Learning List Concepts through Program Induction

**Problem:** Model human learning of structured concepts in two online experiments using a game-based paradigm.

**Approach:** Search over rewrite systems (top) to learn concepts (bottom).

```python
def search_data(0, n=1000, n_steps=50, confidence=2/3):
    dataset = [1, h, score = h, score[h])
    for _ in range(n):
        h.next = random()
        score.next = score[h.next]
    h.score = h.score.next
    best_h = h.take.top
    data.append(h.score)
    return data
```

**Results:** This approach accurately models overall (left) and per-trial performance (middle) in Exp. 1, and predicts curriculum training outperforming random training in Exp. 2.

Learning to Infer Program Sketches

**Problem:** Build systems which write code automatically from the kinds of specifications humans can easily provide, such as examples and natural language instruction.

**Approach:** Learn Program sketches, which serve as an intermediate between pattern recognition and explicit search approaches.

```python
def define(search_data, n=1000, n_steps=50, confidence=2/3):
    dataset = [1, h, score = h, score[h])
    for _ in range(n):
        h.next = random()
        score.next = score[h.next]
    h.score = h.score.next
    best_h = h.take.top
    data.append(h.score)
    return data
```

**Results:** This approach can synthesize programs more accurately than baselines in a variety of domains: list processing from examples (left), text editing from examples (middle), and character and list manipulation from language description (right).

Logical Rule Induction via Neural Theorem Proving

**Problem:** Transform a knowledge base of observations (left) into a logical theory containing generalized predicates and core facts (right).

**Approach:** Learn programs using neuro-symbolic induction. This overview shows one reasoning step.

**Results:** This approach recovers taxonomic relations (left) and outperforms previous state-of-the-art on a variety of problems (right; score is percentage of random initializations leading to solution. δILP is previous state-of-the-art (Evans, Grefenstette, 2018)).