Unifying Vision-and-Language Tasks via Text Generation

Jaemin Cho[1] **Jie Lei** **Hao Tan** **Mohit Bansal**
UNC Chapel Hill
{jmincho, jielei, haotan, mbansal}@cs.unc.edu

Abstract

Existing methods for vision-and-language learning typically require designing task-specific architectures and objectives for each task. For example, a multi-label answer classifier for visual question answering, a region scorer for referring expression comprehension, and a language decoder for image captioning, etc.To alleviate these hassles, in this work, we propose a unified framework that learns different tasks in a single architecture with the same language modeling objective, i.e., multimodal conditional text generation, where our models learn to generate labels in text based on the visual and textual inputs. On 7 popular vision-and-language benchmarks, including visual question answering, referring expression comprehension, visual commonsense reasoning, most of which have been previously modeled as discriminative tasks, our generative approach (with a single unified architecture) reaches comparable performance to recent taskspecific state-of-the-art vision-and-language models. Moreover, our generative approach shows better generalization ability on questions that have rare answers. Also, we show that our framework allows multi-task learning in a single architecture with a single set of parameters, achieving similar performance to separately optimized single-task models. Our code is publicly available at: https://github.com/j-min/VL-T5

Preprint submitted to Elsevier

1 Introduction

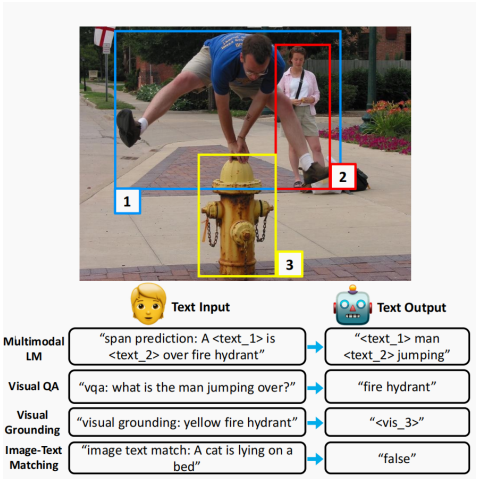


Figure 1: Our unified framework for learning vision-and-language tasks. While existing methods require designing task-specific architectures for different tasks, our framework unifies them together as generating text labels conditioned on multimodal inputs

Recent vision-and-language transformers have made great strides by using a pretraining approach which has led to impressive results in tasks such as visual question answering and referring expression comprehension. Still, these models usually need specific setups for each task, even when they share similar reasoning skills.

To tackle this issue, the authors suggest a unified framework that relies on text generation. They've extended powerful pretrained models like T5 and BART by adding visual abilities

and call these enhanced versions VL-T5 and VL-BART.

This new method allows for multiple tasks through one architecture without needing more parameters. Experiments reveal that this unified approach matches the performance of top models across several benchmarks which also shows better generalization, especially in situations with rare answers.

2 Related Works

Vision-and-Language pretraining: Large-scale language pretraining with transformers ([3]; [4]; [7]) have achieved remarkable success for many natural language understanding tasks([5]) Following this success, image+text pretraining models ([9]) and video+text pretraining models ([8]) have also shown to perform better than previous non-pretraining approaches ([2]) in a wide range of discriminative ([6]) and generative tasks ([1]). In this work, we focus on image+text tasks. While existing image+text models mostly use task-specific architectures and objectives, we seek to design a unified framework across different tasks.

3 Model

We propose a new framework that unifies vision-and language problems as multimodal conditional text generation. We introduce VL-T5 and VL-BART based on two pretrained transformer language models: T5Base and BARTBase. Specifically, we extend their text encoders to multimodal encoders by incorporating image region embeddings as additional input. The overall architecture of our framework is shown in Fig. 2. Since the architecture differences between VL-T5 and VL-BART are minor, we use VL-T5 as an example to illustrate our framework in detail in the rest of this section.

3.1 Visual Embeddings

We represent an input image v with n=36 object regions from a Faster R-CNN trained on Visual Genome. As shown in Fig. 2b, each image region is encoded as a sum of four types of features:

(i) RoI (Region of Interest) object features

(ii) RoI bounding box coordinates

(iii) image IDs $\in \{1, 2\}$

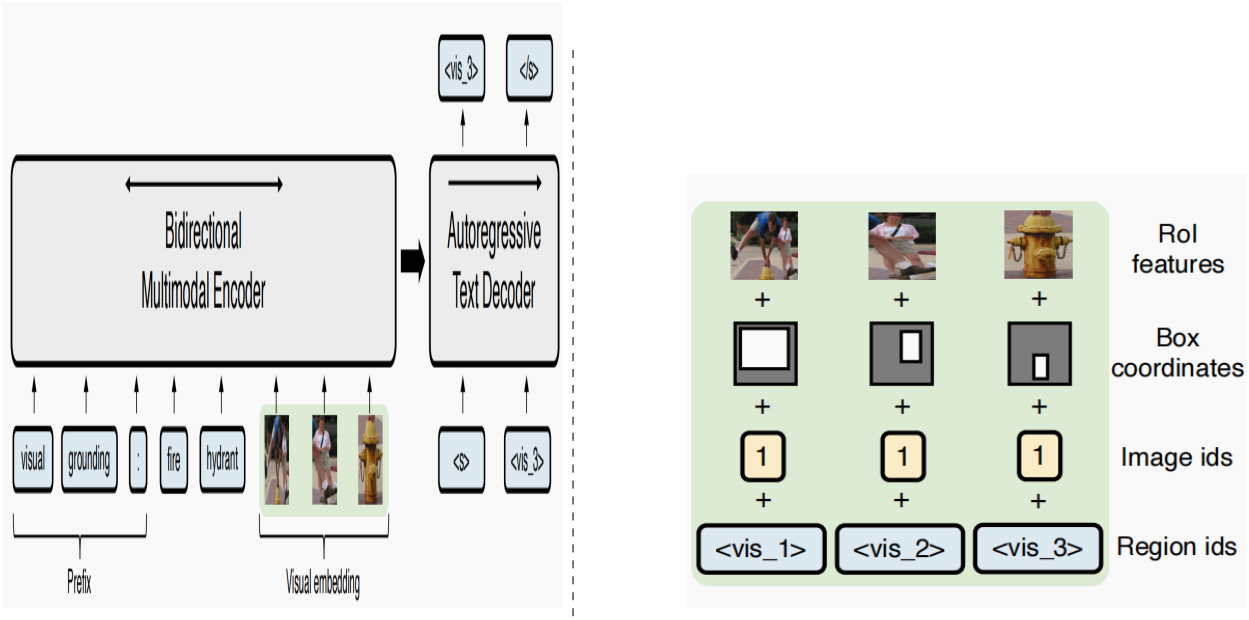
(iv) region IDs $\in \{1, \ldots, n\}$

. RoI features and bounding box coordinates are encoded with a linear layer, while image ids and region ids are encoded with learned embeddings (Devlin et al., 2019). Image ids are used to discriminate regions from different images, and is used when multiple images are given to the model (i.e., in NLVR2 (Suhr et al., 2019), models take two input images). The final visual embeddings are denoted as $e^v = \{e^v_1, \ldots, e^v_n\}$

3.2 Encoder-Decoder Architecture

We use *transformer encoder-decoder architecture* [3] to encode visual and text inputs and generate label text. Our bidirectional multi modal encoder is a stack of m transformer blocks, consisting of a self-attention layer and a fully-connected layer with residual connections. Our decoder is another stack of m transformer blocks similar to the multi modal encoder, where each block has an additional cross-attention layer. As shown in figure 2b, the encoder takes the concatenation of text and visual embedding as input and outputs their contextualized joint representations

$$h = \{h^x_1, \ldots, h^x_{|x|}, h^v_1, \ldots, h^v_n\} = \text{Enc}(e^x, e^v).$$



(a) Our vision-and-language framework

(b) Visual embedding

Figure 2: An illustration of our VL-T5 and VL-BART architectures for visual grounding task. Instead of task-specific architectures, our models use text prefixes to adapt to different tasks. The green block in (a) refers to visual embeddings. (b) shows the components of visual embedding. Note that we reuse the text embeddings of visual sentinel tokens (ex. ¡vis 3¿) as region id embeddings, which allows our models to tackle many discriminative vision-language tasks as text generation, including visual grounding.

Then the decoder iteratively attends to previously generated tokens $y_{<j}$ (via self-attention) and the encoder outputs h (via cross-attention), then predicts the probability of future text tokens

$$P_\theta(y_j|y_{<j}, x, v) = \text{Dec}(y_{<j}, h).$$

We suggest readers to check [4] [1] for more details of our backbone models. For both pre-training (Sec. 4) and downstream tasks (Sec. 5), we train our model parameters θ by minimizing the negative log-likelihood of label text y tokens given input text x and image v:

$$\mathcal{L}_\theta^{GEN} = -\frac{1}{|y|}\sum_{j=1}^{|y|} \log P_\theta(y_j|y_{<j}, x, v) \quad (1)$$

3.3 Task-Specific Methods vs. Our Unified Framework

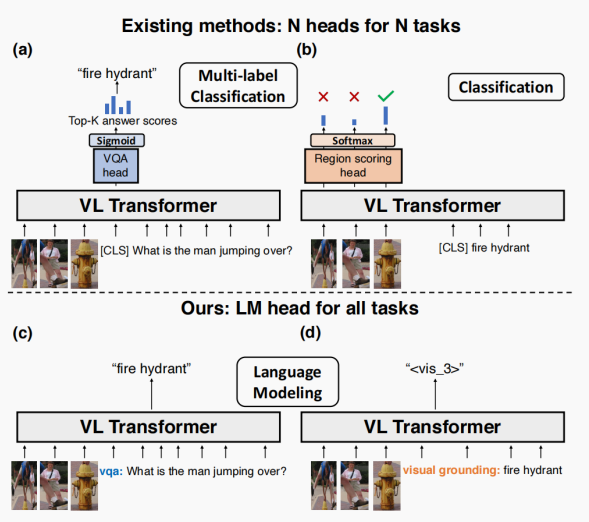


Figure 3: Comparison between existing methods and our frame-work on visual question answering and referring expression comprehension (visual grounding) tasks. While existing methods use task-specific architectures and objectives, our models use the same language modeling architecture and maximum likelihood estimation on label text for all tasks.

We compare our unified framework with existing vision-and-language transformers on two popular tasks: visual question answering ([2]) and referring expression comprehension ([6]).

Visual question answering requires a model to answer a question to a given context image. As shown in figure 3, existing methods ([2]; [1]) typically introduce a multi-layer perception (MLP) multi-label classifier head on top of $h^x_{[cls]}$, which is trained together with the transformer backbone through a binary cross-entropy loss, and weighted with VQA score

$\mathcal{L}^{VQA}_\theta = -$
$\sum_{k=1}^{K} score(a^k, x, v) \log P^{VQA}_\theta(correct|a^k, x, v).$

Referring expression comprehension requires models to localize a target region in an image that is described by a given referring expression. Previous methods tackle this task as multi-class ([1]) or binary ([4]) classification over image regions. For example, UNITER ([1]) introduces an MLP region scoring head on top of the output representations of regions, as shown in figure 3. This region scoring head is jointly trained with the encoder by minimizing negative log-likelihood of target region

$r^* : \mathcal{L}^{REF}_\theta = -\log P^{REF}_\theta(r^*|x, v)$. In contrast to existing methods that develop task-specific architectures and objectives (e.g., the equations above), our unified framework is free from extra model designs for new tasks. As shown in Figure 3 and Table 2, we formulate the task labels to corresponding text, and we learn these different tasks by predicting label text with the same language modeling objective Equation 1.

Method	In-domain	Out-of-domain	Overall
Discriminative			
UNITER$_{\text{Base}}$	74.4	10.0	70.5
VL-T5	70.2	7.1	66.4
VL-BART	69.4	7.0	65.7
Generative			
VL-T5	71.4	13.1	67.9
VL-BART	72.1	13.2	68.6

Table 1: VQA Karpathy-test split accuracy using generative and discriminative methods. We break down the questions into two subsets in terms of whether the best-scoring answer a^* for each question is included in the top-K answer candidates A^{topk}. In domain: $a^* \in A^{topk}$, Out-of-domain: $a^* \notin A^{topk}$

4 Downstream Tasks and Results

In this section, we compare our generative architectures VL-T5 and VL-BART on a diverse set of 7 downstream tasks (details in Appendix) with existing vision-and-language pre-trained transformers.

4.1 Visual Question Answering: VQA and GQA

The visual question answering task requires models to answer a question to a given context image.

Generative vs. Discriminative model: Modern approaches are discriminative models, where they tackle visual question answering tasks as multi-label classification over a predefined set of answer candidates. This strategy achieves strong performance but not generalizes to real-world open-ended scenarios. To quantitatively compare the existing discriminative approaches and our generative approach, we break down VQA questions into in-domain and out-of-domain questions, in terms of whether the best answer a^* for each question is included in the top-K (K=3, 129) answer candidates A^{TopK}. After this split, the in-domain subset contains 24,722 questions, and

Table 2: Input-output formats for pretraining (Sec. 4) and downstream tasks (Sec. 5).

Tasks	Input image	Input text	Target text
Pretraining text (Sec-4)			
Multimodal LM (VL-T5)		span prediction: A $\langle text_1 \rangle$ is $\langle text_2 \rangle$ over a fire hydrant.	$\langle text_1 \rangle$ man $\langle text_2 \rangle$ jumping
Multimodal LM (VL-BART)		denoise: A $\langle mask \rangle$ is $\langle mask \rangle$ over a fire hydrant.	A man is jumping over a fire hydrant
" Visual question answering		vqa: what is the color of the man's shirt?	blue true $\langle vis_3 \rangle$
Image-text matching		image text match: A man with blue shirt is jumping over fire hydrant.	yellow fire hydrant
Visual grounding		visual grounding: yellow fire hydrant	$\langle vis_3 \rangle$
Grounded captioning		caption region: $\langle vis_3 \rangle$	caption¿caption
Downstream tasks (Sec. 5)			
VQA		vqa: [2]	[A]A¿A
GQA		gga: [2]	
[b]NLVR[2]		nlvr: [text]	true/false
VCR Q→A		vcr qa: question [Q] answer: [A]	true/false
VCR QA→R		vcr qar: question [Q] answer: [A] rationale: [R]	true/falseregion id¿region id
RefCOCOg		visual grounding: [referring expression]	caption¿caption
COCO captioning		caption:	caption¿caption
COCO captioning (w/ object tags)		caption with tags: [Tagl Tag2 . . .]	German text¿German text
Multi30K En-De translation		translate English to German: [English text]	

the out-of-domain subset contains 1,558 questions. Table 1 shows the performance. For discriminative baselines, we introduce a sigmoid MLP classifier on top of the decoder representation of the start-of-sequence token $\langle s \rangle$, following **LXMERT** and **UNITER**. Comparing models with the same backbone, we notice that the generative models improve upon the discriminative baselines across all subsets. This improvement is more significant on the out-of-domain subset, where the generative **VL-T5** and **VL-BART** achieve 6 and 6.2 points improvement over their discriminative counterparts, showing the effectiveness of using generative modeling. Compared to the strong discriminative baseline **UNITER$_{\text{Base}}$** (pretrained with 4M extra images), our generative models still show comparable overall performance while significantly outperforming it on the out-of-domain subset (about 3 points).

4.2 Visual Commonsense Reasoning: VCR

Visual Commonsense Reasoning (VCR) ([9]) is a multiple-choice question answering task that requires commonsense reasoning beyond object or action recognition. Each VCR question (Q) has 4 answers (A) and 4 rationales (R), and it can be decomposed into two multiple choice sub-tasks: question answering ($Q \rightarrow A$), and answer justification ($QA \rightarrow A$). The overall task ($Q \rightarrow AR$) requires a model to not only select the correct answer to the question, but also the correct rationale for choosing the answer. Similar to that leverages language model for document ranking, we concatenate context (image+question) with each candidate choice, and let our models generate "true" for the correct choice and generate "false" otherwise, as shown in table 2, During inference, we use $\frac{P(true)}{P(true)+P(false)}$ to rank the choices and select the one with the highest score.

5 Conclusion

In this work, we proposed VL-T5 and VL-BART which tackle vision-and-language tasks with a unified text generation objective. Experiments show VL-T5 and VL-BART can achieve comparable performance with state-of-the-art vision-and-language transformers on di-

verse vision-and-language tasks without handcrafted architectures and objectives. Especially, we demonstrate our generative approach is better suited for open-ended visual question answering. In addition, we also showed it is possible to train seven different tasks simultaneously using a single architecture with single parameters without not losing much performance. It would be an interesting future work to further explore this direction by adding even more tasks.

References

[1] X. Chen, H. Fang, T. Lin, *et al.*, "Microsoft COCO captions: Data collection and evaluation server," *CoRR*, vol. abs/1504.00325, 2015. arXiv: 1504.00325. [Online]. Available: http://arxiv.org/abs/1504.00325.

[2] P. Anderson, X. He, C. Buehler, *et al.*, "Bottom-up and top-down attention for image captioning and VQA," *CoRR*, vol. abs/1707.07998, 2017. arXiv: 1707.07998. [Online]. Available: http://arxiv.org/abs/1707.07998.

[3] A. Vaswani, N. Shazeer, N. Parmar, *et al.*, "Attention is all you need," *CoRR*, vol. abs/1706.03762, 2017. arXiv: 1706.03762. [Online]. Available: http://arxiv.org/abs/1706.03762.

[4] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," *CoRR*, vol. abs/1810.04805, 2018. arXiv: 1810.04805. [Online]. Available: http://arxiv.org/abs/1810.04805.

[5] R. Zellers, Y. Bisk, R. Schwartz, and Y. Choi, "SWAG: A large-scale adversarial dataset for grounded commonsense inference," *CoRR*, vol. abs/1808.05326, 2018. arXiv: 1808.05326. [Online]. Available: http://arxiv.org/abs/1808.05326.

[6] D. A. Hudson and C. D. Manning, "GQA: a new dataset for compositional question answering over real-world images," *CoRR*, vol. abs/1902.09506, 2019. arXiv: 1902.09506. [Online]. Available: http://arxiv.org/abs/1902.09506.

[7] Y. Liu, M. Ott, N. Goyal, *et al.*, "Roberta: A robustly optimized BERT pretraining approach," *CoRR*, vol. abs/1907.11692, 2019. arXiv: 1907.11692. [Online]. Available: http://arxiv.org/abs/1907.11692.

[8] C. Sun, A. Myers, C. Vondrick, K. Murphy, and C. Schmid, "Videobert: A joint model for video and language representation learning," *CoRR*, vol. abs/1904.01766, 2019. arXiv: 1904.01766. [Online]. Available: http://arxiv.org/abs/1904.01766.

[9] Z. Huang, Z. Zeng, B. Liu, D. Fu, and J. Fu, "Pixel-bert: Aligning image pixels with text by deep multi-modal transformers," *CoRR*, vol. abs/2004.00849, 2020. arXiv: 2004.00849. [Online]. Available: https://arxiv.org/abs/2004.00849.