KnowIT VQA: Answering Knowledge-Based Questions about Videos

Noa Garcia,[1] Mayu Otani,[2] Chenhui Chu,[1] Yuta Nakashima[1]
[1]Osaka University, Japan, [2]CyberAgent, Inc., Japan
{noagarcia, chu, n-yuta}@ids.osaka-u.ac.jp
otani_mayu@cyberagent.co.jp

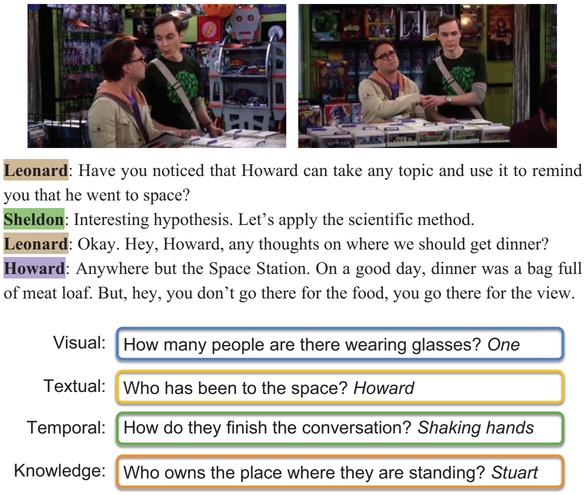
Abstract

We propose a novel video understanding task by fusing knowledge-based and video question answering. First, we introduce KnowIT VQA, a video dataset with 24,282 human-generated question-answer pairs about a popular sitcom. The dataset combines visual, textual and temporal coherence reasoning together with knowledge-based questions, which need of the experience obtained from the viewing of the series to be answered. Second, we propose a video understanding model by combining the visual and textual video content with specific knowledge about the show. Our main findings are: (i) the incorporation of knowledge produces outstanding improvements for VQA in video, and (ii) the performance on KnowIT VQA still lags well behind human accuracy, indicating its usefulness for studying current video modelling limitations.

Introduction

Visual question answering (VQA) was firstly introduced in (Malinowski and Fritz 2014) as a task for bringing together advancements in natural language processing and image understanding. Since then, VQA has experienced a huge development, in part due to the release of a large number of datasets, such as (Malinowski and Fritz 2014; Antol et al. 2015; Krishna et al. 2017; Johnson et al. 2017; Goyal et al. 2017). The current trend for addressing VQA (Anderson et al. 2018; Kim et al. 2017a; Ben-Younes et al. 2017; Bai et al. 2018) is based on predicting the correct answer from a multi-modal representation, obtained from encoding images with a pre-trained convolutional neural network (CNN) and attention mechanisms (Xu et al. 2015), and encoding questions with a recurrent neural network (RNN). These kinds of models infer answers by focusing on the content of the images (e.g. *How many people are there wearing glasses?* in Fig. 1).

Considering that the space spanned by the training question-image pairs is finite, the use of image content as the only source of information to predict answers presents two important limitations. On one hand, image features only capture the static information of the picture, leaving temporal coherence in video unattended (e.g. *How do they finish the conversation?* in Fig. 1), which is a strong constraint in real-world applications. On the other hand, visual content by itself does not provide enough insights for answering questions that require knowledge (e.g. *Who owns the place were they are standing?* in Fig. 1). To address these limitations, video question answering (VideoQA) (Tapaswi et al. 2016; Kim et al. 2017b; Lei et al. 2018) and knowledge-based visual question answering (KBVQA) (Wu et al. 2016; Wang et al. 2018) have emerged independently by proposing specific datasets and models. However, a common framework for addressing multi-question types in VQA is still missing.



Leonard: Have you noticed that Howard can take any topic and use it to remind you that he went to space?
Sheldon: Interesting hypothesis. Let's apply the scientific method.
Leonard: Okay. Hey, Howard, any thoughts on where we should get dinner?
Howard: Anywhere but the Space Station. On a good day, dinner was a bag full of meat loaf. But, hey, you don't go there for the food, you go there for the view.

Visual: How many people are there wearing glasses? *One*
Textual: Who has been to the space? *Howard*
Temporal: How do they finish the conversation? *Shaking hands*
Knowledge: Who owns the place where they are standing? *Stuart*

Figure 1: Types of questions addressed in KnowIt VQA.

The contribution of this work lies in this line, by introducing a general framework in which both video understanding and knowledge-based reasoning are required to answer questions. We first argue that a popular sitcom, such as The Big Bang Theory,[1] is an ideal testbed for modelling knowledge-based questions about the world. With this idea, we created KnowIT VQA,[2] a dataset for KBVQA in videos

[1] https://www.cbs.com/shows/big_bang_theory/

[2] Available at https://knowit-vqa.github.io/

Table 1: Comparison of VideoQA and KBVQA datasets. Answers are either multiple-choice (MC_N with N being the number of choices) or single word. Last four columns refer to the type of questions available in each dataset.

Dataset	**VQA-Type**	**Domain**	**# Imgs**	**# QAs**	**Answers**	**Vis.**	**Text.**	**Temp.**	**Know.**
MovieQA (Tapaswi et al. 2016)	Video	Movie	6,771	14,944	MC_5	✓	✓	✓	-
KB-VQA (Wang et al. 2017)	KB	COCO	700	2,402	Word	✓	-	-	✓
PororoQA (Kim et al. 2017b)	Video	Cartoon	16,066	8,913	MC_5	✓	✓	✓	-
TVQA (Lei et al. 2018)	Video	TV show	21,793	152,545	MC_5	✓	✓	✓	-
R-VQA (Lu et al. 2018)	KB	Visual Genome	60,473	198,889	Word	✓	-	-	-
FVQA (Wang et al. 2018)	KB	COCO, ImgNet	2,190	5,826	Word	✓	-	-	✓
KVQA (Shah et al. 2019)	KB	Wikipedia	24,602	183,007	Word	✓	-	-	✓
OK-VQA (Marino et al. 2019)	KB	COCO	14,031	14,055	Word	✓	-	-	✓
KnowIT VQA (Ours)	VideoKB	TV show	12,087	24,282	MC_4	✓	✓	✓	✓

in which real-world natural language questions are designed to be answerable only by people who is familiar with the show. We then cast the problem as a multi-choice challenge, and introduce a two-piece model that (i) acquires, processes, and maps specific knowledge into a continuous representation inferring the motivation behind each question, and (ii) fuses video and language content together with the acquired knowledge in a multi-modal fashion to predict the answer.

Related Work

Video Question Answering VideoQA addresses specific challenges with respect to the interpretation of temporal information in videos, including action recognition (Maharaj et al. 2017; Jang et al. 2017; Zellers et al. 2019; Mun et al. 2017), story understanding (Tapaswi et al. 2016; Kim et al. 2017b), or temporal coherence (Zhu et al. 2017). Depending on the video source, the visual content of videos may also be associated with textual data, such as subtitles or scripts, which provide an extra level of context for its interpretation. Most of the proposed datasets so far are mainly focused on either the textual or the visual aspect of the video, without exploiting the combination of both modalities. In MovieQA (Tapaswi et al. 2016), for example, questions are mainly plot-focused, whereas in other collections, questions are purely about the visual content, such as action recognition in MovieFIB (Maharaj et al. 2017), TGIF-QA (Jang et al. 2017), and MarioVQA (Mun et al. 2017), or temporal coherence in Video Context QA (Zhu et al. 2017)). Only few datasets, such as PororoQA (Kim et al. 2017b) or TVQA (Lei et al. 2018), present benchmarks for exploiting multiple sources of information, requiring models to jointly interpret multi-modal video representations. Even so, reasoning beyond the video content in these kinds of approaches is complicated, as only the knowledge acquired in the training samples is used to generate the answer.

Knowledge-Based Visual Question Answering Answering questions about a visual query by only using its content constrains the output to be inferred within the space of knowledge contained in the training set. Considering that the amount of training data in any dataset is finite, the knowledge used to predict answers in standard visual question answering is rather limited. In order to answer questions beyond the image content, KBVQA proposes to inform VQA models with external knowledge. The way of acquiring and incorporating this knowledge, however, is still in early stages. For example, (Zhu et al. 2015) creates a specific knowledge base with image-focused data for answering questions under a certain template, whereas more generic approaches (Wu et al. 2016) extract information from external knowledge bases, such as DBpedia (Auer et al. 2007), for improving VQA accuracy. As VQA datasets do not envisage questions with general information about the world, specific KBVQA datasets have been recently introduced, including KB-VQA (Wang et al. 2017) with question-images pairs generated from templates, R-VQA (Lu et al. 2018) with relational facts supporting each question, FVQA (Wang et al. 2018) with supporting facts extracted from generic knowledge bases, KVQA (Shah et al. 2019) for entity identification, or OK-VQA (Marino et al. 2019) with free-form questions without knowledge annotations. Most of these datasets impose hard constraints on their questions, such as being generated by templates (KB-VQA) or directly obtained from existing knowledge bases (FVQA), being OK-VQA the only one that requires handling unstructured knowledge to answer natural questions about images. Following this direction, we present a framework for answering general questions that may or may not be associated with a knowledge base by introducing a new VideoQA dataset, in which questions are freely proposed by qualified workers to study knowledge and temporal coherence together. To the best of our knowledge, this is the first work that explores external knowledge questions in a collection of videos.

KnowIT VQA Dataset

Due to the natural structure of TV shows, in which characters, scenes, and general development of the story can be known in advance, TV data has been exploited for modelling real-world scenarios in video understanding tasks (Nagrani and Zisserman 2017; Fremmann, Cohen, and Lapata 2018). We also rely on this idea and argue that popular sitcoms provide an ideal testbed to encourage progress in knowledge-based visual question answering, due to their additional facilities to model knowledge and temporal coherence over time. In particular, we introduce the KnowIT VQA dataset, (standing for knowledge informed temporal VQA), a collection of videos from The Big Bang Theory annotated with knowledge-based questions and answers about the show.

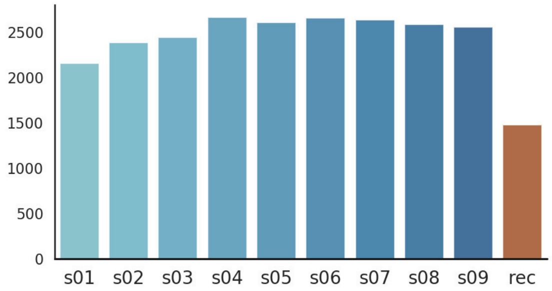
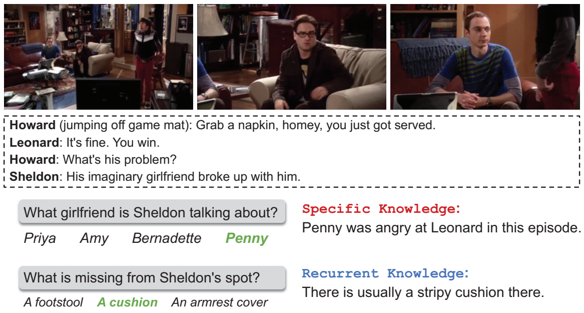


Figure 2: Number of questions by KNOWLEDGE TYPE.



Howard (jumping off game mat): Grab a napkin, homey, you just got served.
Leonard: It's fine. You win.
Howard: What's his problem?
Sheldon: His imaginary girlfriend broke up with him.

What girlfriend is Sheldon talking about?

Priya *Amy* *Bernadette* ***Penny***

Specific Knowledge:
Penny was angry at Leonard in this episode.

What is missing from Sheldon's spot?

A footstool ***A cushion*** *An armrest cover*

Recurrent Knowledge:
There is usually a stripy cushion there.

Figure 3: Episode-specific versus recurrent KNOWLEDGE.

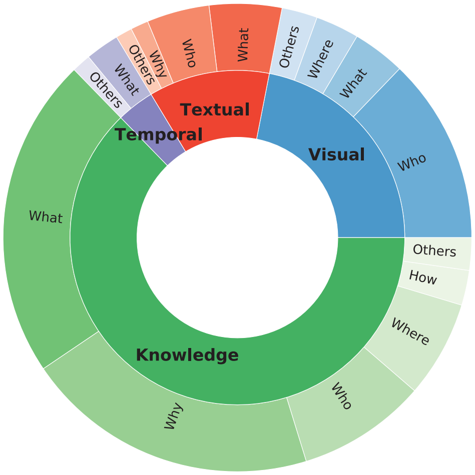


Figure 4: Distribution of questions in the test set by their first word for each question type.

Video Collection

Our dataset contains both visual and textual video data. Videos are collected from the first nine seasons of The Big Bang Theory TV show, with 207 episodes of about 20 minutes long each. For the textual data, we obtained the subtitles directly from the DVDs. Additionally, we downloaded episode transcripts from a specialised website.[3] Whereas subtitles are annotated with temporal information, transcripts associate dialog with characters. We align subtitles and transcripts with dynamic programming so that each subtitle is annotated to both its speaker and its timestamp. Transcripts also contain scene information, which is used to segment each episode into video scenes. Scenes are split uniformly into 20 seconds clips, obtaining 12,264 clips in total.

QA Generation

To generate real-world natural language questions and answers, we used Amazon Mechanical Turk (AMT)[4]. We required workers to have a high knowledge about The Big Bang Theory and instructed them to write knowledge-based questions about the show. Our aim was to generate questions answerable only by people familiar with the show, whereas difficult for new spectators. For each clip, we showed workers the video and subtitles, along with a link to the episode transcript and summaries of all the episodes for extra context. Workers were asked to annotate each clip with a question, its correct answer, and three wrong but relevant answers. The QA generation process was done in batches of one season at a time in two different rounds. During the second round, we showed the already collected data for each clip in order to 1) get feedback on the quality of the collected data and 2) obtain a diverse set of questions. The QA collection process took about 3 months.

Knowledge Annotations

We define knowledge as the information that is not contained in a given video clip. To approximate the knowledge the viewers acquire by watching the series, we annotated each QA pair with expert information:

- KNOWLEDGE: the information that is required to answer the question represented by a short sentence. For example, for the question *Why did Leonard invite Penny to lunch?*, the information *Penny has just moved in* is key to respond the correct answer, *He wanted Penny to feel welcomed into the building*, over the other three candidates.[5]
- KNOWLEDGE TYPE: whether the knowledge is from the same episode (episode-specific) or it occurs repeatedly during the show (recurrent). The distribution between the two classes is shown in Fig. 2, with 6.08% of the samples being recurrent and the rest being almost uniformly distributed over the nine seasons. Examples of recurrent and episode-specific KNOWLEDGE are show in Figure 3.
- QUESTION TYPE: we establish four different types of questions: 1) visual-based (22%), in which the answer is found in the video frames, 2) textual-based (12%), in which the answer is found in the subtitles, 3) temporal-based (4%), in which the answer is predictable from the current video clip at a specific time, 4) knowledge-based (62%), in which the answer is not found in the current

[3] https://bigbangtrans.wordpress.com/

[4] https://www.mturk.com

[5] 1) *Because he didn't have enough money to eat alone*, 2) *Because he wanted Sheldon to practice his social skills*, and 3) *Because he was in love with Penny*.

Table 2: KnowIT VQA data splits and the average lengths.

	Train	Val	Test	Total
# Episodes	167	20	20	207
# Scenes	2,007	225	240	2,472
# Clips	9,731	1,178	1,178	12,087
# QAs	19,569	2,352	2,361	24,282
Len. Subtitles	56.49	55.57	57.45	56.49
Len. Questions	7.5	7.38	7.48	7.49
Len. CA	4.55	4.51	4.46	4.54
Len. WA	4.14	4.12	4.06	4.13
Len. KNOWLEDGE	10.43	10.10	10.30	10.39

clip, but in another sequence of the show. To encourage the development of general purpose models, QUESTION TYPE is only provided for the test set. The distribution of question words in each type is plotted in Figure 4.

Data Splits

We collected 24,282 samples from 12,087 video clips. We randomly split the episodes into training, validation, and test sets, so that questions and clips from the same episode were assigned to the same set. The number of episodes, clips, and QA pairs in each split are detailed in Table 2, as well as the average number of tokens in subtitles, questions, answers, and KNOWLEDGE. Correct answers (CA) are slightly longer than wrong answers (WA), which is a common bias in QA datasets (Tapaswi et al. 2016; Lei et al. 2018).

Dataset Comparison

In Table 1, we compare our dataset against other VideoQA and KBVQA datasets. KBVQA datasets are usually smaller than standard VQA datasets, as QA generation is often more challenging. Nevertheless, KnowIT VQA with 24k questions is the largest KBVQA human-generated dataset, far from the 2.4k questions in KB-VQA, 5.8k in FVQA, and 14k in OK-VQA. Also, KnowIT VQA is the first collection addressing the four aforementioned types of questions. Note that the visual domain in KnowIT VQA is not new, sharing a small portion of videos (about 34%) with TVQA. However, whereas TVQA uses 3.6k clips per show in average, the KNOWLEDGE annotations in our dataset required a larger set of clips, in order to approximate the knowledge that spectators acquire by watching the series.

Human Evaluation

We performed human evaluation on the KnowIT VQA test set with a four-fold aim: 1) to evaluate whether video clips are relevant to answer questions; 2) to evaluate whether the questions *do* require knowledge to be answered; 3) to evaluate whether the KNOWLEDGE annotations are useful for answering the questions in the dataset; and 4) to introduce a human performance baseline for model comparison.

Table 3: Human evaluation on KnowIT VQA test set.

Group	Acc	Group	Acc
Rookies, Blind	0.440	Masters, Blind	0.651
Rookies, Subs	0.562	Masters, Subs	0.789
Rookies, Video	0.748	Masters, Video	0.896

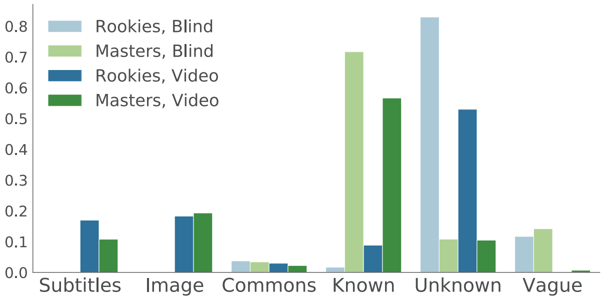


Figure 5: Distribution of reasons for answering by groups.

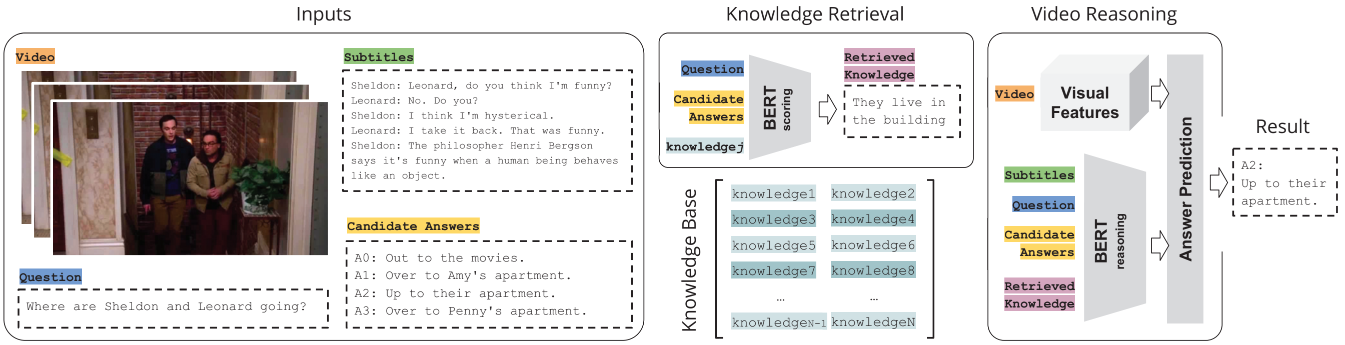
Evaluation Design We used AMT with independent groups of workers for each task.[6] We split workers according to their experience with the show, i.e., *masters*, who have watched at least the first nine seasons of the show, and *rookies*, who have never watched any episode. We conducted two main tasks: evaluation on the questions and evaluation on the KNOWLEDGE annotations.

Evaluation on the questions We further split masters and rookies into 3 different sub-groups according to the data provided to answer each question: Blind (only QAs), Subs (QAs and subtitles), and Video (QAs, subtitles, and clips). For each question in the test set, we asked workers to choose the correct answer from the four given candidates and to provide the reason for their response, from six possible options.[7] In each group, each question was answered by 3 workers. Results are reported in Table 3. The accuracy gap between Subs and Video groups confirms the relevance of the video content in the dataset. With respect to knowledge, the difference between masters and rookies strongly supports the claim that KnowIT VQA is extremely challenging when not knowing the show. When looking into the reasons for choosing the answer (Fig. 5), we saw that masters mostly based their choices on the knowledge acquired when watching the show, whereas rookies admitted not knowing the correct answer in most of their responses.

Evaluation on the knowledge We studied the quality of the collected KNOWLEDGE and its relevance to the questions in the dataset. We asked a group of rookies to answer the questions in the test set. For each question and candidate

[6] Workers participating on the creation of the dataset were not allowed to participate in the evaluation.

[7] i) The answer is in the subtitles, ii) The answer is in the image, iii) The answer is common-sense knowledge, iv) I know the episode, v) I have no idea about the answer, and vi) The question is too vague to be answered.



Sheldon: Leonard, do you think I'm funny?
Leonard: No. Do you?
Sheldon: I think I'm hysterical.
Leonard: I take it back. That was funny.
Sheldon: The philosopher Henri Bergson says it's funny when a human being behaves like an object.

A0: Out to the movies.
A1: Over to Amy's apartment.
A2: Up to their apartment.
A3: Over to Penny's apartment.

Where are Sheldon and Leonard going?

They live in the building

A2: Up to their apartment.

Figure 6: Overview of ROCK. In the knowledge retrieval module, the question and candidate answers are used to retrieve knowledge instances in the KB with the BERT-scoring network. In the video reasoning part, visual features are extracted from the video, whereas subtitles, questions, candidate answers, and the retrieved knowledge instances are fed into the BERT-reasoning network. The visual and language representations are fused and fed into a classifier to predict the correct answer.

answers, we provided the subtitles and video clip. After answering, we showed the associated KNOWLEDGE and asked them to answer again. As a result, we found that the accuracy increased from 0.683 (before KNOWLEDGE exposition) to 0.823 (after KNOWLEDGE exposition), verifying the relevance of the KNOWLEDGE annotations to the questions.

ROCK Model

We propose ROCK (Retrieval Over Collected Knowledge), a model for addressing knowledge-based visual question answering in videos, as depicted in Fig. 6. ROCK is based on the availability of language instances representing the show information in a knowledge base (KB), which ROCK retrieves and fuses with language and spatio-temporal video representations for reasoning over the question and predict the correct answer.

Knowledge Base

We first create a knowledge base (KB) to emulate the knowledge a viewer acquires when watching the series. Differently from previous work (Wu et al. 2016; Wang et al. 2018), which is based on generic knowledge graphs, such as DBpedia (Auer et al. 2007), our problem requires to access specific information about the show. Thus, we rely on the AMT workers annotations provided in the KNOWLEDGE field during the dataset collection.[8]

The collected KNOWLEDGE is provided as natural language sentences. For example, to the question *What was Raj doing at Penny's?*, the annotated KNOWLEDGE is:

Raj wanted to ask Missy on a date, because Howard and Leonard had already asked her but failed, however his medication wore off and he couldn't do it.

As it is unclear how to capture such complex processes in an structured fashion such as knowledge graphs, we build a KB, $K = \{w_j | j = 1, \ldots, N\}$, such that knowledge instances w_j's are represented as natural language sentences. We additionally perform a cleaning process to remove near-duplicate instances, reducing N from 24,282 to 19,821.

Knowledge Base Cleaning To remove near-duplicate samples in the KB, we compute similarities between KNOWLEDGE instances. For each $w_j \in K$, we create an input sentence, w'_j, as a concatenation of strings:

$$w'_j = \texttt{[CLS]} + w_j + \texttt{[SEP]},$$

where [CLS] is a token to indicate the start of the sequence and [SEP] is a token to indicate the end. We tokenise w'_j into a sequence of 60 tokens, $\boldsymbol{\tau}_j$. Let $\mathrm{BERT}_{\mathrm{P}}(\cdot)$ be a pre-trained BERT network (Devlin et al. 2019), which takes as input a sequence of tokens and outputs the vector corresponding to the [CLS] token. We obtain the high-dimensional projection, $\mathbf{p}_j$, of w_j as:

$$\mathbf{p}_j = \mathrm{BERT}_{\mathrm{P}}(\boldsymbol{\tau}_j) \tag{1}$$

To measure similarity between a pair of instances, $w_i, w_j \in K$, we compute a similarity score, β_{ij}, as:

$$\beta_{ij} = \mathrm{sim}(\mathbf{p}_i, \mathbf{p}_j), \tag{2}$$

where $\mathrm{sim}(\cdot, \cdot)$ is the cosine similarity. Next, we build an undirected graph, $G = (V, E)$, in which nodes, $V = \{w_j | j = 1, \cdots, N\}$, correspond to KNOWLEDGE instances, and edges, $e = (w_i, w_j) \in E$, connect instances when $\beta_{ij} > 0.998$.[9] To find near duplicate instances, we create clusters of nodes, C_l with $l = 1, \cdots, L$, by finding all the connected components in G, i.e. C_l corresponds to the l-th subgraph in G, for which all nodes are connected to each other by a path of edges. We randomly choose one node in each cluster and remove the others.

[8] As future work, we plan to study how to automatically learn to generate similar explanations from another module that directly 'watches' the series and extracts knowledge from videos.

[9] We experimentally found 0.998 to be a good tradeoff between near duplicates and semantically similar instances

Knowledge Retrieval Module

Inspired by the ranking system in (Nogueira and Cho 2019), the knowledge retrieval module uses a question q_i and its candidate answers a_i^c with $c \in \{0, 1, 2, 3\}$ to query the knowledge base K and rank knowledge instances $w_j \in K$ according to a relevance score s_{ij}.

We first obtain a sequence input representation x_{ij} as a concatenation of strings:

$$x_{ij} = \texttt{[CLS]} + q_i + a_i^{\alpha_0} + a_i^{\alpha_1} + a_i^{\alpha_2} + a_i^{\alpha_3} + \texttt{[SEP]} + w_j + \texttt{[SEP]},$$

where `[SEP]` separates the input text used for querying and the knowledge to be queried. Although preliminary experiments showed that the order of the answers a_i^c does not have a high impact on the results, for an invariant model we automatically sort the answers according to a prior relevance score. α_c is then the original position of the answer with c-th highest score. Details are provided below.

We tokenise x_{ij} into a sequence of n words $\mathbf{x}_{ij}$[10] and input it into a BERT network, namely BERT-scoring denoted by $\textsc{Bert}_\text{S}(\mathbf{x}_{ij})$, whose output is the vector corresponding to the `[CLS]` token. To compute s_{ij}, we use a fully connected layer together with a sigmoid activation as:

$$s_{ij} = \text{sigmoid}(\mathbf{w}_\text{S}^\top \cdot \textsc{Bert}_\text{S}(\mathbf{x}_{ij}) + b_\text{S}), \tag{3}$$

where $\mathbf{w}_\text{S}$ and b_S are the weight vector and the bias scalar of the fully connected layer, respectively. $\textsc{Bert}_\theta^\text{S}$, $\mathbf{w}_\text{S}$, and b_S are fine-tuned using matching (i.e. $i = j$) and non-matching (i.e. $i \neq j$) QA-knowledge pairs with the following loss:

$$\mathcal{L} = -\sum_{i=j} \log(s_{ij}) - \sum_{i \neq j} \log(1 - s_{ij}) \tag{4}$$

For each q_i, all w_j's in K are ranked according to s_{ij}. The top k ranked instances, i.e. the most relevant samples for the query question, are retrieved.

Prior Score Computation To prevent the model producing different outputs for different candidate answer order, we create an answer order-invariant model by sorting answers, a^c with $c = \{0, 1, 2, 3\}$, according to a prior score, ξ^c.

For a given question q, ξ^c is obtained from predicting the score of a^c being the correct answer. We first build an input sentence e^c as the concatenation of the strings:

$$e^c = \texttt{[CLS]} + q + \texttt{[SEP]} + a^c + \texttt{[SEP]},$$

and we tokenise e^c into a sequence of 120 tokens, $\mathbf{e}^c$. If $\textsc{Bert}_\text{E}(\cdot)$ represents a BERT network whose output is the vector corresponding to the `[CLS]` token, ξ^c is obtained as:

$$\xi^c = \mathbf{w}_\text{E}^\top \textsc{Bert}_\text{E}(\mathbf{e}^c) + b_\text{E}, \tag{5}$$

Finally, all ξ^c with $c = \{0, 1, 2, 3\}$ are sorted in descending order into $\boldsymbol{\xi}$ and answers are ordered according to $\alpha_c = \delta$, where δ is the position of the δ-th highest score in $\boldsymbol{\xi}$.

Video Reasoning Module

In this module, the retrieved knowledge instances are jointly processed with the multi-modal representations from the video content to predict the correct answer. This process contains three components: visual representation, language representation, and answer prediction.

Visual Representation We sample n_f frames from each video clip and apply four different techniques to describe their visual content:

- *Image features*: Each frame is fed into Resnet50 (He et al. 2016) without the last fully-connected layer and is represented by a 2,048-dimensional vector. We concatenate all vectors from the n_f frames and condense it into a 512-dimensional vector using a fully-connected layer.
- *Concepts features*: For a given frame, we use the bottom-up object detector (Anderson et al. 2018) to obtain a list of objects and attributes. We encode all the objects and attributes in the n_f frames into a C-dimensional bag-of-concept representation, which is projected into a 512-dimensional space with a fully-connected layer. C is the total number of available objects and attributes.
- *Facial features*: We use between 3 to 18 photos of the main cast of the show to train the state-of-the-art face recognition network in (Parkhi et al. 2015).[11] For each clip, we encode the detected faces as a F-dimensional bag-of-faces representation, which is projected into a 512-dimensional space with a fully-connected layer. F is the total number of people trained in the network.
- *Caption features*: For each frame, we generate a caption to describe its visual content using (Xu et al. 2015). The n_f captions extracted from each clip are passed to the language representation model.

Language Representation Text data is processed using a fine-tuned BERT model, namely BERT-reasoning. We compute the language input, y^c, as a concatenation of strings:

$$y^c = \texttt{[CLS]} + \textit{caps} + \textit{subs} + q + \texttt{[SEP]} + a^c + w + \texttt{[SEP]},$$

where $caps$ is the concatenated n_f captions (ordered by timestamp), $subs$ the subtitles, and w the concatenated k retrieved knowledge instances. For each question q, four different y^c are generated, one for each of the candidate answers a^c with $c = \{0, 1, 2, 3\}$. We tokenise y^c into a sequence of m words, $\mathbf{y}^c$, as in BERT-scoring. Let $\textsc{Bert}_\text{R}$ denote BERT-reasoning, whose output is the vector corresponding to the `[CLS]` token. For a^c, the language representation $\mathbf{u}^c$ is obtained as $\mathbf{u}^c = \textsc{Bert}_\text{R}(\mathbf{y}^c)$.

[10] Sequences longer than n are truncated, and sequences shorter than n are zero-padded.

[11] Characters trained in the face recognition network are: Amy, Barry, Bernadette, Dr. Beverly Hofstadter, Dr. VM Koothrappali, Emily, Howard, Leonard, Leslie, Lucy, Mary Cooper, Penny, Priya, Raj, Sheldon, Stuart, and Wil Wheaton.

Answer Prediction To predict the correct answer, we concatenate the visual representation $\mathbf{v}$ (i.e. image, concepts, or facial features) with one of the language representations $\mathbf{u}^c$:

$$\mathbf{z}^c = [\mathbf{v}, \mathbf{u}^c], \tag{6}$$

$\mathbf{z}^c$ is projected into a single score, o^c, with a fully-connected layer:

$$o^c = \mathbf{w}_{\text{R}}^\top \mathbf{z}^c + b_{\text{R}}, \tag{7}$$

The predicted answer $\hat{a}$ is obtained with the index of the maximum value in $\mathbf{o} = (o^0, o^1, o^2, o^3)^\top$, i.e., $\hat{a} = a^{\arg\max_c \mathbf{o}}$. Being c^* the correct class, BERT_{R}, $\mathbf{w}_{\text{R}}$, and b_{R} are fine-tuned with the multi-class cross-entropy loss as:

$$\mathcal{L}(\mathbf{o}, c^*) = -\log \frac{\exp(o^{c^*})}{\sum_c \exp(o^c)} \tag{8}$$

Experimental Results

We evaluated and compared ROCK against several baselines on the KnowIT VQA dataset. Results per question type and overall accuracy are reported in Table 4. Models were trained with stochastic gradient descent with momentum of 0.9 and learning rate of 0.001. In BERT implementations, we used the uncased base model with pre-trained initialisation.

Table 4: Accuracy for different methods on KnowIt VQA dataset. ♢ for parts of our model, ★ for our full model.

	Model	Vis.	Text.	Temp.	Know.	All
	Random	0.250	0.250	0.250	0.250	0.250
Answers	Longest	0.324	0.308	0.395	0.342	0.336
	Shortest	0.241	0.236	0.233	0.297	0.275
	word2vec sim	0.166	0.196	0.233	0.189	0.186
	BERT sim	0.199	0.239	0.198	0.226	0.220
QA	word2vec sim	0.108	0.163	0.151	0.180	0.161
	BERT sim	0.174	0.264	0.209	0.190	0.196
	TFIDF	0.434	0.377	0.488	0.485	0.461
	LSTM Emb.	0.444	0.428	0.512	0.515	0.489
	LSTM BERT	0.446	0.464	0.500	0.532	0.504
	♢ ROCK$_{\text{QA}}$	0.542	0.475	0.547	0.535	0.530
	Humans (Rookies, Blind)	0.406	0.407	0.418	0.461	0.440
Subs, QA	LSTM Emb.	0.432	0.362	0.512	0.496	0.467
	LSTM BERT	0.452	0.446	0.547	0.530	0.504
	TVQA$_{\text{SQA}}$	0.602	0.551	0.512	0.468	0.509
	♢ ROCK$_{\text{SQA}}$	0.651	0.754	0.593	0.534	0.587
	Humans (Rookies, Subs)	0.618	0.837	0.453	0.498	0.562
Vis, Subs, QA	TVQA	0.612	0.645	0.547	0.466	0.522
	♢ ROCK$_{\text{VSQA}}$ Image	0.643	0.739	0.581	0.539	0.587
	♢ ROCK$_{\text{VSQA}}$ Concepts	0.647	0.743	0.581	0.538	0.587
	♢ ROCK$_{\text{VSQA}}$ Facial	0.649	0.743	0.581	0.537	0.587
	♢ ROCK$_{\text{VSQA}}$ Caption	0.666	0.772	0.581	0.514	0.580
	Humans (Rookies, Video)	0.936	0.932	0.624	0.655	0.748
Knowledge	★ ROCK Image	0.654	0.681	0.628	0.647	0.652
	★ ROCK Concepts	0.654	0.685	0.628	0.646	0.652
	★ ROCK Facial	0.654	0.688	0.628	0.646	0.652
	★ ROCK Caption	0.647	0.678	0.593	0.643	0.646
	ROCK$_{\text{GT}}$	0.747	0.819	0.756	0.708	0.731
	Humans (Masters, Video)	0.961	0.936	0.857	0.867	0.896

Answers To detect potential biases in the dataset, we evaluated the accuracy of predicting the correct answer by only considering the candidate answers:

- Longest/Shortest: The predicted answer is the one with the largest/smallest number of words.
- word2vec/BERT sim: For word2vec, we use 300-dimensional pre-trained word2vec vectors (Mikolov et al. 2013). For BERT, we encode words with the output of the third-to-last layer of pre-trained BERT. Answers are encoded as the mean of their word representations. The prediction is the answer with the highest cosine similarity to the other candidates in average.

In general, these baselines performed very poorly, with only Longest being better than random. Other than the tendency of correct answers to be longer, results do not show any strong biases in terms of answer similarities.

QA We also evaluated several baselines in which only questions and candidate answers are considered.

- word2vec/BERT sim: Questions and answers are represented by the mean word2vec or pre-trained BERT word representation. The predicted answer is the one with highest cosine similarity to the question.
- TFIDF: Questions and answers are represented as a weighted frequency word vector (tf-idf) and projected into a 512-dimensional space. The question and the four answer candidates are then concatenated and input into a four-class classifier to predict the correct answer.
- LSTM Emb./BERT: Each word in a question or in a candidate answer is encoded through an embedding layer or a pre-trained BERT network and input into an LSTM (Hochreiter and Schmidhuber 1997). The last hidden state of the LSTM is used as a 512-dimensional sentence representation. Question and answers are concatenated and input into a four-class classifier for prediction.
- ROCK$_{\text{QA}}$: ROCK model with $m = 120$ tokens, trained and evaluated only with questions and answers as input.

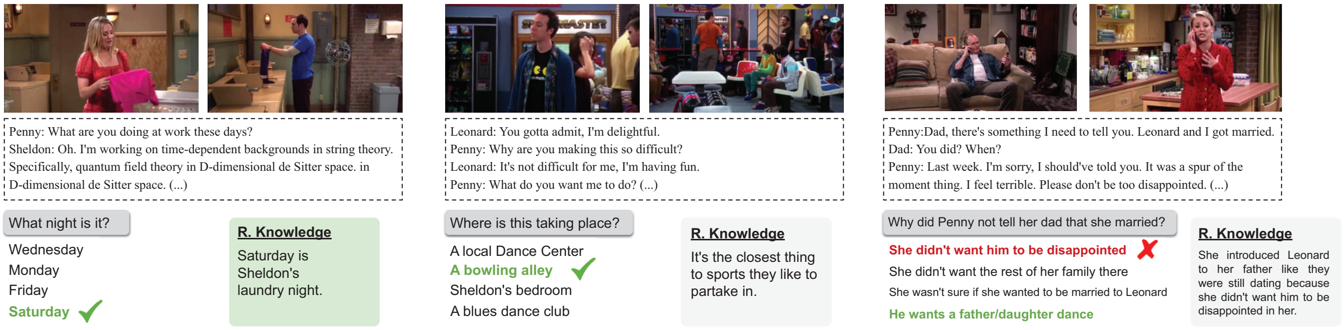
Whereas methods based on sentence similarity performed worse than random, methods with classification layers trained for answer prediction (i.e. TFIDF, LSTM Emb./BERT, and ROCK$_{\text{QA}}$) obtained considerably better accuracy, even outperforming human workers.

Subs, QA Models that use subtitles, questions, and answers as input.

- LSTM Emb./BERT: Subtitles are encoded with another LSTM and concatenated to the question and answer candidates before being fed into the four-class classifier.
- TVQA$_{\text{SQA}}$ (Lei et al. 2018): Language is encoded with a LSTM layer and no visual information is used.
- ROCK$_{\text{SQA}}$: With $m = 120$ tokens, the input sequence only includes subtitles, questions, and candidate answers.

LSTM BERT and ROCK$_{\text{SQA}}$ improved accuracy by a 5.7% with respect to only questions and answers. On the other hand, LSTM Emb. did not improve compared to the models using only QA, which may imply a limitation in the word embeddings to encode long sequences in subtitles.

Vis, Sub, QA VideoQA models based on both language and visual representations.



Penny: What are you doing at work these days?
Sheldon: Oh. I'm working on time-dependent backgrounds in string theory. Specifically, quantum field theory in D-dimensional de Sitter space. in D-dimensional de Sitter space. (...)

What night is it?
Wednesday
Monday
Friday
Saturday ✓

R. Knowledge
Saturday is Sheldon's laundry night.

Leonard: You gotta admit, I'm delightful.
Penny: Why are you making this so difficult?
Leonard: It's not difficult for me, I'm having fun.
Penny: What do you want me to do? (...)

Where is this taking place?
A local Dance Center
A bowling alley ✓
Sheldon's bedroom
A blues dance club

R. Knowledge
It's the closest thing to sports they like to partake in.

Penny:Dad, there's something I need to tell you. Leonard and I got married.
Dad: You did? When?
Penny: Last week. I'm sorry, I should've told you. It was a spur of the moment thing. I feel terrible. Please don't be too disappointed. (...)

Why did Penny not tell her dad that she married?
She didn't want him to be disappointed ✗
She didn't want the rest of her family there
She wasn't sure if she wanted to be married to Leonard
He wants a father/daughter dance

R. Knowledge
She introduced Leonard to her father like they were still dating because she didn't want him to be disappointed in her.

Figure 7: Qualitative results of the ROCK (Image) model. Left: the retrieved knowledge (RK) helps to predict the answer. Middle: the RK is not accurate, but the model still predicts the correct answer. Right: the RK is incorrect and leads to a misprediction.

- TVQA (Lei et al. 2018): State-of-the-art VideoQA method. Language is encoded with a LSTM layer, whereas visual data is encoded into visual concepts.
- $\text{ROCK}_{\text{VSQA}}$: Our model with $m = 120$ tokens and $n_f = 5$ frames. Four different visual representations are used.

$\text{ROCK}_{\text{VSQA}}$ outperformed TVQA by 6.6%, being Concepts the features with the highest accuracy. However, any of the visual models outperformed ROCK_{SQA}, implying strong limitations in current video modelling approaches.

Knowledge Models that exploit KNOWLEDGE to predict the correct answer, i.e. our ROCK model in its full version, with $n = 128$ and $k = 5$ in the knowledge retrieval module, and $m = 512$ in the video reasoning module. Compared to the non-knowledge methods, the inclusion of the knowledge retrieval module increased the accuracy by 6.5%, showing the great potential of knowledge-based approaches in our dataset. Among the visual representations, Image, Concepts, and Facial performed the same. However, when compared against human masters, ROCK lags well behind, suggesting potential room for improvement. When using the annotated KNOWLEDGE instead of the retrieved one (ROCK_{GT}), accuracy is boosted to 0.731, indicating that improvements in the knowledge retrieval module will increase the overall performance. Finally, qualitative results are presented in Fig. 7, providing some insights on the strengths and weaknesses of our model.

Knowledge Retrieval Results Results for the knowledge retrieval module, in terms of recall at K (R@K) and median rank (MR), are shown in Table 5. We tested different arrangements in the input data:

- Only Questions: Candidate answers were not used.
- QA parameter sharing: In the input string, x_{ij}, only one answer, a^c, at a time was used as $x_{ij} = \texttt{[CLS]} + q_i + a^c_i + \texttt{[SEP]} + w_j + \texttt{[SEP]}$, which means that the same parameters are used for the four candidate answers.
- QA prior score: Our proposed method based on ordering answers according to their prior score.

Table 5: Knowledge retrieval module results on test set.

Method	**R@1**	**R@5**	**R@10**	**R@100**	**MR**
Only Questions	0.070	0.169	0.208	0.426	221
QA param sharing	0.083	0.197	0.268	0.557	67
QA prior score	0.114	0.259	0.318	0.576	53

There was a big gap between Only Questions and the other two methods, indicating that candidate answers contained relevant information to retrieve the correct knowledge. The best results were obtained with our proposed prior scoring method, which showed that using all the candidate answers together provided more context for finding the correct knowledge instance.

Conclusion

We presented a novel dataset for knowledge-based visual question answering in videos and proposed a video reasoning model, in which multi-modal video information was combined together with specific knowledge about the task. Our evaluation showed the great potential of knowledge-based models in video understanding problems. However, there is still a big gap with respect to human performance, which we hope our dataset will contribute to reduce by encouraging the development of stronger models.

Acknowledgements

This work is based on results obtained from a project commissioned by the New Energy and Industrial Technology Development Organization (NEDO). It was also partly supported by JSPS KAKENHI No. 18H03264 and JST ACT-I.

References

Anderson, P.; He, X.; Buehler, C.; Teney, D.; Johnson, M.; Gould, S.; and Zhang, L. 2018. Bottom-up and top-down attention for image captioning and visual question answering. In *Proc. CVPR*, 6077–6086.

Antol, S.; Agrawal, A.; Lu, J.; Mitchell, M.; Batra, D.; Lawrence Zitnick, C.; and Parikh, D. 2015. VQA: Visual question answering. In *Proc. ICCV*, 2425–2433.

Auer, S.; Bizer, C.; Kobilarov, G.; Lehmann, J.; Cyganiak, R.; and Ives, Z. 2007. DBpedia: A nucleus for a web of open data. In *The Semantic Web*. Springer. 722–735.

Bai, Y.; Fu, J.; Zhao, T.; and Mei, T. 2018. Deep attention neural tensor network for visual question answering. In *Proc. ECCV*, 20–35.

Ben-Younes, H.; Cadene, R.; Cord, M.; and Thome, N. 2017. MUTAN: Multimodal tucker fusion for visual question answering. In *Proc. ICCV*, 2612–2620.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proc. NAACL*, 4171–4186.

Fremann, L.; Cohen, S. B.; and Lapata, M. 2018. Whodunnit? Crime drama as a case for natural language understanding. *Trans. ACL* 6:1–15.

Goyal, Y.; Khot, T.; Summers-Stay, D.; Batra, D.; and Parikh, D. 2017. Making the V in VQA matter: Elevating the role of image understanding in visual question answering. In *Proc. CVPR*, 6904–6913.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proc. CVPR*, 770–778.

Hochreiter, S., and Schmidhuber, J. 1997. Long short-term memory. *Neural Computation* 9(8).

Jang, Y.; Song, Y.; Yu, Y.; Kim, Y.; and Kim, G. 2017. TGIF-QA: Toward spatio-temporal reasoning in visual question answering. In *Proc. CVPR*, 2758–2766.

Johnson, J.; Hariharan, B.; van der Maaten, L.; Fei-Fei, L.; Zitnick, C. L.; and Girshick, R. 2017. CLEVR: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proc. CVPR*, 1988–1997.

Kim, J.-H.; On, K.-W.; Lim, W.; Kim, J.; Ha, J.-W.; and Zhang, B.-T. 2017a. Hadamard product for low-rank bilinear pooling. *Proc. ICLR*.

Kim, K.-M.; Heo, M.-O.; Choi, S.-H.; and Zhang, B.-T. 2017b. DeepStory: Video story QA by deep embedded memory networks. In *Proc. IJCAI*, 2016–2022.

Krishna, R.; Zhu, Y.; Groth, O.; Johnson, J.; Hata, K.; Kravitz, J.; Chen, S.; Kalantidis, Y.; Li, L.-J.; Shamma, D. A.; et al. 2017. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *IJCV* 123(1):32–73.

Lei, J.; Yu, L.; Bansal, M.; and Berg, T. L. 2018. Tvqa: Localized, compositional video question answering. In *Proc. EMNLP*, 1369–1379.

Lu, P.; Ji, L.; Zhang, W.; Duan, N.; Zhou, M.; and Wang, J. 2018. R-vqa: learning visual relation facts with semantic attention for visual question answering. In *Proc. KDD*, 1880–1889.

Maharaj, T.; Ballas, N.; Rohrbach, A.; Courville, A.; and Pal, C. 2017. A dataset and exploration of models for understanding video data through fill-in-the-blank question-answering. In *Proc. CVPR*, 6884–6893.

Malinowski, M., and Fritz, M. 2014. A multi-world approach to question answering about real-world scenes based on uncertain input. In *Proc. NIPS*, 1682–1690.

Marino, K.; Rastegari, M.; Farhadi, A.; and Mottaghi, R. 2019. Ok-vqa: A visual question answering benchmark requiring external knowledge. In *Proc. CVPR*, 3195–3204.

Mikolov, T.; Sutskever, I.; Chen, K.; Corrado, G. S.; and Dean, J. 2013. Distributed representations of words and phrases and their compositionality. In *Proc. NIPS*.

Mun, J.; Hongsuck Seo, P.; Jung, I.; and Han, B. 2017. MarioQA: Answering questions by watching gameplay videos. In *Proc. ICCV*, 2867–2875.

Nagrani, A., and Zisserman, A. 2017. From Benedict Cumberbatch to Sherlock Holmes: Character identification in TV series without a script. In *Proc. BMVC*.

Nogueira, R., and Cho, K. 2019. Passage re-ranking with bert. *arXiv preprint arXiv:1901.04085*.

Parkhi, O. M.; Vedaldi, A.; Zisserman, A.; et al. 2015. Deep face recognition. In *Proc. BMVC*, 1–6.

Shah, S.; Mishra, A.; Yadati, N.; and Talukdar, P. P. 2019. KVQA: Knowledge-aware visual question answering. In *Proc. AAAI*.

Tapaswi, M.; Zhu, Y.; Stiefelhagen, R.; Torralba, A.; Urtasun, R.; and Fidler, S. 2016. MovieQA: Understanding stories in movies through question-answering. In *Proc. CVPR*, 4631–4640.

Wang, P.; Wu, Q.; Shen, C.; Dick, A.; and Van Den Henge, A. 2017. Explicit knowledge-based reasoning for visual question answering. In *Proc. IJCAI*, 1290–1296.

Wang, P.; Wu, Q.; Shen, C.; Dick, A.; and van den Hengel, A. 2018. FVQA: Fact-based visual question answering. *IEEE Trans. PAMI* 40(10):2413–2427.

Wu, Q.; Wang, P.; Shen, C.; Dick, A.; and van den Hengel, A. 2016. Ask me anything: Free-form visual question answering based on knowledge from external sources. In *Proc. CVPR*, 4622–4630.

Xu, K.; Ba, J.; Kiros, R.; Cho, K.; Courville, A.; Salakhudinov, R.; Zemel, R.; and Bengio, Y. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *Proc. ICML*, 2048–2057.

Zellers, R.; Bisk, Y.; Farhadi, A.; and Choi, Y. 2019. From recognition to cognition: Visual commonsense reasoning. In *Proc. CVPR*.

Zhu, Y.; Zhang, C.; Ré, C.; and Fei-Fei, L. 2015. Building a large-scale multimodal knowledge base system for answering visual queries. *arXiv preprint arXiv:1507.05670*.

Zhu, L.; Xu, Z.; Yang, Y.; and Hauptmann, A. G. 2017. Uncovering the temporal context for video question answering. *IJCV* 124(3):409–421.