

# Bridging Video-text Retrieval with *Multiple Choice Questions*

Yuying Ge<sup>1</sup> Yixiao Ge<sup>2</sup> Xihui Liu<sup>5</sup> Dian Li<sup>4</sup> Ying Shan<sup>2</sup> Xiaohu Qie<sup>3</sup> Ping Luo<sup>1</sup>

<sup>1</sup>The University of Hong Kong <sup>2</sup>ARC Lab, <sup>3</sup>Tencent PCG

<sup>4</sup>Content Understanding Center, <sup>3</sup>Tencent PCG <sup>5</sup>UC Berkeley

yuyingge@hku.hk {yixiaoge, goodli, yingsshan, tigerqie}@tencent.com

xihui.liu@berkeley.edu pluo@cs.hku.hk

<https://github.com/TencentARC/MCQ>

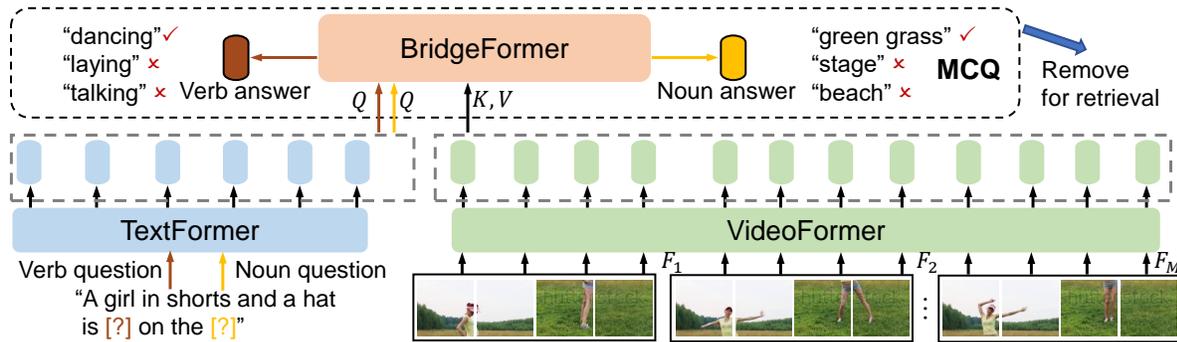


Figure 1. Overview of our novel pretext task, Multiple Choice Questions (MCQ), for video-text pre-training. MCQ is performed using a newly-proposed parametric module BridgeFormer, which associates all-level local features (intermediate tokens) from VideoFormer and TextFormer to answer multiple choice questions in the form of contrastive learning. Given that nouns and verbs carry informative local objects and object motions, we construct a **noun question** (in yellow) and a **verb question** (in red) by erasing the corresponding phrase from the sentence. The BridgeFormer is trained to select the correct erased phrase via visual reasoning with intermediate tokens from VideoFormer, given the questions’ intermediate tokens from TextFormer. The noun and verb questions promote VideoFormer to capture detailed spatial content and temporal information. The semantic associations between video-text intermediate tokens are also enhanced via the proxy task of questions and answers. Note that BridgeFormer is **removed** for downstream retrieval.

## Abstract

Pre-training a model to learn transferable video-text representation for retrieval has attracted a lot of attention in recent years. Previous dominant works mainly adopt two separate encoders for efficient retrieval, but ignore local associations between videos and texts. Another line of research uses a joint encoder to interact video with texts, but results in low efficiency since each text-video pair needs to be fed into the model. In this work, we enable fine-grained video-text interactions while maintaining high efficiency for retrieval via a novel pretext task, dubbed as *Multiple Choice Questions (MCQ)*, where a parametric module BridgeFormer is trained to answer the “questions” constructed by the text features via resorting to the video features. Specifically, we exploit the rich semantics of text (i.e., nouns and verbs) to build questions, with which the video encoder can be trained to capture more regional content and temporal dynamics. In the form of questions and answers, the semantic associations between local video-text

features can be properly established. BridgeFormer is able to be removed for downstream retrieval, rendering an efficient and flexible model with only two encoders. Our method outperforms state-of-the-art methods on the popular text-to-video retrieval task in five datasets with different experimental setups (i.e., zero-shot and fine-tune), including HowTo100M (one million videos). We further conduct zero-shot action recognition, which can be cast as video-to-text retrieval, and our approach also significantly surpasses its counterparts. As an additional benefit, our method achieves competitive results with much shorter pre-training videos on single-modality downstream tasks, e.g., action recognition with linear evaluation.

## 1. Introduction

Pre-training a model to learn transferable representations for video-text retrieval requires the understanding of video concepts, text semantics, and the relationships be-

tween videos and texts. Existing works for video-text pre-training can be divided into two main categories. “Dual-encoder” methods [6, 13, 15, 24, 28, 31, 36, 46, 50] (see Fig. 2 (a)) adopt two separate encoders to contrast video-level and sentence-level representations respectively, ignoring the detailed local information within each modality and the associations between modalities. “Joint-encoder” methods [22, 23, 25, 42, 45, 51] (see Fig. 2 (b)) concatenate texts and videos as inputs to a joint encoder for the interactions between local features of videos and texts, sacrificing the retrieval efficiency (every text-video pair needs to be fed into the encoder during inference) for the benefits of fine-grained feature learning.

To enable fine-grained video-text interactions and at the same time maintaining high retrieval efficiency, we introduce a novel parametric pretext task for video-text pre-training, namely, **Multiple Choice Questions (MCQ)**, which properly bridges texts with videos in all their feature levels. A new module in vitro, termed *BridgeFormer*, makes it possible, as illustrated in Fig. 1. Based on the backbone of a “dual-encoder” framework, BridgeFormer is trained to answer the “questions” generated by the text features via visual reasoning with the video features. MCQ enhances local feature learning within each modality as well as the fine-grained semantic associations cross modalities, and the BridgeFormer can be readily removed when transferring to downstream tasks without the loss of representation discriminativeness.

Specifically, we construct the “questions” by erasing a content phrase from the raw text, and the correct “answer” should be the erased phrase itself. Motivated by the observation that noun and verb phrases in a text carry rich semantic information [50], which can reflect the local objects and object motions in the video respectively, we randomly choose nouns or verbs as our content phrases. BridgeFormer is then trained to select the correct answer from multiple choices (all the erased content phrases in a batch) in the form of contrastive learning by resorting to the local features from the video encoder. Such a proxy training objective enforces the video encoder to capture accurate spatial content (to answer nouns) and temporal dynamics (to answer verbs), promoting the discriminativeness of the local features and the semantic associations between the local video patches and the text phrases.

BridgeFormer connects local features of videos and texts in all feature levels (low-, mid-, and high-level), *i.e.*, taking each stage’s features from the video and text encoders as input. The regularization will be directly imposed on the video and text features, which is different from the video-text feature aggregation by the conventional “joint-encoder”. Therefore, the proxy BridgeFormer only serves for the pre-training step and can be seamlessly removed for downstream retrieval, rendering a flexible and efficient

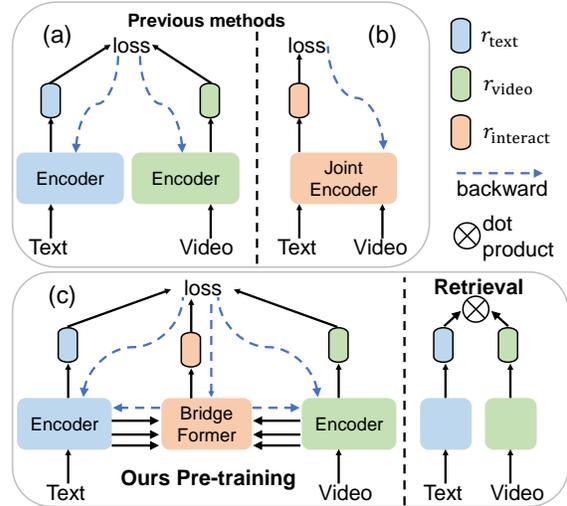


Figure 2. Comparison between existing paradigms and ours for video-text pre-training. Previous dominant methods either (a) adopt two separate encoders to contrast video-level and sentence-level representations, ignoring local associations between videos and texts, or (b) use a joint encoder to interact fine-grained features of videos and texts through concatenating them as inputs, resulting in low efficiency for retrieval. (c) We propose a novel pretext task that uses a BridgeFormer to promote local feature learning and fine-grained video-text associations. For downstream retrieval task, the proxy BridgeFormer is **removed**.

model like the conventional “dual-encoder” methods, *i.e.*, the similarity between video and text representations can be directly measured via dot product.

Our contributions are three-fold. (1) We introduce a novel pretext task, Multiple Choice Questions (MCQ), for video-text pre-training to receive the benefits of both “dual-encoder” and “joint-encoder” methods, *i.e.*, enhancing fine-grained semantic associations between video and text features at the same time preserving high retrieval efficiency. (2) We propose a parametric module, dubbed as BridgeFormer, to realize the pretext task of MCQ, with which the video encoder is trained to be more aware of regional objects and temporal dynamics, and the associations between local video-text features are established. Since the BridgeFormer will be removed on downstream tasks, we do not increase any additional parameters or computational overhead for retrieval compared to vanilla backbones. (3) Extensive results on text-to-video retrieval with different setups (*i.e.*, zero-shot and fine-tune) on five datasets, including the large-scale HowTo100M [29] (1 million videos), demonstrate the large superiority of our method (see Fig. 3 (a)). Furthermore, we evaluate zero-shot action recognition, which can be cast as a video-to-text retrieval task. Our method significantly surpasses its competitive counterparts by a large margin, as demonstrated in Fig. 3 (a). As a bonus, we find our method also benefits single-modality

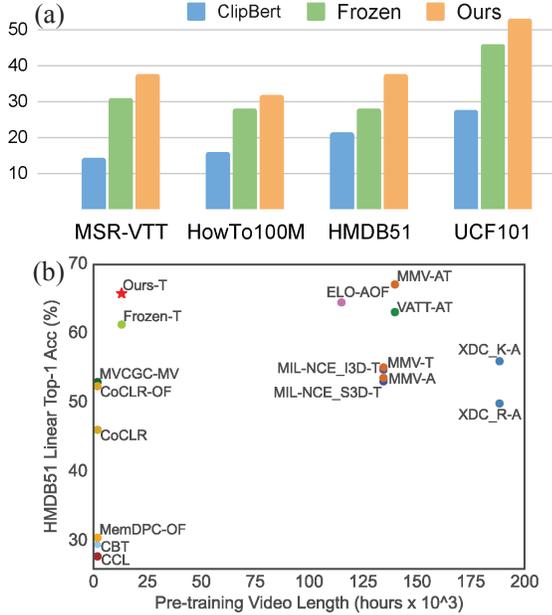


Figure 3. (a) Comparison between recent video-text pre-training methods for zero-shot text-to-video retrieval on MSR-VTT (R@1), HowTo100M (R@50) and zero-shot action recognition (video-to-text retrieval) on HMDB51 (top-1) and UCF101 (top-1). (b) Video length for pre-training and the top-1 accuracy of action recognition with linear evaluation, where “-X” denotes the modality used for pre-training besides videos, *i.e.*, optical flow (OF), motion vector (MV), audio (A), and text (T).

video representations as shown in Fig. 3 (b), where the top-1 accuracy of action recognition with linear evaluation is reported. Despite those considerably longer videos being used in state-of-the-art pre-training methods (*e.g.*, 11× longer in MMV [2] than ours), our method still compares favorably with them.

## 2. Related Work

**Pre-training for video-text retrieval.** Dominant pre-training methods for video-text retrieval can be classified into two categories. Methods in the first category [6, 13, 15, 24, 28, 31, 36, 46, 50] adopt two individual encoders to embed video features and text features, and project them into the same latent space. Contrastive objectives [18, 30] are used here to distinguish paired video-text data with unpaired data. This kind of methods is more favored by large-scale retrieval applications due to its high efficiency. However, simply imposing the regularizations on the final features ([CLS] tokens) from two modalities leads to the insufficient interaction between local video-text representations. Methods in the second category [22, 23, 25, 42, 45, 51] ensemble texts and videos as inputs to a joint encoder for the cross-modality fusion, followed by a binary classifier which is trained to predict whether videos and texts are aligned

or not. Despite they can build local associations between video-text tokens, each pair of video and text candidates needs to be fed into the model for similarity calculation during inference, resulting in extremely low efficiency. In contrast, our method gains the benefits of the above two kinds of methods, *i.e.*, achieving fine-grained video-text interactions while remaining high retrieval efficiency.

**The pretext task of masked word prediction.** Previous cross-modality pre-training work [19, 25, 51] use the pretext of masked word prediction (MWP), which randomly masks a proportion of words in the sentence and regularize the network to predict the masked words from a fixed vocabulary under the condition of visual inputs. Our introduced MCQ pretext task differs from MWP in two ways: (1) Predicting words in MWP imposes the regularizations on low-level word tokens, which may harm the interacted representation learning since the network also needs to serve as a text decoder. In contrast, contrasting answers with content phrases in our MCQ focuses on high-level semantics, showing significantly better results than MWP (will be discussed in experiments). (2) MCQ erases noun and verb phrases to construct informative questions, which reflects salient semantic information in visual features, while MWP randomly masks words (*e.g.*, function words without content).

**Video question answering (VQA).** Works on video question answering (VQA) [7, 27, 37, 49] aims to answer questions about videos through training a model with question and answer pairs, which cannot be directly applied for pre-training as they are deliberately optimized for increasing VQA accuracy. By contrast, our work aims to learn downstream-agnostic generic features for video-text retrieval, where a new pretext task, multiple choice questions, is proposed to enhance the semantic associations between video and text. Our paper *is the first to* use the form of VQA as a pre-training pretext task, with *two key innovations*: the MCQ loss and the BridgeFormer module. BridgeFormer smoothly bridges the final objective of learning well-aligned video and text features with the regularization of a VQA pretext task.

**Video-text retrieval with nouns and verbs.** Works [10, 44, 48, 52] solved video-text retrieval by focusing on verbs and nouns of texts, which are specially designed for retrieval with verbs and nouns as the refined text representations to directly align with videos. By contrast, we exploit the rich semantics of nouns and verbs in the text to build questions for improving text and video encoders.

## 3. Method

We adopt the “dual-encoder” structure for video-text pre-training to realize highly efficient retrieval, and propose a new pretext task, Multiple Choice Questions (MCQ), with a parametric module BridgeFormer, to enhance fine-grained

semantic associations between videos and texts. In this section, we first revisit the dual-encoder in Sec. 3.1. We then introduce the pretext task MCQ in Sec. 3.2 and the pre-training objectives in Sec. 3.3. At last, we describe the architecture of three components including a VideoFormer, a TextFormer, and a BridgeFormer in Sec. 3.4.

### 3.1. Dual-encoder for Video-text Pre-training: a revisit

As shown in Fig. 4, we adopt a dual-encoder structure, which consists of a VideoFormer for learning video representations from raw video frame pixels, and a TextFormer for encoding text representations from natural languages. Given a video and its corresponding text description (e.g., “A girl in shorts and a hat is dancing on the green grass”), we first embed their respective representations from VideoFormer and TextFormer, which are projected to a common embedding space as  $f_v$  and  $f_t$  via two separate linear layers. The similarity between the video and the text is calculated via the dot product between  $f_v$  and  $f_t$ . A contrastive objective [18, 30] is utilized to maximize the similarity between  $f_v$  and  $f_t$  of positive pairs while minimizing the similarity between  $f_v$  and  $f_t$  of negative pairs (A video and its corresponding text description is regarded as a positive pair, and otherwise as a negative pair). The independent dual encoder pathways require only the dot product between video and text representations for similarity calculation in retrieval, which ensures the high efficiency.

### 3.2. Multiple Choice Questions

As shown in Fig. 1, the pretext task MCQ is performed using a parametric module BridgeFormer, which associates all-level intermediate tokens from VideoFormer and TextFormer to answer multiple choice questions. Given observed that noun and verb phrases in a text carry rich semantic information, which can reflect the local objects and object motions in the video respectively, we randomly erase a noun or verb phrase to construct noun or verb questions. BridgeFormer is then trained to select the correct answer from multiple choices (all the erased phrases in a batch) by resorting to the local tokens of VideoFormer in the form of contrastive learning. The pretext task MCQ involves the objectives of answering noun questions and verb questions.

**Answer Noun Question.** Given a video and its corresponding text description (e.g., “A girl in shorts and a hat is dancing on the green grass”), we randomly erase a noun phrase (e.g., “green grass”) as a noun question (e.g., “A girl in shorts and a hat is dancing on the [?]”). As shown in Fig. 4, the noun question is fed into TextFormer for intermediate text tokens  $\{z\}_{noun.q}$ . The intermediate video tokens are extracted from VideoFormer as  $\{z\}_v$ . BridgeFormer takes the noun question tokens  $\{z\}_{noun.q}$  as the query, and video

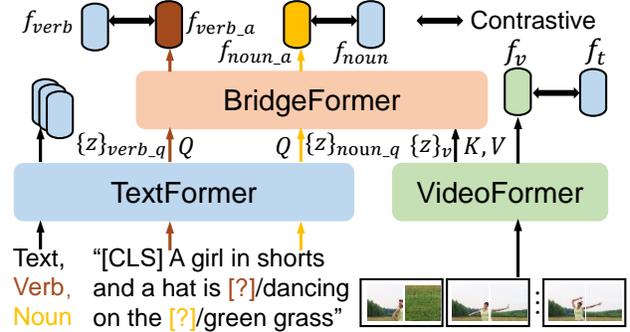


Figure 4. Our pre-training pipeline, which (1) contrasts video representations  $f_v$  with text representations  $f_t$ , (2) trains BridgeFormer to select the correct noun answer by contrasting noun answer representations  $f_{noun.a}$  with noun representations  $f_{noun}$ , (3) trains BridgeFormer to choose the correct verb answer by contrasting verb answer representations  $f_{verb.a}$  with verb representations  $f_{verb}$ . Note that BridgeFormer receives all-level tokens as the input, but we only draw one pathway here for brevity.

tokens  $\{z\}_v$  as the key and value to obtain the noun answer representations through cross-modality attention. The erased noun phrase is fed into TextFormer for noun representations. Similarly, the noun answer representations and the noun representations are projected into a common embedding space as  $f_{noun.a}$  and  $f_{noun}$  via two separate linear layers, and their similarity is calculated via dot product. We adopt a contrastive objective to maximize the similarity between  $f_{noun.a}$  and  $f_{noun}$ , when  $f_{noun}$  is the representations of the correct noun phrase, and minimize the similarity between  $f_{noun.a}$  and  $f_{noun}$ , when  $f_{noun}$  is the representations of other (wrong) noun phrases. Training BridgeFormer to select the correct noun phrase by resorting to video tokens enforces VideoFormer to capture accurate spatial content.

**Answer Verb Question.** Similarly, we randomly erase a verb phrase (e.g., “dancing”) of the text description as a verb question (e.g., “A girl in shorts and a hat is [?] on the green grass”). As shown in Fig. 4, BridgeFormer takes verb question text tokens  $\{z\}_{verb.q}$  from TextFormer as the query, and video tokens  $\{z\}_v$  as the key and value to obtain the verb answer representations. The erased verb phrase is fed into TextFormer for verb representations. The verb answer representations and the verb representations are projected into a common embedding space as  $f_{verb.a}$  and  $f_{verb}$ . A contrastive objective is adopted to maximize the similarity between  $f_{verb.a}$  and  $f_{verb}$ , when  $f_{verb}$  is the representations of the correct verb phrase, and minimize the similarity between  $f_{verb.a}$  and  $f_{verb}$ , when  $f_{verb}$  is the representations of other verb phrases. Training BridgeFormer to choose the correct verb phrase through seeking help from video tokens forces VideoFormer to capture detailed temporal dynamics.

### 3.3. Pre-training Objectives

We adopt the Noise-Contrastive Estimation (NCE) [18, 30] as the contrastive objective and combine three objectives to optimize the entire model in an end-to-end manner as follows,

$$\mathcal{L} = \mathcal{L}_{\text{vanilla}} + \mathcal{L}_{\text{noun}} + \mathcal{L}_{\text{verb}} \quad (1)$$

where  $\mathcal{L}_{\text{vanilla}}$  is the NCE loss between video representations  $f_v$  and text representations  $f_t$ ,  $\mathcal{L}_{\text{noun}}$  is the NCE loss between noun answer representations  $f_{\text{noun},a}$  and noun representations  $f_{\text{noun}}$ ,  $\mathcal{L}_{\text{verb}}$  is the NCE loss between verb answer representations  $f_{\text{verb},a}$  and verb representations  $f_{\text{verb}}$ . We formulate NCE loss as below,

$$\text{NCE}(x_i, y_i) = -\log \frac{\exp(x_i^T y_i / \tau)}{\sum_{j=1}^B \exp(x_i^T y_j / \tau)} \quad (2)$$

where  $B$  is the number of the batch size and the temperature hyper-parameter  $\tau$  is empirically set to 0.05 per [6].

### 3.4. Model Architecture

#### 3.4.1 VideoFormer

**Input.** VideoFormer takes a video  $V \in R^{M \times 3 \times H \times W}$  as input containing variable  $M$  frames of resolution  $H \times W$ . The input video is first divided into  $M \times N$  patches of size  $P \times P$ , where  $N = HW/P^2$ . The video patches  $v \in R^{M \times 3 \times N \times P \times P}$  are fed into a linear projection head with a convolutional layer and are flattened into a sequence of tokens  $z_v \in R^{M \times N \times D}$ , where  $D$  is the number of embedding dimensions. Following BERT [11], a learnable [CLS] token is concatenated to the beginning of the token sequence, which is used to produce the final video representations. Learnable spatial positional embeddings  $E_{\text{pos}} \in R^{(N+1) \times D}$  are added to each video token as the final input token sequence  $z_v^0 \in R^{(1+M \times N) \times D}$  and all patches in the same spatial location in different frames are given the same spatial positional embedding.

**VideoBlock.** The input video token sequence  $\{z\}_v^0$  is fed into VideoFormer, which consists of a stack of VideoBlocks as shown in Fig. 5, adopting the structure of ViT [12]. We make a minor modification to the original ViT to allow for the input of video frames with variable length. Specifically, given  $z_v^{l-1} \in R^{(1+M \times N) \times D}$  from previous VideoBlock, we perform multi-head attention (MSA) [12] for the [CLS] token through attending to all  $(1 + M \times N)$  patches across time and space for temporal and spatial self-attention. For the rest  $(M \times N)$  patch tokens, MSA is performed within each of  $M$  frames with  $N + 1$  tokens ( $N$  patch tokens and 1 [CLS] token) for spatial self-attention. The video representations are obtained from the [CLS] token of the final VideoBlock.

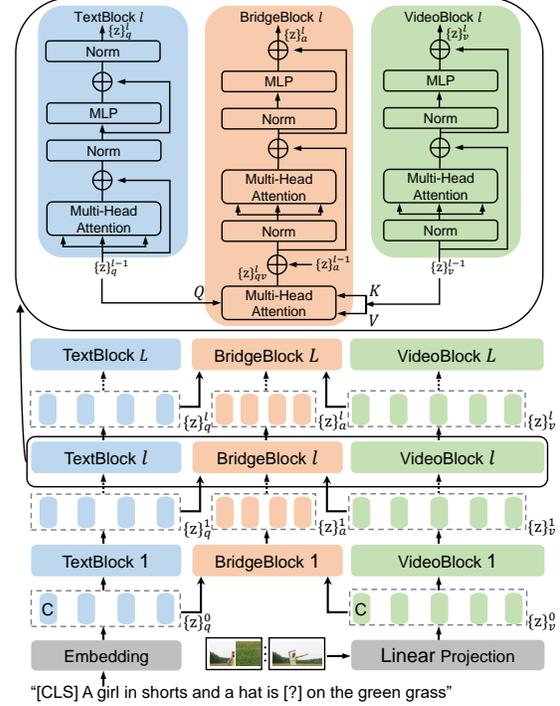


Figure 5. The architecture of TextFormer, VideoFormer and BridgeFormer, which contain a stack of TextBlocks, VideoBlocks and BridgeBlocks respectively. Tokens from all-level VideoBlock and TextBlock are fed into the corresponding BridgeBlock to perform cross-modal attention and then are added to the output tokens of the previous BridgeBlock (if any). Each block performs a series of operations such as multi-head attention [12], normalization (norm) and multi-layer perception [11] (MLP).

#### 3.4.2 TextFormer

**Input.** TextFormer takes three kinds of nature languages as inputs, including a complete text description, noun or verb questions with a noun or verb phrase erased, and the erased noun or verb phrase. A [CLS] token is concatenated to the beginning of the input for final text representations.

**TextBlock.** We adopt a multi-layer bidirectional transformer encoder [38] as TextFormer, which consists of a stack of TextBlocks as shown in Fig. 5.

#### 3.4.3 BridgeFormer

**Input.** BridgeFormer takes noun question or verb question tokens from TextFormer as the query, and video tokens from VideoFormer as the key and value to obtain the answer representations with cross-modality attention.

**BridgeBlock.** BridgeFormer is built upon a vision transformer with a stack of BridgeBlocks as shown in Fig. 5. Specifically, given noun question or verb question text tokens  $\{z\}_q^{l-1} \in R^{L \times D}$  from TextBlock as the query, and

video tokens  $\{z\}_v^{l-1} \in R^{M \times (N \times D)}$  (without the [CLS] token) from VideoBlock as the key and value, BridgeBlock- $l$  obtains the interacted tokens  $\{z\}_{qv}^l$  through performing multi-head attention, which calculates the cross-modality attention between the question text tokens and video patch tokens within each frame. The interacted tokens  $\{z\}_{qv}^l$  added with the output  $\{z\}_a^{l-1}$  from the previous BridgeBlock further go through the attention block for temporal and spatial self-attention as shown in Fig. 5 to obtain the answer tokens  $\{z\}_a^l$ . The answer representations are extracted from the [CLS] token of the final block.

## 4. Experiments

### 4.1. Pre-training Datasets

Following the recent work [6], we jointly pre-train our model on an image dataset Google Conceptual Captions (CC3M) [39] with 3.3M image-text pairs, and a video dataset WebVid-2M [6] with 2.5M video-text pairs. We do not pre-train our model on the large-scale video-text dataset HowTo100M [29] with 136M video-text pairs considering the enormous computation cost. Instead, we use HowTo100M as a large-scale zero-shot text-to-video retrieval benchmark for evaluation, which is in line with real-world applications.

### 4.2. Downstream Tasks

**Text-to-Video Retrieval.** (a). **MSR-VTT** [47] contains 10K YouTube videos with 200K descriptions, which is split into 9K videos for training and 1K videos for test. (b). **MSVD** [9] consists of 1,970 videos from YouTube with 80K descriptions, which is split into 1200, 100 and 670 videos for training, validation and testing. (c). **LSMDC** [35] consists of 118,081 video clips from 202 movies. The validation set and the test set contain 7,408 and 1,000 videos. (d). **DiDeMo** [5] contains 10K Flickr videos with 40K sentences, where the test set contains 1,000 videos. We concatenate all sentence descriptions for a video as a single query following [6]. (e). **HowTo100M** [29] contains 1.22M videos with 136M descriptions. All sentence descriptions for a video are concatenated as a single query. To our knowledge, it is the first time that downstream text-to-video retrieval is evaluated on the large-scale dataset, *i.e.*, HowTo100M. Two setting are explored for evaluation, including **zero-shot** and **fine-tune**.

**Action Recognition.** (a). **HMDB51** [21], which contains 6,766 videos with 51 categories. (b). **UCF101** [40], which contains 13,320 videos with 101 action classes. Three setting are explored for evaluation, including **linear**, where parameters of the learned video encoder are frozen and only a linear classifier is optimized, **fine-tune**, where the video encoder is fine-tuned with the linear classifier, and **zero-**

**shot**, which performs video-to-text retrieval through using the names of the action classes as the text description.

### 4.3. Implementation Details

Videos are resized to  $224 \times 224$  as input. We divide a video into  $M$  equal segments, and randomly sample a single frame from each segment for training while uniformly sample a frame from each segment for testing. VideoFormer contains 12 blocks with patch size  $P = 16$ , and sequence dimension  $D = 768$ . It is initialized with ViT [12] weights trained on ImageNet-21k following [6]. TextFormer adopts the architecture of DistilBERT [38] pre-trained on English Wikipedia and Toronto Book Corpus. The dimension of the common feature space is set to 256. The temperature hyper-parameter of the contrastive objective is set to 0.05. The above implementation details follow the recent work [6] for fair comparison. BridgeFormer contains 12 blocks. We first pre-train our model on the image dataset CC3M and video dataset WebVid-2M using 1 frame for 10 epochs with the batch size of 2048 and the learning rate of  $1 \times 10^{-4}$ . We then pre-train our model on the video dataset WebVid-2M using 4 frames for 4 epochs with the batch size of 800 and the learning rate of  $3 \times 10^{-5}$ . Pre-training takes a total of 25 hours. For downstream tasks, 4 frames for text-to-video retrieval and 16 frames for action recognition are uniformly sampled following the setting of previous work [6, 28].

### 4.4. Main Results

#### 4.4.1 Text-to-Video Retrieval

Table. 1 lists the results on MST-VTT [47]. First of all, our method outperforms all previous work by a large margin. The significantly higher performance of our model under the zero-shot evaluation demonstrates the stronger generalization ability of our pre-trained model. Fine-tuning our pre-trained model on the training set of MSR-VTT also surpasses its counterparts overwhelmingly, showing its advantage in using task-specific data for optimization. Second, while previous work mostly pre-train on HowTo100M [29] with the magnitude exceedingly large than our pre-training dataset CC3M [39] and WebVid-2M [6] (20x larger in the number of video-text pairs), our method still achieves the highest performance with much lower computation cost (*i.e.* VATT [1] takes 3 days using 256 TPUs while ours takes 25 hours using 40 A100.) Third, previous work rely on pre-extracted features from “expert” models as the input of the video encoder (*i.e.* SupportSet [31] uses features from a 34-layer, R(2+1)-D model [43] pre-trained on IG65M [14] as the input), while our model takes raw video frame pixels as inputs and achieves significant performance gain. Finally, compared with previous work [22, 23, 25, 45, 51] that adopt a joint encoder to concatenate videos and texts as inputs and thus every text-video combination needs to be inputted to the model for retrieval, our model only contains a video

Table 1. Experiments of text-to-video retrieval on MSR-VTT test set with 1K videos, where **higher R@k** and **lower MedR** (Median Rank) indicate better performance. **Video Encoder Input**: 3D features from the architectures (Raw Videos means training on raw video frame pixels without using pre-extracted features). **# Pairs PT**: the number of video-text pairs for pre-training. We show results with zero-shot evaluation (top) and fine-tuning evaluation (bottom).

| Method          | Year | Video Encoder Input | PT Dataset          | #Pairs PT | R@1         | R@5         | R@10        | MedR       |
|-----------------|------|---------------------|---------------------|-----------|-------------|-------------|-------------|------------|
| ActBERT [51]    | 2020 | ResNet-3D           | HowTo100M           | 120M      | 8.6         | 23.4        | 33.1        | 36.0       |
| MMV [2]         | 2020 | Raw Videos          | HowTo100M, AudioSet | 138M      | 9.3         | 23.0        | 31.1        | 38.0       |
| MIL-NCE [28]    | 2020 | Raw Videos          | HowTo100M           | 120M      | 9.9         | 24.0        | 32.4        | 29.6       |
| VATT [1]        | 2021 | Raw Videos          | HowTo100M, AudioSet | 138M      | -           | -           | 29.7        | 49.0       |
| NoiseEst [4]    | 2021 | ResNeXt-101         | HowTo100M           | 110M      | 8.0         | 21.3        | 29.3        | 33.0       |
| TACo [50]       | 2021 | I3D, S3D            | HowTo100M           | 120M      | 9.8         | 25.0        | 33.4        | 29.0       |
| VideoCLIP [46]  | 2021 | S3D                 | HowTo100M           | 110M      | 10.4        | 22.2        | 30.0        | -          |
| MCN [8]         | 2021 | ResNeXt-101         | HowTo100M           | 120M      | 10.5        | 25.2        | 33.8        | -          |
| SupportSet [31] | 2021 | R(2+1)D-34          | HowTo100M           | 120M      | 12.7        | 27.5        | 36.2        | 24.0       |
| Frozen [6]      | 2021 | Raw Videos          | CC3M, WebVid-2M     | 5.5M      | 18.7        | 39.5        | 51.6        | 10.0       |
| AVLnet [36]     | 2021 | ResNeXt-101         | HowTo100M           | 120M      | 19.6        | 40.8        | 50.7        | 9.0        |
| Ours            | 2021 | Raw Videos          | CC3M, WebVid-2M     | 5.5M      | <b>26.0</b> | <b>46.4</b> | <b>56.4</b> | <b>7.0</b> |
| ActBERT [51]    | 2020 | ResNet-3D           | HowTo100M           | 120M      | 16.3        | 42.8        | 56.9        | 10.0       |
| UniVL [25]      | 2020 | S3D                 | HowTo100M           | 110M      | 21.2        | 49.6        | 63.1        | 6.0        |
| MMT [13]        | 2020 | S3D                 | HowTo100M           | 120M      | 26.6        | 57.1        | 69.6        | 4.0        |
| HERO [23]       | 2021 | SlowFast            | TV and HowTo100M    | 120M      | 16.8        | 43.4        | 57.7        | -          |
| NoiseEst [4]    | 2021 | ResNeXt-101         | HowTo100M           | 110M      | 17.4        | 41.6        | 53.6        | 8.0        |
| ClipBert [22]   | 2021 | Raw Videos          | COCO, VisGenome     | 5.6M      | 22.0        | 46.8        | 59.9        | 6.0        |
| AVLnet [36]     | 2021 | ResNeXt-101         | HowTo100M           | 120M      | 27.1        | 55.6        | 66.6        | 4.0        |
| VLM [45]        | 2021 | S3D                 | HowTo100M           | 110M      | 28.1        | 55.5        | 67.4        | 4.0        |
| TACo [50]       | 2021 | I3D, S3D            | HowTo100M           | 120M      | 28.4        | 57.8        | 71.2        | 4.0        |
| SupportSet [31] | 2021 | R(2+1)D-34          | HowTo100M           | 120M      | 30.1        | 58.5        | 69.3        | 3.0        |
| VideoCLIP [46]  | 2021 | S3D                 | HowTo100M           | 110M      | 30.9        | 55.4        | 66.8        | -          |
| Frozen [6]      | 2021 | Raw Videos          | CC3M, WebVid-2M     | 5.5M      | 31.0        | 59.5        | 70.5        | 3.0        |
| Ours            | 2021 | Raw Videos          | CC3M, WebVid-2M     | 5.5M      | <b>37.6</b> | <b>64.8</b> | <b>75.1</b> | <b>3.0</b> |

Table 2. Experiments of text-to-video retrieval on different datasets, where **higher R@k** and **lower MedR** (Median Rank) indicate better performance. We show results with zero-shot evaluation (top) and fine-tuning evaluation (bottom).

| (a) MSVD test set with 670 videos. |             |             |             |            | (b) LSMDC test set with 1K videos. |             |             |             |             | (c) DiDeMo test set with 1K videos. |             |             |             |            |
|------------------------------------|-------------|-------------|-------------|------------|------------------------------------|-------------|-------------|-------------|-------------|-------------------------------------|-------------|-------------|-------------|------------|
| Method                             | R@1         | R@5         | R@10        | MedR       | Method                             | R@1         | R@5         | R@10        | MedR        | Method                              | R@1         | R@5         | R@10        | MedR       |
| NoiseEst [4]                       | 13.7        | 35.7        | 47.7        | 12.0       | AVLnet [36]                        | 1.4         | 5.9         | 9.4         | 273.5       | VideoCLIP [46]                      | 16.6        | 46.9        | -           | -          |
| SupportSet [31]                    | 21.4        | 46.2        | 57.7        | 6.0        | NoiseEst [4]                       | 4.2         | 11.6        | 17.1        | 119.0       | Frozen [6]                          | 21.1        | 46.0        | 56.2        | 7.0        |
| Frozen [6]                         | 33.7        | 64.7        | 76.3        | 3.0        | Frozen [6]                         | 9.3         | 22.0        | 30.1        | 51.0        | Ours                                | <b>25.6</b> | <b>50.6</b> | <b>61.1</b> | <b>5.0</b> |
| Ours                               | <b>43.6</b> | <b>74.9</b> | <b>84.9</b> | <b>2.0</b> | Ours                               | <b>12.2</b> | <b>25.9</b> | <b>32.2</b> | <b>42.0</b> | HERO [23]                           | 2.1         | -           | 11.4        | -          |
| NoiseEst [4]                       | 20.3        | 49.0        | 63.3        | 6.0        | NoiseEst [4]                       | 6.4         | 19.8        | 28.4        | 39.0        | CE [24]                             | 16.1        | 41.1        | 82.7        | 8.3        |
| SupportSet [31]                    | 28.4        | 60.0        | 72.9        | 4.0        | MMT [13]                           | 12.9        | 29.9        | 40.1        | 19.3        | ClipBert [22]                       | 20.4        | 48.0        | 60.8        | 6.0        |
| Frozen [6]                         | 45.6        | 79.8        | 88.2        | 2.0        | Frozen [6]                         | 15.0        | 30.8        | 39.8        | 20.0        | Frozen [6]                          | 31.0        | 59.8        | 72.4        | 3.0        |
| Ours                               | <b>52.0</b> | <b>82.8</b> | <b>90.0</b> | <b>1.0</b> | Ours                               | <b>17.9</b> | <b>35.4</b> | <b>44.5</b> | <b>15.0</b> | Ours                                | <b>37.0</b> | <b>62.2</b> | <b>73.9</b> | <b>3.0</b> |

and a text encoder for downstream retrieval, which requires only the dot product between the video and text representations, thus greatly improves efficiency. We further show text-to-video retrieval results on MSVD [9], DiDeMo [5] and LSMDC in Table 2. We can observe that our model achieves the best performance on these three datasets with both zero-shot and fine-tuning evaluation.

Besides evaluating text-to-video retrieval on a relatively small number of videos following previous work (e.g. 1K videos in MSR-VTT test set), we evaluate our model on the large-scale HowTo100M with 1 million videos, which is a more challenging and realistic scenario. Table 3 shows that our pre-trained model surpasses SOTA Frozen [6], ranging from 10K videos to 1M videos. Since our method and

Frozen both adopt two encoders (built on ViT [12] and DistilBERT [38]) for retrieval and are pre-trained on the same datasets, the superior performance of ours proves the effectiveness of our pretext task MCQ in learning powerful representations for text-to-video retrieval.

#### 4.4.2 Action Recognition

We conduct zero-shot action recognition on HMDB51 [21] and UCF101 [40], which can be treated as **video-to-text retrieval** and it is not evaluated in recent methods. As shown in Table 4, our model significantly surpasses its competitive counterparts. The top-1 accuracy of our model averaged on three splits improves 16.3% and 9.9% on HMDB51, 25.3%

Table 3. Experiments of zero-shot text-to-video retrieval on the large-scale HowTo100M, where **higher** R@k and **lower** MedR indicate better performance. “Video Num” denotes the number of sampled videos for evaluation, where 1M denotes the whole set.

| Video Num | Method        | R@50        | R@200       | R@500       | MedR           |
|-----------|---------------|-------------|-------------|-------------|----------------|
| 10K       | ClipBert [22] | 15.8        | 33.6        | 49.8        | 506.0          |
|           | Frozen [6]    | 28.0        | 46.6        | 61.5        | 244.0          |
|           | Ours          | <b>31.6</b> | <b>50.9</b> | <b>65.2</b> | <b>189.0</b>   |
| 50K       | Frozen [6]    | 13.4        | 25.0        | 36.2        | 1247.0         |
|           | Ours          | <b>15.9</b> | <b>28.6</b> | <b>40.2</b> | <b>965.0</b>   |
| 0.1M      | Frozen [6]    | 9.4         | 18.5        | 27.5        | 2519.0         |
|           | Ours          | <b>11.5</b> | <b>21.7</b> | <b>31.2</b> | <b>1907.0</b>  |
| 0.5M      | Frozen [6]    | 4.0         | 8.5         | 13.4        | 12501.0        |
|           | Ours          | <b>5.0</b>  | <b>10.3</b> | <b>15.9</b> | <b>9449.0</b>  |
| 1M        | Frozen [6]    | 2.6         | 5.9         | 9.5         | 24597.0        |
|           | Ours          | <b>3.4</b>  | <b>7.3</b>  | <b>11.6</b> | <b>18612.0</b> |

Table 4. Experiments of zero-shot action recognition (video-to-text retrieval) on HMDB51 and UCF101, in terms of top-1 accuracy. “S” denotes different test splits and “Mean” reports the results averaged on three splits.

| Method        | HMDB51      |             |             |             | UCF101      |             |             |             |
|---------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|-------------|
|               | S1          | S2          | S3          | Mean        | S1          | S2          | S3          | Mean        |
| ClipBert [22] | 20.0        | 22.0        | 22.3        | 21.4        | 27.5        | 27.0        | 28.8        | 27.8        |
| Frozen [6]    | 27.5        | 28.3        | 27.7        | 27.8        | 45.4        | 44.7        | 47.7        | 45.9        |
| Ours          | <b>38.0</b> | <b>36.1</b> | <b>39.1</b> | <b>37.7</b> | <b>51.1</b> | <b>54.3</b> | <b>53.8</b> | <b>53.1</b> |

and 7.2% on UCF101 than the recently proposed ClipBert and Frozen, which shows the great advantage of our model in learning joint representations between videos and languages that enable zero-shot action recognition.

We further evaluate the **single-modality video representations** of our model via action recognition with linear and fully fine-tuning evaluation as shown in Table 5, where the representations from VideoFormer are extracted as the input of a trainable linear classifier. Our method achieves higher accuracy than some previous work that pre-train their model on datasets with considerably longer video time (*e.g.* 14× longer in XDC [3], 10× longer in MIL-NCE [28] and VATT [1]), showing the effectiveness of our method in learning transferable video representations for action recognition. Despite MMV [2] performs better than our method when pre-training on datasets 11× longer than ours with multiple modalities including audio and text besides video, its performance lags far behind ours when only audio and video or text and video are used. We can conclude that our method utilizes the language modality more efficiently to learn stronger video representations with fewer video hours.

#### 4.4.3 CLIP-based Pre-training

Because of the prominent success of the CLIP [34] (Contrastive Language-Image Pre-training) in learning image-text representations, which is pre-trained on 400 million image-text pairs, some recent work [26, 33] utilize the pre-trained CLIP for text-to-video retrieval. We also initialize

Table 5. Experiments of action recognition on HMDB51 and UCF101 with linear evaluation (Lin) and fully fine-tuning evaluation (Full). The evaluation metric is top-1 accuracy. “Mod” denotes the modality used for pre-training besides videos, *i.e.*, optical flow (OF), motion vector (MV), audio (A), text (T). “Len” denotes the video length for pre-training in *kilo* hours.

| Method       | Mod   | Len (K) | HMDB        |             | UCF         |             |
|--------------|-------|---------|-------------|-------------|-------------|-------------|
|              |       |         | Lin         | Full        | Lin         | Full        |
| CCL [20]     | -     | 1.8     | 29.5        | 37.8        | 54.0        | 69.4        |
| CBT [41]     | -     | 1.8     | 29.5        | 44.5        | 54.0        | 79.5        |
| MemDPC [16]  | OF    | 1.8     | 30.5        | 54.5        | 54.1        | 86.1        |
| CoCLR [17]   | OF    | 1.8     | 52.4        | 62.9        | 77.8        | 90.6        |
| MVCGC        | MV    | 1.8     | 53.0        | 63.4        | 78.0        | 90.8        |
| XDC_R [3]    | A     | 188.3   | 49.9        | 61.2        | 80.7        | 88.8        |
| XDC_K [3]    | A     | 188.3   | 56.0        | 63.1        | 85.3        | 91.5        |
| MIL-NCE [28] | T     | 134.5   | 54.8        | 59.2        | 83.4        | 89.1        |
| Frozen [6]   | T     | 13.0    | 61.3        | 66.3        | 87.8        | 89.8        |
| VATT [1]     | A, T  | 139.8   | 63.3        | -           | 89.2        | -           |
| ELO [32]     | A, OF | 115.0   | 64.5        | 67.4        | -           | 93.8        |
| MMV [2]      | A     | 134.5   | 53.6        | -           | 77.1        | -           |
| MMV [2]      | T     | 134.5   | 55.1        | -           | 86.8        | -           |
| MMV [2]      | A, T  | 139.8   | <b>67.1</b> | <b>75.0</b> | <b>91.8</b> | <b>95.2</b> |
| Ours         | T     | 13.0    | 65.8        | 69.8        | 89.1        | 92.3        |

our model from CLIP weights to pre-train a model following the setting of CLIP4Clip [26]. Specifically, we use the pre-trained CLIP (ViT-B/32) as the backbone of VideoFormer and TextFormer, and randomly initialize BridgeFormer. The comparisons between our method and other CLIP-initialized methods are shown in Table 6. We can observe that our CLIP-based pre-trained model achieves higher performance for text-to-video retrieval on three datasets with under both the zero-shot and fine-tune evaluation. Our pretext task MCQ also benefits CLIP-based video-text pre-training for downstream text-to-video retrieval.

#### 4.5. Ablation Studies

In this section, we discuss the effectiveness of our design on the pretext task MCQ through evaluating different models for zero-shot text-to-video retrieval on MSR-VTT, and zero-shot action recognition on HMDB51 and UCF101.

**Is MCQ effective?** Yes. As shown in Table 7, pre-training a model without MCQ pretext task drops performance significantly, where only two separate encoders are adopted to contrast video-level and sentence-level features.

**Does it help to answer noun and verb questions?** Yes. As shown in Table 7, training the BridgeFormer through answering noun questions only or verb questions only both harm performance. Randomly erasing words to construct questions also achieves worse results.

**Do videos help to answer questions?** Yes. As shown in Table 8, when the noun-question and verb-question select answers only through calculating the similarity between question representations and phrase representations from text

Table 6. Text-to-video retrieval results of models initialized from CLIP [34] weights on different datasets under zero-shot and fine-tune evaluation, where **higher** R@k and **lower** MdR (Median Rank) and MnR (Mean Rank) indicate better performance.

| Method             | MSR-VTT     |             |             |            |             | MSVD        |             |             |            |            | LSMDC       |             |             |             |             |
|--------------------|-------------|-------------|-------------|------------|-------------|-------------|-------------|-------------|------------|------------|-------------|-------------|-------------|-------------|-------------|
|                    | R@1         | R@5         | R@10        | MdR        | MnR         | R@1         | R@5         | R@10        | MdR        | MnR        | R@1         | R@5         | R@10        | MdR         | MnR         |
| CLIP-straight [33] | 31.2        | 53.7        | 64.2        | 4.0        | -           | 37.0        | 64.1        | 73.8        | 3.0        | -          | 11.3        | 22.7        | 29.2        | 56.5        | -           |
| CLIP4Clip [26]     | 32.0        | 57.0        | 66.9        | 4.0        | 34.0        | 38.5        | 66.9        | 76.8        | 2.0        | 17.8       | 15.1        | 28.5        | 36.4        | 28.0        | 117.0       |
| Ours               | <b>33.2</b> | <b>58.0</b> | <b>68.6</b> | <b>4.0</b> | <b>25.7</b> | <b>48.4</b> | <b>76.4</b> | <b>85.8</b> | <b>2.0</b> | <b>7.4</b> | <b>15.5</b> | <b>30.7</b> | <b>38.7</b> | <b>22.0</b> | <b>97.9</b> |
| CLIP4Clip [26]     | 43.1        | 70.4        | <b>80.8</b> | 2.0        | 16.2        | 46.2        | 76.1        | 84.6        | 2.0        | 10.0       | 20.7        | 38.9        | 47.2        | 13.0        | 65.3        |
| Ours               | <b>44.9</b> | <b>71.9</b> | 80.3        | <b>2.0</b> | <b>15.3</b> | <b>54.4</b> | <b>82.8</b> | <b>89.4</b> | <b>1.0</b> | <b>6.1</b> | <b>21.8</b> | <b>41.1</b> | <b>50.6</b> | <b>10.0</b> | <b>60.5</b> |

Table 7. Ablation studies on different components of MCQ. Results of zero-shot text-to-video retrieval on MSR-VTT and zero-shot action recognition on HMDB51 and UCF101 are reported.

| Method        | MSR-VTT     |             |             | HMDB51      | UCF101      |
|---------------|-------------|-------------|-------------|-------------|-------------|
|               | R@1         | R@5         | R@10        | Top-1       | Top-1       |
| w/o MCQ       | 22.3        | 43.8        | 52.0        | 33.2        | 45.7        |
| Answer Random | 23.0        | 45.5        | 55.5        | 36.9        | 50.7        |
| Answer Noun   | 24.9        | 46.2        | <b>58.0</b> | 36.2        | 51.8        |
| Answer Verb   | 23.3        | <b>46.7</b> | 57.5        | 36.3        | 51.5        |
| MWP           | 20.6        | 39.7        | 50.1        | 29.0        | 38.7        |
| Highest-level | 23.3        | 46.0        | 56.4        | 36.5        | 47.7        |
| Ours          | <b>26.0</b> | 46.4        | 56.4        | <b>37.7</b> | <b>53.1</b> |

Table 8. Ablation study on the effects of video information when answering the questions. Results on WebVid-2M validation set for noun or verb questions are reported.

|            | Answer Noun |      |      | Answer Verb |      |      |
|------------|-------------|------|------|-------------|------|------|
|            | R@1         | R@5  | R@10 | R@1         | R@5  | R@10 |
| w/o Video  | 6.6         | 17.5 | 24.3 | 4.5         | 12.3 | 17.7 |
| with Video | 58.6        | 81.1 | 87.2 | 40.7        | 64.0 | 73.2 |

encoder without resorting to video tokens through BridgeFormer, the results decrease sharply.

### Multiple Choice Questions vs. Masked Word Prediction.

Training the BridgeFormer to predict the answer in the form of word tokens (similar to existing masked word prediction (MWP)) rather than select the correct answer in a batch of phrases in our MCQ actually hurts performance as shown in Table. 7, which is even lower than the baseline (w/o MCQ).

### All-level features vs. highest-level features for BridgeFormer.

When BridgeFormer takes the highest-level features from the text and video encoders as inputs (a cascading structure) instead of all-level features (a parallel structure), we observe the performance drops as shown in Table. 7 due to the lack of regularization on intermediate features. Even so, using only the highest-level features can also slightly outperform our baseline (w/o MCQ), indicating the effectiveness of our MCQ pretext task. Actually, such a cascading structure is similar to those used in previous works [22, 25, 42] where two separate encoders followed by a cross transformer are adopted. However, the cross transformer in these works cannot be easily removed in the same way as our BridgeFormer for downstream retrieval, *e.g.*, evident 6.7% decreases were observed in [25] in terms of R@1 on text-to-video retrieval, further indicating the flexibility and feasibility of our novel MCQ.

Table 9. The effects of the prompt “[MASK]” for noun and verb representations, where “End”, “Middle” and “Start” denote the location of the prompt. For zero-shot text-to-video retrieval on MSR-VTT, **higher** R@k indicates better performance. For zero-shot action recognition on HMDB51 and UCF101, **higher** top-1 accuracy is better.

| Method     | MSR-VTT     |             |             | HMDB51      | UCF101      |
|------------|-------------|-------------|-------------|-------------|-------------|
|            | R@1         | R@5         | R@10        | Top-1       | Top-1       |
| w/o Prompt | 23.1        | 43.5        | 54.3        | 34.8        | 45.8        |
| End        | 24.2        | <b>45.7</b> | 54.4        | 33.4        | 48.5        |
| Middle     | 24.3        | 43.2        | 53.9        | 33.1        | 46.4        |
| Start      | <b>25.1</b> | 45.4        | <b>55.4</b> | <b>34.9</b> | <b>51.4</b> |

Table 10. Comparisons between the video encoder in our method and Frozen [6]. The evaluation is performed on zero-shot text-to-video retrieval on MSR-VTT, where **higher** R@k and **lower** MdR (Median Rank) indicate better performance. “# Params” denotes the number of parameters of the video encoder (M: million).

| Method     | R@1         | R@5         | R@10        | MdR        | # Params |
|------------|-------------|-------------|-------------|------------|----------|
| Frozen [6] | 18.7        | 39.5        | 51.6        | 10.0       | 114M     |
| Ours       | <b>22.3</b> | <b>43.8</b> | <b>52.0</b> | <b>9.0</b> | 86M      |

### Prompt for Phrase Representation

In our method, BridgeFormer is trained to select the correct answer by contrasting noun answer representations with noun representations, and contrasting verb answer representations with verb representations. Accurate representations for noun and verb phrases are essential. Since TextFormer is trained with full sentences, it fails to encode accurate representations for phrases when it takes a single noun or verb phrase as the input due to the lack of context. Motivated by the success of prompt engineering [34], we add “[MASK]” before the noun and verb phrase (*e.g.* “[MASK] [MASK] [MASK] green grass”) to extract noun or verb representations from TextFormer. We show ablation studies of the prompt “[MASK]” for noun and verb representations in Table. 9, where each model is pre-trained using 1 frame. The model without the prompt “[MASK]” takes a single noun or verb phrase as inputs, and achieves the worse results on both the zero-shot text-to-video retrieval and action recognition, showing that TextFormer cannot understand the semantics accurately with a single noun or verb phrase as inputs. The model with the prompt “[MASK]” at the beginning of the phrase achieves the best results in general, and we adopt this practice in our method.



(a) “An old couple/[?] (Q1) are drinking coffee, and there is a plate of bread/[?] (Q2) on the table in front of them.”



(b) “A girl is walking with a dog/[?] (Q1) near a lake/[?] (Q2), and there is a meadow on her left.”



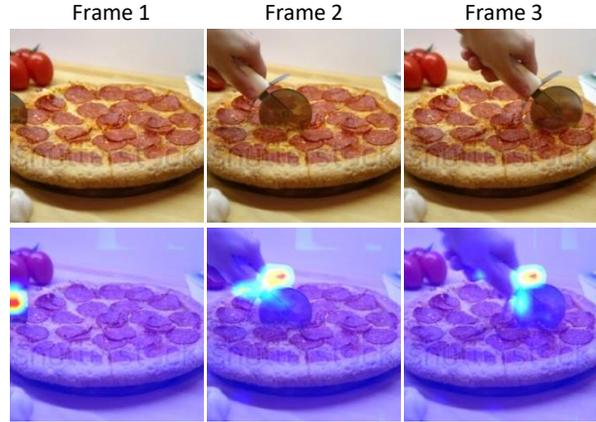
(c) “A woman wearing a pink dress/[?] (Q1) and carrying a black handbag/[?] (Q2) is walking in the park.”



(d) “Parents and kids are playing football/[?] (Q1) on the countryside lawn/[?] (Q2).”

Figure 6. The visualization of the cross-modality attention between the text tokens of **noun questions** (as query) and video tokens (as key and value) from BridgeFormer. In the second column, the noun phrase marked in blue (Q1) is erased as the question, and in the third column, the noun phrase marked in green (Q2) is erased as the question. BridgeFormer attends to video patches with specific object information to answer noun questions.

**Comparison of Video Encoder with Frozen.** Frozen [6] also adopts ViT [12] as the video encoder, and adds temporal attention blocks based on the spatial attention blocks of ViT to encode videos with variable-length sequences. As shown in Table. 10, compared with Frozen, our VideoFormer decreases 28 million parameters. Furthermore, the model without the pretext task MCQ indeed takes the same pre-training approach as Frozen except for the video encoder, and achieves better results for zero-shot text-to-video retrieval on MSR-VTT [47], which proves the efficiency and effectiveness of our VideoFormer.



(a) “A hand is cutting/[?] (Q) the pizza on the wooden table.”



(b) “A man standing on the lake shore is drinking/[?] (Q) hot tea.”

Figure 7. The visualization of the cross-modality attention between the text tokens of **verb questions** (as query) and video tokens (as key and value) from BridgeFormer. Three frames sampled from a video are shown and the verb phrase marked in blue (Q) is erased as the question. BridgeFormer focuses on object motions of video tokens to answer verb questions.

## 4.6. Visualization

In our method, the pretext task MCQ is performed using a parametric module BridgeFormer, to answer multiple choice questions. We construct questions through erasing the content phrases (*i.e.* noun and verb phrases) of the text, and BridgeFormer is trained to select the correct answer from multiple choices by resorting to the local tokens of VideoFormer. Specifically, given question text tokens from TextFormer as the query, and video tokens from VideoFormer as the key and value, BridgeFormer performs cross-modality attention between them.

### 4.6.1 Answering Noun Questions

We first visualize the cross-modality attention between noun question tokens and video tokens in Fig. 6. In the second column, the noun phrase marked in blue (Q1) is erased

as the question, and in the third column, the noun phrase marked in green (Q2) is erased as the question. In Fig. 6 (a), when “an old couple” is erased as the question (Q1), BridgeFormer focuses on video tokens that depict the appearance characteristics of the persons, and when “a plate of bread” is erased (Q2), it focuses on object video tokens on the table. In Fig. 6 (d), when “football” is erased (Q1), BridgeFormer focuses on the object video tokens that can be associated with “play”, and when the location phrase “countryside lawn” is erased (Q2), it pays more attention to the video tokens in the background to infer the answer. BridgeFormer attends to video patches with specific object information to answer questions, which also shows that VideoFormer extracts accurate spatial content from videos.

#### 4.6.2 Answering Verb Questions

We further visualize the cross-modality attention between verb question tokens and video tokens in Fig. 7. Three frames are sampled from a video and the verb phrase marked in blue is erased as the question. In Fig. 7 (a), when the verb “cutting” is erased, BridgeFormer focuses on the motion of the spoon on the pizza, and in Fig. 7 (b), when the verb “drinking” is erased, it follows the movement of the hand holding a cup of water. BridgeFormer focuses on object motions of video tokens to answer verb questions, which also shows that VideoFormer captures temporal dynamics of videos.

## 5. Conclusion

In this work, we introduce a novel pretext task, Multiple Choice Questions (MCQ) for video-text pre-training, which strengthens fine-grained semantic associations between local video and text features, and at the same time preserves high efficiency for retrieval. A parametric module BridgeFormer is trained to answer questions constructed by text features via resorting to video features, and can be readily removed for downstream tasks. Extensive evaluations on the text-to-video retrieval and zero-shot action recognition clearly show the great superiority of our method.

**Limitation.** (1) Off-the-shelf NLP models can not extract completely accurate noun and verb phrases for us to construct questions. (2) The text descriptions and corresponding videos may be actually misaligned in existing video-text datasets, leading to noisy supervision.

**Negative Social Impacts.** Since we do not filter out possible inappropriate videos (*e.g.*, of blood and violence) in the pre-training dataset, our model can be used to search terrible videos for spreading. Utilizing the pre-trained model to filter out those videos and re-training a model can help.

**Acknowledgment** Ping Luo is supported by the General Research Fund of HK No.27208720 and 17212120.

## References

- [1] Hassan Akbari, Linagzhe Yuan, Rui Qian, Wei-Hong Chuang, Shih-Fu Chang, Yin Cui, and Boqing Gong. Vatt: Transformers for multimodal self-supervised learning from raw video, audio and text. *arXiv preprint arXiv:2104.11178*, 2021.
- [2] Jean-Baptiste Alayrac, Adria Recasens, Rosalia Schneider, Relja Arandjelovic, Jason Ramapuram, Jeffrey De Fauw, Lucas Smaira, Sander Dieleman, and Andrew Zisserman. Self-supervised multimodal versatile networks. *NeurIPS*, 2(6):7, 2020.
- [3] Humam Alwassel, Dhruv Mahajan, Bruno Korbar, Lorenzo Torresani, Bernard Ghanem, and Du Tran. Self-supervised learning by cross-modal audio-video clustering. In *NeurIPS 2020*. NeurIPS, 2020.
- [4] Elad Amrani, Rami Ben-Ari, Daniel Rotman, and Alex Bronstein. Noise estimation using density estimation for self-supervised multimodal learning. In *AAAI*, volume 35, pages 6644–6652, 2021.
- [5] Lisa Anne Hendricks, Oliver Wang, Eli Shechtman, Josef Sivic, Trevor Darrell, and Bryan Russell. Localizing moments in video with natural language. In *ICCV*, pages 5803–5812, 2017.
- [6] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval, 2021.
- [7] Aman Chadha, Gurneet Arora, and Navpreet Kaloty. iperceive: Applying common-sense reasoning to multi-modal dense video captioning and video question answering. *arXiv preprint arXiv:2011.07735*, 2020.
- [8] Brian Chen, Andrew Rouditchenko, Kevin Duarte, Hilde Kuehne, Samuel Thomas, Angie Boggust, Rameswar Panda, Brian Kingsbury, Rogerio Feris, David Harwath, et al. Multimodal clustering networks for self-supervised learning from unlabeled videos. *arXiv preprint arXiv:2104.12671*, 2021.
- [9] David Chen and William B Dolan. Collecting highly parallel data for paraphrase evaluation. In *ACL*, pages 190–200, 2011.
- [10] Shizhe Chen, Yida Zhao, Qin Jin, and Qi Wu. Fine-grained video-text retrieval with hierarchical graph reasoning. In *CVPR*, pages 10638–10647, 2020.
- [11] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*, 2019.
- [12] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias

- Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2020.
- [13] Valentin Gabeur, Chen Sun, Karteek Alahari, and Cordelia Schmid. Multi-modal transformer for video retrieval. In *ECCV*, pages 214–229, 2020.
- [14] Deepti Ghadiyaram, Du Tran, and Dhruv Mahajan. Large-scale weakly-supervised pre-training for video action recognition. In *CVPR*, pages 12046–12055, 2019.
- [15] Simon Ging, Mohammadreza Zolfaghari, H Pirsavash, and Thomas Brox. Coot: Cooperative hierarchical transformer for video-text representation learning. In *NeurIPS*, 2020.
- [16] Tengda Han, Weidi Xie, and Andrew Zisserman. Memory-augmented dense predictive coding for video representation learning. In *ECCV*, pages 312–329, 2020.
- [17] Tengda Han, Weidi Xie, and Andrew Zisserman. Self-supervised co-training for video representation learning. *NeurIPS*, 33:5679–5690, 2020.
- [18] Rafal Jozefowicz, Oriol Vinyals, Mike Schuster, Noam Shazeer, and Yonghui Wu. Exploring the limits of language modeling. *arXiv preprint arXiv:1602.02410*, 2016.
- [19] Wonjae Kim, Bokyung Son, and Ildoo Kim. Vilt: Vision-and-language transformer without convolution or region supervision. *arXiv preprint arXiv:2102.03334*, 2021.
- [20] Quan Kong, Wenpeng Wei, Ziwei Deng, Tomoaki Yoshinaga, and Tomokazu Murakami. Cycle-contrast for self-supervised video representation learning. 2020.
- [21] Hildegard Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso Poggio, and Thomas Serre. Hmdb: a large video database for human motion recognition. In *ICCV*, pages 2556–2563. IEEE, 2011.
- [22] Jie Lei, Linjie Li, Luowei Zhou, Zhe Gan, Tamara L Berg, Mohit Bansal, and Jingjing Liu. Less is more: Clipbert for video-and-language learning via sparse sampling. In *CVPR*, pages 7331–7341, 2021.
- [23] Linjie Li, Yen-Chun Chen, Yu Cheng, Zhe Gan, Licheng Yu, and Jingjing Liu. Hero: Hierarchical encoder for video+ language omni-representation pre-training. In *EMNLP*, pages 2046–2065, 2020.
- [24] Yang Liu, Samuel Albanie, Arsha Nagrani, and Andrew Zisserman. Use what you have: Video retrieval using representations from collaborative experts. *arXiv preprint arXiv:1907.13487*, 2019.
- [25] Huaishao Luo, Lei Ji, Botian Shi, Haoyang Huang, Nan Duan, Tianrui Li, Jason Li, Taroon Bharti, and Ming Zhou. Univl: A unified video and language pre-training model for multimodal understanding and generation. *arXiv preprint arXiv:2002.06353*, 2020.
- [26] Huaishao Luo, Lei Ji, Ming Zhong, Yang Chen, Wen Lei, Nan Duan, and Tianrui Li. Clip4clip: An empirical study of clip for end to end video clip retrieval. *arXiv preprint arXiv:2104.08860*, 2021.
- [27] Tegan Maharaj, Nicolas Ballas, Anna Rohrbach, Aaron Courville, and Christopher Pal. A dataset and exploration of models for understanding video data through fill-in-the-blank question-answering. In *CVPR*, pages 6884–6893, 2017.
- [28] Antoine Miech, Jean-Baptiste Alayrac, Lucas Smaira, Ivan Laptev, Josef Sivic, and Andrew Zisserman. End-to-end learning of visual representations from uncensored instructional videos. In *CVPR*, pages 9879–9889, 2020.
- [29] Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In *ICCV*, pages 2630–2640, 2019.
- [30] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.
- [31] Mandela Patrick, Po-Yao Huang, Yuki Asano, Florian Metze, Alexander G Hauptmann, Joao F Henriques, and Andrea Vedaldi. Support-set bottlenecks for video-text representation learning. In *ICLR*, 2020.
- [32] AJ Piergiovanni, Anelia Angelova, and Michael S Ryoo. Evolving losses for unsupervised video representation learning. In *CVPR*, pages 133–142, 2020.
- [33] Jesús Andrés Portillo-Quintero, José Carlos Ortiz-Bayliss, and Hugo Terashima-Marín. A straightforward framework for video retrieval using clip. In *Mexican Conference on Pattern Recognition*, pages 3–12. Springer, 2021.
- [34] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. *arXiv preprint arXiv:2103.00020*, 2021.
- [35] Anna Rohrbach, Marcus Rohrbach, Niket Tandon, and Bernt Schiele. A dataset for movie description. In *CVPR*, pages 3202–3212, 2015.
- [36] Andrew Rouditchenko, Angie Boggust, David Harwath, Brian Chen, Dhiraj Joshi, Samuel Thomas, Kartik Audhkhasi, Hilde Kuehne, Rameswar Panda,

- Rogério Feris, et al. Avlnet: Learning audio-visual language representations from instructional videos. *arXiv preprint arXiv:2006.09199*, 2020.
- [37] Arka Sadhu, Kan Chen, and Ram Nevatia. Video question answering with phrases via semantic roles. *arXiv preprint arXiv:2104.03762*, 2021.
- [38] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*, 2019.
- [39] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *ACL*, pages 2556–2565, 2018.
- [40] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012.
- [41] Chen Sun, Fabien Baradel, Kevin Murphy, and Cordelia Schmid. Learning video representations using contrastive bidirectional transformer. *arXiv preprint arXiv:1906.05743*, 2019.
- [42] Chen Sun, Austin Myers, Carl Vondrick, Kevin Murphy, and Cordelia Schmid. Videobert: A joint model for video and language representation learning. In *ICCV*, pages 7464–7473, 2019.
- [43] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. In *CVPR*, pages 6450–6459, 2018.
- [44] Michael Wray, Diane Larlus, Gabriela Csurka, and Dima Damen. Fine-grained action retrieval through multiple parts-of-speech embeddings. In *ICCV*, pages 450–459, 2019.
- [45] Hu Xu, Gargi Ghosh, Po-Yao Huang, Prahal Arora, Masoumeh Aminzadeh, Christoph Feichtenhofer, Florian Metze, and Luke Zettlemoyer. Vlm: Task-agnostic video-language model pre-training for video understanding. *arXiv preprint arXiv:2105.09996*, 2021.
- [46] Hu Xu, Gargi Ghosh, Po-Yao Huang, Dmytro Okhonko, Armen Aghajanyan, Florian Metze, Luke Zettlemoyer, and Christoph Feichtenhofer. Videoclip: Contrastive pre-training for zero-shot video-text understanding. *arXiv preprint arXiv:2109.14084*, 2021.
- [47] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. Msr-vtt: A large video description dataset for bridging video and language. In *CVPR*, pages 5288–5296, 2016.
- [48] Ran Xu, Caiming Xiong, Wei Chen, and Jason Corso. Jointly modeling deep video and compositional text to bridge vision and language in a unified framework. In *AAAI*, volume 29, 2015.
- [49] Antoine Yang, Antoine Miech, Josef Sivic, Ivan Laptev, and Cordelia Schmid. Just ask: Learning to answer questions from millions of narrated videos. In *ICCV*, pages 1686–1697, 2021.
- [50] Jianwei Yang, Yonatan Bisk, and Jianfeng Gao. Taco: Token-aware cascade contrastive learning for video-text alignment. In *ICCV*, pages 11562–11572, 2021.
- [51] Linchao Zhu and Yi Yang. Actbert: Learning global-local video-text representations. In *CVPR*, pages 8746–8755, 2020.
- [52] Dimitri Zhukov, Jean-Baptiste Alayrac, Ramazan Gokberk Cinbis, David Fouhey, Ivan Laptev, and Josef Sivic. Cross-task weakly supervised learning from instructional videos. In *CVPR*, pages 3537–3545, 2019.