal transformer (CMTrans) to fuse the video-question pair, whereas we design more light-weight cross-modal interaction module (CM). The results in the 2nd row of Tab. 3 indicates that CM has little impact on model performance but reduces model parameters.

TABLE 2: Accuracy (%) comparison on NExT-QA [23]. F: Frame-level feature from ResNet or ViT or their variants. C: Clip-level feature from 3D neural networks. C^+: cross-modal pretrained S3D [55]. R: Region-level feature from Faster R-CNN. Acc@C, T, D, denote accuracy for Causal, Temporal, and Descriptive questions respectively. *: results reproduced with the official code. The **best** and 2nd best results are highlighted in bold and underline respectively.

Methods	Pretrain		Video	Text	NExT-QA Val				NExT-QA Test			
	Dataset	Size			Acc@C (48%)	Acc@T (29%)	Acc@D (23%)	Acc@All	Acc@C (48%)	Acc@T (29%)	Acc@D (23%)	Acc@All
VQA-T* [52]	-	-	C^+	DistilBERT	41.66	44.11	59.97	45.30	42.05	42.75	55.87	44.54
HGA [13]	-	-	F, C	BERT	46.26	50.74	59.33	49.74	48.13	49.08	57.79	50.01
IGV [39]	-	-	F, C	BERT	-	-	-	-	48.56	51.67	59.64	51.34
HQGA [15]	-	-	R, F, C	BERT	48.48	51.24	61.65	51.42	49.04	52.28	59.43	51.75
VQA-T* [52]	HTVQA69M	69M	C^+	DistilBERT	49.60	51.49	63.19	52.32	47.89	50.02	61.87	50.83
P3D-G [65]	-	-	R, C	BERT	51.33	52.30	62.58	53.40	-	-	-	-
ATP [30]	-	-	F	BERT	53.1	50.2	66.8	54.3	-	-	-	-
VGT	-	-	R, F	BERT	52.28	55.09	64.09	55.02	51.62	51.94	63.65	53.68
VGT (PT)	WebVid(WV)	0.18M	R, F	BERT	53.43	56.39	69.50	56.89	52.78	54.54	67.26	55.70
CoVGT	-	-	R, F	RoBERTa	58.80	57.44	69.37	60.01	**58.53**	57.02	66.83	59.42
CoVGT (PT)	WebVid(WV)	0.18M	R, F	RoBERTa	**59.69**	**58.00**	**69.88**	**60.73**	58.00	**57.96**	**68.40**	**59.69**

TABLE 3: Detailed comparison between VGT and VQA-T [70]. CMTrans: cross-modal transformer.

Models	Size (M)	NExT-QA Val			
		Acc@C	Acc@T	Acc@D	Acc@All
VQA-T [52]	600	41.66	44.11	59.97	45.30
CMTrans → CM	573	42.27	44.29	58.17	45.40
S3D → DGT	641	47.53	48.08	62.42	50.02
VGT (DistilBERT)	346	50.71	51.67	66.41	53.46

CoVGT *vs*. HQGA: HQGA [42] constructs graphs on static frames and does not explicitly model the temporal dynamics, whereas we design dynamic visual graphs which exploits the graphs of adjacent frames to regulate the graphs constructed at static frame. Moveover, HQGA extracts language embeddings offline, while we enable online finetuning in an end-to-end fashion. Tab. 2 and Tab. 7 show that our method surpasses HQGA on all tasks across different datasets. Finally, our stronger results come with more sparse video sampling (see Appendix C.5) and without using motion feature. The comparison again points towards the absolute superiority of CoVGT over HQGA.

CoVGT *vs*. P3D-G: While P3D-G [65] also applies transformer over visual graphs, the transformer is monolithic and operates over a graph constructed in a pseudo 3D space. First, a single monolithic transformer cannot reflect the local and global nature of video content. Second, to obtain the pseudo 3D graph and register the objects into it, P3D-G needs to transfer 2D RGB frames into RGB-D ones and merge objects globally throughout a whole video. Both processes may accumulate errors from wrong detections and thus jeopardize the performance. In our model, we design local-to-global graph transformer architecture and only link the object within short video clips. Our method is more reasonable in encoding long videos with rich dynamics.

CoVGT *vs*. ATP: ATP [30] focuses on probing key frames from videos for question answering, by using a frozen vision-language model pretrained on image-text data (*i.e.*, CLIP [3]). It can well answer questions that invoke frame-level information but may fail to jointly reason over multiple frames to link the atomic things together for compositional and temporal video understanding. In contrast, we have the dynamic graph transformer module to realize this, which we believe, largely contributes to our superior performance. For better analysis, we additionally report results on the ATP-hard subset of NExT-QA validation set. The subset highlights video-level visual-language understanding which is in line with our aim. The results in Tab. 4 show that our CoVGT model significantly surpasses both ATP and its temporal version.

TABLE 4: Comparison on ATP-hard subset [30] of NExT-QA.

Methods	NExT-QA Val (ATP-hard subset)	
	Acc@C	Acc@T
ATP [30]	19.6	22.6
Temporal[ATP] [30]	38.4	36.5
HGA [13]	43.3	45.3
CoVGT	**51.8**	**50.5**

TABLE 5: Accuracy (%) comparison on STAR-QA [24]. I: Interaction, S: Sequence, P: Prediction, F: Feasibility. M: Mean. Other results are token from [24].

Methods	STAR-QA Test				
	Acc@I (35.6%)	Acc@S (48.4%)	Acc@P (9.1%)	Acc@F (6.9%)	Acc@M
NS-SR [24]	30.88	31.76	30.23	29.73	30.65
CLEVRER [78]	33.25	32.67	30.69	30.43	31.76
VisualBERT [80]	33.59	37.16	30.95	30.84	33.14
LGCN [14]	39.01	37.97	28.81	26.98	33.19
HCRN [81]	39.10	38.17	28.75	27.27	33.32
ClipBERT [11]	39.81	43.59	32.34	31.42	36.79
VGT	42.38	47.01	41.18	39.13	42.43
VGT (PT)	44.63	49.54	43.44	39.65	44.32
CoVGT	44.83	48.72	41.34	41.04	43.98
CoVGT (PT)	**46.23**	**50.34**	**45.11**	**43.13**	**46.20**

4.3.2 Results on STAR-QA and Causal-VidQA

Tab. 5 shows our results on STAR-QA for visual situated reasoning. CoVGT outperforms the previous SOTA (*i.e.* ClipBERT [26]) on all the defined four tasks by a clear margin, gaining remarkable improvements of 9.4% and 7.2% in mean accuracy with and without pretraining respectively. Again, we find that pretraining steadily boosts performances on all the four tasks, which reveals the strong learning capacity of CoVGT. We also notice that our results surpasses those neuro-symbolic baselines [24], [78] which relies on additionally functional programs for supervision instead of using only the QA annotations. The observation is in line with the recent work [79] which shows the strength of attention over object embedding for complex visual reasoning.

our results on Causal-VidQA [25] are presented in Tab. 6. The results show that CoVGT remarkably surpasses the previous reported SOTA method (B2A [47]) by $\sim$10% in overall accuracy and beats it on all sub-tasks. We additionally reproduce some stronger baselines (Fine-tuning RoBERTa [43]) for better comparison. The results show that CoVGT still outperforms them by a substantial margin. In addition, the strong performance of a pure language model (*i.e.* RoBERTa) suggest that the QA contents are biased to text comprehension (also see examples in Appendix C.6). We find that our method performs well on the commonsense reasoning tasks (*i.e.*, Prediction and Counterfactual). We attribute such strong performance to our contrastive learning strategy as well as the exploitation of advanced language models. Moreover, pretraining

TABLE 6: Accuracy (%) comparison on Causal-VidQA [25]. The results of B2A [47] are reproduced by [25] which uses off-the-shelf BERT without finetuning it. (RoI+RoBERTa)'s results are produced by us via filling the Seg_ID in the texts with the corresponding ground-truth visual object representations (mean-pooled across time for each object.) and sending the combined tokens into RoBERTa for end-to-end finetuning. D: Description, E: Explanation, P: Prediction, C: Counterfactual.

Methods	Video	Text	Causal-VidQA Test								
			Acc@D	Acc@E	Acc@P			Acc@C			Acc@All
					Q → A	Q → R	Q → AR	Q → A	Q → R	Q → AR	
B2A [47]	F, C	BERT	66.21	62.92	48.96	50.22	31.15	53.27	56.27	35.16	49.11
BlindQA*	-	RoBERTa	64.65	70.03	47.39	48.39	31.87	62.94	63.12	45.26	52.95
RoI+RoBERTa*	R	RoBERTa	70.10	72.09	55.46	56.03	38.48	61.15	63.58	45.04	56.43
CoVGT	R, F	RoBERTa	<u>73.46</u>	<u>74.80</u>	<u>58.65</u>	<u>56.38</u>	<u>39.45</u>	**66.99**	<u>64.25</u>	<u>48.48</u>	<u>59.05</u>
CoVGT (PT)	R, F	RoBERTa	**74.36**	**75.55**	**60.74**	**60.41**	**43.30**	<u>65.64</u>	**64.97**	**50.02**	**60.81**

TABLE 7: Accuracy (%) comparison. † denotes our newly curated multiple choices by rectifying the redundant options in TGIF-QA-R [41]. We grey out the results reported in [41] regarding these two tasks as the QAs are slightly different.

Methods	Pretrain		Text	TGIF-QA		TGIF-QA-R	
	Dataset	Size		Action	Trans	Action†	Trans†
LGCN [14]	-	-	GloVe	74.3	81.1	-	-
HGA [13]	-	-	GloVe	75.4	81.0	-	-
HCRN [81]	-	-	GloVe	75.0	81.4	*55.7*	*63.9*
B2A [47]	-	-	GloVe	75.9	82.6	-	-
HOSTR [48]	-	-	GloVe	75.0	83.0	-	-
HAIR [46]	-	-	GloVe	77.8	82.3	-	-
HQGA [15]	-	-	BERT	76.9	85.6	-	-
PGAT [41]	-	-	GloVe	80.6	85.7	<u>*58.7*</u>	<u>*65.9*</u>
MASN [49]	-	-	GloVe	84.4	87.4	-	-
MHN [82]	-	-	GloVe	83.5	90.8	-	-
ClipBERT [26]	VG, COCO	-	BERT	82.8	87.8	-	-
SiaSRea [27]	VG, COCO	-	BERT	79.7	85.3	-	-
MERLOT [28]	YT, CC	183M	BERT	94.0	96.2	-	-
VGT	-	-	BERT	**95.0**	**97.6**	59.9	70.5
VGT (PT)	WV	0.18M	BERT	93.1	<u>97.2</u>	60.5	71.5
CoVGT	-	-	RoBERTa	<u>94.7</u>	**97.6**	<u>60.8</u>	**73.8**
CoVGT (PT)	WV	0.18M	RoBERTa	91.3	96.2	**61.0**	<u>73.2</u>

helps a lot for predicting invisible answers. Additionally, the relatively high accuracy on the separated answer (Q→A) and reason (Q→R) but the low joint prediction accuracy (Q→AR) indicates that model fails to explain the correct answers or wrongly explain the incorrect answers for certain cases. Such failure asks for more future efforts in modelling the consistency between the answers and the corresponding reasons.

4.3.3 Results on TGIF-QA and TGIF-QA-R

In Tab. 7, we compare our method with previous arts on the TGIF-QA [10] and TGIF-QA-R [41] datasets for repeating action recognition and state transition. The results show that VGT or CoVGT surpasses the previous pretraining-free SOTA results significantly by ∼10% (VGT *vs*. MASN [49]: 95.0% *vs*. 84.4%) and 7% (VGT *vs*. MHN [82]: 97.6% *vs*. 90.8%) respectively. It even outperforms the pretraining SOTA (*i.e*. MERLOT [28]) by about 1.0%, even though we do not use external data for cross-modal pretraining. On TGIF-QA-R [41] which fixes the answer bias issue in TGIF-QA, CoVGT improves the previous SOTA (PGAT [41]) by about 2% and 8% respectively on the repeating action and state transition tasks. The results again demonstrate the strength of our method.

However, we notice that the improvements of pretraining are unstable on the 4 tasks. We believe that this is because our method has already achieved strong performance without pretraining, and the noises resulted from the pretraining data jeopardize the performances. A further study by key-word searching the video-text data from WebVid [56] reveals that the pretraining data rarely invoke repeating actions and temporal languages.

4.3.4 Results on Descriptive QA datasets

While we focus on answering inference-type questions that feature temporal dynamics, we find that CoVGT performs favourably well on the recognition-based VideoQA datasets, *i.e*., TGIF-FrameQA and MSRVTT-QA. Concretely, our method surpasses the previous object graph-based (pretraining-free) SOTA and shows competitive results to several pretrained Transformer models (shown in Tab. 8). Nonetheless, we notice that there is still a clear gap between CoVGT and the SOTA pretrained methods (MERLOT [28]). The comparison indicates that pretraining with large-scale data is the key to high-ranking results on these datasets, while dense video sampling and relation modelling seem less effective.

TABLE 8: Accuracy (%) comparison on descriptive QA datasets.

Methods	Pretrain		Text	TGIF -FrameQA	MSRVTT -QA
	Dataset	Size			
HOSTR [48]	-	-	GloVe	58.0	35.9
HAIR [46]	-	-	GloVe	60.2	36.9
MASN [49]	-	-	GloVe	59.5	35.2
CoMVT [16]	-	-	BERT	-	37.3
PGAT [41]	-	-	GloVe	61.1	38.1
HQGA [15]	-	-	BERT	61.3	38.6
SSML [54]	HT100M	100M	BERT	-	35.1
CoMVT [16]	HT100M	100M	BERT	-	39.5
ClipBERT [26]	VG, COCO	-	BERT	60.3	37.4
SiaSRea [27]	VG, COCO	-	BERT	60.2	41.6
VQA-T [70]	HT100M	100M	DistilBERT	-	40.4
VQA-T [70]	HTVQA	69M	DistilBERT	-	41.5
MERLOT [28]	YT,CC	185M	BERT	**69.5**	**43.1**
VGT	-	-	BERT	61.6	39.7
VGT (PT)	WV	0.18M	BERT	<u>61.7</u>	39.7
CoVGT	-	-	RoBERTa	61.6	38.3
CoVGT (PT)	WV	0.18M	RoBERTa	<u>61.7</u>	40.0

4.4 Model Analysis

4.4.1 Dynamic Graph Transformer

DGT *vs*. Mean Pooling. We firstly study the DGT module by substituting it with a simple mean-pooling over the region features; the pooled region features are then interacted with the text features and fed to the global transformer. The ablated model does not capture any spatial and temporal communications between the objects in the local video clips. As shown in the middle part of Tab. 9 (w/o DGT), the performances on all datasets drop, especially on those tasks featuring dynamic visual reasoning. For example, the accuracy drops by more than 5% and 2% on TGIF-QA Action and Transition datasets respectively. On NExT-QA and STAR-QA, it drops by 2% and 3% respectively. This experiment evinces the vital role of DGT.

NTrans and ETrans. We then study the effectiveness of temporal graph transformer in DGT by removing both the node and edge transformers defined in Eqn. (11) and (12). Thus, we consider the graphs that are independently constructed at static frames (depicted in Sec. 3.2). The results (w/o GTrans) show that this ablation degenerates the overall accuracy by about 1% on NExT-QA and 2% on STAR-QA though it performs better than removing the whole DGT module.

TABLE 9: Ablation of architecture designs.

Models	TGIF-QA		NExT-QA Val				STAR Val
	Act	Trans	Acc@C	Acc@T	Acc@D	Acc@All	Acc@M
VGT	**95.0**	**97.6**	**52.28**	**55.09**	64.09	**55.02**	**44.27**
w/o DGT	89.6	95.4	50.10	52.85	64.48	53.22	41.15
w/o TTrans	94.0	97.6	50.86	53.04	64.86	53.74	42.37
w/o NTrans	94.5	97.4	50.79	54.22	63.32	53.84	42.86
w/o ETrans	94.8	97.4	51.25	54.34	**64.48**	54.30	43.06
w/o F_I	93.5	97.0	50.44	53.97	63.32	53.58	42.32

TABLE 10: Study of contrastive learning. RBT: RoBERTa.

Models	$\mathcal{L}_{vqa}$	$\mathcal{L}_{vq}$	RBT	NExT-QA	STAR-QA	TGIF-R-Trans†	TGIF-FQA
[CLS]				45.82	41.91	65.9	56.9
VGT	√			55.02	44.27	70.5	61.6
	√	√		57.15	45.84	71.6	61.3
	√		√	58.45	42.44	72.0	60.6
CoVGT	√	√	√	**60.01**	**45.97**	**73.8**	**61.6**

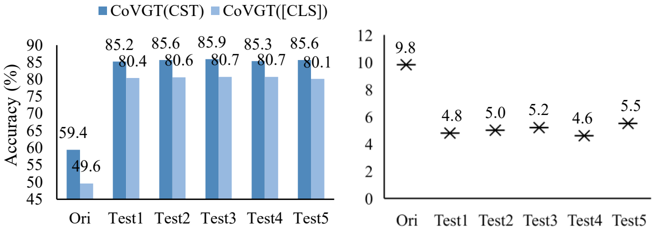
We further study the independent contribution of NTrans and ETrans. We can see that removing any one of them leads to performance drop (VGT *vs*. w/o NTrans, VGT *vs*. w/o ETrans). Also, we find that both transformers help improve the performances separately (w/o GTrans *vs*. w/o NTrans, w/o GTrans *vs*. w/o ETrans) in most tasks. By comparing the results of w/o NTrans and w/o ETrans, we find that the node transformer contributes relatively more to the results. Such difference is reasonable as the update of the node representations will also update the edges.

Finally, the ablation (w/o F_I) in Tab. 9 suggests that F_I complements the object graphs well and contributes steadily to the performances across different datasets.

4.4.2 Contrastive Learning

Contrastive Learning *vs*. Classification. We study a model variant by concatenating the outputs of the DGT module with the token representations from BERT in a way analogous to ClipBERT [26]. The formed text-video representation sequence is fed to a cross-modal transformer for information fusion. Then, the output of the start token (*e.g.* `'[CLS]'` or `'<s>'`) is fed to a 1-way classifier for cross-modal matching in multi-choice QA or a $|\mathcal{A}|$-way classifier in open-ended QA. As shown in the 1st row of Tab. 10, this classification model variant ([CLS]) performs poor. A detailed analysis of the performances (see discussion in Appendix C.2.) indicates that the classification layer results in serious over-fitting problem, especially on NExT-QA which has relatively few training data while the QA contents are complex and diverse. We additionally compare the two kinds of learning strategy on new answer distributions. To this end, we keep unchanged the test questions and their correct answers, but replace the negative answers with randomly sampled answers. Fig. 7(a) suggests that both the classification and contrastively learned models show generalization capability on the newly curated test data. Nonetheless, we find that the performance gap in Fig. 7(b) becomes smaller than it on the original test set. Such difference indicates that the classification model is not good at disambiguating the hard negatives, because the original negative answers are carefully curated to be much more challenging than that of random sampling. These experiments demonstrate the superiority of solving multi-choice QA by contrastive learning over classification.

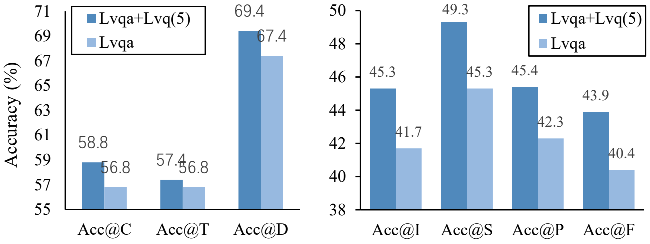
Supervised and Self-supervised Contrastive Learning. Tab. 10 shows that the self-supervised contrastive objective function ($\mathcal{L}_{vq}$) coordinates well with the supervised one ($\mathcal{L}_{vqa}$). It improves the performances on different datasets with different language encoders (BERT [1] by default and RoBERTa [43]) by a clear margin. The detailed results on NExT-QA and STAR-QA (see Fig. 8) show that the improvements are also stable across different tasks. Concretely, the improvements on the causal and temporal reasoning tasks of NExT-QA are more remarkable than that of the descriptive task. Our explanation is that the descriptive questions are relatively simple and involve few visual concepts for cross-modal correspondence learning, *e.g.*, `'what/where is this happening'`. Therefore, the self-supervised objective helps little in that case. All of the questions in STAR-QA are populated from scene graph annotations and challenge visual relation reasoning, so the improvements are significant on all tasks. In particular, we find that the improvements on the rare questions (*i.e.*, questions in the Feasibility groups) are quite impressive though they have few training samples. The improvements, along with our observation of the train and validation accuracy during training, suggest that $\mathcal{L}_{vq}$ can alleviate the over-fitting problem and hence enhance the model's generalization capability.



(a) Performance

(b) Performance gap.

Fig. 7: Analysis of model generalization to different test sets. We randomly sample 5 times to generate 5 different test sets.



(a) NExT-QA

(b) STAR-QA

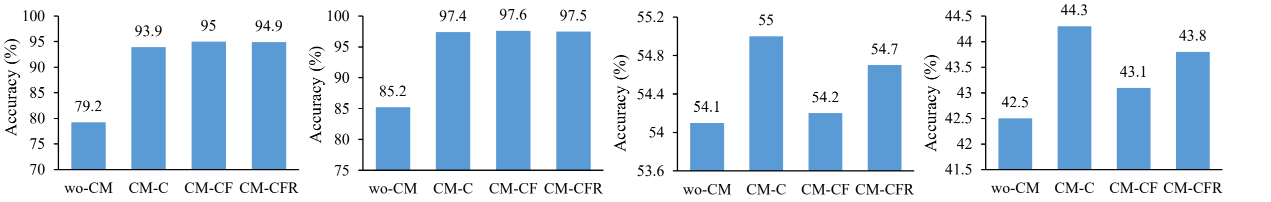
Fig. 8: Comparison of our methods with and without $\mathcal{L}_{vq}$ on specific tasks. The results are based on the language encoder RoBERTa. $\mathcal{L}_{vq}(5)$ means 5 negative questions.

TABLE 11: Study the number of negative questions in $\mathcal{L}_{vq}$.

#Negative samples	0	4	9	14	19
Accuracy (%)	58.45	**60.01**	59.85	59.67	59.81

Negative Sample Mining. We study the number of negative questions in self-supervised contrastive learning ($\mathcal{L}_{vq}$) based on NExT-QA. For supervised learning ($\mathcal{L}_{vqa}$), the number of negative answers is kept the same as the original multiple choices. Tab. 11 shows that the number of negative samples has relatively little impact (*e.g.*, less than 1.0%) on the accuracy. In addition, learning with different number of negative samples can steadily improve over the baseline that does not use video-question correspondence as auxiliary supervision. Intriguingly, we find that the best result is achieved with 4 negative samples. The number is in line with the number of native answers in multi-choice QA.

In Tab. 12, we study the effectiveness of hard negative sampling. Our observations are as follows: 1) The hard negative sampling can steadily improve the performances over a random selection (row 2 and 3 *vs*. row 1). 2) Our method of parsing the questions to obtain the question types works well in experiments.



(a) TGIF-Action (b) TGIF-Trans (c) NExT-QA (d) STAR-QA

Fig. 9: Investigation of cross-modal interaction.

TABLE 12: Study hard negative samples. Random: randomly sample the negatives. PQ: our method by parsing the questions. GT: use the ground-truth type annotations.

Methods	NExT-QA	STAR-QA	Causal-VidQA
Random	59.35	45.40	59.44
Type (PQ)	**60.01**	45.97	59.67
Type (GT)	59.93	**45.98**	**59.97**

TABLE 13: Contrastive pretraining with a different number of negative samples. Results are reported on NExT-QA test set.

#Negative	4	15	31	63	127	255
Zero-shot	31.9	33.1	**34.5**	34.2	34.5	34.3
Fine-tune	55.1	55.3	**55.7**	**55.7**	55.5	55.2

It can achieve equivalent results to the method of using the ground-truth question types (row 2 *vs*. row 3). 3) We surprisingly find that a random selection of the negatives can also contribute to the performance (row 1). Such observation reveals the strength of our contrastive learning.

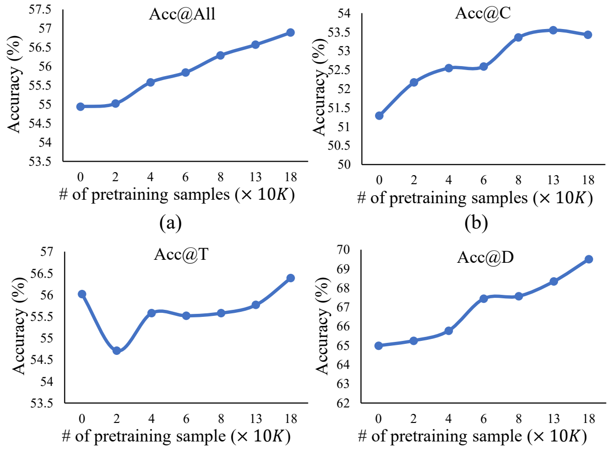
4.4.3 Cross-modal Interaction

In Fig. 9, we investigate several implementation variants of the cross-modal interaction module as depicted in Sec. 3.5. The results suggest that it is better to integrate textual information at both the frame- and clip-level outputs (*i.e*., Eqn. (15) and (16) respectively) for TGIF Action and Transition datasets, while a simple interaction at the clip-level (by default) brings the optimal results on other datasets. Based on these observations, we operate the cross-modal interaction module at both the frame- and clip-level outputs for TGIF Action and Transition, and keep the default implementation (interaction at the clip-level) for other datasets.

Compared with the baselines without cross-modal interaction, all three kinds of interactions help to improve the performance. We notice that the improvement on TGIF is more than 10%. Our explanation is as follows: GIFs are trimmed short videos that only contain the QA-related visual content. The simple video content greatly eases the challenge in spatial-temporal grounding of the positive answer, especially when most of the negative answers do not appear in the short GIFs and can be well distinguished by advanced language models. Therefore, the cross-modal interaction performs more effectively on this dataset. While the video clips in STAR-QA are also trimmed, its multiple choices are carefully curated to include at least one distractor answer that appears in the same video clip. Thus, the improvements are relatively smaller.

4.5 Pretraining and Finetuning

Number of Negative Samples. In Tab. 13, we study pretraining with a different number of negative samples based on VGT. We find that: 1) the number of negative samples mostly affects zero-shot QA performances; when finetuning on the target dataset, the differences become smaller (*e.g*. $< 1\%$); and 2) relatively more negative samples give rise to better results, and the best result is achieved at 31 and 63. Note that the number of negative answers in the target datasets is much smaller (*e.g*. 4 or 5).



(a) (b) (c) (d)

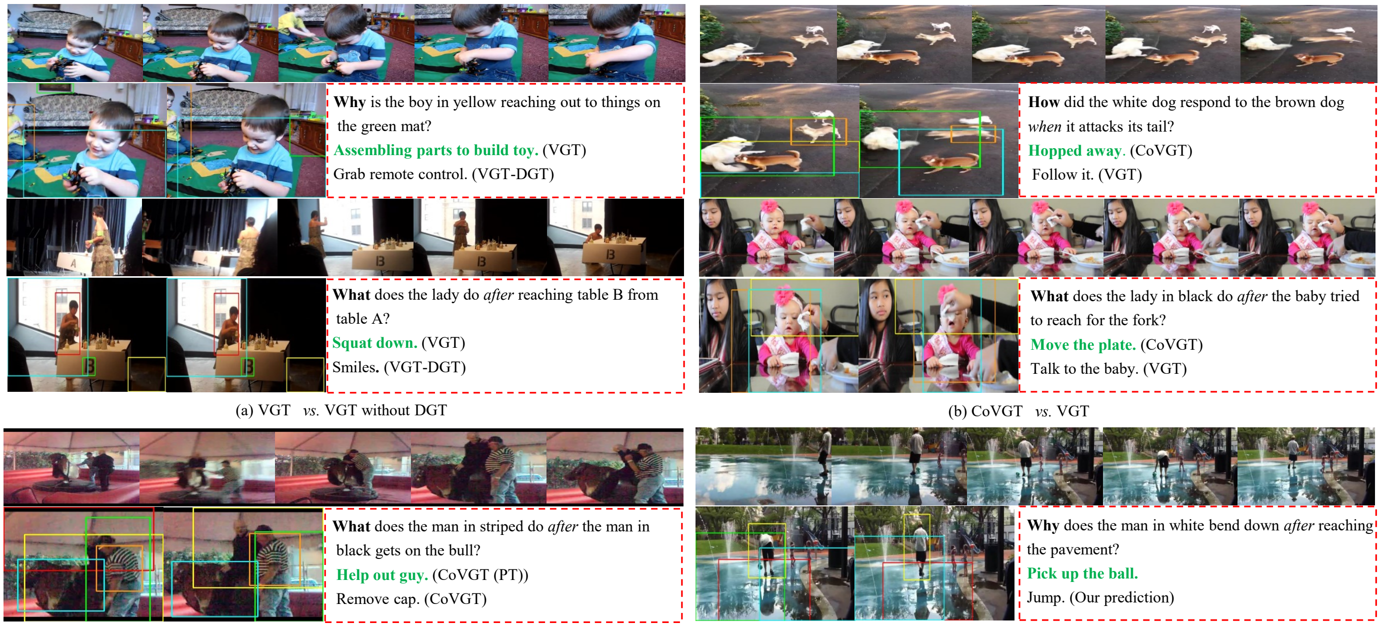
Fig. 10: Pretraining with different amount of data. Results are reported on NExT-QA validation set.

TABLE 14: Study of finetuning the pretrained (PT) weights.

Models	$\mathcal{L}_{vqa}$	$\mathcal{L}_{vq}$	$\mathcal{L}_{\text{MLM}}$	PT	Acc@C	Acc@T	Acc@D	Acc@All
VGT	√				52.28	55.09	64.09	55.02
	√		√		49.41	54.59	64.74	53.46
	√			√	52.28	55.77	69.11	56.02
	√		√	√	**53.43**	**56.39**	**69.50**	**56.89**
CoVGT	√	√		√	58.42	**58.25**	67.82	59.83
	√	√	√	√	**59.69**	58.00	**69.88**	**60.73**

Amount of Pretraining Data. In Fig. 10 we study VGT's performances with different size of pretraining data. Generally, we can see that there is a clear tendency of performance improvements for the overall accuracy (Acc@All) when more data are available. A more detailed analysis shows that these improvements mostly come from a stronger performance in answering causal (Acc@C) and descriptive (Acc@D) questions. It seems that to answer the descriptive questions well, we just need more data for pretraining. However, for answering temporal questions, it demands relatively more data to yield positive effect, or otherwise pretraining helps little and even hurts the performance. This could be because our pretraining data (*i.e*. WebVid) rarely invokes temporal descriptions and the proxy tasks are not tailored for temporal reasoning. Finally, answering casual questions is still the most challenging tasks since the accuracy on causal questions is the lowest among the three tasks categorised in NExT-QA. The observations advocate more future efforts in exploring pretraining to better handle temporal and causal visual reasoning problems.

Fine-tuning. In Tab. 14, we study whether finetuning with masked language modelling (MLM) will also help QA performance. The results show that adding the MLM objective for finetuning the pretrained weights can steadily improve the perfor-


Why is the boy in yellow reaching out to things on the green mat?
Assembling parts to build toy. (VGT)
Grab remote control. (VGT-DGT)
What does the lady do *after* reaching table B from table A?
Squat down. (VGT)
Smiles. (VGT-DGT)
How did the white dog respond to the brown dog *when* it attacks its tail?
Hopped away. (CoVGT)
Follow it. (VGT)
What does the lady in black do *after* the baby tried to reach for the fork?
Move the plate. (CoVGT)
Talk to the baby. (VGT)
What does the man in striped do *after* the man in black gets on the bull?
Help out guy. (CoVGT (PT))
Remove cap. (CoVGT)
Why does the man in white bend down *after* reaching the pavement?
Pick up the ball.
Jump. (Our prediction)

(a) VGT *vs.* VGT without DGT

(b) CoVGT *vs.* VGT

(c) PT CoVGT *vs.* CoVGT

(d) Failure Case

Fig. 11: Visualization of typical predictions on NExT-QA [23]. The ground-truth answers are highlighted in green.

TABLE 15: Comparison of memory and time based on NExT-QA [23]. (2m×8: 2 minutes per epoch and 8 epochs in total.)

Models	Acc(%)	#Params	GPU Memory		Time	
			Train	Infer	Train	GFLOPs
VQA-T [52]	45.30	156.5M	5.6G	2.6G	2m×8	2.5
VGT (DistilBERT)	53.46	90.5M	10.0G	3.5G	5m×7	3.9
VGT (BERT)	55.02	133.7M	16.2G	3.9G	7m×5	7.1
CoVGT (RoBERTa)	60.01	148.9M	19.6G	4.0G	7m×7	12.7

mance for both VGT and CoVGT. Nevertheless, when learning the model from scratch with MLM, such advantage disappears as there is a performance drop on causal and temporal reasoning tasks. The observations indicate that learning with masked language modelling on causal and temporal reasoning tasks demands more data. We also analyze finetuning by adapting the multi-choice QA-pairs to descriptions (see Appendix C.4), so as to reduce the task gap between pretrain and finetune. Our conclusion is that the adaptation method does help improve the performance but the improvement is not stable, and thus we do not use it.

4.6 Model Efficiency

We compare CoVGT with VQA-T in Tab. 15 for better understanding of model efficiency. Experiments are conducted on 1 Tesla V100 GPU with batch size 64 (GFLOPs are based on 1 example). i) **Memory:** CoVGT has comparable training parameters (148.9M *vs*. 156.5M) and model size with VQA-T (568M *vs*. 600M). The RoBERTa encoder in CoVGT takes large portion of the parameters, the vision part is lightweight with only 24M parameters. CoVGT needs more GPU memory for training. Yet, the memory for inference is fairly small and close to that of VQA-T. ii) **Time:** CoVGT's running speed is lower than VQA-T. The results reveal that the lower speed is mainly resulted from the language encoder. However, CoVGT converges much faster and needs much fewer epochs (total FLOPs) to get results superior to VQA-T. For example, on NExT-QA, CoVGT's accuracy at epoch 1 is 50.76% and 55.1% at epoch 2, which already significantly surpass VQA-T's best result (45.30%) achieved at epoch 8. Also, CoVGT's result without pretraining can surpasses that of VQA-T pretrained with million-scale data. In that sense, VGT needs much fewer total FLOPs than VQA-T and other similar pretrained models to achieve better results of visual reasoning.

4.7 Qualitative Analysis

VGT *vs.* VGT-DGT. In Fig. 11(a), the ablated model wrongly answers the 1st question with an atomic action `'grab'` without DGT to weave together a series of *boy-toy* interactions (*e.g.*, `'tough, grab, plug , ...'`) to achieve `'assemble'`. For the 2nd question, prediction like `'squat down'` requires the model to capture the object's temporal state changes, otherwise the model tends to predict the static action `'smiles'` for answer. The examples demonstrate the effectiveness of DGT in modelling the compositions and temporal-dynamics.

CoVGT *vs.* VGT. Fig. 11(b) reveals that CoVGT is able to predict the answers that have never been the correct answers during training (*e.g.* `'hopped away'` and `'move the plate'`). Yet, the words `'hop'` and `'plate'` can be found in the training questions. Such generalization capability of predicting unseen/rare answers mainly thanks to CoVGT's self-contrastive learning strategy between the relevant and irrelevant questions.

PT CoVGT *vs.* CoVGT. Fig. 11(c) suggests that pretraining (PT) helps the model predict the abstract answers, *e.g.* `'help'`. This is understandable since the abstract words often correspond to diverse video contents, and thus demand more data for learning.

Failure Cases. The example in Fig. 11(d) shows a failure case where our model wrongly answers the question with `'jump'`. The visualization of the detection results suggest that the detection model fail to detect the small object, *e.g.* `'ball'`. The case indicates that understanding of the fine-grained object interaction is still challenging and needs more future efforts.

4.8 Limitations and Opportunities

We discuss several limitations and leave them as future efforts. First, we pre-sample and -extract video features offline, which may leave out some key frames and objects that are important for question answering (see our analysis in Sec. 4.7). We believe that an online approach which can take into account more video contents (*e.g.*, at different training iterations) could be helpful for performance improvement. Second, our method benefits from large-scale language models; accordingly it requires more memory and time for inference (see Tab. 15). Therefore, study of light-weight models that are capable of complex video reasoning is a promising direction. Finally, our improvement on open-ended QA (common setting for recognition-based QA) is smaller than that of multi-choice QA (common setting for video reasoning) (Tab. 8 *vs*. other tables in Sec. 4.3), though we have shown its effectiveness for both tasks. Thus, it is interesting to explore more effective approaches for open-ended QA. Our findings in Tab. 8 and Fig. 10(d) suggest that data is essential whereas relation modeling seems less effective. Finally, our experiments are tied to the VideoQA task, it would be interesting to examine our DGT video encoder in other video understanding tasks.

5 Conclusion

This paper introduced a contrastively learned video graph transformer (CoVGT) model for VideoQA in a totally contrastive manner. The model mainly includes: 1) a dynamic visual graph transformer module along with a local-to-global hierarchical architecture for video encoding, 2) separate visual and textual transformers along with a light-weight cross-modal interaction module for cross-modal information encoding, communication and comparison, and 3) joint fully-supervised and self-supervised contrastive objective function for parameter optimization. To validate CoVGT's effectiveness and the contribution of each component, we conducted extensive experiments on a wide range of benchmarks that challenge the various aspects of cross-modal video understanding. The results show that CoVGT can remarkably improve the previous SOTA results on video reasoning tasks and obtain competitive results on descriptive QA tasks. We further demonstrated that our model can also benefit from cross-modal pretraining. Our success suggests that with careful engineering of architectures and learning strategy, one can significantly lower the burden of handling large-scale video data for pretraining, yet achieve comparable or superior performance.

Acknowledgments

This research is supported by the Sea-NExT Joint Lab. The research is also supported by the National Natural Science Foundation of China under grant 61932009.

References

[1] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *NAACL*, 2019.

[2] R. Bommasani, D. A. Hudson, E. Adeli, R. Altman, S. Arora, S. von Arx, M. S. Bernstein, J. Bohg, A. Bosselut, E. Brunskill *et al.*, "On the opportunities and risks of foundation models," *arXiv preprint arXiv:2108.07258*, 2021.

[3] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," in *ICML*. PMLR, 2021, pp. 8748–8763.

[4] J. Lu, D. Batra, D. Parikh, and S. Lee, "Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks," in *NeurIPS*, 2019, pp. 13–23.

[5] W. Su, X. Zhu, Y. Cao, B. Li, L. Lu, F. Wei, and J. Dai, "Vl-bert: Pre-training of generic visual-linguistic representations," in *ICLR*, 2020.

[6] C. Sun, A. Myers, C. Vondrick, K. Murphy, and C. Schmid, "Videobert: A joint model for video and language representation learning," in *ICCV*, 2019, pp. 7464–7473.

[7] J. Li, R. Selvaraju, A. Gotmare, S. Joty, C. Xiong, and S. C. H. Hoi, "Align before fuse: Vision and language representation learning with momentum distillation," in *NeurIPS*, vol. 34, 2021.

[8] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *NeurIPS*, vol. 30, 2017.

[9] Y. Zhong, J. Xiao, W. Ji, Y. Li, W. Deng, and T.-S. Chua, "Video question answering: Datasets, algorithms and challenges," in *EMNLP*, 2022, pp. 6439–6455.

[10] Y. Jang, Y. Song, Y. Yu, Y. Kim, and G. Kim, "Tgif-qa: Toward spatio-temporal reasoning in visual question answering," in *CVPR*, 2017, pp. 2758–2766.

[11] C. Fan, X. Zhang, S. Zhang, W. Wang, C. Zhang, and H. Huang, "Heterogeneous memory enhanced multimodal attention model for video question answering," in *CVPR*, 2019, pp. 1999–2007.

[12] J. Gao, R. Ge, K. Chen, and R. Nevatia, "Motion-appearance co-memory networks for video question answering," in *CVPR*, 2018, pp. 6576–6585.

[13] P. Jiang and Y. Han, "Reasoning with heterogeneous graph alignment for video question answering," in *AAAI*, 2020.

[14] D. Huang, P. Chen, R. Zeng, Q. Du, M. Tan, and C. Gan, "Location-aware graph convolutional networks for video question answering," in *AAAI*, vol. 34, no. 07, 2020, pp. 11 021–11 028.

[15] J. Xiao, A. Yao, Z. Liu, Y. Li, W. Ji, and T.-S. Chua, "Video as conditional graph hierarchy for multi-granular question answering," in *AAAI*, 2022, pp. 2804–2812.

[16] P. H. Seo, A. Nagrani, and C. Schmid, "Look before you speak: Visually contextualized utterances," in *CVPR*, 2021, pp. 16 877–16 887.

[17] D. Xu, Z. Zhao, J. Xiao, F. Wu, H. Zhang, X. He, and Y. Zhuang, "Video question answering via gradually refined attention over appearance and motion," in *ACM MM*, 2017, pp. 1645–1653.

[18] H. Xu, G. Ghosh, P.-Y. Huang, D. Okhonko, A. Aghajanyan, F. Metze, L. Zettlemoyer, and C. Feichtenhofer, "Videoclip: Contrastive pre-training for zero-shot video-text understanding," in *EMNLP*, 2021, pp. 6787–6800.

[19] A. Yang, A. Miech, J. Sivic, I. Laptev, and C. Schmid, "Learning to answer visual questions from web videos," in *arXiv preprint arXiv:2205.05019*, 2022.

[20] Y. Yu, J. Kim, and G. Kim, "A joint sequence fusion model for video question answering and retrieval," in *ECCV*, September 2018.

[21] L. Zhu and Y. Yang, "Actbert: Learning global-local video-text representations," in *CVPR*, 2020, pp. 8746–8755.

[22] X. Shang, D. Di, J. Xiao, Y. Cao, X. Yang, and T.-S. Chua, "Annotating objects and relations in user-generated videos," in *ICMR*, 2019, pp. 279–287.

[23] J. Xiao, X. Shang, A. Yao, and T.-S. Chua, "Next-qa: Next phase of question-answering to explaining temporal actions," in *CVPR*, 2021, pp. 9777–9786.

[24] B. Wu, S. Yu, Z. Chen, J. B. Tenenbaum, and C. Gan, "Star: A benchmark for situated reasoning in real-world videos," in *NeurIPS*, 2021.

[25] J. Li, L. Niu, and L. Zhang, "From representation to reasoning: Towards both evidence and commonsense reasoning for video question-answering," in *CVPR*, 2022, pp. 21 273–21 282.

[26] J. Lei, L. Li, L. Zhou, Z. Gan, T. L. Berg, M. Bansal, and J. Liu, "Less is more: Clipbert for video-and-language learning via sparse sampling," in *CVPR*, 2021, pp. 7331–7341.

[27] W. Yu, H. Zheng, M. Li, L. Ji, L. Wu, N. Xiao, and N. Duan, "Learning from inside: Self-driven siamese sampling and reasoning for video question answering," *NeurIPS*, vol. 34, 2021.

[28] R. Zellers, X. Lu, J. Hessel, Y. Yu, J. S. Park, J. Cao, A. Farhadi, and Y. Choi, "Merlot: Multimodal neural script knowledge models," in *NeurIPS*, vol. 34, 2021.

[29] T.-J. Fu, L. Li, Z. Gan, K. Lin, W. Y. Wang, L. Wang, and Z. Liu, "Violet: End-to-end video-language transformers with masked visual-token modeling," in *arXiv preprint arXiv:2111.12681*, November 2021.

[30] S. Buch, C. Eyzaguirre, A. Gaidon, J. Wu, L. Fei-Fei, and J. C. Niebles, "Revisiting the" video" in video-language understanding," in *CVPR*, 2022, pp. 2917–2927.

[31] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, 2016, pp. 770–778.

[32] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," *NeurIPS*, vol. 28, 2015.

[33] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*,

"An image is worth 16x16 words: Transformers for image recognition at scale," in *ICLR*, 2020.

[34] G. Bertasius, H. Wang, and L. Torresani, "Is space-time attention all you need for video understanding?" in *ICML*. PMLR, 2021, pp. 813–824.

[35] Z. Liu, J. Ning, Y. Cao, Y. Wei, Z. Zhang, S. Lin, and H. Hu, "Video swin transformer," *arXiv preprint arXiv:2106.13230*, 2021.

[36] S. Xie, C. Sun, J. Huang, Z. Tu, and K. Murphy, "Rethinking spatiotemporal feature learning: Speed-accuracy trade-offs in video classification," in *ECCV*, 2018, pp. 305–321.

[37] V. Agarwal, R. Shetty, and M. Fritz, "Towards causal vqa: Revealing and reducing spurious correlations by invariant and covariant semantic editing," in *CVPR*, 2020, pp. 9690–9698.

[38] Y. Niu, K. Tang, H. Zhang, Z. Lu, X.-S. Hua, and J.-R. Wen, "Counterfactual vqa: A cause-effect look at language bias," in *CVPR*, 2021, pp. 12 700–12 710.

[39] Y. Li, X. Wang, J. Xiao, W. Ji, and T.-S. Chua, "Invariant grounding for video question answering," in *CVPR*, 2022, pp. 2928–2937.

[40] Y. Li, X. Wang, J. Xiao, and T.-S. Chua, "Equivariant and invariant grounding for video question answering," in *Proceedings of the 30th ACM International Conference on Multimedia*, 2022, p. 4714–4722.

[41] L. Peng, S. Yang, Y. Bin, and G. Wang, "Progressive graph attention network for video question answering," in *ACM MM*, 2021, pp. 2871–2879.

[42] J. Xiao, P. Zhou, T.-S. Chua, and S. Yan, "Video graph transformer for video question answering," in *ECCV*. Springer, 2022, pp. 39–58.

[43] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," *arXiv preprint arXiv:1907.11692*, 2019.

[44] X. Li, J. Song, L. Gao, X. Liu, W. Huang, X. He, and C. Gan, "Beyond rnns: Positional self-attention with co-attention for video question answering," in *AAAI*, 2019, pp. 8658–8665.

[45] J. Jiang, Z. Chen, H. Lin, X. Zhao, and Y. Gao, "Divide and conquer: Question-guided spatio-temporal contextual attention for video question answering," in *AAAI*, vol. 34, no. 07, 2020, pp. 11 101–11 108.

[46] F. Liu, J. Liu, W. Wang, and H. Lu, "Hair: Hierarchical visual-semantic relational reasoning for video question answering," in *ICCV*, October 2021, pp. 1698–1707.

[47] J. Park, J. Lee, and K. Sohn, "Bridge to answer: Structure-aware graph interaction network for video question answering," in *CVPR*, 2021, pp. 15 526–15 535.

[48] L. H. Dang, T. M. Le, V. Le, and T. Tran, "Hierarchical object-oriented spatio-temporal reasoning for video question answering," in *IJCAI*, August 2021.

[49] A. Seo, G.-C. Kang, J. Park, and B.-T. Zhang, "Attend what you need: Motion-appearance synergistic networks for video question answering," in *ACL*, 2021, pp. 6167–6177.

[50] J. Xiao, X. Shang, X. Yang, S. Tang, and T.-S. Chua, "Visual relation grounding in videos," in *ECCV*. Springer, 2020, pp. 447–464.

[51] S. Xiao, L. Chen, K. Gao, Z. Wang, Y. Yang, and J. Xiao, "Rethinking multi-modal alignment in video question answering from feature and sample perspectives," *arXiv preprint arXiv:2204.11544*, 2022.

[52] A. Yang, A. Miech, J. Sivic, I. Laptev, and C. Schmid, "Just ask: Learning to answer questions from millions of narrated videos," in *ICCV*, 2021, pp. 1686–1697.

[53] A. Miech, D. Zhukov, J.-B. Alayrac, M. Tapaswi, I. Laptev, and J. Sivic, "Howto100m: Learning a text-video embedding by watching hundred million narrated video clips," in *ICCV*, 2019, pp. 2630–2640.

[54] E. Amrani, R. Ben-Ari, D. Rotman, and A. Bronstein, "Noise estimation using density estimation for self-supervised multimodal learning," in *AAAI*, vol. 35, no. 8, 2021, pp. 6644–6652.

[55] A. Miech, J.-B. Alayrac, L. Smaira, I. Laptev, J. Sivic, and A. Zisserman, "End-to-end learning of visual representations from uncurated instructional videos," in *CVPR*, 2020, pp. 9879–9889.

[56] M. Bain, A. Nagrani, G. Varol, and A. Zisserman, "Frozen in time: A joint video and image encoder for end-to-end retrieval," in *ICCV*, 2021, pp. 1728–1738.

[57] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *ECCV*. Springer, 2014, pp. 740–755.

[58] R. Krishna, Y. Zhu, O. Groth, J. Johnson, K. Hata, J. Kravitz, S. Chen, Y. Kalantidis, L.-J. Li, D. A. Shamma *et al.*, "Visual genome: Connecting language and vision using crowdsourced dense image annotations," *IJCV*, vol. 123, no. 1, pp. 32–73, 2017.

[59] P. Sharma, N. Ding, S. Goodman, and R. Soricut, "Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning," in *ACL*, 2018, pp. 2556–2565.

[60] P. W. Battaglia, J. B. Hamrick, V. Bapst, A. Sanchez-Gonzalez, V. Zambaldi, M. Malinowski, A. Tacchetti, D. Raposo, A. Santoro, R. Faulkner *et al.*, "Relational inductive biases, deep learning, and graph networks," *arXiv preprint arXiv:1806.01261*, 2018.

[61] D. Kreuzer, D. Beaini, W. Hamilton, V. Létourneau, and P. Tossou, "Rethinking graph transformers with spectral attention," *NeurIPS*, vol. 34, 2021.

[62] L. Wang, X. Chang, S. Li, Y. Chu, H. Li, W. Zhang, X. He, L. Song, J. Zhou, and H. Yang, "Tcl: Transformer-based dynamic graph modelling via contrastive learning," *arXiv preprint arXiv:2105.07944*, 2021.

[63] C. Ying, T. Cai, S. Luo, S. Zheng, G. Ke, D. He, Y. Shen, and T.-Y. Liu, "Do transformers really perform badly for graph representation?" *NeurIPS*, vol. 34, 2021.

[64] S. Yun, M. Jeong, R. Kim, J. Kang, and H. J. Kim, "Graph transformer networks," *NeurIPS*, vol. 32, 2019.

[65] A. Cherian, C. Hori, T. K. Marks, and J. Le Roux, "(2.5+ 1) d spatio-temporal scene graphs for video question answering," in *AAAI*, vol. 36, no. 1, 2022, pp. 444–453.

[66] S. Geng, P. Gao, M. Chatterjee, C. Hori, J. Le Roux, Y. Zhang, H. Li, and A. Cherian, "Dynamic graph representation learning for video dialog via multi-modal shuffled transformers," in *AAAI*, 2021.

[67] S. Kim, S. Jeong, E. Kim, I. Kang, and N. Kwak, "Self-supervised pre-training and contrastive representation learning for multiple-choice video qa," in *AAAI*, vol. 35, no. 14, 2021, pp. 13 171–13 179.

[68] Z. Liang, W. Jiang, H. Hu, and J. Zhu, "Learning to contrast the counterfactual samples for robust visual question answering," in *EMNLP*, 2020, pp. 3285–3292.

[69] Y. Kant, A. Moudgil, D. Batra, D. Parikh, and H. Agrawal, "Contrast and classify: Training robust vqa models," in *ICCV*, 2021, pp. 1604–1613.

[70] A. Yang, A. Miech, J. Sivic, I. Laptev, and C. Schmid, "Just ask: Learning to answer questions from millions of narrated videos," in *ICCV*, 2021, pp. 1686–1697.

[71] A. v. d. Oord, Y. Li, and O. Vinyals, "Representation learning with contrastive predictive coding," *arXiv preprint arXiv:1807.03748*, 2018.

[72] Y.-H. H. Tsai, S. Divvala, L.-P. Morency, R. Salakhutdinov, and A. Farhadi, "Video relationship reasoning using gated spatio-temporal energy graph," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019, pp. 10 424–10 433.

[73] P. Anderson, X. He, C. Buehler, D. Teney, M. Johnson, S. Gould, and L. Zhang, "Bottom-up and top-down attention for image captioning and visual question answering," in *CVPR*, 2018, pp. 6077–6086.

[74] R. Krishna, I. Chami, M. Bernstein, and L. Fei-Fei, "Referring relationships," in *CVPR*, 2018, pp. 6867–6876.

[75] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *ICLR*, 2017.

[76] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "Imagenet large scale visual recognition challenge," *IJCV*, vol. 115, no. 3, pp. 211–252, 2015.

[77] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *ICLR*, 2015.

[78] K. Yi, C. Gan, Y. Li, P. Kohli, J. Wu, A. Torralba, and J. B. Tenenbaum, "Clevrer: Collision events for video representation and reasoning," *ICLR*, 2020.

[79] D. Ding, F. Hill, A. Santoro, M. Reynolds, and M. Botvinick, "Attention over learned object embeddings enables complex visual reasoning," *NeurIPS*, vol. 34, 2021.

[80] L. H. Li, M. Yatskar, D. Yin, C.-J. Hsieh, and K.-W. Chang, "Visualbert: A simple and performant baseline for vision and language," *arXiv preprint arXiv:1908.03557*, 2019.

[81] T. M. Le, V. Le, S. Venkatesh, and T. Tran, "Hierarchical conditional relation networks for video question answering," in *CVPR*, 2020, pp. 9972–9981.

[82] M. Peng, C. Wang, Y. Gao, Y. Shi, and X.-D. Zhou, "Multilevel hierarchical network with multiscale sampling for video question answering," *IJCAI*, 2022.

[83] X. Shang, J. Xiao, D. Di, and T.-S. Chua, "Relation understanding in videos: A grand challenge overview," in *Proceedings of the 27th ACM International Conference on Multimedia*, 2019, pp. 2652–2656.

[84] J. Ji, R. Krishna, L. Fei-Fei, and J. C. Niebles, "Action genome: Actions as compositions of spatio-temporal scene graphs," in *CVPR*, 2020, pp. 10 236–10 247.

[85] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. L. Zitnick, and D. Parikh, "Vqa: Visual question answering," in *ICCV*, 2015, pp. 2425–2433.

[86] J. Xu, T. Mei, T. Yao, and Y. Rui, "Msr-vtt: A large video description dataset for bridging video and language," in *CVPR*, 2016, pp. 5288–5296.

Junbin Xiao has completed his PhD and now serves as a Research Fellow at the Department of Computer Science (CS), National University of Singapore (NUS). Before that, he received his M.S.Eng degree from the Institute of Computing Technology (ICT), Chinese Academy of Sciences (CAS) and B.E degree from Sichuan University (SCU). His research focuses on visual relation oriented VideoQA. He has published relevant papers in the top-tier conferences: CVPR, ECCV, AAAI, ACM MM and EMNLP. He also serves as reviewer for: CVPR, ICCV, ECCV, AAAI, TMM, ToMM and TNNLS. He co-organized and served as program committee member for the video relation understanding challenge on MM'19 and MM'20.

Pan Zhou is currently a senior Research Scientist at Sea AI Lab (SAIL) of Sea group. Before that, he worked in Salesforce as a research scientist. He completed his Ph.D. at National University of Singapore (NUS) and Master at Peking University. His research interests include deep learning theory and applications, noncovex/convex optimization. He has published papers in ICLR, ICML, NeurIPS, CVPR, ICCV, ECCV, AAAI, IJCAI and journals: TPAMI, TIP. He serves as reviewer for top conferences: ICML, NeurIPS, CVPR, ICCV, AAAI and journals: TPAMI, IJCV, TIP, TNNLS and TCSVT. He is awarded the Microsoft Research Asia Fellowship.

Angela Yao is currently an assistant professor and leads the Computer Vision and Machine Learning (CVML) group at National University of Singapore. She received her Ph.D degree from ETH and BASc degree from University of Toronto, Canada. Her research interests include video understanding, 3D pose estimation, activity recognition, sequence and time series modelling. Her research lies at the intersection of computer vision, machine learning, and human-computer interaction, with a particular emphasis on developing models and algorithms that can understand human actions and interactions in video data. She serves as Area Chair and Reviewer for ICLR, ICML, NeurIPS, CVPR, ICCV, IJCAI and Program Chair for ICCV, ECCV and 3DGV. She once led the Visual Computing Group at the University of Bonn, Germany and co-founded a startup on smart parking in Zurich, Switzerland.

Yicong Li is currently a Ph.D. candidate at the Institute of Data Science, National University of Singapore. Prior to that, he received a bachelor's degree and a Master's degree from Huazhong University of Science and Technology and Columbia University, respectively. In 2020, he started his Ph.D. study at the National University of Singapore with a research focus on multimodal learning. His publications include some top conferences such as CVPR, ACM MM, AAAI and EMNLP. He serves as reviewer for CVPR.

Richang Hong is the Professor and Executive Dean at School of Computer and Informatics, Hefei University of Technology, Hefei, China. He received his Ph.D. degree from the University of Science and Technology of China (USTC). He worked as a research fellow at National University of Singapore (NUS). His current research interests include multimedia, language and vision and social media. He has authored over 200 journal and conference papers in these areas and the Google Scholar citations for those papers is more than 16000. He served as editor of the IEEE TCSVT, IEEE TMM, IEEE TCSS, IEEE TBD, ACM TOMM, NPL and the guest editors of several international journals, a steering committee member of MMM (international conference on multimedia modeling) conference series since 2019, and the technical program chairs of PCM'2018, etc. He also served as area chairs of ACM Multimedia, SIGIR etc. since 2016 and a technical program committee member of over 20 prestigious international conferences, and a reviewer of over 20 prestigious international journals. He is a recipient of the Best Paper Award in ACM Multimedia 2010, Best Paper Award in ACM ICMR 2015 and Best Paper Honorable Mention Award of IEEE trans. Multimedia 2015. Dr. Hong joined CCF, CSIG, CAAI, IEEE and ACM. He is the deputy director of multimedia technical committee of CSIG and the secretary of the ACM SIGMM China Chapter.

Shuicheng Yan (Fellow, IEEE) is currently a visiting professor at BAAI, Beijing, China. Previously, he was the director of Sea AI Lab (SAIL) and Group Chief Scientist of Sea. He is an Fellow of Academy of Engineering, Singapore, IEEE Fellow, ACM Fellow, AAAI Fellow and IAPR Fellow. His research areas include computer vision, machine learning and multimedia analysis. Till now, he has published over 600 papers in top international journals and conferences, with Google Scholar Citation over 90,000 times and H-index 135. He had been among "Thomson Reuters Highly Cited Researchers" in 2014, 2015, 2016, 2018, 2019. Dr. Yan's team has received winner or honorable-mention prizes for 10 times of two core competitions, Pascal VOC and ImageNet (ILSVRC), which are deemed as "World Cup" in the computer vision community. Also his team won over 10 best paper or best student paper prizes and especially, a grand slam in ACM MM, the top conference in multimedia, including Best Paper Award, Best Student Paper Award and Best Demo Award.

Tat-Seng Chua is the KITHCT Chair Professor at the School of Computing, National University of Singapore (NUS). He is also the Distinguished Visiting Professor of Tsinghua University, the Visiting Pao Yue-Kong Chair Professor of Zhejiang University, and the Distinguished Visiting Professor of Sichuan University. Dr. Chua was the Founding Dean of the School of Computing from 1998-2000. His main research interests include unstructured data analytics, video analytics, conversational search and recommendation, and robust and trustable AI. He is the Co-Director of NExT, a joint research Center between NUS and Tsinghua University, and Sea-NExT, a joint Lab between Sea Group and NExT. Dr. Chua is the recipient of the 2015 ACM SIGMM Achievements Award, and the winner of the 2022 NUS Research Recognition Award. He is the Chair of steering committee of Multimedia Modeling (MMM) conference series, and ACM International Conference on Multimedia Retrieval (ICMR) (2015-2018). He is the General Co-Chair of ACM Multimedia 2005, ACM SIGIR 2008, ACM Web Science 2015, ACM MM-Asia 2020, and the upcoming ACM conferences on WSDM 2023 and TheWebConf 2024. He serves in the editorial boards of three international journals. Dr. Chua is the co-Founder of two technology startup companies in Singapore. He holds a PhD from the University of Leeds, UK.

This appendix gives additional introduction to the paper ”Contrastive Video Question Answering via Video Graph Transformer”. It includes three major parts: A. more information of the experimented datasets, B. the implementation details, and C. more result discussion.

Appendix A Datasets

We briefly introduce the particulars of each dataset as follows: **NExT-QA** [23] is a manually annotated dataset that benchmarks the causal and temporal object interaction reasoning. Its videos are sourced from VidOR [22], [83] and cover various daily activities. By default, each question has 5 options with 1 correct answer and 4 distractor answers.

TGIF-QA [10] challenges repeating action recognition and temporal state transition. The videos are short GIFs and are trimmed to contain only the content of interested for the paired questions. It provides 5 options for each question and requires the models to pick the correct one. **TGIF-QA-R** [41] derives from TGIF-QA action and state transition tasks by fixing the answer bias issue. Therefore, It shares the same videos, questions and correct answers with TGIF-QA. In our experiment, we further remove the redundant candidate answers in TGIF-QA-R for better evaluation.

STAR-QA [24] benchmarks situated visual reasoning. It is based on Action Genome [84] to curate questions and answers that verify a wide range of reasoning capabilities about human-object interaction, temporal sequence analysis, action prediction, and feasibility inference. Its videos feature single person indoor activities.

Causal-VidQA [25] goes beyond visual evidence reasoning to study visual commonsense in videos. It sets four type of questions: Description, Explanation, Prediction and Counterfactual ones. The tasks are defined as multi-choice selection. In particular, a correct prediction for the questions in prediction and counterfactual require *both* the answers and the corresponding reasons to match the ground-truth ones.

TGIF-FrameQA is a sub-task defined in the TGIF-QA dataset [10]. It mimics ImageQA [85] by posing questions that invoke a single frame for answer. The QA pairs in **MSRVTT-QA** [17] are automatically generated from video descriptions [86]. The two datasets focus on the recognition of the video objects/attributes/actions/activities; their questions rarely invoke temporal relations.

Appendix B Implementation Details

For experiments on STAR [24], we obtain the video segments associated with each QA pair according to the given time stamps; the segments are then treated as independent videos for processing. Particularly, we only use the QA annotations for training following the standard in video question answering. In addition, the final submission to the evaluation server is obtained by training with both the train and validation data following the original paper [24]. For Causal-VidQA, we process the videos in the same way as other datasets and ground-truth visual object annotations are not used. For TGIF-Action and Transition tasks, we find that adding randomness to the multiple choices in self-supervised contrastive learning will hurt the performance, and thus we do not use it.

For contrastive learning on target datasets, we obtain the hard negative answers and questions according to the question types. While the question types maybe provided in some datasets, we simply obtain them according to the starting three question words[2] for adaptability. Moreover, we use the same rule for all datasets. We find that this works well though the obtained question types may not be perfect. For specific training, we first train the whole model end to end, and then freeze the language model to fine-tune the other parts of the best model obtained at the $1st$ stage. The best results obtained in the two stages are determined as final results. For pretraining with weakly-paired video-text data [56], we preprocess the videos in the same way as for QA (*i.e.* samples 8 clips and 10 regions per frame) and pretrain the model with an initial learning rate of 5×10^{-5} and batch size 64. Besides, a text token is corrupted at a probability of 15% in masked language modelling. Following [43], [52], a corrupted token will be replaced with 1) the ‘[MASK]’ token by a chance of 80%, 2) a random token by a chance of 10%, and 3) the same token by a chance of 10%. We train the model by maximal 2 epochs which gives to the best generalization results. Our pretraining costs about 2 hours on 4 Tesla V100 GPUs.

Appendix C Result Analysis

C.1 Results Per Question Type on NExT-QA

We compare CoVGT with the recent methods that have reported accuracy per question type on NExT-QA. As shown in Tab. 16, our methods outperforms the competitors significantly in answering all type of questions except for those in the counting group. The weaker performance on counting could be majorly due to our sparse sampling, *e.g.* 32 frames per video and 5 regions per frame. In addition, the smoothing effect of attention (either Transformer or GNN) could also jeopardize the counting performance.

C.2 Contrastive Learning

Contrastive Learning *vs*. Classification. To further study whether the poor performance of the classification model variant (Sec. 4.4.2) is caused by the cross-modal transformer or the classification layer. We keep our cross-modal interaction mechanism (introduced in Sec. 3.5) and discard the cross-modal transformer. Thus, we directly send the cross-modal interacted video representation f^{qv} to a classification layer for multi-choice classification. There are $|\mathcal{A}_{mc}|$ such video representations obtained by interacting the video with different question-answer pairs, and each is mapped to a scalar that gives the probability of the candidate answer to be a correct one. The results in Tab. 17 show that this implementation variant (CM) wins slightly on the cross-modal transformer (CMTrans). The small improvements could be attributed to that our cross-modal interaction do not have self-attention weights (single modality) which usually take large portion of the attention distribution in cross-modal transformers. Thus, our cross-modal interaction focuses more on cross-modal information exchange, and thus benefits the final results. However, the small improvement does not change the fact that a classification setting is sub-optimal for VideoQA, at least in the pretraining-free setting together with advanced language models.

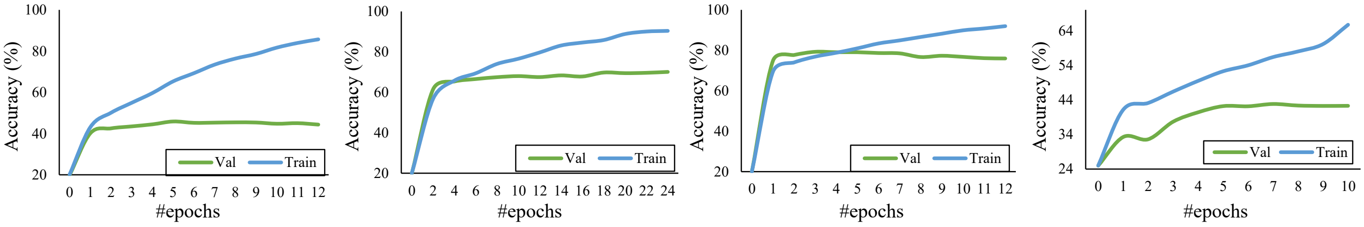
C.3 Study of QA Short-Cut in Open-ended QA

Tab. 18 shows that the QA short-cut in Eqn. (22) does contribute to the performance. The results suggest that we can take advantage of the language biases for better performance on the test data.

[2] why, what, where, which, who, how (many)/(times) + is(are)/does(do).

TABLE 16: Results per question type on NExT-QA val set.

Methods	Acc@C			Acc@T			Acc@D				Acc@All
	Why (36%)	How (12%)	Overall (48%)	Prev&Next (16%)	Present (13%)	Overall (29%)	Count (4%)	Location (5%)	Other (7%)	Overall (16%)	
HGA [13]	46.99	44.22	46.26	49.53	52.49	50.74	44.07	72.54	55.41	59.33	49.74
TrajG [51]	52.81	47.44	51.40	51.11	53.70	52.17	46.89	75.25	58.03	62.03	53.30
P3D-G [65]	52.39	48.36	51.33	50.91	54.28	52.30	46.02	77.08	58.31	62.58	53.40
CoVGT	59.77	56.08	58.80	**56.16**	59.28	57.44	**50.85**	**82.71**	67.21	69.37	60.01
CoVGT(PT)	**60.65**	**56.95**	**59.69**	56.06	**60.78**	**58.00**	47.46	82.37	**70.82**	**69.88**	**60.73**



(a) NExT-QA

(b) TGIF-QA (Action)

(c) TGIF-QA (Trans)

(d) STAR-QA

Fig. 12: Accuracy with regard to the training epochs.

TABLE 17: A detailed study of the classification model variants.

Models	NExT-QA Val			
	Acc@C	Acc@T	Acc@D	Acc@All
CMTrans ([CLS])	42.96	46.96	53.02	45.82
CM (CLS)	**44.46**	**47.33**	**54.70**	**46.98**

TABLE 18: Study of the QA short-cut prediction.

Methods	TGIF-FrameQA	MSRVTT-QA
VGT	**61.6**	**39.7**
VGT w/o QA	61.2	39.1

C.4 Study of Finetuning

QA Adaptation. We additionally study fine-tuning VGT by converting the QA-pairs into descriptions so as to reduce the data difference between pretrain and finetune. In our implementation, we convert the question whose pattern is clear and easy to adapt, otherwise we directly concatenate the answer behind the question with a special token '[SEP]'. For instance,`"why is the baby crying? fell backwards"` to `"the baby crying [SEP] fell backwards"`, and `"what did the man do after squatting down? wash hands."` to `"the man [SEP] wash hands [SEP] after squatting down"`. Our results in Tab. 14 (Adapt All) show that such adaptation benefits the performances on causal questions but hurts the performance of others. We speculate the main reason is that the converted descriptions of a given sample are quite similar for temporal and descriptive questions since the candidate answers are short but they share the same long question. As a result, the descriptions render it hard to disambiguate between the correct and incorrect answers. Based on such observation, a better alternative is to only adapt the questions in the causal group. We can see that the overall performance improves on the validation set (Adapt C). Nevertheless, we find that model does not generalize well to the test set, *i.e.*, 55.08% which is worse than 55.7% obtained by the model without considering adaption. As the experiment is not our focus, we stop here and hope that our pioneer attempt can spark more effective and interesting works.

TABLE 19: Study of fine-tuning.

Methods	NExT-QA Val			
	Acc@C	Acc@T	Acc@D	Acc@All
FT (baseline)	53.43	**56.39**	69.50	56.89
FT Adapt All	**54.89**	54.40	67.57	56.69
FT Adapt C	**54.89**	55.15	**69.76**	**57.29**

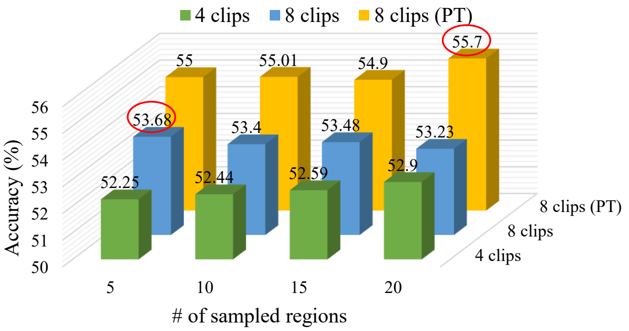
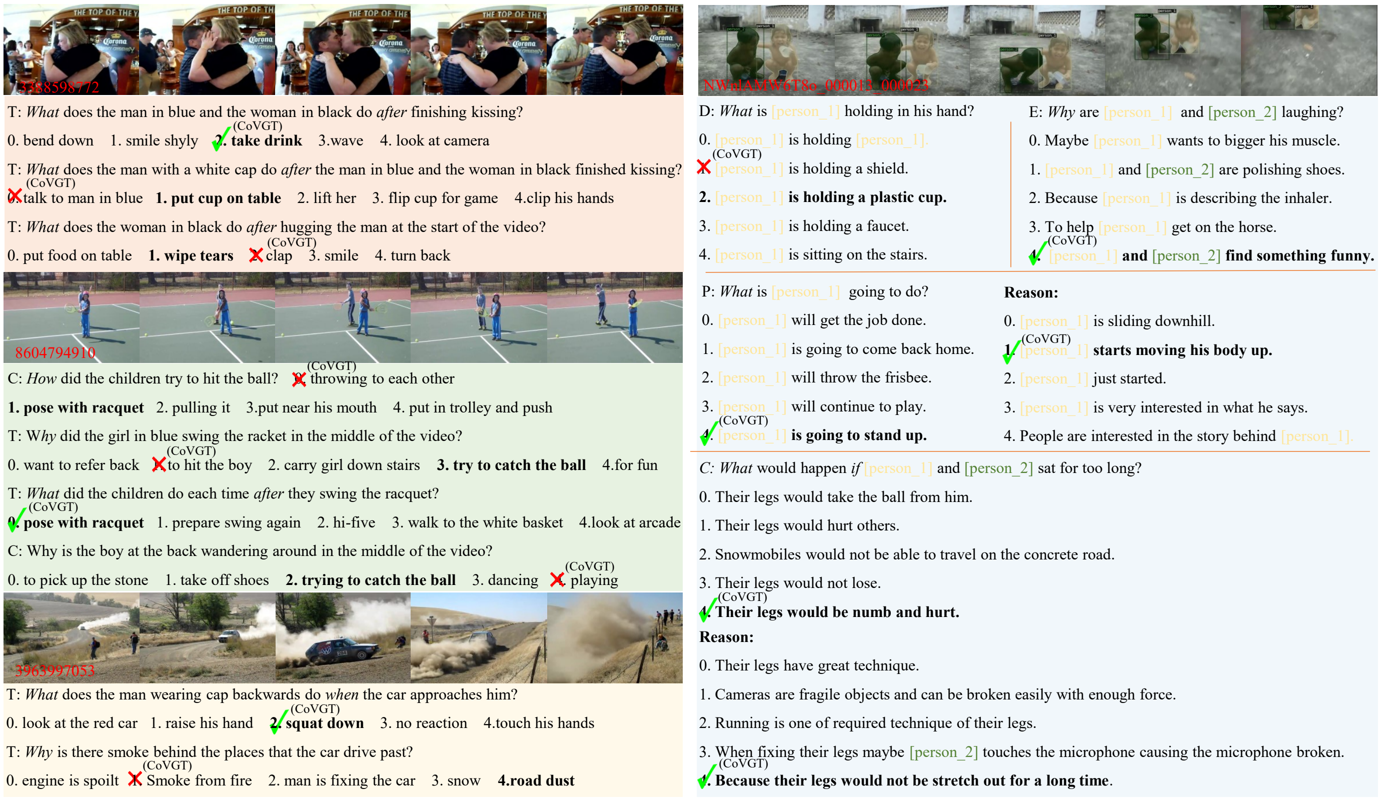


Fig. 13: Investigation of sampled video clips and region proposals per frame. Results are reported on NExT-QA test set.

C.5 Video Sampling

In Fig. 13, we study the effect of sampled video clips and region proposals on NExT-QA [23] based on the VGT model. Regarding the number of sampled video clips, we find that the setting of 8 clips steadily wins on 4 clips. This is understandable as the videos in NExT-QA are relatively long. As for the sampled regions, when learning the model from scratch, the setting of 5 regions gives relatively better result, *e.g.*, 53.68%. Nonetheless, when pretraining are considered, the setting of 20 regions gives better result, *e.g.*, 55.70%. Such difference could be due to that learning with more regions can yield over-fitting issues when the dataset is not large enough, since the constructed graph become much larger and more complex. Our speculation is also supported by the observation that the accuracy increases with the number of regions when we only sample 4 video clips (less graph nodes). Based on the observations, we use 5 regions in the pretraining-free experiments and 10 regions in the *pretrain and finetune* experiments for CoVGT to balance between accuracy and efficiency.



T: *What* does the man in blue and the woman in black do *after* finishing kissing?

0. bend down 1. smile shyly ✓(CoVGT) 2. **take drink** 3.wave 4. look at camera

T: *What* does the man with a white cap do *after* the man in blue and the woman in black finished kissing?

✗(CoVGT) 0. talk to man in blue **1. put cup on table** 2. lift her 3. flip cup for game 4.clip his hands

T: *What* does the woman in black do *after* hugging the man at the start of the video?

0. put food on table **1. wipe tears** ✗(CoVGT) 2. clap 3. smile 4. turn back

C: *How* did the children try to hit the ball? ✗(CoVGT) 0. throwing to each other

1. pose with racquet 2. pulling it 3.put near his mouth 4. put in trolley and push

T: W*hy* did the girl in blue swing the racket in the middle of the video?

0. want to refer back ✗(CoVGT) 1. to hit the boy 2. carry girl down stairs **3. try to catch the ball** 4.for fun

T: *What* did the children do each time *after* they swing the racquet?

✓(CoVGT) **0. pose with racquet** 1. prepare swing again 2. hi-five 3. walk to the white basket 4.look at arcade

C: Why is the boy at the back wandering around in the middle of the video?

0. to pick up the stone 1. take off shoes **2. trying to catch the ball** 3. dancing ✗(CoVGT) 4. playing

T: *What* does the man wearing cap backwards do *when* the car approaches him?

0. look at the red car 1. raise his hand ✓(CoVGT) **2. squat down** 3. no reaction 4.touch his hands

T: *Why* is there smoke behind the places that the car drive past?

0. engine is spoilt ✗(CoVGT) 1. Smoke from fire 2. man is fixing the car 3. snow **4.road dust**

D: *What* is [person_1] holding in his hand?

0. [person_1] is holding [person_1].

✗(CoVGT) 1. [person_1] is holding a shield.

2. [person_1] is holding a plastic cup.

3. [person_1] is holding a faucet.

4. [person_1] is sitting on the stairs.

E: *Why* are [person_1] and [person_2] laughing?

0. Maybe [person_1] wants to bigger his muscle.

1. [person_1] and [person_2] are polishing shoes.

2. Because [person_1] is describing the inhaler.

3. To help [person_1] get on the horse.

✓(CoVGT) **4. [person_1] and [person_2] find something funny.**

P: *What* is [person_1] going to do?

0. [person_1] will get the job done.

1. [person_1] is going to come back home.

2. [person_1] will throw the frisbee.

3. [person_1] will continue to play.

✓(CoVGT) **4. [person_1] is going to stand up.**

Reason:

0. [person_1] is sliding downhill.

✓(CoVGT) **1. [person_1] starts moving his body up.**

2. [person_1] just started.

3. [person_1] is very interested in what he says.

4. People are interested in the story behind [person_1].

C: What would happen *if* [person_1] and [person_2] sat for too long?

0. Their legs would take the ball from him.

1. Their legs would hurt others.

2. Snowmobiles would not be able to travel on the concrete road.

3. Their legs would not lose.

✓(CoVGT) **4. Their legs would be numb and hurt.**

Reason:

0. Their legs have great technique.

1. Cameras are fragile objects and can be broken easily with enough force.

2. Running is one of required technique of their legs.

3. When fixing their legs maybe [person_2] touches the microphone causing the microphone broken.

✓(CoVGT) **4. Because their legs would not be stretch out for a long time**.

Fig. 14: Prediction results on (left) NExT-QA [23] and (right) Causal-VidQA [25]. Ground-truth answers are highlighted in **bold**. Note: The masks and bounding boxes are attached in the video from Causal-VidQA and not our detection.

C.6 More Qualitative Analysis

We show some of our prediction results in Fig. 14. The examples from NExT-QA suggest that understanding the fine-grained video object interactions is of great challenge. Although we have remarkably improved the previous SOTA results, we find that our model still fails to answer a large number of questions. The failure cases show that the model can answer most of the questions reasonably, but the answers are irrelevant to the factual video contents. The observation indicates that our model can well understand the questions yet still not strong enough in comprehending the fine-grained video contents or finding the cross-model correspondences between vision and text. On the other hand, the example from Causal-VidQA shows that our model can successfully answer the questions that emphasize temporal dynamics and find the reasonable explanations as well, such as "going to stand up" and "start moving his body up".