Zero-Shot Video Question Answering via Frozen Bidirectional Language Models
Antoine Yang, Antoine Miech, Josef Sivic, Ivan Laptev, Cordelia Schmid

To cite this version:

HAL Id: hal-03807016
https://inria.hal.science/hal-03807016v2
Submitted on 10 Oct 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
Abstract

Video question answering (VideoQA) is a complex task that requires diverse multi-modal data for training. Manual annotation of question and answers for videos, however, is tedious and prohibits scalability. To tackle this problem, recent methods consider zero-shot settings with no manual annotation of visual question-answer. In particular, a promising approach adapts frozen autoregressive language models pretrained on Web-scale text-only data to multi-modal inputs. In contrast, we here build on frozen bidirectional language models (BiLM) and show that such an approach provides a stronger and cheaper alternative for zero-shot VideoQA. In particular, (i) we combine visual inputs with the frozen BiLM using light trainable modules, (ii) we train such modules using Web-scraped multi-modal data, and finally (iii) we perform zero-shot VideoQA inference through masked language modeling, where the masked text is the answer to a given question. Our proposed approach, FrozenBiLM, outperforms the state of the art in zero-shot VideoQA by a significant margin on a variety of datasets, including LSMDC-FiB, iVQA, MSRVTT-QA, MSVD-QA, ActivityNet-QA, TGIF-FrameQA, How2QA and TVQA. It also demonstrates competitive performance in the few-shot and fully-supervised setting. Our code and models are publicly available at [1].

1 Introduction

Video question answering (VideoQA) is a challenging task that requires fine-grained multi-modal understanding. State-of-the-art approaches to VideoQA [49, 119, 121] rely on large video datasets manually annotated with question-answer pairs. Yet, collecting such annotations is time consuming, expensive and therefore not scalable. This has motivated the development of zero-shot VideoQA approaches [113, 114, 122], that use no visual question-answer annotation for training; see Figure 1.

Recently, a promising line of work builds on frozen large autoregressive language models [22, 75, 102, 107, 116, 123] for zero-shot visual question answering. This has been motivated by the findings from GPT-3 [8] which exhibits strong zero-shot text-only question answering abilities from large

---

**Antoine Yang**1,2, **Antoine Miech**3, **Josef Sivic**4, **Ivan Laptev**1,2, **Cordelia Schmid**1,2
1Inria Paris 2Département d’informatique de l’ENS, CNRS, PSL Research University 3DeepMind 4CIIRC CTU Prague

https://antoyang.github.io/frozenbilm.html

---

Figure 1: Our model FrozenBiLM builds on a pretrained and frozen bidirectional language model (BiLM), and is trained from Web-scraped video-caption pairs. FrozenBiLM excels in the zero-shot video question answering task without using any explicit visual question-answer supervision.

1 Introduction

Video question answering (VideoQA) is a challenging task that requires fine-grained multi-modal understanding. State-of-the-art approaches to VideoQA [49, 119, 121] rely on large video datasets manually annotated with question-answer pairs. Yet, collecting such annotations is time consuming, expensive and therefore not scalable. This has motivated the development of zero-shot VideoQA approaches [113, 114, 122], that use no visual question-answer annotation for training; see Figure 1.

Recently, a promising line of work builds on frozen large autoregressive language models [22, 75, 102, 107, 116, 123] for zero-shot visual question answering. This has been motivated by the findings from GPT-3 [8] which exhibits strong zero-shot text-only question answering abilities from large

---

4Czech Institute of Informatics, Robotics and Cybernetics at the Czech Technical University in Prague.
autoregressive language models. Such models [8, 79, 92, 103] can predict an arbitrarily long sequence of text, one token at each step from left to right. However, they usually require billion parameters to work well, making them computationally expensive to train, and challenging to deploy in practice.

In contrast, recent work in natural language [72, 86, 87, 98] demonstrates strong zero-shot performance for lighter bidirectional language models (BiLM). Such models [20, 28, 39, 48, 68, 85] can predict a few masked tokens in an input sequence given left and right context in a single forward pass. These works cast downstream tasks in cloze form [101], similar to the masked language modeling task (MLM) [20] solved by these models at pretraining. This motivates us to tackle diverse zero-shot multi-modal tasks (open-ended VideoQA [109], multiple-choice VideoQA [52] and fill-in-the-blank [73]) by formulating them in cloze form and leveraging the text-only knowledge of pretrained BiLM.

To adapt a pretrained BiLM to multi-modal inputs, we combine it with a frozen pretrained visual backbone and a set of lightweight additional modules including adapters [31]. We train these modules on Web-scrapped video-text data using a simple visually-conditioned MLM loss. We preserve the uni-modal knowledge of a BiLM by freezing its weights. To our knowledge, our approach is the first to explore the zero-shot visual-linguistic capabilities of frozen non-autoregressive language models.

We show that our approach largely improves the state of the art on various zero-shot VideoQA benchmarks. Furthermore, we demonstrate that frozen bidirectional language models perform better while being cheaper to train than frozen autoregressive language models [102]. Moreover, our ablation studies show (i) the ability of our model to effectively perform zero-shot multi-modal reasoning using both visual cues and speech transcripts, (ii) the importance of adapters combined with frozen pretrained language models, (iii) the impact of multi-modal data scale, (iv) the impact of the language model size and of bidirectional modeling. Our approach also performs competitively in the fully-supervised setting. Indeed, we show the benefits of freezing the weights of a BiLM when using VideoQA training data, while updating considerably less parameters compared to alternative methods. Finally, we introduce a new few-shot VideoQA task in which we finetune our pretrained model on a small fraction of the downstream training dataset, and show promising results in this setting.

In summary, our contributions are three-fold:

(i) We present FrozenBiLM, a framework that handles multi-modal inputs using frozen bidirectional language models and enables zero-shot VideoQA through masked language modeling.

(ii) We provide an extensive ablation study and demonstrate the superior performance of our framework in the zero-shot setting when compared to previous autoregressive models.

(iii) Our approach improves the state of the art in zero-shot VideoQA by a significant margin. FrozenBiLM also demonstrates competitive performance in the fully-supervised setting and shows strong results in the few-shot VideoQA setting which we introduce.

Our code and trained models are publicly available at [1].

2 Related Work

Zero-shot VideoQA. A vast majority of VideoQA approaches rely on relatively small, manually annotated VideoQA datasets [3, 10, 11, 16–18, 23, 26, 27, 33, 36, 37, 40–43, 49, 50, 53, 64, 67, 76, 77, 83, 88, 89, 93, 100, 112, 115, 117, 124, 128]. Recently, a few work [113, 122] have explored zero-shot approaches for VideoQA, where models are only trained on automatically mined video clips with short text descriptions. In contrast to VideoQA annotations, such video-text pairs are readily-available at scale on the Web [6, 74, 121]. In particular, Yang et al. [113] automatically generate VideoQA training data using language models [79] pretrained on a manually annotated text-only question-answer corpus [80]. Reserve [122] uses GPT-3 [8] to rephrase questions into sentences completed by a multi-modal model. In contrast to these prior works [113, 122], our method does not require any kind of explicitly annotated language dataset or the use of data generation pipelines for zero-shot VideoQA. Note that BLIP [59] studies a related setting where a model trained on manually annotated image-question-answer triplets is transferred to VideoQA, which is a less challenging task. Also note that VideoCLIP [110] considers a related zero-shot multiple-choice video-to-text retrieval task as VideoQA, but in this setting the model is not provided with natural language questions.

"Cloze test" is an exercise test where certain portions of text are occluded or masked and need to be filled-in.
Visual language models. As language models require large amounts of training data to perform well [30], recent works have studied transferring pretrained language models [8, 105] to image-text tasks. VisualGPT [13] and VC-GPT [71] showed the benefit of initializing the weights of an image captioning model with a pretrained autoregressive language-only model. Recent work pushed this idea further by freezing the weights of a pretrained autoregressive language model for tackling vision and language tasks [2, 22, 75, 102, 107, 116, 123]. Our approach also leverages a frozen pretrained language model. Similar to MAGMA [22], we also use adapter layers [31, 32]. However, we differ from these approaches as we propose to instead use lighter bidirectional masked language models, instead of autoregressive ones, and rely on a masked language modeling objective (MLM) instead of an autoregressive one. Moreover, our model is specifically designed for videos, for which high-quality visual question answering annotation is even more scarce compared to still images [22, 75, 102, 116]. We also explore the use of the speech modality, and tackle tasks which are challenging for autoregressive language models such as video-conditioned fill-in-the-blank [73]. Finally we show in Section 4.3 the superior performance of frozen bidirectional language models in comparison with autoregressive ones [102].

Masked Language Modeling in vision and language. The MLM objective was initially introduced in natural language [20, 48, 68] to pretrain bidirectional transformers and learn generic representations. This approach achieved state-of-the-art results in many language tasks after finetuning on downstream datasets. Its success inspired numerous works to adapt it to train multi-modal transformer models on paired visual-linguistic data [15, 24, 25, 29, 34, 44, 54, 57, 62, 60, 65, 58, 56, 69, 70, 90, 91, 96, 97, 99, 104, 106, 118, 121, 126, 127]. However, these works typically use it to learn generic visual-linguistic representations by updating the transformer weights, and then use expensive manual supervision to train randomly initialized task-specific answer classifiers for VQA [15, 25, 57, 58, 65, 69, 90, 91, 96, 99, 106, 118] or VideoQA [24, 54, 56, 104, 121]. In contrast, we tackle zero-shot VideoQA, i.e. without using any manual annotation. Moreover, we do not update the transformer weights during cross-modal training, but instead exhibit the benefits of freezing these weights after text-only pretraining, for both zero-shot and fully-supervised VideoQA (see Sections 4.2 and 4.5).

3 Method

This section presents our approach to tackle zero-shot video question answering. Here, zero-shot means that we do not use any visual question answering annotation and only rely on scalable data from the Web. Our approach starts with two strong pretrained components: (i) a text-only bidirectional masked language model (BiLM) pretrained on data from the Internet, which has the capability of zero-shot question answering but is not capable of visual reasoning, and (ii) a vision encoder pretrained to map images to text descriptions, but which does not have the ability to perform visual question answering. We aim at connecting these two components while keeping the language component frozen to avoid catastrophic forgetting [19], where the large language model would specialize to a new task while forgetting its initial capabilities. The end-goal is to design a unified model having the best of both worlds: visual understanding capabilities of a powerful visual encoder and question answering capabilities of a powerful language model. This requires several technical innovations, which are described in the rest of this section. First, we explain in Section 3.1 how we augment a frozen pretrained bidirectional masked language model with new layers to enable joint video and language reasoning, see Figure 2. Second, we present in Section 3.2 how we train these layers on video-text data scraped from the Web [6]. Finally, we describe in Section 3.3 how we enable zero-shot predictions for several video-language downstream tasks, including open-ended VideoQA, by casting them in a cloze form, similar to the masked language modeling task solved during training.

3.1 Architecture

The proposed architecture, illustrated in Figure 2, brings together a powerful frozen pretrained bidirectional language model with a strong visual encoder. The difficulty lies in enabling multi-modal reasoning while keeping the large language model frozen. To address this challenge, we unify these two models via a visual-to-text projection module together with small adapter modules inserted within the frozen language model. Next, we describe in more detail the three main components of the architecture: (i) the frozen pretrained bidirectional language model, (ii) the pretrained video encoder and (iii) the lightweight modules that seamlessly connect the two components.
**Frozen Bidirectional Masked Language Model.** Our method starts from a pretrained bidirectional language model based on a Transformer encoder [103]. The input text is decomposed into a sequence of tokens $x = \{x_i\}_t^T \in [1, V]^L$ by a tokenizer of a vocabulary size $V$. The language model, parameterized by $\theta$, makes use of an embedding function $g_\theta$ which independently transforms each token into a $D$-dimensional continuous embedding $t = \{t_i\}_t^T := \{g_\theta(x_i)\}_t^T \in \mathbb{R}^T \times D$, a Transformer encoder $f_\theta$ which computes interactions between all input tokens and outputs contextualized representations $t' = \{t'_i\}_t^T$, and a masked language modeling (MLM) classifier head $m_\theta$ which independently maps the $D$-dimensional continuous embedding for each token $t'_i$ to a vector of logits parameterizing a categorical distribution over the vocabulary $V$. This distribution is referred to by $\log p_\theta(x) := \{m_\theta(t'_i)\}_t^T \in \mathbb{R}^T \times V$. We assume that the language model is pretrained, i.e. $\theta$ has been optimised with a standard MLM objective [20] on a large dataset of text from the Web. We show in Section 4.2 that this text-only pretraining has a crucial importance for zero-shot VideoQA.

**Pretrained Video Encoder.** The video is represented by a sequence of frames $y = \{y_i\}_t^T$. Each frame is forwarded separately through a visual backbone $h_\phi$, which outputs one feature vector per frame $u = \{u_i\}_t^T := \{h_\phi(y_i)\}_t^T \in \mathbb{R}^T \times D_u$. In detail, the visual backbone is CLIP ViT-L/14 [21, 78] at resolution 224 × 224 pixels, pretrained to map images to text descriptions with a contrastive loss on 400M Web-scraped image-text pairs. The backbone is kept frozen throughout our experiments. Note that a CLIP-baseline for zero-shot VideoQA results in poor performance, see Section 4.4.

**Connecting the Frozen Language and Frozen Vision components.** The video features are incorporated into the language model as a prompt [55, 63, 125] $v$ of length $T$ (Figure 2, left). This prompt is obtained by linearly mapping the visual features $u$ to the text token embedding space via a visual-to-text projection $P \in \mathbb{R}^{D_u \times D}$, i.e. $v = \{v_i\}_t^T := \{P(u_i)\}_t^T$. The prompt is then concatenated with the text embeddings before being forwarded to the transformer encoder that models joint visual-linguistic interactions. We show in Section 4.2 that incorporating the input video considerably improves zero-shot VideoQA results. In addition, to learn powerful multi-modal interactions while keeping the transformer encoder weights frozen, we equip the transformer encoder with lightweight adapter modules $A$ [31] (Figure 2, right). We use an adapter which transforms the hidden state $z$ with a multi-layer perceptron transformation and a residual connection, i.e. $A(z) = z + W^\text{up} \psi(W^\text{down} z)$ with $W^\text{down} \in \mathbb{R}^{D \times D_h}$, $W^\text{up} \in \mathbb{R}^{D_h \times D}$, $D$ the hidden dimension of the transformer, $D_h$ the bottleneck dimension, and $\psi$ a ReLU activation function. $D_h$ is typically set to be smaller than $D$ such that the adapters are lightweight. In detail, we add an adapter module before the layer normalization, after each self-attention layer and each feed-forward layer of the transformer encoder.
3.2 Cross-modal training

We wish to train the newly added modules introduced in the previous section (shown in orange in Figure 2) for the VideoQA task. This is hard because we assume that no explicit manual annotation for the VideoQA task is available, such annotations being expensive and therefore hard to obtain at scale. Instead we train our architecture using only readily-available video-caption pairs scraped from the Web. Such data is easy to obtain [6, 74, 121], ensuring the scalability of our approach.

During training, we keep the weights of the pretrained BiLM and pretrained visual backbone frozen as previously explained. We train from scratch the parameters of (i) the visual-to-text projection module \( P \) and (ii) the adapter modules \( A \). We show in Section 4.2 the importance of freezing the BiLM weights combined with training the adapter modules. Note that all normalization layers [5] of the pretrained BiLM are also updated to adjust to the new distribution of the training data. We denote all the trainable parameters of our model by the subscript \( \mu \). In practice, they sum up to about 5% of the BiLM parameters, hence the training of our model is computationally efficient.

We use a visually-conditioned masked language modeling objective (MLM), in which some text tokens \( \{x_m\} \) are randomly masked and the model has to predict these tokens based on the surrounding text tokens and the video input. Formally, we minimize the following loss:

\[
L_{\mu}(x, y) = -\frac{1}{M} \sum_m \log p_{\mu}(\tilde{x}, y)_{x_m}^{x_m},
\]

where \( \tilde{x} \) is the corrupted text sequence, \( y \) is the sequence of video frames, \( p_{\mu}(\tilde{x}, y)_{x_m}^{x_m} \) is the probability for the (masked) \( m \)-th token in \( \tilde{x} \) to be \( x_m \), and \( M \) is the number of masks in the sequence \( \tilde{x} \). In detail, we follow [20] and corrupt 15% of text tokens, replacing them 80% of the time with a mask token, 10% of the time with the same token and 10% of the time with a randomly sampled token.

3.3 Adapting to downstream tasks

After training, our model is able to fill gaps in the input text given an input video together with left and right textual context as part of the input text. We wish to apply our model out-of-the-box to predict an answer given a question about a video. The video can optionally come with textual subtitles obtained using automatic speech recognition. To avoid using manual supervision, we formulate the downstream tasks in cloze form [86, 101], i.e. such that the model only has to fill-in a mask token in the input prompt similarly to the MLM objective optimized during training. The adaptation to the downstream tasks brings several challenges, as described next. First, we describe how we formulate the input text prompts for several downstream tasks. Then, we explain how we map the mask token from the input text prompt to an answer via a frozen answer embedding module. Finally, we present how we finetune our architecture in a supervised setting.

**Input prompt engineering.** We describe how we design the input text prompts for several downstream video-language tasks. Each downstream task is formulated as a masked language modeling problem. This allows us to apply FrozenBiLM out-of-the-box. A [CLS] token and a [SEP] token are respectively inserted at the start and the end of each sequence following [20].

*Open-ended VideoQA.* Given a question and a video, the task is to find the correct answer in a large vocabulary \( \mathcal{A} \) of about 1K answers. Answers are concise, i.e. the great majority of answers consist of one word [35, 109, 113, 120]. We design the following prompt:

```
```

*Multiple-choice VideoQA.* Given a question and a video, the task is to find the correct answer in a small number of candidates \( C \), typically up to 5 choices [52, 60]. We set the vocabulary to \( \mathcal{A} = \{Yes, No\} \) and compute a confidence score for each candidate by using the following prompt:

```
```

We choose the best option by selecting the candidate with the highest Yes logit value.

*Video-conditioned fill-in-the-blank task.* Given a video and a sentence with a blank space, the task is to fill in the blank with the correct word from a vocabulary \( \mathcal{A} \) of about 1K answers. We replace the blank in the sentence with a mask token, and design the following prompt:

```
```


Note that all prompts are prepended with the video prompt (see Section 3.1) before being forwarded to the transformer encoder.

**Answer embedding module.** For each downstream task, we wish to map the mask token in the input text prompt to an actual answer prediction in the set of possible answers \( \mathcal{A} \), as described above. For this we use the *frozen* MLM classifier head \( m_\theta \). However, \( m_\theta \in \mathbb{R}^{V \times D} \) covers \( V \) different tokens where \( V \gg N \) and \( N \approx 1,000 \) is the size of \( \mathcal{A} \). Therefore, we introduce a task-specific answer classification head \( l \) which linearly maps a contextualized mask representation \( t'_i \) to a vector of logits parameterizing a categorical distribution over the vocabulary \( \mathcal{A} \), i.e. \( l \in \mathbb{R}^{N \times D} \). We set the weights of this answer module \( l \) with the corresponding weights of the pretrained MLM classifier \( m_\theta \) for one-token answers. In the case of multi-token answers, we average the weights of their different tokens. We, hence, enable zero-shot inference at test time. We also discuss other alternative strategies to handle multi-token answers in Appendix Section D.6.

**Fully-supervised training.** To evaluate our approach on fully-supervised benchmarks, we also explore finetuning of our model on datasets that provide manual annotations for the target task. To this end, we train the same parameters as explained in Section 3.2, while keeping the transformer weights and the answer embedding module *frozen*. For open-ended VideoQA and video-conditioned fill-in-the-blank, we use a cross-entropy loss on the task-specific vocabulary \( \mathcal{A} \). For multiple-choice VideoQA, we use a binary cross-entropy loss applied to each answer candidate. We show in Section 4.5 the benefit of *freezing* the language model weights during fully-supervised training.

### 4 Experiments

This section demonstrates the benefits of our *FrozenBiLM* framework and compares our method to the state of the art. We first outline our experimental setup in Section 4.1. We then present ablation studies in Section 4.2. Next we compare our bidirectional framework to its autoregressive variant in Section 4.3. The comparison to the state of the art in zero-shot VideoQA and qualitative results are presented in Section 4.4. Finally, we finetune our model on the VideoQA task in Section 4.5, where we show few-shot and fully-supervised results.

#### 4.1 Experimental setup

**Frozen bidirectional language model.** We use a tokenizer based on SentencePiece [47] with \( V = 128,000 \), and a bidirectional language model with 900M parameters, DeBERTa-v2-XLarge [28], trained with the MLM objective on a corpus of 160G text data. We also show how our approach generalizes to other MLM-pretrained bidirectional language models such as BERT [20] in Section 4.2.

**Datasets.** For training we use the publicly available WebVid10M dataset [6], which consists of 10 million of video-text pairs scraped from the Shutterstock website where video captions are obtained from readily-available alt-text descriptions. We evaluate results on eight downstream datasets covering a wide range of textual and video domains (e.g. GIFs, YouTube videos, TV shows, movies), and multiple VideoQA paradigms: open-ended VideoQA (iVQA [113], MSRVT-TQA [109], MSVD-QA [109], ActivityNet-QA [120] and TGIF-QA FrameQA [35]), multiple-choice VideoQA (How2QA [60] and TVQA [52]) and video-conditioned fill-in-the-blank (LSMDC-Fill-in-the-blank [73]). Unless stated otherwise, we report top-1 test accuracy using the original splits for training, validation and test. For How2QA, we report results on the public validation set for comparison with prior work [88, 113, 119]. For TVQA, we report results on the validation set for the ablation studies and on the hidden test set for the comparison to the state of the art. More details are included in Appendix Section C.1.

**Implementation Details.** The training for 2 epochs on WebVid10M lasts 20 hours on 8 Tesla V100 GPUs. We give further details in Appendix Section C.2.

#### 4.2 Ablation studies

In this section, we evaluate the zero-shot performance of different variants of our method. By default, we use the *frozen* pretrained DeBERTa-v2-XLarge language model and train the visual-to-text-
projection layer together with adapters for 2 epochs on WebVid10M. We refer to this default model as \textit{FrozenBiLM}. This model uses three input modalities in terms of video, question, and speech.

**Ablation of the model training.** We ablate the effect of initializing parameters of the language model, freezing its weights and training adapters in Table 1. We observe that the language model pretraining is crucial. Indeed, a model with randomly initialized language weights (row 1) performs poorly compared to models initialized with language pretrained weights (rows 2 to 4). Moreover, the model which updates the language model weights (row 2) during cross-modal training performs considerably worse compared to variants that freeze them (rows 3 and 4). This shows the benefit of freezing the language model for zero-shot VideoQA. We also notice the benefit of the adapter layers by comparing rows 3 and 4, especially for multiple-choice datasets. Finally, we note that training variants with the frozen language model is twice faster compared to updating all parameters, as there is a significantly lower number of parameters to be trained.

**Impact of modalities.** Table 2 shows the impact of the visual and speech modalities on the zero-shot performance of our model. First, we evaluate the text-only performance of our model using neither visual input nor speech input in row 1. We can observe that adding speech (row 2) marginally improves the results and that the importance of speech highly depends on the dataset. When adding vision (rows 3 and 4), the performance increases significantly, e.g. +13.6% accuracy on iVQA and +22.1% on MSVD-QA between rows 4 and 2. Finally, the model with vision also benefits from the speech, e.g. +16.5% accuracy on How2QA and +29.5% accuracy on TVQA (compare rows 3 and 4).

Note that in practice, speech is missing for many videos, as we obtain the speech directly from the YouTube API and many videos are no longer available. Exceptions are How2QA and TVQA for which the authors [52, 61] provide speech for all videos. Consequently, we have speech data for only 44.3%, 14.2%, 8.2%, 7.1% and 25.3% of test samples in LSMDC-FiB, iVQA, MSRVTT-QA, MSVD-QA and ActivityNet-QA respectively. GIFs in TGIF-QA do not contain speech.

**Size of the cross-modal training dataset.** Zero-shot results of \textit{FrozenBiLM} after training for a fixed number of iterations on different fractions of WebVid10M are shown in Table 3. We construct these subsets such that larger subsets include the smaller ones. We find that performance increases monotonically with more multi-modal training data.

**Size of the language model.** In Table 4, we ablate the importance of the language model size for the zero-shot performance. Note that when comparing different language models, we use no adapters to avoid biases related to the choice of the bottleneck dimension hyperparameter [31]. We find that using the 900M-parameter DeBERTA-V2-XLarge (row 6) outperforms the 300M-parameter BERT-Large (row 5) which also improves over the 100M-parameter BERT-Base (row 4).
Table 4: Comparison of autoregressive language models (top) and bidirectional language models (bottom) for zero-shot VideoQA. All variants are trained on WebVid10M for the same number of epochs.

<table>
<thead>
<tr>
<th>Method</th>
<th>Language Model</th>
<th># LM params</th>
<th>Train time (GPUH)</th>
<th>VQA MSRVTT-QA</th>
<th>MSVD-QA</th>
<th>ActivityNet-QA</th>
<th>TGIF-QA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Autoregressive</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1. GPT-Neo-1.3B</td>
<td>1.3B</td>
<td>200</td>
<td>6.6</td>
<td>4.2</td>
<td>10.1</td>
<td>17.8</td>
<td>14.4</td>
</tr>
<tr>
<td>2. GPT-Neo-2.7B</td>
<td>2.7B</td>
<td>360</td>
<td>9.1</td>
<td>7.7</td>
<td>17.8</td>
<td>17.4</td>
<td>20.1</td>
</tr>
<tr>
<td>3. GPT-J-6B</td>
<td>6B</td>
<td>820</td>
<td>21.4</td>
<td>9.6</td>
<td>26.7</td>
<td>24.5</td>
<td>37.3</td>
</tr>
<tr>
<td>Bidirectional</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4. BERT-Base</td>
<td>110M</td>
<td>24</td>
<td>12.4</td>
<td>6.4</td>
<td>11.7</td>
<td>16.7</td>
<td>23.1</td>
</tr>
<tr>
<td>5. BERT-Large</td>
<td>340M</td>
<td>50</td>
<td>12.9</td>
<td>7.1</td>
<td>13.0</td>
<td>19.0</td>
<td>21.5</td>
</tr>
<tr>
<td>6. DeBERTa-V2-XLarge</td>
<td>890M</td>
<td>160</td>
<td>27.3</td>
<td>16.8</td>
<td>32.2</td>
<td>24.7</td>
<td>41.0</td>
</tr>
</tbody>
</table>

4.3 Comparison with frozen autoregressive models

In this section, we compare our bidirectional framework using language models of various sizes to the larger, autoregressive GPT-based counterparts recently used for zero-shot image question answering [102, 116]. For fair comparison, we adapt autoregressive models to video and language inputs similarly as our bidirectional models. In detail, autoregressive variants train a similar visual-to-text projection by using a left-to-right language modeling loss [102]. All models in our comparison are trained on WebVid10M for the same number of epochs. At inference, autoregressive variants use the same template as [102] to which we prepend speech subtitles, greedily decode sequences as [102], and use the same answer vocabulary as bidirectional models. Autoregressive variants select the top answer that maximizes the log-likelihood when appended to the question prompt. Here also, we use no adapters for all models, such that the architecture of autoregressive models closely follows [102]. This is to avoid biases related to the tuning of the bottleneck reduction hyperparameter in the adapters [31].

We compare autoregressive and bidirectional language models in terms of accuracy and efficiency in Table 4. We observe that our bidirectional framework (rows 4-6) achieves significantly better zero-shot performance-efficiency trade-off compared to its autoregressive counterpart (rows 1-3). For instance, our framework with BERT-Base [20] (row 4) outperforms the autoregressive variant based on GPT-Neo-1.3B [7] (row 1) which uses 12 times more parameters and 8 times more training time. Likewise, our framework with DeBERTa-V2-XLarge [28] (row 6) improves over the autoregressive variant based on GPT-J-6B [105] (row 3) that has 7 times more parameters and requires 5 times more training time, showing the efficiency of our bidirectional framework for zero-shot VideoQA.

4.4 Comparison to the state of the art for zero-shot VideoQA

Quantitative comparison. Table 5 presents results of our method in comparison to the state of the art in zero-shot VideoQA settings [113], i.e. when using no manually annotated visual data for training.

<table>
<thead>
<tr>
<th>Method</th>
<th>Training Data</th>
<th>Speech</th>
<th>Fill-in-the-blank</th>
<th>Open-ended</th>
<th>Multiple-choice</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>LSMDC</td>
<td>VQA MSRVTT-QA</td>
<td>MSVD-QA</td>
</tr>
<tr>
<td>Random</td>
<td></td>
<td></td>
<td>0.1</td>
<td>0.1</td>
<td>0.1</td>
</tr>
<tr>
<td>CLIP ViT-L/14 [78]</td>
<td>400M image-texts</td>
<td>✗</td>
<td>1.2</td>
<td>9.2</td>
<td>2.1</td>
</tr>
<tr>
<td>Just Ask [114]</td>
<td>HowToVQA69M+</td>
<td>✗</td>
<td>—</td>
<td>13.3</td>
<td>5.6</td>
</tr>
<tr>
<td>Reserve [122]</td>
<td>YTC-BiLM</td>
<td>✗</td>
<td>31.0</td>
<td>—</td>
<td>5.8</td>
</tr>
<tr>
<td>FrozenBiLM (Ours)</td>
<td>WebVid10M</td>
<td>✓</td>
<td>35.9</td>
<td>26.2</td>
<td>16.9</td>
</tr>
<tr>
<td>FrozenBiLM (Ours)</td>
<td>WebVid10M</td>
<td>✓</td>
<td>35.5</td>
<td>26.8</td>
<td>16.7</td>
</tr>
</tbody>
</table>

Table 5: Comparison with the state of the art for zero-shot VideoQA.

Importance of the suffix. Our text input prompts include a suffix just to the right of the mask token which consists in a point and an end-of-sentence token for the variant without speech (or a point followed by the speech subtitles for the variant with speech). We found that removing this suffix leads to a considerable drop of performance (e.g. the test accuracy on MSVD-QA in the row 3 of Table 2 drops from 33.7% to 2.8%). Note that we do not observe such a large drop in performance when removing the [CLS] token e.g. the accuracy on MSVD-QA drops only from 33.8% to 33.2%. This shows that the bidirectional nature of our framework is a key factor for the performance. Intuitively, this suffix forces the model to provide a concise answer. Such a hard constraint cannot be given to unidirectional autoregressive models compared next in Section 4.3. We further ablate the importance of the prompt design in Appendix Section D.7.
Figure 3: Zero-Shot VideoQA. Qualitative comparison between Just Ask [114] (row 3 in Table 5), our model (row 4 in Table 5), its unfrozen variant (row 2 in Table 1) and its text-only variant (row 2 in Table 2). The first two examples are from iVQA [113] and the last three examples are from ActivityNet-QA [120].

Table 6: Comparison with the state of the art, and the variant UnFrozenBiLM which does not freeze the language model weight, on fully-supervised benchmarks. * denotes results obtained with speech input.

Our approach outperforms previous methods by a significant margin on all 8 datasets. In particular, FrozenBiLM outperforms Reserve [122], which is trained on one billion YouTube video clips jointly with vision, language and sound, Just Ask [114], which uses large-scale automatically generated VideoQA data, and a CLIP baseline [78] matching the text concatenating question and answer to the middle frame of the video. Note that FrozenBiLM performs competitively even when using no speech input. Finally, we note that BLIP [59] has a different definition of zero-shot where a network finetuned on the image-VQA dataset [4] is evaluated directly on VideoQA datasets. Our Appendix presents results where we outperform BLIP [59] in their settings (Section D.1) and also includes an analysis of results by question type (Section D.3). In summary, our evaluation shows the excellent performance of our model in the challenging zero-shot setup.

Qualitative results. Figure 3 illustrates qualitative results of zero-shot VideoQA for our FrozenBiLM model and compares them to Just Ask [114], as well as to variants of our approach that do not freeze the language model (UnFrozenBiLM) and use no visual modality (text-only), as evaluated in Section 4.2. We observe that the unfrozen variant can predict answers that lack text-only commonsense reasoning, e.g., in the third example, it is unlikely that a sitting man is swimming. The text-only variant does have strong language understanding, but makes visually-unrelated predictions. In contrast, consistently with our quantitative results, our model FrozenBiLM is able to correctly answer various questions, showing both a strong textual commonsense reasoning and a complex multi-modal understanding. We show additional qualitative results in Appendix Section A.

4.5 Freezing the BiLM is also beneficial in supervised settings

Fully-supervised VideoQA. We next present an evaluation in a supervised setup where we finetune FrozenBiLM on a downstream VideoQA task. We emphasize that we also keep our pretrained language
model weights frozen all throughout finetuning. As shown in Table 6, our approach improves the state of the art on LSMDC-FiB, iVQA, MSRVTT-QA, MSVD-QA, ActivityNet-QA and How2QA. In particular, FrozenBiLM outperforms strong recent baselines such as All-in-one [104] on 2/3 datasets, VIOLET [24] on 3/4 datasets and MERLOT [121] on 4/5 datasets. Our approach has significantly less trainable parameters compared to the state of the art [24, 104, 121] as we freeze the weights of the pretrained language model. We ablate this major difference in Table 6, and find that our FrozenBiLM with the frozen language model performs better and trains twice faster compared to UnFrozenBiLM where we update the language model during training. This shows that freezing the language model is not only beneficial for zero-shot but also in fully-supervised settings, therefore suggesting that our FrozenBiLM framework also provides a parameter-efficient solution for VideoQA training. We also note that FrozenBiLM performs competitively even without speech input, although speech helps significantly for the performance on LSMDC, How2QA and TVQA.

**Few-shot VideoQA.** The low number of trainable parameters when training FrozenBiLM makes it particularly well-suited in the low data regime. To verify this, we explore a few-shot VideoQA setting where we finetune our pretrained model using varying fractions of VideoQA training data. From Table 7 we observe significant improvements over zero-shot when using only 1% of training data. Finally, we show in Appendix Section D.5 that freezing the BiLM highly benefits the few-shot performance, consistently with the results in the zero-shot and fully-supervised settings.

5 Conclusion

We have presented FrozenBiLM, a framework that extends frozen bidirectional language models to multi-modal inputs by training additional modules on Web-scraped data, and that tackles zero-shot VideoQA through masked language modeling. We have provided extensive ablation studies and shown the efficiency of our framework compared to its autoregressive variant. FrozenBiLM improves the state-of-the-art zero-shot VideoQA on various datasets, performs competitively in fully-supervised settings and exhibits strong performance in the few-shot VideoQA setting we newly introduce.

**Limitations.** Promising directions not explored in this work include scaling the size of a bidirectional language model to several billion parameters, and additional training on large datasets of YouTube videos with accompanying speech transcripts and/or audio [122]. Also, our model cannot be applied out-of-the-box to complex multi-modal text generation tasks such as video captioning.

**Broader Impact.** We have showed the superior compute-efficiency of our bidirectional framework compared to autoregressive models for zero-shot VideoQA, and believe it is a step towards reducing the environmental impact of such research and its applications [95]. In addition, our models might reflect biases present in videos and captions from Shutterstock used to train our model, the text data used to train the language model or the images and captions used to train the visual backbone. It is important to keep this in mind when deploying, analysing and building upon these models.

**Acknowledgements.** This work was granted access to the HPC resources of IDRIS under the allocation 2022-AD01101670R2 made by GENCI. The work was funded by a Google gift, the French government under management of Agence Nationale de la Recherche as part of the "Investissements d’avenir" program, reference ANR-19-P3IA-0001 (PRAIRIE 3IA Institute), the Louis Vuitton ENS Chair on Artificial Intelligence, the European Regional Development Fund under project IMPACT (reg. no. CZ.02.1.01/0.0/0.0/15 003/0000468). We thank anonymous reviewers for giving interesting feedback. We thank Gaspard Beugnot, Clémence Bouvier and Pierre-Louis Guhur for proofreading.
References


[87] Timo Schick and Hinrich Schütze. It’s not just size that matters: Small language models are also few-shot learners. In *NAACL*, 2021.


Appendix

In this Appendix, we present the following items:

(i) Additional qualitative examples of zero-shot VideoQA predictions (Section A)
(ii) A qualitative analysis of the frozen self-attention patterns in FrozenBiLM (Section B)
(iii) Additional information about our experimental setup (Section C), including datasets (Section C.1) and implementation details (Section C.2)
(iv) Additional experimental results (Section D), including a comparison to BLIP [59] in their zero-shot VideoQA settings (Section D.1), results on zero-shot image-VQA (Section D.2), detailed zero-shot VideoQA results segmented per question type (Section D.3), zero-shot results with different random seeds (Section D.4), additional ablation studies in few-shot settings (Section D.5), zero-shot settings (Sections D.6 and D.7) and fully-supervised settings (Section D.8)

A Qualitative examples for zero-shot VideoQA

To complement the qualitative examples shown in Figure 3, Figure 4 and the video video_examples.mp4 illustrate additional qualitative results of zero-shot VideoQA for our FrozenBiLM model and compares them to Just Ask [114], as well as to variants of our approach that do not freeze the language model (UnFrozenBiLM) and use no visual modality (text-only), as evaluated in Section 4.2. Consistently with the analysis done in Section 4.4, we observe that the unfrozen variant can predict answers that lack text-only commonsense reasoning, e.g. in the first example of Figure 4b, the word follow is grammatically incorrect; in the second example of Figure 4b, it is unlikely that a singer plays a toad. The text-only variant does have strong language understanding, but makes visually-unrelated predictions. In contrast, consistently with our quantitative results (see Tables 1, 2 and 5), our model FrozenBiLM is able to correctly answer various questions in the diverse VideoQA paradigms (open-ended VideoQA, video-conditioned fill-in-the-blank, multiple-choice VideoQA), showing both a strong textual commonsense reasoning and a complex multi-modal understanding.

Our zero-shot model still underperforms compared to VideoQA-supervised models (see Table 7) and we analyze its failure cases in Figure 4a. Qualitatively, we find that the zero-shot model can fail on examples requiring complex temporal or spatial understanding e.g. in the third example of the second row, the model does not detect a toaster behind the woman; in the second example of the second row, it gets confused as the person browses through many different tabs from their phone. It can also be semantically inaccurate, as in the first example of the second row, the model confuses a restaurant with a bakery; in the fourth example of the second row, it confuses a chicken with another kind of bird.

B Qualitative analysis of the frozen self-attention patterns in FrozenBiLM

We show in Section 4.2 that the visual modality is crucial for the zero-shot VideoQA performance. Here we further analyze qualitatively how, for zero-shot VideoQA, our model makes use of the visual modality through self-attention layers which are frozen after text-only pretraining. Figure 5 illustrates the self-attention patterns in FrozenBiLM for the second example in the first row of Figure 4. Despite the freezing, we observe that these layers actually enable visual-linguistic interactions. Indeed, in the first layer (Figure 4, left), the [CLS], [MASK] and [SEP] tokens significantly attend to the visual tokens. Moreover, we observe substantially different patterns in the last layer (Figure 4, right): while the [MASK] token still attends to visual tokens, the different visual tokens at different timesteps attend between each other and the [CLS] and [SEP] tokens mainly attend to other text tokens. Consistently with results presented in Section 4.2, this qualitative analysis suggests that the frozen self-attention layers in FrozenBiLM do enable visual-linguistic interactions.

C Experimental setup

In this section we first present additional information on the used datasets (Section C.1) and then describe implementation details (Section C.2).
(a) Zero-Shot open-ended VideoQA. The first row illustrates successful predictions on the iVQA dataset [113] (leftmost example) and the ActivityNet-QA dataset [120] (three rightmost examples). The second row illustrates incorrect predictions on the iVQA dataset.

(b) Zero-shot video-conditioned fill-in-the-blank successful predictions on the LSMDC-FiB dataset [73].

(c) Zero-shot multiple-choice VideoQA. The first and second rows illustrate successful predictions on the How2QA dataset [60] and the TVQA dataset [52], respectively.

Figure 4: Zero-Shot VideoQA. Qualitative comparison between Just Ask [114] (row 1 in Table 5), our model (row 4 in Table 5), its unfrozen variant (row 2 in Table 1) and its text-only variant (row 2 in Table 2), for zero-shot VideoQA. The last column of each row illustrates a single video example with two frames, while other columns illustrate each video example with one frame. We show more examples on our webpage [1].
Figure 5: *FrozenBiLM* self-attention visualization for zero-shot VideoQA. Visualization of the attention weights between the different visual tokens from the video prompt and the textual tokens from the text embedder, for the second example of the first row in Figure 4. A column corresponds to the weights of the different visual and text tokens for the given token. These attention weights are averaged across all 24 heads, and renormalized by the maximum weight for each token (i.e. each column) for the purpose of visualization. Lighter colors correspond to higher attention weights (see the colorbar on the right). In the first layers (left), we observe that the multi-modal interactions mainly flow through the [CLS], [MASK] and [SEP] tokens, and that there is little interaction between the different visual tokens. In the last layers (right), we observe that visual tokens attend to each other and the [MASK] token attends to the visual tokens, while the [CLS] and [SEP] tokens mainly attend to text tokens. Note that the self-attention weights are frozen after text-only pretraining.

C.1 Datasets

In this section, we give further details about the downstream datasets we use. Their licenses are mentioned in our code in the separate folder `code`.

**LSMDC-FiB** [73] is an open-ended video-conditioned fill-in-the-blank task which consists in predicting masked words in sentences that describe short movie clips [81, 82]. It contains 119K video clips and 349K sentences, split into 297K/22K/30K for training/validation/testing.

**iVQA** [113] is a recently introduced open-ended VideoQA dataset, focused on objects, scenes and people in instructional videos [74]. It excludes non-visual questions, and contains 5 possible correct answers for each question for a detailed evaluation. It contains 10K video clips and 10K questions, split into 6K/2K/2K for training/validation/testing.

**MSRVTT-QA** [109], **MSVD-QA** [109] and **TGIF-FrameQA** [35] are popular open-ended VideoQA benchmarks automatically generated from video descriptions [12, 66, 111]. Questions are of five types for MSRVTT-QA and MSVD-QA: what, who, how, when and where; and four types for TGIF-QA: object, number, color and location. MSRVTT-QA contains 10K video clips and 243K question-answer pairs, split into 158K/12K/73K for training/validation/testing. MSVD-QA contains
1.8K video clips and 51K question-answer pairs, split into 32K/6K/13K for training/validation/testing. TGIF-QA contains 46K GIFs and 53K question-answer pairs, split into 39K/13K for training/testing.

**ActivityNet-QA** [120] is an open-ended VideoQA dataset consisting of long videos [9] (3 minutes long on average), and covering 9 question types (motion, spatial, temporal, yes-no, color, object, location, number and other). It contains 5.8K videos and 58K question-answer pairs, split into 32K/18K/8K for training/validation/testing.

**How2QA** [60] is a multiple-choice VideoQA dataset focused on instructional videos [74]. Each question is associated with one correct and three incorrect answers. It contains 28K video clips and 38K questions, split into 35K/3K for training/validation.

**TVQA** [52] is a multiple-choice VideoQA dataset focused on popular TV shows. Each question is associated with one correct and four incorrect answers. It contains 22K video clips and 153K questions, split into 122K/15K/15K for training/validation/testing. The test set is hidden and only accessible a limited number of times via an online leaderboard.

### C.2 Implementation details

**Architecture hyperparameters.** We truncate text sequences up to \( L = 256 \) tokens. Video features are extracted by sampling \( T = 10 \) frames, each resized at \( 224 \times 224 \) pixels, from the video. These frames are sampled at temporally equal distance, with a minimum distance of 1 second. For videos shorter than \( T \) seconds, we pad the video prompt up to \( T \) tokens. The dimension of the visual features from ViT-L/14 [21] is \( D_f = 768 \). The transformer encoder from DeBERTa-V2-XLarge [28] has 24 layers, 24 attention heads, a hidden dimension of \( D = 1536 \) and an intermediate dimension in the feed-forward layers of 6144. For the adapters [31], we use a bottleneck dimension of \( D_h = D^8 = 192 \).

**Training.** For all training experiments, we use the Adam optimizer [45] with \( \beta = (0.9, 0.95) \) and no weight decay. We use Dropout [94] with probability 0.1 in the adapters and in the transformer encoder. When finetuning the language model weights, we divide the batch size by a factor 2 so to accommodate with the GPU memory constraints.

**Cross-modal training.** To train on WebVid10M, we use a total batch size of 128 video-caption pairs split in 8 NVIDIA Tesla V100 GPUs. We use a fixed learning rate of \( 3e^{-5} \) for the variant with adapters. We find that the variant without adapters that freezes the language model weights prefers a higher learning rate of \( 3e^{-4} \), and that the variant *UnfrozenBiLM* that finetunes the language model weights prefers a lower one of \( 1e^{-5} \).

**Downstream task finetuning.** To finetune our model on downstream datasets, we use a total batch size of 32 video-question-answer triplets (respectively 32 video-sentence pairs) split in 4 NVIDIA Tesla V100 GPUs for open-ended VideoQA datasets (respectively video-conditioned fill-in-the-blank datasets) and 16 video-question pairs split in 8 NVIDIA Tesla V100 GPUs for multiple-choice VideoQA datasets. We train for 20 epochs for all downstream datasets except LSMDC-FiB for which we find that training for 5 epochs leads to similar validation results. We warm up the learning rate linearly for the first 10% of iterations, followed by a linear decay of the learning rate (down to 0) for the remaining 90%. On each dataset, we run a random search and select the learning rate based on the best validation results. We search over 10 learning rates in the range \([1e^{-5}, 1e^{-4}]\) for variants that freeze the language model weights, and \([5e^{-6}, 5e^{-5}]\) for the variant *UnfrozenBiLM* that finetunes the language model weights.

**Answer vocabulary for open-ended VideoQA.** In the zero-shot setting, we use an answer vocabulary composed of the top 1,000 answers in the corresponding training dataset, following [121]. In the fully-supervised setting, we experiment both with the vocabulary composed of the top 1,000 answers and the vocabulary composed of all answers appearing at least twice in the corresponding training dataset and choose the one leading to best validation results. Following [121], questions with out-of-vocabulary answer are not used for finetuning, and are automatically considered as incorrect during evaluation.

### D Experiments

In this section, we complement the experiments presented in Section 4. We first present a comparison with BLIP [59] in their zero-shot settings in Section D.1. In Section D.3 we show detailed zero-shot
Table 8: Results of our model after cross-modal training, finetuning on the open-ended image-VQA dataset [4] and directly evaluating on open-ended VideoQA without using any VideoQA supervision, as in BLIP [59].

<table>
<thead>
<tr>
<th>Method</th>
<th>Pretraining Data</th>
<th>Finetuning Data</th>
<th>VQA MSRVTT-QA MSVD-QA ActivityNet-QA TGF-QA</th>
</tr>
</thead>
<tbody>
<tr>
<td>BLIP [59]</td>
<td>129M image-text pairs</td>
<td>VQA</td>
<td>19.2</td>
</tr>
<tr>
<td>FrozenBiLM (no image-VQA training) WebVid10M</td>
<td>0</td>
<td>VQA</td>
<td>14.6</td>
</tr>
<tr>
<td>FrozenBiLM (no cross-modal training)</td>
<td>0</td>
<td>VQA</td>
<td>34.6</td>
</tr>
</tbody>
</table>

FrozenBiLM (Ours) WebVid10M VQA

Table 9: Zero-shot VideoQA results segmented per question type on the ActivityNet-QA dataset, compared with Just Ask [113].

<table>
<thead>
<tr>
<th>Method</th>
<th>Motion</th>
<th>Spatial</th>
<th>Temporal</th>
<th>Yes-No</th>
<th>Color</th>
<th>Object</th>
<th>Location</th>
<th>Number</th>
<th>Other</th>
</tr>
</thead>
<tbody>
<tr>
<td>Just Ask [113]</td>
<td>2.3</td>
<td>1.1</td>
<td>0.3</td>
<td>36.3</td>
<td>11.3</td>
<td>4.1</td>
<td>6.5</td>
<td>0.2</td>
<td>4.7</td>
</tr>
<tr>
<td>FrozenBiLM</td>
<td>12.7</td>
<td>6.8</td>
<td>1.6</td>
<td>53.2</td>
<td>16.5</td>
<td>17.9</td>
<td>18.1</td>
<td>26.2</td>
<td>25.8</td>
</tr>
</tbody>
</table>

Table 10: Zero-shot VideoQA results segmented per question type on the MSRVTT-QA dataset (left) and the MSVD-QA dataset (right), compared with Just Ask [113].

<table>
<thead>
<tr>
<th>Method</th>
<th>MSRVTT-QA</th>
<th>MSVD-QA</th>
</tr>
</thead>
<tbody>
<tr>
<td>What</td>
<td>1.8</td>
<td>28.7</td>
</tr>
<tr>
<td>Who</td>
<td>0.7</td>
<td>55.0</td>
</tr>
<tr>
<td>Number</td>
<td>66.3</td>
<td>11.4</td>
</tr>
<tr>
<td>Color</td>
<td>0.6</td>
<td>9.2</td>
</tr>
<tr>
<td>When</td>
<td>0.6</td>
<td>9.3</td>
</tr>
<tr>
<td>Where</td>
<td>4.5</td>
<td>26.0</td>
</tr>
<tr>
<td>What</td>
<td>7.8</td>
<td>45.0</td>
</tr>
<tr>
<td>Who</td>
<td>1.7</td>
<td>69.9</td>
</tr>
<tr>
<td>Number</td>
<td>74.3</td>
<td>56.3</td>
</tr>
<tr>
<td>Color</td>
<td>18.3</td>
<td>5.2</td>
</tr>
<tr>
<td>When</td>
<td>3.5</td>
<td>17.9</td>
</tr>
<tr>
<td>Where</td>
<td>0.0</td>
<td></td>
</tr>
</tbody>
</table>

VideoQA results segmented per question category and compare our method with Just Ask [113]. Next we analyze the impact of the random seed used in the cross-modal training on the zero-shot VideoQA results in Section D.4. We also show the importance of freezing the language model in few-shot settings in Section D.5. Finally we show the benefit of cross-modal training and adapter training in fully-supervised settings in Section D.8.

D.1 Comparison with BLIP

In addition to the zero-shot results presented in Section 4.4, we here investigate a different but related zero-shot setting defined in BLIP [59], where a network trained on manually annotated image-VQA annotations is evaluated directly on open-ended VideoQA datasets. In detail, BLIP uses the open-ended image-VQA dataset [4] for finetuning after pretraining on 129M image-text pairs, including COCO [14] and Visual Genome [46] which are manually annotated. To adapt our model to this setting, we finetune our model FrozenBiLM pretrained on WebVid10M on the image-VQA dataset using the same procedure as for finetuning on VideoQA datasets (see Section 3.3), i.e. notably with a frozen language model. In particular, we finetune on VQA for 10 epochs with an initial learning rate of $1 \times 10^{-5}$ which is warmed up for the first 10% iterations, and linearly decayed to 0 for the remaining 90% iterations. Table 8 shows that the resulting model not only improves over our model without image-VQA finetuning (i.e. in zero-shot mode as defined in Section 1) or our model trained on VQA only (i.e. without cross-modal training), but also substantially outperforms BLIP on both MSRVTT-QA and MSVD-QA. These results further demonstrate the strong capabilities of FrozenBiLM in settings where no VideoQA annotation is available.

D.2 Results on zero-shot image-VQA

We next evaluate our pretrained model on the VQAv2 [4] validation set in the zero-shot setting, i.e., without any supervision of visual questions and answers. Frozen [102] achieves 29.5% accuracy in this setting using an autoregressive language model. In comparison, our FrozenBiLM model is 7 times smaller than Frozen and achieves 45.0% accuracy. We conclude that our model can perform competitively on the image-VQA tasks despite being tailored for videos.

D.3 Detailed zero-shot VideoQA results segmented per question category

We complement the comparison to the state of the art for zero-shot VideoQA given in Section 4.4 with results segmented per question type for ActivityNet-QA in Table 9, and for MSRVTT-QA and
Table 11: Comparison with the state of the art for zero-shot VideoQA, reporting mean and standard deviation over 5 cross-modal training runs with different random seeds. Results on TVQA are reported on the validation set given that the hidden test set can only be accessed a limited number of times.

<table>
<thead>
<tr>
<th>Variant</th>
<th>Supervision</th>
<th>Fill-in-the-blank</th>
<th>Open-ended</th>
<th>Multiple-choice</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>LSMDC</td>
<td>iVQA</td>
<td>MSRVTT-QA</td>
</tr>
<tr>
<td>1. UnFrozenBiLM</td>
<td>0% (zero-shot)</td>
<td>37.1</td>
<td>27.0</td>
<td>17.6</td>
</tr>
<tr>
<td>2. FrozenBiLM</td>
<td>0% (zero-shot)</td>
<td>51.5</td>
<td>26.8</td>
<td>16.7</td>
</tr>
<tr>
<td>3. UnFrozenBiLM</td>
<td>1% (few-shot)</td>
<td>46.2</td>
<td>23.5</td>
<td>33.4</td>
</tr>
<tr>
<td>4. FrozenBiLM</td>
<td>1% (few-shot)</td>
<td>56.9</td>
<td>31.1</td>
<td>36.0</td>
</tr>
<tr>
<td>5. UnFrozenBiLM</td>
<td>10% (few-shot)</td>
<td>52.6</td>
<td>29.5</td>
<td>38.9</td>
</tr>
<tr>
<td>6. FrozenBiLM</td>
<td>10% (few-shot)</td>
<td>59.9</td>
<td>35.3</td>
<td>41.7</td>
</tr>
<tr>
<td>7. UnFrozenBiLM</td>
<td>100% (fully-supervised)</td>
<td>58.9</td>
<td>37.7</td>
<td>45.0</td>
</tr>
<tr>
<td>8. FrozenBiLM</td>
<td>100% (fully-supervised)</td>
<td>63.5</td>
<td>39.6</td>
<td>47.0</td>
</tr>
</tbody>
</table>

Table 12: Few-shot results, by finetuning FrozenBiLM using a small fraction of the downstream training dataset, compared with the variant UnFrozenBiLM which does not freeze the language model weights. Results on TVQA are reported on the validation set given that the hidden test set can only be accessed a limited number of times.

### D.4 Impact of the random seed on zero-shot VideoQA

To verify the robustness of our approach with respect to the random seed, we run cross-modal training for FrozenBiLM with 5 different random seeds. We report the mean and standard deviation of zero-shot accuracy in Table 11, compared with state-of-the-art approaches that only report their results based on a single run. We observe that the random seed does not affect the comparison to prior work done in Section 4.4 in the main paper, as our model improves over previous work for zero-shot VideoQA [78, 114, 122] by significant margins.

### D.5 Freezing the language model is also beneficial in few-shot settings

Sections 4.2 and 4.5 demonstrate that freezing the language model combined with training adapters outperforms finetuning the language model in the zero-shot and fully-supervised settings. In Table 12, we further show that freezing the language model combined with training adapters outperforms finetuning the language model in the few-shot setting as defined in Section 4.5 (compare rows 3 and 4, or rows 5 and 6). Interestingly, the difference is larger when using 1% of the downstream training dataset (rows 3 and 4) compared to using 10% (rows 5 and 6) or 100% (rows 7 and 8). These results demonstrate that our approach is particularly efficient in settings where VideoQA annotations are scarce.

### D.6 Ablation of the multi-token inference strategy

For multi-token answers in the open ended VideoQA setting, our FrozenBiLM simply averages the weights of different answer tokens. However, such simple scheme does not preserve the semantic structure of the answer. Hence we here investigate and compare another possible inference strategy in the zero-shot setting and discuss potential sources of improvement. We take inspiration from [38] and performs zero-shot VideoQA inference by using multiple mask tokens decoded in parallel. Then, for each video-question pair, we do one forward pass through the model per possible number of mask tokens (typically, 1 to 5) in order to score all possible answers in vocabulary $A$. The score of a given answer is then obtained by multiplying the probability of its individual tokens,
Table 13: Impact of the inference strategy on the zero-shot open-ended VideoQA performance.

<table>
<thead>
<tr>
<th>Inference Strategy</th>
<th>Fill-in-the-blank</th>
<th>iVQA</th>
<th>MSRVTT-QA</th>
<th>MSVD-QA</th>
<th>ActivityNet-QA</th>
<th>TGIF-QA</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Average token embeddings</td>
<td>51.5</td>
<td>26.8</td>
<td>16.7</td>
<td>33.8</td>
<td>25.9</td>
<td>41.9</td>
</tr>
<tr>
<td>2. Multiple mask tokens</td>
<td>51.0</td>
<td>27.0</td>
<td>17.1</td>
<td>34.4</td>
<td>26.1</td>
<td>42.0</td>
</tr>
</tbody>
</table>

Table 14: Impact of the number of frames \( T \) used by the model, the hidden dimension \( D_h \) in the adapters and the visual backbone on the zero-shot VideoQA results. All models are trained on WebVid10M and use multi-modal inputs (video, speech and question) at inference.

possibly normalized by its number of tokens. As shown in Table 13, we observe that such a decoding strategy (row 2) does not significantly improve the accuracy of our model over the one used in FrozenBiLM (row 1). We hypothesize that this is due to the fact that the current open-ended VideoQA datasets [35, 109, 113, 120] contain a great majority of short answers, e.g. 99% of the answers in the MSRVTT-QA test set are one-token long with our tokenizer [47]. Additionally, a possible solution to further improve the decoding in this alternative scheme is to increase the length of the masked spans at pretraining, as in [39]. [84] provides another potential solution to score multi-token answers in our framework, by masking tokens one by one and computing pseudo-likelihood scores.

D.7 Additional ablation studies in the zero-shot setting

We here complement zero-shot ablation studies reported in Section 4.2. We analyze the impact of the number of frames \( T \) used by the model, the hidden dimension in the adapters \( D_h \) and the size and pretraining of the visual backbone in Table 14. All models use the same setting as described in Section 4.2 and detailed in Section C. We first observe that using 10 frames significantly improves over using a single frame (compare rows 1 and 5). Next we note that using a hidden dimension of 96 or 384 in the adapters instead of 192 does not change the results significantly (see rows 2, 3 and 6). Moreover, we find that scaling up the size of the visual backbone is beneficial, as using ViT-L/14 instead of ViT-B/16, both being trained on CLIP [78], slightly improves the results (compare rows 4 and 6). Furthermore, we observe that the pretraining of the visual backbone is crucial, as using ViT-B/16 pretrained on 400M image-text pairs from CLIP significantly improves over using ViT-B/16 pretrained on ImageNet-21K, i.e. 22M image-label pairs (compare rows 4 and 5).

Finally, we ablate the importance of the prompt design on the zero-shot VideoQA performance. We report results with alternative prompts in Tables 15 and 16. We find that replacing the words “Question”, “Answer” and “Subtitles” by “Q”, “A” and “S”, respectively, in the templates described in Section 3.3 does not impact the zero-shot VideoQA accuracy (compare rows 2 and 1 in Tables 15 and 16). However, completely removing “Question”, “Answer”, “Subtitles” and “is it” in the templates results in a significant drop of performance (compare rows 3 and 1 in Tables 15 and 16). We conclude that it is important to have tokens that link the different textual inputs.

D.8 Cross-modal training and adapters are crucial for fully-supervised performance

We have examined the impact of cross-modal training and training various parameters of our architecture on the zero-shot VideoQA performance in Section 4.2. In Table 17, we complement these ablation studies by analyzing the importance of cross-modal training and training various parameters for the fully-supervised VideoQA performance. For this, we train on downstream datasets a variant with no adapters, and a variant without cross-modal training, following the same procedure as explained in Section 3.3 and detailed in Section C. We find that cross-modal training is significantly beneficial for the fully-supervised setting (compare rows 3 and 4). Similar to conclusions made in Section 4.5, training adapters while freezing the language model outperforms finetuning the language.
Table 15: Impact of the prompt on the zero-shot open-ended VideoQA performance.

Table 16: Impact of the prompt on the zero-shot multiple-choice VideoQA performance.

Table 17: Importance of cross-modal training and training various parameters for fully-supervised VideoQA. All models are finetuned on downstream VideoQA datasets, and use multi-modal inputs (video, speech and question) at inference.

model in fully-supervised settings (see rows 1 and 4). Finally, we note that training adapters has a considerable importance on the performance in fully-supervised settings (compare rows 2 and 4). These results further demonstrate the strength of our approach in the fully-supervised setup.