

# 12-in-1: Multi-Task Vision and Language Representation Learning

Jiasen Lu<sup>3\*</sup> Vedanuj Goswami<sup>1\*</sup> Marcus Rohrbach<sup>1</sup> Devi Parikh<sup>1,3</sup> Stefan Lee<sup>2</sup>

<sup>1</sup>Facebook AI Research <sup>2</sup>Oregon State University <sup>3</sup>Georgia Institute of Technology

{vedanuj, mrf}@fb.com leestef@oregonstate.edu {jiasenlu, parikh}@gatech.edu

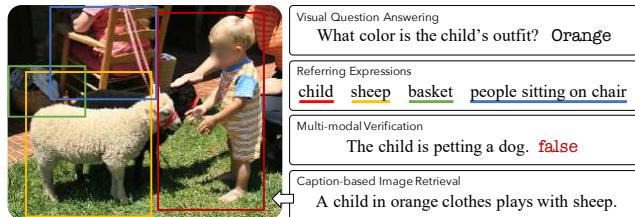
## Abstract

*Much of vision-and-language research focuses on a small but diverse set of independent tasks and supporting datasets often studied in isolation; however, the visually-grounded language understanding skills required for success at these tasks overlap significantly. In this work, we investigate these relationships between vision-and-language tasks by developing a large-scale, multi-task training regime. Our approach culminates in a single model on 12 datasets from four broad categories of task including visual question answering, caption-based image retrieval, grounding referring expressions, and multi-modal verification. Compared to independently trained single-task models, this represents a reduction from approximately 3 billion parameters to 270 million while simultaneously improving performance by 2.05 points on average across tasks. We use our multi-task framework to perform in-depth analysis of the effect of joint training diverse tasks. Further, we show that finetuning task-specific models from our single multi-task model can lead to further improvements, achieving performance at or above the state-of-the-art.*

## 1. Introduction

A compelling reason to study language and vision jointly is the promise of language as a universal and natural interface for visual reasoning problems – useful both in specifying a wide range of problems and in communicating AI responses. However, the current research landscape for visually-grounded language understanding is a patchwork of many specialized tasks like question answering or caption generation, each supported by a handful of datasets. As such, progress in this field has been measured by the independent improvement of bespoke models designed and trained for each of these specific tasks and datasets.

The recent rise of general architectures for vision-and-language [1, 23, 24, 27, 43, 45, 54] reduces the architectural differences across tasks. These models pretrain common architectures on self-supervised tasks to learn general visio-linguistic representations then fine-tune for specific



**Figure 1:** We introduce an approach for effective multi-task learning, training a single model on 12 popular vision-and-language datasets. This single model performs at par or even better than independent task-specific state-of-the-art approaches for many tasks.

datasets; however, the result is still a menagerie of independent task-specific models rather than a single unified model. This is dissatisfying in practice – the model that understands questions cannot ground noun phrases, the grounding model cannot retrieve images based on a description, and so forth. Further, this approach does not scale well as each new task requires storing a new model.

Beyond being intellectually dissatisfying, this task-based fracturing leaves quite a lot on the table. While individual tasks present different challenges and diverse interfaces, the underlying associations between language and visual concepts are often common across tasks. For example, learning to ground the referring expression “small red vase” requires understanding the same concepts as answering the question “What color is the small vase?”. Training multiple tasks jointly can potentially pool these different sources of grounding supervision. Further, developing models that can perform well on a wide range of tasks simultaneously can help guard against the research community overfitting to specific datasets and metrics.

In this work, we develop a multi-task model for discriminative vision-and-language tasks based on the recently proposed ViLBERT [27] model. We consider four categories of tasks – training jointly on a total of 12 different datasets. Our results not only show that a single model can perform all these tasks, but also that joint training can improve the performance compared to single-task training with the same architecture. Before undertaking this effort, it was not obvious to us that this would be the case – multitask training

\*Equal contribution

is notoriously challenging and vision-and-language datasets vary greatly in size, interface, and difficulty. Our model attains improvements of 0.25 to 4.19 absolute points from multi-task training – improving over corresponding single-task models for 11 out of 12 tasks. Further, we demonstrate that multi-task training is an effective pretraining step for single-task models – leading to further gains and setting a new state-of-the-art for 7 out of 12 tasks.

Large-scale multi-task learning is challenging as datasets can vary in size and difficulty. To address these issues, we introduce a dynamic stop-and-go training scheduler, task-dependent input tokens, and simple hyper-parameter heuristics. Using our proposed pipeline, we were able to train many multi-task models with varying datasets – assessing the relationships between different vision-and-language tasks in terms of their performance when trained together.

To summarize, we make the following contributions:

- We systematically analyze the joint training relationships between different of vision-and-language datasets and tasks and present a *Clean V&L Multi-Task setup*, which ensures no train-test leaks across task.
- We develop a single multi-task model trained on 12 popular V&L datasets. Compared to a set of independent models, this represents a reduction from  $\sim 3$  billion parameters to  $\sim 270$  million while simultaneously *improving* average performance by 2.05 points.
- We demonstrate that multi-task training is useful even in cases where single-task performance is paramount. On average, fine-tuning from our multi-task model for single tasks resulted in an average improvement of 2.98 points over baseline single-task trained models.

## 2. Vision-and-Language Tasks

### 2.1. Task-Groups and Datasets

We consider 12 popular vision and language datasets. These datasets cover a wide range of tasks and require diverse grounding granularity and reasoning skills. We group related datasets into four groups to facilitate our analysis:

**Vocab-based VQA.** Given an image and a natural-language question, select an answer from a fixed vocabulary. We consider three popular datasets for this group – VQA<sub>v2</sub> [15], GQA [17], and Visual Genome (VG) QA [21].

**Image Retrieval.** Given a caption and a pool of images, retrieve the target image that is best-described by the caption. We consider COCO [7] and Flickr30K [35] captioning datasets for this task-group.

**Referring Expressions.** Given a natural language expression and an image, identify the target region that is referred to by expression. The expression can vary greatly across datasets from simple noun phrases to multi-round dialogs.

	% Row-Task Test Images in Column-Task Train/Val Set											
	[A]	[B]	[C]	[D]	[E]	[F]	[G]	[H]	[I]	[J]	[K]	[L]
[A] VQA2.0 [15]	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
[B] VG QA [21]	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
[C] GQA [17]	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
[D] COCO [7]	100%	43%	33%	0%	0%	0%	0%	0%	7%	46%	0%	0%
[E] Flickr30k [35]	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	98%	0%
[F] RefCOCO [19]	100%	36%	27%	100%	0%	0%	0%	66%	8%	62%	0%	0%
[G] RefCOCO+ [19]	100%	38%	27%	100%	0%	0%	0%	66%	8%	62%	0%	0%
[H] RefCOCOg [30]	100%	41%	31%	100%	0%	53%	53%	0%	8%	63%	0%	0%
[I] Visual 7W [55]	50%	100%	79%	48%	0%	8%	8%	10%	0%	24%	0%	0%
[J] GuessWhat [13]	100%	40%	31%	96%	0%	20%	20%	26%	7%	0%	0%	0%
[K] SNLI-VE [49]	0%	0%	0%	0%	94%	0%	0%	0%	0%	0%	0%	0%
[L] NLVR <sup>2</sup> [44]	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%

**Table 1:** Percentage of row-task test images that are present in column-tasks train/val images.

We consider phrase grounding in RefCOCO(+g) [19, 30], Pointing questions in Visual7W [55], and dialog sequences in the GuessWhat [13]. We note that these language inputs vary significantly in terms of detail and structure.

**Multi-modal Verification.** Given one or more images and a natural language statement, judge the correctness or predict their semantic relationship. We consider NLVR<sup>2</sup> [44] and SNLI-VE [49]. In NLVR<sup>2</sup>, two images are given and the statement must be true for both to be true. In SNLI-VE, image-statement pairs are classified as representing an entailment, contradiction, or neutral. That is, whether the content of the image confirms, refutes, or is insufficient to comment on the truth of the corresponding statement.

### 2.2. A Clean V&L Multi-Task Setup

Many V&L tasks are built on top of each other and share significant overlap in terms of individual images. However, as each task is often examined in isolation, there does not exist an in-depth analysis of this overlap across different V&L tasks. Table 1 shows the percentage of test images for the target tasks which are present in other tasks’ train/val sets. As we can see, there exists significant overlap across tasks. Even though different tasks require different inputs and outputs, other task annotations will provide clues about the visual grounding – for example, a referring expression for a “blue striped ball” at training could unfairly improve a VQA model’s ability to answer “What color is the striped ball?” for the same image at test time. To avoid information leakage from the annotations of other tasks, we propose a *cleaned* multi-task split for V&L tasks where test images are removed from train/val for all the tasks. We stress that the test sets are not modified in any way, so our results are comparable to prior work. Cleaning results in about an 11% reduction in training data on average across datasets. Full details of this process and statistics regarding cleaned dataset size are available in the supplement.

## 3. Approach

### 3.1. Base Architecture

There has been a flurry of recent work developing general vision-and-language model architectures that are amenable to large-scale self-supervised pretraining. [1, 23,

24, 27, 43, 45, 54]. By pretraining general representations and then finetuning on single downstream tasks, these models set state-of-the-art in many tasks. For the base architecture in our experiments, we take the ViLBERT model proposed by Lu *et al.* [27]. We describe it here briefly.

At the interface level, ViLBERT takes as input an image  $I$  and text segment  $Q$  represented as the sequence  $\{\text{IMG}, v_1, \dots, v_T, \text{CLS}, w_1, \dots, w_T, \text{SEP}\}$  where  $\{v_i\}_{i=1}^T$  are image region features [2],  $\{w_j\}_{j=1}^T$  are word tokens, and the IMG, CLS, and SEP tokens are special markers. The model then outputs embeddings for each input  $\{h_{v_i}\}_{i=1}^T$ ,  $\{h_{w_j}\}_{j=1}^T$ ,  $h_{\text{IMG}}$ ,  $h_{\text{CLS}}$ , and  $h_{\text{SEP}}$ . As in [27], we take  $h_{\text{IMG}}$  and  $h_{\text{CLS}}$  as holistic image and text representations.

Internally, ViLBERT consists of two parallel BERT-style [14] models operating over image regions and text segments. Each stream is a series of transformer blocks (TRM) [48] connected by co-attentional transformer layers (Co-TRM) which enable information exchange between modalities. We use the default parameter setting, which has 6 / 12 layers of TRM for visual / linguistic streams respectively.

Like many of the models of this class, ViLBERT is pre-trained on the **Conceptual Caption dataset** [39] with two ‘proxy’ tasks: *masked multi-modal modelling* and *multi-modal alignment prediction*. The first randomly masks approximately 15% of both words and image tokens and reconstructs them given the remaining inputs. The later tasks the model with predicting whether an image and caption correspond or not. After pretraining, the model can be finetuned for strong performance for various downstream tasks.

We make two important modifications to this pretraining process. First, when masking visual regions we also mask other regions with significant overlap ( $> 0.4$  IoU) to avoid leaking visual information. This forces the model to rely more heavily on language to predict image content. Second, we do not enforce the masked multi-modal modelling loss when sampling a negative (unmatching) caption for multi-modal alignment prediction. This will effectively remove the noise introduced by negative samples. While orthogonal to our primary contribution of multi-task learning, we found these modifications to make the baseline model more effective. For further discussion, see the supplemental material. All models we present are first pretrained in this manner.

### 3.2. Multi-Task Learning

We consider a simple multi-task model where each task has a task-specific ‘head’ network that branches off a common, shared ‘trunk’ ViLBERT model. As such, we learn shared trunk parameters  $\theta_s$  and a set of task-specific layers  $\{\theta_t\}_{t=1}^T$  for  $T$  tasks. Our goal is to learn parameters  $\theta_s \cup \{\theta_t\}_{t=1}^T$  that minimize loss across all tasks. Details on heads and other modifications follow.

**Task Token.** While relying on the same groundings, different tasks may still require the model to process inputs differently – *e.g.* referring expressions just require grounding while VQA must follow grounding with additional reasoning. To enable this, we augment the query with a task token  $\text{TASK}_t$  such that the new input format is  $\{\text{IMG}, v_1, \dots, v_n, \text{CLS}, \text{TASK}_t, w_1, \dots, w_m, \text{SEP}\}$ . The architecture can then leverage this task information in a bottom-up manner. In what follows, we describe the task-specific heads by task groups.

**Vocab-Based VQA Output:** We compute an overall image-query representation as an element-wise product between the holistic  $h_{\text{IMG}}$  and  $h_{\text{CLS}}$  representations. As in [2, 17], we treat vocab-based VQA as a multi-label classification task – assigning a soft target score to each answer based on its relevancy to the ground truth answer. We compute scores for a set of the pre-defined answers  $A$  by using a two-layer MLP on top of the overall representation:

$$P_v(A|I, Q) = \sigma(\text{MLP}(h_{\text{IMG}} \odot h_{\text{CLS}})) \quad (1)$$

where  $\sigma$  is the sigmoid function. Due to the answer vocabulary differences, VQA and VG QA share the MLP and answer vocabulary while GQA learns a separate one.

**Image Retrieval Output:** Using the same overall representation, we compute an alignment score between image-caption pairs as:

$$\text{Rel}(I, Q) = W_i(h_{\text{IMG}} \odot h_{\text{CLS}}) \quad (2)$$

where  $W_i \in \mathbb{R}^{d \times 1}$  is shared across COCO and Flickr30k image retrieval tasks. As in [27], we train a 4-way multiple-choice against hard-negatives selected off-line and then fixed. Recent work has used online hard-negative mining [8, 23] but this is costly to compute.

**Referring Expressions Output:** We rerank a set of region proposals [50] given the referring expression. We pass the final representation  $h_{v_i}$  for each image region  $i$  into a learned projection  $W_r \in \mathbb{R}^{d \times 1}$  to predict a matching score.

$$\text{Rel}(v_i, Q) = W_r h_{v_i} \quad (3)$$

Note that  $Q$  may be either a phrase, question or dialog based on different tasks (RefCOCO+/g, Visual7W, GuessWhat).  $W_r$  is shared across all the referring expression tasks.

**Multi-modal Verification Output:** Taking NLVR<sup>2</sup> as an example, the input is a concatenation of two images ( $I_0$  and  $I_1$ ) and a statement  $Q$ , that the model must judge the validity of the statement given the images. We consider this a classification problem given an embedding that encodes the two image-statement pairs ( $I_0, Q$ ) and ( $I_1, Q$ ). The output probability is predicted by a 2-layer MLP with softmax:

$$P_v(C|I_0, I_1, Q) = \text{softmax} \left( \text{MLP} \left( \begin{bmatrix} h_{\text{IMG}}^0 \odot h_{\text{CLS}}^0 \\ h_{\text{IMG}}^1 \odot h_{\text{CLS}}^1 \end{bmatrix} \right) \right) \quad (4)$$

where  $[]$  is concatenation. For SNLI-VE, the input is a single image and statement. We thus learn a separate classifier of the same form that predicts the sentiment (entailment, neutral, contradiction) from the inputs.

### 3.3. Large-Scale Multitask Training

With 6 task heads, 12 datasets, and over 4.4 million individual training instances – training our multi-task ViLBERT model is a daunting proposition. Multi-task learning (especially at this scale) poses significant challenges as learning objectives have complex and unknown dynamics and may compete [41]. Further, vision-and-language datasets vary significantly in size and difficulty. For instance, a single epoch of VG (our largest dataset) corresponds to 19.8 epochs of RefCOCOg (our smallest). Likewise, when trained in isolation RefCOCOg converges in 5K iterations whereas VQAv2 takes 84K iterations (over 16 times more). Below, we describe the details of our multi-task training approach and techniques to overcome these challenges.

**Pretraining.** All our models are pretrained on Conceptual Caption dataset [39] including our self-supervised task modifications as described in Sec. 3.1.

**Round-Robin Batch-Level Sampling.** We consider a round-robin batch-level sampling regime that cycles through each task from the beginning of multi-task training. As such, one multi-task iteration consists of each task forwarding a batch and updating parameters in sequence.

**Dynamic Stop-and-Go.** As noted earlier, different tasks have different difficulties and dataset sizes. Consequently, simply cycling through all tasks may drastically overtrain smaller tasks leading to overfitting. Typically early-stopping provides a strong defense to this phenomenon; however, stopping a task in multi-task training introduces problems with *catastrophic forgetting* as the base network drifts over time due to other tasks. We introduce an intuitive but effective dynamic stop and go (DSG) mechanism to avoid these problems. We monitor the validation loss  $s_t$  of each task  $t$ , computing it once per task epoch. If performance improvement is less than 0.1% over 2 epochs, we consider it *Converged* and shift it into *stop* mode. In DSG *stop* mode, a task only updates every iter-gap ( $\Delta$ ) iterations. If validation performance degrades by 0.5% from the task’s best measured performance while in *stop* mode, the task is considered *Diverged* and is returned to DSG *go*. This procedure is shown in Algorithm 1.

**Curriculum Learning.** Inspired by prior multi-task literature [4] [31], we experimented with both curriculum and anti-curriculum strategies based on task difficulty. Specifically, for anti-curriculum we first train on the slowest-converging task-group G1 (Vocab-Based VQA) before starting full round-robin multi-task training. Inversely for the curriculum setting we first train on our fastest-

---

#### Algorithm 1: DSG for Multi-Task Learning

---

```

 $n_t \leftarrow$  number of iterations per epoch for task  $t$ 
 $\Delta \leftarrow$  size of gap between iterations in stop mode
 $DSG_t \leftarrow go$ 
for  $i \leftarrow 1$  to  $MaxIter$  :
  for  $t \in Tasks$  :
    if  $DSG_t = go$  or ( $DSG_t = stop$  and  $i \bmod \Delta = 0$ ) :
      Compute task loss  $L_t(\theta)$  and gradient  $\nabla_t(\theta)$ 
      Update  $\theta \leftarrow \theta - \epsilon \nabla_t(\theta)$ , where  $\theta = \theta_s \cup \theta_t$ 
    if  $i \bmod n_t = 0$  :
      Compute validation score  $s_t$  on task  $t$ 
      if  $DSG_t = go$  and Converged( $s_t$ ) :
         $DSG_t \leftarrow stop$ 
      else if  $DSG_t = stop$  and Diverged( $s_t$ ) :
         $DSG_t \leftarrow go$ 
  end
end

```

---

converging task-group G3 (Referring Expressions). Different from previous observation [31, 33], we found that using no curriculum leads to superior performance when combined with other strategies proposed in this section.

**Setting Multi-Task Hyperparameters.** We follow a simple design philosophy – identify simple heuristics based on hyper-parameters tuned for each task in single-task training. This significantly reduces the burden of searching for joint-training hyper-parameters. See the supplement for a full list of per task learning rates, batch sizes, and other settings. Our code has been made available<sup>1</sup>.

**Batch Size:** For multi-task, we keep the batch size tuned for single-task training for each task.

**Warm-up Duration:** We found it important to set warm-up duration relative to the largest dataset. Specifically, we run linear warm-up over  $\eta * N$  iterations where  $N$  is the max. number of iterations taken to train any dataset in the single-task setting. We observe significant performance degradation for harder tasks when warm-up was shorter. We set  $\eta$  to 0.1 for our experiments.

**Loss Scaling:** Our model has shared and task-specific parameters and we found it important to maintain separate learning rates. For the shared base model, we set the the base learning rate to the minimum over all single-task dataset parameters. To accommodate variable learning rates for each dataset, we scale the task loss for each dataset by the ratio of task target learning rate over base learning rate.

## 4. Experiments and Results

### 4.1. Single-Task Performance

To establish baseline performance for the ViLBERT architecture that forms the backbone of our multi-task experiments, we first train single-task models on top of the base ViLBERT architecture (Section 3) for each of our 12 datasets. Rows 1 and 2 in Table 2 show the performance of these models trained on the full and cleaned

<sup>1</sup><https://github.com/facebookresearch/vilbert-multi-task>



		Vocab-based VQA (G1)			Image Retrieval (G2)		Referring Expression (G3)					Verification (G4)		# params (# models)	All Tasks Average	
		VQAv2	GQA	VG QA	COCO	Flickr30k	COCO	COCO+	COCog	V7W	GW	NLVR <sup>2</sup>	SNLI-VE			
	Clean	test-dev	test-dev	val	test(R1)	test(R1)	test	test	test	test	test	testP	test			
1	Single-Task (ST)	71.82	58.19	34.38	65.28	61.14	78.63	71.11	72.24	80.51	62.81	74.25	76.72	3B (12)	67.25	
2	Single-Task (ST)	✓	71.24	59.09	34.10	64.80	61.46	78.17	69.47	72.21	80.51	62.53	74.25	76.53	3B (12)	67.03
3	Group-Tasks (GT)	✓	72.03	59.60	36.18	65.06	66.00	80.23	72.79	75.30	81.54	64.78	74.62	76.52	1B (4)	68.72
4	All-Tasks (AT)	✓	72.57	60.12	36.36	63.70	63.52	80.58	73.25	75.96	82.75	65.04	78.44	76.78	<b>270M (1)</b>	69.08
5	All-Tasks <sub>w/o</sub> G4	✓	72.68	62.09	36.74	64.88	64.62	80.76	73.60	75.80	83.03	65.41	-	-	266M (1)	-
6	GT $\xrightarrow{\text{finetune}}$ ST	✓	72.61	59.96	35.81	66.26	66.98	79.94	72.12	75.18	81.57	64.56	74.47	76.34	3B (12)	68.81
7	AT $\xrightarrow{\text{finetune}}$ ST	✓	72.92	60.48	36.56	65.46	65.14	80.86	73.45	76.00	83.01	65.15	<b>78.87</b>	76.73	3B (12)	69.55
8	AT $\xrightarrow{\text{finetune}}$ ST		<b>73.15</b>	<b>60.65</b>	<b>36.64</b>	<b>68.00</b>	<b>67.90</b>	<b>81.20</b>	<b>74.22</b>	<b>76.35</b>	<b>83.35</b>	<b>65.69</b>	<b>78.87</b>	<b>76.95</b>	3B (12)	<b>70.24</b>

**Table 2:** Comparison of our multi-task models to single-task performance. We find multi-task training (rows 3–5) provides significant gains over single-task training (rows 1–2) while reducing the parameter count from over 3 billion to 270 million. Further, following multi-task training by task-specific fine-tuning (rows 6–9) further gains can be made at the cost of increased parameters.

		Trained With					Trained With						
Relative PERF		G1	G2	G3	G4	Avg.	G1 & G2	G1 & G3	G1 & G4	G2 & G3	G2 & G4	G3 & G4	Avg.
	G1 (VQAv2)	-	0.38%	0.38%	-0.20%	0.19%	-	-	-	0.63%	-0.08%	0.18%	0.24%
	G2 (Flickr30k)	0.46%	-	0.23%	-4.13%	-1.15%	-	1.24%	0.49%	-	-	-4.36%	-0.88%
	G3 (Visual7W)	0.39%	0.78%	-	0.24%	0.47%	0.86%	-	0.19%	-	0.29%	-	0.44%
	G4 (NLVR <sup>2</sup> )	2.29%	1.47%	0.67%	-	1.48%	3.69%	3.22%	-	2.73%	-	-	3.21%
	Avg.	1.04%	0.88%	0.43%	-1.36%	-	2.27%	2.23%	0.34%	1.68%	0.10%	-2.09%	-

**Table 3:** Pair-wise (left) and triple-wise (right) inter-group representative task analysis. Each entry is the relative performance change from single-task training for the row-task when jointly trained with the column-task(s).

datasets, respectively. As expected, reducing the training set size through cleaning results in lower performance in most cases. Our improvements over the pretraining objective (Sec 3.1) results in better downstream tasks performance (71.82 vs. 70.55 on VQA and 61.46 vs. 58.20 on Flickr30k Recall@1). See the supplementary for full comparison. Overall, our base architecture is competitive with prior work and a good starting point for multi-task learning.

## 4.2. Intra-Group Multi-task Performance

We begin with the most intuitive multi-task setting – jointly training tasks within the same groups. As grouped tasks are typically highly related, this is akin to some existing data augmentation practices (*e.g.* adding Visual Genome (VG) QA data when training VQA). Note this corresponds to four separate multi-task models – one for each group.

Table 2 row 3 shows the result of intra-group multi-task training. Comparing with single-task models trained on the same data (row 2), we see meaningful improvements of between 0.37% (NLVR<sup>2</sup>) and 4.54% (Flickr30k retrieval) points for 11 out of 12 tasks (only SNLI-VE did not improve). Comparing to row 1, we see that intra-group multi-task training overcomes the data-loss from cleaning with an average score of 68.72, outperforming the single-task models trained on the full datasets which have an average score of 67.25. Further, the total number of parameters drops by a factor of 3× – going from 12 full models to only 4.

## 4.3. Inter-Group Multi-task Performance

**Representative Task Analysis.** We next consider the interplay between different task-groups. For efficiency, we consider multi-task training with representative tasks from each group – specifically VQA (G1), Retrieval Flickr30k (G2), Visual7W (G3), and NLVR<sup>2</sup> (G4). These were selected to maximize diversity in underlying image sources. We examine their relationships by jointly training all pairs and triplets of tasks under our multi-task training approach.

Table 3 (left) shows the results of training each representative task pair. Each entry is the percent change from single-task performance for the row-task when jointly trained with the column-task. As such, the Avg. row (bottom) shows the mean impact each column-task has on other tasks, and likewise the Avg. column (right) shows the mean impact other tasks have on each row-task. For instance, we find that adding VQA (G1) benefits other tasks with an average improvement of +1.04%. Interestingly, adding NLVR<sup>2</sup> (G4) degrades other tasks on average (-1.36%) while making significant gains itself (+1.48%). This is primarily due to a -4.13% interaction with G2. Table 3 (right) shows all task triplets. Gains in the paired-experiments are not simply additive. In the pair-wise analysis, G3 gained +0.39% and +0.78% from G1 and G2 respectively. As before, G4 has some strong negative effects on other groups (-4.36% G2 with G3 & G4) but these effects can be regulated by other tasks (+0.49% G2 with G1 & G4).

Task	Split	SOTA	UNITER [8]		Ours <sub>AT</sub>	Ours <sub>AT→ST</sub>
			BERT <sub>B</sub>	BERT <sub>L</sub>	BERT <sub>B</sub>	BERT <sub>B</sub>
VQA	test-dev	-	72.27	<b>73.24</b>	72.57	73.15
VG QA	val	-	-	-	36.36	<b>36.64</b>
GQA	test-dev	60.00 [45]	-	-	60.12	<b>60.65</b>
IR COCO	test (R1)	<b>68.50</b> [23]	-	-	63.70	68.00
IR Flickr30k	test (R1)	-	71.50	<b>73.66</b>	63.52	67.90
RefCOCO	test	-	80.21	80.88	80.58	<b>81.20</b>
RefCOCO+	test	-	72.90	73.73	73.25	<b>74.22</b>
RefCOCog	test	-	74.41	75.77	75.96	<b>76.35</b>
Visual 7W	test	72.53 [16]	-	-	82.75	<b>83.35</b>
GuessWhat	test	61.30 [13]	-	-	65.04	<b>65.69</b>
NLVR <sup>2</sup>	testP	-	77.87	<b>79.50</b>	78.44	78.87
SNLI-VE	test	-	78.02	<b>78.98</b>	76.78	76.95
# params			602M	2.1B	<b>270M</b>	3B
(# models)			(7 x 86M)	(7 x 303M)	<b>(1 x 270M)</b>	(12 x 250M)

**Table 4:** Comparison to recent SOTA. For image retrieval (IR) COCO and Flickr we report R1 scores on the 1K test set.

**Full Multi-task Results.** We move to our main result – a single model trained on all 12 datasets. The results of this All-Tasks (AT) model are shown in Table 2 row 4. This model outperforms independent single-task models trained on the same data (row 2) for 11 out of 12 tasks and improve the average score by 2.05 points (69.08 vs. 67.03). We reiterate for emphasis, average performance *improves* by 2.05 points while *reducing* the number of parameters from over 3 billion to 270 million (a  $12\times$  reduction). This is also true for comparison with single-task models trained on full datasets (row 1) by a similar margin of 1.83 points.

Our AT model also outperforms the Group-Task (GT) models (row 3) despite having 4x fewer parameters (avg. 69.08 vs 68.72). This implies that despite their diversity, tasks across different groups can benefit from joint training.

We observed from the representative task analysis that G4 tends to have a negatively effect other groups during joint training. To validate this observation on all tasks, we train an All-Task model without G4 (row 5). This model achieves higher avg. score of 67.96 for G1+G2+G3 compared to the full AT model’s 67.39. NLVR<sup>2</sup> (G4) presents two images per description and often one matches while the other does not. Despite the alignment with one image, the instance as a whole is negative. We speculate that this supervision may interfere with the standard caption-image alignment objective in Flickr30k.

#### 4.4. Multi-Task Learning as Pretraining

For some applications, single task performance may be paramount and justify storing a task-specific model. Even then, fine-tuning from a multi-task trained model may allow the model to take advantage of the additional, diverse supervision captured during multi-task training. Following [26], we finetune our trained multi-task models (GT and AT) on each downstream task and show results in Table 2. Rows 6 and 7 show that finetuning from the all-task model (AT) outperforms finetuning from the group-task models (GT) with

	VQA	COCO Retrieval			Flickr Retrieval			FG
		R1	R5	R10	R1	R5	R10	R1
OmniNet [36]	55.76	-	-	-	-	-	-	-
HDC [33]	69.28	57.40	88.40	95.60	56.10	82.90	89.40	57.39
Ours	<b>72.70</b>	<b>65.16</b>	<b>91.00</b>	<b>96.20</b>	<b>65.06</b>	<b>88.66</b>	<b>93.52</b>	<b>64.61</b>

**Table 5:** Comparison with other multi-task models. VQA score is on test-dev and the retrieval tasks on their respective 1K test split. For Flickr Grounding (FG) we report R1 on Flickr30K test.

an average score of 69.51 vs. 68.81. For comparison with our multi-task models, these are finetuned on the cleaned datasets which are 11% smaller on average. To compare to prior work, we also finetune on the full dataset for individual tasks (Row 8) and observe further improvements. Recall that our multi-task model was trained on cleaned data so there is no possibility of test leak here. These model outperform single-task models without multi-task pretraining (row 1) by a large margin (70.24 vs. 67.25 avg. score).

#### 4.5. Comparison with Existing Work

In Table 4 we compare with existing state-of-the-art. We draw special comparison with the recent UNITER [8] architecture as it is similar to our base ViLBERT model. Like ViLBERT, UNITER is a general BERT-based vision-and-language architecture pretrained through self-supervised tasks and then finetuned for each downstream task. We show two UNITER columns corresponding to their underlying BERT model – either Base B or Large L. Our ViLBERT model uses the smaller BERT<sub>B</sub>. Our single all-task model (Ours<sub>AT</sub>) achieves competitive performance to state-of-the-art task-specific models. Our single-task finetuned models (Ours<sub>AT→ST</sub>) surpass state-of-the-art on 7 out of 12 tasks.

Table 5 compares our method with other recently proposed multi-modal, multi-task learning approaches – OmniNet [36] and Hierarchical Dense Co-Attention (HDC) [33]. OmniNet is trained on part-of-speech tagging, image captioning, visual question answering, and video activity recognition, while HDC is trained on image caption retrieval, visual question answering, and visual grounding. We train a multi-task model on the same tasks and cleaned datasets used in HDC [33]. Flickr Grounding is a new task that we include for this comparison. Our multi-task model outperforms these approaches by a large margin.

### 5. Analysis and Ablation Study

**Ablations on task token and training strategies.** To verify our design choices, we perform ablations for different task token granularity and multi-task training strategies. The results are shown in Table 6. We report average group and overall average performance. Detailed breakdown for each task can be found in supplement.

For task tokens, our default setting is with a different

	Task Token	Dynamic Stop-and-Go	G1	G2	G3	G4	All Tasks Average
AT (our)							
1 token per dataset	✓	✓	<b>56.35</b>	<b>63.61</b>	<b>75.52</b>	<b>77.61</b>	<b>69.08</b>
2 token per head	✓	✓	55.95	61.48	75.35	77.37	68.52
3 w/o task token		✓	55.67	62.55	75.38	76.73	68.53
4 w/o DSG	✓		55.50	62.92	75.24	76.31	68.52
5 w/ curriculum			54.68	61.21	75.19	76.70	67.24
6 w/ anti-curriculum			55.82	59.58	73.69	75.94	67.98
7 vanilla multitask			54.09	61.45	75.28	76.71	67.92

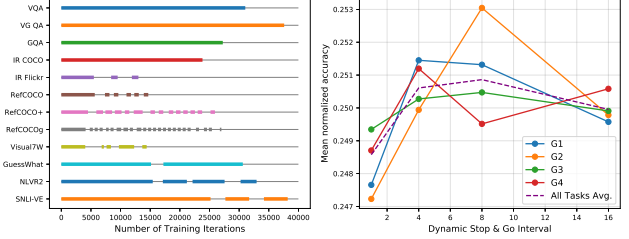
**Table 6:** Ablations on our design choices and comparison to curriculum and anti-curriculum learning multi-task approaches.

task token per dataset (12 total, Row 1). We compare this with two ablations: one task token per output head (4 total, Row 2) and no task tokens (Row 3). We observe that task-specific tokens lead to better performance compared to head-based tokens (avg. 69.08 vs. 68.52) and no task tokens (avg. 69.08 vs. 68.53). This shows that task-aware feature embedding is useful even within the same output space; *e.g.* per-task tokens may help differentiate noun phrases and pointing questions in Referring Expression.

For multi-task training schedule, we compare our dynamic stop-and-go (DSG) (Row 3) with Curriculum (Row 5) and Anti-Curriculum (Row 6) approaches discussed in Sec. 3. We consider convergence rate as a measure of task difficulty. For Curriculum, we first train tasks in G4 and then train all tasks together (easier  $\rightarrow$  harder). For Anti-Curriculum, we train G1 tasks first and then train on all tasks together (harder  $\rightarrow$  easier). Table 6 shows our dynamic stop-and-go training schedule outperforms anti-curriculum (avg. 68.52 vs. 67.98) and curriculum (avg. 68.53 vs. 67.24). Row 7 shows results of a ‘vanilla’, round-robin training scheme with no task tokens or training scheduling. The average score of vanilla multitask is close to anti-curriculum (67.92 vs. 67.98). Consistent with prior work [31], performance on harder tasks (G1) is worse compared to anti-curriculum. Our full training regime outperforms this significantly (avg. 69.08 vs. 67.92).

**Behavior of Dynamic Stop-and-Go training.** To characterize our dynamic stop-and-go training scheme, we visualize the dynamic training schedule in Fig. 2 (left) – bold lines indicate normal `go` training and thin lines are `stop` states when datasets receive sparser updates at a fixed iteration gap (every 4th iteration here). We see that smaller datasets quickly converge and enter `stop` state training early. As the base model drifts over time, they periodically return to full `go` state training to adjust. Interestingly, after some cycles of this, they enter the `stop` state and continue with only sparse updates for the rest of training.

Another aspect of dynamic stop-and-go training is the sparsity of updates in the `stop` state. Fig. 2 (right) shows the mean normalized accuracy for each group for multi-task models trained with different iteration gaps ( $\Delta$ ). We observe that raising  $\Delta$  (*i.e.* updating more sparsely) improves performance initially but degrades for larger values. Abso-



**Figure 2:** Left: Visualization of Dynamic stop-and-go during multi-task training. Solid line indicates in the `go` mode while thin line indicates `stop` mode. Right: Mean accuracy (normalized group-wise for easier comparison) for each group with different iter-gap  $\Delta$  for Dynamic stop-and-go.

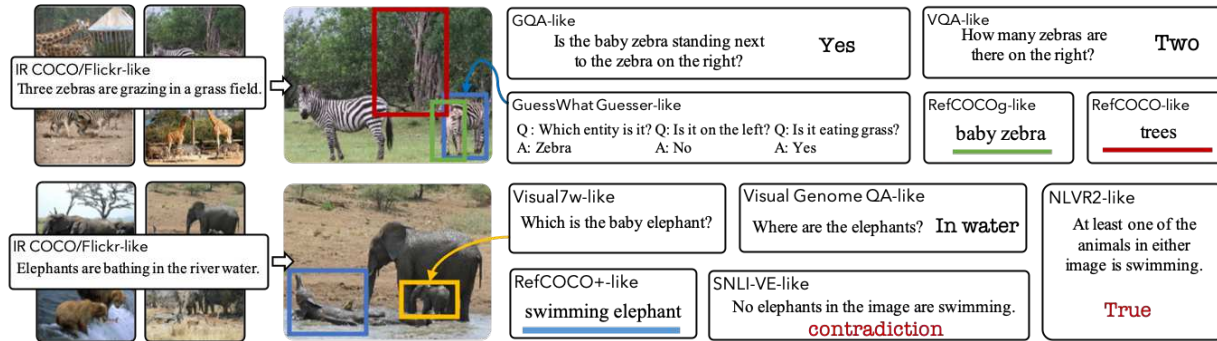
lute and per-task scores are provided in the supplement.

**Multi-Task visual grounding consistency.** Given the common shared base model, one question is whether multi-task models exhibit more consistent visual groundings than independent task-specific models. For example, does a model that correctly answers “What color is the largest dog?” also correctly ground the referring expression “largest dog”? To assess this, we consider 1500 images from the RefCOCO/+ test sets that also have VQA annotations such that for each image  $I_i$  there are associated questions  $\{q^{(i)}\}$  and referring expressions  $\{r^{(i)}\}$ . To measure the overlap in visual concepts between a question  $q_j^{(i)}$  and reference  $r_k^{(i)}$ , we count overlapping nouns and adjectives (identified using a part-of-speech tagger [47]) and denote this  $d(q_j^{(i)}, r_k^{(i)})$ . Armed with this notion of similarity, we consider each question-reference pair for each image (total 111,275 combinations) and compute a weighted accuracy. A pair is considered correct if the question was answered correctly and the referent was localized. Each pair is weighed by their overlap  $d(q_j^{(i)}, r_k^{(i)})$ . Note that if  $q_j^{(i)}$  and  $r_k^{(i)}$  do not have any common visual concept ( $d(q_j^{(i)}, r_k^{(i)}) = 0$ ), the correctness of this pair does not affect the overall metric.

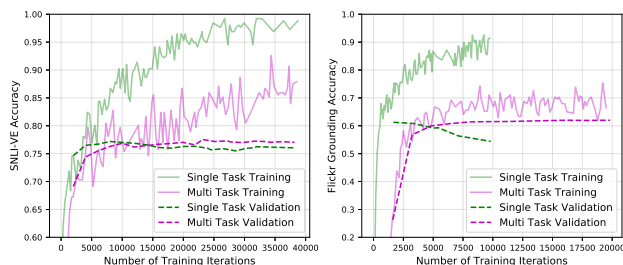
We evaluate our Single-Task (ST), All-Task (AT), and finetuned from All-Task (AT $\rightarrow$ ST) models on the proposed metric. AT consistently outperforms ST (55.40 % vs. 58.30%) and AT $\rightarrow$ ST achieves the best performance (64.64%). This shows our model trained on multiple tasks achieve better visual grounding consistency across different tasks. Further analysis can be found in the supplement.

**Regularizing effects of multi-task learning.** We find multi-task training to have a regularizing effect on tasks which overfit when trained separately. In Fig. 4 we plot the training and validation curves for two tasks (SNLI-VE and Flickr Grounding) where single task training overfits quickly. On the other hand when trained in a multi-task setup with all other tasks, the validation score improves and there is no overfitting.

**Qualitative examples.** Figure 3 shows example outputs of



**Figure 3:** Our single model (Our<sub>AT</sub>) can perform a multitude of V&L tasks: caption and image retrieval, question answering, grounding phrases, guessing image regions based on a dialog, verifying facts about a pair of images, natural language inferences from an image, etc. Here we show outputs of our model for a variety of inputs (that mimic tasks from the 12 datasets it has been trained on).



**Figure 4:** Multi-Task training acts as a regularizer.

our models. Due to space limitation, we provide extensive visualizations in the supplement.

## 6. Related Work

**Multi-task learning.** There has been substantial interest in multi-task learning [6, 38], *i.e.* training a single model for multiple tasks at once. Advances in multi-task learning have been developed in the context of vision [5, 20, 32, 42, 52, 53], language [10, 25, 26, 31, 37], and robotics [18, 34, 46]. Among them, Standley *et al.* [41] studies how different vision tasks are related to each other. Strezoski *et al.* [42] studies layer-wise task routing for different vision tasks. McCann *et al.* [31] pose ten natural language processing (NLP) tasks as question answering tasks. MT-DNN [26] combines multi-task learning with pretraining [14] to improve the learning of text representations. Despite this progress, it is still challenging to train a single model on many tasks that can outperform or even match their single-task counterparts. To enhance the training scheme, BAM [9] applies knowledge distillation where single-task models teach the multi-task model. Raffel *et al.* [37] explore different sampling strategies for NLP tasks. We focus on multi-task learning for V&L tasks.

**Vision and language.** While we address 12 V&L tasks in Sec. 2.1, we do miss some families of tasks including image and video captioning [7], visual dialog [12], embodied question answering [11] and instruction following [3]. Different from earlier work [16, 22, 28, 29, 50, 51, 55] which de-

sign bespoke architecture for different tasks, recently proposed models for V&L [1, 8, 23, 24, 27, 43, 45, 54] provide a common architecture that can be pretrained using self-supervised losses and adapted to many vision and language tasks. However, these models still require task specific finetuning, which may easily overfit on small dataset. Our single model jointly learns from multiple V&L tasks and achieves competitive performance. Further, multi-task training provides a better visiolinguistic representation for task specific finetuning than self-supervised objectives.

**Multi-task V&L learning.** Recent work [33, 36, 40] also explores multi-task learning in V&L. HDC [33] trains a multi-task network on multiple datasets and uses a hyperparameter search method to determine which layer output should be taken for each task. Our method does not need any hyperparameter search to choose outputs for different tasks and outperforms both [36] and [33]. [40] is a concurrent work that does multi-task training on 12 dialogue datasets (only two with images). Our work differs in that we focus on a variety of vision and language tasks.

## 7. Conclusion

In this work, we develop a training regime and experimental setting for large-scale, multi-modal, multi-task learning. As one part of this, we introduce a novel task scheduling approach to help avoid over- or under-training tasks with differing sizes or difficulties. Using this framework, we explore the relationships between 12 vision-and-language datasets – our single multi-task model outperforms 12 single-task models. We find multi-task training can lead to significant gains over independent task training. Further, we show that multi-task learning is an effective pre-training task for training state-of-the-art single-task models.

**Acknowledgement.** The GaTech effort was supported in part by NSF, AFRL, DARPA, ONR YIPs, ARO PECASE, Amazon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the U.S. Government, or any sponsor.



## References

- [1] Chris Alberti, Jeffrey Ling, Michael Collins, and David Reitter. Fusion of detected objects in text for visual question answering. *arXiv preprint arXiv:1908.05054*, 2019. 1, 2, 8
- [2] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. Bottom-up and top-down attention for image captioning and visual question answering. In *CVPR*, pages 6077–6086, 2018. 3
- [3] Peter Anderson, Qi Wu, Damien Teney, Jake Bruce, Mark Johnson, Niko Sünderhauf, Ian Reid, Stephen Gould, and Anton van den Hengel. Vision-and-language navigation: Interpreting visually-grounded navigation instructions in real environments. In *CVPR*, 2018. 8
- [4] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th annual international conference on machine learning*, pages 41–48. ACM, 2009. 4
- [5] Felix JS Bragman, Ryutaro Tanno, Sebastien Ourselin, Daniel C Alexander, and Jorge Cardoso. Stochastic filter groups for multi-task cnns: Learning specialist and generalist convolution kernels. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1385–1394, 2019. 8
- [6] Rich Caruana. Multitask learning. *Machine learning*, 28(1):41–75, 1997. 8
- [7] 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. *CoRR*, abs/1504.00325, 2015. 2, 8
- [8] Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. Uniter: Learning universal image-text representations. *arXiv preprint arXiv:1909.11740*, 2019. 3, 6, 8
- [9] Kevin Clark, Minh-Thang Luong, Urvashi Khandelwal, Christopher D Manning, and Quoc V Le. Bam! born-again multi-task networks for natural language understanding. *arXiv preprint arXiv:1907.04829*, 2019. 8
- [10] Ronan Collobert and Jason Weston. A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th international conference on Machine learning*, pages 160–167. ACM, 2008. 8
- [11] Abhishek Das, Samyak Datta, Georgia Gkioxari, Stefan Lee, Devi Parikh, and Dhruv Batra. Embodied Question Answering. In *CVPR*, 2018. 8
- [12] Abhishek Das, Satwik Kottur, Khushi Gupta, Avi Singh, Deshraj Yadav, Jose M. F. Moura, Devi Parikh, and Dhruv Batra. Visual dialog. In *CVPR*, 2017. 8
- [13] Harm De Vries, Florian Strub, Sarath Chandar, Olivier Pietquin, Hugo Larochelle, and Aaron Courville. Guess-what?! visual object discovery through multi-modal dialogue. In *CVPR*, 2017. 2, 6
- [14] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 3, 8
- [15] 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
- [16] Ronghang Hu, Marcus Rohrbach, Jacob Andreas, Trevor Darrell, and Kate Saenko. Modeling relationships in referential expressions with compositional modular networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 6, 8
- [17] Drew A Hudson and Christopher D Manning. Gqa: a new dataset for compositional question answering over real-world images. *arXiv preprint arXiv:1902.09506*, 2019. 2, 3
- [18] Max Jaderberg, Volodymyr Mnih, Wojciech Marian Czarnecki, Tom Schaul, Joel Z Leibo, David Silver, and Koray Kavukcuoglu. Reinforcement learning with unsupervised auxiliary tasks. *arXiv preprint arXiv:1611.05397*, 2016. 8
- [19] Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara Berg. Referitgame: Referring to objects in photographs of natural scenes. In *EMNLP*, 2014. 2
- [20] Iasonas Kokkinos. Ubertnet: Training a universal convolutional neural network for low-, mid-, and high-level vision using diverse datasets and limited memory. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6129–6138, 2017. 8
- [21] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *IJCV*, 123(1):32–73, 2017. 2
- [22] Kuang-Huei Lee, Xi Chen, Gang Hua, Houdong Hu, and Xiaodong He. Stacked cross attention for image-text matching. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 201–216, 2018. 8
- [23] Gen Li, Nan Duan, Yuejian Fang, Daxin Jiang, and Ming Zhou. Unicoder-vl: A universal encoder for vision and language by cross-modal pre-training. *arXiv preprint arXiv:1908.06066*, 2019. 1, 2, 3, 6, 8
- [24] Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. Visualbert: A simple and performant baseline for vision and language. *arXiv preprint arXiv:1908.03557*, 2019. 1, 2, 8
- [25] Xiaodong Liu, Jianfeng Gao, Xiaodong He, Li Deng, Kevin Duh, and Ye-Yi Wang. Representation learning using multi-task deep neural networks for semantic classification and information retrieval. 2015. 8
- [26] Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. Multi-task deep neural networks for natural language understanding. *arXiv preprint arXiv:1901.11504*, 2019. 6, 8
- [27] Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *arXiv preprint arXiv:1908.02265*, 2019. 1, 2, 3, 8
- [28] Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. Hierarchical question-image co-attention for visual question answering. In *Advances In Neural Information Processing Systems*, pages 289–297, 2016. 8

- [29] Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. Neural baby talk. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7219–7228, 2018. 8
- [30] 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
- [31] Bryan McCann, Nitish Shirish Keskar, Caiming Xiong, and Richard Socher. The natural language decathlon: Multitask learning as question answering. *arXiv preprint arXiv:1806.08730*, 2018. 4, 7, 8
- [32] Ishan Misra, Abhinav Shrivastava, Abhinav Gupta, and Martial Hebert. Cross-stitch networks for multi-task learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3994–4003, 2016. 8
- [33] Duy-Kien Nguyen and Takayuki Okatani. Multi-task learning of hierarchical vision-language representation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 10492–10501, 2019. 4, 6, 8
- [34] Emilio Parisotto, Jimmy Lei Ba, and Ruslan Salakhutdinov. Actor-mimic: Deep multitask and transfer reinforcement learning. *arXiv preprint arXiv:1511.06342*, 2015. 8
- [35] Bryan A Plummer, Liwei Wang, Chris M Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *ICCV*, 2015. 2
- [36] Subhojeet Pramanik, Priyanka Agrawal, and Aman Hussain. Omninet: A unified architecture for multi-modal multi-task learning. *arXiv preprint arXiv:1907.07804*, 2019. 6, 8
- [37] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*, 2019. 8
- [38] Sebastian Ruder. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098*, 2017. 8
- [39] 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. 3, 4
- [40] Kurt Shuster, Da Ju, Stephen Roller, Emily Dinan, Y-Lan Boureau, and Jason Weston. The dialogue dodecathlon: Open-domain knowledge and image grounded conversational agents, 2019. 8
- [41] Trevor Standley, Amir R Zamir, Dawn Chen, Leonidas Guibas, Jitendra Malik, and Silvio Savarese. Which tasks should be learned together in multi-task learning? *arXiv preprint arXiv:1905.07553*, 2019. 4, 8
- [42] Gjorgji Strezoski, Nanne van Noord, and Marcel Worring. Many task learning with task routing. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1375–1384, 2019. 8
- [43] Weijie Su, Xizhou Zhu, Yue Cao, Bin Li, Lewei Lu, Furu Wei, and Jifeng Dai. Vi-bert: Pre-training of generic visual-linguistic representations. *arXiv preprint arXiv:1908.08530*, 2019. 1, 2, 8
- [44] Alane Suhr, Stephanie Zhou, Ally Zhang, Iris Zhang, Huajun Bai, and Yoav Artzi. A corpus for reasoning about natural language grounded in photographs. In *ACL*, 2019. 2
- [45] Hao Tan and Mohit Bansal. Lxmert: Learning cross-modality encoder representations from transformers. *arXiv preprint arXiv:1908.07490*, 2019. 1, 2, 6, 8
- [46] Yee Teh, Victor Bapst, Wojciech M Czarnecki, John Quan, James Kirkpatrick, Raia Hadsell, Nicolas Heess, and Razvan Pascanu. Distral: Robust multitask reinforcement learning. In *Advances in Neural Information Processing Systems*, pages 4496–4506, 2017. 8
- [47] Kristina Toutanova, Dan Klein, Christopher D Manning, and Yoram Singer. Feature-rich part-of-speech tagging with a cyclic dependency network. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 173–180. Association for computational Linguistics, 2003. 7
- [48] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 3
- [49] Ning Xie, Farley Lai, Derek Doran, and Asim Kadav. Visual entailment task for visually-grounded language learning. *arXiv preprint arXiv:1811.10582*, 2018. 2
- [50] Licheng Yu, Zhe Lin, Xiaohui Shen, Jimei Yang, Xin Lu, Mohit Bansal, and Tamara L Berg. MATTNET: Modular attention network for referring expression comprehension. In *CVPR*, 2018. 3, 8
- [51] Rowan Zellers, Yonatan Bisk, Ali Farhadi, and Yejin Choi. From recognition to cognition: Visual commonsense reasoning. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019. 8
- [52] Tianzhu Zhang, Bernard Ghanem, Si Liu, and Narendra Ahuja. Robust visual tracking via structured multi-task sparse learning. *International journal of computer vision*, 101(2):367–383, 2013. 8
- [53] Zhanpeng Zhang, Ping Luo, Chen Change Loy, and Xiaoou Tang. Facial landmark detection by deep multi-task learning. In *European conference on computer vision*, pages 94–108. Springer, 2014. 8
- [54] Luowei Zhou, Hamid Palangi, Lei Zhang, Houdong Hu, Jason J Corso, and Jianfeng Gao. Unified vision-language pre-training for image captioning and vqa. *arXiv preprint arXiv:1909.11059*, 2019. 1, 2, 8
- [55] Yuke Zhu, Oliver Groth, Michael Bernstein, and Li Fei-Fei. Visual7w: Grounded question answering in images. In *CVPR*, 2016. 2, 8