Learning to Answer Visual Questions from Web Videos
Antoine Yang, Antoine Miech, Josef Sivic, Ivan Laptev, Cordelia Schmid

To cite this version:
Antoine Yang, Antoine Miech, Josef Sivic, Ivan Laptev, Cordelia Schmid. Learning to Answer Visual Questions from Web Videos. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2022, 10.1109/tpami.2022.3173208. hal-03664182

HAL Id: hal-03664182
https://inria.hal.science/hal-03664182v1
Submitted on 10 May 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.
Learning to Answer Visual Questions from Web Videos

Antoine Yang, Antoine Miech, Josef Sivic, Ivan Laptev, Cordelia Schmid

Abstract—Recent methods for visual question answering rely on large-scale annotated datasets. Manual annotation of questions and answers for videos, however, is tedious, expensive and prevents scalability. In this work, we propose to avoid manual annotation and generate a large-scale training dataset for video question answering making use of automatic cross-modal supervision. We leverage a question generation transformer trained on text data and use it to generate question-answer pairs from transcribed video narrations. Given narrated videos, we then automatically generate the HowToVQA69M dataset with 69M video-question-answer triplets. To handle the open vocabulary of diverse answers in this dataset, we propose a training procedure based on a contrastive loss between a video-question multi-modal transformer and an answer transformer. We introduce the zero-shot VideoQA task and the VideoQA feature probe evaluation setting and show excellent results, in particular for rare answers. Furthermore, our method achieves competitive results on MSRVTT-QA, ActivityNet-QA, MSVD-QA and How2QA datasets. We also show that our VideoQA dataset generation approach generalizes to another source of video web and text data. We use our method to generate the WebVidVQA3M dataset from the WebVid dataset, i.e., videos with alt-text annotations, and show its benefits for training VideoQA models. Finally, for a detailed evaluation we introduce iVQA, a new VideoQA dataset with reduced language bias and high-quality manual annotations. Code, datasets and trained models are available on our project webpage1.

Index Terms—Video Question Answering, Cross-Modal Supervision, Question Generation, Zero-Shot Learning

1 INTRODUCTION

Answering questions about videos requires a detailed understanding of the visual content and its association with the natural language. Indeed, given the large diversity of questions, methods for Video Question Answering (VideoQA) should reason about scenes, objects and human actions as well as their complex temporal interactions.

Current approaches to VideoQA rely on deep fully-supervised models trained on manually annotated datasets with question and answer pairs [28, 38, 42, 43, 50, 52, 58]. Collecting and annotating VideoQA datasets, however, is cumbersome, time consuming, expensive and therefore not scalable. As a result, current VideoQA datasets are relatively small (see Figure 2). This limitation hinders the progress in the field as state-of-the-art VideoQA models often require a large amount of training data.

In this work, we address the scale issue with a new approach for automatically generating VideoQA datasets as illustrated in Figure 1. The idea is to leverage cross-modal supervision together with text-only tools for question generation and to automatically annotate VideoQA data from a large amount of videos with readily-available text annotations in the form of transcribed narrations or “alt-text” annotations available with the video on the Internet. Inspired by the recent progress in language generation using transformer-based language models [12], we leverage transformers trained on a question-answering text corpus to generate a diverse set of non-scripted questions and corresponding open-vocabulary answers from text. By applying these transformers to speech transcripts of narrated videos from the large-scale HowTo100M dataset [69] we create HowToVQA69M, an open-ended VideoQA dataset with 69 million video-question-answer triplets and a diverse set of more than 16M unique answers (see Figure 3). We also extend our approach to web videos with readily-available alt-text descriptions and generate the WebVidVQA3M dataset from the WebVid2M dataset [9]. As shown in Figure 2 our HowToVQA69M and WebVidVQA3M datasets are orders of magnitude larger compared to prior VideoQA datasets.

Fig. 1: Given videos with transcribed narration (left) or videos with “alt-text” annotations (right), we leverage language models and cross-modal supervision to obtain large-scale VideoQA data. Top: Example frame with the corresponding text annotation. Bottom: automatically generated question and answer pair.

1. https://antoyang.github.io/just-ask.html

— Recent methods for visual question answering rely on large-scale annotated datasets. Manual annotation of questions and answers for videos, however, is tedious, expensive and prevents scalability. In this work, we propose to avoid manual annotation and generate a large-scale training dataset for video question answering making use of automatic cross-modal supervision. We leverage a question generation transformer trained on text data and use it to generate question-answer pairs from transcribed video narrations. Given narrated videos, we then automatically generate the HowToVQA69M dataset with 69M video-question-answer triplets. To handle the open vocabulary of diverse answers in this dataset, we propose a training procedure based on a contrastive loss between a video-question multi-modal transformer and an answer transformer. We introduce the zero-shot VideoQA task and the VideoQA feature probe evaluation setting and show excellent results, in particular for rare answers. Furthermore, our method achieves competitive results on MSRVTT-QA, ActivityNet-QA, MSVD-QA and How2QA datasets. We also show that our VideoQA dataset generation approach generalizes to another source of video web and text data. We use our method to generate the WebVidVQA3M dataset from the WebVid dataset, i.e., videos with alt-text annotations, and show its benefits for training VideoQA models. Finally, for a detailed evaluation we introduce iVQA, a new VideoQA dataset with reduced language bias and high-quality manual annotations. Code, datasets and trained models are available on our project webpage.

Index Terms—Video Question Answering, Cross-Modal Supervision, Question Generation, Zero-Shot Learning

1 INTRODUCTION

Answering questions about videos requires a detailed understanding of the visual content and its association with the natural language. Indeed, given the large diversity of questions, methods for Video Question Answering (VideoQA) should reason about scenes, objects and human actions as well as their complex temporal interactions.

Current approaches to VideoQA rely on deep fully-supervised models trained on manually annotated datasets with question and answer pairs [28, 38, 42, 43, 50, 52, 58]. Collecting and annotating VideoQA datasets, however, is cumbersome, time consuming, expensive and therefore not scalable. As a result, current VideoQA datasets are relatively small (see Figure 2). This limitation hinders the progress in the field as state-of-the-art VideoQA models often require a large amount of training data.

In this work, we address the scale issue with a new approach for automatically generating VideoQA datasets as illustrated in Figure 1. The idea is to leverage cross-modal supervision together with text-only tools for question generation and to automatically annotate VideoQA data from a large amount of videos with readily-available text annotations in the form of transcribed narrations or “alt-text” annotations available with the video on the Internet. Inspired by the recent progress in language generation using transformer-based language models [12], we leverage transformers trained on a question-answering text corpus to generate a diverse set of non-scripted questions and corresponding open-vocabulary answers from text. By applying these transformers to speech transcripts of narrated videos from the large-scale HowTo100M dataset [69] we create HowToVQA69M, an open-ended VideoQA dataset with 69 million video-question-answer triplets and a diverse set of more than 16M unique answers (see Figure 3). We also extend our approach to web videos with readily-available alt-text descriptions and generate the WebVidVQA3M dataset from the WebVid2M dataset [9]. As shown in Figure 2 our HowToVQA69M and WebVidVQA3M datasets are orders of magnitude larger compared to prior VideoQA datasets.

Fig. 1: Given videos with transcribed narration (left) or videos with “alt-text” annotations (right), we leverage language models and cross-modal supervision to obtain large-scale VideoQA data. Top: Example frame with the corresponding text annotation. Bottom: automatically generated question and answer pair.

1. https://antoyang.github.io/just-ask.html
answers in HowToVQA69M. To address this problem and to enable video question answering with highly diverse questions and answers, we introduce a training procedure based on contrastive learning between a video-question multi-modal transformer and an answer transformer that can handle free-form answers. This bypasses the need to define a discrete set of answer classes.

The goal of our work is to advance truly open-ended and generic solutions to VideoQA. To evaluate generalization, we propose a new zero-shot VideoQA task where we prohibit any manual supervision of visual data during training, and a new VideoQA feature probe evaluation setting where only the final projection layers of the network are finetuned on the target dataset. Our VideoQA model, trained on HowToVQA69M, demonstrates excellent zero-shot results on multiple existing datasets, especially for rare answers. Additionally, we find that our VideoQA model exhibits strong performance in the VideoQA feature probe evaluation setting. Moreover, when finetuned on target datasets, our model achieves competitive results on MSRVTT-QA [30], ActivityNet-QA [111], MSVD-QA [100] and How2QA [56]. We further show the generalizability of our approach by showing the benefits of WebVidVQA3M for training VideoQA models.

Initial experiments have shown that existing benchmarks for open-ended VideoQA [100], [110] contain a language bias [34], i.e., their questions can often be answered without looking at the video. To better evaluate the impact of visual information in VideoQA, we introduce a new open-ended VideoQA dataset (iVQA) with manually collected questions and answers, where we exclude questions that could be answered without watching the video. Moreover, to account for multiple possible answers, iVQA contains five independently collected answers for each question.

In summary, our work makes the following three contributions:

(i) We introduce an approach to automatically generate a large-scale VideoQA dataset, HowToVQA69M. Relying on cross-modal supervision, we use transformers trained on an existing text-only question-answering corpus and generate video-question-answer triplet pairs from videos and transcribed narrations. We also apply our method to video alt-text pairs and generate the WebVidVQA3M dataset.

(ii) We train a VideoQA model on the automatically generated data via contrastive learning between a multi-modal video-question transformer and an answer transformer. We show the efficiency of our model for the new zero-shot VideoQA task and the new VideoQA feature probe task. Our model achieves competitive results in four existing VideoQA benchmarks.

(iii) Finally, we introduce a new manually annotated open-ended VideoQA benchmark iVQA that excludes non-visual questions and contains multiple possible answers for each question.

Code, datasets and trained models are available at https://antoyang.github.io/just-ask.html

2 RELATED WORK

Visual Question Answering (VQA). VQA is typically tackled by classifying the image-question (or video-question) representation into a fixed vocabulary of answers. Various approaches to combine spatial image representations and sequential question representations have been proposed [6], [11], [50], [66], [99], [101], [106]. More specifically to the video domain (VideoQA), spatio-temporal video representations in terms of motion and appearance have been used in [23], [28], [32], [38], [41], [42], [43], [50], [51], [52], [58], [72], [79], [100], [102], [109], [114], [122].

Methods above are limited to pre-defined vocabularies of answers and are difficult to apply outside of specific datasets. To address this problem, Hu et al. [37] propose a joint embedding where image-question representations can be matched with free-form answers. Our VideoQA model follows this idea, but instead of relying on manually annotated datasets of limited scale, we train it on a large-scale VideoQA dataset that we automatically generate. In contrast to some previous works using additional video features such as subtitles [10], [43], [44], [48], [53], [54], [56], [62], [69], [95], [105], our video representation is exclusively based on visual information, as we focus on the detailed visual understanding of videos.

To evaluate the generalization of VQA models, Teney and Hengel [90] define zero-shot VQA by answering previously unseen questions, which is a related but less challenging task compared to the zero-shot VQA task we propose in Section 6.2. Vatashsky and Ullman [93] address VQA using COCO image annotations [61], while our zero-shot model is trained with no manual annotations. Our proposed zero-shot VQA task is analogous to zero-shot video retrieval [68] or zero-shot action recognition [73]. We further propose a VQA feature probe evaluation setting where only the final heads of the network are finetuned on the downstream dataset while all other pretrained weights are kept frozen. This setting is analogous to the linear probe evaluation setting commonly used in self-supervised image recognition [13], [14], [19] or self-supervised action recognition [73] but with multiple layers in the head rather than just a single (linear) layer.

Visual question generation (VQG) has been introduced in [70]. The methods in [60] and [81] propose to jointly learn VQG and VQA to improve the image VQA task. However, these works do not generate questions to obtain additional training data, but use visual data annotation for VQG as an additional loss.

VideoQA datasets. Manually collecting and annotating video-question-answer triplets is cumbersome, costly and difficult to scale. As a result, current VideoQA datasets [15], [21], [22], [27], [33], [41], [47], [53], [56], [71], [72], [83], [89], [97], [100], [108], [110], [111], [113] are limited in size, as the largest, TGIFF-QA [41], contains only 72K annotated clips (see Figure 2 for more details). To address this issue, several works have explored leveraging manually annotated video descriptions [41], [54], [100], [113], [115], [116], [117] for automatic generation of VideoQA datasets, using rule-based [35], [76] approaches. Similarly, in the image domain, Banerjee et al. [10] has recently proposed to use annotated image captions from COCO [18] to generate question-
answer pairs using a template-based approach [76]. Instead, we propose to use video annotations in the form of transcribed narrations or alt-text descriptions that are available at large-scale with no manual supervision. Moreover, rule-based generation requires the manual creation of rules by experts which is expensive, and has also been recently outperformed by neural question generation [26], [107], [119] as used in our approach.

**Large-scale pretraining for vision and language.** Several recent methods [4], [20], [24], [29], [40], [55], [57], [59], [64], [65], [85], [88], [104], [118] pretrain multi-modal vision-language representations, such as transformers, using datasets with image captions, e.g., COCO [18], Conceptual Captions [82] and Visual Genome [49]. These methods are often optimized using generic objectives such as masked language losses and losses for text-image matching and image caption generation. In our work, we pretrain models using large amounts of narrated videos. In contrast to task-agnostic pretraining in the previous work, we show the benefits of our approach.

A preliminary version of this article has appeared in [103].

3 LARGE-SCALE GENERATION OF VIDEOQA DATA

This section presents our approach to generate large-scale VideoQA datasets from videos with readily available text annotations. We illustrate the proposed approach on instructional videos with text annotations in the form of transcribed narrations, which in many cases describe the content of the videos. Section 3.1 presents our proposed generation procedures. Section 3.2, then, describes the resulting HowToVQA69M dataset. Our approach can be easily adapted to other type of content, for example, shorter web-videos with with readily text annotations in the form of alt-text, as will be shown in the result section (Section 6).

### 3.1 Generating video-question-answer triplets

We tackle the task of generating video-question-answer triplets from a large-scale instructional video dataset with transcribed spoken narration [69]. This is a challenging task because of transcription errors and lack of punctuation. We also wish to obtain highly diverse data. To address these issues, we propose to leverage powerful language models trained on text data. Our approach is illustrated in Figure 3 and details are given next.

We first present details about the generation procedure. Let $s$ be the transcribed speech data obtained with automatic speech recognition (ASR). First, we use a recurrent neural network $p$ to infer punctuation in the transcribed speech data. We denote the punctuated transcript as $p(s)$. We extract video clips $v$ temporally aligned with the inferred sentences $p(s)$ using the ASR timestamps. We found that the generation works significantly better when applied to sentences rather than the original sentence fragments from the HowTo100M dataset, see Table 1. Second, for each sentence, we apply a transformer $T_a$, to extract a set of potential answers: $a = T_a(p(s))$. Third, we use another transformer $T_q$ to generate a question given each transcript sentence and each extracted answer such that: $q = T_q(a, p(s))$. The output is a set of video-question-answer triplets $(v, q, a)$. 

![Fig. 3: Our automatic approach for large-scale generation of video-question-answer triplets from narrated (subtitled) videos. First, at the language-only training phase (left), the transformer-based answer extractor $T_a$ and question generator $T_q$ are trained on a manually annotated text-only question-answer corpus. Then video-question-answer triplets are automatically generated from narrated videos (right). Individual sentences are extracted from the ASR-transcribed narration using a punctuator $p$. Each extracted sentence is analyzed with an answer extractor $T_a$ and a question generator $T_q$ to produce answer $a$ and question $q$. The timestamps of the narration are used to obtain a video clip $v$ temporarily aligned to the extracted sentence to form the output video-question-answer triplet $(v, q, a)$]
We now explain details of the language models and their training procedure. For ASR, we follow [69] and use the readily-available ASR data provided by YouTube. For punctuation $p$, we use the BRNN model from [21] and the weights available at [1] trained on IWSLT2011 [29]. For $T_a$ and $T_q$, we use the transformer-based T5-small and T5-base models [74], respectively. We follow [3], [17], [63] and use the weights available at [2] trained for answer span extraction and answer-aware question generation, respectively, on SQuADv1 [75]. SQuADv1 is a text-only question-answering dataset consisting of questions for which the answer is a segment of text extracted from a paragraph.

Fig. 4: Examples of video-question-answer triplets generated from narrated videos in our HowToVQA69M dataset. The green color (first row) indicates relevant examples, the orange color (second row) indicates a failure of the question-answer generation, and the red color (third row) indicates that the generated question-answer is unrelated to the visual content.

Fig. 5: Statistics of the HowToVQA69M dataset. (a) Distribution of length of questions and answers. (b) Distribution of video clip duration in seconds.

Fig. 6: Word clouds extracted from the HowToVQA69M dataset showing its diverse vocabulary and the words characteristic to speech such as okay, right, or oh.

3.2 HowToVQA69M: a large-scale VideoQA dataset
We have applied the previously described procedure to all 1.2M original videos from the HowTo100M dataset [69]. The result is HowToVQA69M, a dataset of 69,270,581 video clip, question and answer triplets $(v, q, a)$. HowToVQA69M is two orders of magnitude larger than any of the currently available VideoQA datasets (see Figure 2). On average, each original video results in 43 video clips, where each clip is associated to 1.2 question-answer
Manual evaluation of our generation method (with and without punctuation) on a random sample of 100 examples compared with a rule-based question-answer generation of [35]. Numbers are obtained with majority voting between 5 annotators.

| Question Type | Total | Correct Samples (%) | QA Generation Failure (%) | QA unrelated to video (%) |
|---------------|-------|---------------------|--------------------------|--------------------------|
| Attribute     | 25    | 28                  | 32                       | 40                       |
| Object        | 17    | 41                  | 24                       | 35                       |
| Action        | 16    | 69                  | 19                       | 13                       |
| Counting      | 13    | 23                  | 15                       | 62                       |
| Place         | 7     | 0                   | 86                       | 14                       |
| People        | 7     | 0                   | 43                       | 57                       |
| Other         | 15    | 13                  | 27                       | 60                       |

TABLE 2: Manual evaluation of our generation method on 100 randomly chosen generated examples split by question type. Results are obtained by majority voting among 5 annotators.

The HowToVQA69M dataset is highly diverse and contains over 16M unique answers, where over 2M unique answers appear more than once and over 300K unique answers appear more than ten times. Examples of (v, q, a) triplets from the HowToVQA69M dataset are illustrated in Figure 4, showing the diversity and the noise in the automatically obtained annotations in HowToVQA69M.

Statistical analysis of HowToVQA69M. Figure 5 shows the statistics of the HowToVQA69M dataset in terms of the question length, answer length and video clip duration. Questions and answers contain 8.7 and 2.4 words on average respectively. Overall, HowToVQA69M contains longer answers than downstream VideoQA datasets like MSRVTT-QA, MSVD-QA or ActivityNet-QA, for which answers are on average shorter than 2 words. Each clip lasts 12.1 seconds on average. The distribution of clip duration has a peak at around seven seconds with a long tail of longer clips. These statistics demonstrate the diversity of our HowToVQA69M dataset, in terms of videos, questions and answers.

Word clouds for questions and answers in HowToVQA69M are shown in Figure 6 and illustrate the diverse vocabulary in HowToVQA69M as well as the presence of speech-related words such as "okay", "right", "oh".

Manual evaluation of HowToVQA69M. As shown in Figure 7, HowToVQA69M annotations are noisy, which can be attributed to: (i) errors in speech transcription, (ii) speech not describing the video content, or (iii) errors in question-answer generation. We manually evaluate the quality of 100 randomly sampled (v, q, a) triplets in HowToVQA69M by collecting 5 different annotations for each triplet to reduce variance and report results in Table 1. Among 100 triplets generated by our method we find 30 to be correctly generated and matching well to the video content, 31 are incorrectly generated and 39 are correctly generated but unrelated to the video content. To demonstrate the influence of the different components of our automatic question-answer generation procedure, we compare our results with (i) a variant of our approach that does not split transcribed narrations into sentences using a punctuator, and (ii) a rule-based approach [35] for question-answer generation. Table 1 confirms the importance of punctuation and demonstrates the superior performance of our generation method compared to [35]. Further comparison with [35] is given in Section 6.6. In terms of inter-rater agreement, for the 300 generated video-question-answer triplets (100 for each generation method), 94 were in an agreement of all 5 annotators, 198 in an agreement of at least 4 annotators, and 299 in an agreement of at least 3 annotators. This high agreement of annotators demonstrates the reliability of the results in Table 1.

We further manually classify the 100 video-question-answer triplets obtained with our method by the question type (“Attribute”, “Object”, “Action”, “Counting”, “Place”, “People”, or “Other”), evaluate the quality of generated triplets for different question types and report results in Table 2. Out of the 6 most common categories, we observe that questions related to “Action” lead to the best annotations, “Counting” questions lead to the highest number of QAs unrelated to the video content, and questions related to “Place” lead to the highest number of QA generation errors. Qualitatively, we found that actions are often depicted in the video, while counted quantities (e.g. time, weight, length) mentioned in the speech are hard to guess from the video only. We describe next how we use HowToVQA69M to train our VideoQA model.

4 VIDEOQA MODEL AND TRAINING PROCEDURE

This section presents our VideoQA model (Section 4.1) and describes the training procedure (Section 4.2). Figure 7 gives an overview of the model.

4.1 VideoQA model

As illustrated in Figure 7, our VideoQA model is composed of two branches: (i) a video-question module \( f \) based on a transformer and a mapping from the CLS token with a linear function. It takes a pair of video \( v \) and question \( q \) as input, models the multi-modal temporal interactions between \( v \) and \( q \) and then outputs an embedding vector \( f(v, q) \in \mathbb{R}^d \). (ii) The second branch is a text encoder \( g \) that embeds an answer \( a \) as \( g(a) \in \mathbb{R}^d \).

We will denote our model as VQA-T, standing for VideoQA-Transformer. Note that using the joint (video, question) and answer embeddings allows us to deal with a large open vocabulary of answers present in our new HowToVQA69M dataset as the model can measure similarity between the input video-question embedding and the embedding of any answer. This is in contrast to using a classification answer module \([38], [42], [43], [50], [122]\) that can choose only from a fixed predefined vocabulary.
of answers. Our embedding can be also easily finetuned on the different downstream VideoQA datasets, which may contain new answers that have not been seen at training. In contrast, the classification answer module has to be retrained when the vocabulary of answers changes. Next, we give details of the language and video representations. Further details about the model are provided in Appendix A.

**Word representation.** The question and answer are separately tokenized with the WordPieces embedding [96] and fed to DistilBERT [78]. DistilBERT is a light version of BERT [25] pretrained in a self-supervised fashion on English Wikipedia and the Toronto Book Corpus [121].

**Video representation.** We use a frozen S3D [98] pretrained on HowTo100M [69] using MIL-NCE [68]. This model is pretrained from scratch on HowTo100M only.

### 4.2 Training procedure

This section describes the training of our VideoQA model on the HowToVQA69M dataset and its finetuning on downstream VideoQA datasets.

**Training on HowToVQA69M.** We wish to make a pair of video and question \((v, q)\) close to its correct answer \(a\) measured by the dot product of their embeddings, \(f(v, q) \cdot g(a)\). In contrast, the incorrect answers should be far, i.e., the dot product with their embeddings should be small. This can be done by maximizing the following contrastive objective:

\[
\max _{f,g} \sum _{i=1} ^n \log \left( \frac{e^{f(v_i,q_i) \cdot g(a_i)}}{e^{f(v_i,q_i) \cdot g(a_i)} + \sum _{(v',q',a') \sim N_i} e^{f(v',q') \cdot g(a')}} \right),
\]

where \((v_i,q_i,a_i)\) represents a generated triplet (video clip, question, answer) from HowToVQA69M. Given a specific positive triplet \((v_i,q_i,a_i)\), we construct the set \(N_i\) of negative triplets by concatenating incorrect answers \(a_j\) within the training batch to the video-question pair \((v_i,q_i)\) as: \((v_i,q_i,a_j)\) with \(a_j \neq a_i\). In particular, if the same negative answer \(a_j\) is present multiple times in a batch, we only count it once. We found that sampling the same negative answer multiple times leads to worse results (see Section 6.9), which we believe is due to different distributions of answers in the pretraining and downstream datasets. Removing duplicate negatives helps to mitigate this difference.

**Finetuning on downstream VideoQA datasets.** We leverage the model pretrained on HowToVQA69M and finetune it on a downstream VideoQA dataset that typically has a smaller vocabulary of answers \(V\) (e.g. \(|V| \sim 4000\)). To this end, we adapt the training objective in (1) by constructing the negative set \(N_i\) from all incorrect answers in \(V\). Note that in such setting (1) becomes equivalent to optimizing the standard cross-entropy objective. In the specific case of multiple-choice VideoQA, the set of negatives \(N_i\) is the set of incorrect answers for each sample.

**Masked Language Modeling (MLM).** In addition to the contrastive loss (1) we apply the masking loss [25] to question tokens during both pretraining and finetuning. We found this to have a positive regularization effect when finetuning the DistilBERT weights (see Section 6.9).

### 5 iVQA: a new VideoQA evaluation dataset

In this section we present our Instructional VideoQA dataset (iVQA). We start from a subset of HowTo100M videos and manually annotate video clips with questions and answers. We aim (i) to provide a well-defined evaluation by including five correct answer annotations per question and (ii) to avoid questions which can be answered without watching the video. The dataset is described below.

**iVQA Data Collection.** iVQA videos are obtained by randomly sampling 7-30 sec. video clips from the HowTo100M dataset [69]. We avoid overlap between datasets and make sure iVQA and HowToVQA69M have no videos in common. Each clip is manually annotated with one question and 5 answers on Amazon Mechanical Turk. We ask workers to annotate questions about objects and scenes in the video and remove videos that could not be annotated. The correctness of annotations is manually verified by the authors. Moreover, we manually reduce the language bias by excluding questions that could be answered without watching the video. To increase diversity, each question is answered by 5 different workers. The answers are restricted to 4 words and are complemented by a confidence level. Questions that receive multiple answers with low confidence are removed. We further describe our data collection interfaces in [103] (Appendix C.1).

**Statistical analysis of iVQA.** iVQA contains 10,000 video clips with one question and five corresponding answers per clip. We split the dataset into 60%/20%/20% train/validation/test subsets. Figure 8 shows the distributions of question length, answer length, clip duration, clip relative start time in the original video and question types. The average duration of video clips is 18.6 seconds. Clip duration and start time distributions are almost uniform because we randomly sampled both the duration and the start time to obtain the clips, which results in a high video content diversity.
Most questions are about objects (What questions make up 91\% of the data), while some are about places (Where questions make up 5\% of the data) and people (Who questions make up 1\% of the data). On average, questions and answers contain 7.6 and 1.1 words, respectively. Answers are in great majority one or two words, which is a result of our collection procedure.

The majority of questions have a consensus between at least 2 annotators, i.e. at least 2 annotators providing the same answer. In detail, we observe that 27.0\% of questions lead to a perfect consensus among the five answer annotators, 48.4\% of questions lead to a consensus among at least four annotators, and 77.3\% lead to a consensus among at least three annotators. All but six questions lead to a consensus between at least two annotators. Additionally, 27.5\% of questions have two different answers that had a consensus between at least two annotators. Similarly to [7], this motivates us to define the following accuracy measure for a given answer $a$: $acc(a) = \min(\# \text{ground truth answers} = a, 1)$. This metric assigns 100\% accuracy to answers confirmed by at least 2 annotators, 50\% accuracy to answers confirmed by only 1 annotator and 0\% otherwise. Note that this definition is specific to our set-up where we have multiple ground truth answers per question.

Word clouds for questions and answers in the iVQA dataset in Figure 9 demonstrate the relation of iVQA to the domains of cooking, hand crafting and gardening. These word clouds also indicate that questions in iVQA often require spatial reasoning (behind, front, right, left) and temporal understanding (first, end, left, beginning) of the video. The most frequent answer (spoon) in iVQA corresponds to 2\% of all answers in the dataset. In contrast, the most frequent answers in other existing VideoQA datasets account for more than 9\% of all answers in these datasets (we have verified this for MSRVTT-QA, MSVD-QA and ActivityNet-QA). As a consequence, the most frequent answer baseline is significantly lower for our iVQA dataset compared to other VideoQA datasets. We further evaluate the language bias in iVQA in Section 6.8.

### Baselines

To evaluate the contribution of the visual modality, we compare our $VQA-T$ model with its language-only variant $QAT$. $QAT$ does not use video input, i.e. we set the input $v$ of the video-question transformer to zero during both training and testing (see Figure 7). To evaluate our generated dataset, we also compare $VQA-T$ trained on HowToVQA69M and on HowTo100M. Since HowTo100M has no $(v, q, a)$ triplets, we only train the $f$ branch of $VQA-T$ on HowTo100M using the standard masking and cross-modal matching losses [20], [56], [64], [87], [120]. In the zero-shot setting we evaluate $VQA-T$ trained on HowTo100M by computing $f(v, [q, a])$ for concatenated pairs of questions and answers $[q, a]$. During finetuning we also initialize

### Datasets

We use three datasets for training and five datasets for evaluation as described below. We follow previous evaluation protocols for open-ended settings [50] and use a fixed vocabulary of training answers. Unless stated otherwise, we report top-1 test accuracy and use original splits for training, validation and test.

For training we use our new HowToVQA69M dataset introduced in Section 3.2 with 90\% and 10\% videos in training and validation subsets. For comparison, we also train our model using a large-scale text-video dataset, HowTo100M [69], that contains videos with transcribed narrations but no video-question-answer triplets. Test and validation videos of downstream datasets are excluded from HowTo100M and HowToVQA69M. To evaluate the general applicability of our approach, we generate another automatic VQA dataset based on WebVid2M [9], which consists of 2.5M video-text pairs scraped from the web where video captions are obtained from readily-available alt-text descriptions, see Section 6.7.

We evaluate results on four open-ended VideoQA downstream datasets: MSRVTT-QA [100], MSVD-QA [100], ActivityNet-QA [110] and our new iVQA dataset (see Section 5). We also evaluate on a multiple-choice VideoQA dataset How2QA [56] where each question is associated with one correct and three incorrect answers. For MSRVTT-QA and MSVD-QA, we follow [50] and use a vocabulary of the top 4000 training answers for MSRVTT-QA, and all 1852 training answers for MSVD-QA. For our iVQA dataset and ActivityNet-QA, we consider all answers that appear at least twice in the training set, resulting in 2348 answers for iVQA and 1654 answers for ActivityNet-QA.

### Baselines

To evaluate the contribution of the visual modality, we compare our $VQA-T$ model with its language-only variant $QAT$. $QAT$ does not use video input, i.e. we set the input $v$ of the video-question transformer to zero during both training and testing (see Figure 7). To evaluate our generated dataset, we also compare $VQA-T$ trained on HowToVQA69M and on HowTo100M. Since HowTo100M has no $(v, q, a)$ triplets, we only train the $f$ branch of $VQA-T$ on HowTo100M using the standard masking and cross-modal matching losses [20], [56], [64], [87], [120]. In the zero-shot setting we evaluate $VQA-T$ trained on HowTo100M by computing $f(v, [q, a])$ for concatenated pairs of questions and answers $[q, a]$. During finetuning we also initialize

### 6 Experiments

This section demonstrates the benefits of training using our generated HowToVQA69M dataset and compares our method to the state of the art. We first outline the used datasets, baseline methods and implementation details in Section 6.1. Then we present results for the novel zero-shot VideoQA task in Section 6.2. Next we present results for the novel VideoQA feature probe evaluation setting in Section 6.3. The comparison to the state of the art in VideoQA and alternative training strategies is given in Section 6.4. Section 6.5 presents results for rare answers and split per question type. Then we compare our VideoQA generation approach to previous methods in Section 6.6. We also apply our approach to another video-text datasets in Section 6.7. Finally, we show the importance of the visual modality in iVQA in Section 6.8 and present ablation studies in Section 6.9.
TABLE 3: Comparison with baselines for zero-shot VideoQA. Top-1 and top-10 (for open-ended datasets) accuracy are reported.

| Method                  | Pretraining Data | iVQA  | MSRVTT-QA | MSVD-QA | ActivityNet-QA | How2QA |
|-------------------------|------------------|-------|-----------|---------|----------------|--------|
|                         |                  | Top-1 | Top-10    | Top-10  | Top-10         | Top-10 |
| Random                  | ∅                | 0.09  | 0.9       | 0.02    | 0.05           | 0.05   |
| QA-T                    | HowToVQA69M      | 4.4   | 23.2      | 2.5     | 6.5            | 4.8    |
| VQA-T                   | HowTo100M        | 1.9   | 11.9      | 0.3     | 3.4            | 1.4    |
| VQA-T (Ours)            | HowToVQA69M      | 12.2  | 43.3      | 2.9     | 8.8            | 7.5    |

TABLE 4: Probe evaluation of different pretraining strategies. In each case, only the last projection layers in the model were finetuned on the downstream VideoQA datasets. Top-1 accuracy is reported.

| Pretraining data | iVQA  | MSRVTT | MSVD | ActivityNet | How2QA |
|------------------|-------|--------|------|-------------|--------|
|                  | Top-1 | Top-10 | Top-10 | Top-10      | Top-10 |
| ∅                | 3.8   | 23.2   | 21.8  | 22.4        | 22.4   |
| QA-T             | 11.4  | 27.0   | 29.5  | 27.6        | 64.7   |
| QA-T             | 13.8  | 27.0   | 32.9  | 24.7        | 63.9   |
| VQA-T            | 24.5  | 32.9   | 39.0  | 30.6        | 72.9   |

TABLE 5: Benefits of pretraining our VQA-T model on our new HowToVQA69M dataset (last row) compared to no pretraining (first row) or pretraining on HowTo100M (second row). In each case our VQA-T model was then finetuned on the downstream VideoQA datasets. Top-1 accuracy is reported.

| Pretraining data | iVQA  | MSRVTT | MSVD | ActivityNet | How2QA |
|------------------|-------|--------|------|-------------|--------|
|                  | Top-1 | Top-10 | Top-10 | Top-10      | Top-10 |
| ∅                | 35.4  | 41.5   | 46.3  | 84.4        | 51.1   |
| QA-T             | 3.8   | 23.2   | 21.8  | 22.4        | 22.4   |
| QA-T             | 11.4  | 27.0   | 29.5  | 27.6        | 64.7   |
| QA-T             | 13.8  | 27.0   | 32.9  | 24.7        | 63.9   |
| VQA-T            | 24.5  | 32.9   | 39.0  | 30.6        | 72.9   |

6.2 Zero-shot VideoQA

In this section, we address the zero-shot VideoQA task where we prohibit any manual supervision of visual data during training. We explore this setup to evaluate the generalization of VQA-T trained on HowToVQA69M to unseen downstream datasets. For consistency, we use the vocabulary of answers from downstream datasets during testing (see Section 6.1).

Zero-shot results are presented in Table 5. We first observe that the use of visual cues by VQA-T outperforms QA-T when both models are trained on HowToVQA69M. This demonstrates the importance of the cross-modality in HowToVQA69M despite the VideoQA annotation being exclusively generated from text-only methods. Since HowToVQA69M has been generated using no manual annotation of visual data, our approach is scalable and can lead to further improvements by increasing the dataset size, as we discuss in Section 6.9.

Training on HowToVQA69M significantly outperforms the training on HowTo100M and the random baseline. This confirms the advantage of our HowToVQA69M dataset for the VideoQA task over other generic text-video datasets that do not contain video-question-answer triplets. We emphasize that our training does not use any information about target VideoQA datasets. Qualitative results for zero-shot VideoQA are presented for our approach and compared with baselines in Figure 10. We observe that QA-T (trained on HowToVQA69M) provides plausible but video-unrelated answers to the questions. Moreover, VQA-T (trained on...
TABLE 6: Results of our VQA-T model with different training strategies, on subsets of iVQA, MSRVTT-QA, MSVD-QA and ActivityNet-QA, corresponding to four quartiles with Q1 and Q4 corresponding to samples with the most frequent and the least frequent answers, respectively.

| Pretraining Data | Finetuning | iVQA | MSRVTT-QA | MSVD-QA | ActivityNet-QA |
|------------------|------------|------|-----------|---------|----------------|
|                  |            | Q1   | Q2        | Q3      | Q4             |
|                  |            | Q1   | Q2        | Q3      | Q4             |
|                  |            | Q1   | Q2        | Q3      | Q4             |
|                  |            | Q1   | Q2        | Q3      | Q4             |
|                  |            | Q1   | Q2        | Q3      | Q4             |

TABLE 7: Effect of our per questioning type on MSRVTT-QA and MSVD-QA.

| Pretraining Data | Finetuning | iVQA | MSRVTT-QA | MSVD-QA | ActivityNet-QA |
|------------------|------------|------|-----------|---------|----------------|
|                  |            | Q1   | Q2        | Q3      | Q4             |
|                  |            | Q1   | Q2        | Q3      | Q4             |
|                  |            | Q1   | Q2        | Q3      | Q4             |
|                  |            | Q1   | Q2        | Q3      | Q4             |
|                  |            | Q1   | Q2        | Q3      | Q4             |

TABLE 8: Effect of our per questioning type on ActivityNet-QA.

TABLE 9: Comparison of state of the art on MSRVTT-QA and MSVD-QA (top-1 accuracy).

| Method       | Pretraining data | MSRVTT-QA | MSVD-QA |
|--------------|------------------|-----------|---------|
| E-SA         | [100]            | 29.3      | 27.6    |
| ST-TP        | [41]             | 30.9      | 31.3    |
| AMU          | [107]            | 32.5      | 32.0    |
| Co-mem       | [42]             | 32.0      | 31.7    |
| HME          | [28]             | 33.0      | 33.7    |
| LAGCN        | [38]             | —         | 34.3    |
| HGA          | [43]             | 35.5      | 34.7    |
| QueST        | [42]             | 34.6      | 36.1    |
| HCRN         | [50]             | 35.6      | 36.1    |
| MASN         | [79]             | 35.2      | 38.0    |
| Bridge to Answer | [72]       | 36.9      | 37.2    |
| OCRL+LOGNet  | [23]             | —         | 36.0    | 38.2 |
| ClipBERT     | [52]             | 37.4      | —       |
| Jin et al.   | [44]             | 37.6      | 38.2    |
| SSML         | [5]              | 35.1      | 35.1    |
| CoMVT        | [80]             | 39.5      | 42.6    |
| SiaSamRea    | [109]            | 41.6      | 45.5    |
| MERLOT       | [172]            | 43.1      | —       |

TABLE 10: Comparison of state of the art on ActivityNet-QA and the public val set of How2QA (top-1 accuracy).

| VQA-T         | HowToVQA69M+   | HowToVQA69M+ |
|---------------|----------------|---------------|
|               | HowToVQA69M+   | HowToVQA69M+  |
|               | HowToVQA69M+   | HowToVQA69M+  |
|               | HowToVQA69M+   | HowToVQA69M+  |

TABLE 11: Comparison of state of the art on ActivityNet-QA and the public val set of How2QA (top-1 accuracy).
from the same domain as HowToVQA69M. Hence, the automatic generation of training data for other domains using our method can lead to further improvements on other datasets.

We compare our pretrained model to the state-of-the-art in VideoQA in Tables 9 and 10. Notably, VQA-T pretrained on HowToVQA69M outperforms previous methods using comparable pretraining data on all tested datasets. In particular, our method improves over CoMVT [80] that has been pretrained on HowTo100M. We note that the recent SiaSamRea approach [109] improves over our method on MSRVTT-QA (+0.1%) and ActivityNet-QA (+0.9%), but achieves lower results on MSVD-QA (-0.8%) and How2QA (-0.3%). However, SiaSamRea leverages manually annotated visual data for pretraining (COCO [18] and Visual Genome [49]). We also note that MERLOT [112] improves over our method on MSVT-QA and ActivityNet-QA, but uses the YT-Temporal-180M dataset for pretraining. This dataset includes HowTo100M but is significantly larger and more diverse (6 millions YouTube videos instead of 1 million).

### 6.5 Analysis of rare answers and question types

#### Results for rare answers
Training on downstream VideoQA datasets typically leads to particularly large improvements for questions with most frequent answers. As shown in Table 5, our approach brings significant improvements both for common and rare answers compared to models trained from scratch or pretrained on HowTo100M. We also find that our pretrained model, in the zero-shot setting, performs similarly across the different quartiles, with the exception of ActivityNet-QA, which includes in its most common answers yes, no. Interestingly, for the most rare answers in iVQA (Q3 and Q4) our model without finetuning (zero-shot mode) outperforms finetuned models that have not been pretrained on HowToVQA69M. We conclude that VideoQA specific pretraining on additional large-scale, diverse data helps improve generalization of VideoQA models.

Note that in order to have a consistent evaluation with other experiments, we keep the same train vocabulary at test time. This implies that a significant part of answers in the test set is considered wrong because the answer is not in the vocabulary. This represents 16% of answers in iVQA, 3% of answers in MSRVTT-QA, 6% for MSVD-QA and 19% for ActivityNet-QA. Note, however, that our joint embedding framework could allow for different vocabularies to be used at the training and test time.

#### Results split per question type
We also present results per question type for MSRVTT-QA, MSVD-QA and ActivityNet-QA in Tables 7 and 8. Compared to the model trained from scratch or the model pretrained on HowTo100M, we observe consistent improvements by our model for most categories.
6.6 Comparison of VideoQA generation methods and VideoQA training datasets

Comparison of VideoQA generation methods. We compare our question-answer generation approach to Heilman et al. [35], that was notably used in [100], [113], [115], [116], [117] to generate VideoQA data from video descriptions. We run the method of [35] on sentences extracted from HowTo100M, apply our pretraining method on the generated data and show results in Table 12 Note that we do not choose MSRVTT-QA and MSVD-QA as downstream datasets for this comparison because their evaluation sets were automatically generated using Heilman et al. [35]. We find that our generation method leads to significantly better performance both in zero-shot and finetuning settings. We supplement this quantitative comparison with a qualitative comparison shown in Figure 11. We found that compared to [35] our generation method provides higher quality as well as higher diversity of question-answer pairs when applied to the uncurated sentences extracted from speech in narrated videos. This further demonstrates the benefit of our transformer-based question-answer generation approach compared to previous rule-based methods.

Comparison of VideoQA training datasets. We also evaluate the importance of our generated HowToQA69M dataset by comparing our results to cross-dataset transfer using existing VideoQA datasets. We define cross-dataset transfer as a procedure where we pretrain our VideoQA model on a VideoQA dataset where we pretrain our VideoQA model on a VideoQA dataset and then finetune and test it on another VideoQA dataset. The training follows the procedure described for finetuning in Section 4.2. We report results for cross-dataset transfer in Table 12. We compare our results to cross-dataset transfer using existing VideoQA datasets. We define cross-dataset transfer as a procedure where we pretrain our VideoQA model on a VideoQA dataset and then finetune and test it on another VideoQA dataset. The training follows the procedure described for finetuning in Section 4.2. We report results for cross-dataset transfer in Table 12. We note that we do not use MSVD-QA as downstream dataset as its test set has been automatically generated with the same method [35] as MSRVTT-QA. As can be observed, our approach with pretraining on HowToQA69M significantly outperforms cross-dataset transfer models using the previously largest VideoQA dataset (MSRVTT-QA), or the largest manually annotated VideoQA dataset (ActivityNet-QA), both for the zero-shot and finetuning settings, on all four downstream datasets. We emphasize that our dataset is generated relying on text-only annotations, while MSRVTT-QA was generated using manually annotated video descriptions and ActivityNet-QA was manually collected. These
6.7 Generalization to other video-text datasets

In this section, we show that our VideoQA generation approach can be generalized to other sources of non-manually annotated video-text paired data. For this, we extend and apply our generation pipeline presented in Section 3.1 to video-alt-text description, i.e., alt-text HTML attribute associated with videos, from the WebVid2M dataset [9].

WebVidVQA3M dataset. We first explain how we adapt our generation pipeline detailed in Section 3.1 to video-alt-text pairs. As captions in WebVid2M are relatively short, we do not apply the punctuation model but directly apply the question-answer generation models on the captions. Captions in WebVid2M are also not temporally localized, so the generated question-answers are not temporally localized either. They instead refer to the whole videos, which are typically short (4 seconds on average). Applying our generation pipeline to WebVid2M [9], we generate WebVidVQA3M, a dataset of 3,476,610 question-answers associated with 2,404,871 videos. Examples of generated samples are illustrated in Figure 12. These examples show that despite a substantial visual-linguistic domain difference compared to HowTo100M, our approach is able to generate relevant VideoQA data. We believe that qualitatively, the generated QA data from WebVidVQA3M are of better quality than the generated QA data from HowToVQA69M (see Section 4.2). We argue that WebVid2M [9] has a better visual-linguistic correlation and a higher quality of text data compared to HowTo100M [29], which facilitates the VideoQA generation.

Benefits of training on WebVidVQA3M. We next apply our pretraining method on the generated data and show results in Table 15. We also explore combining both datasets with a simple curriculum learning strategy, where our model initially pretrained on HowToVQA69M is further trained on WebVidVQA3M. We find that training only on WebVidVQA3M gives competitive performance both in the zero-shot setting and the finetuning setting. Notably, it significantly improves over the variant trained from scratch in the finetuning setting. This shows that our approach can be generalized to other sources of video and text data. Additionally, we find that combining the two datasets for pretraining results in additional improvements both for zero-shot and finetuning. Therefore, the difference with previous methods is also increased (see Tables 9 and 10). Note that as WebVidVQA3M is significantly smaller than HowToVQA69M, our training runs faster on this dataset (20 GPUH instead of 350 GPUH), which gives a practical advantage to WebVidVQA3M. We have open-sourced WebVidVQA3M annotations to facilitate future research.

6.8 Importance of the visual modality in iVQA

We show in Table 14 that QA-T is a strong baseline compared to VQA-T on existing VideoQA datasets, when both are trained from scratch. However, on iVQA, VQA-T improves even more over QA-T than with other datasets, as measured by absolute improvement in top-1 accuracy. This suggests that the visual modality is more important in iVQA than in other VideoQA datasets.

6.9 Ablation studies

Pretraining losses. As shown in Table 15, removing duplicate negative answers in our contrastive loss, as discussed in Section 4.2, is beneficial notably in the zero-shot setting. Moreover, adding the MLM loss during pretraining improves the downstream results for both zero-shot and finetuning when used in combination with our contrastive learning strategy. These results motivate our proposed pretraining approach.

Importance of scale. Results of our method after pretraining on different fractions of HowToVQA69M are shown in Table 16. We construct these subsets such that larger subsets include the smaller ones. These results suggest that the scale is an important factor and that we can expect further improvements with additional pretraining data, both in the zero-shot and finetuning settings.

7 Conclusion

We propose a novel and scalable approach for training VideoQA models without manually annotated visual data. We automatically generate HowToVQA69M—a large-scale VideoQA training dataset generated from narrated videos with readily-available speech transcripts, significantly exceeding existing datasets by size and diversity. We demonstrate several benefits of pretraining on HowToVQA69M. We are the first to demonstrate zero-shot VideoQA results while using no manually annotated images or videos for training. We also introduce the VideoQA feature probe evaluation setting and show strong generalization capabilities of the multi-modal representation learnt by our pretrained model. Furthermore, finetuning our HowToVQA69M pretrained model on downstream tasks achieves competitive performance on MSRVTT-QA, ActivityNet-QA, MSVD-QA and How2QA. Moreover, we show that our approach generalizes to other sources of web videos by generating the WebVidVQA3M from video-alt-text pairs and showing its benefits for VideoQA training. We further validate our approach on our new manually-collected iVQA benchmark.

Acknowledgments

This work was granted access to the HPC resources of IDRIS under the allocation 2020-101267 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 and the European Regional Development Fund under project IMPACT (reg. no. CZ.02.1.01/0.0/0.0/15 003/0000468). We thank Pierre-Louis Guhur and Makarand Tapaswi for advice on using Amazon Mechanical Turk, Eloïse Berthier, Quentin Le Lidec and Elliot Chane-Sane for the manual evaluation of generated VideoQA data, and Ignacio Rocco for proofreading.

REFERENCES

[1] Punctuator. https://github.com/ottokart/punctuator2. 2017.
[2] Question generation using transformers. https://github.com/patil-suraj/question_generation. 2020.
[3] Chris Alberti, Daniel Andor, Emily Pitler, Jacob Devlin, and Michael Collins. Synthetic QA corpora generation with roundtrip consistency. In ACL, 2019.
[4] Chris Alberti, Jeffrey Ling, Michael Collins, and David Reitter. Fusion of detected objects in text for visual question answering. In IJCNLP, 2019.
[5] Elad Amrani, Rami Ben-Ari, Daniel Rotman, and Alex Bronstein. Noise estimation using density estimation for self-supervised multimodal learning. In AAAI, 2021.
[6] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-up and top-down attention for image captioning and visual question answering. In CVPR, 2018.
[7] Stanislav Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. VQA: Visual question answering. In ICCV, 2015.
[8] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. Layer normalization. arXiv preprint arXiv:1607.06450, 2016.
[9] Max Bain, Arshia Nagrani, Gail Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. In ICCV, 2021.
[10] Pratyay Banerjee, Tejas Gokhale, Yezhou Yang, and Chitta Baral. WeaQA: Weak supervision via captions for visual question answering. In Foundations of the Asociation for Computational Linguistics: ACL-IJCNLP 2021, 2021.
[11] Hedi Ben-Younes, Rémi Cadene, Matthieu Cord, and Nicolas Thome. MUTAN: Multimodal tucker fusion for visual question answering. In CVPR, 2017.
[12] Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. arXiv preprint arXiv:2005.14165, 2020.
[13] Mathilde Caron, Ishan Misra, Julian Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Emerging properties of transformers. In NeurIPS, 2020.
[14] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Multi-modal transformer for video retrieval. In IJCAI Workshops, 2020.
[15] Xi Chen, Binghao Fan, Xiaofan Zhang, Shu Zhang, Wensheng Wang, Chi Zhang, and Heng Huang. Heterogeneous memory enhanced multimodal attention model for video question answering. In CVPR, 2019.
[16] Marcello Federico, Sebastian Stüker, Luisa Bentivogli, Michael Paul, Mauro Cettolo, Teresa Herrmann, Jan Nickeus, and Giovanni Moretti. The IWSLT 2011 evaluation campaign on automatic talk translation. In LREC, 2012.
[17] Akira Fukui, Dong Huk Park, Daylen Yang, Anna Rohrbach, Trevor Darrell, and Marcus Rohrbach. Multimodal compact bilinear pooling for visual question answering and visual grounding. In EMNLP, 2016.
[18] Valentin Gabeur, Chen San, Karteek Alahari, and Cordelia Schmid. Multi-modal transformer for video retrieval. In ECCV, 2020.
[19] Jiaying Gao, Runzhou Ge, Kan Chen, and Ram Nevatia. Motion- appearance co-memory networks for video question answering. In CVPR, 2018.
[20] Noa Garcia, Mayu Otani, Chenhui Chu, and Yuta Nakashima. KnowIT VQA: Answering knowledge-based questions about videos. In AAAI, 2020.
[21] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the V in VQA matter: : Elevating the role of image understanding in visual question answering. In CVPR, 2017.
[22] Michael Heilman and Noah A Smith. Good question! Statistical ranking for question generation. In ACL, 2010.
[23] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (GELUs). arXiv preprint arXiv:1606.08415, 2016.
[24] Hexiang Hu, Wei-Lun Chao, and Fei Sha. Learning answer embeddings for visual question answering. In CVPR, 2018.
[25] Deng Huang, Peihao Chen, Runhao Zeng, Qing Du, Mingkui Tan, and Chuang Gan. Location-aware graph convolutional networks for video question answering. In AAAI, 2020.
[26] Zhiheng Huang, Zhaoyang Zeng, Yupan Huang, Bei Liu, Dongmei Fu, and Jianlong Fu. Seeing out of the box: End-to-end pre-training for vision-language representation learning. In CVPR, 2021.
[27] Zhiheng Huang, Zhaoyang Zeng, Bei Liu, Dongmei Fu, and Jianlong Fu. Pixel-BERT: Aligning image pixels with text by deep multi-modal transformers. arXiv preprint arXiv:2004.08849, 2020.
[28] Junyoung Kim, Yoo. Modality shifting attention network for multi-modal video question answering. In AAAI, 2020.
[29] Pin Jiang and Yahong Han. Reasoning with heterogeneous graph alignment for video question answering. In AAAI, 2020.
[30] Weike Jin, Zhou Zhao, Xiaochun Cao, Jieming Zhu, Xiuping He, and Yuting Zhuang. Adaptive spatio-temporal graph enhanced vision-language representation for video qa. IEEE Transactions on Image Processing, 2021.
[31] Hyounghun Kim, Zineng Tang, and Mohit Bansal. Dense-caption matching and frame-selection gating for temporal localization in VideoQA. In ACL, 2020.
[32] Junyoung Kim, Minuk Ma, Trung Pham, Kyungsu Kim, and Chang D Yoo. Modality shifting attention network for multi-modal video question answering. In CVPR, 2020.
[33] Kyung-Min Kim, Min-Oh Heo, Seong-Ho Choi, and Byoung-Tak Zhang. Deepstory: Video story qa by deep embedded memory networks. In IJCNN, 2017.
[34] Seonhoon Kim, Seoheyong Jeong, Eunbyul Kim, Inho Kang, and No-jun Kwak. Self-supervised pre-training and contrastive representation learning for multiple-choice video qa. In AAAI, 2021.
[35] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, Michael Bernstein, and Li Fei-Fei. Visual Genome: Connecting language and vision using crowdsourced dense image annotations. IJCV, 2016.
[36] Thao Minh Le, Vuong Le, Svetha Venkatesh, and Truyen Tran. Hier-
formers. In CVPR, 2022.

[105] Zekun Yang, Noa Garcia, Chenhui Chu, Mayu Otani, Yuta Nakashima, and Haruo Takeamura. BERT representations for video question answering. In WACV, 2020.

[106] Zichao Yang, Xiaodong He, Jianfeng Gao, Li Deng, and Alex Smola. Stacked attention networks for image question answering. In CVPR, 2016.

[107] Kaichun Yao, Libo Zhang, Tiejian Luo, Lili Tao, and Yanjun Wu. Teaching machines to ask questions. In IJCAI, 2018.

[108] Yunan Ye, Zhou Zhao, Yinming Li, Long Chen, Jun Xiao, and Yueting Zhuang. Video question answering via attribute-augmented attention network learning. In ACM SIGIR, 2017.

[109] Weijiang Yu, Haoteng Zheng, Mengfei Li, Lei Ji, Lijun Wu, Nong Xiao, and Nan Duan. Learning from inside: Self-driven siamese sampling and reasoning for video-question answering. In NeurIPS, 2021.

[110] Zhou Yu, Dejing Xu, Jun Yu, Ting Yu, Zhou Zhao, Yueting Zhuang, and Dacheng Tao. ActivityNet-QA: A dataset for understanding complex web videos via question answering. In AAAI, 2019.

[111] Amir Zadeh, Michael Chan, Paul Pu Liang, Edmund Tong, and Louis-Philippe Morency. Social-IQ: A question answering benchmark for artificial social intelligence. In CVPR, 2019.

[112] Rowan Zellers, Ximing Lu, Jack Hessel, Youngjae Yu, Jae Sung Park, Jize Cao, Ali Farhadi, and Yejin Choi. MERLOT: Multimodal neural script knowledge models. In NeurIPS, 2021.

[113] Kuo-Hao Zeng, Tseng-Hung Chen, Ching-Yao Chuang, Yuan-Hong Liao, Juan Carlos Niebles, and Min Sun. Leveraging video descriptions to learn video question answering. In AAAI, 2017.

[114] Zheng-Jun Zha, Jiawei Liu, Tianhao Yang, and Yongdong Zhang. Spatiotemporal-textual co-attention network for video question answering. ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM), 2019.

[115] Zhou Zhao, Shuwen Xiao, Zehan Song, Chujie Lu, Jun Xiao, and Yueting Zhuang. Open-ended video question answering via multi-modal conditional adversarial networks. IEEE Transactions on Image Processing, 2020.

[116] Zhou Zhao, Qifan Yang, Deng Cai, Xiaofei He, Yueting Zhuang, Zhou Zhao, Qifan Yang, Deng Cai, Xiaofei He, and Yueting Zhuang. Video question answering via hierarchical spatio-temporal attention networks. In IJCAI, 2017.

[117] Zhou Zhao, Zhu Zhang, Shuwen Xiao, Zhou Yu, Jun Yu, Deng Cai, Fei Wu, and Yueting Zhuang. Open-ended long-form video question answering via adaptive hierarchical reinforced networks. In IJCAI, 2018.

[118] Luowei Zhou, Hamid Palangi, Lei Zhang, Houdong Hu, Jason J Corso, and Jianfeng Gao. Unified vision-language pre-training for image captioning and VQA. In AAAI, 2020.

[119] Qingyu Zhou, Nan Yang, Furu Wei, Chuanqi Tan, Hangbo Bao, and Ming Zhou. Neural question generation from text: A preliminary study. In National CCF Conference on Natural Language Processing and Chinese Computing, 2017.

[120] Linchao Zhu and Yi Yang. ActBERT: Learning global-local video-text representations. In CVPR, 2020.

[121] Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In ICCV, 2015.

[122] Yueting Zhuang, Dejing Xu, Xin Yan, Wenzhuo Cheng, Zhou Zhao, Shiliang Pu, and Jun Xiao. Multichannel attention refinement for video question answering. ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM), 2020.
**APPENDIX**

In this Appendix, we give additional architecture details for our VideoQA model in Section A and additional implementation details in Section B.

**APPENDIX A**

**VideoQA Architecture**

Our architecture, detailed in Figure 13, has two main modules: (i) a video-question multi-modal transformer (top) and (ii) an answer transformer (bottom). Details are given next.

**Video-question multi-modal transformer.** The input video representation, obtained from a fixed S3D model [98], is composed of $t$ features denoted $v = [v_1, ..., v_t] \in \mathbb{R}^{d_v \times t}$ where $d_v$ is the dimension of the video features, and $t$ is the number of extracted features, one per second. The contextualized representation of the question, provided by the DistilBERT model [78], is composed of $l$ token embeddings denoted as $q = [q_1, ..., q_l] \in \mathbb{R}^{d_q \times l}$ where $d_q$ is the dimension of the DistilBERT embedding and $l$ is the number of tokens in the question. The inputs to our video-question multi-modal transformer are then defined as a concatenation of question token embeddings and video features

$$u(v, q) = [\tilde{q}_1, ..., \tilde{q}_l, \tilde{v}_1, ..., \tilde{v}_t] \in \mathbb{R}^{d \times (l+t)},$$

with

$$\tilde{q}_s = dp(\sigma(W_q q_s + b_q) + pos_s + mod_q),$$

and

$$\tilde{v}_s = dp(\sigma(W_v v_s + b_v) + pos_s + mod_v),$$

where $W_q \in \mathbb{R}^{d_q \times d}$, $b_q \in \mathbb{R}^d$, $W_v \in \mathbb{R}^{d_v \times d}$, $b_v \in \mathbb{R}^d$ and learnable parameters, $mod_q \in \mathbb{R}^d$ and $mod_v \in \mathbb{R}^d$ are learnt modality encodings for video and question, respectively, and $[pos_1, ..., pos_{l+t}] \in \mathbb{R}^{d \times (l+t)}$ are fixed sinusoidal positional encodings. $\sigma$ is a Gaussian Error Linear Unit [36] followed by a Layer Normalization [8] and $dp$ refers to Dropout [84].

The multi-modal transformer is a transformer with $N$ layers, $h$ heads, dropout probability $p_d$, and hidden dimension $d_h$. The outputs of the multi-modal transformer $[Q_1, ... Q_l, V_1, ... V_t] \in \mathbb{R}^{d \times (l+t)}$ are contextualized representations over tokens in the question and temporal video representations. Finally, the fused video-question embedding $f(v, q)$ is obtained as

$$F(Q_1) = W_{vq} dp(Q_1) + b_{vq},$$

where $W_{vq} \in \mathbb{R}^{d \times d}$, $b_{vq} \in \mathbb{R}^d$ are learnable parameters and $Q_1$ is the multi-modal contextualized embedding of the [CLS] token in the question, as shown in Figure 7.

**Answer transformer.** The contextualized representation of the answer, provided by the DistilBERT model [78], is composed of $m$ token embeddings denoted as $a = [a_1, ..., a_m] \in \mathbb{R}^{d_a \times m}$ where $d_a$ is the dimension of the DistilBERT embedding and $m$ is the number of tokens in the answer. Our answer embedding $g(a)$ is then obtained as

$$G(a_1) = W_a a_1 + b_a,$$

where $W_a \in \mathbb{R}^{d_a \times d}$, $b_a \in \mathbb{R}^d$ are learnable parameters and $a_1$ is the contextualized embedding of the [CLS] token in the answer, as shown in Figure 13.

---

**APPENDIX B**

**Additional Experimental Details**

Below we include additional details regarding the VideoQA generation, hyperparameter settings, training, masked language modeling and pretraining on HowTo100M.

**VideoQA generation.** The input sequence to the answer extractor and question generation transformers are truncated and padded up to a maximum of 32 tokens. The question decoding is done with beam search keeping track of the 4 most probable states at each level of the search tree. We have used the original captions (including stop words) from the HowTo100M dataset [69] and removed word repetitions from adjacent clips.

**VideoQA model.** We use the following hyperparameters: $t = 20$, $m = 10$, $d = 512$, $d_h = 2048$, $N = 2$, $H = 8$, $p_d = 0.1$, $d_q = d_a = 768$, $d_v = 1024$. The video features are sampled at equally spaced timestamps, and padded to length $t$. Sequences of question and answer tokens are truncated and padded to length $l$ and $m$, respectively. Attention is computed only on non-padded sequential video and question features.

**Training.** For finetuning, we use a cosine annealing learning rate schedule with initial value of $1 \times 10^{-5}$. We use the Adam optimizer with batch size of 256 and training runs for 20 epochs. The final model is selected by the best performance on the validation set.

**Masked Language Modeling.** For the masked language modeling objective, a token is corrupted with a probability 15%, and replaced 80% of the time with [MASK], 10% of the time with the same token and 10% of the time with a randomly sampled token. To guess which token is masked, each sequential question output $Q_i$ of the multi-modal transformer is classified in a vocabulary of 30,522 tokens, and we use a cross-entropy loss.

**Pretraining on HowTo100M.** For video-text cross-modal matching, we sample one video negative and one text negative per same token and 10% of the time with a randomly sampled token. To guess which token is masked, each sequential question output $Q_i$ of the multi-modal transformer is classified in a vocabulary of 30,522 tokens, and we use a cross-entropy loss.

The cross-modal matching module is used to perform zero-shot VideoQA for the variant $VQA-T$ trained on HowTo100M, by computing scores for $f(v, [q, a])$ for all possible answers $a$, for each video-question pair $(v, q)$. 