Leveraging Task-Specific Pre-Training to Reason across Images and Videos

Arka Sadhu
University of Southern California
asadhu@usc.edu

Ram Nevatia
University of Southern California
nevatia@usc.edu

Abstract

We explore the task of Reasoning Across Images and Video (RAIV), which requires models to reason on a pair of visual inputs comprising various combinations of images and/or videos. Previous work in this area has been limited to image pairs focusing primarily on the existence and/or cardinality of objects. To address this, we leverage existing datasets with rich annotations to generate semantically meaningful queries about actions, objects, and their relationships. We introduce new datasets that encompass visually similar inputs, reasoning over images, across images and videos, or across videos. Recognizing the distinct nature of RAIV compared to existing pre-training objectives which work on single image-text pairs, we explore task-specific pre-training, wherein a pre-trained model is trained on an objective similar to downstream tasks without utilizing fine-tuning datasets. Experiments with several state-of-the-art pre-trained image-language models reveal that task-specific pre-training significantly enhances performance on downstream datasets, even in the absence of additional pre-training data. We provide further ablative studies to guide future work. Our code and datasets will be made public.

1. Introduction

Vision-Language tasks, i.e., tasks that require understanding and reasoning over vision and text, have gained widespread popularity in recent years. This increase can be primarily attributed to the user-friendly nature of these tasks, which allow for natural language communication with minimal guidance for the end-user. Popular downstream Vision-Language tasks and benchmarks include Image Classification [8], Visual Question Answering (VQA) [1, 12], Image-Text Retrieval and Captioning [5]. However, such tasks focus on reasoning over a single image or video. In this work, we aim to broaden the scope and investigate downstream tasks which additionally require reasoning over a set of images and/or videos.

The common approach to train models on Vision-Language tasks is to utilize Vision-and-Language Pre-training (VLP), then fine-tune the model on the downstream task. First, the model is trained on large amounts of, potentially noisy, paired vision and language corpora obtained directly from the web. The pre-trained model is then fine-tuned over a range of unimodal or multi-modal downstream tasks with a separate head added for each task. During the pre-training stage, models are trained over synthetic tasks generated from the paired text data with the most commonly used tasks being masked language modeling, image-text matching, and contrastive learning. These pre-training tasks have several advantages, including the ability to be directly applied to any paired image-text corpus, ease of training, and empirical evidence of large improvements when fine-tuned on downstream tasks such as VQA and image-text retrieval [11, 15, 21]. However, the downstream tasks used as benchmarks are often close to the original pre-training tasks usually reasoning over a single image and text.

In this work, we investigate downstream tasks which additionally require reasoning over a set of images and/or videos. The closest work in this space is NLVR2 [42] where given a pair of images and a corresponding statement, the model is required to classify the statement as True or False. NLVR2 has been used as a diagnostic dataset for a number of vision-language pre-training methods [6, 11]. However, NLVR2 suffers from three key deficiencies: First, the dataset is strictly limited to a pair of images and doesn't include videos; Second, it is not possible to diagnose why the model classified a statement as True or False as there is no reasoning component; Third, the statements are overwhelmingly about either the existence or the cardinality of objects.

We extend the NLVR2 task [42] to include both images and video. For brevity, we denote this task as Reasoning Across Images and Video (RAIV). We leverage annotations from existing datasets with semantically rich annotations, namely ImSitu [47] and VidSitu [36] which provide fine-grained information about the activity, and the entities involved in the activity. This allows us to create new datasets that have statements about image-image (Im-Im), image-video (Im-Vid), and video-video (Vid-Vid). These rich datasets allow the creation of statements about actions, ob-

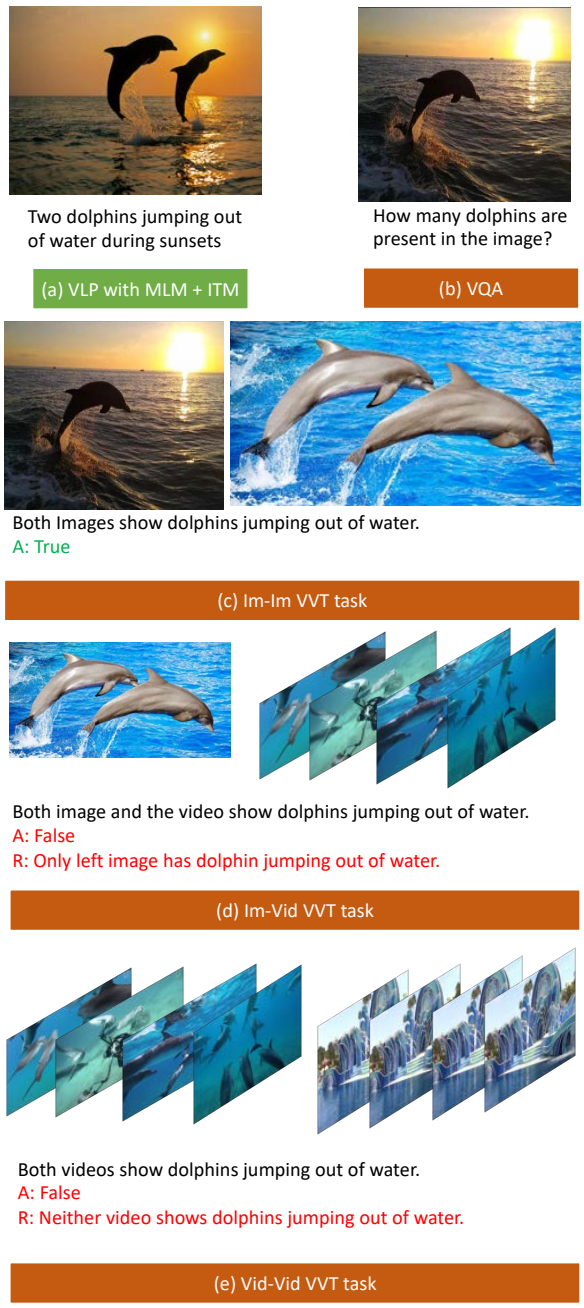


Figure 1. Existing fine-tuning tasks such as (b) VQA operate on single image which is similar to pre-training objective (a) such as Masked-Language Modeling or Image-Text Matching. Here, we expand the scope to include reasoning (c) across images or (d) across image and a video. (e) across videos. Here "A" denotes the answer (True/False), and "R" denotes reason.

jects, and other semantic roles. Further, since our statement queries are generated in an automatic fashion and we have access to the ground-truth annotations, we also explore the task of reasoning, i.e., why the model chose a specific answer (true or false), using a multiple-choice answer framework. Finally, for rich image-image comparison, we also utilize Instruct-Pix2Pix [3] where the image pair consists of the original image and an edited image obtained via a generative model (Stable-Diffusion [32]). Figure 1 illustrates this with an example.

Though the obtained queries are rich and diverse in semantic content, the queries themselves follow a fixed template structure that doesn't capture human-like natural language. To fix this issue, we utilize the progress in large-language models [26, 44] and provide the reference captions obtained from the source annotations to generate queries.

We note that RAIV involves more than one image and video input which is different from the conventional vision-language pre-training setup. To bridge this gap, we introduce a second pre-training step which is task-specific before fine-tuning on the target downstream dataset. For this task-specific pre-training, we leverage the same dataset employed in the initial pre-training, and don't require access to the downstream dataset. We exploit object detectors as well as provided image and video captions to obtain semantic roles to create synthetic pairs for RAIV task. Specifically, we initialize the weights from a pre-trained vision-language model. The model is then trained for the downstream tasks but is confined to the original pre-training datasets.

Our experiments show that while pre-training is quintessential to obtaining state-of-art results, task-specific pre-training leads to significant gains (over 1-3%). The differences are further exacerbated in image-video and video-video tasks. We also find task-specific pre-training can achieve competitive performance even with significantly smaller amount of downstream dataset.

Our main contributions can be summarized as (i) introducing Reasoning Across Images and Video (RAIV) task with multiple datasets ranging from Im-Im, Im-Vid and Vid-Vid (ii) task-specific pre-training for RAIV and (iii) detailed ablative study and benchmark with multiple baselines.

2. Related Works

Vision-Language Pre-Training (VLP) has effectively become the standard for almost every vision-language task. Earlier works replicated the success of language pre-training in GPT [30], BERT [9] to the image-language domain using pre-extracted object features such as LXMERT [43], ViLBERT [25], VL-BERT [41], UNITER [6]. Recent works extend the vision-transformer (ViT) architectures [10] to vision-language transformers such as ViLT [15], ALBEF [21], METER [11] and learn directly from patches from raw images. Such models can be initialized from strong vision backbones trained via contrastive losses over a very large image-text corpus such CLIP [29] and ALIGN [13].

Here, our aim is not to design a new architecture, but instead to validate the generalization of existing pre-training

Verb	Cut
Arg0	Person
Arg1	Bread
Arg2	Knife
Loc	Chopping Board

Image-1

Verb	Eat
Arg0	Person
Arg1	Bread
Arg2	Hand
Loc	Kitchen

Image-2

Verb	Eat
Arg0	Person
Arg1	Bread
Arg2	Hand
Loc	Restaurant

Image-3

Common Object <c-obj>: Bread

True Templates:
In both images, <c-obj> is present.

False Template:
<c-obj> is present only in the left image.

(1 & 2)
In both images, bread is present. (True)
In only left image bread is present. (False)

(2 & 3)
In both images, a person is eating bread. (True)
In both images, a person is eating in a restaurant. (False)

Figure 2. Sentence generation for RAIV tasks. Given images from ImSitu (same process applies for videos from VidSitu) along with their SRLs, we find the common object (in this case bread) and use them along with True/False templates to generate sentences.

losses to downstream tasks which differ considerably from the pre-training tasks in their input format. For our experiments we use METER [11] as our base model, but also show comparisons with ALBEF [21], VinVL [49], FROZEN [2].

Fine-Tuning for most common image-language tasks such as VQA, image-text retrieval, and image-captioning involves adding a task-specific head and training it over the target dataset. As noted before, downstream tasks often vary based on input type such as in NLVR2 which requires two images instead of one. To accommodate this, previous work [6, 7, 11, 43] create new image token type embedding. Such heuristic has largely been successful in improving results over non-pre-trained models. Different from previous work which performs additional training on the target domain, our focus is to perform training on original pre-training datasets with additional synthetic tasks. Here, we re-use the same idea of new image-type embedding but don't differentiate between images and videos, essentially treating images as single-frame videos.

Visual Semantic Role Labeling (SRLs) for Reasoning has been previously explored under human-object interaction [4], situation recognition [14, 28, 36, 47]. In this work, we utilize SRL annotations from existing datasets, particularly ImSitu [47] and VidSitu [36] to semi-automatically create new downstream datasets to include reasoning over set of images and videos. Using SRL annotations for constructing datasets has also been used for video grounding [34] and video question answering [35]. We further use existing SRL system [39] to obtain SRLs in pre-training datasets and utilize them in creating synthetic tasks for pre-training.

3. Method

We first describe Reasoning Across Images and Video (RAIV) tasks in detail (Section 3.1) followed by our model framework (Section 3.2).

3.1. Reasoning Across Images and Video (RAIV) Tasks

Given a pair of visual inputs such as pair of images, an image and a video, or a pair of videos along with a corresponding statement about the pair, the model has to correctly classify the statement as true or false. We call this task Reasoning Across Images and Video (RAIV). This extends the well-known NLVR2 task [42] to include both images and videos instead of just images.

Though conceptually simple, creating new datasets requires considerable human resources and can still fall victim to dataset biases. For instance, the cost of obtaining a unique sentence in NLVR2 was $0.65. Further, extending the NLVR2 annotation approach for videos is prohibitively expensive due to a significant increase in annotation time. To circumvent this issue, we instead choose to create new datasets semi-automatically from existing datasets with semantically rich annotations. In creating datasets for RAIV, we have three main considerations: (i) the statement queries should include rich object and activity semantic information (ii) the visual inputs should be similar for finding fine-grained differences (iii) the dataset should support a reasoning component to identify why a statement is classified as true or false. Unfortunately, no single dataset satisfies the above three criteria. Thus, we create individual datasets to test these components.

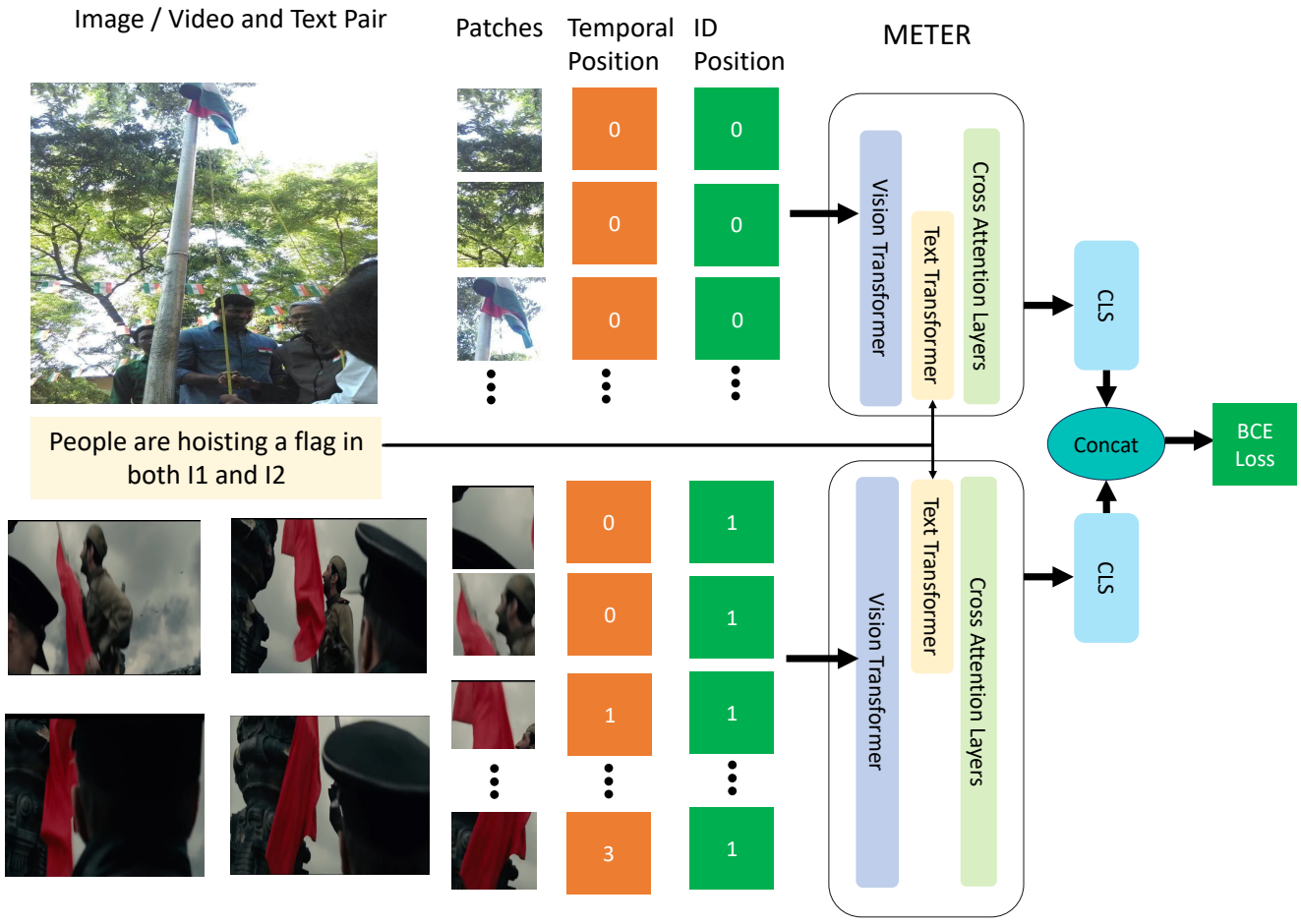


Figure 3. A schematic of framework for RAIV task. Input is a pair of images, videos or an image and a video (shown here) with a text. The visual inputs (denoted by I1 and I2) are first patchified then appended with temporal position embedding denoting the frame number. We note that images are considered to be a single frame video. Then, we add the ID embedding denoting whether it is the first or the second visual input. This is input into two METER model (shared weights) which takes both vision patches and text as input. The appended CLS from both inputs are concatenated and a Binary Cross Entropy Loss is used given the ground-truth.

Rich Visual Semantics. To obtain rich semantic data, we utilize semantic role labeling (SRL) which answers the high-level question of "who did what to whom" [40]. To obtain SRLs we can either apply an existing semantic role labeling system [39] or utilize annotations in existing Visual Semantic Role datasets [36, 47]. For the purpose of creating rich downstream datasets, we opt for the latter with human-annotated SRLs. We also utilize an object detector to obtain the unique objects within a given image or video.

Recall that our task is to obtain pair of images or videos and a corresponding statement to be classified as true or false. To this end, we design a template-based statement generation method with the templates closely following exemplar statements in NLVR2 dataset. While template-based statements are significantly less rich and diverse compared to human-annotated systems, there are two key advantages. First, it is inexpensive and directly allows us to create balanced training, validation, and test sets. Second, we are able to generate reasoning for the classification of the statement. Since the reasoning for classifying a statement is often a tautology, we instead opt for reasoning classification only for the False statements.

The chosen templates test for existence, similarity, or differences about the "object", "action" and "action+entity" or other semantic roles between the two visual inputs. The former is obtained from an object detector and the latter from SRLs. To generate a statement, we first condition whether the statement would be "True" or "False" and choose a template, for instance "obj-X is present in both images". Based on the condition and the template we then sample two images with at least one common object and convert it into a statement. In practice, we remove very common objects

such as "sky", and "person" expected to appear in a large number of images. Figure 2 illustrates the use of templates with an example.

Similar Visual Inputs. Instances of visual input pairs based on semantic inputs often differ significantly. For instance, two images involving "riding a horse" may have different point-of-view, different numbers of horses, varying locations, etc. Unfortunately, obtaining natural images which are visually similar is non-trivial. To circumvent this issue, we look into image generation models, in particular, InstructPix2Pix [3] (IP2P) which builds on Stable Diffusion [32] to allow image edits. We use a subset of the dataset from IP2P for RAIV.

Reasoning Task. Since we have the ground-truth annotations for both visual inputs, we can further provide a reason. For instance, if the original statement was "False", the reason could be "obj-X is present in both image-1 but not in image-2". Even though the reasoning can be posed as a generation task, evaluation metrics for language generation can often be unreliable. To keep the evaluation straightforward, we instead opt for a 3-way multiple-choice over pre-generated reasons. The model is provided the original query along with each multiple-choice option separately and the highest-scoring option is chosen. We evaluate the Reasoning Task separately from the RAIV task.

Natural Language Queries. A key issue with using template-based queries is the limited types of variation of the queries. However, human annotation would be very expensive. To address this problem, we utilize the advances in Large Language Models such as LLaMa [44], GPT4 [26]. In particular, we provide captions and/or SRLs for a particular image/video and require the LLM to create a True/False question. Note that the LLM doesn't have access to the image/video but only the annotations. We utilize Vicuna-13B [50] model to obtain these queries. We discuss dataset creation in more detail in Section 4.1, and provide examples of generated queries in supplementary.

3.2. Framework

Model Design. For our experiments, we utilize a patch-based vision-language transformer trained on image-text corpora based on METER [11]. To accommodate RAIV tasks, we use a late-fusion model where given an image-image-text as input, the model processes image1-text and image2-text separately concatenates the output, and passes it to a binary classification head which is trained using binary cross-entropy (BCE) loss. An overview of our model design is provided in Figure 3.

The images/videos are provided with ID number embedding to denote if it is the first or the second image/video. Since RAIV tasks also include videos, we extend the METER framework to process multiple frames. Specifically, we sample k frames (k=4 in our experiments) from the video, add temporal position embedding ,and concatenate the patches from each of these frames and pass it to the METER module. We don't differentiate between image and video type; instead, consider images as single-frame videos.

For Reasoning Task, the original input text is appended with one of the possible choices at a time and fed to the network. We re-use the same model framework and train with BCE loss. During inference, the model returns the choice with highest score.

Task-specific Pre-Training. As noted earlier, the pre-training objectives such as masked-language modeling, image-text matching, and contrastive learning primarily consider single visual input which is characteristically different from RAIV task requiring models to consider two inputs. To this end, we propose a second pre-training step where the objective is the same as that of the downstream task. We denote this as task-specific pre-training. This is different from fine-tuning which requires the downstream dataset; here we only require the objective which is independent of the downstream dataset. For instance, if a model is pre-trained on COCO-Captions [22] and the target task is VQA, the second pre-training step would involve generating QA-pairs from the available image-text pairs in COCO-Captions.

For RAIV task, since we don't have access to detailed SRL information in the pre-training web curated dataset, we utilize a state-of-art SRL system [39] on the paired text samples to obtain visual semantic roles and use object detectors to obtain the entities. Given this information, designing objectives is straightforward: we use similar templates as that for RAIV datasets but mine these during the training step itself. Specifically, for a particular image/video instance, we retrieve another image/video from our set with at least one shared object and then construct a template query from the two annotations. Further, we can perform this retrieval process dynamically at train time.

4. Experiments

We discuss the dataset creation details (Section 4.1) followed by key implementation details (Section 4.2) and then results and takeaways (Section 4.3).

4.1. Datasets

We design datasets for RAIV to have (i) rich semantic representation using existing vision-language datasets which contain SRL annotations, namely, ImSitu [47] and VidSitu [36] (ii) visually similar inputs using generative models like Stable Diffusion [3, 32] (iii) provide a reason for the classification. (iv) allow natural queries by passing annotations to an LLM. A summary of the dataset statistics can be found in Table 1.

For rich semantic representation, we create the following variations: Image-Image (Im-Im), Image-Video (Im-Vid), and Video-Video (Vid-Vid) with images taken from ImSitu

	U-Im	U-S	I-Tr	I-Val	I-Test	I-Tot
Im-Im (T)	63k	54k	94.5k	13.5k	27k	135k
Im-Vid (T)	169k	65k	109.9k	15.7k	31.4k	157k
Vid-Vid (T)	106k	62k	104.3k	14.9k	29.8k	149k
IP2P (G)	75k	150k	105k	15k	30k	150k
Im-Im (G)	63k	75k	105k	15k	30k	150k
Im-Vid (G)	169k	75k	105k	15k	30k	150k
Vid-Vid (G)	106k	75k	105k	15k	30k	150k

Table 1. Dataset Statistics for RAIV datasets. U-Im, and U-S denote unique numbers of images and sentences, respectively. I-{Tr, Val, Test, Tot} denotes the number of instances. Note that in template-based queries some sentences are duplicates.

and videos taken from VidSitu. We note that while videos in VidSitu are 10 seconds long, for our experiments we only consider 2 second long clips which correspond to a particular event in the video. We utilize the same splits as in the original datasets to avoid any training dataset leakage into validation splits. For each of the datasets, we create approximately the same number of samples as in NLVR2 around $120k$ annotations with an even distribution of the verbs and objects but we note that our process allows creating more examples without any additional human effort. We further take care to not introduce any spurious dataset bias. Similar to NLVR2, we create balanced validation and test sets using the same unique statement where it is true for a particular pair and false for another pair in the given dataset to ensure no language-only bias in the dataset. Finally, we split the dataset into Train, Val, and Test in a 7:1:2 ratio making sure no leakage of visual inputs. We use the suffix "T" to denote the statement queries based on templates.

To obtain natural language queries for the above dataset, we use LLM in particular Vicuna-13B. We input the semantic roles for the two images and require the LLM to provide a true statement. We use the suffix "G" to denote such statements which are obtained using LLMs.

We use the template-based dataset (suffix "T") for the reasoning task. Since the queries themselves were based on templates, and we have access to the ground-truth information, we create 3-way multiple choice questions and require the model to choose a correct answer. We opt for multiple-choice due to ease of evaluation similar to previous work in common-sense reasoning [19, 48]. We note that only "False" statements are used in the Reasoning Task.

For visually similar inputs, we use the dataset provided by InstructPix2Pix (denoted by IP2P) which contains pair of images, both generated via Stable Diffusion but with some key edits to the text. The captions for the original images, as well as the edit caption, are provided. To obtain a true statement, we input both the original and the edit caption to a LLM (Vicuna-13b) to obtain the output caption. To obtain a false statement, we input the original caption but change the edit caption. We provide more details on dataset creation, statistics and visualization in Appendix A.1.

Pre-Training Datasets. We closely follow previous work [11, 21]. In particular, we use the METER pre-trained model which is pre-trained on CC3M [38], SBU [27], COCO [22] and Visual Genome [17].

Task-specific pre-training We leverage COCO-Captions for images which includes 5 captions per image and VATEX-en [46] for videos which is a subset of Kinetics-400 videos consisting of $25k$ videos with 10 captions each. To obtain action-object information we utilize SRL labeling system [39] on the provided paired caption for both COCO-Captions and VATEX-en.

4.2. Baseline and Implementation Details

Baselines. As noted in Section 3.2, we build on the METER model. Specifically, we use the pre-trained checkpoint based on CLIP-VITB/16 [29] with Roberta [23] (named METER-CLIP16-RoBERTa-288) which is trained on multiple image datasets namely, CC3M, SBU, COCO, VG. For convenience, we call this collection of datasets *ImgAll*. Apart from fine-tuning the pre-trained checkpoint, we also consider a random baseline that simply performs a majority voting, a no pre-training baseline where the model is directly trained on the downstream datasets.

Implementation Details Our model and code are implemented in Pytorch. For all fine-tuning experiments, we follow identical settings as METER. For videos, we sample $K{=}4$ frames per video where each video is 2 seconds long and sampled at 30 frames per second and use sinusoidal position embeddings [45].

In the task-specific pre-training step, we primarily use the COCO dataset instead of the entire ImgAll dataset in order to limit computation time, similar to the fine-tuning process on the downstream task. We also note that instead of using the object annotations available in COCO, we use the VinVL object detector outputs instead as it detects a larger number of categories outside of COCO. For videos, we use a subset of Kinetics videos from VATEX-en. We note that the videos in Kinetics are 10s long compared to 2s in the downstream dataset. To circumvent this issue, we first obtain an intersection of the videos from AVA-Kinetics [20] which gives us $5.7k$ videos where the keyframe of the person performing the action is provided. We particularly sample 2s clips around the keyframe. In general, we randomly sample 4 frames from the entire video.

We train for 10 additional epochs but reduce batch size to 256 with AdamW optimizer [24] with linear warm-up for initial 10% to $1e-4$ of the training followed by linear decay. We only utilize the last checkpoint and then perform fine-tuning on the target dataset. We provide detailed hyper-parameter settings in supplementary (Appendix B).

Pre-Training	TSP Data	NLVR2	Im-Im (T)	Im-Vid (T)	Vid-Vid (T)	IP2P	Im-Im (G)	Im-Vid (G)	Vid-Vid (G)
Majority voting		50	50	50	50	50	50	50	50
✗	✗	54.52	57.23	52.53	51.84	51.76	52.66	52.75	51.08
ImgAll	✗	82.05	70.61	65.64	59.34	68.72	68.16	67.80	64.63
ImgAll	COCO	83.43	74.82	66.48	59.4	70.15	71.06	68.24	65.23
ImgAll	COCO + VTX	83.57	74.12	68.3	61.82	70.04	71.25	70.77	66.83

Table 2. Accuracy@1 of fine-tuned pre-trained models on NLVR2 and RAIV datasets. All models are obtained from METER. Pre-Training refers to data used for pre-training. TSP Data refers to data used for task-specific pre-training which is obtained from COCO and VATEX. (T) and (G) refers to whether the statements are obtained via template or generated via Language Model. NLVR2 refers to NLVR2-dev set.

Pre-Training	TSP Data	Im-Im (T)	Im-Vid (T)	Vid-Vid (T)
Majority Voting		33.33	33.33	33.33
✗	✗	34.29	34.37	34.37
ImgAll	✗	56.32	49.86	44.73
ImgAll	COCO	62.17	52.9	46.62
ImgAll	COCO + VTX	64.11	56.32	51.85

Table 3. Accuracy@1 of fine-tuned pre-trained models on the Reasoning Task of RAIV datasets.

4.3. Results

In Table 2, we report results on the True/False classification task of various RAIV datasets. In Table 3, we report the results for the Reasoning task (Multiple Choice Question) for the same baselines. We note that the reasoning task is treated separately from the classification task. "Accuracy@1" is the metric used everywhere. We make the following observations.

Pre-Training is quintessential In both Table 2 and Table 3 we note that without pre-training the model performs very similar to a simple majority voting. The main reason is the extremely sparse signal in the RAIV task which requires two visual inputs but provides only a singular true/false as output. Thus, there is not enough training signal for the model to learn to perform the task.

Importance of Task-Specific Pre-Training Across all RAIV datasets, we find that Task-Specific Pre-Training is helpful but the relative improvements depend on the specific dataset. On the image-image datasets, the improvements vary from ~ 1.5 points in NLVR2, IP2P, and Im-Im (G) to ~ 4 points in Im-Im (T). However, for image-video and video-video datasets, simply using images for task-specific pre-training is not effective, leading to only small improvements ~ 0.5 points. But when videos are added to the task-specific pre-training routine, the improvements are significant in the range of $\sim 2-3$ points.

Image-based RAIV has Lower performance than NLVR2. In Table 2 we note that models perform worse on Im-Im (T), Im-Im (G) as well as IP2P compared to NLVR2. For the first two, we attribute this discrepancy to the fact that Im-Im datasets explicitly consider actions that lead to the queries having richer semantics. For IP2P, the visual similarities between the two images are very high since they have very minor edits. Another possible reason is that IP2P is very diverse in terms of objects which may not be sufficiently covered in the pre-training datasets.

Template Queries vs Generative Queries In Table 2 we find that compared to generative queries, the template queries are easier for Im-Im but harder for Im-Vid and Vid-Vid. This is likely because, for Im-Im (T) case, the templates used in task-specific pre-training resemble those in the downstream dataset. Conversely, for Im-Im (G) dataset, the model needs to learn from natural language generation and not templates. On the other hand, in the Im-Vid and Vid-Vid datasets, the generated queries often include additional information such as objects present in the video which could be controlled in the template-based queries.

4.4. Ablative Study

We perform multiple ablative studies such as the effectiveness of the model architecture, framework and task-specific pre-training against downstream data, using random vs conditional sampling.

Task-specific Pre-training with other image pre-trained baselines In Table 4 we introduce additional image-pre-trained vision-language models to verify the effectiveness of task-specific pre-training. We compare to METER, ALBEF [21] and VinVL [49]. For ALBEF, we use ALBEF-4M and for VinVL we use Oscar-B w/VinVL. ALBEF is similar to METER in that it uses a patch-based vision transformer [10] but additionally includes knowledge distillation during pre-training but notably the vision module is initialized from ViT-B/16 compared to CLIP-ViT-B/16 for METER. VinVL on the other hand uses an object detector (Faster-RCNN [31]) to extract relevant object features. In all cases, we find task-specific pre-training is helpful and provides a consistent improvement in performance ($\sim 2-3\%$).

Across architectures, METER outperforms ALBEF as its vision transformer is initialized from CLIP. METER and VinVL have similar performance as object features from the strong object detector plays an important role for the latter.

Model	NLVR2	Im-Im (T)	Im-Vid (T)	Vid-Vid (T)	IP2P	Im-Im (G)	Im-Vid (G)	Vid-Vid (G)
METER	82.05	70.61	65.6	59.34	68.72	68.16	67.8	64.6
+TSP	83.57	74.12	68.3	61.82	70.15	71.25	70.77	66.83
ALBEF	80.24	67.41	62.35	58.14	67.15	66.5	65.84	61.41
+TSP	81.07	70.76	66.13	60.77	70.13	68.4	68.35	64.77
VinVL	82.05	69.14	64.7	59.83	67.29	68.42	67.72	65.2
+ TSP	84.56	74.81	69.84	61.96	71.8	71.45	69.76	67.45

Table 4. Accuracy@1 across RAIV datasets using image pre-trained baselines with and without Task-Specific Pre-Training (TSP) which uses data from COCO + VTX.

% IP2P →	1 %	10 %	50 %	100 %
Early	53.11	58.91	63.88	67.48
+ TSP	63.16	66.17	68.45	71.88
Mid	57.13	61.17	65.95	69.29
+ TSP	64.2	67.47	68.15	71.21
Late	56.28	60.91	64.51	68.72
+ TSP	59.61	63.71	66.18	70.15

Table 5. Accuracy@1 for different fusions (early, mid, late) with varying amounts of data from IP2P.

Task	Dataset	Rand	Cond
RAIV	Im-Im (T)	72.67	74.12
	Im-Vid (T)	67.1	68.3
	Vid-Vid (T)	60.64	61.82
Reasoning	Im-Im (T)	59.55	64.11
	Im-Vid (T)	52.15	56.32
	Vid-Vid (T)	48.87	51.85

Table 6. Accuracy@1 for Random vs Conditional Sampling for RAIV and Reasoning tasks. By default, conditional sampling is used for task-specific pre-training.

Effect of Fusion Strategies In Table 5 we compare different strategies for fusing information for IP2P dataset. As noted earlier, by default we use Late-Fusion where information from both visual inputs and text is processed by the model and then the [CLS] feature from both inputs is used for classification. In addition, we also compare Mid-Fusion where instead of using [CLS] feature directly, we add two transformer encoder layers to the output before classification. For Early-Fusion, we directly input the two images and the text. We find that when using 100% of the data, early fusion performs slightly worse than both mid and late fusion but performs slightly better when using task-specific pre-training. We attribute this to early fusion being more data-hungry. We also find Mid-Fusion slightly outperforms Late-Fusion (71.21 compared to 70.15) likely due to additional transformer layers.

Using Limited Fine-Tuning Data In Table 5, we also compare effect of using limited data for fine-tuning. We note that obtaining high-quality data tailored for downstream tasks is often expensive. Thus, task-specific pre-training which leverages existing pre-training data with different objectives is an attractive alternative. We find this to be the case, especially for Mid-Fusion where using task-specific pre-training and fine-tuning on just 10% of IP2P data leads to similar performance as directly fine-tuning on the entire downstream dataset.

Sampling strategy in Task-specific Pre-training During the task-specific pre-training stage, since creating the visual pairs is performed on the go and different sampling strategies can be utilized. For a given image, we could either sample a random image (Rand) or we could condition it on some objective such as having at least one common object (Cond). Comparing the two for both RAIV and the Reasoning task, we find the conditional sampling to be useful likely due to training on harder examples.

Visualization We provide qualitative analysis of our model outputs in Appendix C.

5. Conclusion

In this work, we explore Reasoning Across Images and Video (RAIV) task which involves classifying a statement about a pair of visual inputs (images, videos or a mixed combination) as true or false. We introduce multiple datasets to study RAIV with semantically rich queries, and visually similar inputs as well as allow reasoning for the provided answers. We investigate the potential for task-specific pre-training which involves additional pre-training on objectives similar to the downstream task but confined to the original pre-training dataset. Our experiments validate the effectiveness of including task-specific pre-training for improved downstream performance.

Acknowledgement: We thank the anonymous reviewers for their suggestions. This research was supported, in part, by the Office of Naval Research under grant #N00014-21-1-2802

Appendix

The appendix includes

1. Details on dataset creation and statistics.
2. Implementation details for the various baselines.
3. Visualization of outputs.

A. Datasets

A.1. Creating Datasets for RAIV

We first discuss the creation of Im-Im, Im-Vid and Vid-Vid datasets which are aimed to have semantically rich representations.

RAIV tasks involve a pair of images/videos and a given statement to be classified as True or False. We create multiple datasets using existing vision-language datasets which contain SRL annotations, namely, ImSitu [47] and VidSitu [36]. The main reason for choosing datasets with SRL annotations is to obtain high-quality "action+object" information in the image or video. We first summarize these two datasets.

Briefly, the ImSitu dataset is created by first obtaining a set of verbs and their corresponding roles from FrameNet [33]. Then top image results are retrieved from the web which includes the particular verb, followed by a strict annotation pipeline to denote the various entities participating in the action. The VidSitu dataset, which serves as an extension of ImSitu to videos, obtains 10-second-long movie clips with multiple actions. Each video is then segmented into five 2-second events, with each segment annotated with a verb obtained from PropBank [16]. Then, a referring expression is used to denote the entities appearing in the videos, which are filled in the various roles.

For both ImSitu and VidSitu, we obtain the "object" information from an object detector. We utilize VinVL [49] which involves a FasterRCNN [31] trained on multiple object detection datasets OpenImages [18], COCO [22], Visual Genome [17] and Object365 [37], and then fine-tuned on Visual Genome.

We note that both ImSitu and VidSitu use different sets of verbs for annotations. Since our datasets include both images and videos, we simplify our setting by only utilizing verbs that are common to both datasets. While this reduces the total amount of available data, it hugely simplifies the dataset creation pipeline. We also prune verbs with less than 20 annotations in either dataset. This results in 243 verbs which are shared in both datasets.

Another issue arises in the semantic role labeling formats for the two datasets. ImSitu annotations are based on FrameNet [33] whereas VidSitu annotations are based on PropBank [16]. We use existing heuristics based on the ordering and the use of roles to map the SRLs from FrameNet to Propbank annotations. Since we are mostly concerned about the "action+object" setting and not the individual roles such as instruments or tools, noise in this conversion doesn't adversely affect the dataset quality. Further, the annotations for the entities in VidSitu have referring expressions or phrases describing the entity which is different from entity annotation in ImSitu containing only a single noun. We circumvent this issue by considering only the lemmatized noun for the referring expressions. We also avoid very common objects such as "person" which is usually associated with the agent performing an action.

With both ImSitu and VidSitu datasets in hand, we now create RAIV datasets. We create the following variations: Image-Image (Im-Im), Image-Video (Im-Vid) and Video-Video (Vid-Vid) with images taken from ImSitu and videos taken from VidSitu. We note that while videos in VidSitu are 10 seconds long, for our experiments we only consider 2 second long clips which correspond to a particular event in the video. We further ensure the event is not duplicated in the next segment to avoid annotated entities not appearing within the given segment. After pruning, we are left with $63k$ images from ImSitu and $106k$ video segments from VidSitu. We utilize the same splits as in the original datasets to avoid any training dataset leakage into validation splits. For each of the datasets, we create approximately the same number of samples as in NLVR2 around $120k$ annotations with an even distribution of the verbs and objects but we note that our process allows creating more examples without any additional human effort.

We further take care to not introduce any spurious dataset bias. We follow NLVR2 in creating balanced validation and test sets by using the same unique statement where it is true for a particular pair and false for another pair in the given dataset to ensure no language-only bias in the dataset. The resulting datasets are suffixed with "T" to denote the statements are generated using templates resulting in Im-Im (T), Im-Vid (T), and Vid-Vid (T).

As our datasets are created semi-automatically, we also provide reasons for the false statements. For ease of evaluation we follow previous work in common-sense reasoning [19, 48] involving multiple-choice question setup where three reasons are provided and only one of the reasons is correct. The options are also generated via templates to prevent any language-only biases.

We summarize our pipeline for creating RAIV template datasets, i.e., Im-Im (T), Im-Vid (T), Vid-Vid (T) below.

1. Unify the annotations for ImSitu and VidSitu datasets, in particular the verbs.
2. Create mapping of objects, actions, and action+objects to image/video IDs in the datasets.
3. Sample a particular template based on object, action, or action + object. Then choose a particular object, action,

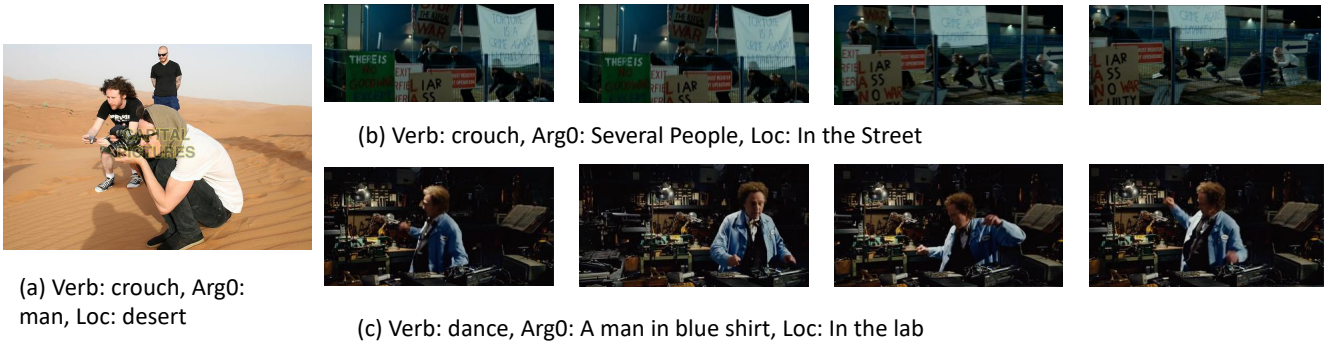


Figure 4. Example creation of generating template-based queries.

action+object.

4. Choose a particular image/video satisfying the above criteria.
5. Choose two other image/video, one which satisfies and another which doesn't satisfy the criteria. This provides us with a True and False statement.
6. In previous step, choosing them at random makes the problem too simple, so we condition it on having at least one shared SRL such as verb, object or location.
7. For the false statement, provide the reason for being false.
8. Repeat the process until enough samples are obtained.

We illustrate this with an example in Fig 4. Suppose the chosen template was "action", "In both images, people are doing X" where "X" is the action. Assume the chosen action was "crouch". Let the first sample chosen be Fig 4 (a). Given this image, we choose a "true" video as in Fig 4(b) and "false" video (c). Further, for the "false" pair, we know both contain the verb crouch, so we can provide the reason "people crouch in I1 but not in I2.".

We note we restrict to limited possible templates yet covering a wide-variety of possibility based on whether it is "action", "object" or "action + object". The possible templates are:

1. "In both I1 and I2, {p1}."
2. "In at least one of I1 or I2, {p1}."
3. "In exactly one of I1 or I2, {p1}."
4. "In neither I1 nor I2, {p1}.

Here, {p1} is short for placeholder and {Image} refers to Image1 or Image2. We also note that the clause can be easily modified such as "In both I1 and I2, {p1}" is same as "{p1} in both I1 and I2". The placeholder {p1} depends on the type of template. For instance, if it is object, it is "Obj is present", for actions it is "Subj is performing Verb". These templates can then be used to get the reasoning in the form of: "In both I1 and I2", "In I1 but not in I2", "In I2 but not in I1" or "In neither I1 nor I2".

Note that for during training, the SRLs are obtained from a pre-trained SRL detection system on the provided captions such as [39].

For validation and test sets, we utilize all the available annotaitons. For instance, VidSitu provides 10 verb annotations for each segment. Thus, when comparing for same verb, we consider all 10 annotations. Similarly, for other SRLs. This makes our validation and test sets more robust to noisy ground-truth data.

A.2. Creating Natural Language Queries

As noted in main paper, we utilize LLMs to create Natural Language Queries. We note that there are both pros and cons of using natural language queries as opposed to template queries. The main advantage of templated queries is that the output sentence has very controlled information and as a result we can create a reasoning question directly from the template. However, such model is of little practical use.

On the other hand, natural language queries can be directly used by end-user but obtaining natural language queries via humans is prohibitively expensive. Instead, we opt to use natural language queries using LLMs. However, we note that use of LLMs can cause errors in the generated sentence and there is no easy way to rectify them. Further, the obtained LLM outputs cannot be used for reasoning.

To generate the queries, we use Vicuna-13B [50] model which is initialized from LLaMA [44] and trained on outputs from ChatGPT [26] a closed-source model by OpenAI.

We use the LLM in two ways: (i) to create Im-Im (G), Im-Vid (G) and Vid-Vid (G) which are generated counterparts to the original templated datasets introduced above (ii) to create IP2P dataset which is obtained from InstructPix2Pix. While used in similar ways, there are some key distinctions.

For Im-Im, Im-Vid and Vid-Vid datasets, we directly take all the visual input pairs, obtain their annotation information

and pass it to the LLM and require it to generate a True statement. The obtained statement is then matched to another input pair for which it is false. Essentially, the “T” and “G” counterparts of the dataset have same visual input pairs but the exact sentences are different.

We prompt our LLM based on the original input query in the templated dataset. We use the following input:

""" Provide a True statement comparing the two images with the following information:

Image 1: {SRL} Image 2: {SRL}

The statement should be in the form of "{Template}, ...", only point out about {Image}. """

Here, {SRL} denotes the semantic roles for the given image/video, the {Template} denotes the chosen template as noted in previous section, and {Image} denotes which image was chosen (I1 or I2) for the true statement.

For instance, if the original query involved Fig4 (a), (c) with the template “In exactly one of”, with action+object, the input would be:

""" Provide a True statement comparing the two images with the following information:

Image 1: Verb: crouch, Subj: man, Loc: desert

Image 2: Verb: dance, Subj: a man in blue shirt, Loc: in the lab

The statement should be in the form of "In exactly one of the images, ...", only point out about image 1 """ This returns the output: "In exactly one of the images, a man is crouching in a desert." The same true statement is considered “false” for the other pair taken from corresponding “(T)” dataset.

For IP2P, the images are created using Stable-Diffusion. We have access to the image-caption pairs as well as the edit caption. To create a true statement, we provide the LLM with the original caption and the edit caption and ask it compare the images. To create a false statement, we keep the original caption but change the edit caption. Here, for each image pair we have a unique true and false statement.

To generate these queries:

1. Choose a given image-pair from IP2P dataset. This has an associated original caption, edit caption, and an output caption.
2. Randomly sample another edit caption different from the given edit caption. Denote this as edit caption 2.
3. Use LLM to compare original caption, and edit-caption for True statement. Similarly, use original caption and edit caption 2 for False statement.

For generating output with a new caption we use the following prompt:

""" The original image caption: ...

The original image is modified with the instruction: ...

Provide a true statement comparing original and new image. """

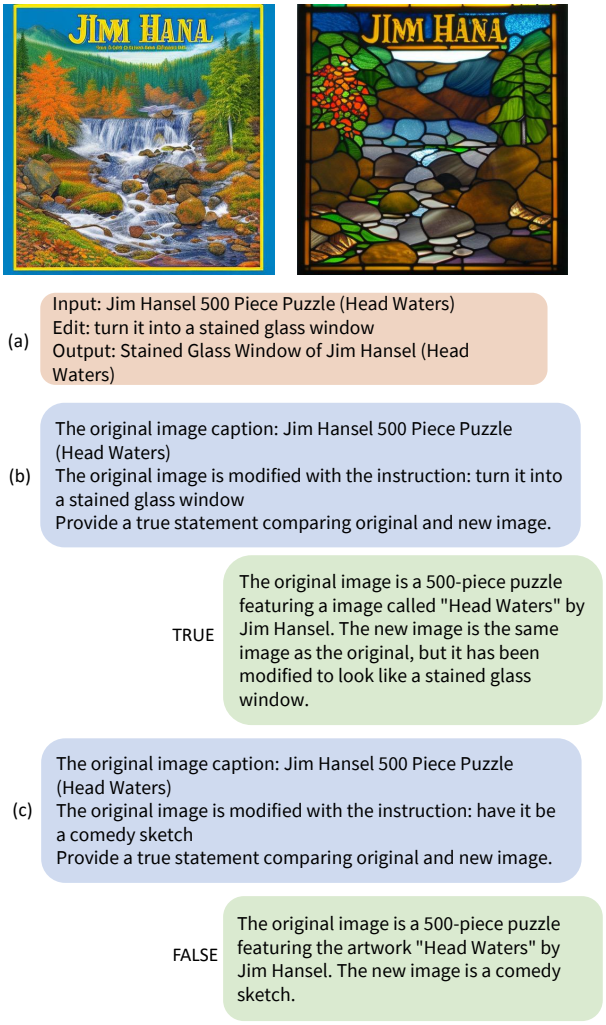


Figure 5. Illustration of generating queries for IP2P dataset. For a given image pair, we have (a) the sentence queries used to generate the image via Stable Diffusion (b) Using LLM to generate a True statement and (c) False statement by changing the edit caption.

As an example, to generate true statement for given pair : """ The original image caption: Jim Hansel 500 Piece Puzzle (Head Waters)

The original image is modified with the instruction: turn it into a stained glass window

Provide a true statement comparing original and new image. """

Instead for a false pair:

""" The original image caption: Jim Hansel 500 Piece Puzzle (Head Waters)

The original image is modified with the instruction: have it be a comedy sketch

Provide a true statement comparing original and new image. """


Q: In both I1 and I2, a person is swimming in a pool.

PrA: True
GtA: False

PrR: A person is swimming in a pool in both I1 and I2.
GtR: A person is swimming in a pool in I2 but not in I1

(a)

Q: A man kneels in exactly one of I1 and I2.

PrA: False
GtA: False

PrR: A man kneels in both I1 and I2
GtR: A man kneels in both I1 and I2.

(b)

Q: In neither I1 nor I2, a person is reading a book

PrA: False
GtA: False

PrR: A person is reading a book in both I1 and I2.
GtR: A person is reading a book in I2 but not in I1.

(c)

Figure 6. Model Predictions vs Ground-Truth for template-based ("T") validation datasets. (a) Im-Im (T), (b) Im-Vid (T), (c) Vid-Vid (T). PrA and GtA refer to Predicted and Ground-truth Answers respectively. PrR and GtR refer to predicted and ground-truth reasoning respectively.

B. Implementation Details

Implementation Details Our model and code are implemented in Pytorch. For all fine-tuning experiments, we follow identical settings as METER. For each dataset, we separately fine-tune the model for 10 epochs with differential learning rates of $1e^{-5}$ and $1e^{-4}$ for the bottom and top layers respectively.

We use 288×288 as the image dimension in all cases. For videos, we sample $K = 4$ frames per video where each video is 2 seconds long and sampled at 30 frames per second. For images, we simply provide a single temporal position embedding while for videos we have K temporal position embeddings. We use sinusoidal position embeddings following previous work [45].

In the task-specific pre-training step, we primarily use the COCO dataset instead of the entire ImgAll dataset in order to limit computation time, similar to the fine-tuning process on the downstream task. We also note that instead of using the object annotations available in COCO, we use the VinVL object detector outputs instead as it detects a larger number of categories outside of COCO. For videos, we use a subset of Kinetics videos from VATEX-en. We note that the videos in Kinetics are 10s long compared to 2s in the downstream dataset. To circumvent this issue, we first

Q: The original image features a beautiful young woman with curly blond hair on a black leather sofa, while the modified image features the same woman with a cat added to the scene.

PrA: True
GtA: True.

Q: The original image is a photograph of a beautiful young woman with curly blond hair sitting on a black leather sofa, while the modified image is a sculpture of the same woman in the same pose and setting.

PrA: False
GtA: False.

(a)

Q: The original image shows a wooden house next to the Iceland sea, while the new image depicts the same wooden house but with a haunted theme.

PrA: True
GtA: True

Q: The modified image has snow, which is not present in the original image.

PrA: True
GtA: False

(b)

Figure 7. Model Predictions vs Ground-truth for IP2P dataset. For a given pair of images, both the chosen True and False sentences are shown.

obtain an intersection of the videos from AVA-Kinetics [20] which gives us $5.7k$ videos where the keyframe of the person performing the action is provided. We particularly sample 2s clips around the keyframe. In general, we randomly sample 4 frames from the entire video.

We train for 10 epochs but reduce batch size to 256 with AdamW optimizer [24] with linear warm-up for initial 10% to $1e-4$ of the training followed by linear decay. We only utilize the last checkpoint and then perform fine-tuning on the target dataset. Most of our experiments are carried on 4x 2080Ti and 4x 3090Ti machines.

C. Visualization

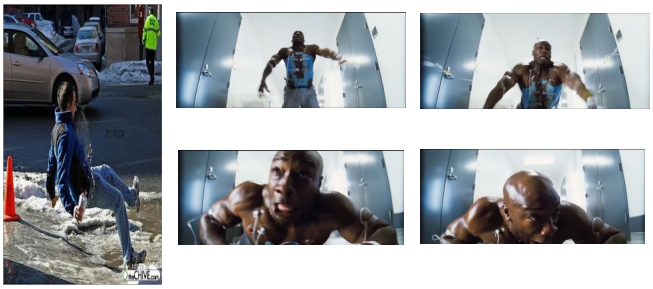
We provide qualitative examples from our dataset and outputs of our model as follows:

1. On Template-based queries and Reasoning, namely, Im-Im (T), Im-Vid(T), Vid-Vid (T) in Figure 6
2. IP2P Generated queries in Figure 7
3. On Generated queries, Im-Im (G), Im-Vid(G), Vid-Vid(G) in Figure 8

Q: In both images, a woman is performing an action with a rope in a gymnasium. The action being performed is skipping in the first image and climbing in the second image.

PrA: True
GtA: True

(a)



Q: In exactly one of the images, a man in white pants is depicted as falling.

PrA: False
GtA: True

(b)

Q: In at least one of the images, a girl with brown hair is depicted as grabbing a CD.

PrA: False
GtA: True

(c)

Figure 8. Model Predictions vs Ground-Truth for generated queries ("G") validation datasets. (a) Im-Im (G), (b) Im-Vid (G), (c) Vid-Vid (G). PrA and GtA refer to Predicted and Ground-truth Answers respectively.

References

[1] Aishwarya Agrawal, Jiasen Lu, Stanislaw Antol, Margaret Mitchell, C. Lawrence Zitnick, Devi Parikh, and Dhruv Batra. Vqa: Visual question answering. *International Journal of Computer Vision*, 123:4–31, 2015. 1

[2] Max Bain, Arsha Nagrani, Gül Varol, and Andrew Zisserman. Frozen in time: A joint video and image encoder for end-to-end retrieval. *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1708–1718, 2021. 3

[3] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. *arXiv preprint arXiv:2211.09800*, 2022. 2, 5

[4] Yu-Wei Chao, Yunfan Liu, Xieyang Liu, Huayi Zeng, and Jia Deng. Learning to detect human-object interactions. *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 381–389, 2018. 3

[5] Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco captions: Data collection and evaluation server. *ArXiv*, abs/1504.00325, 2015. 1

[6] Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. Uniter: Universal image-text representation learning. In *ECCV*, 2020. 1, 2, 3

[7] Jaemin Cho, Jie Lei, Hao Tan, and Mohit Bansal. Unifying vision-and-language tasks via text generation. In *ICML*, 2021. 3

[8] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 1

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. 2

[10] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *ICLR*, 2021. 2, 7

[11] Zi-Yi Dou, Yichong Xu, Zhe Gan, Jianfeng Wang, Shuohang Wang, Lijuan Wang, Chenguang Zhu, Nanyun Peng, Zicheng Liu, and Michael Zeng. An empirical study of training end-to-end vision-and-language transformers. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 18145–18155, 2022. 1, 2, 3, 5, 6

[12] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. *International Journal of Computer Vision*, 127:398–414, 2017. 1

[13] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc V. Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *ICML*, 2021. 2

[14] Zeeshan Khan, C.V. Jawahar, and Makarand Tapaswi. Grounded video situation recognition. *ArXiv*, abs/2210.10828, 2022. 3

[15] Wonjae Kim, Bokyung Son, and Ildoo Kim. Vilt: Vision-and-language transformer without convolution or region supervision. In *ICML*, 2021. 1, 2

[16] Paul R Kingsbury and Martha Palmer. From treebank to propbank. In *LREC*, pages 1989–1993, 2002. 9

[17] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, Michael Bernstein, and Li Fei-Fei. Visual genome: Connecting language and vision using crowdsourced dense image annotations. 2016. 6, 9

[18] Alina Kuznetsova, Hassan Rom, Neil Gordon Alldrin, Jasper R. R. Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, Tom Duerig, and Vittorio Ferrari. The open images dataset v4. *International Journal of Computer Vision*, 128:1956–1981, 2020. 9

[19] Jie Lei, Licheng Yu, Mohit Bansal, and Tamara L. Berg. Tvqa: Localized, compositional video question answering. In *EMNLP*, 2018. 6, 9

[20] Ang Li, Meghana Thotakuri, David A. Ross, João Carreira, Alexander Vostrikov, and Andrew Zisserman. The ava-kinetics localized human actions video dataset. *ArXiv*, abs/2005.00214, 2020. 6, 13

[21] Junnan Li, Ramprasaath R. Selvaraju, Akhilesh Deepak Gotmare, Shafiq R. Joty, Caiming Xiong, and Steven C. H. Hoi. Align before fuse: Vision and language representation learning with momentum distillation. In *NeurIPS*, 2021. 1, 2, 3, 6, 7

[22] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 5, 6, 9

[23] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019. 6

[24] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. 6, 13

[25] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *NeurIPS*, 2019. 2

[26] OpenAI. Gpt-4 technical report. *ArXiv*, abs/2303.08774, 2023. 2, 5, 10

[27] Vicente Ordonez, Girish Kulkarni, and Tamara Berg. Im2text: Describing images using 1 million captioned photographs. *Advances in neural information processing systems*, 24, 2011. 6

[28] Sarah Pratt, Mark Yatskar, Luca Weihs, Ali Farhadi, and Aniruddha Kembhavi. Grounded situation recognition. In *ECCV*, 2020. 3

[29] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 2, 6

[30] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019. 2

[31] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39:1137–1149, 2015. 7, 9

[32] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2021. 2, 5

[33] Josef Ruppenhofer, Michael Ellsworth, Myriam Schwarzer-Petruck, Christopher R Johnson, and Jan Scheffczyk. Framenet ii: Extended theory and practice. Technical report, International Computer Science Institute, 2016. 9

[34] Arka Sadhu, Kan Chen, and Ram Nevatia. Video object grounding using semantic roles in language description. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 3

[35] Arka Sadhu, Kan Chen, and R. Nevatia. Video question answering with phrases via semantic roles. In *NAACL*, 2021. 3

[36] Arka Sadhu, Tanmay Gupta, Mark Yatskar, Ram Nevatia, and Aniruddha Kembhavi. Visual semantic role labeling for video understanding. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2021. 1, 3, 4, 5, 9

[37] Shuai Shao, Zeming Li, Tianyuan Zhang, Chao Peng, Gang Yu, Xiangyu Zhang, Jing Li, and Jian Sun. Objects365: A large-scale, high-quality dataset for object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8430–8439, 2019. 9

[38] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2556–2565, Melbourne, Australia, July 2018. Association for Computational Linguistics. 6

[39] Peng Shi and Jimmy Lin. Simple bert models for relation extraction and semantic role labeling. *arXiv preprint arXiv:1904.05255*, 2019. 3, 4, 5, 6, 10

[40] Emma Strubell, Patrick Verga, Daniel Andor, David Weiss, and Andrew McCallum. Linguistically-informed self-attention for semantic role labeling. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5027–5038, Brussels, Belgium, Oct.-Nov. 2018. Association for Computational Linguistics. 4

[41] Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. Vl-bert: Pre-training of generic visual-linguistic representations. In *International Conference on Learning Representations*, 2020. 2

[42] Alane Suhr, Stephanie Zhou, Ally Zhang, Iris Zhang, Huajun Bai, and Yoav Artzi. A corpus for reasoning about natural language grounded in photographs. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6418–6428, Florence, Italy, July 2019. Association for Computational Linguistics. 1, 3

[43] Hao Tan and Mohit Bansal. Lxmert: Learning cross-modality encoder representations from transformers. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*, 2019. 2, 3

[44] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023. 2, 5, 10

[45] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 6, 12

[46] Xin Wang, Jiawei Wu, Junkun Chen, Lei Li, Yuan-Fang Wang, and William Yang Wang. Vatex: A large-scale, high-quality multilingual dataset for video-and-language research. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019. 6

[47] Mark Yatskar, Luke Zettlemoyer, and Ali Farhadi. Situation recognition: Visual semantic role labeling for image understanding. In *Conference on Computer Vision and Pattern Recognition*, 2016. 1, 3, 4, 5, 9

[48] Rowan Zellers, Yonatan Bisk, Ali Farhadi, and Yejin Choi. From recognition to cognition: Visual commonsense reasoning. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6713–6724, 2019. 6, 9

[49] Pengchuan Zhang, Xiujun Li, Xiaowei Hu, Jianwei Yang, Lei Zhang, Lijuan Wang, Yejin Choi, and Jianfeng Gao. Vinvl: Making visual representations matter in vision-language models. *CVPR 2021*, 2021. 3, 7, 9

[50] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric. P Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena, 2023. 5, 10