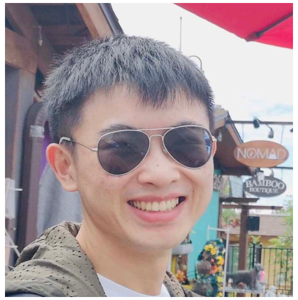
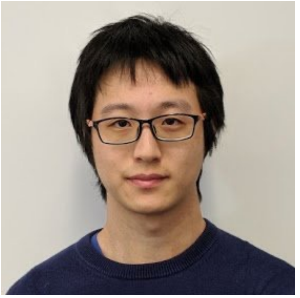
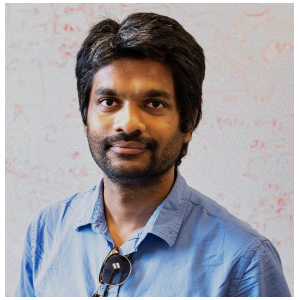
Unifying Vision-and-Language Tasks via Text Generation

Jaemin Cho Jie Lei Hao Tan Mohit Bansal



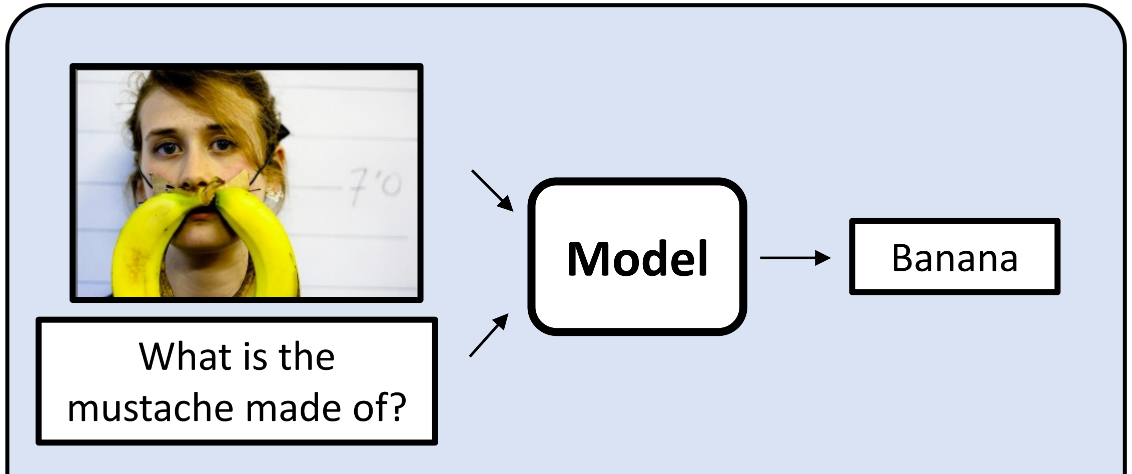




MURGe-Lab @ UNC Chapel Hill

ICML
International Conference On Machine Learning

Vision-and-Language Tasks



Visual Question Answering

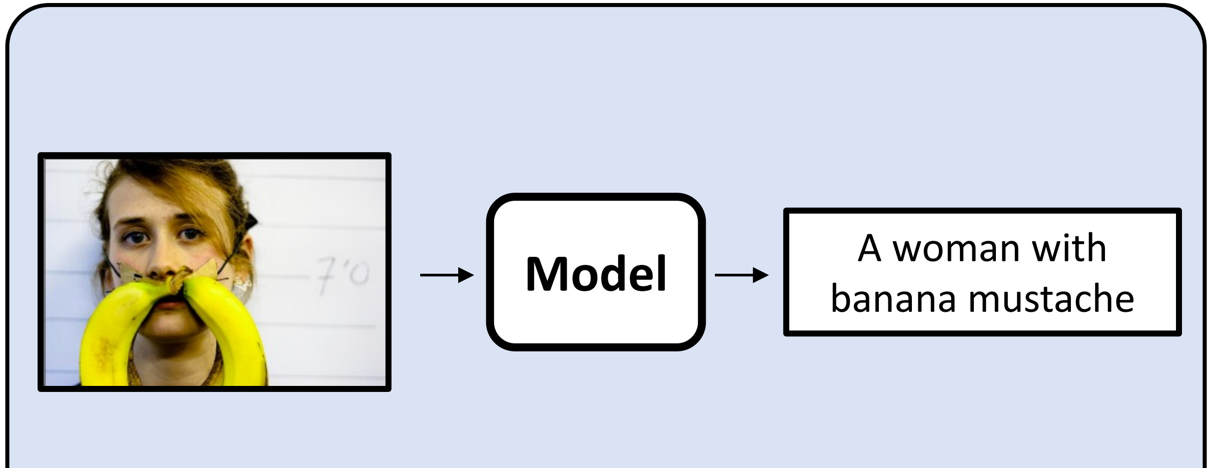
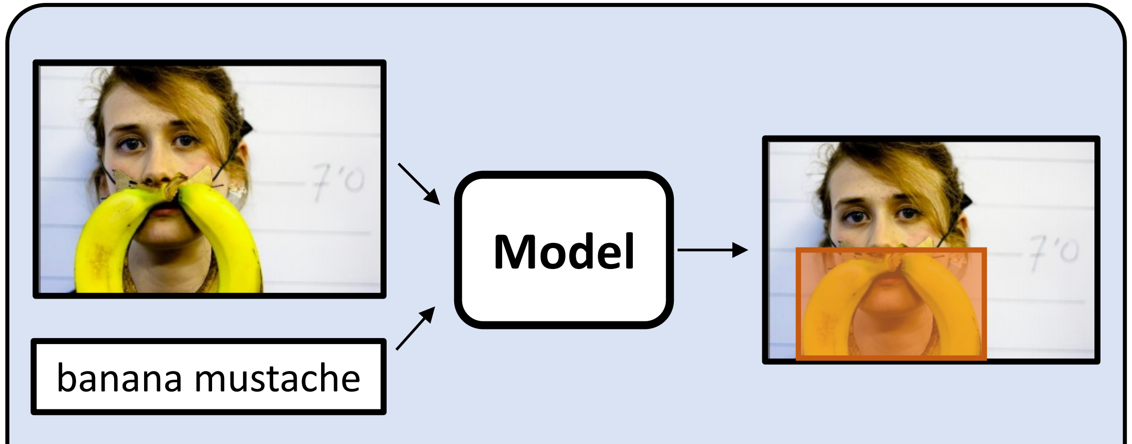
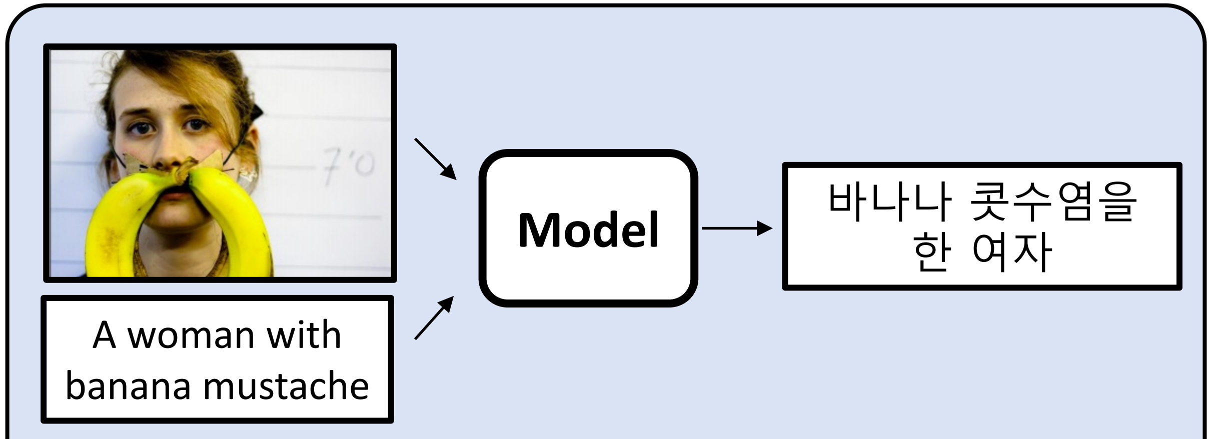


Image Captioning

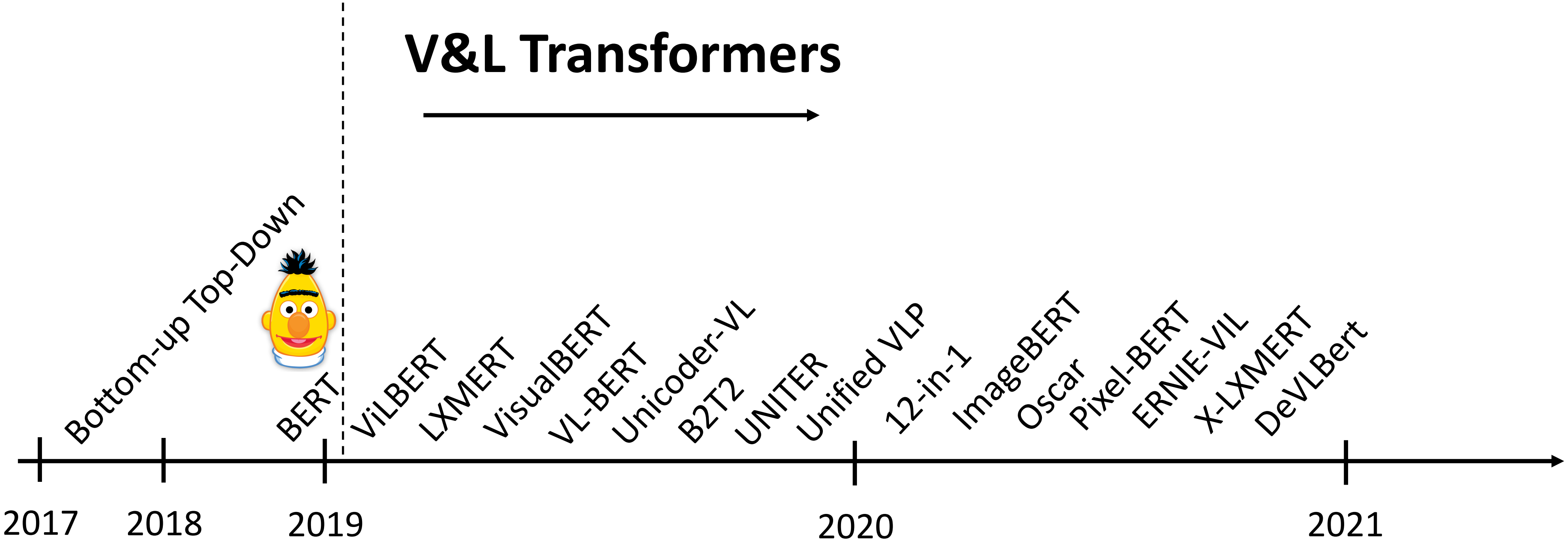


Visual Grounding

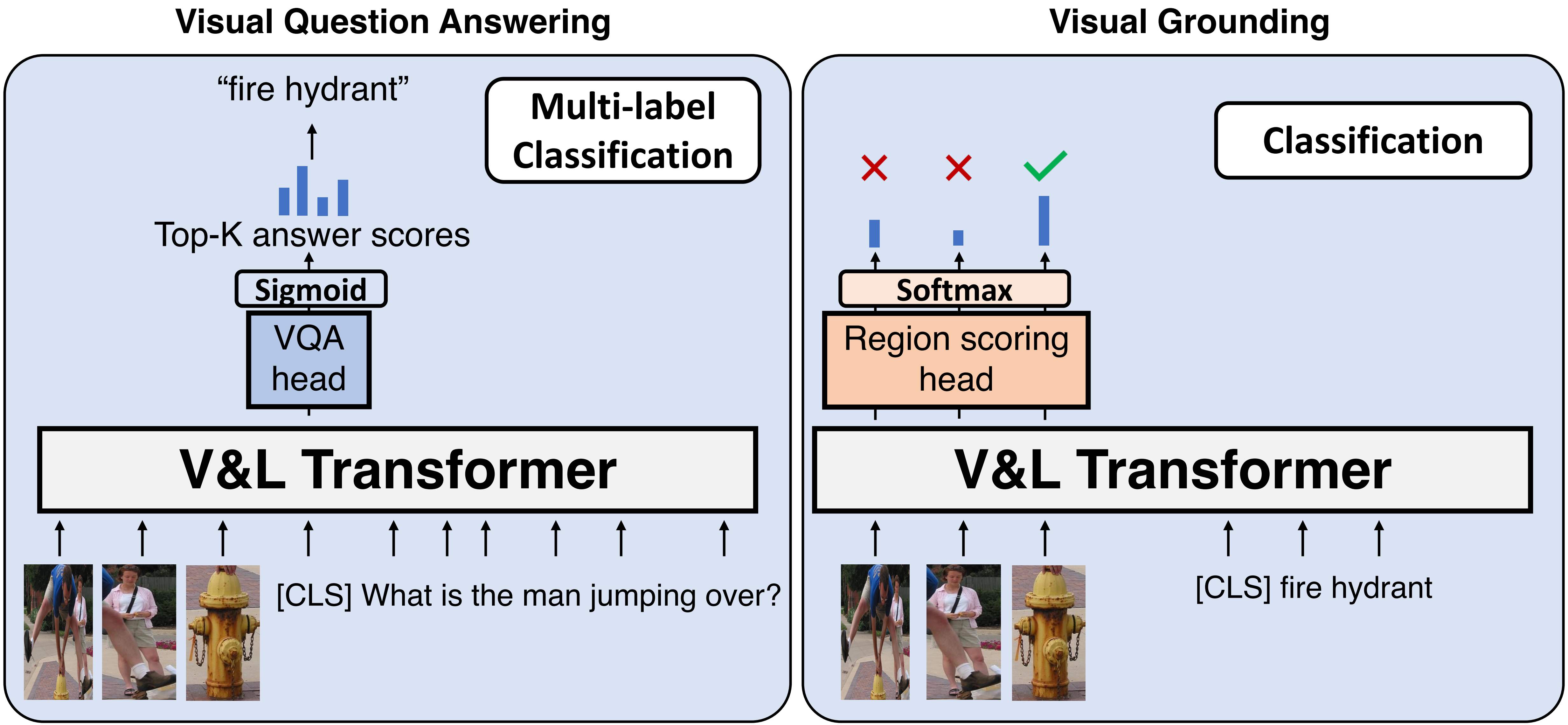


Multimodal Machine Translation (En-Kr)

Vision-and-Language Pretraining



Task-specific Architecture / Objective



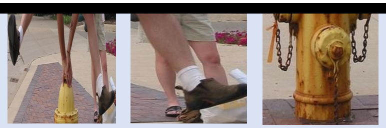
Background | Method | Experiments

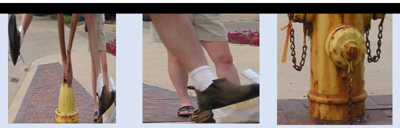
Task-specific Architecture / Objective

Visual Question Answering

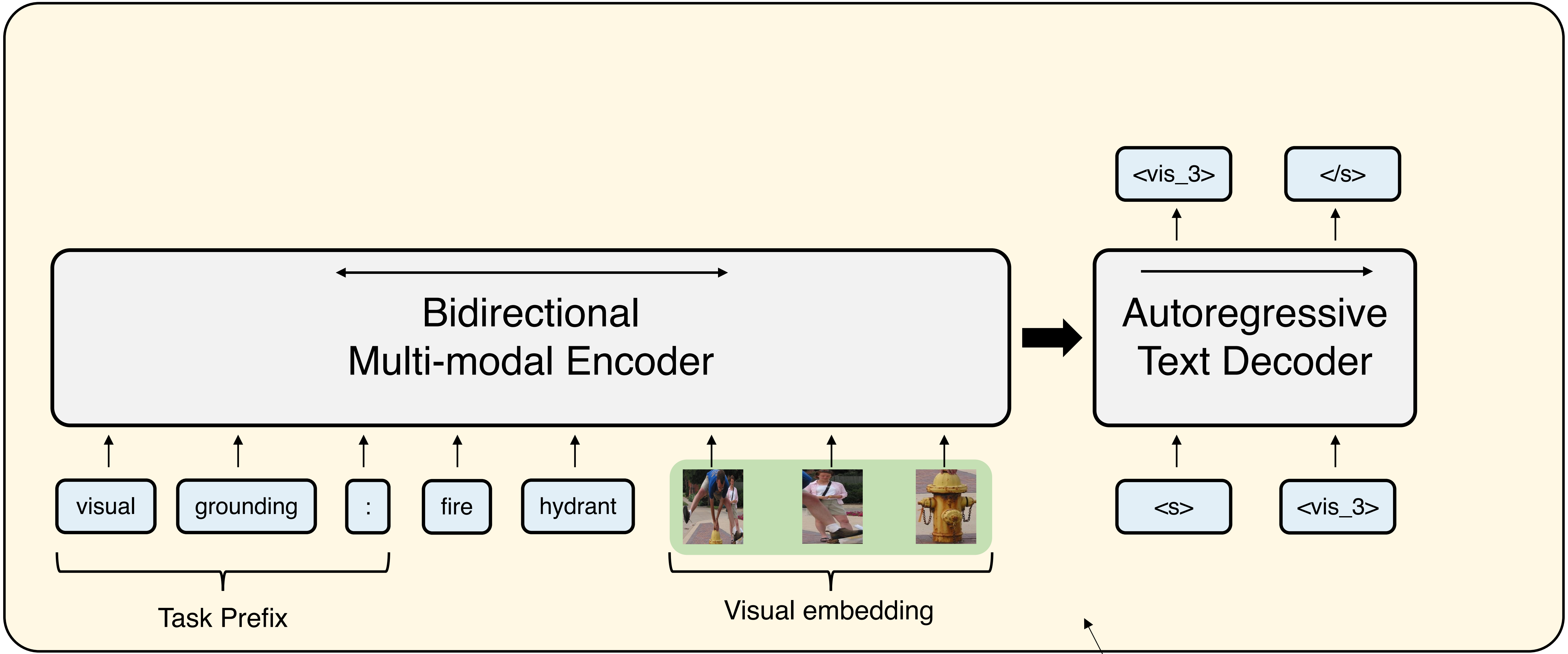
Visual Grounding

Can we tackle all V&L tasks with single objective?



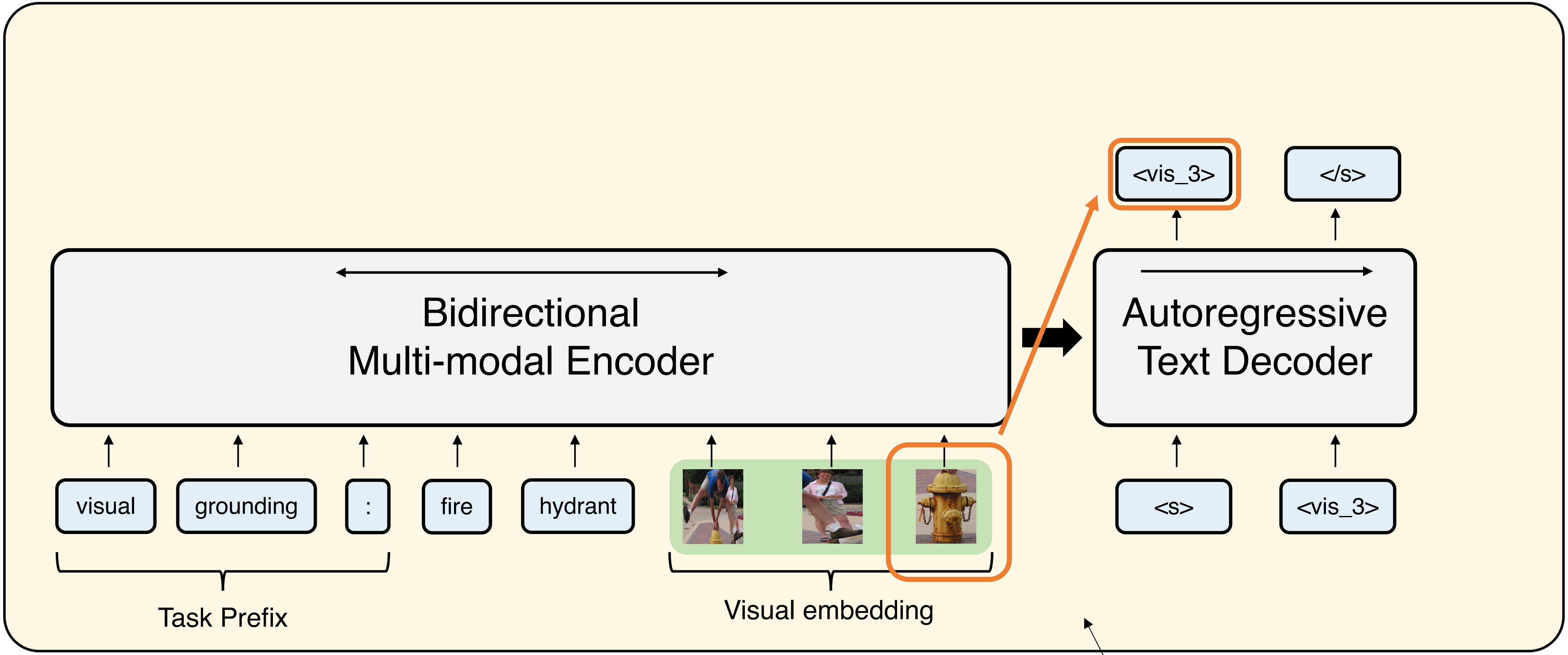


V&L Tasks as Text Generation



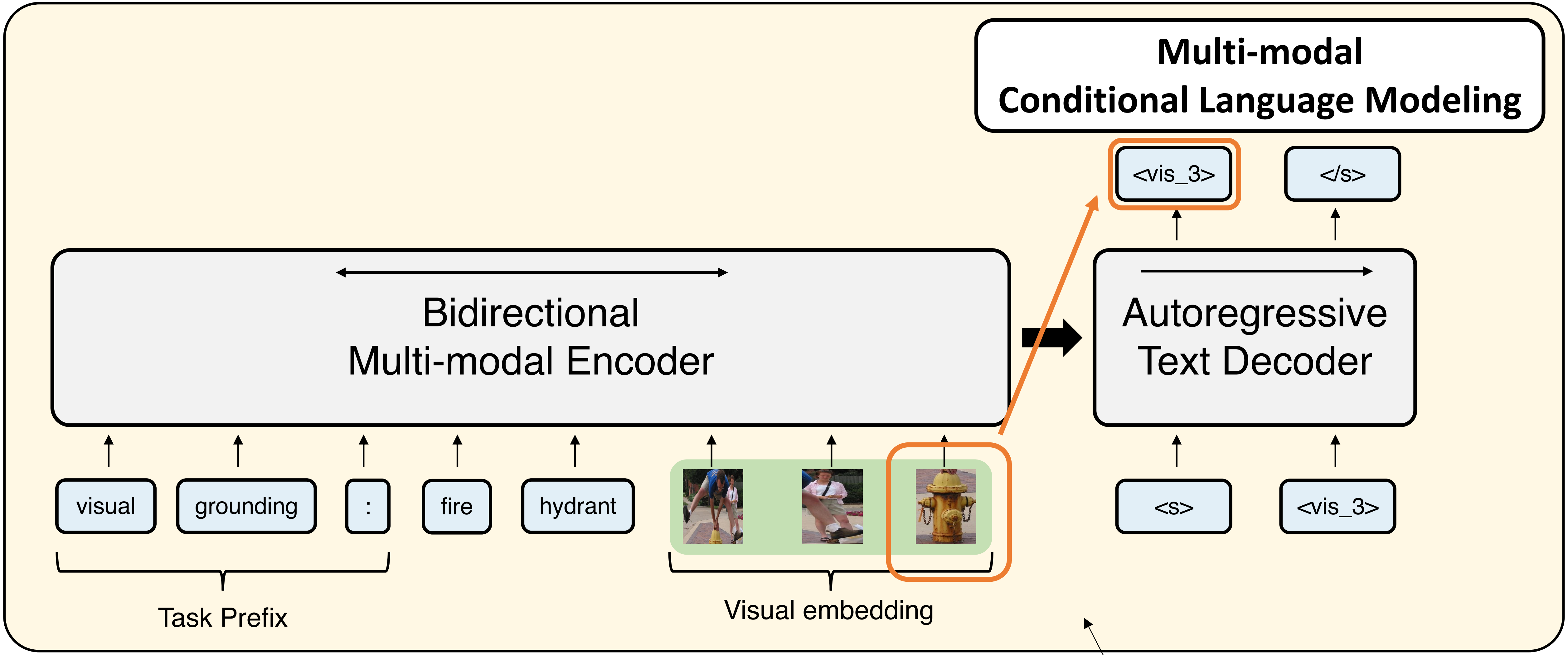
Weights are initialized from off-the-shelf Seq2Seq LMs (e.g., T5)

V&L Tasks as Text Generation



Weights are initialized from off-the-shelf Seq2Seq LMs (e.g., T5)

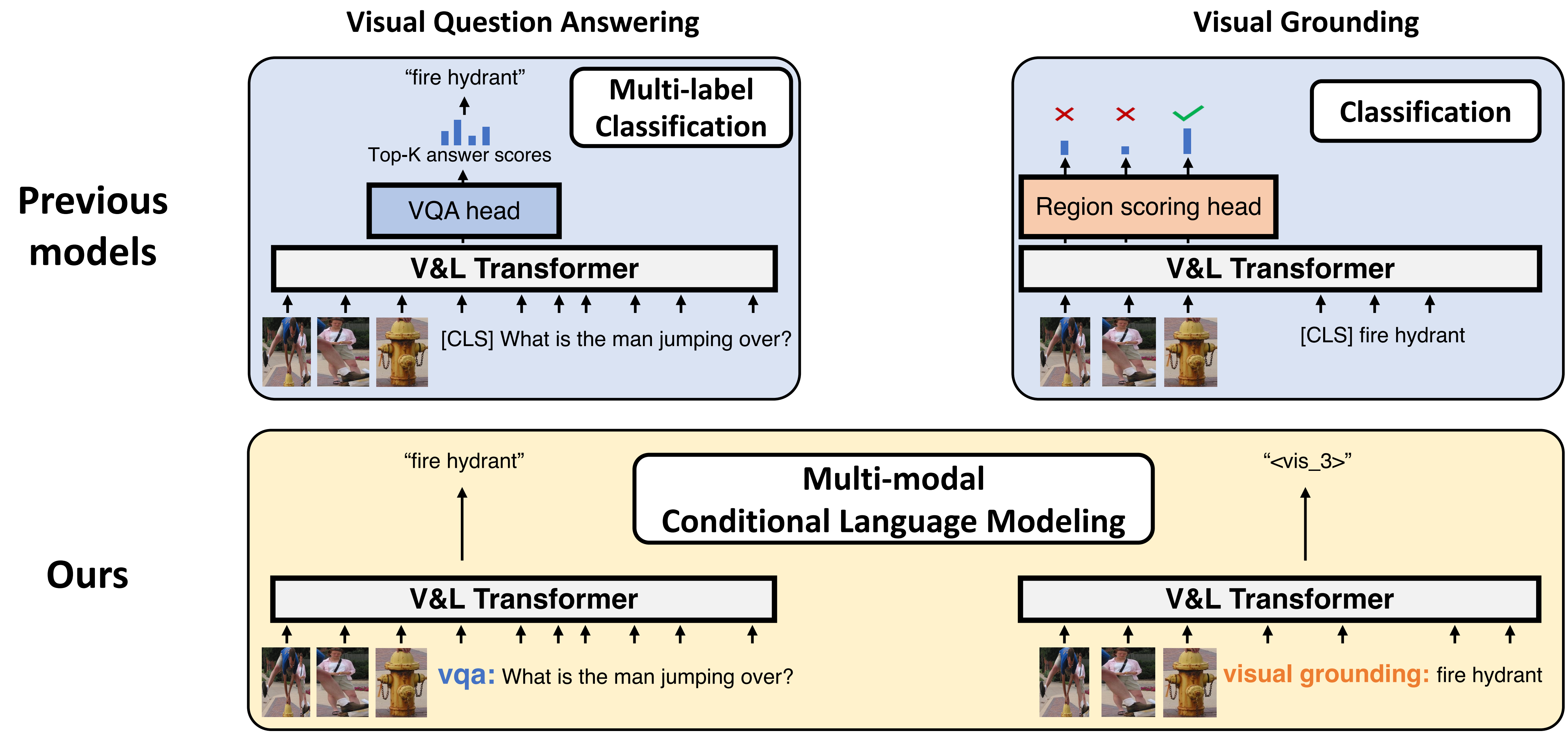
V&L Tasks as Text Generation



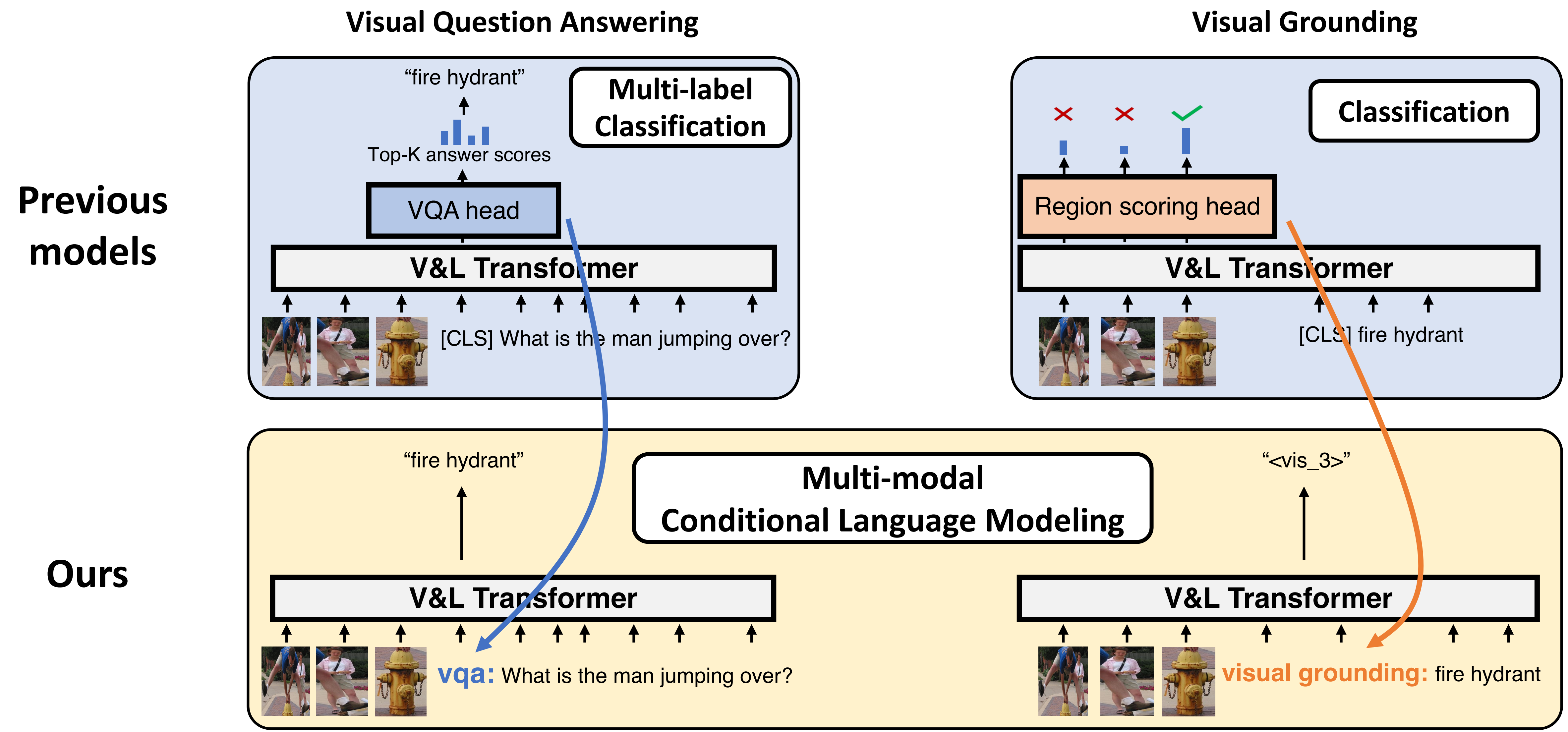
Weights are initialized from off-the-shelf Seq2Seq LMs (e.g., T5)

Background | Method | Experiments

V&L Tasks as Text Generation



V&L Tasks as Text Generation

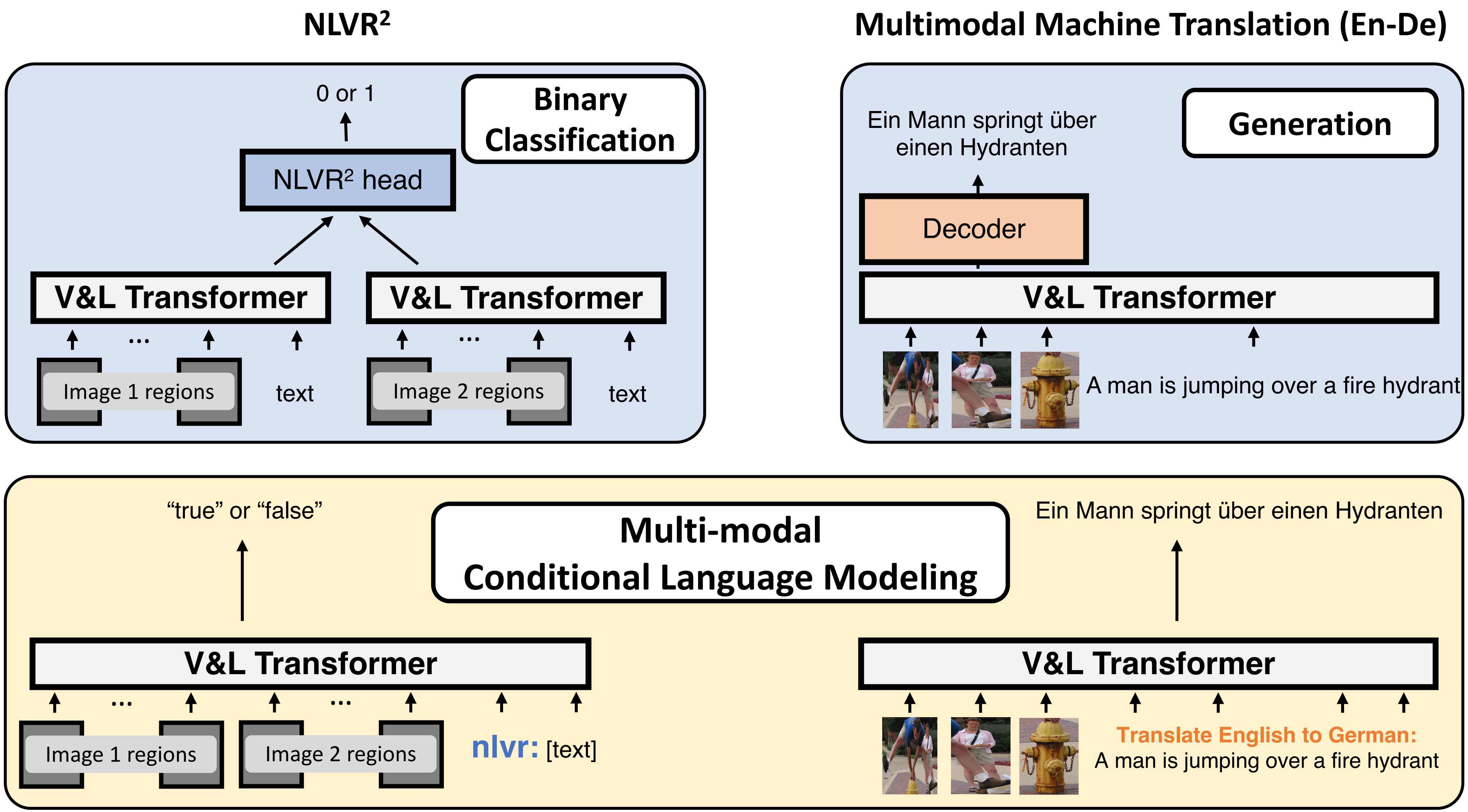


Background | Method | Experiments

V&L Tasks as Text Generation

Previous models

Ours



Comparable to Baselines on Downstream Tasks

Table 2. Single model performance on downstream tasks. Note that the baseline models adopt task-specific objectives and architectures, whereas our models tackle all tasks, including discriminative tasks (e.g., RefCOCOg), as text generation with a single architecture and objective. ⋆ See our discussion in Sec.5.3.

Method	# Pretrain Images	Discriminative tasks					Generative tasks	
		VQA test-std Acc	GQA test-std Acc	NLVR^2 test-P Acc	RefCOCOg test^d Acc	VCR Q→ AR test Acc	COCO Cap Karpathy test CIDEr	Multi30K En-De test 2018 BLEU
LXMERT	180K	72.5	60.3	74.5	-	-	-	-
ViLBERT	3M	70.9	-	-	-	54.8	-	-
UNITER$_{Base}$	4M	72.9	-	77.9	74.5	58.2	-	
Unified VLP	3M	70.7	-	-	-	-	117.7	-
Oscar$_{Base}$	4M	73.4	61.6	78.4	-	-	123.7	-
XGPT	3M	-	-	-	-	-	120.1	-
MeMAD	-	-	-	-	-	-	-	38.5
VL-T5	180K	70.3	60.8	73.6	71.3	58.9	116.5	38.6
VL-BART	180K	71.3	60.5	70.3	22.4*	48.9	116.6	28.1

Our models

Background Method **Experiments**

Comparable to Baselines on Downstream Tasks

Table 2. Single model performance on downstream tasks. Note that the baseline models adopt task-specific objectives and architectures, whereas our models tackle all tasks, including discriminative tasks (e.g., RefCOCOg), as text generation with a single architecture and objective. ⋆ See our discussion in Sec.5.3.

	Method	# Pretrain Images	Discriminative tasks					Generative tasks	
			VQA test-std Acc	GQA test-std Acc	NLVR^2 test-P Acc	RefCOCOg test^d Acc	VCR Q→ AR test Acc	COCO Cap Karpathy test CIDEr	Multi30K En-De test 2018 BLEU
Closest baseline	LXMERT	180K	72.5	60.3	74.5	-	-	-	-
	ViLBERT	3M	70.9	-	-	-	54.8	-	-
	UNITER$_{Base}$	4M	72.9	-	77.9	74.5	58.2	-	
	Unified VLP	3M	70.7	-	-	-	-	117.7	-
	Oscar$_{Base}$	4M	73.4	61.6	78.4	-	-	123.7	-
	XGPT	3M	-	-	-	-	-	120.1	-
	MeMAD	-	-	-	-	-	-	-	38.5
Our models	VL-T5	180K	70.3	60.8	73.6	71.3	58.9	116.5	38.6
	VL-BART	180K	71.3	60.5	70.3	22.4*	48.9	116.6	28.1

Better Generalization on Rare Answers

Table 3. 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*: $a^* \notin A^{topk}$.

Method	In-domain	Out-of-domain	Overall
Discriminative			
$\text{UNITER}_{\text{Base}}$	**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

Better Generalization on Rare Answers

Table 3. 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*: $a^* \notin A^{topk}$.

Method	In-domain	Out-of-domain	Overall
Discriminative			
$\text{UNITER}_{\text{Base}}$	**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

Generative > Discriminative
on same backbone

Better Generalization on Rare Answers

Table 3. 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*: $a^* \notin A^{topk}$.

Method	In-domain	Out-of-domain	Overall
Discriminative			
$\text{UNITER}_{\text{Base}}$	**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

Generative approaches
better generalize on rare answers

Multi-task Learning with Single Set of Parameters

Table 9. Single-task vs. Multi-task finetuning results on 7 tasks. With a single set of parameters, our multi-task model achieves similar performance to separately optimized single-task models. We denote the number of parameters of single VL-T5 model as P.

Method	Finetuning tasks	# Params	Discriminative tasks					Generative tasks	
			VQA Karpathy test Acc	GQA test-dev Acc	$NLVR^2$ test-P Acc	RefCOCOg $test^d$ Acc	VCR val Acc	COCO Caption Karpathy test CIDEr	Multi30K En-De test2018 BLEU
VL-T5	single task	7P	67.9	60.0	73.6	71.3	57.5	116.1	38.6
VL-T5	all tasks	P	67.2	58.9	71.6	69.4	55.3	110.8	37.6

Similar performance with fewer parameters

Thanks!

Code: https://github.com/j-min/VL-T5

Jaemin Cho, Jie Lei, Hao Tan, Mohit Bansal

{jmincho, jielei, haotan, mbansal}@cs.unc.edu