**lliGym**: Natural Language Visual Reasoning with Reinforcement Learning
Anne Wu, Kianté Brantley, Noriyuki Kojima, and Yoav Artzi
https://lil.nlp.cornell.edu/lliGym

**lliGym** is a new RL benchmark for studying natural language visual reasoning with interaction.

- Human-written highly-compositional natural language with diverse reasoning challenges
- Each statement describes a large set of valid goal states, creating a reward computation challenge
- Policy must learn complex language-conditioned goal equivalence to generalize well
- Contemporary methods show non-trivial performance, but remain far from human performance

---

**Designing lliGym: Making NLVR Interactive**

NLVR (Suhr et al. 2017) is a supervised benchmark for visual reasoning with rich compositional reasoning.

### Two Types of Environments

<table>
<thead>
<tr>
<th>Tower</th>
<th>Scatter</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADD(BOX, COLOR) REMOVE(BOX) STOP</td>
<td>ADD(x, y, SHAPE, COLOR, SIZE) REMOVE(x, y) STOP</td>
</tr>
<tr>
<td>13 actions</td>
<td>1,064,001 actions</td>
</tr>
</tbody>
</table>

Optional grid simplification

### Two Types of Starting States

<table>
<thead>
<tr>
<th>Scratch</th>
<th>Fliplt</th>
</tr>
</thead>
<tbody>
<tr>
<td>One of the grey box has exactly six objects</td>
<td>There is no square closely touching the bottom of a box</td>
</tr>
<tr>
<td>TRUE</td>
<td>FALSE</td>
</tr>
</tbody>
</table>

**Scratch**
- Start from an empty state
- Goal: satisfy statement
- Target boolean: always true

**Fliplt**
- Start from an NLVR image
- Goal: flip boolean value
- Target boolean: true or false

---

**Annotation for Reward Computation**

**Key challenge:** evaluating a state’s correctness requires resolving language meaning

**Solution:** annotate all 2,661 statements with executable formal meaning representations

**Example of Language Statement Annotation**

**Statement:**
the grey box with least number of objects contains only one black object

**Python program annotation:**
```python
count = filter_obj_min(
    all_boxes,
    key=lambda x: x
    count(x).all_items_in_box(),
    lambda y: y black(y)) == 1
```

---

**Experiments and Results**

**Models:** C3+BERT (CNN + BERT) and ViLT

**Learning algorithms:** PPO and PPO+SF

### Accuracy

- Various levels of difficulty
- Overall: non-trivial performance
- Stop forcing (PPO+SF) helps: illustrates the exploration challenge

### Mean Reward

- Reward shows similar trends to accuracy
- Gap to expert illustrates a long way to go
- Random policy: shows task difficulty

**Training curves**

- Alternative reward using NLVR images shows no effective learning
- Exact reward computation is critical, especially with the large set of valid goal states

---

More in the paper

- Finer-grained analysis and rollout statistics (trajectory length, action types, etc.)
- Semantic and syntactic analysis
- Analysis of error patterns