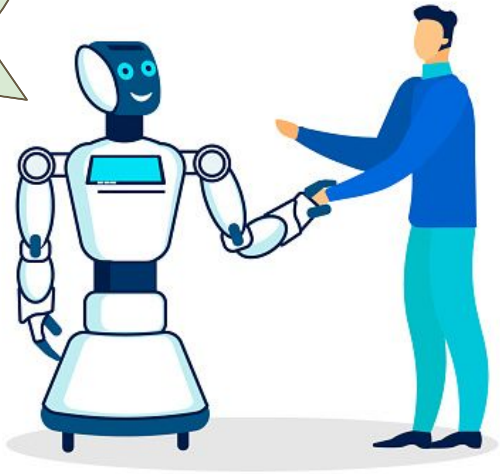
USC Viterbi
School of Engineering

CLiMB 🧗
A Continual Learning Benchmark for Vision-and-Language Tasks

Tejas Srinivasan, Ting-Yun Chang, Leticia Pinto-Alva, Georgios Chochlakis, Mohammad Rostami, Jesse Thomason

Multimodal Agents that can be Deployed

Responses/actions by reasoning over inputs from multiple modalities

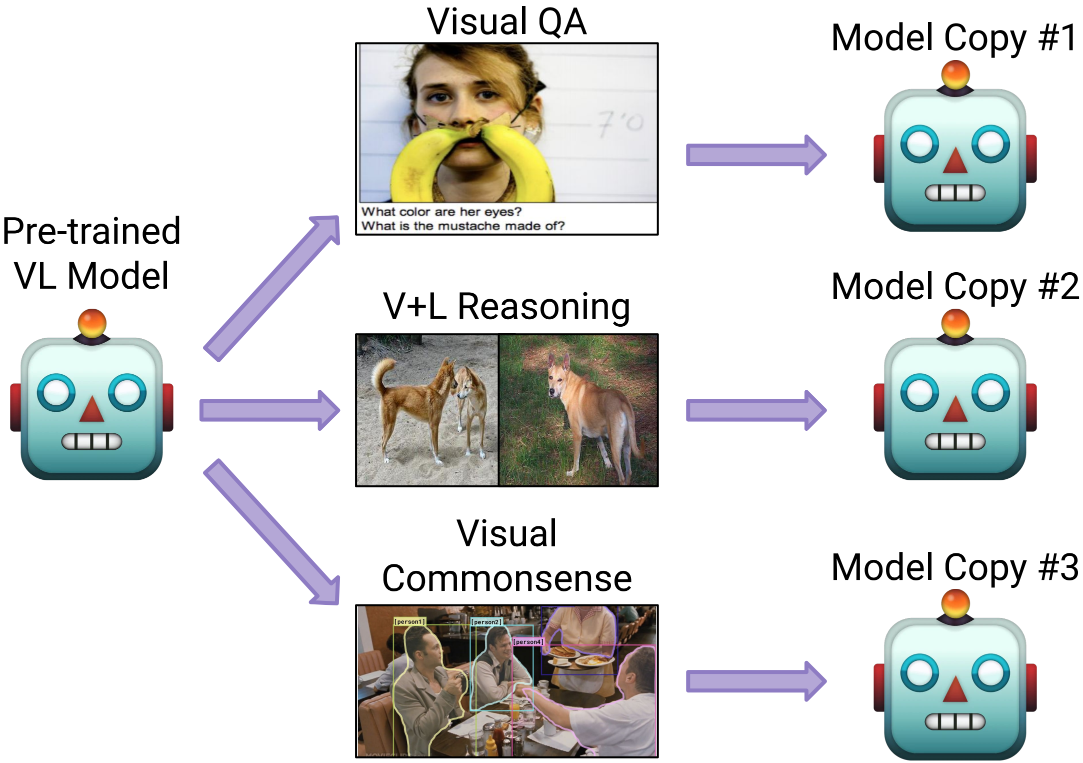


Visual QA + Dialog

Image/Scene Classification

Knowledge-based QA

Paradigms of VL Deployment: Single-Task Finetuning



Pre-trained VL Model

Visual QA

What color are her eyes?
What is the mustache made of?

V+L Reasoning

Visual Commonsense

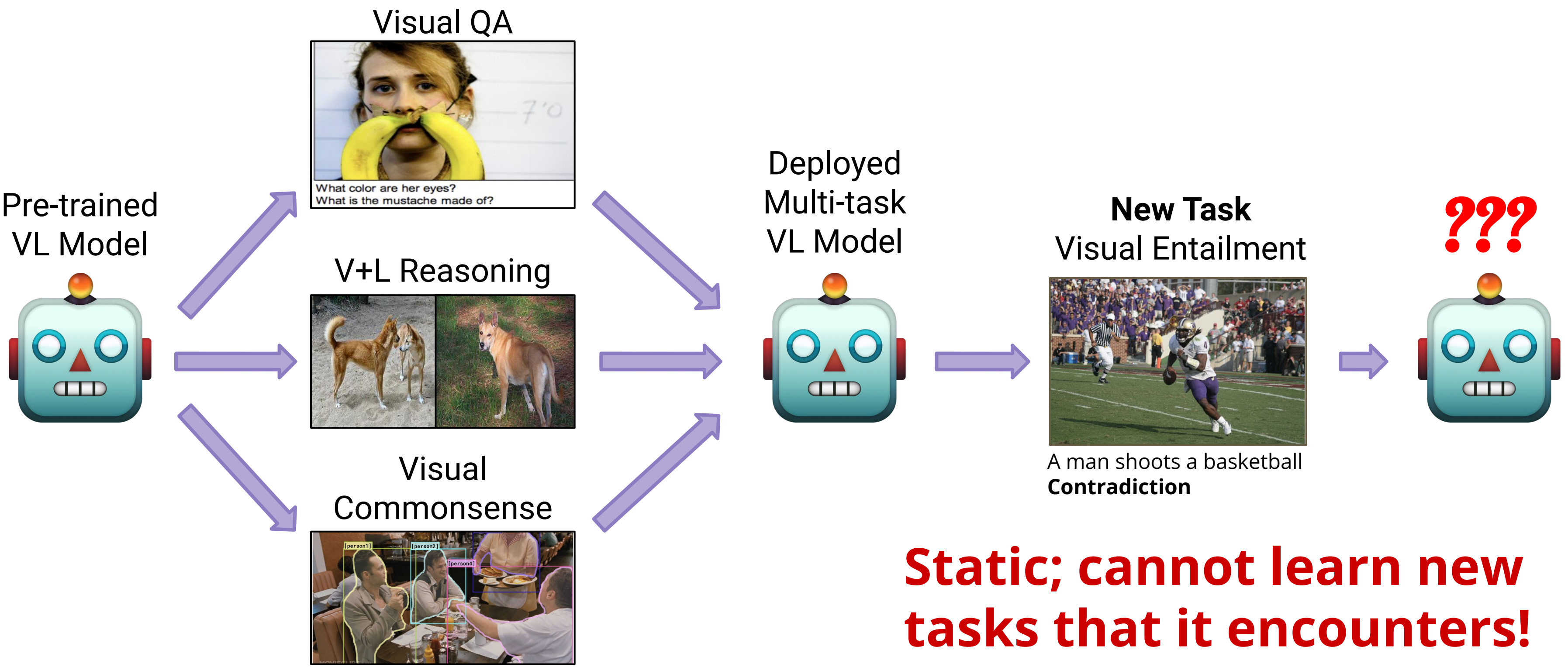
Model Copy #1

Model Copy #2

Model Copy #3

Need to store a copy of the model for each task!

Paradigms of VL Deployment: Multi-Task Learning



Pre-trained
VL Model

Visual QA

What color are her eyes?
What is the mustache made of?

V+L Reasoning

Visual
Commonsense

Deployed
Multi-task
VL Model

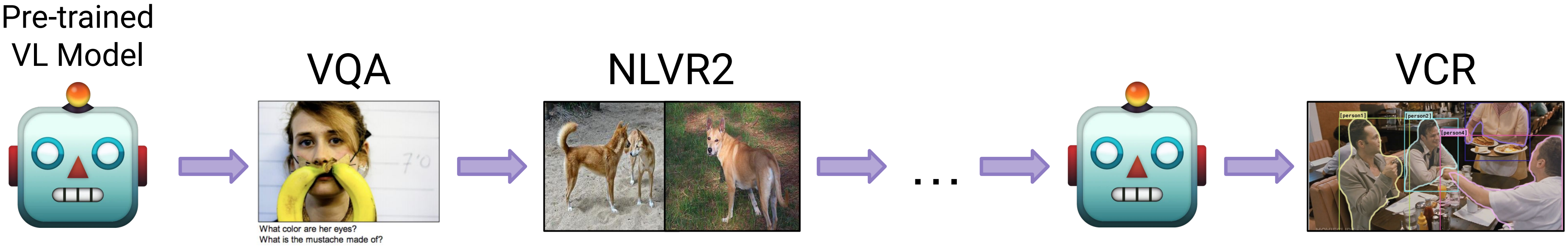
New Task
Visual Entailment

A man shoots a basketball
Contradiction

???

Static; cannot learn new tasks that it encounters!

Paradigms of VL Deployment: Continual Learning



Pre-trained
VL Model

VQA

NLVR2

...

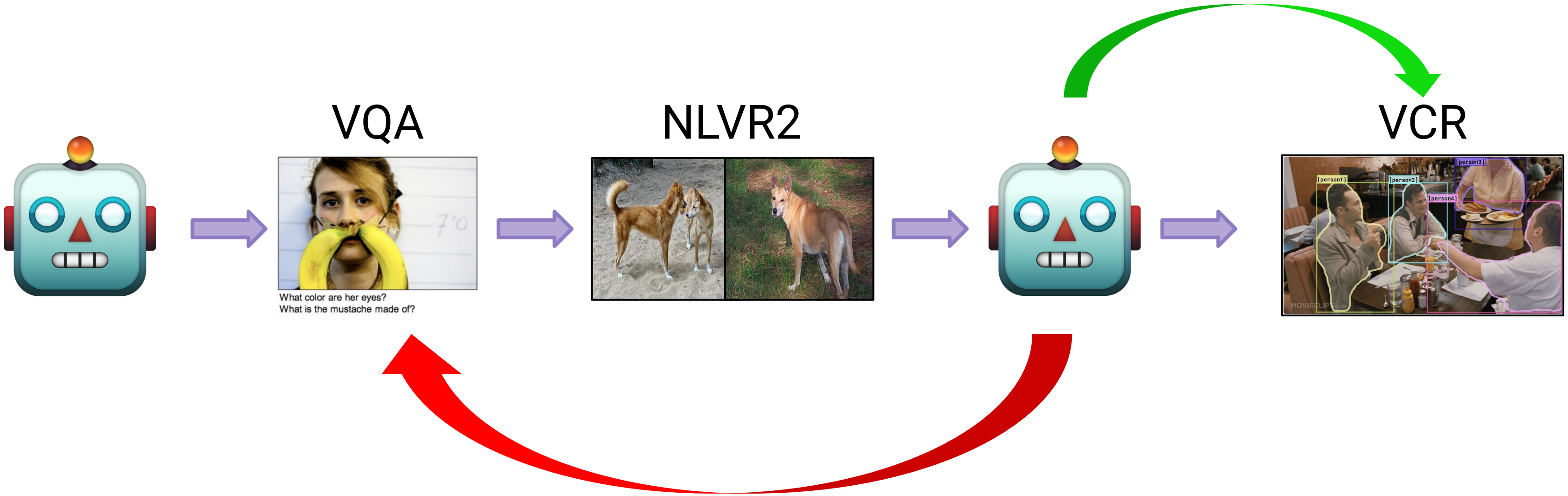
VCR

Dynamic, continually evolving paradigm

Unexplored in multimodal domain!

Challenges of Multimodal Continual Learning Deployment

Transfer knowledge
to learn new tasks



VQA

What color are her eyes?
What is the mustache made of?

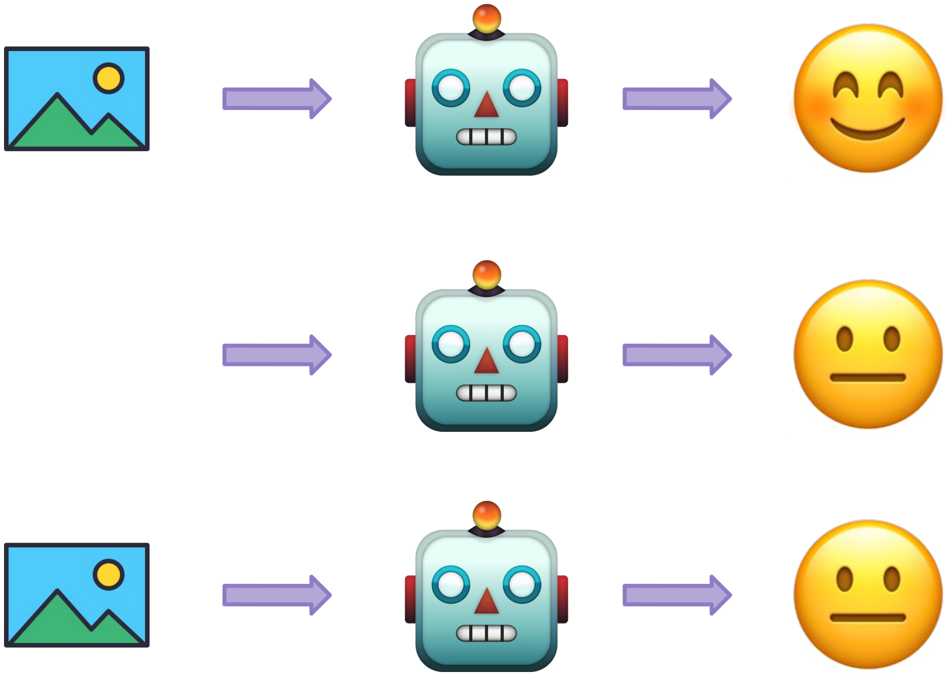
NLVR2

VCR

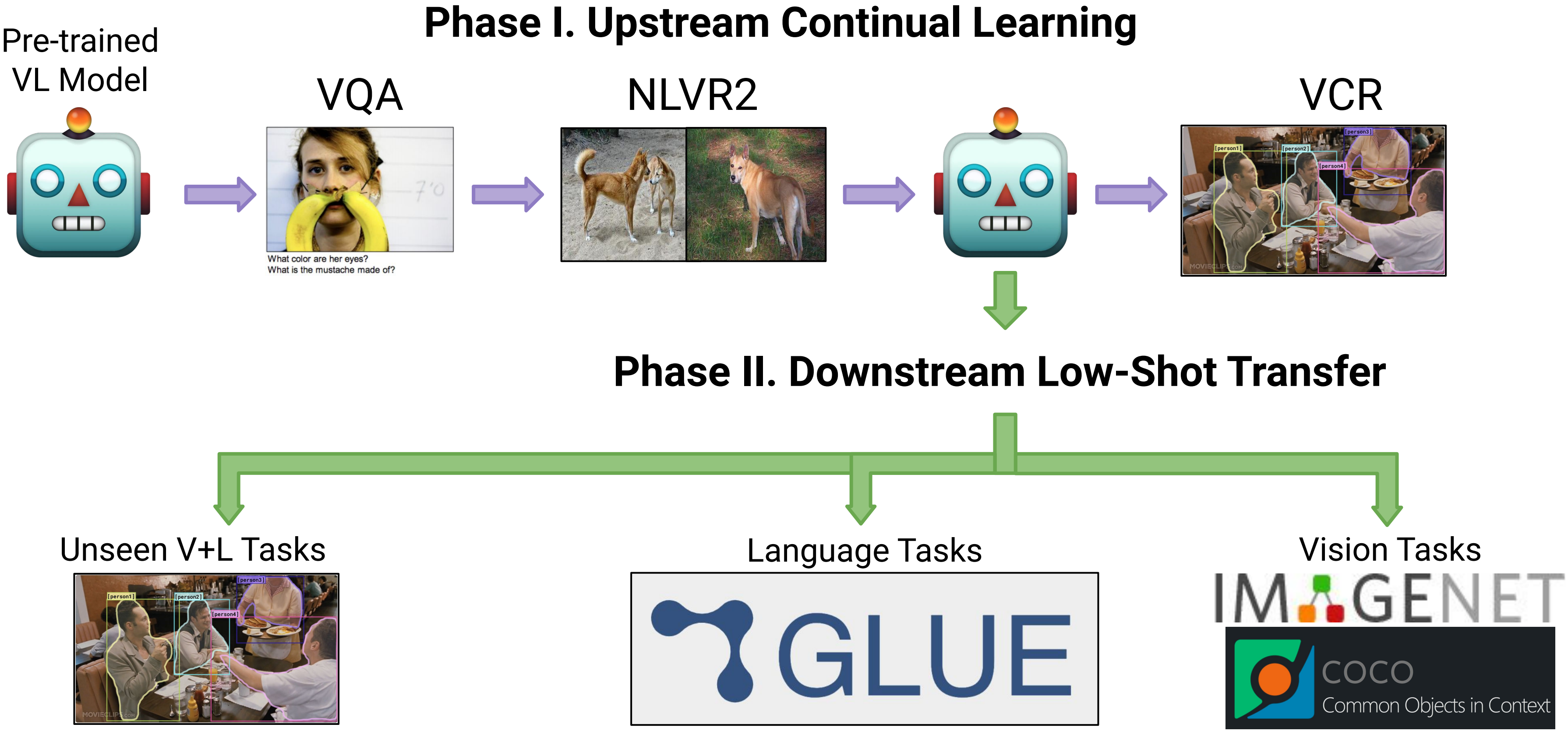
Don't forget
old tasks!!

Challenges of Multimodal Continual Learning Deployment

Not guaranteed to have all modalities when encountering new tasks!



CLiMB: The Continual Learning in Multimodality Benchmark



Phase I. Upstream Continual Learning

Pre-trained VL Model

VQA

NLVR2

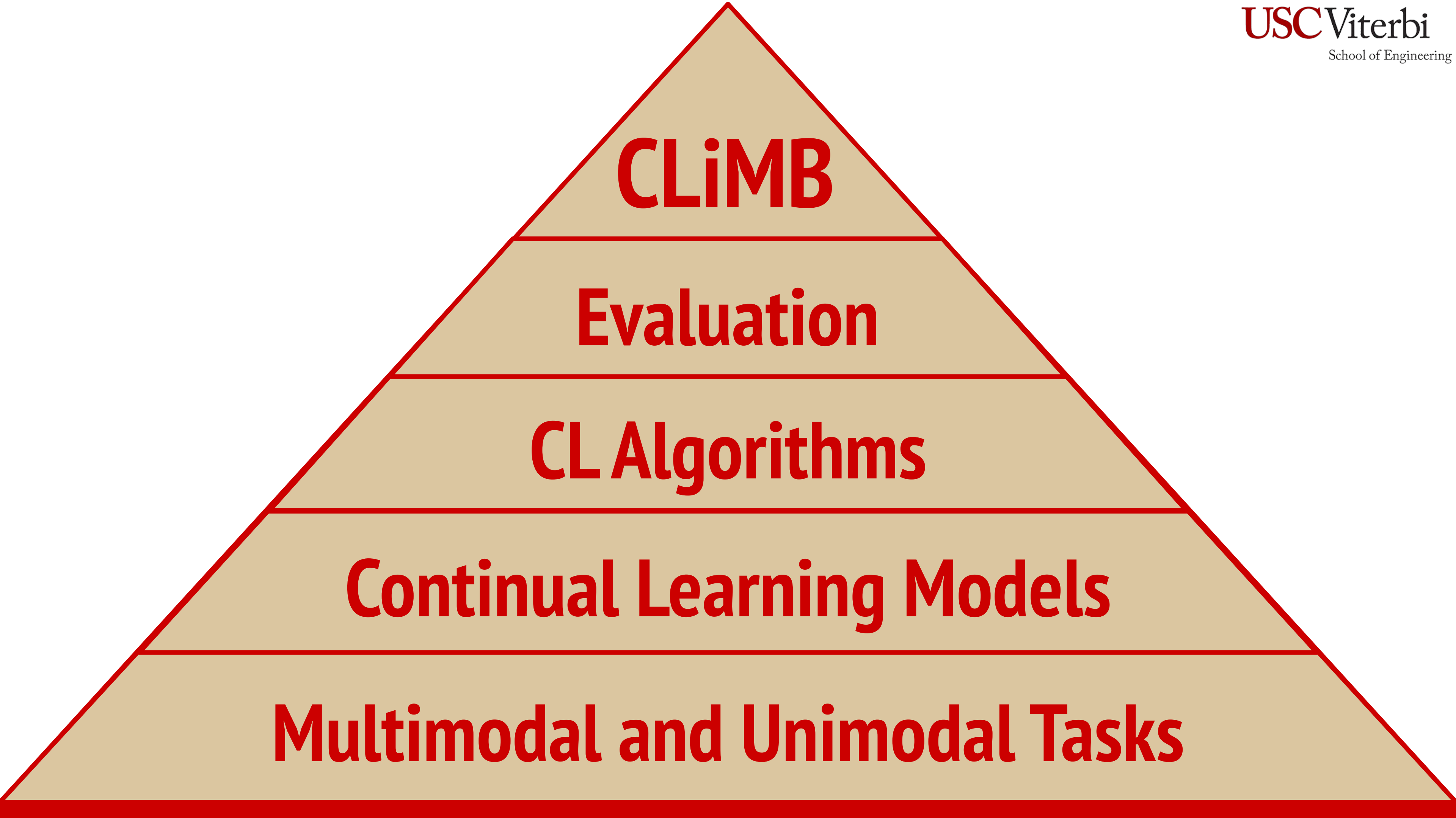
VCR

Phase II. Downstream Low-Shot Transfer

Unseen V+L Tasks

Language Tasks

Vision Tasks



CLiMB

Evaluation

CL Algorithms

Continual Learning Models

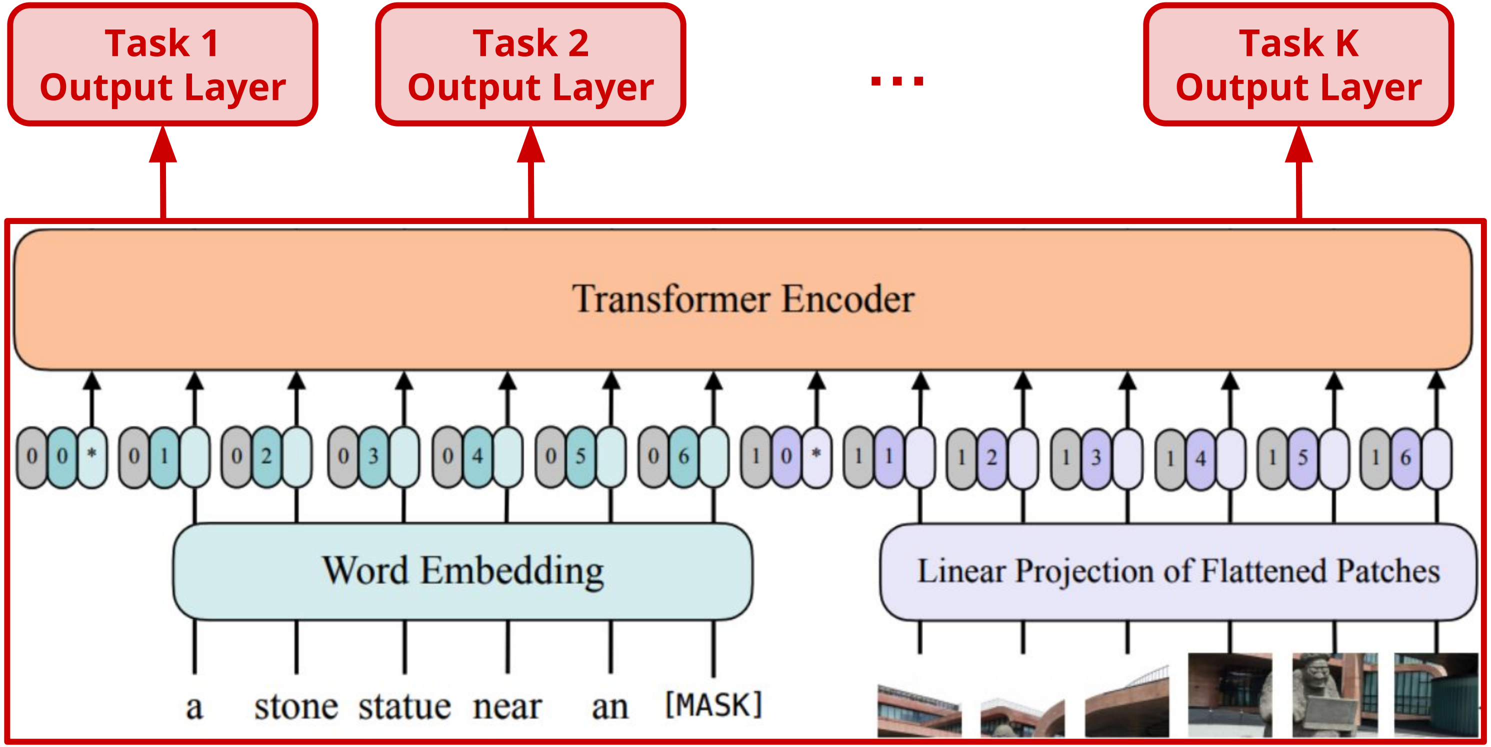
Multimodal and Unimodal Tasks

I. Multimodal and Unimodal Tasks

Vision-and-Language Tasks	● Visual Question Answering (VQAv2) ● Natural Language Visual Reasoning (NLVR2) ● Visual Entailment (SNLI-VE) ● Visual Commonsense Reasoning (VCR)
Language-Only Tasks	● IMDb, SST-2 Sentiment Classification ● HellaSwag ● CommonsenseQA ● Physical Interaction QA (PIQA)
Vision-Only Tasks	● ImageNet-1K Image Classification ● iNaturalist2019 Image Classification ● Places365 Image Classification ● MS-COCO Object Detection

CLiMB can be easily extended to include new multimodal and unimodal tasks!

II. Continual Learning Models

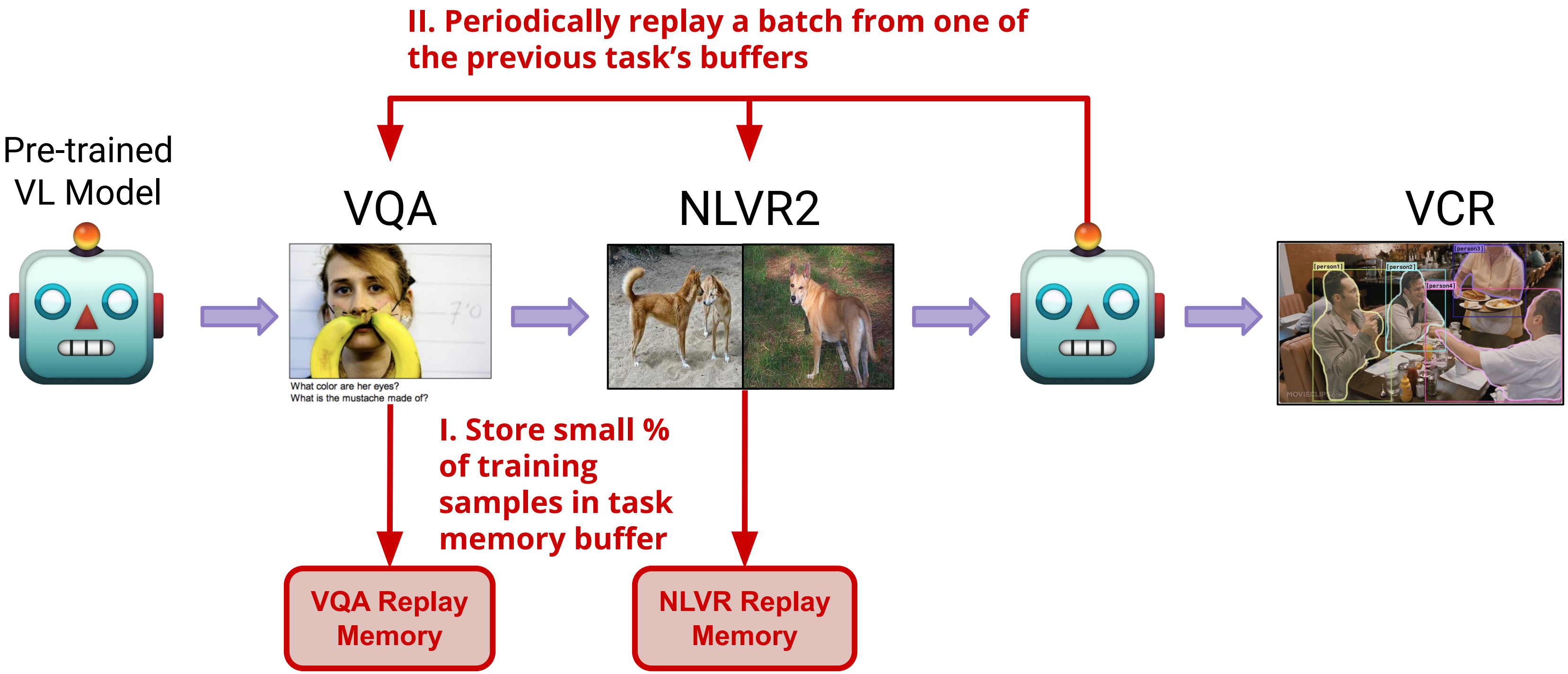


III. Continual Learning Algorithms

Currently, CLiMB supports 6 different Continual Learning algorithms:

- **Sequential Fine-tuning:** Fine-tune full encoder and task-specific layers
- **Frozen Encoder:** Train only task-specific layers
- **Frozen Bottom-K:** Fine-tune only top encoder layers and task layers
  - We set K=9
- **Experience Replay (ER)**
- **Elastic Weight Consolidation (EWC)**
- **Adapters**

Experience Replay



II. Periodically replay a batch from one of the previous task's buffers

Pre-trained VL Model

VQA

NLVR2

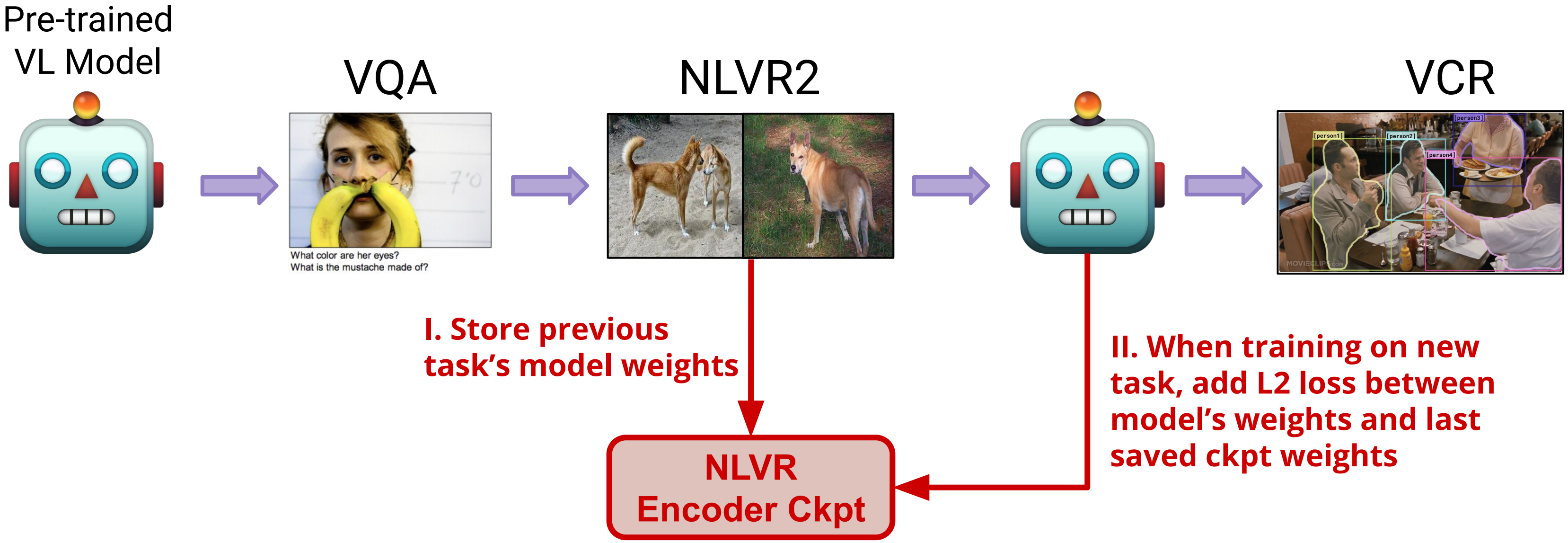
VCR

I. Store small % of training samples in task memory buffer

VQA Replay Memory

NLVR Replay Memory

Elastic Weight Consolidation



Pre-trained VL Model

VQA

What color are her eyes?
What is the mustache made of?

NLVR2

VCR

I. Store previous task’s model weights

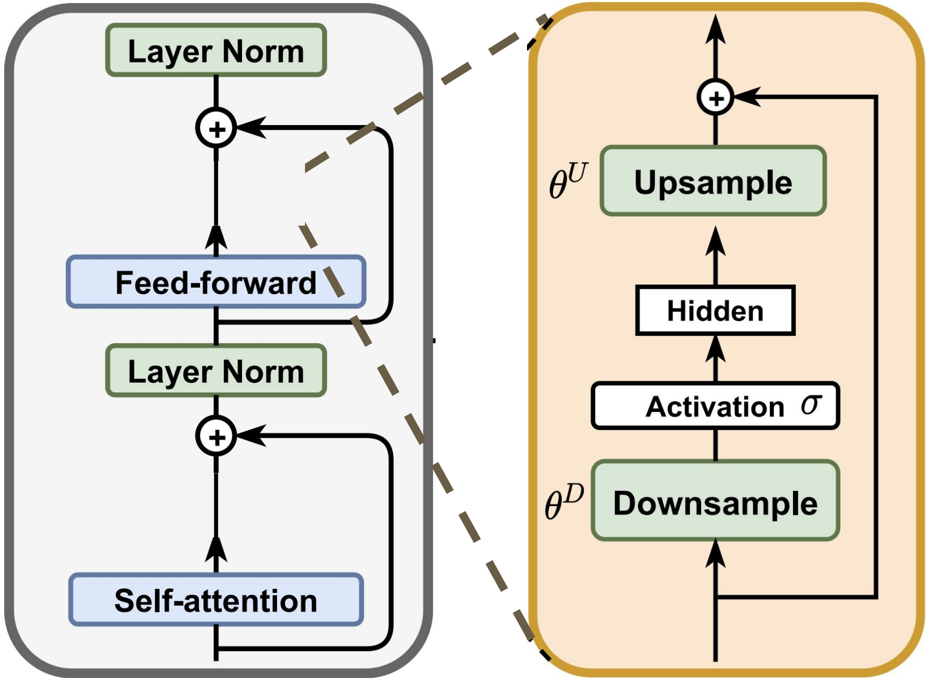
NLVR Encoder Ckpt

II. When training on new task, add L2 loss between model’s weights and last saved ckpt weights

Adapters

Insert new task-specific parameters into Transformer layers

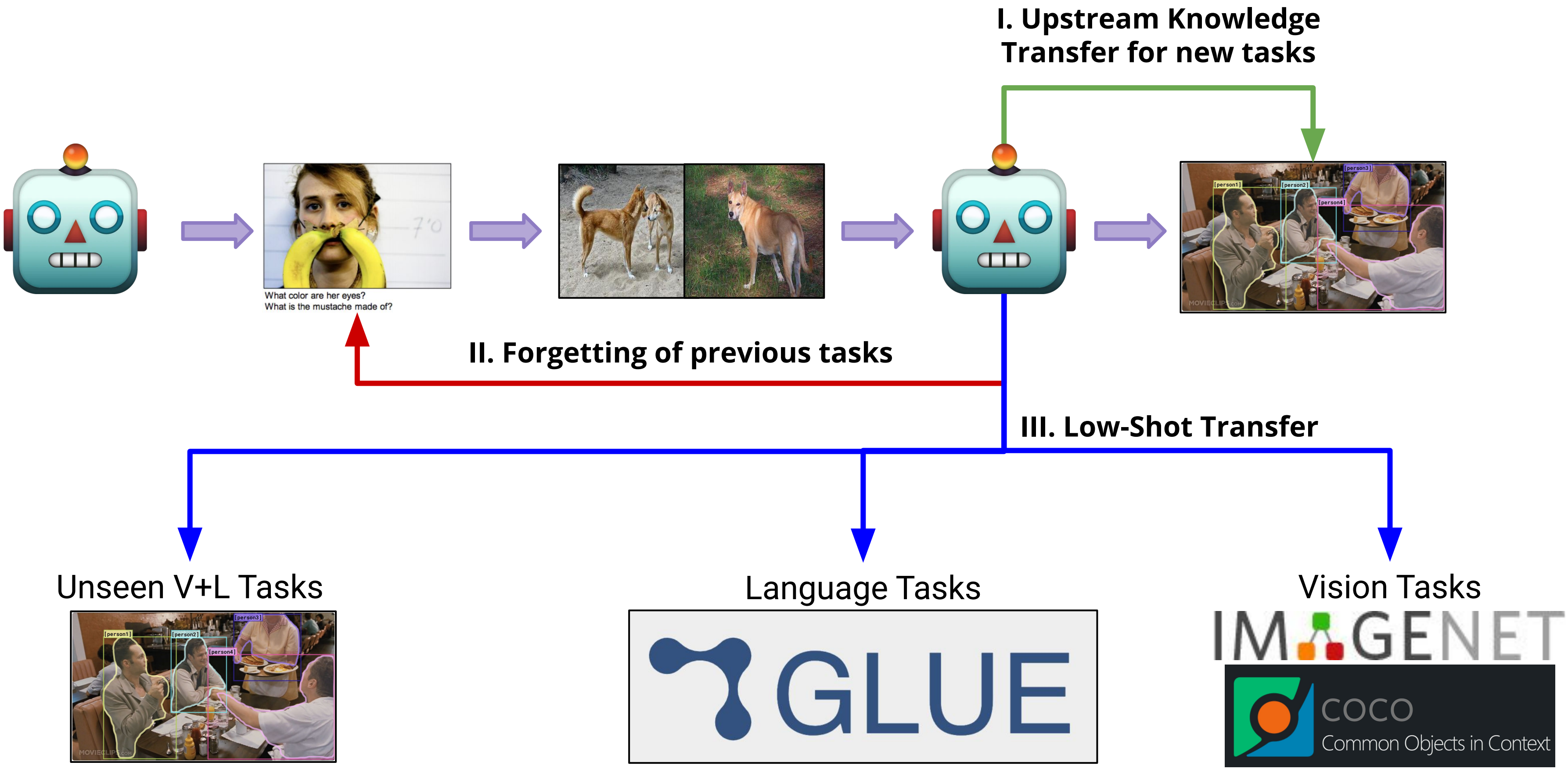
- Transformer parameters kept frozen - **no forgetting!**
- **Fewer learnable parameters, faster to train**
- **Comparable performance as full model fine-tuning**
- **No cross-task knowledge transfer**



Transformer Layer

Adapter Module

IV. Evaluation



I. Upstream Knowledge Transfer for new tasks

What color are her eyes?
What is the mustache made of?

II. Forgetting of previous tasks

III. Low-Shot Transfer

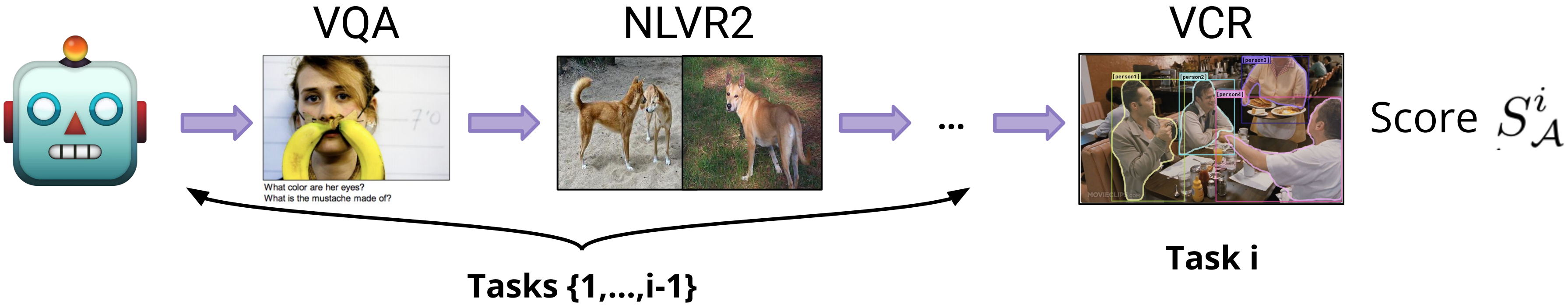
Unseen V+L Tasks

Language Tasks

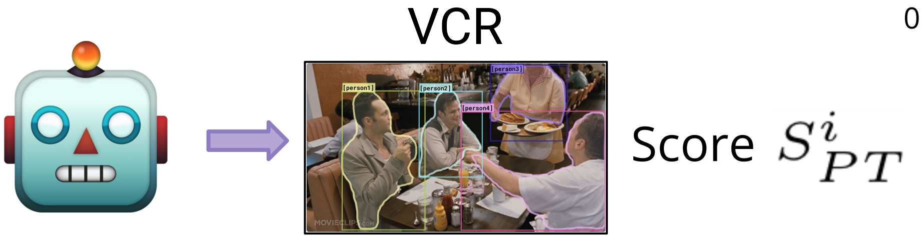
Vision Tasks

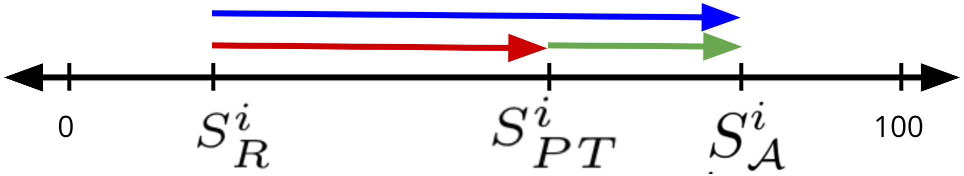
Upstream Evaluation I: Upstream Knowledge Transfer

With Continual Learning Algorithm A:



Without Continual Learning:





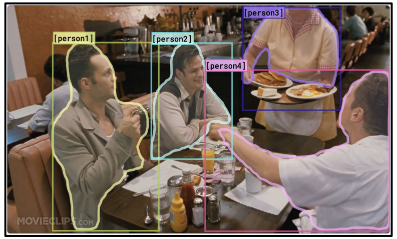
$$\mathbb{T}_{UK}(i) = \frac{S^i_A - S^i_{PT}}{S^i_{PT} - S^i_R}$$

Upstream Evaluation II: Forgetting Transfer

Directly evaluate on previously learned task j

Score $S_{\mathcal{A}}^{j\leftarrow i}$

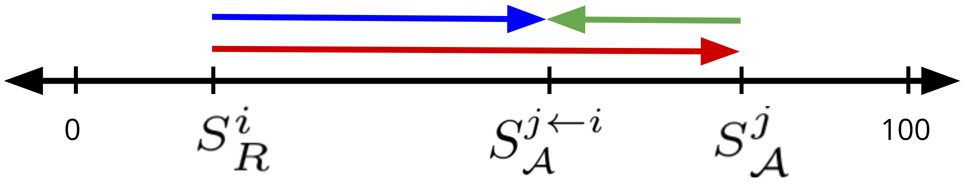
Task j: VCR



Score $S_{\mathcal{A}}^{j}$

Task i: SNLI-VE

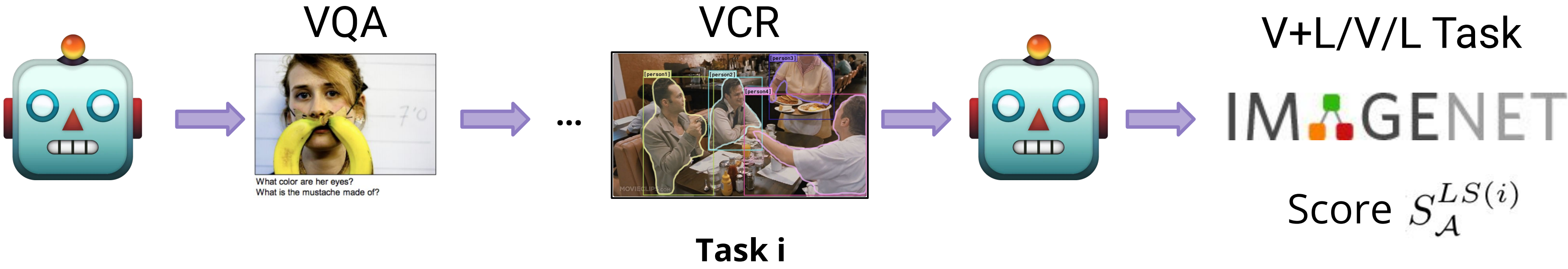
A man shoots a basketball
Contradiction



$$\mathbb{T}_F(j \leftarrow i) = \frac{S_{\mathcal{A}}^{j} - S_{\mathcal{A}}^{j\leftarrow i}}{S_{\mathcal{A}}^{j} - S_R^{j}}$$

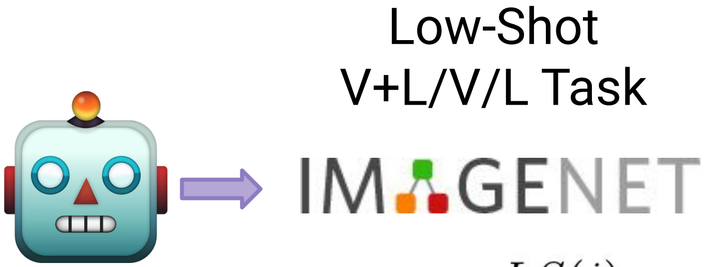
Downstream Evaluation: Low-Shot Transfer

With Continual Learning Algorithm $\mathcal{A}$:

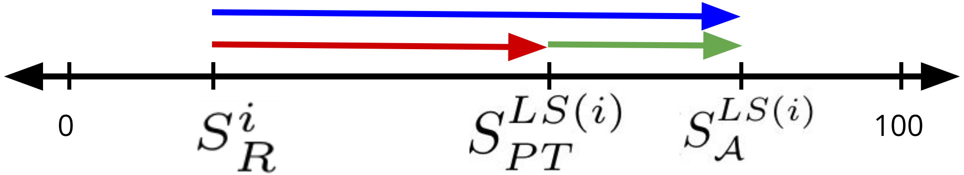


Score $S_{\mathcal{A}}^{LS(i)}$

Without Continual Learning:



Score $S_{PT}^{LS(i)}$



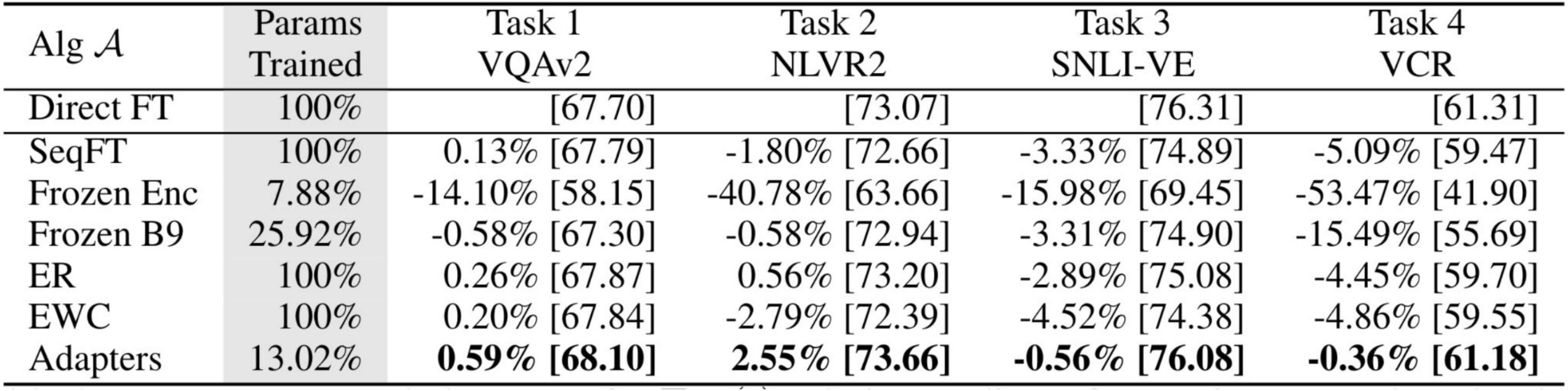
$$\mathbb{T}^{M}_{LS}(i) = \frac{S_{\mathcal{A}}^{LS(i)} - S_{PT}^{LS(i)}}{S_{PT}^{LS(i)} - S_R^i}$$

Experiments I: Upstream Continual Learning

- 4 V+L Tasks, ordered **VQA → NLVR2 → SNLI-VE → VCR**
- **ViLT**-based continual learning model
- **6** different Continual Learning algorithms

Results I: Upstream Continual Learning

Upstream Knowledge Transfer: How does Continual Learning affect model's ability to learn newly arriving tasks?



Alg $\mathcal{A}$	Params Trained	Task 1 VQAv2	Task 2 NLVR2	Task 3 SNLI-VE	Task 4 VCR
Direct FT	100%	[67.70]	[73.07]	[76.31]	[61.31]
SeqFT	100%	0.13% [67.79]	-1.80% [72.66]	-3.33% [74.89]	-5.09% [59.47]
Frozen Enc	7.88%	-14.10% [58.15]	-40.78% [63.66]	-15.98% [69.45]	-53.47% [41.90]
Frozen B9	25.92%	-0.58% [67.30]	-0.58% [72.94]	-3.31% [74.90]	-15.49% [55.69]
ER	100%	0.26% [67.87]	0.56% [73.20]	-2.89% [75.08]	-4.45% [59.70]
EWC	100%	0.20% [67.84]	-2.79% [72.39]	-4.52% [74.38]	-4.86% [59.55]
Adapters	13.02%	**0.59% [68.10]**	**2.55% [73.66]**	**-0.56% [76.08]**	**-0.36% [61.18]**

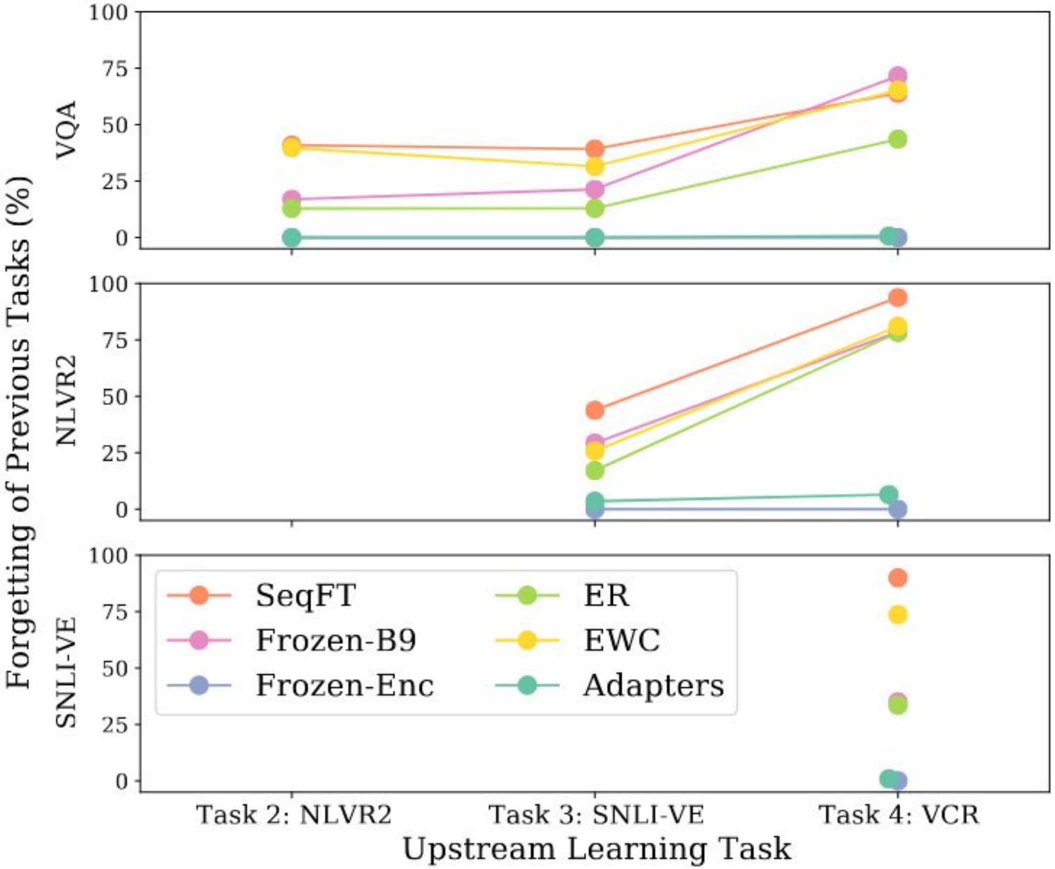
- **More continual learning hurts ability to learn new tasks**
- **Adapters do not show negative transfer**, comparable to full model fine-tuning

USC Viterbi
School of Engineering

Results I: Upstream Continual Learning

Forgetting: How does learning new tasks affect model's performance on already-learned tasks?

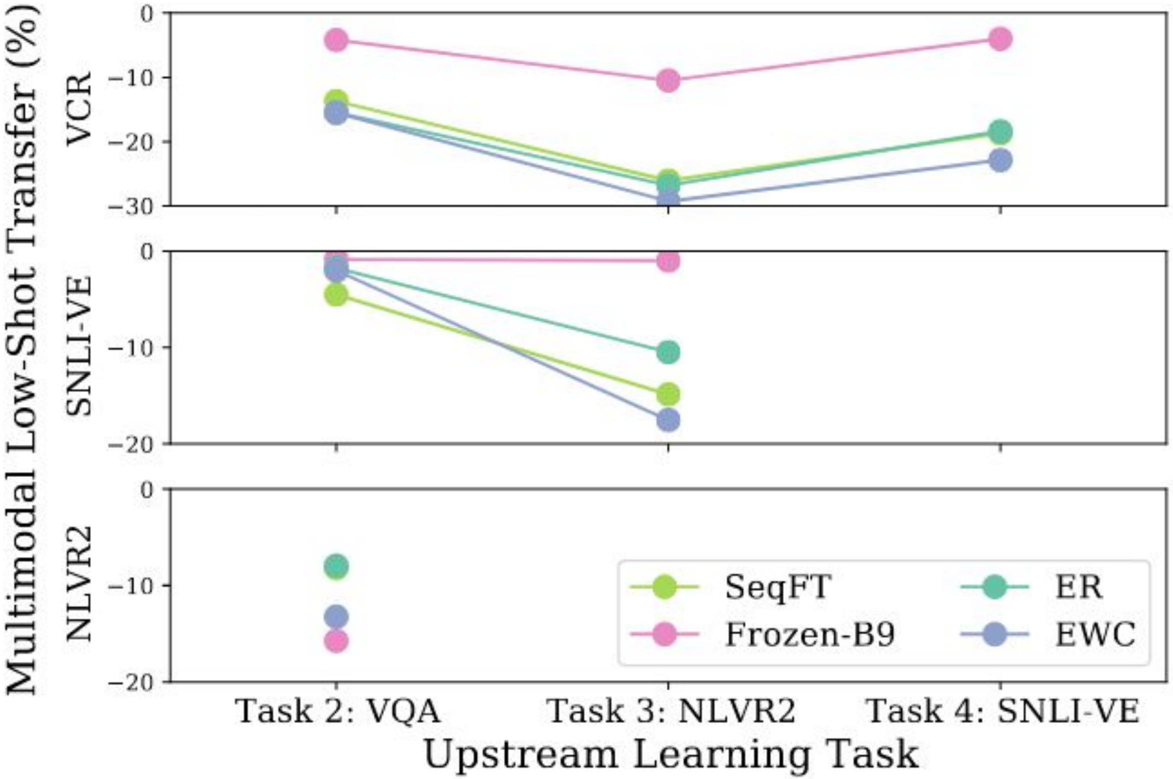
- **More fine-tuned params == more forgetting**
- **ER > EWC**
- **Adapters >>>>**
- **Forgetting more severe after VCR**



Experiments and Results II: Downstream Low-Shot Transfer

Low-Shot Transfer to Unseen V+L Tasks

- **Low-Shot transfer is always negative**
- **Unsurprising — CL also hurts model transfer with full training data**

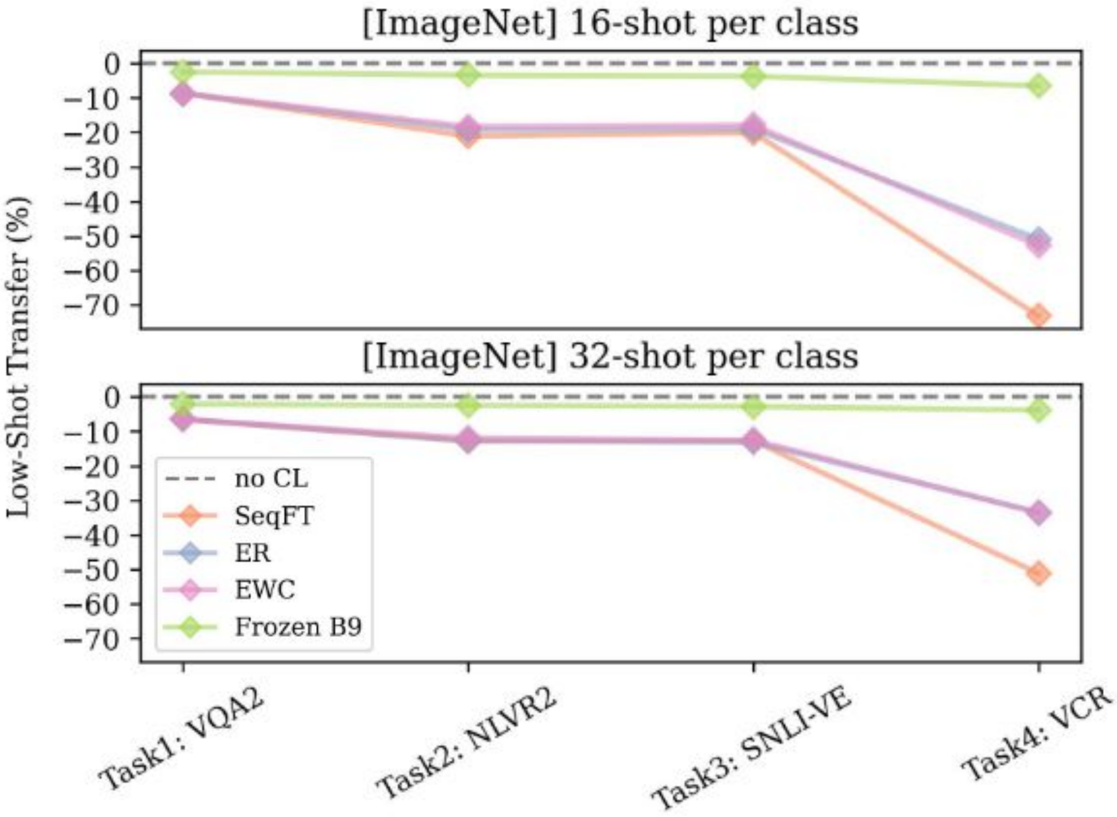


Experiments and Results II: Downstream Low-Shot Transfer

Low-Shot Transfer to Vision-Only Tasks

Language prompt: “This is an image.”

- **ViLT achieves good low-shot performance on vision tasks**
- **CL hurts low-shot transfer**
- **NLVR2 and VCR have more negative effect**

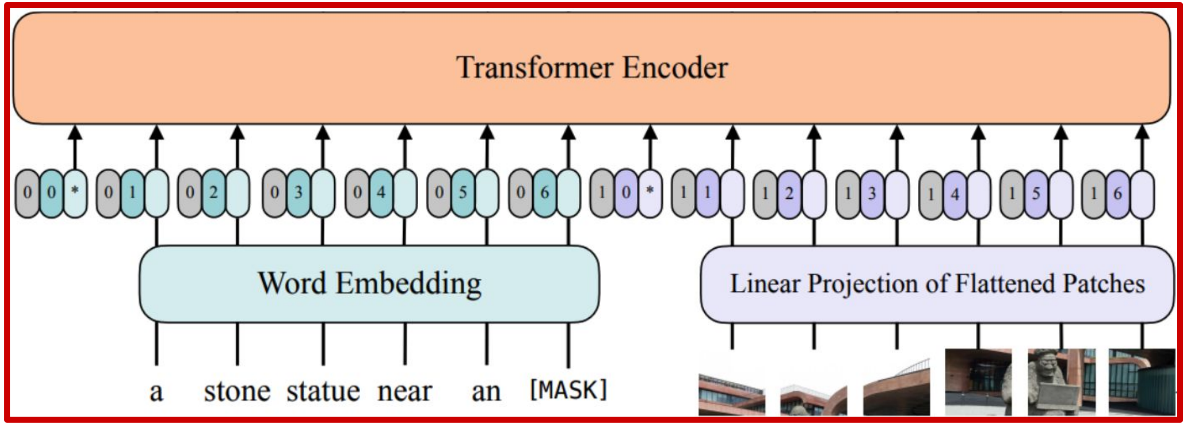


Experiments and Results II: Downstream Low-Shot Transfer

Low-Shot Transfer to Language-Only Tasks

Adapting ViLT for NLP tasks:

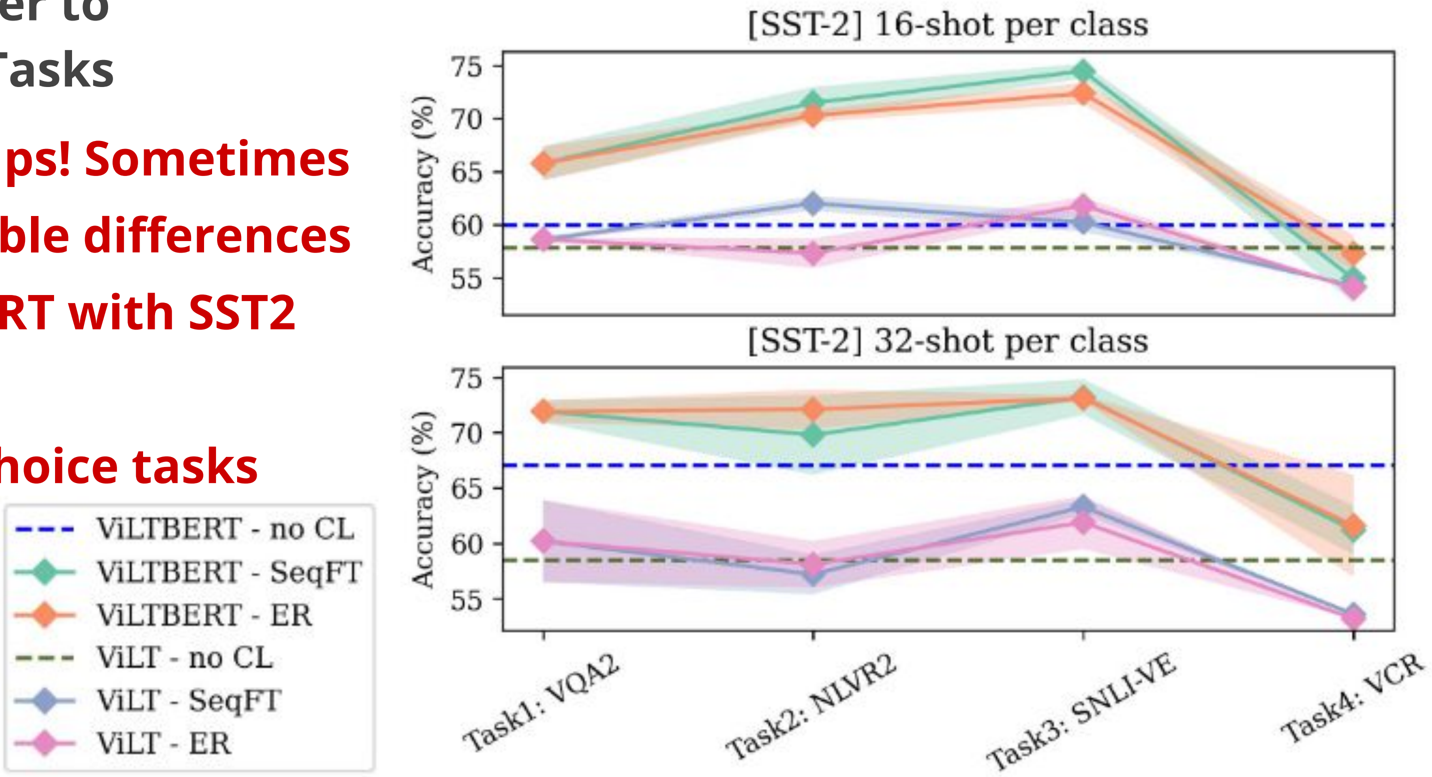
- Use **"average" MS-COCO image** for in-distribution visual input
- **Extend** language position embeddings
- **ViLT-BERT:** Replace language input embeddings with BERT representations



Experiments and Results II: Downstream Low-Shot Transfer

Low-Shot Transfer to Language-Only Tasks

- **Upstream CL helps! Sometimes**
- **ViLT sees negligible differences**
- **CL helps ViLT-BERT with SST2**
- **VCR hurts SST2**
- **CL hurts multi-choice tasks**



Conclusions

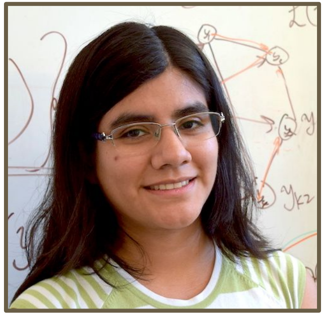
- We propose **CLiMB**, a benchmark to study CL in multimodal settings
- CLiMB is an **extensible community tool** for studying tasks, model architectures, and CL algorithms.
- **Existing Continual Learning methods fail** at:
  - generalizing well to sequences of multimodal tasks
  - Enabling low-shot adaptation to multi/unimodal tasks
- **Adapters are most effective** at preserving pre-trained model knowledge and forgetting mitigating
- There is **a need for new research** into Continual Learning strategies for this challenging multimodal setting.

Future Directions

- **Adapters that share knowledge across tasks**
- **Multimodal Adapters**
- Studying **multimodal distribution shifts**
- Building a **task-agnostic** modeling framework:
    - Sequence-to-sequence task formulations
    - Integrating **generalist models** into CLiMB
    - **Embodied** navigation, task completion

Acknowledgements

Ting-Yun Chang✨



Leticia Pinto Alva✨

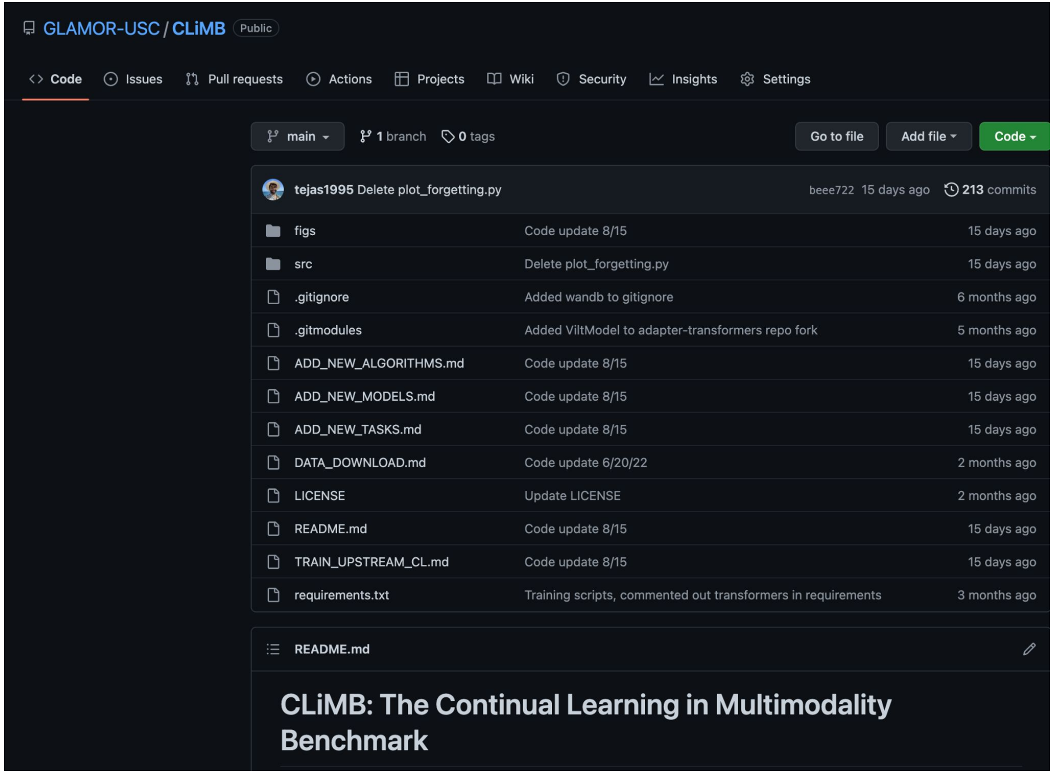
Georgios Chochlakis

Mohammad Rostami

Jesse Thomason ✨

Thank You!!

https://github.com/GLAMOR-USC/CLiMB



tejas1995 Delete plot_forgetting.py beee722 15 days ago 213 commits

figs	Code update 8/15	15 days ago
src	Delete plot_forgetting.py	15 days ago
.gitignore	Added wandb to gitignore	6 months ago
.gitmodules	Added ViltModel to adapter-transformers repo fork	5 months ago
ADD_NEW_ALGORITHMS.md	Code update 8/15	15 days ago
ADD_NEW_MODELS.md	Code update 8/15	15 days ago
ADD_NEW_TASKS.md	Code update 8/15	15 days ago
DATA_DOWNLOAD.md	Code update 6/20/22	2 months ago
LICENSE	Update LICENSE	2 months ago
README.md	Code update 8/15	15 days ago
TRAIN_UPSTREAM_CL.md	Code update 8/15	15 days ago
requirements.txt	Training scripts, commented out transformers in requirements	3 months ago

README.md

CLiMB: The Continual Learning in Multimodality Benchmark