

HowToCaption: Prompting LLMs to Transform Video Annotations at Scale Supplementary Material

Nina Shvetsova^{*1,2,3}, Anna Kukleva^{*2}, Xudong Hong^{2,6},
Christian Rupprecht⁴, Bernt Schiele², Hilde Kuehne^{1,3,5}

¹Goethe University Frankfurt, ²MPI for Informatics, SIC, ³University of Bonn,
⁴University of Oxford, ⁵MIT-IBM Watson AI Lab, ⁶Saarland University
{nshvetso, akukleva}@mpi-inf.mpg.de

In the appendix, we provide additional experimental results, implementation and dataset details, and qualitative examples. Additionally, we discuss limitations and responsibility to human subjects. Specifically, we first discuss the experimental results of using MiniGPT-4 [19] to generate visually grounded captions in Sec. A. Then, we provide additional experimental results in Sec. B, including prompt engineering experiments in Sec. B.3, ablations of our filtering & alignment method in Sec. B.4, and robustness analysis in Sec. B.5. Then, we provide additional implementation details in Sec. C, statistics of the HowToCaption dataset in Sec. D, and qualitative examples in Sec. E. Finally, we discuss our method limitations in Sec. F and responsibility to human subjects in Sec. G.

A Grounding Captions to Video Content with MiniGPT-4

Our generated captions with Vicuna-13B are based solely on ASR subtitles. To additionally ground the produced captions on visual content, we experiment with the recent MiniGPT-4 model [19]. The MiniGPT-4 consists of the frozen Vicuna-13B model and a visual encoder with a Q-Former [6] that projects visual features from an image into tokens in a language model embedding space that are later treated as word tokens in the Vicuna-13B model. To ground generated captions on the visual modality, we create a grid image from 4 uniformly sampled frames from a video clip and slightly adapt the prompt to encourage the LLM to utilize the provided image for generating captions (Tab. A.1). We apply our approach to obtain visually grounded captions with the MiniGPT-4 model and obtain *HowToCaption-grounded*. For this dataset, we follow exactly the same hyperparameters that we use for HowToCaption. In Tab. A.2, we evaluate the downstream retrieval performance of the T-V model trained on HowToCaption-grounded. The dataset shows mixed results compared to the HowToCaption; while it is beneficial for the MSR-VTT and the MSVD dataset, performance on the YouCook2 dataset drops. To facilitate further analysis, we will release both caption sets: the ASR-based only HowToCaption, produced by the Vicuna-13B, and HowToCaption-grounded, produced by the MiniGPT-4.

* equal contribution

Table A.1: Prompts for the Vicuna-13B and MiniGPT-4 models. Difference is highlighted with bold.

Vicuna-13B	MiniGPT-4
I will give you an automatically recognized speech with timestamps from a video segment that is cut from a long video. Write a summary for this video segment. Write only short sentences. Describe only one action per sentence. Keep only actions that happen in the present time. Begin each sentence with an estimated timestamp. Here is this automatically recognized speech: <ASR with timestamps>	I will give you an automatically recognized speech with timestamps and an image with four frames from a video segment that is cut from a long video. Write a summary for this video segment based on both: video frames and speech. Write only short sentences. Describe only one action per sentence. Keep only actions that happen in the present time. Begin each sentence with an estimated timestamp. Here is the image with four frames: <grid-image here> . Here is the automatically recognized speech: <ASR with timestamps>

Table A.2: Comparison of HowToCaption and HowToCaption-grounded datasets obtained with Vicuna-13b and MiniGPT-4 large language models, respectively. For each dataset, we train a T-V model and report downstream zero-shot text-video retrieval performance.

Dataset	YouCook2				MSR-VTT				MSVD				LSMDC			
	R1	R5	R10	MR	R1	R5	R10	MR	R1	R5	R10	MR	R1	R5	R10	MR
HowToCaption (Vicuna-13B)	13.4	33.1	44.1	15	37.6	62.0	73.3	3	44.5	73.3	82.1	2	17.3	31.7	38.6	29
HowToCaption-grounded (MiniGPT-4)	12.4	29.8	39.9	20.5	38.3	62.5	73.2	2	46.2	73.9	82.5	2	16.8	31.0	38.7	27

B Additional Experimental Evaluation

B.1 Dataset Quality

In this section, we provide an additional quantitative evaluation of the quality of captions in our HowToCaption dataset.

Evaluation Using the HIREST Dataset. First, we evaluate the caption quality using the HIREST dataset [18] as the ground truth. The HIREST dataset contains step captions for a small subset of video clips from the HowTo100M dataset. Since the video clip boundaries of the HIREST dataset, our HowToCaption dataset, and the HowTo100M dataset differ, we use a subset of 423 HIREST (video clip, caption) pairs that have corresponding (video clip’, caption’) and (video clip”, caption”) pairs in the HowToCaption and HowTo100M datasets respectively, such as the temporal boundaries of (video clip and video clip’) overlaps with > 0.5 IoU, as well as (video clip and video clip”) overlaps with > 0.5 IoU. Using HIREST captions as ground truth, we report the quality of captions in the HowTo100M and HowToCaption datasets using standard metrics such as BLEU@4 and METEOR, as shown in Tab. B.1. Despite the common disadvantages of classical metrics, which focus more on wording rather than semantics, we observe consistent improvement in the captions of our HowToCaption dataset over the ASR subtitles.

User Study. To quantitatively evaluate the “human-written”-like quality of our captions, we performed a user study with 60 random captions: 20 from HowToCaption, 20 ASR subtitles, and 20 captions from the downstream datasets,

Table B.1: Caption quality evaluation in the HowTo100M and HowToCaption datasets using the HIREST dataset as the ground truth. We use a subset of 423 HIREST video clip-caption pairs that temporally overlap with video clip-caption pairs in both the HowToCaption and HowTo100M datasets with > 0.5 IoU. We report standard text generation metrics.

	BLEU@4	METEOR	ROUGE-L	CIDEr
HowTo100M with ASRs	0.4	9.8	11.4	3.1
HowToCaption (ours)	1.0	9.3	13.6	29.6

Table B.2: Additional experiments with LLM prompts. We report modifications that we have done in comparison to our default prompt, which is highlighted. With each prompt, we obtain 2M video-text pairs from 100k HowTo100M videos that we later use for T-V model training (low-recourse setup). Downstream zero-shot text-video retrieval performance is reported.

Prompt	YouCook2		MSR-VTT		MSVD		LSMDC		Average	
	R10↑	MR↓	R10↑	MR↓	R10↑	MR↓	R10↑	MR↓	R10↑	MR↓
I will give you an automatically recognized speech with timestamps from a video segment that is cut from a long video. Write a summary for this video segment. Write only short sentences. Describe only one action per sentence. Keep only actions that happen in the present time. Begin each sentence with an estimated timestamp. Here is this automatically recognized speech: <ASR with timestamps in the format “n”: “ASR”> (ours)	40.6	19	72.0	3	81.6	2	37.7	30	58.0	13.5
Modification: Write a summary for this video segment. → Write a likely summary for this video segment.	40.8	18.5	71.4	3	81.5	2	37.7	30	57.9	13.4
Modification: Write a summary for this video segment. → Write a creative summary for this video segment.	40.0	19	71.6	3	81.2	2	37.8	27	57.7	12.8
Modification: <ASR with timestamps in the format “n”: “ASR”> → <ASR with timestamps in the format “minutes”:“seconds”: “ASR subtitle”>	40.8	18.5	71.5	3	81.2	2	37.2	29	57.7	13.1

namely YouCook2, MSRVT, MSVD, and LSMDC (5 captions per dataset), which served as a control group. We asked users to identify *human-written video captions*. Using majority voting from 30 participants, 90% of the control group, 50% of HowToCaption, and only 15% of ASR captions were chosen as “human-written” captions. This demonstrates that our captions are significantly more “human-written” than the ASR subtitles.

B.2 Additional Comparisons with SOTA in Text-Video Retrieval

In addition to comparing dual encoder-only models in zero-shot text-to-video retrieval, in Tab. B.3, we compare methods that use a visual-text matching head for *re-ranking* the best candidates [1, 5]. Specifically, we evaluate our model with all BLIP architecture blocks used in video captioning evaluation (as described in Sec. 4.4 of the main paper and in Appendix C) by re-ranking 128

Table B.3: Comparison of zero-shot text-to-video retrieval methods with re-ranking. We include methods that perform a re-ranking of the top-N retrieved candidates using a visual-text matching head. *BLIP_{CapFilt-L} (initialization of our model). †Text-video model only (without audio and subtitles).

Method	V. Encoder	YouCook2				MSR-VTT				MSVD				LSMDC			
		R1	R5	R10	MR	R1	R5	R10	MR	R1	R5	R10	MR	R1	R5	R10	MR
BLIP* [6]	ViT-B	10.7	24.1	32.3	44.5	41.4	63.3	72.8	2	45.6	71.3	79.7	2	20.5	36.5	44.2	18.5
BLIP*, HowToCaption-finetuned	ViT-B	18.15	38.68	50.4	10	44.3	66.6	76.6	2	49	76.2	84.2	2	19.6	34.3	42.9	18
BLIP, COCO-finetuned [6]	ViT-B	-	-	-	-	43.3	65.6	74.7	2	48.7	75.8	83.1	2	22.3	38.4	44.1	18
Unmasked Teacher-17M [7]	ViT-L	-	-	-	-	42.6	64.4	73.1	-	49.9	77.7	85.3	-	25.2	43	50.5	-
mPLUG, COCO-finetuned [5]	ViT-L	-	-	-	-	44.3	66.4	75.4	-	-	-	-	-	-	-	-	-
mPLUG-2 [17]	ViT-L	-	-	-	-	47.1	69.7	79	-	-	-	-	-	24.1	43.8	52	-
VAST [1]	ViT-G	-	-	-	-	49.3	68.3	73.9	2	-	-	-	-	-	-	-	-
VAST† [1]	ViT-G	15.7	35.2	45.6	14	48.5	71.2	79.8	2	50.6	76.2	84.1	1	23.2	40.9	48.9	12
VAST†, HowToCaption-finetuned	ViT-G	19.7	43.6	53.9	8	50	73.2	81.4	1	54.8	80.9	87.2	1	27.7	46.5	54.6	7

best candidates from dual encoder model predictions using a visual-text matching head. Moreover, we fine-tune the state-of-the-art text-video VAST model [1] for 9k interactions on the HowToCaption dataset with a batch size of 64, a learning rate of 5e-06, using 4 frames per video clip, and following other training/evaluation parameters of VAST. We observe that fine-tuning on the HowToCaption dataset boosts the performance of both models. Furthermore, the text-video VAST model fine-tuned on the HowToCaption dataset outperforms all other models.

B.3 Additional Results in Prompt Engineering

In Tab. B.4, we provide an additional evaluation of language prompts. First, we experiment with phrases such as “write a *likely* summary...” and “write a *creative* summary...”. While the keyword “likely” almost does not change downstream performance, the keyword “creative” is not beneficial for 3 out of 4 datasets. We also experiment with another timestamp format in the LLM prompt. Namely, instead of using “*n*’s” (such as 0s, 65s), we use “*minutes*:*seconds*” format (such as 00:00, 01:05). We found that simple timestamp format “*n*’s” results in a higher performance.

B.4 Ablations of Filtering & Alignment Post-processing

We present ablations of our alignment & filtering post-processing in Tab. B.5. In Tab. B.5a, we ablate two modifications of the fine-tuning and alignment processes for the second round of filtering & alignment. We observe that the dataset obtained after the second round of filtering & alignment without these modifications shows lower performance than the dataset obtained with the first round (using the BLIP model). We attribute this to forgetting during fine-tuning. However, we note that both proposed modifications boost performance, as well as their combination.

In Tab. B.5b, we also analyze if more than two rounds of filtering & alignment lead to a better quality dataset. We employ 20k iterations of fine-tuning of the

Table B.4: Additional experiments with LLM prompts. We report modifications that we have done in comparison to our default prompt, which is highlighted. With each prompt, we obtain 2M video-text pairs from 100k HowTo100M videos that we later use for T-V model training (low-recourse setup). Downstream zero-shot text-video retrieval performance is reported.

Prompt	YouCook2		MSR-VTT		MSVD		LSMDC		Average	
	R10↑	MR↓	R10↑	MR↓	R10↑	MR↓	R10↑	MR↓	R10↑	MR↓
I will give you an automatically recognized speech with timestamps from a video segment that is cut from a long video. Write a summary for this video segment. Write only short sentences. Describe only one action per sentence. Keep only actions that happen in the present time. Begin each sentence with an estimated timestamp. Here is this automatically recognized speech: <ASR with timestamps in the format “n”: “ASR”> (ours)	40.6	19	72.0	3	81.6	2	37.7	30	58.0	13.5
Modification: Write a summary for this video segment. → Write a likely summary for this video segment.	40.8	18.5	71.4	3	81.5	2	37.7	30	57.9	13.4
Modification: Write a summary for this video segment. → Write a creative summary for this video segment.	40.0	19	71.6	3	81.2	2	37.8	27	57.7	12.8
Modification: <ASR with timestamps in the format “n”: “ASR”> → <ASR with timestamps in the format “minutes”:“seconds”: “ASR subtitle”>	40.8	18.5	71.5	3	81.2	2	37.2	29	57.7	13.1

T-V model on the obtained dataset after each filtering & alignment round. We do not observe any performance boost with more filtering & alignment rounds.

We further ablate the range of alignment time offsets T and filtering threshold κ used in our alignment & filtering method in Tab. B.5c and Tab. B.5d respectively. We found that alignment with up to $T = 10$ seconds offsets and filtering with the threshold that selects 25M most similar video-text pairs result in the highest performance.

B.5 Robustness Analysis

We further analyze the robustness of our HowToCaption method to noisy input. First, we examine how the method performs if the video channel is corrupted, specifically containing only empty black frames. We replace all 1.2M videos with videos of the same length but containing only black frames while keeping the original ASR subtitles of the videos. Consequently, the generated captions from subtitles remain the same, but the videos are corrupted. After our alignment & filtering procedure, we find that only $\sim 1.8\%$ of generated captions are matched. Therefore, our method filters out almost all captions for corrupted videos. Furthermore, we test the case of data where ASR subtitles do not match the videos. For each video among the 1.2M videos, we randomly assign ASR subtitles from another video of the dataset. In this case, generated captions from subtitles are the same, but they do not correspond to the videos. We find that after applying the HowToCaption method to such input data, only $\sim 2.4\%$ of captions are

Table B.5: Ablation of our alignment & filtering method. With each post-processing variant, we obtain a dataset that we later use for T-V model training. Downstream zero-shot text-video retrieval performance is reported. Options used to obtain the main results are highlighted.

(a) Components of our filtering & alignment method.

Caption post-processing	YouCook2		MSR-VTT		MSVD		LSMDC		Average	
	R10↑	MR↓	R10↑	MR↓	R10↑	MR↓	R10↑	MR↓	R10↑	MR↓
Alignment & filtering (using the BLIP)	42.4	17	71.7	3	82.2	2	38.5	29.5	58.7	12.9
Alignment & filtering after second round	42.4	17	69.4	3	81.2	2	38.1	33	57.8	13.8
+ regularization L_{align}	44.3	15	71.9	3	81.9	2	39.0	28	59.3	12.0
+ averaging similarities of the finetuned and original model	43.7	15	72.8	3	82.0	2	39.6	27	59.5	11.8
+ regularization L_{align} + averaging similarities of the finetuned and original model (ours)	44.1	15	73.3	3	82.1	2	38.6	29	59.5	12.3

(b) Rounds of alignment & filtering with finetuning of the T-V model after each round.

Caption post-processing	YouCook2		MSR-VTT		MSVD		LSMDC		Average	
	R10↑	MR↓	R10↑	MR↓	R10↑	MR↓	R10↑	MR↓	R10↑	MR↓
Alignment & filtering (using the BLIP) = 1 round	42.4	17	71.7	3	82.2	2	38.5	29.5	58.7	12.9
Alignment & filtering after 2'nd round (ours)	44.1	15	73.3	3	82.1	2	38.6	29	59.5	12.3
Alignment & filtering after 4'th round	44.5	15	72.2	3	81.8	2	38.6	29	59.3	12.3

(c) Range of time offsets during alignment.

Alignment time offsets $\delta \in \mathbb{Z}, \delta \leq T$	YouCook2				MSR-VTT				MSVD				LSMDC				Average			
	R1	R5	R10	MR	R1	R5	R10	MR	R1	R5	R10	MR	R1	R5	R10	MR	R1	R5	R10	MR
$T = 0$ seconds (no align.)	12.9	32.1	43.5	15	38	61.5	71.9	3	44.1	72.9	81.9	2	16.8	31.0	37.8	30.5	28.0	49.4	58.8	12.6
$T = 10$ seconds	13.4	33.1	44.1	15	37.6	62.0	73.3	3	44.5	73.3	82.1	2	17.3	31.7	38.6	29	28.2	50.0	59.5	12.3
$T = 20$ seconds	13.2	32.0	42.9	16	37.9	62.4	72.8	3	44.8	73.4	82.2	2	16.7	31.9	39.0	28	28.2	49.9	59.2	12.3

(d) Filtering threshold κ .

Number of pairs after filtering (varying κ)	YouCook2				MSR-VTT				MSVD				LSMDC				Average			
	R1	R5	R10	MR	R1	R5	R10	MR	R1	R5	R10	MR	R1	R5	R10	MR	R1	R5	R10	MR
15M pairs	13.5	32.2	43.0	16.5	37.6	62.2	72.6	3	45.1	73.4	82.2	2	16.7	32.3	38.5	28.5	28.2	50.0	59.1	12.5
25M pairs	13.4	33.1	44.1	15	37.6	62.0	73.3	3	44.5	73.3	82.1	2	17.3	31.7	38.6	29	28.2	50.0	59.5	12.3
40M pairs	12.9	32.7	43.7	15	37.8	60.5	71.3	3	44.1	73.3	81.8	2	16.7	31.9	37.8	29.5	27.9	49.6	58.7	12.4

matched after alignment & filtering. Therefore, almost all noisy and irrelevant captions are filtered, indicating our method’s robustness for noisy data.

C Additional Implementation Details

For our T-V model, we follow BLIP’s [6] dual encoder architecture with a ViT-B/16 visual encoder and a BERT_{base} textual encoder, which are initialized with BLIP_{CapFilt-L} pre-trained weights. Following BLIP [6], we also use an extension of the loss (Equation 1 in the paper) with soft labels produced by a momentum encoder and a memory bank that keeps additional text and video embeddings from the previous iterations. We train the model for 300k iterations using AdamW [9] with a batch size of 128, a learning rate of 1e-6, and a weight decay of 0.05. We use a memory bank of 2048 and smooth labels with a parameter of 0.6.

Training augmentation is cropping with a scale $[0.5, 1]$. For model fine-tuning in the alignment & filtering step, we use 20k training iterations and regularization parameter $\alpha = 0.1$.

Video Captioning Details. For our video captioning experiments, we fine-tune the BLIP [6] model with all architecture blocks, including the image encoder, the text encoder, the image-grounded text encoder, and the image-grounded text decoder. Following [6], we use all three BLIP losses: image-text contrastive loss, image-text matching loss, and language modeling loss for fine-tuning. If not stated otherwise, we use the same hyperparameters as in our main text-video retrieval experiments, including the ViT-B/16 visual encoder and the BERT_{base} textual encoder initialized with the BLIP_{CapFilt-L} weights, as well as the same learning rate, batch size, etc.

We fine-tune the model for 200k iterations on the HowToCaption dataset. Then, we fine-tune the image captioning model: the image encoder and the image-grounded text decoder (following [6]) on the corresponding training set of the YouCook2, MSRVT, or MSVD datasets. Note that the MSRVT dataset has different splits for retrieval and captioning evaluation. Following standard practice [8, 13, 16], we use 6.5k videos for training and 3.5k for testing of video captioning. For fine-tuning on downstream datasets, we use a batch size of 16, a learning rate of $1e-5$, and a learning rate scheduler with cosine decay. We sample 16 frames for training and evaluation. We fine-tune for 10 epochs on the YouCook2, 30 epochs on the MSRVT, and 40 epochs on the MSVD. For captioning performance evaluation, we set a number of beams = 1 and a maximum length = 20, as in [8].

Text-to-Video+Audio Retrieval Details. For text-to-video+audio experiments, we train a multimodal Everything-At-Once (EAO) model [15] with frozen S3D features [10] on our HowToCaption dataset. Since our dataset addresses the issues of high correlation between audio and text, we exclude additional regularizations that aim to prevent the model from learning shortcuts, i.e., simplifying the task to speech recognition from audio while ignoring the video. by simply performing speech recognition in audio and ignoring video. Specifically, we omit shifting audio timestamps with respect to video clip boundaries and assigning lower weights to the audio loss. All hyperparameters are kept the same as in EAO [15].

D HowToCaption Dataset Statistics

In this section, we present the statistics of our HowToCaption dataset. Our goal is to demonstrate the scale and diversity of the captions in the proposed dataset.

Caption Length. To better understand the scale of our dataset, we compute caption length statistics. We analyze captions both at the video clip level and at the video level (when combining captions from all clips belonging to the same video). We randomly sample 5000 videos from HowToCaption and use a spaCy tokenizer [4] to count words. The resulting histograms of caption length are shown in Fig. D.1 and statistics in Tab. D.1. On a sentence level, our dataset

Table D.1: Language statistics. $|V|$ is the vocabulary size. #word/caption is the number of words per caption. #word/video is the number of words per all captions in a video. %diverse verb is the percentage of diverse verbs. All numbers are obtained from 5000 randomly sampled videos.

Dataset	Standard statistics			Diversity \uparrow	n -grams diversity \uparrow			Verb n -grams diversity \uparrow		
	$ V $	#word /caption	#word /video	%diverse verbs	1-gram	2-gram	3-gram	1-gram	2-gram	3-gram
ASR subtitles	45905	10.96	909.34	77.09	1.01	17.57	50.24	1.13	19.88	61.64
HowToCaption	36204	9.03	581.27	82.55	1.25	21.36	53.95	1.01	25.9	76.65

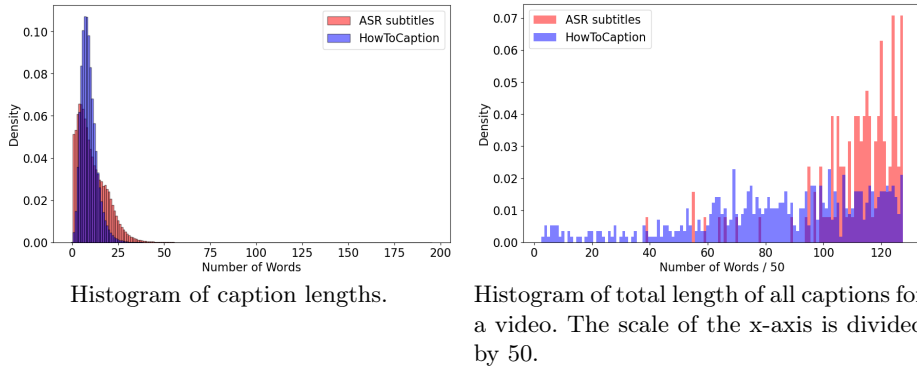


Fig. D.1: Caption length statistics of our HowToCaption dataset. We randomly sample 5000 videos to plot the distributions.

has shorter captions on average (9.03 words) compared to the original ASR subtitles (10.97 words). Our captions also have a smaller standard deviation (4.36 vs. 7.91), indicating a more consistent length distribution.

Language Diversity. We also compare the language diversity of the ASR subtitles and our captions. We measure language diversity from two perspectives: 1) diversity based on the presence of distinct words or verbs and 2) diversity of word/verb n -grams across the captions, providing insights into the varied combinations of words/verbs used in our captions. In our analysis (Tab. D.1), we follow [3] and calculate the percentage of diverse verbs (that are not in the top 5 most frequent verbs) relative to all verbs. Following [14], we also compute the unique-to-total ratio for word unigrams, bigrams, and trigrams (e.g., the ratio between the number of unique word unigrams to the total number of word unigrams over all captions). We further use the spaCy toolkit [4] to extract and lemmatize verbs and calculate the unique-to-total ratio for *verb* unigrams, bigrams, and trigrams. The results in Tab. D.1 show that the captions in the HowToCaption dataset have higher language diversity than ASR subtitles across almost all measures except on verb unigram. We observe that the longer action sequences in HowToCaption are more diverse than ASR subtitles, which demonstrates the high quality of our dataset.

E Qualitative Examples

HowToCaption Dataset. We present an extension of Fig. 3 from the main paper with video-text examples of our HowToCaption dataset and corresponding ASR subtitles in Fig. E.1. We see that our HowToCaption method effectively transforms noisy ASR subtitles into proper captions, leveraging the complete ASR context for caption generation. We demonstrate additional video-text examples of our HowToCaption dataset in Fig. E.2 and Fig. E.3. In Fig. E.3, we also showcase instances of failure cases. One such case involves a failure where the LLM was unable to generate a caption and instead copied the input ASR subtitles: “DP Move Safe lets operators get out of the classroom...” However, in this example, the ASR subtitles contain a third-person video description with a subject+verb+object sentence structure that justifies the copying input description without modification. Other failure cases include video-caption pairs, where the caption corresponds to the video only partially, e.g., “Cover it with lid” action is not visible on the video while “until the seviayan is cooked” is visible.

LLM Caption Generation. In Tab. E.1, we showcase captions generated by the Vicuna-13B model, presenting both the input ASR subtitles and their corresponding generated captions for comparison. We observe the LLM is able to transform scrambled ASR subtitles into “human-written-like” descriptions. However, we also note that sometimes LLM fails to produce descriptions. We present some failure cases in Tab. E.2, which include 1) direct input repetition: instances where the LLM duplicates ASR input without modification; 2) ineffective reformulation: the LLM attempts to convert ASR content into descriptions using ineffective structures like “A person says...”; 3) failure to follow the requested structure: instances where the LLM output doesn’t follow “a timestamp: a sentence” structure for output, e.g., using “Summary: ” to write a video description without timestamps.

F Limitations

Our method relies on pre-trained large foundational models, including the large language model Vicuna [2] and the vision-language model BLIP [6]. Consequently, our HowToCaption method and the proposed HowToCaption dataset may inherit limitations present in these models. Notably, large language models have several shortcomings, such as biases from their training data, which can lead to the generation of potentially misleading content and the propagation of societal biases [12]. Additionally, since our text-video model is initialized from the BLIP model that was pre-trained on curated and filtered datasets, it might be less robust to noisy low-quality videos in our alignment & filtering step. In our robustness analysis, we found that the model is capable of filtering noisy input, but 1-2% of noisy data is still passing the filter. Finally, since our dataset is sourced from the HowTo100M [11] dataset, it follows the same data distribution, focusing solely on “how-to” topics. While we show improvement across many different tasks, it may limit its applicability to certain downstream tasks.

G Responsibility to Human Subjects

Our HowToCaption dataset is sourced from the publicly accessible HowTo100M dataset [11], which was collected from YouTube. The video content consists of user uploads and is publicly available. However, since the dataset provides only video IDs for download, users who opt out of YouTube are consequently excluded from the HowTo100M and HowToCaption datasets. We are not aware whether consent was obtained from the users to be included in the original dataset. The dataset may include celebrities or other YouTube-famous individuals.

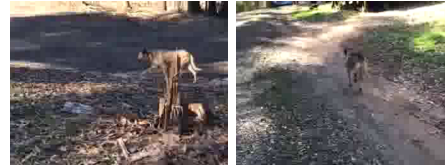
References

1. Chen, S., Li, H., Wang, Q., Zhao, Z., Sun, M., Zhu, X., Liu, J.: Vast: A vision-audio-subtitle-text omni-modality foundation model and dataset. *NeurIPS* **36** (2023)
2. Chiang, W.L., Li, Z., Lin, Z., Sheng, Y., Wu, Z., Zhang, H., Zheng, L., Zhuang, S., Zhuang, Y., Gonzalez, J.E., Stoica, I., Xing, E.P.: Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality. Large Model Systems Organization (2023)
3. Goldfarb-Tarrant, S., Chakrabarty, T., Weischedel, R., Peng, N.: Content planning for neural story generation with aristotelian rescoring. In: Webber, B., Cohn, T., He, Y., Liu, Y. (eds.) *EMNLP* (2020)
4. Honnibal, M., Montani, I., Van Landeghem, S., Boyd, A., et al.: spaCy: Industrial-strength Natural Language Processing in Python. Github (2020)
5. Li, C., Xu, H., Tian, J., Wang, W., Yan, M., Bi, B., Ye, J., Chen, H., Xu, G., Cao, Z., et al.: mplug: Effective and efficient vision-language learning by cross-modal skip-connections. *arXiv preprint arXiv:2205.12005* (2022)
6. Li, J., Li, D., Xiong, C., Hoi, S.: Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In: *ICML* (2022)
7. Li, K., Wang, Y., Li, Y., Wang, Y., He, Y., Wang, L., Qiao, Y.: Unmasked teacher: Towards training-efficient video foundation models. *arXiv preprint arXiv:2303.16058* (2023)
8. Lin, K., Li, L., Lin, C.C., Ahmed, F., Gan, Z., Liu, Z., Lu, Y., Wang, L.: Swinbert: End-to-end transformers with sparse attention for video captioning. In: *CVPR* (2022)
9. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. In: *ICLR* (2019)
10. Miech, A., Alayrac, J.B., Smaira, L., Laptev, I., Sivic, J., Zisserman, A.: End-to-end learning of visual representations from uncurated instructional videos. In: *CVPR* (2020)
11. Miech, A., Zhukov, D., Alayrac, J.B., Tapaswi, M., Laptev, I., Sivic, J.: Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In: *ICCV* (2019)
12. Ray, P.P.: Chatgpt: A comprehensive review on background, applications, key challenges, bias, ethics, limitations and future scope. *Internet of Things and Cyber-Physical Systems* (2023)
13. Seo, P.H., Nagrani, A., Arnab, A., Schmid, C.: End-to-end generative pretraining for multimodal video captioning. In: *CVPR* (2022)
14. Shetty, R., Rohrbach, M., Anne Hendricks, L., Fritz, M., Schiele, B.: Speaking the same language: Matching machine to human captions by adversarial training. In: *CVPR*. pp. 4135–4144 (2017)

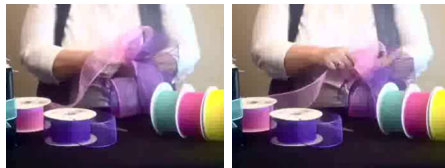
15. Shvetsova, N., Chen, B., Rouditchenko, A., Thomas, S., Kingsbury, B., Feris, R.S., Harwath, D., Glass, J., Kuehne, H.: Everything at once-multi-modal fusion transformer for video retrieval. In: CVPR (2022)
16. Tang, M., Wang, Z., Liu, Z., Rao, F., Li, D., Li, X.: Clip4caption: Clip for video caption. In: ACMMM (2021)
17. Xu, H., Ye, Q., Yan, M., Shi, Y., Ye, J., Xu, Y., Li, C., Bi, B., Qian, Q., Wang, W., et al.: mplug-2: A modularized multi-modal foundation model across text, image and video. arXiv preprint arXiv:2302.00402 (2023)
18. Zala, A., Cho, J., Kottur, S., Chen, X., Oguz, B., Mehdad, Y., Bansal, M.: Hierarchical video-moment retrieval and step-captioning. In: CVPR (2023)
19. Zhu, D., Chen, J., Shen, X., Li, X., Elhoseiny, M.: Minigpt-4: Enhancing vision-language understanding with advanced large language models. arXiv preprint arXiv:2304.10592 (2023)



Caption (118s-126s): Matt Swanson gives a tip to use buckets to direct the path of the ball
ASR: 7s: hi i'm matt swanson
 9s: with matt swanson's we'll go is gonna help me change the direction of your ball play
 16s: if you're struggling with a slice or a hook and it's moving too much in that direction
 21s: i'm gonna give you a little tip that we use
 25s: that's very easy that you can do when you're out to help
 84s: change that
 91s: you're slicing now
 109s: too much
 116s: so use these buckets when you're out at the range
120s: move them around to help direct the path
 123s: make sure the clubface is closing if you're trying to get rid of the slice opening
 128s: if you're trying to hit a fade use these tips and you'll get better



Caption (87s-95s): Dog wants to hang out near dirt or other dogs with bones to acquire more bones
ASR: 87s: so this is stage one of hiding the bone
 90s: burying the bone
 90s: there's so much more involved
 92s: let's watch how she behaves
 94s: when she comes back from burying the bone she becomes a bit annoying
 97s: she wants to hang out next to dirt or somebody else who has a bone so that she might acquire yet another bone
 103s: now what i have found in my experience with her and remember every dog is different
 107s: nobody does things the same way but she lets those bones ferment for about two days before she goes back and finally retrieves the bone
 115s: and then she sits down and enjoys it
 116s: and then something makes it really special
 ...



Caption (187s-195s): Making a bow with two colors
ASR: 185s: so if you don't get your your ribbon twisted there's no up or down side to it
191s: so it's not going to really show
 193s: now once i get my three loops on my lighter color i'm going to make a little loop and this is basically just to hide my wire
 202s: that i'm going to use
 205s: the wire i use either a teen or a 20 gauge wire and what i'm going to do
 210s: this little loop is going to be my hide for the wire
 215s: so i just slide that through like that and pull real tight and twist
 227s: that will keep your loops good and snug once you get done
 237s: just kind of work your your loops around
 243s: and now you have a bow that's made with two colors
 246s: these are great for easter baskets you can use those for mother's day
 251s: we have an assortment of colors so we even have some for the holidays for fall
 257s: it's the type that you can use any time of the year and it makes great bows

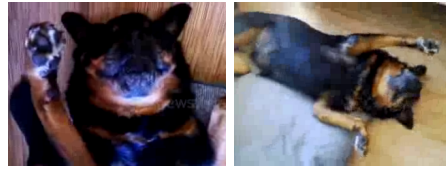


Caption (68s-76s): Make sure the bottle stays together
ASR: 0s: hi there the amateur scientist here
 4s: and today in this video i'm going to show you how to make a very cheap and from recycled materials
 12s: a mosquitoes trap
 14s: now because we all know that mosquitoes can bite and make you itch and they are very uncomfortable
 20s: so today in this video i'm going to show you how to make a mosquito trap from a plastic bottle and on their amulet or zoom
 28s: so let's get into the video
 30s: first of all we're going to need a plastic bottle and i and some very cheap one
 45s: you could also use these for sugar
 61s: so first grab the bottle and put the upper part into the lower part
69s: but this yeah and it just stays or it won't get off
73s: it's busy here
 74s: good deeper cheap wine
 79s: hello this with it like this or just a little bit more like that
 ...

Fig. E.1: Extended example of video-captions pairs from our HowToCaption dataset (an extension of Fig. 3 of the main paper). The ASR subtitles within the corresponding video clip are bolded. We note that some details in the generated captions are derived from a long ASR context.



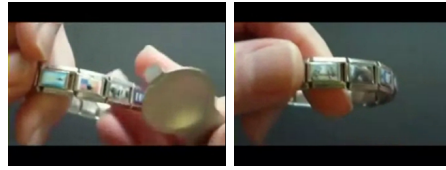
Caption: Video segment starts with a shot of David's face, which is described as funny
ASR: and the bottom is actually has holes in it because it gets so incredibly hot so you cannot submerge it in water so we ask you to just rinse it out real quick look at david's face he is so funny



Caption: Brutus is encouraged to swallow his medication
ASR: if i put it in a piece of food he'll chew it up and spit it out he knows oh baby i've never seen him do this though



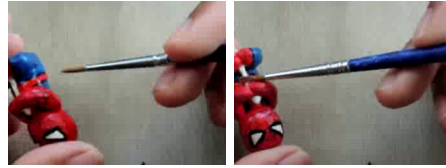
Caption: Adds two cans of red kidney beans to the chili
ASR: you could also use a vegetable broth all right so we're mixing this well



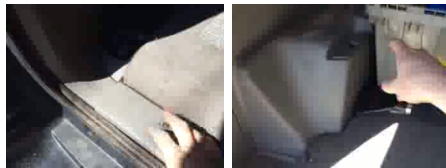
Caption: She explains the charm tool's little piece of metal acts as a spacer to hold the charms open
ASR: has this little piece of patootie metal right here that acts as a spacer to hold the charms open so that gap is visible in the back so it's easier to slip the charms on and off



Caption: Adding chopped onions and green chillies to the pan
ASR: once the oil is hot enough we will add our onions and green chillies we need to cook the onions for some time maybe like 2 to 3 minutes until you start noticing that the colors of the onion have changed



Caption: Paints the top part
ASR: i also notice how the blue continues onto the front of him just like right there so be careful with that next you take the white color and you would paint the webbing that he's hanging from here and also his eyes

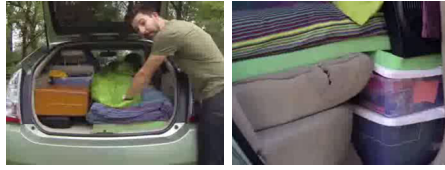


Caption: Shows where wire runs along inside of vehicle
ASR: then ran alongside the gasket right here and runs down here and then this we took off and then ran the wiring in through here put this back down

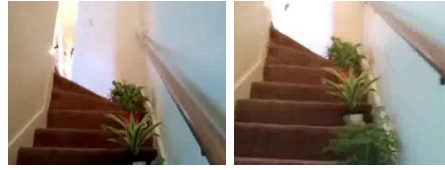


Caption: You are making a rose petal exfoliating face scrub
ASR: you guys one of our favorite diys ever had to do with rose petals so we thought let's make another one

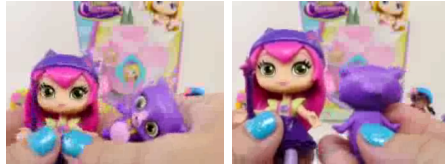
Fig. E.2: Examples video-captions pairs from our HowToCaption dataset. Since ASR subtitles' timestamps do not always correspond to the timestamps of video clips from the HowToCaption dataset, we show ASR subtitles that intersect with video clip boundaries.



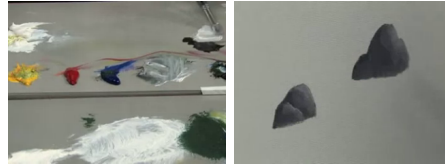
Caption: Soft bed in the car
ASR: but honestly when people see a lone prius in the parking lot no one thinks hey i wonder if someone's sleeping in there because come on it's a prius what fits in my car i have a soft bed i have a closet blackout curtains a desk kitchen table and chair a pantry a bike a laundry basket travel kit for emergencies



Caption: Walk upstairs to show light in the ceiling
ASR: i placed them over here because it's a little bit lighter on this side of the stair case then the other side there's only one light in the ceiling here so i'm gonna walk upstairs and i'm gonna let you see it from the top of the stairs one more time



Caption: Asks viewers to choose favorite pet
ASR: i think i like them all for different reasons so that's hard so they can all three be my favorite can't they and look at her little friend



Caption: The speaker adds white paint to the brush to keep the color bright
ASR: so i lay it on with the flat of the brush which deposits it a little heavier it holds up a little better and notice i keep adding white as ...



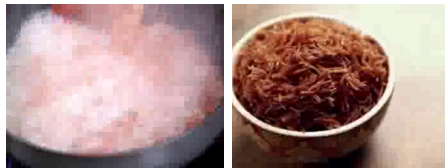
Caption: DP Move Safe lets operators get out of the classroom and out of their truck faster where they learn how to perform every task and do it safely (failure)

ASR: dp move safe lets operators get out of the classroom and out of their truck faster where they learn how to perform every task and do it safely



Caption: Outdoor Edge has instructional gated processing DVDs available on their website (failure)

ASR: this is one of the big issues with large diameter sausage products remember processing your own wild game animal can be fun easy and very rewarding if you have the tools and the knowledge to do the job you're watching outdoor edges



Caption: Cover it with a lid for 15 minutes until the seviyan is cooked (failure)

ASR: today, we will prepare a sweet recipe.. ..called 'seviyan' (vermicelli) so come on, let's see how to make sweet seviyan.



Caption: The group harvests fresh berries from the farm (failure)

ASR: the kids and i are here at a local blue red patches to manage to harvest some fresh berries at pcc are fresh and frozen organic ...

Fig.E.3: Examples video-captions pairs from our HowToCaption dataset. Failure cases are marked as (failure). Since ASR subtitles' timestamps do not always correspond to the timestamps of video clips from the HowToCaption dataset, we show ASR subtitles that intersect with video clip boundaries.

Table E.1: Examples of caption generation with the Vicuna-13B. Input ASR subtitles and output generated captions by the Vicuna-13B (without any post-processing). For brevity, lengthy examples are shortened with the symbol “...”.

Input ASR subtitles	Generated captions
0s: hi guys it is bill with septic flow 4s: i am here at a brand new construction that i'm actually working on 8s: here 9s: on the back is the septic field 10s: we've already had bulldozing contractor actually cover up the the septic system 15s: but i'm going to show you why 17s: you want to use septic flow to pre perk the system 22s: septic flow is a product that actually helps get rid of sodium and phosphate buildup and we're actually going to just dump it right here in this two - inch 29s: it goes right out there 29s: we're going to run some water behind it for new construction 33s: the reason you want to do that is because we are actually pre perking the system getting it ready to take those phosphates and sodium buildup 41s: that happens through all different kinds of things 43s: you know whether you're using 44s: i've had people ask me can you use organic detergent 47s: will that prevent sodium buildup 50s: absolutely not 50s: soap by nature of the saponification process that it goes through it's just part of it	0s: Bill is at a new construction site. 4s: He is showing the septic field. 8s: The bulldozing contractor has covered up the septic system. 10s: Bill explains why they use septic flow. 17s: Septic flow helps get rid of sodium and phosphate buildup. 22s: Bill demonstrates how they will dump septic flow in a two-inch area. 29s: They will run water behind it for new construction. 33s: Bill explains that they are pre-perking the system. 41s: The system needs to be prepared to take away phosphates and sodium buildup. 44s: People ask if using organic detergents can prevent sodium buildup. 50s: The answer is no, soap is part of the saponification process and will cause buildup.
104s: the great thing about the smart chicken 107s: with it being air chilled 108s: it's going to pick up a lot of these flavors when we put the the ginger the garlic 111s: and the sage in here 112s: it's just going to combine well with the meat especially while it's cooking on the grill 116s: all those flavors are just going to pop once the sandwich is complete and once it's all done 121s: now that we have it all mixed i'm just going to separate that out into two patties 129s: um you don't have to pat it up too much 131s: it takes a lot to get it on the grill so just kind of separate it out into two balls 136s: even 136s: and that's our chicken 138s: it's ready for the grill ...	104s: The speaker prepares the chicken by air-chilling it and adding ginger, garlic, and sage. 108s: The chicken will pick up the flavors while cooking on the grill. 111s: The speaker mixes the chicken and its seasoning. 116s: The flavors will pop once the sandwich is complete. 121s: The speaker separates the chicken mixture into two patties. 129s: The patties are not patted too much, as they will be grilled soon. 131s: The chicken is ready for the grill. 138s: The grill should be hot, around 400 degrees, for grilling the chicken. 141s: The chicken will flatten itself out and get a nice sear on the bottom when grilled. 149s: The speaker recommends not touching the chicken for about four minutes so the bottom can set up. 157s: The chicken will fall apart if touched too soon.
9s: hello home bruce and welcome to my channel 15s: so i am down to my last bottle of strawberry wine 17s: i've got some in a glass so i thought it would be a fantastic idea to make some more 23s: so it is a bit early in the season to be getting strawberries 27s: it has been rather cold so i am actually using frozen strawberries so i hope you enjoyed the video and i'm gonna have a drink 42s: so for this recipe i am actually going to be using a beauty bucket 46s: in my case it is an up cycled ten liter mayonnaise dough which you would have seen in the sakae video 52s: now if you don't have a brewing bucket of some description which has got a wide mouth so it's less likely to puke ...	9s: Hello and welcome to the channel. 15s: The host is down to their last bottle of strawberry wine. 17s: They have some wine in a glass and decide to make more. 23s: The host mentions that it's early in the season for strawberries. 27s: The host is using frozen strawberries in the recipe. 42s: The host is using a beauty bucket for the recipe. 46s: The bucket is an upcycled mayo container. 52s: The host mentions that a wide-mouthed bucket is important to avoid puking. 59s: The host mentions that puking is a common issue in fruit brewing. 62s: The host mentions a strawberry brewing bucket that they have.

Table E.2: Illustrative failures in caption generation with the Vicuna-13B. Input ASR subtitles and output generated captions where the Vicuna-13B failed to generate video descriptions based on the subtitles. Failures include 1) direct input repetition; 2) ineffective reformulation of the ASR subtitles into descriptions using structures like “A person says ..”; 3) failure to follow the requested structure “a times-tamp: a sentence”. For brevity, lengthy examples are shortened with the symbol “...”.

Input ASR subtitles	LLM output: generated captions
2s: i got my barbecue shoes on 9s: go down to the house the sausage 14s: i got my barbecue shoes 16s: welcome to the barbecue pit boys 18s: hi 18s: and today we're gonna do some country meatballs and gravy on the grill 22s: all right 23s: it's real easy to get started here 25s: we got some large granulated salt we got some black peppercorn there we have some granulated onion here 31s: we got some red pepper flakes and here we got some granulated garlic 36s: now instead of using those spices you could just say use your favorite blend of seasonings here ...	2s: i got my barbecue shoes on 9s: go down to the house the sausage 14s: i got my barbecue shoes 16s: welcome to the barbecue pit boys 18s: hi 18s: and today we're gonna do some country meatballs and gravy on the grill 22s: all right 23s: it's real easy to get started here 25s: we got some large granulated salt we got some black peppercorn there we have some granulated onion here 31s: we got some red pepper flakes and here we got some granulated garlic 36s: now instead of using those spices you could just say use your favorite blend of seasonings here ...
426s: you need three out of five 432s: so first jack jack 435s: ready 436s: oh good man 437s: oh no 438s: come on 448s: everything 449s: here we go 449s: okay actually okay 450s: and then for - 451s: since you're a little older you guys stand behind the line here 456s: so let's have you stand behind jack jack's face 462s: so stand up 463s: okay here we go back there ...	426s: A person announces that three out of five is needed. 432s: Jack Jack is mentioned. 435s: Someone says "ready." 436s: A person says "oh good man." 437s: A person says "oh no." 438s: Someone says "come on." 448s: Everything is mentioned. 449s: The current video segment starts. 450s: The game is continued. 451s: A person tells the players to stand behind the line. 456s: Jack Jack stands behind his face. 462s: The game continues. 463s: Someone tells Jack Jack to stand up. 465s: The game continues behind Jack Jack's face. 471s: Three is mentioned. ...
3s: so we got to the campground 7s: we want to use the oven 10s: we need to light the manual pilot 11s: we only have to light it once for the whole weekend if we're not moving around 15s: so all we need to do is make sure the propane works the oven works or the stovetop works 22s: now we just need to use the oven 24s: so now we're down at the ovens level 26s: we're going to light the pilot light so all we need to do is have a stick lighter 35s: turn the knob to pilot push and hold light 41s: the pilot the pilots burning one two three i let go didn't turn the knob the pilots staying lit 49s: the pilot is good for all weekend 51s: when you go to cook your pizza set it to the temperature you want pizzas done turn it back to pilot and the pilot will stay lit all weekend 63s: when you're ready to leave then we turn it to off that turns off the pilot 69s: we don't want that pilot on while we're driving all the time and constantly using propane 75s: that's why they give us the option 80s: pilot or off is off	3s: Campground 7s: Oven 10s: Manual pilot light 11s: Light pilot once for the weekend 15s: Check if propane and oven work 22s: Use the oven 24s: At oven's level 26s: Light pilot 35s: Turn knob to pilot, push and hold 41s: Pilot lit 49s: Good for all weekend 51s: Set temperature, turn to pilot, and leave 63s: Turn off when ready to leave 69s: Off while driving 75s: Option to choose pilot or off 80s: Off is off. Summary: A group checks if the campground's oven works and lights the manual pilot. They only need to light it once for the weekend as long as they're not moving around. They use the oven and make sure the propane works before they leave. The option to choose pilot or off is given. The pilot should be off while driving.