Virginia Tech — Virginia Polytechnic Institute and State University | Spatial Data Management Lab | George Mason University | Sanghani Center for Artificial Intelligence & Data Analytics

Can LLM Find the Green Circle? Investigation and Human-guided Tool Manipulation for Compositional Generalization

Min Zhang[1], Jianfeng He[1], Shuo Lei[1], Murong Yue[2], Linhan Wang[1], Chang-Tien Lu[1]
[1]Virginia Tech, [2]George Mason University

Introduction

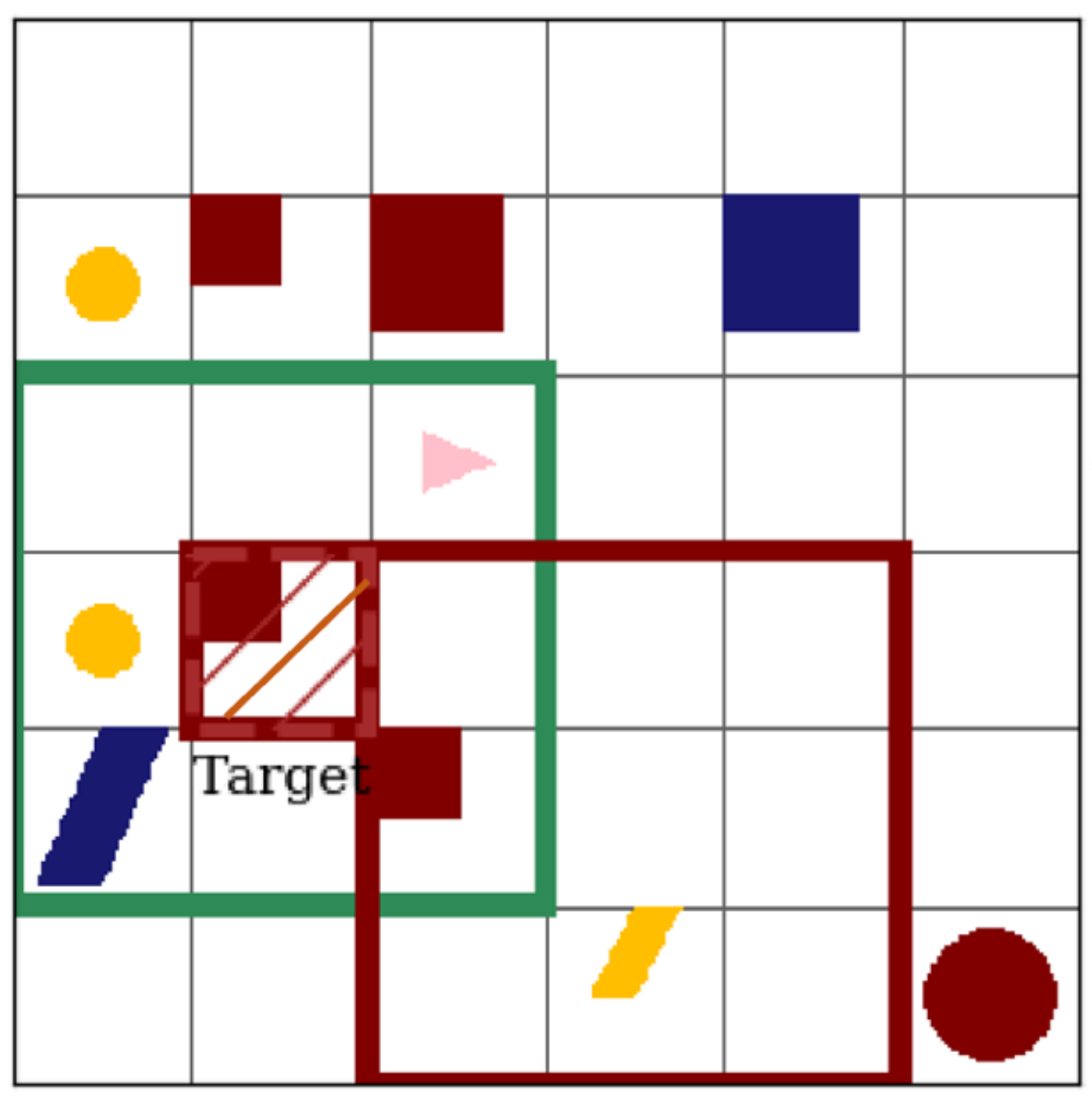
Background

- ➢ Natural languages are composed by individual components.
- ➢ Optimal models should generalize its understanding of components when presented with new combinations.
- ➢ LLMs show great generalization ability via in-context learning.

Research Questions

- ➢ Q1: Can prevailing ICL methods perform well on this task?
- ➢ Q2: How to improve LLM's ability of compositional generalization?
- ➢ Q3: Where does the ability come from?

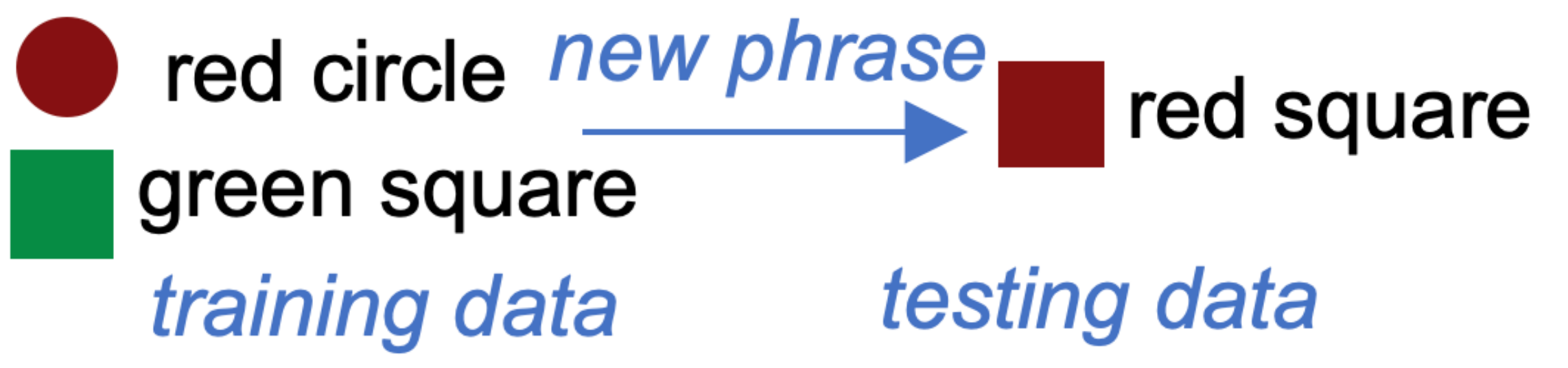
Task



Compositional

Find the small red square that is inside of a big box and in the same row as a yellow circle.

Generalization



● red circle — *new phrase* → 🟥 red square
🟩 green square
training data — *testing data*

Motivation

Chain-of-Thought (CoT):

step 1: find the yellow circle , …
Step 2: find the red square,
get obj3 …
Step 3: filter the position
get obj3, obj5
…

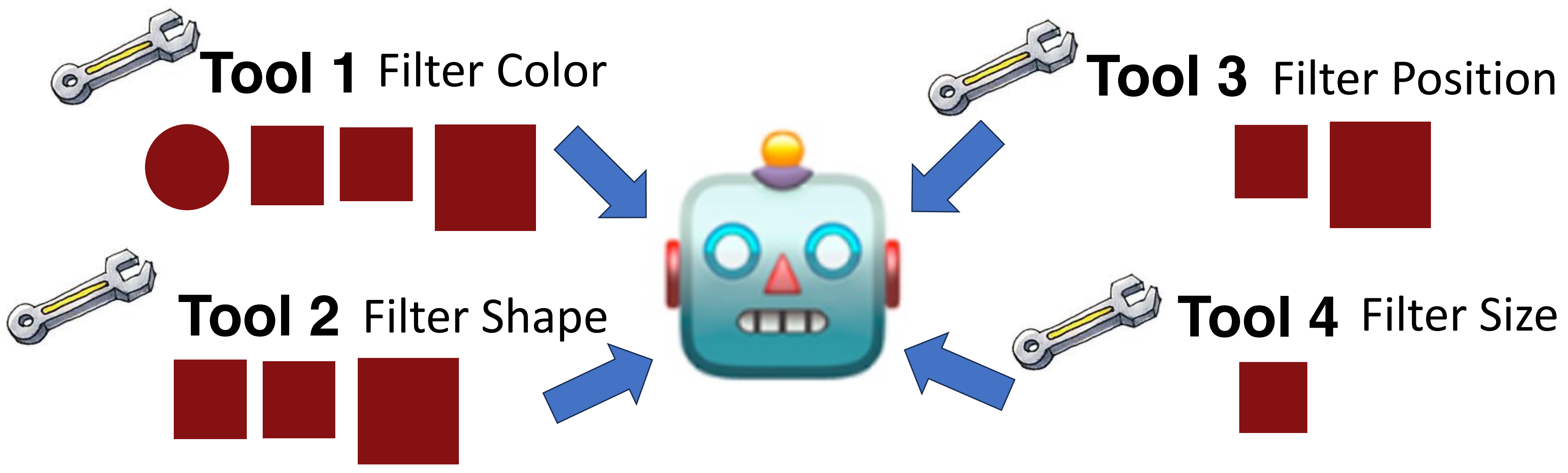
Program-of-Thought (PoT):

```
step1_size = 'small'
for obj in all_objs:
    if obj['size'] < 2:
        candidates.append(obj)
…
```

matching errors ***code logic errors***

cumulative errors

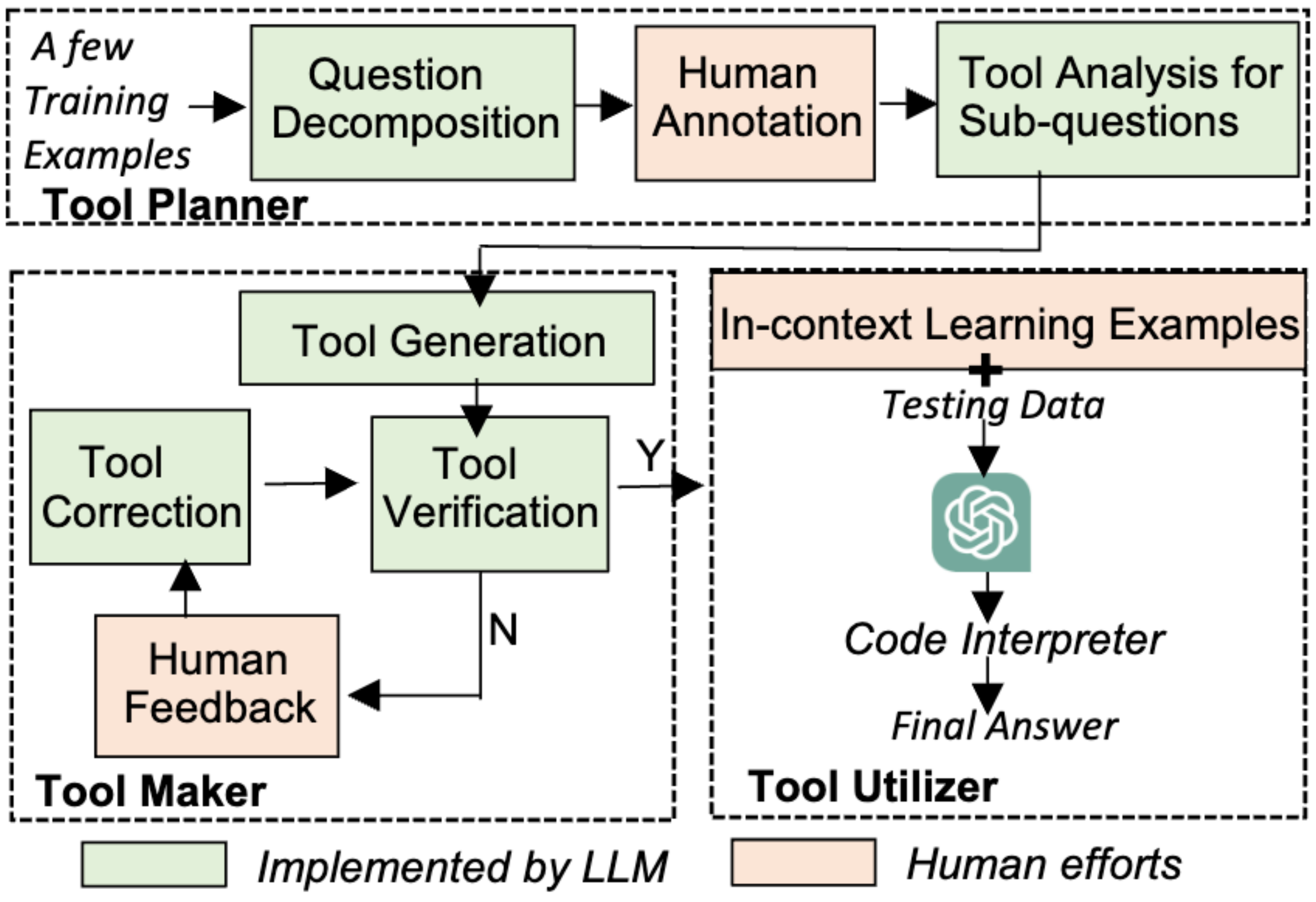
Our method



Tool 1 Filter Color
Tool 2 Filter Shape
Tool 3 Filter Position
Tool 4 Filter Size

Tool Generation and Usage

Implementation



- ▪ Decompose questions into sub-questions
- ▪ Make tools for sub-questions
- ▪ Combine tools to solve the whole question
- ▪ (Minimal) Human efforts on **a few** examples to correct LLMs

Input: *obj_0: (column=0, row=2, shape=box, color=green, size=3)*
obj_1: (column=1, row=3, shape=square, color=red, size=2) …
Output: *Answer: obj_1*

```
step1_color = 'yellow'
cand1 = filter_color(step1_color)
position = 'same row'
cand2 = filter_position(position)
condition_pairs = [(step1_obj, step1_relation)]
answer = combine_relation(condition_pairs)
```

Call Tools

Parameter Adaptation

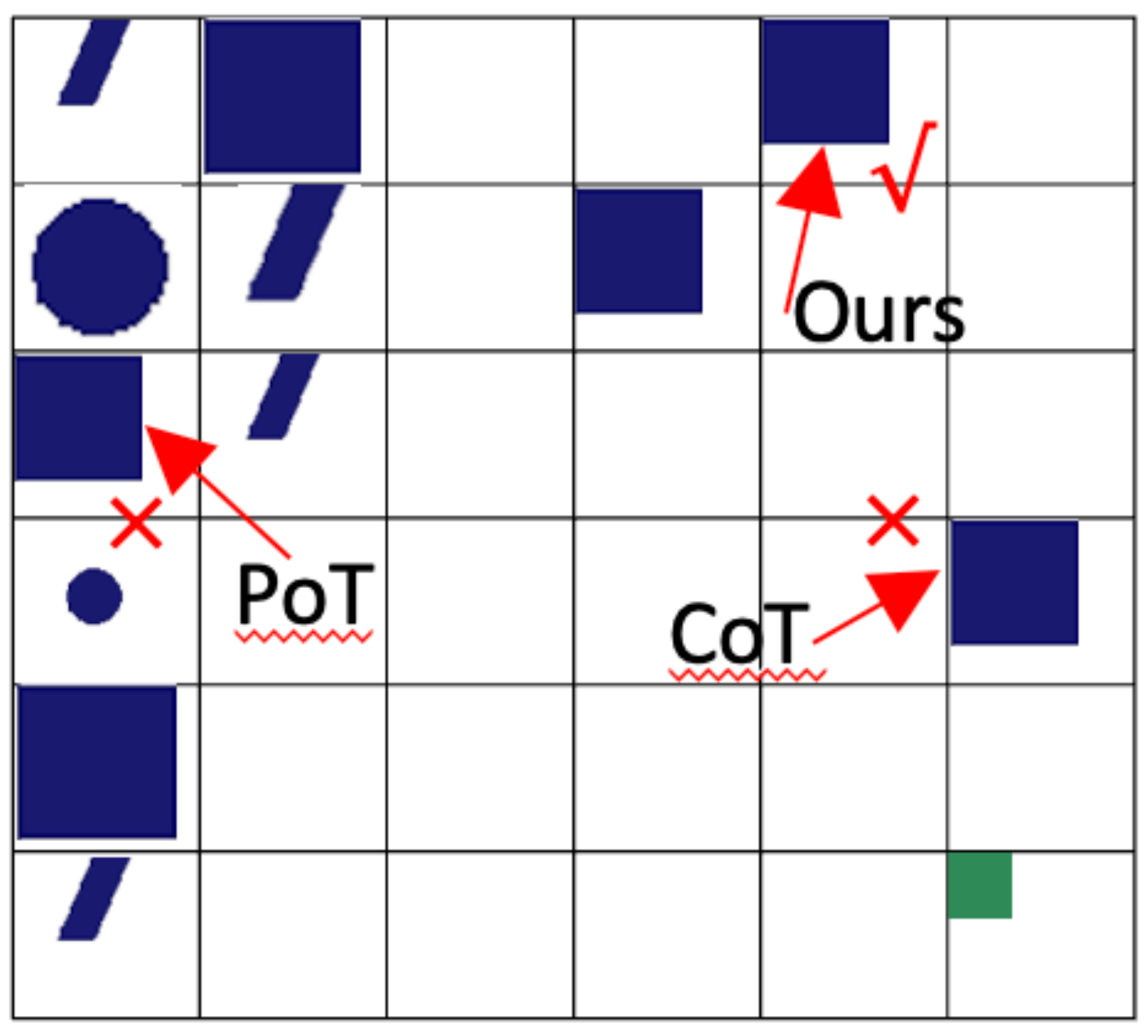
Result

Dataset	Test Split	Training or Finetuning				In-context Learning				
		MM-LSTM [14]	GCN-LSTM [20]	MM-TRF [15]	Gro-CoT [16]	Zero-Shot [10]	Stand. [11]	CoT [12]	PoT [13]	HTM-(Ours)
ReaSCAN	A1	50.4	92.3	96.7	99.6	31	34.8	40	96.2	98.6
	A2	14.7	42.1	58.9	93.1	24.2	25.2	30.2	93.8	97.6
	A3	50.9	87.5	93.3	98.9	28	28.6	40.2	96.2	98.6
	B1	52.2	69.7	79.8	93.9	20.2	24.6	25.4	77.8	99.4
	B2	39.4	52.8	59.3	86	19.8	33.8	31.8	65.4	100
	C1	49.7	57	75.9	76.3	13.6	19.6	16	79.6	95.4
	C2	25.7	22.1	25.5	27.3	20.2	20.6	15.6	4.8	95.8
	AVG	40.4	60.5	69.9	82.2	22.4	26.7	28.4	73.4	**97.9**
GSRR	S1	86.5	-	94.7	99.9	26.8	39	37.2	91	98.8
	S2	40.1	-	64.4	98.6	26.8	42	31.6	93.6	99.6
	S3	86.1	-	94.9	99.9	29	33.8	38	91.2	99
	S4	5.5	-	49.6	99.7	29	39.4	35.8	89.2	98.8
	S5	81.4	-	59.3	99.5	24	34.4	42.8	61.8	99
	S6	81.8	-	49.5	96.5	26.6	31.2	28.8	84.7	99.2
	AVG	58.9	-	63.5	98.8	27	36.6	35.7	85.3	**99.1**

The more challenging the test splits, the greater the improvement!
e.g. Accuracy 27.3%→95.8% on C2.

Case Study

Find the small blue square that is in the same row as a blue cylinder that is in the same column as a blue circle



Replace language with random four letters.

Semantic Representation
↓
Symbolic Representation

-	semantic			symbolic		
-	P1	P2	P3	P1	P2	P3
Zero-Shot	78.6	28.2	20.0	67.6	17.2	14.0
Stand.	78	33.6	22.0	68.8	28.6	20.4
CoT	95.8	43.8	19.0	97.6	37.4	21.0
PoT	100	98.4	97.8	94.4	88.4	81.2
HTM (Ours)	100	99.6	98.6	100	99.8	98.2

The ability arises from pattern combinations rather than relying solely on semantics learned from pretraining.