Symbolic metaprogram search improves learning effciency and explains rule learning in humans

Corresponding Author: Joshua S. Rule

Presented by: Hao Su

Joshua S. Rule

- Postdoctoral scholar with Steven Piantadosi and Alison Gopnik at UC Berkeley.
- Completed PhD with Josh Tenenbaum at MIT.
- Uses behavioral experiments and computational methods to study how people develop conceptual systems and leverage them to accomplish their goals
- Work revolves around an idea called the child as hacker.

Rule Learning

Motivation

Throughout their lives, humans seem to learn a variety of rules for things like applying category labels, following procedures, and explaining causal relationships.

We quesion: how people acquire **algorithmically rich rules**?

The goal is to propose a model that resembles humans' ability in rule learning, thus providing a valid hypothesis that explains people's conceptual systems.

Observation/Requirements

- representations should be interpretable
- learning should also be possible from sparse data
- learning should require only moderate amounts of computation and search

Task Statement

What is the target of our rule-learning?

The task is to find a funtion that maps a given input to a given output. An example is to find an f that satisfies

$$[1,3,9,7] \xrightarrow{f} [1,1,3,3,9,9,7,7]$$

$$[9,2] \xrightarrow{f} [9,9,2,2]$$

Spoiler allert: $f = (\lambda\ \mathrm{xs}(\mathrm{if}\ (\mathrm{empty}\ \mathrm{xs})\mathrm{xs}\ [(\mathrm{head}\ \mathrm{xs}), (\mathrm{head}\ \mathrm{xs})|(f(\mathrm{tail}\ \mathrm{xs}))]))$ clearly is the f we've been looking for.

Why is this so difficult?

The f above is composed using *primitives*, and rule learning, or *program induction*, or *program synthesis*, is faced with the hard program of searching the program space, which grows **exponentially** according to the program length and number of primitives.

Why I chose this paper: The parts that deserve out attention

~~Dr. Dai asked me to.~~

Methods and Paradigms proposed in this paper

Model: **MPL**(MetaProgram Learner)

- A novel knowledge representation
- A search scheme to consider: custom MCMC(Markov Chain Monte Carlo)

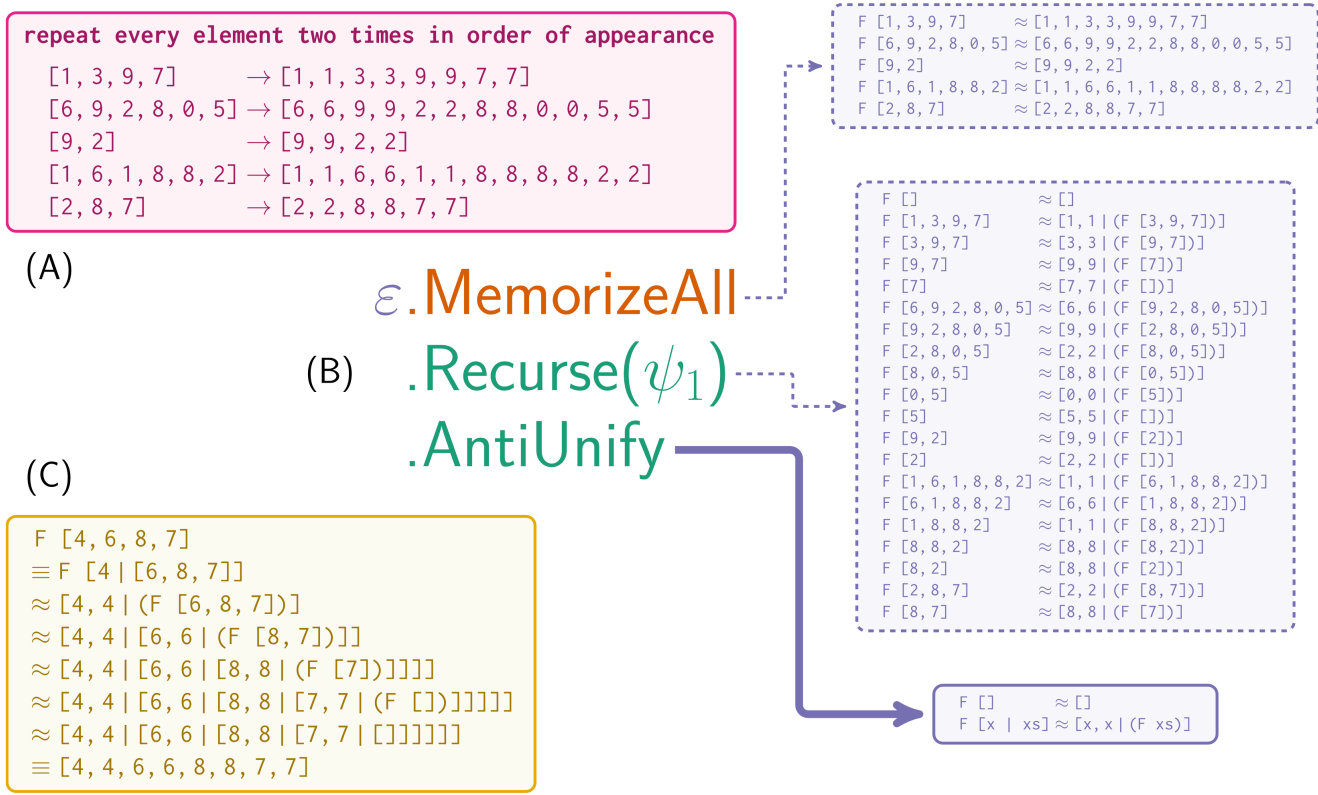
Don’t panic. We’ll walk through them.

Personal Take in Reading this paper

Ignore the human behavioral analysis part.

MPL: a proper introduction

Instead of using only *object-level primitives*(e.g., `head xs`), MPL defines and employs *metaprimitives*.



More on Metaprimitives

Some Other Examples

MemorizeAll(t)

```
# where datum 1: C [1, 38, 19, 4] = [19]
# where datum 2: C [31, 41, 59, 62, 5] = [59]
# where datum 3: C [68, 47, 3, 6, 0, 9, 77] = [3]
C [31, 41, 59, 62, 5] = [59]
C [x, y, 19, z] = [19]
```

```
C [1, 38, 19, 4] = [19]
C [68, 47, 3, 6, 0, 9, 77] = [3]
C [31, 41, 59, 62, 5] = [59]
C [x, y, 19, z] = [19]
```

DeleteRule(t, ψ)

```
C [1, 38, 19, 4] = [19]
C [68, 47, 3, 6, 0, 9, 77] = [3]
C [31, 41, 59, 62, 5] = [59]
C [x, y, 19, z] = [19]
```

```
C [1, 38, 19, 4] = [19]
C [68, 47, 3, 6, 0, 9, 77] = [3]
C [x, y, 19, z] = [19]
```

Variablize(t, ψ)

```
C [w, x, y, z] = [y]
C [91, 12, 63, 42, 35] = [63]
```

```
C [w, x, y, z] = [y]
C [91, 12, x, 42, 35] = [x]
```

AntiUnify(t)

```
C [25] = []
C [0] = []
C [9, 25] = [9 | (C [25])]
C [81, 9, 25] = [81 | (C [9, 25])]
C [50, 0] = [50 | (C [0])]
C [28, 50, 0] = [28 | (C [50, 0])]
C [37, 28, 50, 0] = [37 | (C [28, 50, 0])]
```

```
C [x] = []
C [x | y] = [x | (C y)]
```

Comparison with MetaRules

```
metarule([P,Q],[P,A,B],[[Q,A,B]]).
metarule([P,Q,R],[P,A,B],[[Q,A,B],[R,A,B]]).
metarule([P,Q,R],[P,A,B],[[Q,A,C],[R,C,B]]).
```

Discussion on Metaprimitives

Advantages

- Combines object-level content and structured transformation
- Can leverage the internal structure of data.
- Is interpretable and resembles the way human access concepts to a certain degree.

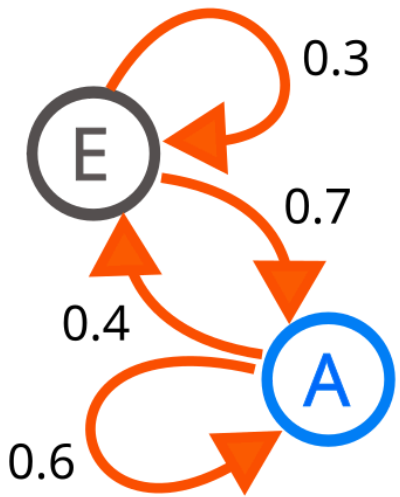
Cons

- Is domain specific. Can be hard to generalize to other problems.
- The current implementation of metaprimitives can be sub-optimal: more of an intuition or a diretion than a already powerful instrument.

Search Scheme

The paper is super ablative on this part so I'm not crystal clear how the search is done either, yet we can surely explore the basic tatics.

A quick recap on Markov Chain and Monte Carlo



Markov Chain Monte Carlo Search

with parallel tempering

MAP(Maxium A Posteriori) inference in Bayesian posterior

$$p(H|D) \propto p(D|H)p(H)$$

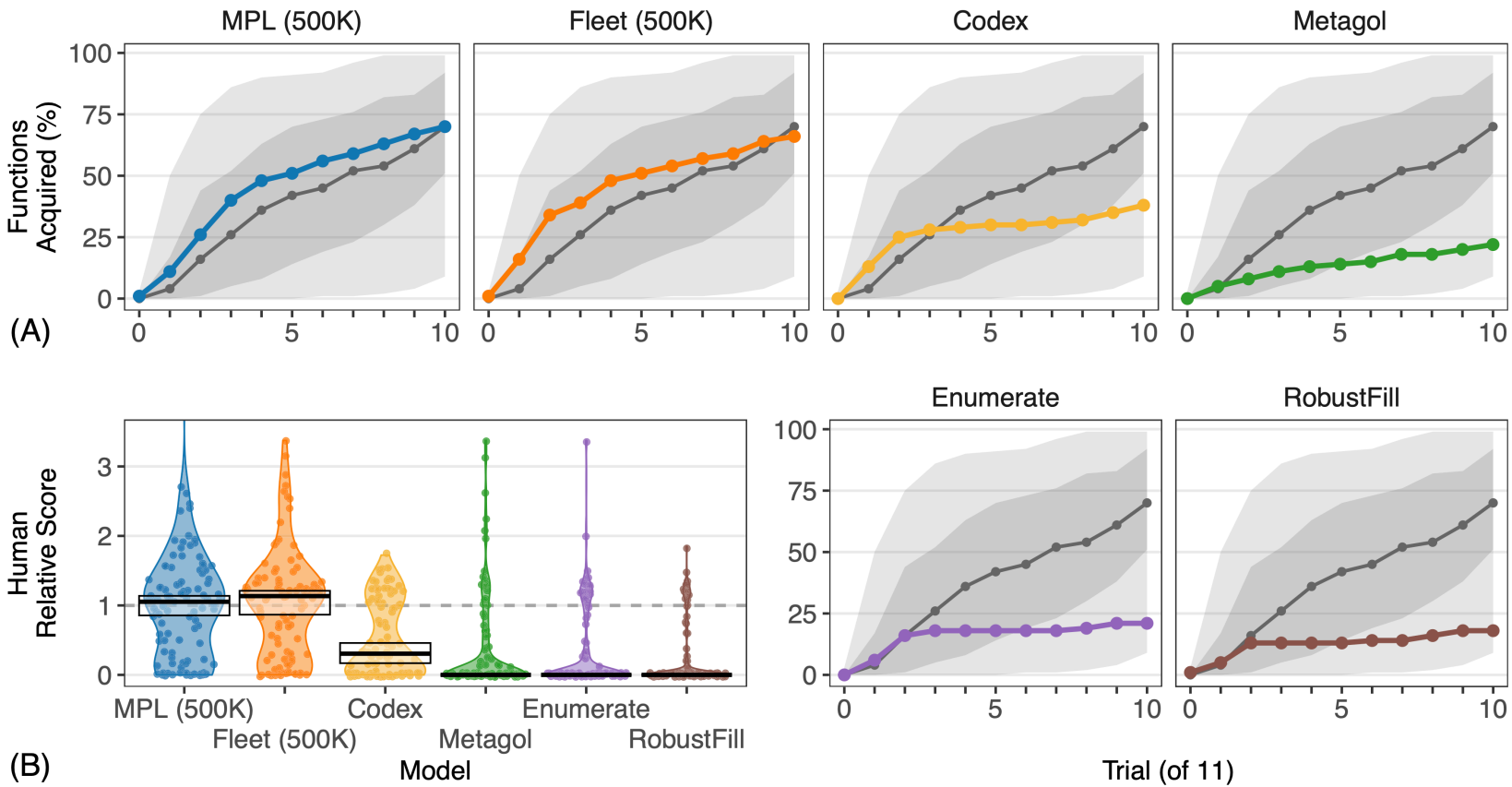
where H is the prior distribution of metaprograms and D is the data given, and $p(D|H)$ is a prefix-based likelihood.

Parallel Tempering

$$p(H|D) \propto p(D|H)^{\frac{1}{T}}p(H)$$

Tree Proposal

Results


MPL (500K)
Fleet (500K)
Codex
Metagol
Functions Acquired (%)
100
75
50
25
0
0
5
10
(A)
Enumerate
RobustFill
Human Relative Score
3
2
1
0
MPL (500K)
Fleet (500K)
Codex
Metagol
Enumerate
RobustFill
Model
(B)
Trial (of 11)

Thanks For Your Attention

Questions are welcome. References will be updated on future versions of the slides.

Powered by Slidev