Towards General Purpose Vision Systems: An End-to-End Task-Agnostic Vision-Language Architecture

Tanmay Gupta[1] Amita Kamath[1] Aniruddha Kembhavi[1] Derek Hoiem[2]
[1]PRIOR @ Allen Institute for AI [2]University of Illinois at Urbana-Champaign
https://prior.allenai.org/projects/gpv

Abstract

Computer vision systems today are primarily N-purpose systems, designed and trained for a predefined set of tasks. Adapting such systems to new tasks is challenging and often requires non-trivial modifications to the network architecture (e.g. adding new output heads) or training process (e.g. adding new losses). To reduce the time and expertise required to develop new applications, we would like to create general purpose vision systems that can learn and perform a range of tasks without any modification to the architecture or learning process. In this paper, we propose GPV-1*, a task-agnostic vision-language architecture that can learn and perform tasks that involve receiving an image and producing text and/or bounding boxes, including classification, localization, visual question answering, captioning, and more. We also propose evaluations of generality of architecture, skill-concept*[1] *transfer, and learning efficiency that may inform future work on general purpose vision. Our experiments indicate* GPV-1 *is effective at multiple tasks, reuses some concept knowledge across tasks, can perform the Referring Expressions task zero-shot, and further improves upon the zero-shot performance using a few training samples.*

1. Introduction

Computer vision systems today are N-purpose learners — designed, trained, and limited to N predetermined tasks. Single-purpose models specialize in a single task, and adapting them to a new task or dataset requires an architecture change, minimally replacing the last classification layer. Multi-purpose models, such as Mask-RCNN [13], simultaneously solve more than one task, but the architecture and learning are tailored to specific tasks which must be defined in advance. In vision-language models [32], dedicated output heads are typically used for each task and dataset.

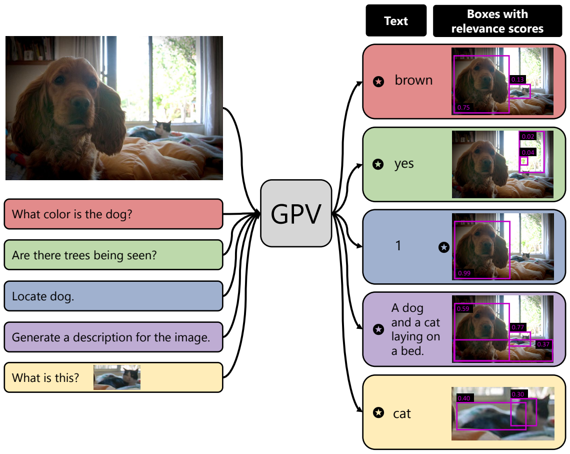
Text
Boxes with relevance scores
What color is the dog?
Are there trees being seen?
Locate dog.
Generate a description for the image.
What is this?
GPV
brown
0.13
0.75
yes
0.02
0.04
1
0.99
A dog and a cat laying on a bed.
0.59
0.77
0.37
cat
0.40
0.30

Figure 1. **A task-agnostic vision-language architecture.** GPV-1 takes an image and a natural language task description and outputs bounding boxes, confidences and text. GPV-1 can be trained end-to-end on any task that requires a box or text output, without any architecture modifications such as adding a new task-head. Results correspond to a model trained to perform VQA, localization, captioning, and classification tasks. Star indicates the output modality supervised during training for each task.

Analogous to a general purpose computer, a general purpose vision (GPV) system is designed to carry out many vision tasks, not all known at the time of design, constrained only by its input modalities, memory/instructions, and output modalities. General purpose systems enable new applications to be developed without knowledge of or access to the underlying mechanics. The NLP community has made significant progress in this direction with sequence-to-sequence transformer-based models, such as T5 [41] and GPT-3 [3], which can be trained to solve many language tasks without changing the architecture. We believe such advances are now possible within computer vision, though with many new challenges.

In this paper, we propose an end-to-end trainable task-agnostic vision-language architecture, GPV-1, as a step towards general purpose vision systems. As input, our system

[1] For this work, we define concepts, skills and tasks as follows: **Concepts** – nouns (*e.g. car, person, dog*), **Skills** – operations that we wish to perform on the given inputs (*e.g.* classification, object detection, image captioning), **Tasks** – predefined combinations of a set of skills performed on a set of concepts (*e.g.* ImageNet classification task involves the skill of image classification across 1000 concepts).

receives an image and a text description of a task. The system outputs bounding boxes, confidences, and text that are relevant to the task and image. A user can input an image and query the system with a variety of requests such as *"What make is the blue car?"* (visual question answering), *"Locate all the sedans"* (localization), and *"Describe the image"* (captioning). Each query elicits a different response using output heads that are shared across tasks. Defining the task through natural language allows the user to request GPV-1 to perform or learn a task without knowledge of its architecture or previous training. For example, our experiments show that GPV-1 can perform the referring expressions task without any training examples for that task and, when provided training examples, learns more quickly than special purpose models.

Beyond performing well on trained skill-concept combinations (contained in training tasks), GPV systems should be able to learn new tasks efficiently with the same architecture as well as generalize to novel skill-concept combinations for learned skills by transferring concept knowledge from other skills. These abilities are not usually applicable or measured in specialized systems. Therefore, we propose evaluations that measure three forms of generality:

- **Generality of architecture**: Learn any task within a broad domain specified only through input/output modalities without change to network structure (e.g. learn to classify bird species, without adding new output heads)
- **Generality of concepts across skills**: Perform tasks in skill-concept combinations not seen during training (e.g. localize "muskrat" after learning to answer questions about "muskrats")
- **Generality of learning**: Learn new tasks sample-efficiently with minimal loss to performance on previously learned tasks

To test generality of architecture, we train and evaluate our system's ability to perform visual question answering (VQA), captioning, object classification, and object localization on the COCO dataset [29], as well as test zero-shot generalization to a referring expression task. To test generality of concepts across skills, we present a new split of the COCO images and corresponding task annotations called COCO-SCE (Skill-Concept Evaluation). In COCO-SCE, some concepts (objects) are held-out from each task but exposed via other tasks, and then evaluate performance on samples containing held-out concepts. To test generality of learning, we fine-tune our system on the referring expressions task and measure its learning curve and extent of forgetting previously learned tasks.

In summary, our main contributions include: (1) **An end-to-end trainable, task-agnostic vision-language architecture** for learning and performing classification, grounding, visual question answering, captioning, and other tasks that involve image, text and bounding box modalities. (2) **Evaluation** that tests generality of architecture, skill-concept transfer, and learning ability.

2. Related Work

Single-purpose vision-language models. Over the last decade, specialized and effective approaches have been developed for vision-language tasks, including image captioning [10,21,25,33,47,54], phrase grounding [37,38,43], referring expression comprehension [22, 34], visual question answering (VQA) [2,11,16,48,51,55], visual dialog [7], and text-to-image generation [6]. Advances that have pushed the performance envelope include cross-model transformer architectures [45], powerful self-supervised [3, 8, 28] and multitask [41] language models, pretrained visual representations from object and attribute detectors [1, 57] or text conditioned detectors [20], and large-scale image/video-text [19,27,39,56] pretraining.

N-purpose vision-language models. Several recent works aim to unify vision-language tasks with a common architecture. UniT trains a single model for 7 tasks including detection and vision-language tasks but uses task specific heads and does not support captioning. 12-in-1 [32] jointly trains VilBERT [31] on 12 vision-language tasks but with 6 output heads (1 per task group). VL-T5 [5] adapts T5 [41], a text-to-text architecture pretrained on a mix of self-supervised and supervised tasks, to jointly train on vision-language tasks with only a text generation head (T5's text decoder). Both of these approaches rely on pre-extracted bounding boxes and region features from an object and attributes detector [1] and are not end-to-end trainable. E2E-VLP [53] presents an end-to-end trainable architecture that is extensively pretrained with masked language modeling, image-text matching, captioning, and object detection objectives, each with a different output head. However, the pretrained model is finetuned separately on each task and therefore does not support multiple tasks with a common set of weights. On the other hand, GPV-1 is both end-to-end trainable and jointly trained on multiple vision-language tasks. Our architecture takes an image and a textual task description as inputs, and has an output head per modality, namely text, bounding boxes, and relevance scores. Other exciting efforts towards creating general purpose vision architectures include Perceiver [18] and Perceiver IO [17], but their potential for multitask learning and utility for vision-language tasks such as VQA and captioning remains to be explored.

Task descriptions as a means to architecture generality. Several works in the natural language domain have tried to blur or erase artificial task boundaries by framing each task as text-to-text transformation with the task specified through a task description. Task descriptions range from templated prompts [41] to natural language descriptions [36, 49]. Kumar et al. [26] show that multiple tasks,

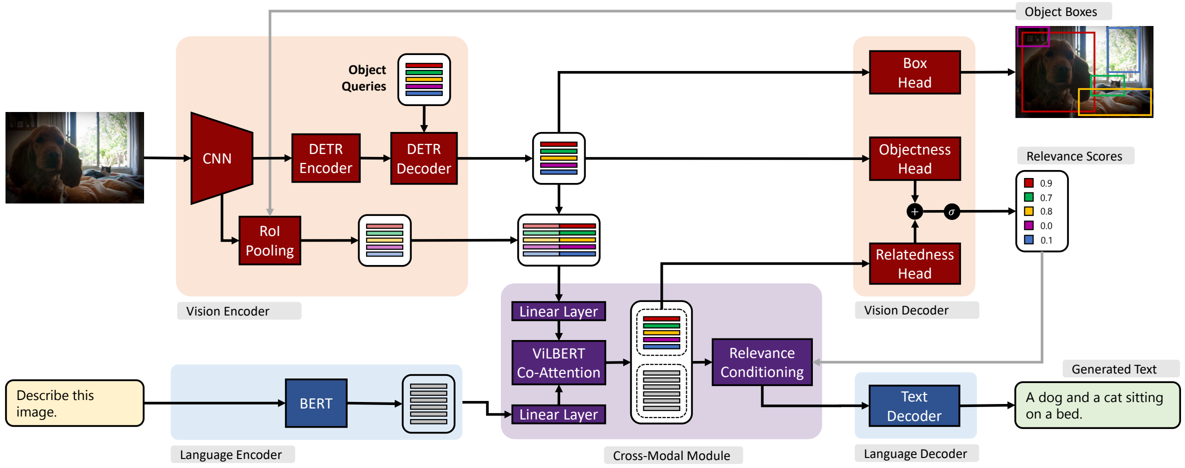

Figure 2. **Architecture of** GPV-1. **Vision**, **language**, and **cross-modal** modules are color-coded (see Sec. 3 for details).

such as part-of-speech tagging, question answering, and classification, can be formulated as a sequence-to-sequence transformation and solved with a single task-agnostic architecture but with separate parameters trained for each task. Works such as DecaNLP [35] and UnifiedQA [23] have trained single models to perform multiple tasks by reformulating each task as question answering allowing the individual task-performances to benefit from more data with diverse supervision while sharing model parameters. Works such as T5 [41], GPT [3,40] have also highlighted the transfer learning capabilities of unified models specially in zero-shot and few-shot scenarios.

Skill-Concept Evaluation. Few works attempt to learn a concept for one task and apply it to another, e.g. learning to categorize an image as "aardvark" and being able to detect or answer questions about aardvarks. As one example, Gupta et al. [12] show that formulating visual recognition and VQA in terms of inner products of word and image-region representations leads to inductive transfer between recognition and VQA tasks. Other works focus on unidirectional transfer to a single task such as captioning [15] or VQA [50]. With our COCO-SCE benchmark, we propose a systematic evaluation of generality of concepts across four standard vision-language tasks by holding out certain concepts from each task while exposing them via other tasks and then measuring performance separately on seen and held-out concepts for each task.

3. The GPV-1 model

3.1. Architecture Overview

The most distinctive aspect of our GPV-1 system is that tasks are defined through natural language text input, instead of multi-head outputs. Most systems, for example, that perform ImageNet [9] classification and COCO detection would have one 1000-class confidence output head and another 80-class box and confidence output head. More tasks or more datasets would require more output heads. Once trained, such a system will always produce 1,080 types of confidence and 80 classes of bounding boxes.

GPV-1 does not have explicit task boundaries and instead takes in a natural language *task description* such as "What is sitting on the sofa?" (VQA), "Find all instances of dogs" (localization), "What is going on in the image" (captioning), or "What kind of object is this?" (classification). GPV-1 interprets and performs all tasks using the same language/vision/cross-modal encoders and decoders. In training, the localization task has bounding box ground truth, while others such as classification, question answering, and captioning have text ground truth. Yet, all tasks involve common skills such as interpreting the task description, localizing objects, representing image regions, and determining relevance to the task. A new task, such as referring expressions which has bounding box ground truth, can be defined simply by providing new inputs ("Find the man wearing a green shirt") and output supervision (bounding box). Thus, limited only by modalities that it can sense and produce, GPV-1 can be trained to perform a wide range of tasks without task-specific modifications to the architecture or learning.

Fig. 2 provides an overview of GPV-1's architecture consisting of a visual encoder, language encoder, vision-language co-attention module, and output heads for the supported output modalities – boxes, relevance scores, and text. First, we encode the image using the CNN backbone and the transformer encoder-decoder from DETR [4], an end-to-end trainable object detector. Simultaneously, the natural language task description is encoded with BERT [8]. Then, to cross-contextualize representations from the visual and language encoders, we use ViLBERT's co-attention module [31]. Box and objectness heads predict task-agnostic bounding boxes and scores. The relatedness head predicts a task-specific score for each output box that is combined with the objectness scores to obtain relevance scores. The text decoder is an autoregressive transformer decoder that generates text output with relevance-conditioned outputs from cross-modal module serving as memory.

3.2. Vision modules

We use a DETR based **visual encoder**. A ResNet-50 backbone [14] extracts a convolutional feature map that is fed into DETR's transformer encoder to get contextualized features for every grid location. The transformer decoder takes as input R $(= 100)$ object queries (learned constant vectors) and the contextualized grid features and produces region descriptors per object query. The main intuition is that the object queries serve as learnable anchors, and the transformer encoder-decoder trained on detection eliminates the need for non-maximum suppression as a post-processing step. The complete region encoding is obtained by concatenating DETR's transformer features, which encode location and limited appearance information, with RoI pooled features from the CNN backbone.

As a **vision decoder**, GPV-1 uses DETR's box head to predict bounding boxes from region descriptors, resulting in R region proposals. These bounding boxes are used for grounding and detection tasks as well as for RoI pooling from the CNN backbone. We also replace DETR's 80-way object classification layer with a binary objectness classification layer, which contributes to determining relevance.

3.3. Language modules

The **language encoder** is used to encode the task description. We use BERT's WordPiece tokenizer [52] to obtain sub-word tokens for the language input and a pretrained BERT model to compute representations. Subword tokenization provides robustness to out-of-vocabulary words, and large scale language model pretraining allows GPV-1 to better handle paraphrases of language queries and zero-shot generalization to novel task descriptions, assuming semantic similarity to previously seen descriptions in the BERT embedding space.

The **language decoder** outputs words to classify, describe, or answer the input. Specifically, the sequence of co-attended region representations and language query's token representations are concatenated to construct a single sequence that serves as memory for the transformer text decoder. At each generation step, the sequence of words generated thus far are fed into the decoder along with the memory and a distribution over the vocabulary words is predicted to sample the next word. The inputs to the transformer decoder are trainable word embeddings. The output logit for a vocabulary word is obtained by taking dot product between the embedding vector output by the decoder and a linearly transformed BERT encoding of the word.

3.4. Cross-modal modules

The region descriptors from the vision modules and sub-token representations from the language module are transformed by linear layers to equal dimension vectors and fed into ViLBERT's **co-attention** layers for cross-contextualization. The relatedness head uses the co-attended region features to predict logits that indicate relevance of regions to the task description. These logits are added to logits from the objectness head and transformed into region-relevance scores by a sigmoid activation. These relevance scores are used to rank bounding boxes or indicate importance of regions to performing the task.

Relevance conditioning modulates the co-attended visual features with relevance scores. Specifically, the relevance score s of each region is used to weight learned vectors $\{v_{\text{rel}}, v_{\text{nrel}}\}$, which are added to the region features before feeding to the decoder. This conditioning enables supervision from the text decoder to affect the relatedness and objectness heads. In this way, a model trained to produce captions for images of peacocks may learn to localize peacocks, and, conversely, the ability to localize peacocks may translate to improved caption quality.

3.5. Training

Each training sample consists of an image, a task description, and targets. Depending on the task, targets could consist of ground truth bounding boxes, text, or both. In each training iteration, we uniformly draw samples across all tasks to construct mini-batches. For all samples that contain a text target, we maximize the log-likelihood of the ground truth text. For all samples that contain bounding boxes as targets, we use DETR's Hungarian loss for training the box and relevance prediction.

Initialization. We initialize all vision modules except the last linear layer in the objectness head with weights from DETR pretrained on either COCO or COCO-SCE (Sec. 4.2) object detection data. BERT is pretrained on BooksCorpus [59] and English Wikipedia.

Optimization. We train GPV-1 with a batch size of 120 and AdamW optimizer [30]. We keep DETR weights frozen for the first 10 epochs and finetune all modules except BERT for 30 more epochs. For learning rate (LR), we do a warm-up over the first 4 epochs to a maximum of 10^{-4} followed by linear decay to 0. Following DETR, we apply gradient clipping on visual module parameters and use a maximum learning rate of 10^{-5} for the CNN backbone. We use a $0.05\times$ lower text loss weight for captioning since more words are in the target text than other tasks.

4. Tasks and Data

Our experiments involve 5 tasks using images from the COCO dataset and annotations from the COCO, VQA V2 [11], and REFCOCO+ [22] datasets. Sec. 4.1 describes how these tasks are posed to our general purpose system along with respective losses and metrics used for training and evaluation. Sec. 4.2 details how samples are created for each task from the original annotations and introduces our COCO-SCE split for testing the generalization of concepts across skills.

4.1. Tasks

Our experiments mainly involve 4 tasks – VQA, Captioning, Localization, and Classification. We only use Referring Expressions to test the learning ability of GPV-1.

VQA aims to answer a question given an image. The input is an image/text pair, and the output is text. While training, the loss employed is the negative log likelihood of the ground truth answer text. We use the standard VQA evaluation metric (annotator-agreement weighted answer accuracy) [2] to report results.

Captioning aims to produce a description of an image. The input is an image and a prompt, such as "Describe the image" or "What is going on in the image?", and the output is text. While training, the loss employed is the negative log likelihood of the annotated caption. The evaluation metric reported is CIDEr-D [46] that measures the similarity of the generated and ground truth captions.

Localization aims to produce a tightly fitting bounding box to an object. The input is an image and a prompt, such as "Find all instances of dogs" or "Locate the chairs", and the output is a set of ranked bounding boxes. Training uses DETR's Hungarian loss. Evaluation is an average of per-query average-precision (AP) with a 0.5 bounding box intersection over union (IOU) threshold. For example, if an image contains two target objects and the correctness of the top four ranked boxes is {True, False, False, True}, the AP is (1/1+2/4)/2=0.75 (every-point interpolation). The reported number is AP averaged over samples.

Classification aims to assign a category to a region. The input is an image patch and a prompt such as "What is this thing?" or "What object is this?", and the output is text. In principle, GPV-1 can produce any category label within the large vocabulary of the text decoder, including words that it has not seen within its classification training data. However, for evaluation, a K-way classification is performed by suppressing outputs that do not correspond to any of the applicable K categories. The training loss used is the negative log likelihood of text output, and evaluation is accuracy averaged over samples.

Referring expressions (RefExp) aims to localize a single region that corresponds to a phrase. The input is an image and a referring expression such as "the man wearing a green shirt", and the output is one bounding box. While the training loss and evaluation is the same as localization, the key distinction is disambiguation of the referred instance among other instances of the same object category in the image.

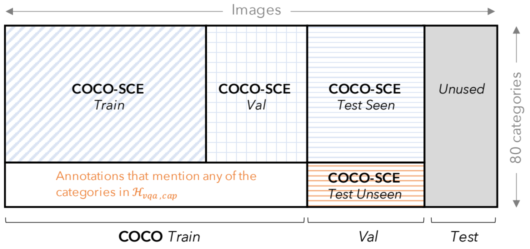

Figure 3. COCO-SCE: A split of COCO images and annotations to test the generalization of concepts across skills. Schematic shows train, val and test samples used for VQA.

4.2. Data

We present experiments using images from the richly annotated COCO dataset. We use question and answer annotations from the VQA V2 dataset, referring expressions from REFCOCO+, and COCO annotations for other tasks.

Data samples. VQA samples consist of the original questions as prompts paired with the most-agreed answer among annotators. For captioning, COCO provides 5 captions per image, each of which is treated as a different sample paired with one of 14 captioning prompt templates. We generate localization samples for each object category in the image using one of 18 prompt templates paired with all instances for the category. For classification, we create a sample for each object category in the image by choosing one of the instances (cropped using the ground truth box) paired with one of 4 prompt templates. RefExp samples consist of referring expressions as prompts with corresponding boxes.

Data splits. We present results for GPV-1 and baselines on two data splits. First we train and evaluate models using the standard data splits for the corresponding tasks. This provides results for GPV-1 in the context of past work. Then, to test the ability of vision systems to generalize concepts across skills, we present a new split of the above annotations, named COCO-SCE (Skill-Concept Evaluation).

COCO-SCE. Fig. 3 presents a schematic of the proposed COCO-SCE splits. The 80 classes of COCO are split into 3 disjoint sets, specifying which tasks can use them for training and validation:

- $\mathcal{H}_{vqa,cap}$: 10 classes held-out from the VQA and captioning tasks in the train/val sets
- $\mathcal{H}_{cls,loc}$: 10 different classes held-out from the classification and localization tasks in the train/val sets
- $\mathcal{S}$: 60 remaining classes are not held out from any tasks

When a category is held out, any annotations containing that word are not used for training or val. E.g., if *boat* is a held out category for VQA, then the annotation {"What color is the boat?", "Blue"} would be excluded from the train/val set. Other annotations from the same image may still be used, e.g. {"Is it a sunny day?", "Yes"}. Also, the classification and localization annotations for *boat* would be included in train/val for respective tasks. The assignment of

Split	Model	VQA	Cap.	Loc.	Class.
COCO-SCE	[a] Specialized Model	56.6	0.832	62.4	75.2
	[b] 1-Task GPV-1	55.9	0.855	**64.8**	75.3
	[c] Multitask GPV-1	**58.8**	**0.908**	64.7	**75.4**
COCO	[d] Specialized Model	60.1	0.961	**75.2**	83.3
	[e] Multitask GPV-1	**62.5**	**1.023**	73.0	**83.6**

Table 1. **Comparison to special purpose baselines** (COCO-SCE **and** COCO **splits**): Our jointly trained GPV-1 compares well to specialized single-task baselines as well as GPV-1 trained on individual task data. On COCO split, we report test-server results for VQA and captioning and validation results for localization and classification as the annotations for test images are hidden. On COCO-SCE split, we report test results for all tasks.

categories to $\mathcal{H}_{vqa,cap}$, $\mathcal{H}_{cls,loc}$, and $\mathcal{S}$ is random, except that we assign *person* to $\mathcal{S}$, because it is so common.

Images in COCO-SCE train and val sets come from COCO train set, and images in COCO-SCE test set are those in the COCO validation set (as COCO test annotations are hidden). COCO-SCE train and val splits are created by first creating an 80-20 partition of COCO train images and then for each task discarding samples that expose the held-out categories for that task through the annotations. On the test set we report performance separately for samples belonging to "seen" (e.g. $\mathcal{S} \cup \mathcal{H}_{cls,loc}$ for VQA) and "unseen" (e.g. $\mathcal{H}_{vqa,cap}$ for VQA) categories for each task.

5. Experiments

Our experiments evaluate GPV-1 for its effectiveness compared to specialized models (Sec. 5.1), its ability to apply learned skills to unseen concepts for that skill (Sec. 5.2), its efficiency at learning new skills, and retention of previously learned skills (Sec. 5.3). Sec 5.4 provides ablations. Our COCO-SCE experiments are carefully designed to ensure that compared methods train on the same amount of skill data (although some models may have access to data from another skill) and to enable evaluation of concept transfer across skills by avoiding exposing the held-out concepts via pretraining on Conceptual Captions [44] or Visual Genome [1]. ImageNet pretraining while not ideal, is unavoidable as most vision models including DETR rely on it to bootstrap learning.

5.1. Generality *vs*. Effectiveness

Is generality of GPV-1 at the cost of effectiveness? We compare GPV-1 to competitive special purpose models designed for each task – ViLBERT [31] (VQA), VLP [58] (captioning), Faster-RCNN [42]) (localization) and Resnet-50 [14] (classification). To avoid conflating effectiveness of architecture with availability of more data, we retrain these models to only use COCO and VQA V2 annotations. For ViLBERT and VLP this requires replacing Visual Genome [24] bottom-up features [1] with Faster-RCNN features trained only on COCO and no pretraining on Conceptual Captions [44].

Tab. 1 shows that on the COCO-SCE split, the general purpose GPV-1 architecture trained on individual tasks compares favorably to each special purpose model (rows *a* vs *b*). Also, the generality of GPV-1 enables it to be jointly trained on all 4 tasks, leading to sizeable gains on 2 tasks and comparable results on others (rows *b* vs *c*). The same trends also hold when we compare models on the original COCO data-splits (rows *d* vs *e*), validating that these trends are not merely a product of our proposed splits. Together, these results establish that the generality of GPV-1 is not at the expense of effectiveness.

5.2. Skill-Concept Generalization

We wish to test generality of concepts across skills, *i.e*. the ability of a model to perform well on novel skill-concept combinations that were unseen during training. When training on a single task on COCO-SCE a model does not have access to any annotation on held-out concepts. For example, a model trained only on VQA will never see a question or answer about *horse* $\in \mathcal{H}_{vqa,cap}$. However, when training on all tasks, the model learns to localize and classify *horse* images. Therefore we expect the model to apply the acquired skill of question answering to answer questions about *horse* without explicitly being trained on *horse* VQA data.

Tab. 2 shows the performance of the specialized models and the 1-Task and Multitask GPV-1 models on the COCO-SCE full test split as well as separately on the subset of test data categorized as "seen" and "unseen" (see Fig. 3 for a schematic of these subsets for the VQA task). The 1-Task GPV-1 (row *b*) trained on individual tasks serves as a baseline to account for learned priors and dataset biases by the GPV-1 architecture. We observe significant gains by Multitask GPV-1 (row *c*) on the "unseen" subset across all tasks, particularly over the specialized models (row *c* vs row *a*) – indicating that the general purpose architecture is better suited at learning skills and then applying them to concepts that were unseen for that skill. We also report the performance of Multitask GPV-1 trained on the COCO training split (row *d*). Since this split exposes the model to held-out concepts for all tasks, it can serve as a loose upper bound for the "unseen" split.

5.3. Learning Generalization

A system exhibits good learning generalization if it can learn new skills sample-efficiently without forgetting previously-learned skills.

Learning ability. Fig. 4 (left) shows learning curves for GPV-1 and GPV-1-Loc when finetuning on the Referring Expressions task. GPV-1-Loc is pretrained on only the localization task (the only other task that has bounding-box supervision) while GPV-1 is pretrained on all four tasks. Multitask GPV-1 demonstrates much better zero-shot per-

Model	VQA			Captioning			Localization			Classification		
	Test	*Seen*	*Unseen*	Test	*Seen*	*Unseen*	Test	*Seen*	*Unseen*	Test	*Seen*	*Unseen*
[a] Specialized Model	56.6	57.2	45.2	0.832	0.867	0.501	62.4	68.1	7.4	75.2	83.0	0.0
[b] 1-Task GPV-1	55.9	56.5	41.9	0.855	0.891	0.524	**64.8**	**69.8**	16.4	75.3	**83.1**	0.0
[c] Multitask GPV-1	**58.8**	**59.3**	**47.7**	**0.908**	**0.944**	**0.560**	64.7	68.8	**25.0**	**75.4**	82.6	**5.4**
[d] Multitask GPV-1 **Oracle**	61.4	61.3	64.0	1.018	0.997	0.939	73.0	72.7	76.0	83.6	83.4	85.7

Table 2. **Skill-Concept Generalization**: Multitask achieves higher performance overall, especially for “Unseen” concepts. Classification and Localization “Seen” performance slightly decreases, likely because all tasks share the same images and VQA and captioning are more weakly supervised. GPV oracle performance, with no concepts held out, provides an upper-bound on “Unseen”. Rows *a,b,c* are trained and tested on the smaller COCO-SCE data split, while *d* uses the COCO split.

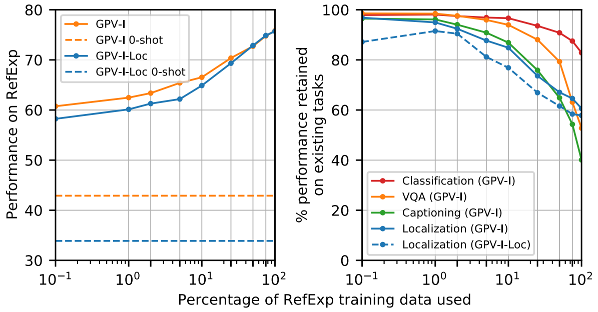

Figure 4. **Learning new skills and retention of previous skills.** *Left:* On REFCOCO+, multitask pretrained GPV-1 improves the 0-shot performance over single task pretrained GPV-1-Loc. GPV-1 also learns the new skill quicker than GPV-1-Loc, particularly in the lower data regime. *Right:* As REFCOCO+ training data increases, GPV-1 does forget existing skills but multitask GPV-1 is more resilient to forgetting than GPV-1-Loc (note that x-axes are log-scaled).

formance as well as better sample-efficiency in the low data regime. The learning of attributes and additional nouns provides a better starting point for referring expressions; e.g., while the localization-trained model starts with the ability to localize *person*, the multitask model is also familiar with *red* and *sweater* through captions and VQA and may better localize “the person wearing a red sweater”.

Retention. Fig. 4 (right) shows the percent of performance retained on the original tasks as GPV-1 is trained with increasing amounts of REFCOCO+ training data. Interestingly, Multitask GPV-1 forgets slower than GPV-1-Loc on the localization task. Localization and captioning suffer the most from catastrophic forgetting while classification shows robust retention. GPV-1 does not have explicit mechanisms for addressing forgetting, but our results highlight the importance of such mechanisms for general purpose learning.

5.4. Ablations

Tab. 3 ablates key factors that make GPV-1 effective. Finetuning end-to-end (as opposed to keeping DETR weights frozen) contributes to performance across all tasks (rows *a* vs *c*). RoI pooling significantly boosts performance for VQA, slightly for captioning, but leads to slight drop for localization and classification (rows *a* vs *b*).

	VQA	Cap.	Loc.	Class.
[a] Multitask GPV-1	**58.8**	**0.908**	64.7	75.4
[b] *w/o RoI features*	54.9	0.898	**65.3**	**76.6**
[c] *w/o Fine-Tuning*	56.4	0.883	63.4	71.5

Table 3. **Ablation**: Augmenting the vision transformer features with RoI features extracted from the CNN backbone helps VQA significantly and Captioning slightly, but is detrimental to Localization and Classification. The transformer features may be sufficient to fully model localization and classification, while VQA and Captioning benefit from additional information in the RoI features. Fine-tuning helps all tasks.

6. Limitations and Conclusion

The GPV-1 architecture can be trained to perform any image task that can be described and performed using words or boxes. Our experiments show that this generality does not come at the expense of accuracy, as GPV-1 compares well to specialized systems when trained on individual tasks and outperforms when trained jointly. However, several challenges remain. Generality of GPV-1 is at the cost of runtime efficiency compared to specialized systems. For example, using GPV-1 for detection requires a separate localization inference per object category. GPV-1 also achieves some skill-concept generalization, as measured on our COCO-SCE split, but a large gap to oracle indicates significant room for improvement. Our referring expression comprehension experiments show that while GPV-1 learns more quickly and forgets more slowly when trained on multiple tasks, catastrophic forgetting remains a challenge. While COCO-SCE does provide a controlled test bed for studying GPVs, our evaluation is limited to skills and concepts based on COCO. Finally, due to lack of an image generation head, GPV-1 currently does not support image manipulation or generation tasks such as colorization and segmentation. GPV-1 also does not handle non-image inputs such as videos or point clouds. Extending the capabilities of GPV-1 to new tasks and input types is an exciting challenge for future work.

Supplemental material contains additional training and dataset details, task prompts, ablations, analysis, potential negative impacts, and qualitiative results.

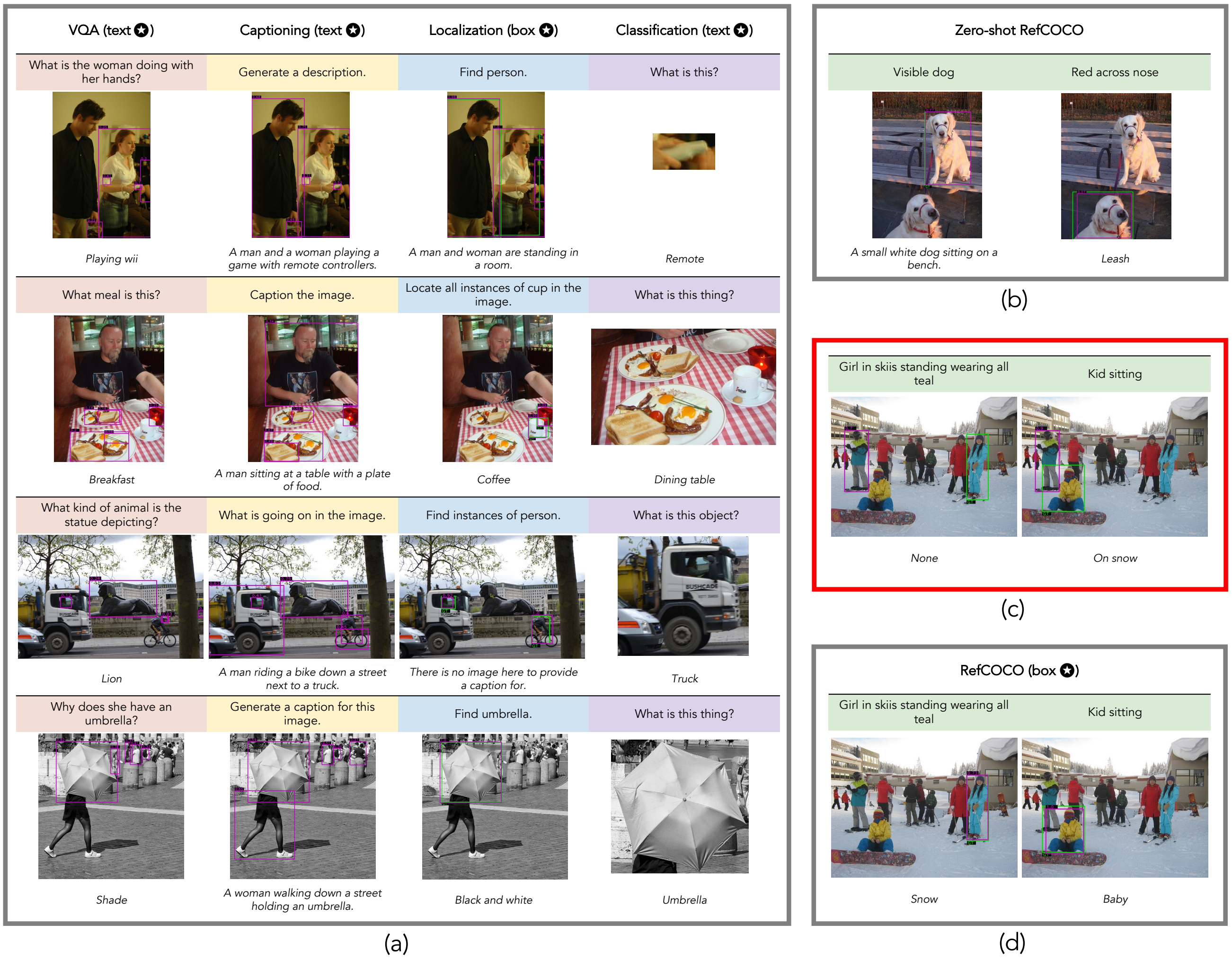
VQA (text ✪)
Captioning (text ✪)
Localization (box ✪)
Classification (text ✪)
What is the woman doing with her hands?
Generate a description.
Find person.
What is this?
Playing wii
A man and a woman playing a game with remote controllers.
A man and woman are standing in a room.
Remote
What meal is this?
Caption the image.
Locate all instances of cup in the image.
What is this thing?
Breakfast
A man sitting at a table with a plate of food.
Coffee
Dining table
What kind of animal is the statue depicting?
What is going on in the image.
Find instances of person.
What is this object?
Lion
A man riding a bike down a street next to a truck.
There is no image here to provide a caption for.
Truck
Why does she have an umbrella?
Generate a caption for this image.
Find umbrella.
What is this thing?
Shade
A woman walking down a street holding an umbrella.
Black and white
Umbrella
(a)
Zero-shot RefCOCO
Visible dog
Red across nose
A small white dog sitting on a bench.
Leash
(b)
Girl in skiis standing wearing all teal
Kid sitting
None
On snow
(c)
RefCOCO (box ✪)
Girl in skiis standing wearing all teal
Kid sitting
Snow
Baby
(d)

Figure 5. **Qualitative Results:** Prompts are shown in colored boxes (one color per task) with box and text predictions below. (a) GPV-1 learns to output the expected modality (indicated by star) for each task, but also provides unsolicited yet informative commentary for the localization task and relevant regions for VQA and captioning. (b) GPV-1 can perform 0-shot referring expression comprehension. GPV-1 learns to correct zero-shot mistakes (c) when finetuned on annotations for the task (d).

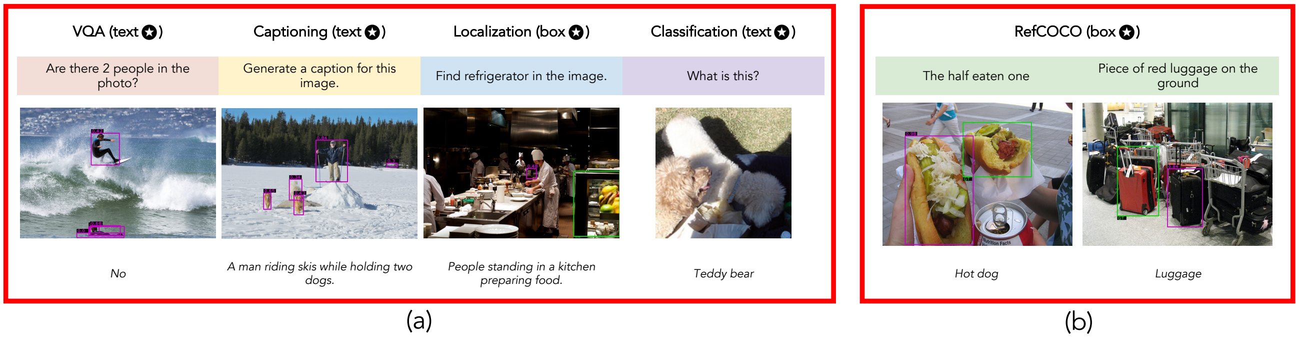
VQA (text ✪)
Captioning (text ✪)
Localization (box ✪)
Classification (text ✪)
Are there 2 people in the photo?
Generate a caption for this image.
Find refrigerator in the image.
What is this?
No
A man riding skis while holding two dogs.
People standing in a kitchen preparing food.
Teddy bear
(a)
RefCOCO (box ✪)
The half eaten one
Piece of red luggage on the ground
Hot dog
Luggage
(b)

Figure 6. **Failure Cases:** (a) GPV-1 fails to count the number of people despite localizing them, and is unable to locate objects such as ski poles and refrigerators. (b) RefCoco+ failures showing the model locating an incorrect object from the correct category.

Acknowledgments. This work is partially supported by ONR MURI Award N00014-16-1-2007 and ONR award N00014-21-1-2705.

References

[1] Peter Anderson, X. He, C. Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-up and top-down attention for image captioning and visual question answering. *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6077–6086, 2018. 2, 6

[2] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C. Lawrence Zitnick, and Devi Parikh. VQA: Visual question answering. In *ICCV*, 2015. 2, 5

[3] T. Brown, B. Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, G. Krüger, T. Henighan, R. Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, E. Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, J. Clark, Christopher Berner, Sam McCandlish, A. Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. *NeurIPS*, abs/2005.14165, 2020. 1, 2, 3

[4] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020. 3

[5] Jaemin Cho, Jie Lei, Haochen Tan, and Mohit Bansal. Unifying vision-and-language tasks via text generation. In *ICML*, 2021. 2

[6] Jaemin Cho, Jiasen Lu, Dustin Schwenk, Hannaneh Hajishirzi, and Aniruddha Kembhavi. X-lxmert: Paint, caption and answer questions with multi-modal transformers. In *EMNLP*, 2020. 2

[7] Abhishek Das, Satwik Kottur, Khushi Gupta, Avi Singh, Deshraj Yadav, José Moura, Devi Parikh, and Dhruv Batra. Visual dialog. In *CVPR*, 2017. 2

[8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina N. Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*, 2019. 2, 3

[9] Wei Dong, Richard Socher, Li Li-Jia, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *CVPR*, 2009. 3

[10] Ali Farhadi, Seyyed Hejrati, Amin Sadeghi, Peter Young, Cyrus Rashtchian, Julia Hockenmaier, and David A. Forsyth. Every picture tells a story: Generating sentences from images. In *ECCV*, 2010. 2

[11] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the V in VQA matter: Elevating the role of image understanding in Visual Question Answering. In *CVPR*, 2017. 2, 4

[12] Tanmay Gupta, Kevin J. Shih, Saurabh Singh, and Derek Hoiem. Aligned image-word representations improve inductive transfer across vision-language tasks. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 4223–4232, 2017. 3

[13] Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross Girshick. Mask R-CNN. In *ICCV*, 2017. 1

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 4, 6

[15] Lisa Anne Hendricks, Subhashini Venugopalan, Marcus Rohrbach, Raymond J. Mooney, Kate Saenko, and Trevor Darrell. Deep compositional captioning: Describing novel object categories without paired training data. *CVPR*, pages 1–10, 2016. 3

[16] Drew A Hudson and Christopher D Manning. GQA: A new dataset for real-world visual reasoning and compositional question answering. In *CVPR*, 2019. 2

[17] Andrew Jaegle, Sebastian Borgeaud, Jean-Baptiste Alayrac, Carl Doersch, Catalin Ionescu, David Ding, Skanda Koppula, Andrew Brock, Evan Shelhamer, Olivier J. H'enaff, Matthew M. Botvinick, Andrew Zisserman, Oriol Vinyals, and João Carreira. Perceiver io: A general architecture for structured inputs & outputs. *ArXiv*, abs/2107.14795, 2021. 2

[18] Andrew Jaegle, Felix Gimeno, Andrew Brock, Andrew Zisserman, Oriol Vinyals, and João Carreira. Perceiver: General perception with iterative attention. In *ICML*, 2021. 2

[19] 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

[20] Aishwarya Kamath, Mannat Singh, Yann LeCun, Ishan Misra, Gabriel Synnaeve, and Nicolas Carion. Mdetr - modulated detection for end-to-end multi-modal understanding. *ICCV*, pages 1760–1770, 2021. 2

[21] Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *CVPR*, 2015. 2

[22] Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara Berg. ReferItGame: Referring to objects in photographs of natural scenes. In *EMNLP*, 2014. 2, 4

[23] Daniel Khashabi, Sewon Min, Tushar Khot, Ashish Sabharwal, Oyvind Tafjord, P. Clark, and Hannaneh Hajishirzi. Unifiedqa: Crossing format boundaries with a single qa system. In *EMNLP*, 2020. 3

[24] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, Michael Bernstein, and Li Fei-Fei. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *IJCV*, 123:32–73, 2017. 6

[25] Girish Kulkarni, Visruth Premraj, Sagnik Dhar, Siming Li, Yejin Choi, Alexander C Berg, and Tamara L Berg. Baby talk: Understanding and generating image descriptions. In *CVPR*, 2011. 2

[26] Ankit Kumar, Ozan Irsoy, Peter Ondruska, Mohit Iyyer, James Bradbury, Ishaan Gulrajani, Victor Zhong, Romain Paulus, and Richard Socher. Ask me anything: Dynamic memory networks for natural language processing. In *ICML*, pages 1378–1387, 2016. 2

[27] Jie Lei, Linjie Li, Luowei Zhou, Zhe Gan, Tamara L. Berg, Mohit Bansal, and Jingjing Liu. Less is more: Clipbert for video-and-language learning via sparse sampling. In *CVPR*, 2021. 2

[28] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *ACL*, 2020. 2

[29] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common objects in context. In *ECCV*, 2014. 2

[30] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. 4

[31] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. In *NeurIPS*, 2019. 2, 3, 6

[32] Jiasen Lu, Vedanuj Goswami, Marcus Rohrbach, D. Parikh, and Stefan Lee. 12-in-1: Multi-task vision and language representation learning. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10434–10443, 2020. 1, 2

[33] Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. Neural baby talk. In *CVPR*, 2018. 2

[34] Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan L Yuille, and Kevin Murphy. Generation and comprehension of unambiguous object descriptions. In *CVPR*, 2016. 2

[35] B. McCann, N. Keskar, Caiming Xiong, and R. Socher. The natural language decathlon: Multitask learning as question answering. *ArXiv*, abs/1806.08730, 2018. 3

[36] Swaroop Mishra, Daniel Khashabi, Chitta Baral, and Hanna Hajishirzi. Cross-task generalization via natural language crowdsourcing instructions. In *ACL*, 2022. 2

[37] Bryan Plummer, Arun Mallya, Christopher Cervantes, Julia Hockenmaier, and Svetlana Lazebnik. Phrase localization and visual relationship detection with comprehensive image-language cues. In *ICCV*, 2017. 2

[38] Bryan Plummer, Liwei Wang, Chris Cervantes, Juan Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *ICCV*, 2015. 2

[39] 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, 2021. 2

[40] A. Radford, Jeffrey Wu, R. Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019. 3

[41] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, W. Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67, 2020. 1, 2, 3

[42] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in Neural Information Processing Systems*, 2015. 6

[43] Anna Rohrbach, Marcus Rohrbach, Ronghang Hu, Trevor Darrell, and Bernt Schiele. Grounding of textual phrases in images by reconstruction. In *ECCV*, 2016. 2

[44] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *ACL*, 2018. 6

[45] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 2

[46] Ramakrishna Vedantam, C. L. Zitnick, and D. Parikh. Cider: Consensus-based image description evaluation. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4566–4575, 2015. 5

[47] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. Show and tell: A neural image caption generator. In *CVPR*, 2015. 2

[48] Peng Wang, Qi Wu, Chunhua Shen, Anthony Dick, and Anton van den Hengel. FVQA: Fact-based visual question answering. *PAMI*, 40(10):2413–2427, 2018. 2

[49] Orion Weller, Nicholas Lourie, Matt Gardner, and Matthew E. Peters. Learning from task descriptions. *ArXiv*, abs/2011.08115, 2020. 2

[50] Spencer Whitehead, Hui Wu, Heng Ji, Rogério Schmidt Feris, Kate Saenko, and Uiuc MIT-IBM. Separating skills and concepts for novel visual question answering. *CVPR*, pages 5628–5637, 2021. 3

[51] Qi Wu, Peng Wang, Chunhua Shen, Anthony Dick, and Anton van den Hengel. Ask me anything: Free-form visual question answering based on knowledge from external sources. In *CVPR*, 2016. 2

[52] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Łukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv:1609.08144*, 2016. 4

[53] Haiyang Xu, Ming Yan, Chenliang Li, Bin Bi, Songfang Huang, Wenming Xiao, and Fei Huang. E2e-vlp: End-to-end vision-language pre-training enhanced by visual learning. In *ACL*, 2021. 2

[54] Kelvin Xu, Jimmy Ba, Ryan Kiros, Aaron Courville, Ruslan Salakhutdinov, Richard Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *ICML*, 2015. 2

[55] Rowan Zellers, Yonatan Bisk, Ali Farhadi, and Yejin Choi. From recognition to cognition: Visual commonsense reasoning. In *CVPR*, 2019. 2

[56] Rowan Zellers, Ximing Lu, Jack Hessel, Youngjae Yu, Jae Sung Park, Jize Cao, Ali Farhadi, and Yejin Choi. Merlot: Multimodal neural script knowledge models. *ArXiv*, abs/2106.02636, 2021. 2

[57] Pengchuan Zhang, Xiujun Li, Xiaowei Hu, Jianwei Yang, Lei Zhang, Lijuan Wang, Yejin Choi, and Jianfeng Gao. Vinvl: Revisiting visual representations in vision-language models. In *CVPR*, 2021. 2

[58] Luowei Zhou, Hamid Palangi, Lei Zhang, Houdong Hu, Jason J. Corso, and Jianfeng Gao. Unified vision-language pre-training for image captioning and vqa. *AAAI*, 2020. 6

[59] Yukun Zhu, Ryan Kiros, Richard Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *ICCV*, 2015. 4