Multimodal Generative Models: Unification, Planning Agents, Evaluation

Mohit Bansal
Talk Outline

A journey of multimodal generative models for enhancing their unification, interpretable planning/programming, evaluation:

• **Unified/Universal Multimodal Learning** (for Generalizability, Shared Knowledge, Efficiency)
  - VLT5: Unifying Vision-and-Language Tasks via Text Generation [ICML 2021]
  - TVLT: Textless Vision-Language Transformer [NeurIPS 2022]

• **Interpretable Multimodal Generation via LLM Planning/Programming Agents** (for Understanding, Control, Faithfulness, OOD)
  - VideoDirectorGPT: Consistent Multi-Scene Video Generation via LLM-Guided Planning [COLM 2024]
  - DiagrammerGPT: Generating Diagrams via LLM Planning [COLM 2024]; EnvGen: Adapting Environments via LLMs for Training Embodied Agents [COLM 2024]

• **Evaluation of Multimodal Generation Models** (of Fine-grained Skills, Faithfulness, Social Biases)
  - DALL-Eval: Probing the Reasoning Skills and Social Biases of Text-to-Image Generation Models [ICCV 2023]
  - Davidsonian Scene Graph: Improving Reliability in Fine-grained Evaluation for Text-to-Image Generation [ICLR 2024]

• **Next Big Challenges**: trade-offs, structure, non-verbal, interaction, reasoning, causality, long-distance fine-grained evaluation, efficiencies
Talk Outline

A journey of multimodal generative models for enhancing their unification, interpretable planning/programming, evaluation:

- **Unified/Universal Multimodal Learning** (for Generalizability, Shared Knowledge, Efficiency)
  - VLT5: Unifying Vision-and-Language Tasks via Text Generation [ICML 2021]
  - TVLT: Textless Vision-Language Transformer [NeurIPS 2022]

- **Interpretable Multimodal Generation via LLM Planning/Programming Agents** (for Understanding, Control, Faithfulness, OOD)
  - VideoDirectorGPT: Consistent Multi-Scene Video Generation via LLM-Guided Planning [COLM 2024]
  - DiagrammerGPT: Generating Diagrams via LLM Planning [COLM 2024]; EnvGen: Adapting Environments via LLMs for Training Embodied Agents [COLM 2024]

- **Evaluation of Multimodal Generation Models** (of Fine-grained Skills, Faithfulness, Social Biases)
  - DALL-Eval: Probing the Reasoning Skills and Social Biases of Text-to-Image Generation Models [ICCV 2023]
  - Davidsonian Scene Graph: Improving Reliability in Fine-grained Evaluation for Text-to-Image Generation [ICLR 2024]

- **Next Big Challenges**: trade-offs, structure, non-verbal, interaction, reasoning, causality, long-distance fine-grained evaluation, efficiencies
Language: Pre-training → Fine-tuning

Motivation: the amount of data is limited in downstream tasks and pre-training enables much more data.

Language Pre-training:
Text in Wikipedia
~2500M Tokens (i.e., Words)

Language Fine-tuning
Movie Review [Maas et al., ACL 2011]
~2.5M Tokens (i.e., Words)

Transformer + Linear Layers

Transformer
[Vaswani, NeurIPS 2017]

Pre-training
[Devlin et al., NAACL 2019]

Model
[Peters et al., NAACL 2018],
[Devlin et al., NAACL 2019]

Sentiment Analysis

Vision: Pre-training $\rightarrow$ Fine-tuning

Motivation: the amount of data is limited in downstream tasks and pre-training enables much more data.

**Visual Pre-training:**
- ImageNet
  - [Deng, CVPR 2009]
  - 1.3M Images, 1000 Labels

**Visual Fine-tuning:**
- MS COCO
  - [Lin, ECCV 2009]
  - 120K Images, 80 Labels

**Image Classification**
- DenseNet
  - [Huang, CVPR 2017]

**Object Detection**
- Faster RCNN
  - [Ren, NeurIPS 2015]
Pre-training of Single Modality Tasks

Limitation: Single-modality pre-trained models are not aware of the interactions between vision and language

Visual Pre-training:

Language Pre-training:

Visual Question Answering, Navigation, Grounding, ...

Multimodal Fusion Layers
**Large-Scale Cross-Modal Pre-training: LXMERT**

- LXMERT combines knowledge from text, vision and cross-modal matching: vision-language transformers with 3 encoders (object relations, language, cross-modal) & 5 pretraining tasks: masked-LM, masked-Object-Prediction (feature regression+label classification), cross-modality matching, image-QA.

- Achieved big gains + sota on several VL tasks such as VQA, GQA, NLVR2, VizWiz, etc.

![Diagram of LXMERT model](Image)
Tons of Specialized Vision-and-Language Pretraining Models

- Many different architectures (single vs. multi-stream), attention methods, objective functions, encoder/decoders, output heads, specialized modules (OCR/ASR/Tokenizers), etc., etc.!

[Sun et al., 2019; Tan and Bansal, 2019; Lu et al., 2019; Li et al., 2019; Su et al., 2019; Chen et al., 2020; Zhou et al., 2020; Li et al., 2020; inter alia]
Part 1: Unified/Universal Multimodal Learning

**VL-T5 (ICML 2021)**
all multimodal tasks via text generation

**TVLT (NeurIPS 2022)**
video modeling without text (audio as images)

**UDOP (CVPR 2023)**
document image/text/layout with single architecture

**CoDi (NeurIPS 2023)**
generating any-to-any input-output modality combination
Part 1: Unified/Universal Multimodal Learning

**VL-T5 (ICML 2021)**
all multimodal tasks via text generation

**TVLT (NeurIPS 2022)**
video modeling without text (audio as images)

**UDOP (CVPR 2023)**
document image/text/layout with single architecture

**CoDi (NeurIPS 2023)**
generating any-to-any input-output modality combination
**Diverse Vision-and-Language Tasks (and Specialized Models)**

**Visual Question Answering**
- **Question:** What is the mustache made of?
- **Answer:** Banana
- **Model:** e.g., MCAN

**Visual Grounding**
- **Object:** banana mustache
- **Model:** e.g., MAttNet

**Image Captioning**
- **Image:** A woman with banana mustache
- **Model:** e.g., BUTD/AoANet

**Multimodal Machine Translation (En-Kr)**
- **Image:** A woman with banana mustache
- **Text:** 바나나 코트수염을 한 여자
e.g., ImageiT

References:
- Anderson et al., 2018. Bottom-Up and Top-Down Attention for Image Captioning and Visual Question Answering
- Yu et al., 2019. Deep Modular Co-Attention Networks for Visual Question Answering
- Huang et al., 2019. Attention on Attention for Image Captioning
- Yu et al., 2018. MAttNet: Modular Attention Network for Referring Expression Comprehension
Task-specific Architectures / Objectives / Modules

Visual Question Answering

V&L Transformer 1

“fire hydrant”

Top-K answer scores

Sigmoid

VQA head

Multi-label Classification

[CLS] What is the man jumping over?

Visual Grounding

V&L Transformer 2

[CLS] fire hydrant

Classification

Region scoring head

Softmax
Can we tackle all V&L tasks with a single objective?
VL-T5: Many Multimodal Tasks as Text Generation

[Cho et al., ICML 2021]
VL-T5: Many Multimodal Tasks as Text Generation

Bidirectional Multimodal Encoder

Visual embedding

Multimodal Conditional Language Modeling

Autoregressive Text Decoder

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

[Cho et al., ICML 2021]
**VL-T5: Many Multimodal Tasks as Text Generation**

### Visual Question Answering

**Previous models**

- **VQA head**
- **Top-K answer scores**
- **Multi-label Classification**
- **V&L Transformer**
- **[CLS] What is the man jumping over?**

**Ours**

- **V&L Transformer**
- **vqa: What is the man jumping over?**
- **Multi-modal Conditional Language Modeling**

### Visual Grounding

**Previous models**

- **Region scoring head**
- **Classification**
- **V&L Transformer**
- **[CLS] fire hydrant**

**Ours**

- **V&L Transformer**
- **visual grounding: fire hydrant**
- **"<vis_3>"**

---

[Cho et al., ICML 2021]
VL-T5: Many Multimodal Tasks as Text Generation

Previous models

V&L Transformer

NLVR$^2$ head

0 or 1

Binary Classification

V&L Transformer

V&L Transformer

Image 1 regions

... text

Image 2 regions

text

Multimodal Machine Translation (En-De)

Multi-modal
Conditional Language Modeling

Ours

“true” or “false”

Ein Mann springt über einen Hydranten

Decoder

V&L Transformer

A man is jumping over a fire hydrant

Ein Mann springt über einen Hydranten

Translate English to German:

A man is jumping over a fire hydrant

V&L Transformer

nlvr: [text]

[Cho et al., ICML 2021]
Unified Architecture Comparable to Specialized Models

<table>
<thead>
<tr>
<th>Method</th>
<th>Pretrain Images</th>
<th>Discriminative tasks</th>
<th>Generative tasks</th>
</tr>
</thead>
<tbody>
<tr>
<td>LXMERT</td>
<td>180K</td>
<td>VQA test-std Acc 72.5</td>
<td>COCO Cap test 116.5</td>
</tr>
<tr>
<td>ViLBERT</td>
<td>3M</td>
<td>GQA test-std Acc 60.3</td>
<td>Karpathy test</td>
</tr>
<tr>
<td>UNITER&lt;sub&gt;Base&lt;/sub&gt;</td>
<td>4M</td>
<td>NLVR&lt;sup&gt;2&lt;/sup&gt; test-P Acc 74.5</td>
<td>CIDEr test</td>
</tr>
<tr>
<td>Unified VLP</td>
<td>3M</td>
<td>RefCOCOg test&lt;sub&gt;d&lt;/sub&gt; Acc 54.8</td>
<td>Multi30K En-De test</td>
</tr>
<tr>
<td>Oscar&lt;sub&gt;Base&lt;/sub&gt;</td>
<td>4M</td>
<td>VCR Q→AR test Acc 58.2</td>
<td>2018 BLEU test</td>
</tr>
<tr>
<td>XGPT</td>
<td>3M</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MeMAD</td>
<td>-</td>
<td></td>
<td></td>
</tr>
<tr>
<td>VL-T5</td>
<td>180K</td>
<td>VQA test-std Acc 70.3</td>
<td></td>
</tr>
<tr>
<td>VL-BART</td>
<td>180K</td>
<td>GQA test-std Acc 60.8</td>
<td></td>
</tr>
</tbody>
</table>

[Cho et al., ICML 2021]
Multi-task Learning with Single Shared Set of Parameters

<table>
<thead>
<tr>
<th>Method</th>
<th>Finetuning tasks</th>
<th># Params</th>
<th>Discriminative tasks</th>
<th>Generative tasks</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>VQA Karpathy test Acc</td>
<td>COCO Caption Karpathy test</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>GQA test-dev Acc</td>
<td>CIDEr</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>NLVR² test-P Acc</td>
<td>Multi30K En-De</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>RefCOCOg test&lt;sup&gt;d&lt;/sup&gt; Acc</td>
<td>test2018 BLEU</td>
</tr>
<tr>
<td>VL-T5</td>
<td>single task</td>
<td>7P</td>
<td>67.9</td>
<td>116.1</td>
</tr>
<tr>
<td></td>
<td>all tasks</td>
<td>P</td>
<td>67.2</td>
<td>110.8</td>
</tr>
<tr>
<td>VL-T5</td>
<td></td>
<td></td>
<td>60.0</td>
<td>57.5</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>73.6</td>
<td>110.8</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>71.3</td>
<td>57.5</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>57.5</td>
<td>38.6</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>69.4</td>
<td>37.6</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>55.3</td>
<td></td>
</tr>
</tbody>
</table>

Similar performance with 1/7<sup>th</sup> = 14% parameters!
Multi-task Learning with Single Shared Set of Parameters

<table>
<thead>
<tr>
<th>Method</th>
<th>Finetuning tasks</th>
<th># Params</th>
<th>Discriminative tasks</th>
<th>Generative tasks</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>VQA Karpathy test Acc</td>
<td>COCO Caption Karpathy test CIDEr</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>GQA test-dev Acc</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>NLVR² test-P Acc</td>
<td>Multi30K En-De test2018 BLEU</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>RefCOCOg test\textsuperscript{d} Acc</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>VCR Val Acc</td>
<td></td>
</tr>
<tr>
<td>VL-T5</td>
<td>single task</td>
<td>7P</td>
<td>67.9</td>
<td>116.1</td>
</tr>
<tr>
<td></td>
<td>all tasks</td>
<td>P</td>
<td>67.2</td>
<td>110.8</td>
</tr>
<tr>
<td>VL-T5</td>
<td></td>
<td></td>
<td>60.0</td>
<td>38.6</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>73.6</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>71.3</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>57.5</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>58.9</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>71.6</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>69.4</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>55.3</td>
<td></td>
</tr>
</tbody>
</table>

Similar performance with $1/7^{th} = 14\%$ parameters!

- Also performs better on rare/unseen categories!
### Multi-task Learning with Single Shared Set of Parameters

<table>
<thead>
<tr>
<th>Method</th>
<th>Finetuning tasks</th>
<th># Params</th>
<th>Discriminative tasks</th>
<th>Generative tasks</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td>VQA Karpathy test Acc</td>
<td>COCO Caption Karpathy test CIDEr</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>GQA test-dev Acc</td>
<td>Multi30K En-De test2018 BLEU</td>
</tr>
<tr>
<td>VQA</td>
<td>single task</td>
<td>7P</td>
<td>67.9</td>
<td>116.1</td>
</tr>
<tr>
<td>VQA</td>
<td>all tasks</td>
<td>P</td>
<td>67.2</td>
<td>110.8</td>
</tr>
<tr>
<td>VQA</td>
<td></td>
<td></td>
<td>60.0</td>
<td>57.5</td>
</tr>
<tr>
<td>VQA</td>
<td></td>
<td></td>
<td>73.6</td>
<td>116.1</td>
</tr>
<tr>
<td>VQA</td>
<td></td>
<td></td>
<td>71.3</td>
<td>110.8</td>
</tr>
<tr>
<td>VQA</td>
<td></td>
<td></td>
<td>55.3</td>
<td>38.6</td>
</tr>
</tbody>
</table>

Similar performance with $1/7^{th} = 14\%$ parameters!

- Also performs better on rare/unseen categories!
- Many follow-up useful works on unification:
  - e.g., SimVLM, Flamingo, OFA, UnifiedIO, BLIP-2, CoCa, PaLi, etc.

---

Wang et al., 2021, SimVLM: Simple Visual Language Model Pretraining with Weak Supervision
Alayrac et al., 2022, Flamingo: a Visual Language Model for Few-Shot Learning
Wang et al., 2022, OFA: Unifying Architectures, Tasks, and Modalities Through a Simple Sequence-to-Sequence Learning Framework
Lu et al., 2022, Unified-IO: A Unified Model for Vision, Language, and Multi-Modal Tasks
Li et al., 2023, BLIP-2: Bootstrapping Language-Image Pre-training with Frozen Image Encoders and Large Language Models
Yu et al., 2022, CoCa: Contrastive Captioners are Image-Text Foundation Models
Chen et al., 2023, PaLi: A Jointly-Scaled Multilingual Language-Image Model
Part 1: Unified/Universal Multimodal Learning

**VL-T5 (ICML 2021)**
all multimodal tasks via text generation

**TVLT (NeurIPS 2022)**
video modeling without text (audio as images)

**UDOP (CVPR 2023)**
document image/text/layout with single architecture

**CoDi (NeurIPS 2023)**
generating any-to-any input-output modality combination
TVLT: Textless Vision-Language Transformer

- Unified ViT-style patch embeddings for both video and audio inputs
- MAE-style enc-dec: multimodal joint encoder; decoder weights are shared for video & audio decoding
- Two objectives: (1) masked autoencoding, (2) contrastive learning

[Tang et al., NeurIPS 2022]
TVLT: Textless Vision-Language Transformer

- Results: Audio-based TVLT (w/o any ASR/tokenization/text modules!) performs competitively with text-based model on diverse tasks: image-retrieval, video-retrieval, visual-QA, multimodal sentiment analysis, emotion analysis (while also being much more efficient = 28x faster inference, 1/3 #parameters)!

[Tang et al., NeurIPS 2022]
Part 1: Unified/Universal Multimodal Learning

**VL-T5 (ICML 2021)**
all multimodal tasks via text generation

**TVLT (NeurIPS 2022)**
video modeling without text (audio as images)

**UDOP (CVPR 2023)**
document image/text/layout with single architecture

**CoDi (NeurIPS 2023)**
generating any-to-any input-output modality combination
UDOP: Unifying Vision, Text, Layout for Universal Document Processing

- Unifies text, image, layout modalities (w/o specialized modules incl. OCR or layout-specific architectures) with varied task formats, doing document understanding + generation/editing via masked image reconstruction.

Vision-Text-Layout Transformer

- Text reconstruction with layout. `<text_layout_0>` Retail: Week of March 14, 1994
- Visual text recognition. `<text_0>` `<100><350><118><372>` `<text_0>` Week of March 14, 1994
- Question answering. What is the date?
- Layout modeling. `<layout_0>` Ship Date `<layout_0>` to Retail: Week of March 14, 1994
- Layout analysis. Title
- Masked image reconstruction. Ship Date to Retail: Week of March 14, 1994

Vision Outputs
- Ship Date ...

Text Outputs
- `<100><350><118>` `<372>` ...

Layout Outputs
- `<text_layout_0>` Ship Date `<0><16><2><20>`
- `<text_0>` Ship Date
- Week of March 14, 1994
- `<layout_0>` `<100><350><118><372>`
- Title `<20><50><40><80>`

[Amended: Tang et al., CVPR 2023]
UDOP: Unifying Vision, Text, Layout for Universal Document Processing

- Unifies text, image, layout modalities (\textit{w/o specialized modules incl. OCR or layout-specific architectures}) with varied task formats, doing document understanding + generation/editing via masked image reconstruction.

- State-of-the-art & rank-1 on 8 DocAI tasks / DUE-benchmark, e.g., document-VQA, table-NLI, table-QA, doc-IE, etc. across diverse data domains like finance reports, academic papers, and websites.
UDOP: Unifying Vision, Text, Layout for Universal Document Processing

[Image of a sample document with red boxes highlighting selected text]
Part 1: Unified/Universal Multimodal Learning

**VL-T5 (ICML 2021)**
all multimodal tasks via text generation

**TVLT (NeurIPS 2022)**
video modeling without text (audio as images)

**UDOP (CVPR 2023)**
document image/text/layout with single architecture

**CoDi (NeurIPS 2023)**
generating any-to-any input-output modality combination
CoDi: Any-to-Any Multimodal Generation

CoDi can jointly generate any-to-any combinations from video, image, audio, and text, via composable diffusion.

https://codi-gen.github.io/ [Tang et al., NeurIPS 2023]
CoDi: Any-to-Any Multimodal Generation

- New generative-AI foundation model that allows **any combination of input modalities & generates any combination of output modalities** (text, audio, image, video) – can help create diverse ‘many-modal’ stories using different types of inputs on the storyboard!

- BUT training such a model presents **significant costs**, as the # combinations for input and output modalities scales **exponentially** & training datasets **missing** for many combinations of modalities.

- We propose “Bridging Alignment” strategy to **efficiently model the exponential number** of input-output combinations with a **linear number** of training objectives.

- Allows CoDi to freely condition on any input combination+generate any group of modalities, even if **not present in the training data**.

[https://codi-gen.github.io/](https://codi-gen.github.io/) [Tang et al., NeurIPS 2023]
CoDi: Any-to-Any Multimodal Generation

https://codi-gen.github.io/
CoDi: Any-to-Any Multimodal Generation

- **Stage 1**: We train a latent diffusion model (LDM) for each modality. They can be trained independently, ensuring high-quality generation for each modality. For conditional generation, e.g., audio+language→image, the input modalities are projected into a shared feature space, and the output LDM attends to this combination of input features.

- This multimodal conditioning mechanism prepares the diffusion model to condition on any combination of modalities without directly training for such settings.
CoDi: Any-to-Any Multimodal Generation

- **Stage 2**: We add a cross-attention module to each LDM and an environment encoder to project the LDM latent variables into a shared/mixed space.

- This enables CoDi to seamlessly mix/generate any group of output modalities, w/o training on all generation combinations (with linear # training objectives).

[Tang et al., NeurIPS 2023]

https://codi-gen.github.io/
CoDi: Any-to-Any Multimodal Generation

Audio + Image → Text + Image

https://codi-gen.github.io/
CoDi: Any-to-Any Multimodal Generation

Audio + Image → Text + Image

"Playing piano in a forest."

https://codi-gen.github.io/
CoDi: Any-to-Any Multimodal Generation

Text + Image → Video

"Eating on a coffee table."

https://codi-gen.github.io/ [Tang et al., NeurIPS 2023]
CoDi: Any-to-Any Multimodal Generation

Text + Image → Video

"Eating on a coffee table."

https://codi-gen.github.io/ [Tang et al., NeurIPS 2023]
CoDi-2: Interleaved & Interactive Any-to-Any Generation (allows Reasoning)

Composition and Concept Learning

Learn the subject in . Generate with it on the concept represented by

Multimodal Editing

Edit this image in the vibe of (Raining Sound)

Exemplar Learning

What's the edit between and ? Apply it to the image and tell us what the effect is.

CoDi-2: In-Context, Interleaved, and Interactive Any-to-Any Generation

https://codi-2.github.io/
CTRL-Adapter: Efficient+Versatile Adaptation of Any Control to Any Diffusion

[Lin et al., 2024]

https://ctrl-adapter.github.io/
Talk Outline

A journey of multimodal generative models for enhancing their unification, interpretable planning/programming, evaluation:

• **Unified/Universal Multimodal Learning** (for Generalizability, Shared Knowledge, Efficiency)
  • VLT5: Unifying Vision-and-Language Tasks via Text Generation [ICML 2021]
  • TVLT: Textless Vision-Language Transformer [NeurIPS 2022]
  • UDOP: Unifying Vision, Text, and Layout for Universal Document Processing [CVPR 2023]

• **Interpretable Multimodal Generation via LLM Planning/Programming Agents** (for Understanding, Control, Faithfulness, OOD)
  • VideoDirectorGPT: Consistent Multi-Scene Video Generation via LLM-Guided Planning [COLM 2024]
  • DiagrammerGPT: Generating Diagrams via LLM Planning [COLM 2024]; EnvGen: Adapting Environments via LLMs for Training Embodied Agents [COLM 2024]

• **Evaluation of Multimodal Generation Models** (of Fine-grained Skills, Faithfulness, Social Biases)
  • DALL-Eval: Probing the Reasoning Skills and Social Biases of Text-to-Image Generation Models [ICCV 2023]
  • Davidsonian Scene Graph: Improving Reliability in Fine-grained Evaluation for Text-to-Image Generation [ICLR 2024]

• **Next Big Challenges**: trade-offs, structure, non-verbal, interaction, reasoning, causality, long-distance fine-grained evaluation, efficiencies
Part 2: Interpretable Multimodal Generation with LLM Planning/Reasoning

**VPGen (NeurIPS 2023)**
LLM Planning for Image Generation

**VideoDirectorGPT (COLM 2024)**
LLM Planning for Multi-Scene, Consistent Video Generation

**DiagrammerGPT (COLM 2024)**
LLM Planning for Open-Domain Diagram Generation
Background: Text-to-Image Generation with Blackbox Models

A truck is behind a motorcycle.

two Pikachu on a table.

Blackbox Text-to-Image Generation (e.g., DALL-E, Imagen, Stable Diffusion)

[Cho et al., NeurIPS 2023]
Background: Text-to-Image Generation with Blackbox Models

Good visual quality! But important semantic issues...
• lack of fine-grained layout planning/control 😞
• lack of interpretability behind generation process
• lack of faithfulness to input (incl. hallucinations and OOD scenarios)

[Cho et al., NeurIPS 2023]
VPGen: Visual Programming for Step-by-Step T2I Generation

two Pikachus
on a table
Given an image caption, determine objects and their counts to draw an image.

Caption: two Pikachus on a table

| pikachu | (2)     |
| table   | (1)     |
Given an image caption, determine objects and their counts to draw an image. Caption: two Pikachus on a table

Objects:
- pikachu (2)
- table (1)

Visualized Layout:
- pikachu (x1,y1,x2,y2)
- table (x1,y1,x2,y2)
Given an image caption, determine objects and their counts to draw an image.

Caption: two Pikachus on a table

Objects:
pikachu (2) table (1)
Skill-based Results
Our VPGen shows improved spatial control

- Generation via layout programs promotes better **understanding+planning** of structure/scale/spatial relations, including **out-of-distribution/unseen** cases (also allows **explicit control** over these properties via manual, interpretable corrections of unfaithful parts)!

<table>
<thead>
<tr>
<th>Model</th>
<th>VPEVAL Skill Score (%)</th>
<th>Count “3 boats”</th>
<th>Spatial “a truck is behind a motorcycle”</th>
<th>Scale “a remote that is bigger than a cat”</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stable Diffusion v1.4</td>
<td>97.3</td>
<td>47.4</td>
<td>22.9</td>
<td>11.9</td>
</tr>
<tr>
<td>Stable Diffusion v2.1</td>
<td>96.5</td>
<td>53.9</td>
<td>31.3</td>
<td>14.3</td>
</tr>
<tr>
<td>Karlo</td>
<td>95.0</td>
<td>59.5</td>
<td>24.0</td>
<td>16.4</td>
</tr>
<tr>
<td>minDALL-E</td>
<td>79.8</td>
<td>29.3</td>
<td>7.0</td>
<td>6.2</td>
</tr>
<tr>
<td>DALL-E Mega</td>
<td>94.0</td>
<td>45.6</td>
<td>17.0</td>
<td>8.5</td>
</tr>
<tr>
<td>VPGen (F30)</td>
<td>96.8</td>
<td>55.0</td>
<td>39.0</td>
<td>23.3</td>
</tr>
<tr>
<td>VPGen (F30+C+P)</td>
<td>96.8</td>
<td><strong>72.2</strong></td>
<td><strong>56.1</strong></td>
<td><strong>26.3</strong></td>
</tr>
</tbody>
</table>

Large improvements on structural control:
- Counting
- Spatial relation
- Relative size/scale comparison

(OOD/unseen scenes)

[Cho et al., NeurIPS 2023]

https://vp-t2i.github.io/
VideoDirectorGPT: Consistent Multi-Scene Video Generation via LLM-Guided Planning/Reasoning

VideoDirectorGPT

Prompt
A hungry cat is finding food

Video Planner (GPT-4)
- Scene Descriptions
- Entities (names + layouts)
- Backgrounds
- Consistency Groupings

Video Plan

Video Generator (Layout2Vid)
- Spatial-Temporal Blocks
- Guided 2D Attention

Multi-Scene Video

https://videodirectorgpt.github.io/

[Lin et al., COLM 2024]
A hungry cat is finding food

**Video Planner (GPT-4)**

**Scene Description**

Scene 1

A cat is lying down on a bed

Frame 1: 
{'a fluffy Siamese cat': [0.25, 0.25, 1.00, 0.70], 'a plush beige bed': [0.00, 0.50, 1.00, 1.00]}

Frame 2: 
{'a fluffy Siamese cat': [0.25, 0.25, 1.00, 0.70], 'a plush beige bed': [0.00, 0.50, 1.00, 1.00]}

...
A hungry cat is finding food

Video Planner (GPT-4)

Video Plan

Video Generator (Layout2Vid)

Scene 1

Entities (names + layouts) with Consistency/Coreference Grouping

Frame 1: {'a fluffy Siamese cat': [0.25, 0.25, 1.00, 0.70], 'a plush beige bed': [0.00, 0.50, 1.00, 1.00]}
Frame 2: {'a fluffy Siamese cat': [0.25, 0.25, 1.00, 0.70], 'a plush beige bed': [0.00, 0.50, 1.00, 1.00]}

Scene 2

Frame 1: {'a fluffy Siamese cat': [0.55, 0.25, 0.85, 0.55], 'a plush beige bed': [0.00, 0.60, 1.00, 1.00]}
Frame 2: {'a fluffy Siamese cat': [0.50, 0.30, 0.80, 0.60], 'a plush beige bed': [0.00, 0.60, 1.00, 1.00]}

Scene 3

Background

Person

VideoDirectorGPT

https://videodirectorgpt.github.io/

[Lin et al., COLM 2024]
A hungry cat is finding food.

**Scene 1**

- **Frame 1**: 
  - A fluffy Siamese cat: [0.25, 0.25, 1.00, 0.70]
  - Plush beige bed: [0.00, 0.50, 1.00, 1.00]

- **Frame 2**: 
  - A fluffy Siamese cat: [0.25, 0.25, 1.00, 0.70]
  - Plush beige bed: [0.00, 0.50, 1.00, 1.00]

---

**Scene 2**

- **Frame 1**: 
  - A fluffy Siamese cat: [0.55, 0.25, 0.85, 0.55]
  - Plush beige bed: [0.00, 0.60, 1.00, 1.00]

- **Frame 2**: 
  - A fluffy Siamese cat: [0.50, 0.30, 0.80, 0.60]
  - Plush beige bed: [0.00, 0.60, 1.00, 1.00]

---

**Scene 3**

- **Frame 1**: 
  - A fluffy Siamese cat: [0.15, 0.20, 0.40, 0.45]
  - Gourmet cat snack: [0.50, 0.45, 0.80, 0.65]

- **Frame 2**: 
  - A fluffy Siamese cat: [0.35, 0.30, 0.60, 0.55]
  - Gourmet cat snack: [0.50, 0.45, 0.80, 0.65]

---

**Background**

- Bedroom
- Bedroom
- Kitchen

**Video Planner**

- **Scene Description**
  - Scene 1: A cat is lying down on a bed
  - Then she gets up
  - She goes to the kitchen and eats a snack

---

**Entities (names + layouts) with Consistency/Coreference Grouping**

- Frame 1: 
  - A fluffy Siamese cat: [0.25, 0.25, 1.00, 0.70]
  - Plush beige bed: [0.00, 0.50, 1.00, 1.00]

- Frame 2: 
  - A fluffy Siamese cat: [0.25, 0.25, 1.00, 0.70]
  - Plush beige bed: [0.00, 0.50, 1.00, 1.00]

---

**VideoDirectorGPT**

- **Video Planner (GPT-4)**
- **Video Plan**
- **Video Generator (Layout2Vid)**
A hungry cat is finding food.

Scene 1
A cat is lying down on a bed.
Frame 1: {'a fluffy Siamese cat': [0.25, 0.25, 1.00, 0.70], 'a plush beige bed': [0.00, 0.50, 1.00, 1.00]}
Frame 2: {'a fluffy Siamese cat': [0.25, 0.25, 1.00, 0.70], 'a plush beige bed': [0.00, 0.50, 1.00, 1.00]}
...
Scene 2
Then she gets up.
Frame 1: {'a fluffy Siamese cat': [0.55, 0.25, 0.85, 0.55], 'a plush beige bed': [0.00, 0.60, 1.00, 1.00]}
Frame 2: {'a fluffy Siamese cat': [0.50, 0.30, 0.80, 0.60], 'a plush beige bed': [0.00, 0.60, 1.00, 1.00]}
...
Scene 3
She goes to the kitchen and eats a snack.
Frame 1: {'a fluffy Siamese cat': [0.15, 0.20, 0.40, 0.45], 'gourmet cat snack': [0.50, 0.45, 0.80, 0.65]}
Frame 2: {'a fluffy Siamese cat': [0.35, 0.30, 0.60, 0.55], 'gourmet cat snack': [0.50, 0.45, 0.80, 0.65]}
...

Background
Bedroom
Scene: [1, 2, 3]
Kitchen
Scene: [3]
A hungry cat is finding food. Scene 1

Scene 1: A cat is lying down on a bed. Frame 1: {'a fluffy Siamese cat': [0.25, 0.25, 1.00, 0.70], 'a plush beige bed': [0.00, 0.50, 1.00, 1.00]}
Frame 2: {'a fluffy Siamese cat': [0.25, 0.25, 1.00, 0.70], 'a plush beige bed': [0.00, 0.50, 1.00, 1.00]}
...

Scene 2: Then she gets up. Frame 1: {'a fluffy Siamese cat': [0.55, 0.25, 0.85, 0.55], 'a plush beige bed': [0.00, 0.60, 1.00, 1.00]}
Frame 2: {'a fluffy Siamese cat': [0.50, 0.30, 0.80, 0.60], 'a plush beige bed': [0.00, 0.60, 1.00, 1.00]}
...

Scene 3: She goes to the kitchen and eats a snack. Frame 1: {'a fluffy Siamese cat': [0.15, 0.20, 0.40, 0.45], 'gourmet cat snack': [0.50, 0.45, 0.80, 0.65]}
Frame 2: {'a fluffy Siamese cat': [0.35, 0.30, 0.60, 0.55], 'gourmet cat snack': [0.50, 0.45, 0.80, 0.65]}
...

Bedroom

Kitchen

VideoDirectorGPT

Video Planner (GPT-4)

Video Plan

Video Generator (Layout2Vid)

https://videodirectorgpt.github.io/ [Lin et al., COLM 2024]
Understanding of Basic Physics

Gravity

A stone thrown into the sky

Perspective

A car is approaching from a distance

https://videodirectorgpt.github.io/  
[Lin et al., COLM 2024]
Movement of Static Objects vs. Dynamic Objects

“A {bottle/airplane} moving from left to right.”

Static objects
-> Movements of Camera

Objects that can move
-> Movements of Object (+ Camera)

[https://videodirectorgpt.github.io/]
[Lin et al., COLM 2024]
Multi-Sentence to Multi-Scene Video (Coref-SV)

Scene 1: mouse is holding a book and makes a happy face.
Scene 2: he looks happy and talks.
Scene 3: he is pulling petals off the flower.
Scene 4: he is ripping a petal from the flower.
Scene 5: he is holding a flower by his right paw.
Scene 6: one paw pulls the last petal off the flower.
Scene 7: he is smiling and talking while holding a flower on his right paw.

ModelScopeT2V

❌ fails to keep “mouse” through all scenes

VideoDirectorGPT (Ours)

✔️ the “mouse” is consistent through all scenes + layout control
(also helps plan+generate OOD/unseen affordances/scenes)

https://videodirectorgpt.github.io/

[Lin et al., COLM 2024]
Single Sentence to Multi-Scene Video (HiREST)

make a strawberry surprise

**GPT-4 generated sub-scene descriptions:**
- a young man in a red apron washes ripe red strawberries in a silver sink
- a young man in a red apron carefully cuts the strawberries on a wooden chopping board with a sharp knife
- a young man in a red apron places cut strawberries, banana, and Greek yogurt into an electric blender
- a young man in a red apron blends ingredients together until smooth in an electric blender
- a young man in a red apron pours the smoothie into a tall glass
- a young man in a red apron places a scoop of vanilla ice cream on top of the smoothie in a tall glass
- a young man in a red apron places a strawberry on top of the ice cream for garnishing
- a young man in a red apron serves the Strawberry Surprise on a ceramic plate

**ModelScopeT2V**

- no actual process shown on how to “make” the strawberry surprise

**VideoDirectorGPT (Ours)**

- step-by-step + consistent process on how to “make” the strawberry surprise

[https://videodirectorgpt.github.io/](https://videodirectorgpt.github.io/)

[Lin et al., COLM 2024]
Human-in-the-Loop Video Editing+Control

Make the horse smaller

Add “grassland” background

Add “night street” background

https://videodirectorgpt.github.io/

[Lin et al., COLM 2024]
Scene 1: a <S> then gets up from a plush beige bed.
Scene 2: a <S> goes to the cream-colored kitchen and eats a can of gourmet snack.
Scene 3: a <S> sits next to a large floor-to-ceiling window.
Quantitative Evaluation & Human Evaluation

<table>
<thead>
<tr>
<th>Method</th>
<th>VPEval Skill-based</th>
<th>ActionBench-Direction</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Object</td>
<td>Count</td>
</tr>
<tr>
<td>ModelScopeT2V</td>
<td>89.8</td>
<td>38.8</td>
</tr>
<tr>
<td>VIDEODIRECTORGPT (Ours)</td>
<td><strong>97.1</strong></td>
<td><strong>77.4</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Method</th>
<th>ActivityNet Captions</th>
<th>Coref-SV</th>
<th>HiREST</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>FVD (↓)</td>
<td>FID (↓)</td>
<td>Consistency (↑)</td>
</tr>
<tr>
<td>ModelScopeT2V</td>
<td>980</td>
<td>18.12</td>
<td>46.0</td>
</tr>
<tr>
<td>ModelScopeT2V (with GT co-reference; oracle)</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>VIDEODIRECTORGPT (Ours)</td>
<td><strong>805</strong></td>
<td><strong>16.50</strong></td>
<td><strong>64.8</strong></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Evaluation category</th>
<th>Human Preference (%) ↑</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>VIDEODIRECTORGPT (Ours)</td>
</tr>
<tr>
<td>Quality</td>
<td>54</td>
</tr>
<tr>
<td>Text-Video Alignment</td>
<td>54</td>
</tr>
<tr>
<td>Object Consistency</td>
<td>58</td>
</tr>
</tbody>
</table>

[Lin et al., COLM 2024]

https://videodirectorgpt.github.io/
A diagram showing the Earth revolve around the sun four times, one of each solstice and equinox. It also ...

Diagram Plan from GPT-4

Entities:
images [earth (I0), earth (I1), ...]
text labels [“Vernal...” (T0), ...]
Entity Locations:
I0: [39, 11, 17, 21], ...
Entity Relations:
I0 has an arrow to I1; ...

Initial Plan Visualization

Refined Plan After Feedback

DiagramGLIGEN

with Text Label Rendering

https://diagrammergpt.github.io/ [Zala et al., COLM 2024]
A diagram showing the earth, moon, and sun with text labels.

Input Prompt

LLM Diagram Planner

Initial Plan Generation

Initial Diagram

User Edits

User Refined Plan

New Diagram

Let me:
(1) align the text labels
(2) move all the objects up and make them a bit larger

[Zala et al., COLM 2024]

https://diagrammergpt.github.io/
EnvGen: LLM-Planned Adaptive Environment Generation for Training Agents

(a) RL agents: Explore skills with rewards

(b) LLM agents: Explore skills with LLM knowledge

(c) EnvGen (Ours): Bootstrap skill exploration with LLM-generated environments

Step 1: Generate training environments
Loop $N_{\text{Cycle}}$ times

Step 2: Train small agent in generated environments

Step 3: Train and measure agent performance in the original environment

Step 4: Share agent performance with LLM

https://envgen-llm.github.io/ [Zala et al., COLM 2024]
EnvGen: LLM-Planned Adaptive Environment Generation for Training Agents

--

### Game and task description
You are an environment designer agent for a game called “Crafters”. Your job is to design a few environments which can be used to teach an agent how to play...

### Game Objectives
Here is a list of things an agent would need to learn how to do:
- collect_coal, collect_diamond...

### Controllable Simulator Settings
Here is a list of parameters you can control when making an environment:
- target_biome: grassland | mountain | beaches | natural...

Here is a list of items the agent can start with:
- wood_pickaxe: 0-1...

### Simulator Constraints
Here is a list of constraints:
- natural biome will set the environment to have all the biomes
- coal, iron, and diamond can only be found in a mountain biome...

### Output Template
Output in the following format:
```
Environment 1:
```
```
  json
  {
    "environment_settings": {
      "target_biome": "grassland",
      "coal_rarity": "common",
      "iron_rarity": "common",
      "diamond_rarity": "rare"
    },
    "inventory_settings": {
      "wood": 3,
      "stone": 0,
      "wood_pickaxe": 1
    }
  }
```

Purpose: The agent can learn to collect wood and craft items like a wood sword or a table...

### Feedback to update environments
Those environments resulted in the agent improving up to these scores:
- collect_coal: 38% +/- 6%
- defeat_skeleton: 10% +/- 4%
- make_stone_pickaxe: 31% +/- 3%
- ...

Could you generate new environments based on these scores?

---

Step 1: Generate training environments

---

Step 2: Train small agent in generated environments

---

Step 3: Train and measure agent performance in the original environment

---

Step 4: Share agent performance with LLM

---

Env 1: Basic Resource Collection and Crafting
```
...
```

Env 2: Advanced Resource Collection and Combat Training
```
...
```

Env 3: Survival and Crafting Mastery
```
...
```

---

Small RL Agent

---

Original Environment

---

Zala et al., COLM 2024

---

https://envgen-llm.github.io/
Talk Outline

A journey of multimodal generative models for enhancing their unification, interpretable planning/programming, evaluation:

- **Unified/Universal Multimodal Learning** (for Generalizability, Shared Knowledge, Efficiency)
  - VLT5: Unifying Vision-and-Language Tasks via Text Generation [ICML 2021]
  - TVLT: Textless Vision-Language Transformer [NeurIPS 2022]

- **Interpretable Multimodal Generation via LLM Planning/Programming Agents** (for Understanding, Control, Faithfulness, OOD)
  - VideoDirectorGPT: Consistent Multi-Scene Video Generation via LLM-Guided Planning [COLM 2024]
  - DiagrammerGPT: Generating Diagrams via LLM Planning [COLM 2024]; EnvGen: Adapting Environments via LLMs for Training Embodied Agents [COLM 2024]

- **Evaluation of Multimodal Generation Models** (of Fine-grained Skills, Faithfulness, Social Biases)
  - DALL-Eval: Probing the Reasoning Skills and Social Biases of Text-to-Image Generation Models [ICCV 2023]
  - Davidsonian Scene Graph: Improving Reliability in Fine-grained Evaluation for Text-to-Image Generation [ICLR 2024]

- **Next Big Challenges**: trade-offs, structure, non-verbal, interaction, reasoning, causality, long-distance fine-grained evaluation, efficiencies
Part 3: Evaluation of Multimodal Generation

Heusel et al., 2017, GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium
Xu et al., 2018, AttnGAN: Fine-Grained Text to Image Generation with Attentional Generative Adversarial Networks
Hessel et al., 2021, CLIPScore: A Reference-free Evaluation Metric for Image Captioning
Hinz et al., 2022, Semantic Object Accuracy for Generative Text-to-Image Synthesis
VPEval: Visual Programming for Explainable T2I Evaluation

**Text-to-Image Evaluation**

two Pikachus on a table

**Evaluation Model**
(e.g., CLIP, BLIP-2)

**Score**

- How did they compute this score?
- What does the score mean/compare?
- Which parts of the generated image incorrect/unfaithful to the prompt?

https://vp-t2i.github.io/

[Cho et al., NeurIPS 2023]
VPEval: Visual Programming for Explainable T2I Evaluation
Open-ended Evaluation

Open-ended Interpretable Evaluation Program

# Task description + module description
Given an image description, generate programs that verifies if the image description is correct.

# In-context examples
Description: A man posing for a selfie in a jacket and bow tie.

... Example text prompt
objectEval(image, 'man');
vqa(image, 'who is posing for a selfie?', 'man,woman,boy,girl', 'man')

... Example evaluation program

# New text prompt
Description: A white slope covers the background, while the foreground features a grassy slope with several rams grazing and one measly and underdeveloped evergreen in the foreground.
VPEval: Visual Programming for Explainable T2I Evaluation

Open-ended Evaluation

Open-ended Interpretable Evaluation Program

```python
# Task description + module description
Given an image description, generate programs that verifies if the image description is correct.
...

# In-context examples
Description: A man posing for a selfie in a jacket and bow tie.
...
objectEval(image, 'man');
vqa(image, 'who is posing for a selfie?', 'man,woman,boy,girl', 'man')
...

# New text prompt
Description: A white slope covers the background, while the foreground features a grassy slope with several rams grazing and one measly and underdeveloped evergreen in the foreground.
```

# Generated Program
```python
objectEval(image, 'ram');
objectEval(image, 'evergreen');
countEval(image, 'ram', '>1');
countEval(image, 'evergreen', '==1');
vqa(image, 'what is in the foreground?', 'grassy slope,beach,field,forest', 'grassy slope');
...
```

https://vp-t2i.github.io/

[Cho et al., NeurIPS 2023]
VPEval: Visual Programming for Explainable T2I Evaluation

Open-ended Evaluation

Open-ended Interpretable Evaluation Program

# Task description + module description
Given an image description, generate programs that verifies if the image description is correct.

# In-context examples
Description: A man posing for a selfie in a jacket and bow tie.

... objectEval(image, 'man');
vqa(image, 'who is posing for a selfie?', 'man,woman,boy,girl', 'man')

# New text prompt
Description: A white slope covers the background, while the foreground features a grassy slope with several rams grazing and one measly and underdeveloped evergreen in the foreground.

# Generated Program
objectEval(image, 'ram');
objectEval(image, 'evergreen');
countEval(image, 'ram', '>1');
countEval(image, 'evergreen', '==1');
vqa(image, 'what is in the foreground?', 'grassy slope,beach,field,forest', 'grassy slope');

Visual + Textual Explanations of Errors/Hallucinations

Incorrect ✗

no "ram" object found.

Correct ✓

"evergreen" object found.

Incorrect ✗

there are 8 "evergreen" objects, not 1.

Correct ✓

Q: "what is in the foreground?" A: grassy slope.

ChatGPT

https://vp-t2i.github.io/

[Cho et al., NeurIPS 2023]
Davidsonian Scene Graph: Improving Reliability in Fine-grained Evaluation for T2I

Complex, non-atomic questions

Q: “is there a red motorcycle?”

Invalid questions

Q1: “is there a motorcycle?” → A: No
Q2: “is the motorcycle red?” → A: Yes

Invalid question;
The question checks multiple aspects at once!

= “is there a motorcycle?” → Yes +
“is the motorcycle red?” → No

Q2 is invalid;
If there is not motorcycle, no need to check its color!

[Cho et al., ICLR 2024]
Questions w/ desired properties (following Davidsonian formal semantics):
- Atomic
- Unique
- Full semantic coverage
- Valid dependencies

Answering Questions, while avoiding answering the invalid questions

Question Answering (QA)

Score: 6/7 = 0.86

Score: 3/7 = 0.43

[Cho et al., ICLR 2024]
DALL-Eval: Measuring Social Biases

How skewed are the distributions?

Template | [G] who works as a/an [P]
---|---
Gender [G] | a person / a man / a woman

- accountant
- animator
- architect
- assistant
- athlete
- author
- baker
- biologist
- builder
- butcher
- career counselor
- caretaker
- chef
- civil servant
- clerk
- comic book writer
- company director
- cook
- decorator
- dentist
- designer
- diplomat
- director
- doctor
- economist
- editor
- electrician
- engineer
- executive
- farmer
- film director
- flight attendant
- garbage collector
- geologist
- hairdresser
- jeweler
- journalist
- judge
- juggler
- lawyer
- lecturer
- lexicographer
- librarian
- assistant
- magician
- makeup artist
- manager
- miner
- musician
- nurse
- optician
- painter
- personal assistant
- photographer
- pilot
- plumber
- police officer
- politician
- porter
- prison officer
- professor
- puppeteer
- receptionist
- sailor
- salesperson
- scientist
- secretary
- shop assistant
- sign language interpreter
- singer
- soldier
- solicitor
- surgeon
- tailor
- teacher
- translator
- travel agent
- trucker
- TV presenter
- veterinarian
- waiter
- web designer
- writer
Conclusion + Big Challenges / Research Directions

• **Trade-off** of blackbox **pretraining** vs. **modular structure** (incl. faithfulness, efficiency, interpretability/understanding, human-in-loop/control, OOD, fairness/bias, privacy)?

• **Other modalities** (non-verbal gesture/gaze, action-interaction)?

• **Long-distance** text/video understanding+generation, **causal/counterfactual**?

• **Fine-grained** evaluation of **skills/consistency/bias/faithfulness+hallucination**?

• **Continual learning** when new/unseen information keeps coming?

• **Unlearning** of outdated/wrong/unsafe/private information?

• **Efficiency** w.r.t. many axes: time, storage, memory, carbon footprint, etc.?
Thank you!

Webpage: http://www.cs.unc.edu/~mbansal/
Email: mbansal@cs.unc.edu

MURGe-Lab: https://murgelab.cs.unc.edu/
(thanks to our awesome students for all the work I presented!)

We are hiring PhD students + Postdocs!