



Massachusetts  
Institute of  
Technology



PENCIL  
code

# Modeling Student Learning with 3.8 Million Program Traces

November 14, 2025

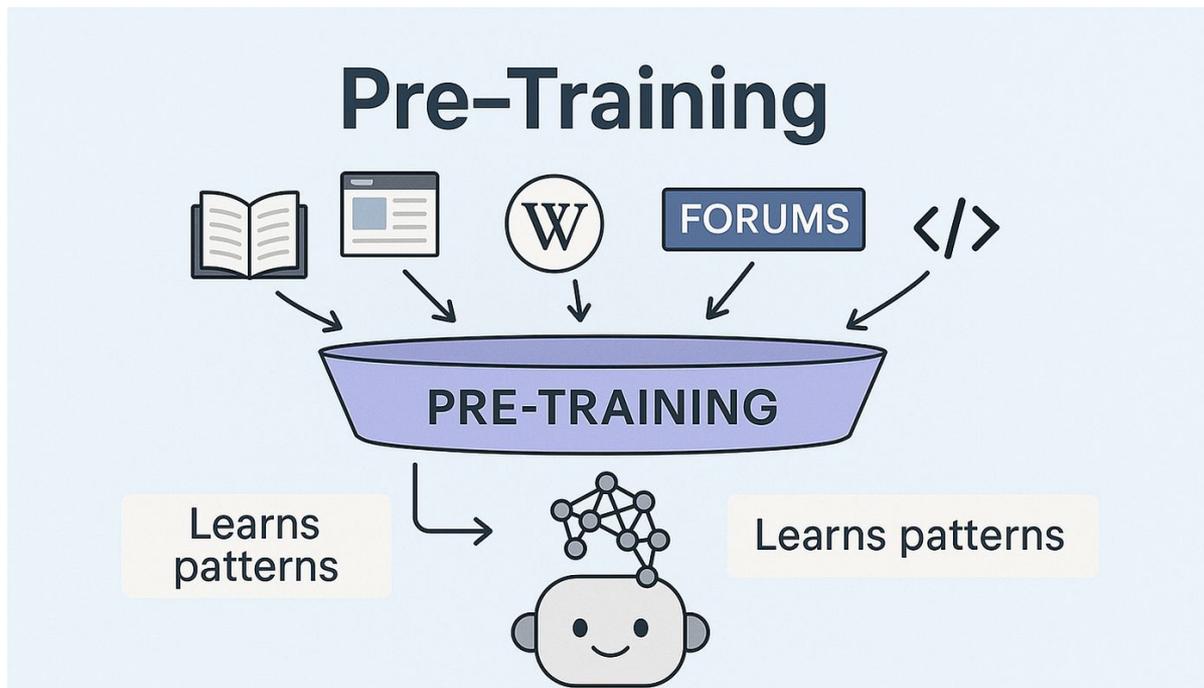
Alexis Ross\*, **Megha Srivastava\***, Jeremiah Blanchard, Jacob Andreas

# About Me

- PhD student at Stanford CS Department
- Interested in **AI+Education**
  - How can AI *help* improve access to high-quality education (e.g. intelligent tutoring systems)?
  - How does AI *hurt* humans' cognitive skills (e.g. overreliance, skill atrophy)?
- Contact: [megha@cs.stanford.edu](mailto:megha@cs.stanford.edu)



# Motivation: Pre-training relies on internet-scale data



# Motivation: Most data was created via some “edit process”

## Jane Austen: Revision history

Help

Article Talk

Read View source View history Tools

View logs for this page (view filter log)

### Filter revisions

External tools: Find addition/removal (Alternate) · Find edits by user (Alternate) · Page statistics · Pageviews · Fix dead links · Copyvio Detector

For any version listed below, click on its date to view it. For more help, see Help:Page history and Help:Edit summary. (cur) = difference from current version, (prev) = difference from preceding version, m = minor edit, → = section edit, ← = automatic edit summary

(newest | oldest) View (newer 50 | older 50) (20 | 50 | 100 | 250 | 500)

### Compare selected revisions

- [\(cur | prev\)](#)  [07:24, 31 October 2025](#) [TracyHyrax](#) (talk | contribs) .. (103,626 bytes) (−27) .. (*Update link*)
- [\(cur | prev\)](#)  [13:55, 27 October 2025](#) [Schazjmd](#) (talk | contribs) .. (103,653 bytes) (−22) .. (*→Early years to age 20: repl with template*)
- [\(cur | prev\)](#)  [08:46, 27 October 2025](#) [Masato.harada](#) (talk | contribs) m .. (103,675 bytes) (−62) .. (*→Early years to age 20: c/e*)
- [\(cur | prev\)](#)  [03:32, 27 October 2025](#) [Woodridge37](#) (talk | contribs) .. (103,737 bytes) (+130) .. (*→Early years to age 20*)
- [\(cur | prev\)](#)  [07:51, 20 October 2025](#) [Masato.harada](#) (talk | contribs) .. (103,607 bytes) (+35) .. (*Link, italics.*)
- [\(cur | prev\)](#)  [00:41, 20 October 2025](#) [CareCareSC](#) (talk | contribs) m .. (103,572 bytes) (+33) .. (*→top: added movie*) (*Tags: Mobile edit, Mobile app edit, Android app edit, App section source*)
- [\(cur | prev\)](#)  [09:22, 14 October 2025](#) [Jevansen](#) (talk | contribs) .. (103,539 bytes) (−42) .. (*Removing from Category:19th-century English writers has diffusing category using Cat-a-lot*)
- [\(cur | prev\)](#)  [15:27, 2 October 2025](#) [Humanities Scholar](#) (talk | contribs) m .. (103,581 bytes) (+28) .. (*Fixed URL*)
- [\(cur | prev\)](#)  [15:12, 2 October 2025](#) [Humanities Scholar](#) (talk | contribs) m .. (103,553 bytes) (+456) .. (*Added sentence about her moderate rather than modest fame and about critics on her mild reputation*)
- [\(cur | prev\)](#)  [14:53, 2 October 2025](#) [Humanities Scholar](#) (talk | contribs) m .. (103,097 bytes) (+122) .. (*Changed line about her modest fame to moderate fame. Added detail about public vs. private fame.*)
- [\(cur | prev\)](#)  [09:27, 2 October 2025](#) [No Swan So Fine](#) (talk | contribs) .. (102,975 bytes) (0) .. (*→Honours: clarify*)
- [\(cur | prev\)](#)  [09:26, 2 October 2025](#) [No Swan So Fine](#) (talk | contribs) .. (102,975 bytes) (+412) .. (*→Honours: + Statue of Jane Austen*)

### Appearance

Text

- Small
- Standard
- Large

This page always uses small font size

Width

- Standard
- Wide

Color (beta)

- Automatic
- Light
- Dark

# Motivation: Most data was created via some “edit process”

The screenshot shows the Wikipedia revision history for the article "Jane Austen". At the top, there is a search bar and navigation links. The main content area is titled "Jane Austen: Revision history" and includes a "Filter revisions" section. Below this, there are external tools and a list of revisions. The revisions list includes columns for "cur | prev", date and time, user name, and byte changes. A large orange box is overlaid on the right side of the page, containing the text "What can we learn from edit traces?".

WIKIPEDIA The Free Encyclopedia

Search Wikipedia

Donate Create account Log in

## Jane Austen: Revision history

Article Talk

View logs for this page (view filter log)

Filter revisions

External tools: Find addition/removal (Alternate) · Find edits by user (Alternate) · Page statistics · Pageviews · Fix dead links · Copyvio Detector

For any version listed below, click on its date to view it. For more help, see Help:Page history and Help:Edit summary. (cur) = difference from current version. (prev) = difference from preceding version, m = minor edit, → = section edit, ← = automatic edit summary

(newest | oldest) View (newer 50 | older 50) (20 | 50 | 100 | 250 | 500)

Compare selected revisions

- (cur | prev) 07:24, 31 October 2025 TracyHyrax (talk | contribs) .. (103,626 bytes) (-27) .. (Update link)
- (cur | prev) 13:55, 27 October 2025 Schazjmd (talk | contribs) .. (103,653 bytes) (-22) .. (→Early years to age 20: repl with template)
- (cur | prev) 08:46, 27 October 2025 Masato.harada (talk | contribs) m .. (103,675 bytes) (-62) .. (→Early years to age 20: c/e)
- (cur | prev) 03:32, 27 October 2025 Woodridge37 (talk | contribs) .. (103,737 bytes) (+130) .. (→Early years to age 20)
- (cur | prev) 07:51, 20 October 2025 Masato.harada (talk | contribs) .. (103,607 bytes) (+35) .. (Link, italics.)
- (cur | prev) 00:41, 20 October 2025 CareCareSC (talk | contribs) m .. (103,572 bytes) (+33) .. (→top: added movie) (Tags: Mobile edit, Mobile app edit, Android app edit, App section source)
- (cur | prev) 09:22, 14 October 2025 Jevansen (talk | contribs) .. (103,539 bytes) (-42) .. (Removing from Category:19th-century English writers has diffusing category using Cat-a-lot)
- (cur | prev) 15:27, 2 October 2025 Humanities Scholar (talk | contribs) m .. (103,581 bytes) (+28) .. (Fixed URL)
- (cur | prev) 15:12, 2 October 2025 Humanities Scholar (talk | contribs) m .. (103,553 bytes) (+456) .. (Added sentence about her moderate rather than modest fame and about critics on her mild reputation)
- (cur | prev) 14:53, 2 October 2025 Humanities Scholar (talk | contribs) m .. (103,097 bytes) (+122) .. (Changed line about her modest fame to moderate fame. Added detail about public vs. private fame.)
- (cur | prev) 09:27, 2 October 2025 No Swan So Fine (talk | contribs) .. (102,975 bytes) (0) .. (→Honours: clarify)
- (cur | prev) 09:26, 2 October 2025 No Swan So Fine (talk | contribs) .. (102,975 bytes) (+412) .. (→Honours: + Statue of Jane Austen)

Appearance hide

Text

Small

Standard

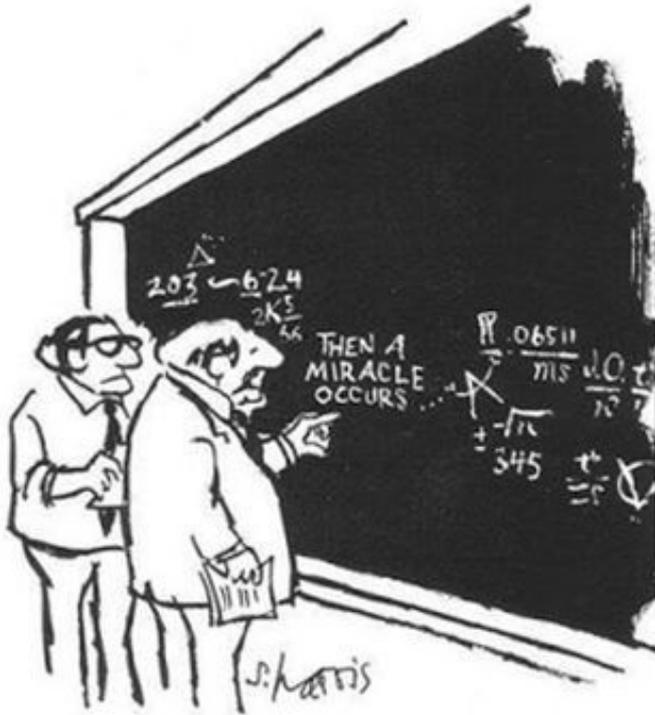
Large

Light

Dark

### What can we learn from edit traces?

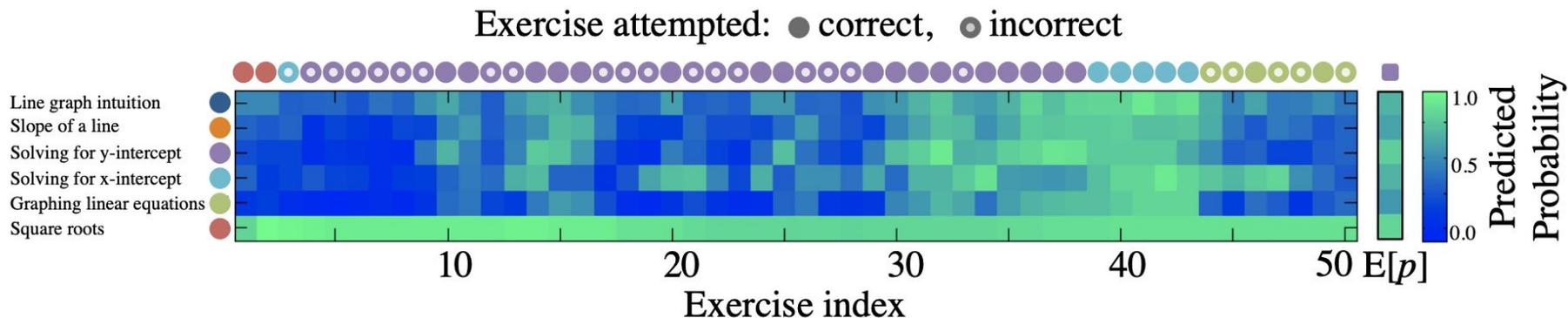
# Importance of showing “edits” in education



"I THINK YOU SHOULD BE MORE EXPLICIT HERE IN STEP TWO."

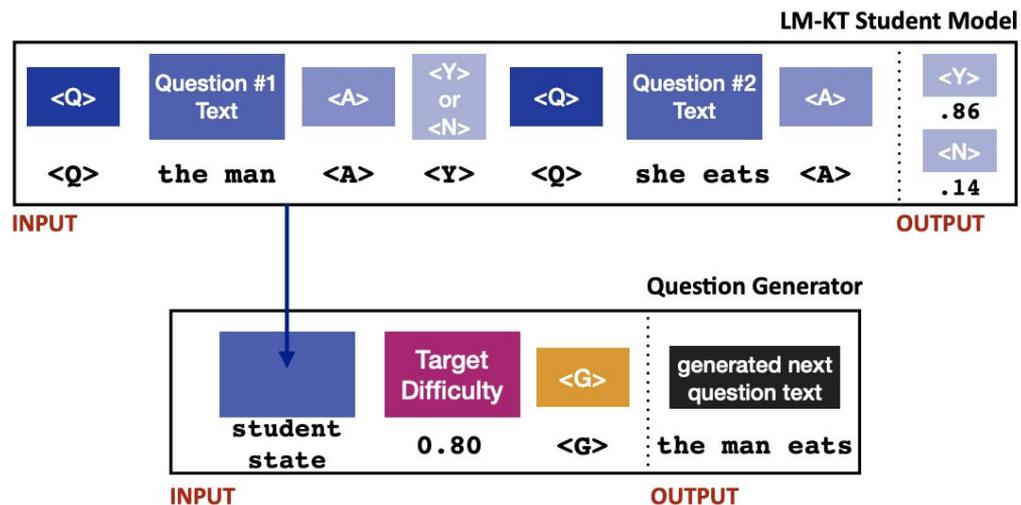
- Showing work encourages student to understand a concept beyond “memorizing”
- Helps teacher learn what concepts a student understands
- Helps student get more granular feedback

# Prior Work: Deep Knowledge Tracing



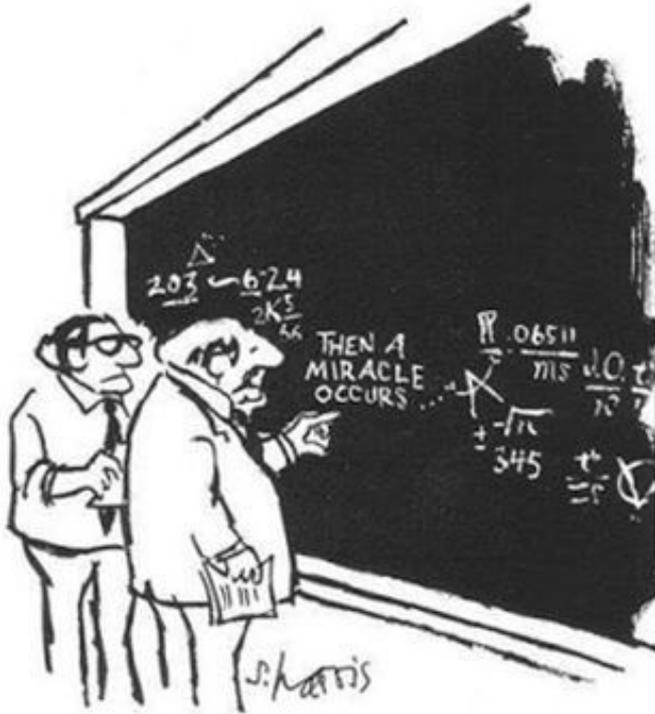
- First deep learning approach to student modeling
- LSTM over student answer history
- Only **final student answer (correct/incorrect)** and **simple discrete featurization**

# Prior Work: LM-Knowledge Tracing



- First language-model based approach to student modeling
- Student answer history placed in-context
- **Text featurization** helps capture complex learning concepts (e.g. grammar, person)
- Only **final student answer**

# Importance of showing “edits” in education



"I THINK YOU SHOULD BE MORE EXPLICIT HERE IN STEP TWO."

- Encourages student to understand a concept beyond “memorizing”
- Helps teacher learn what concepts a student understands
- Helps student get more granular feedback

A good student model should also capture a student’s **process** during problem-solving

# Model “reasoning” has strange behavior

r/ChatGPT • 9 mo. ago  
njuff22

## O1 mini randomly thought in Chinese in the middle of solving a problem

Funny

ChatGPT o1-mini

OK, let me see. The issue seems to be that `update_player_buttons` is called before defining the zoom buttons. Adding the zoom buttons first in `setup_sidebar` could resolve this.

**Prioritizing updates**  
I'm thinking about whether defining zoom buttons before `update_player_buttons` or ensuring the function acknowledges their existence is the solution. Adjust `setup_sidebar` or adjust button lists.

**Amending the issue**  
It's becoming clear that the solution involves outlining the actions in `setup_sidebar` to incorporate the zoom buttons before `update_player_buttons` is triggered.

搞明白问题所在

To resolve the issue, the assistant should clarify that `self.zoom_in_button` and `self.zoom_out_button` must be defined before `update_player_buttons` is called. This ensures the code functions correctly.

**Revisiting method order**  
OK, let me see. I'm thinking through the sequence of defining zoom buttons and updating player buttons, ensuring definitions precede their use to avoid conflicts.

**Arranging the layout**  
I'm organizing the sidebar setup by placing zoom buttons before updating player buttons. This ensures the buttons are correctly defined upfront.

Message ChatGPT

# Reasoning in code generation models

Find the conjugate of a complex number  $z$  in Python.

- Reasoning traces → significant increase in performance
- **Often synthetically generated**
- Always filtering for traces that lead to correct answer

*Do they reflect how real users reason and solve programming problems?*

Okay, I need to find the conjugate of a complex number  $z$  in Python. Hmm, let me think about how complex numbers work. I remember that a complex number has a real part and an imaginary part. The conjugate is when you change the sign of the imaginary part. So if  $z$  is  $a + bj$ , the conjugate is  $a - bj$ .

In Python, complex numbers are represented with a real and imaginary part. I think there's a method or function that can help with this. Oh right, I remember that the 'conjugate()' method is available for complex numbers. So if I have a complex number  $z$ , I can call  $z.conjugate()$  to get its conjugate.

Wait, should I create an example to test this? Let's say  $z$  is  $3 + 4j$ . Then the conjugate should be  $3 - 4j$ . If I run  $(3+4j).conjugate()$ , does it return  $(3-4j)$ ? I think it does. Alternatively, I can also use the built-in 'complex' function with the real and imaginary parts, but the method seems more straightforward.

Another approach is using the 'cmath' module, but I think the method is sufficient here. So the steps are: define the complex number, then call the conjugate method on it. That should give me the desired result.  
</think>

To find the conjugate of a complex number ' $z$ ' in Python, you can use the 'conjugate()' method directly on the complex number.

**Example:**

```
python
z = 3 + 4j
conjugate_z = z.conjugate()
print(conjugate_z) # Output: (3-4j)
```

This method efficiently returns the complex conjugate by changing the sign of the imaginary part.

# Why do we want to model student programming behavior?

1. Predict whether a student will solve a problem correctly
  - a. Useful for assessments / standardized testing
2. Predict how long a student will take to solve certain concepts
  - a. Useful for feedback / intelligent tutoring system design
3. Help a student recover from a bug / incorrect program state
4. Capture personalization style (comments, colors)
5. Simulate a “population” of students for curricula design

# Outline

1. Overview of Pencil Code dataset
2. Evaluation Metrics
3. Experimental Results
  - a. Behavioral
  - b. Representational: Code Embeddings
  - c. Representational: Student Embeddings
  - d. Adapting to new students
  - e. Error recovery + steering
4. Discussion

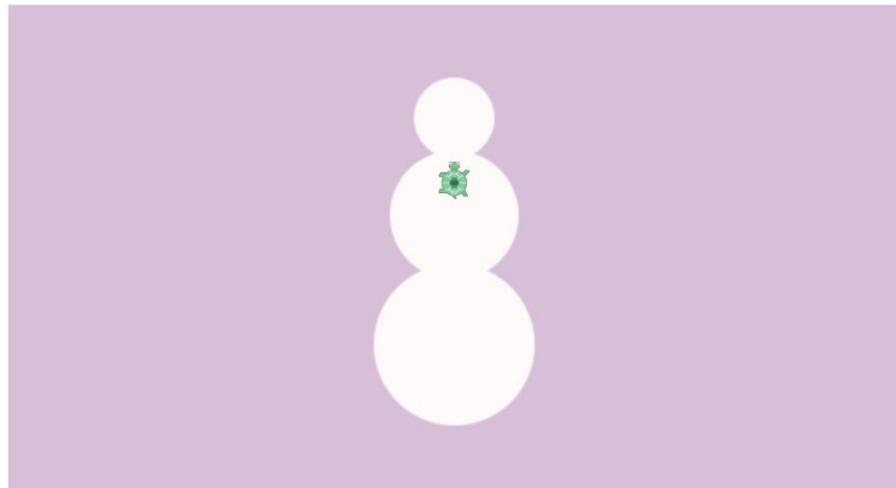
# Pencil Code

- Used across a variety of school districts for intro programming assignments
  - Typically K-12 students
- Students can use both block-based (similar to Scratch) and script-based programming (CoffeeScript)
- Many educational resources with sample programming assignments

# Dream it. Code it.

Learn professional programming languages using an editor that lets you work in either blocks or text. Create art, music, games, and stories. Or invent a program that will change the world.

Let's play!



Draw

Create art



Jam

Make music



Imagine

Code an adventure

## Get Creative

Get some ideas for [art](#), [music](#), or [games](#). Or to code anything, [start on a blank page](#).

**Blocks** ▼

- Control
- Art
- Text
- Sound
- Operators
- Sprites
- Snippets

fd 100

rt 90

lt 90

bk 100

rt 180, 100

lt 180, 100

speed 10

speed Infinity

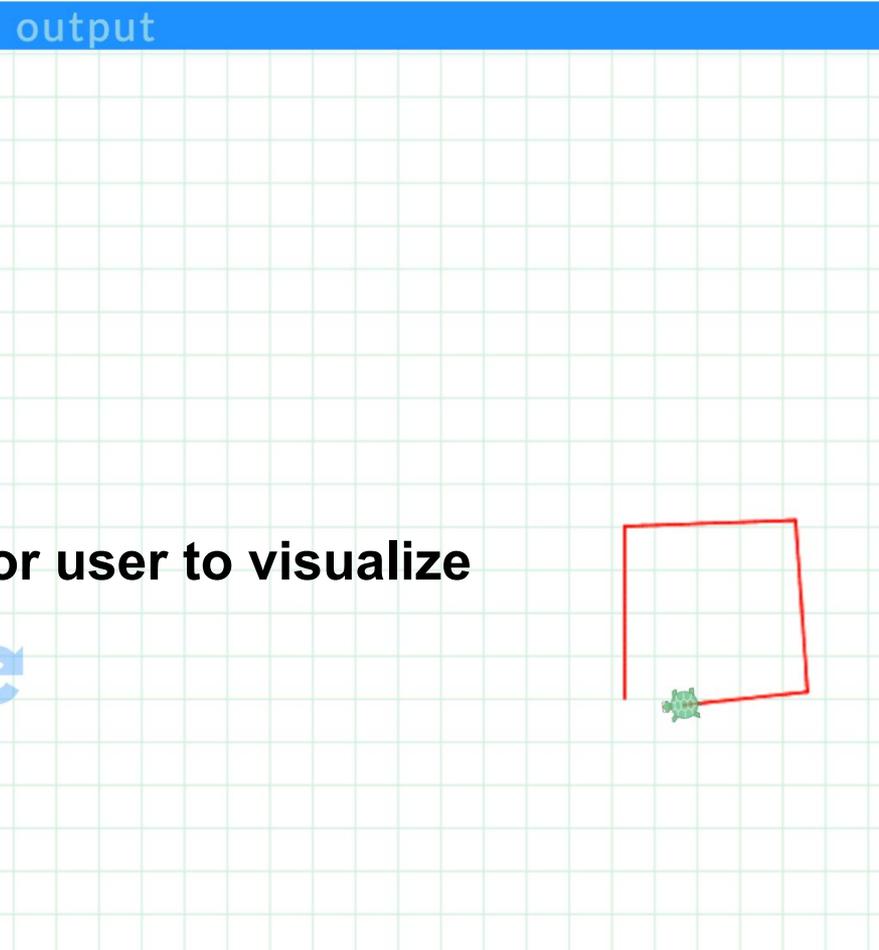
home()

turnto 270

moveto 100, 50

movexy 30, 20

```
1 speed 2
2 pen ▼ red
3 for [1..25]
4   fd ▼ 100
5   rt ▼ 88
6
```



**“Execute” button for user to visualize**



# Program Trace: snowman

- Sequence of programs at each “execute” step for an assignment

CODE 1 (2018-12-13 16:48:42):

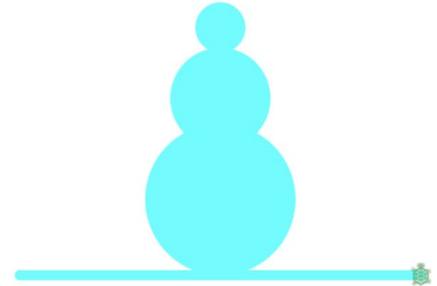
```
Speed 40  
bk 50  
dot aqua, 150  
fd 100  
dot aqua, 100  
fd 70  
dot aqua, 50
```

CODE 2 (2018-12-13 16:48:50):

```
speed 40  
bk 50  
dot aqua, 150  
fd 100  
dot aqua, 100  
fd 70  
dot aqua, 50
```

CODE 3 (2018-12-13 16:50:03):

```
speed 40  
bk 50  
dot aqua, 150  
fd 100  
dot aqua, 100  
fd 70  
dot aqua, 50  
jumpto -200, -125
```



# Program Trace: snowman

- Sequence of programs at each “execute” step for an assignment

```
CODE 1 (2018-12-10 16:11:38):  
#<NAME>, 3rd, Snowman  
dot white, 50  
fd 100
```

```
CODE 2 (2018-12-10 16:12:17):  
#<NAME>, 3rd, Snowman  
dot white, 100
```

```
CODE 3 (2018-12-10 16:13:46):  
#<NAME>, 3rd, Snowman  
for [1..3]  
dot snow, 100
```



# Program Trace: snowman

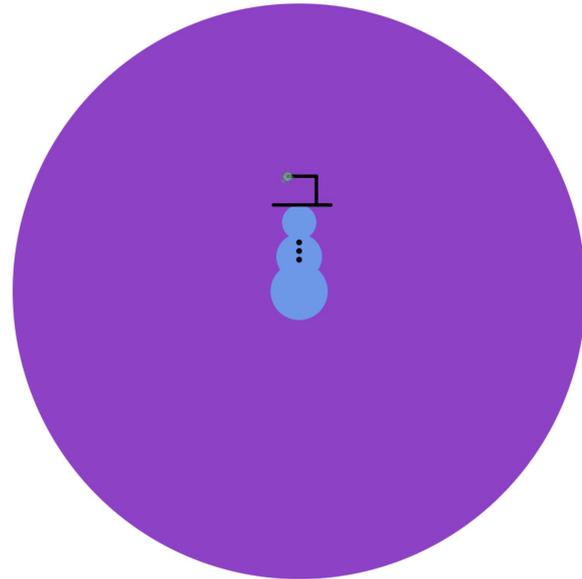
- Sequence of programs at each “execute” step for an assignment

CODE 1 (2018-11-14 12:18:03):

```
#<NAME>, snowman
speed 20
pen darkorchid
dot 1000
pen cornflowerblue
dot 100
fd 60
dot 80
fd 60
dot 60
pen null
#buttons
bk 35
dot 10
...
```

CODE 2 (2018-11-14 12:18:10):

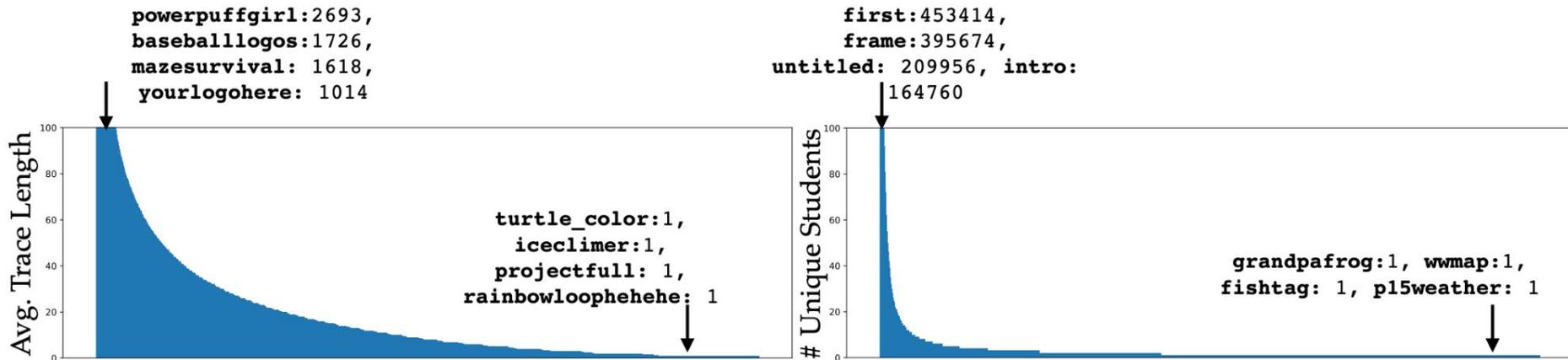
```
#<NAME>, snowman
speed 1
pen darkorchid
dot 1000
pen cornflowerblue
dot 100
fd 60
dot 80
fd 60
dot 60
pen null
#buttons
bk 35
dot 10
...
```



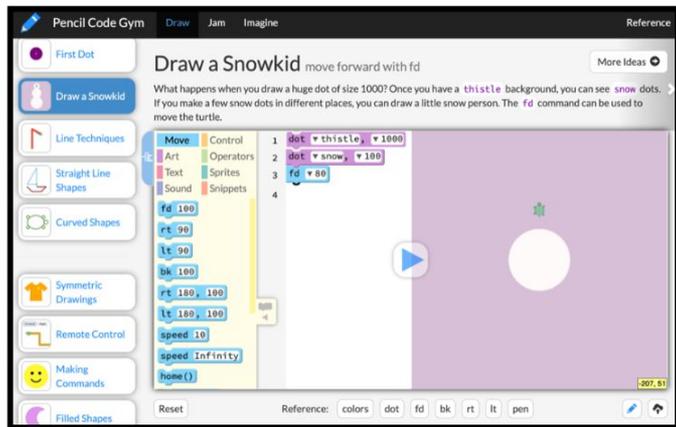
# Pencil Code Dataset Overview

- 3.8 Million Program Traces
- > 1 Million Students
- > 200k unique assignments
- 2016-2025 range
- Heavy long-tail behavior

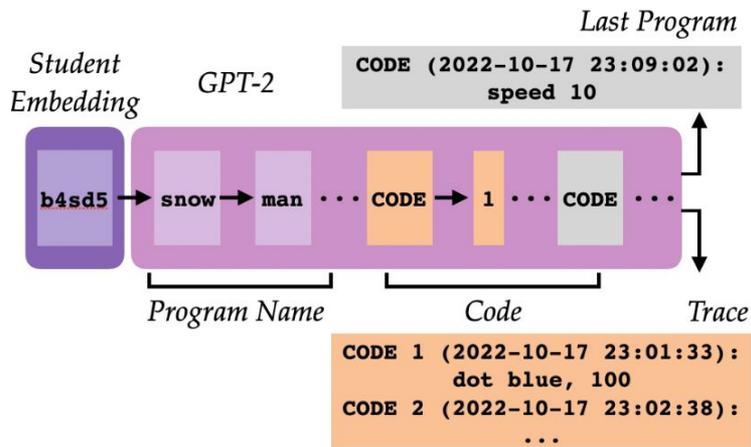
For the **same assignment**, students can show strong **individual variation**.



## A. Pencil Code Interface

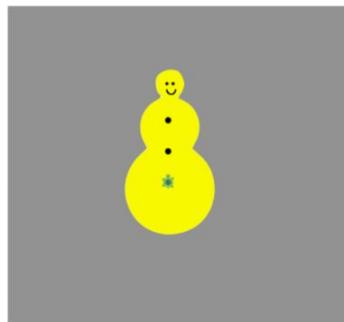


## B. Model Architecture



## C. Final Program State

```
#[ANONYMIZED], 1, Snowman
speed 10
dot yellow, 150
  fd 100
dot yellow, 100
  fd 70
dot yellow, 50
  rt 90
  fd 5
dot black, 5
  rt 180
  fd 10
dot black, 5
  lt 90
  fd 10
pen black, 2
  lt 180, 7
  pu()
moveto -3, 111
dot black, 10
  bk 50
dot black, 10
  bk 50
dot black, 10
```



## D. Synthetic Trace

```
#[ANONYMIZED], 1, Snowman
speed 10

#[ANONYMIZED], 1, Snowman
speed 10
dot yellow, 150

#[ANONYMIZED], 1, Snowman
speed 10
dot yellow, 150
fd 100

#[ANONYMIZED], 1, Snowman
speed 10
dot yellow, 150
fd 100
```

## E. User Trace

```
#[ANONYMIZED], 1, Snowman
dot blue, 100

#[ANONYMIZED], 1, Snowman
dot blue, 100
fd 100
dot blue, 50

#[ANONYMIZED], 1, Snowman
dot blue, 200
fd 100
dot blue, 100

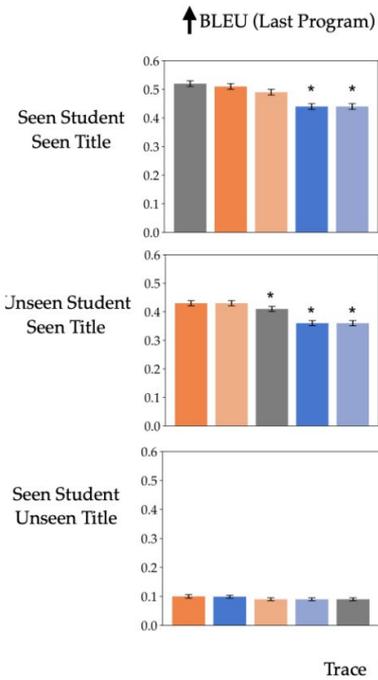
#[ANONYMIZED], 1, Snowman
dot blue, 150
fd 100
dot blue, 100
```

## F. Trace Edit Types

small add	100%	43%
large add		7%
small del		
large del		
color		7%
number		43%
comment		
function		

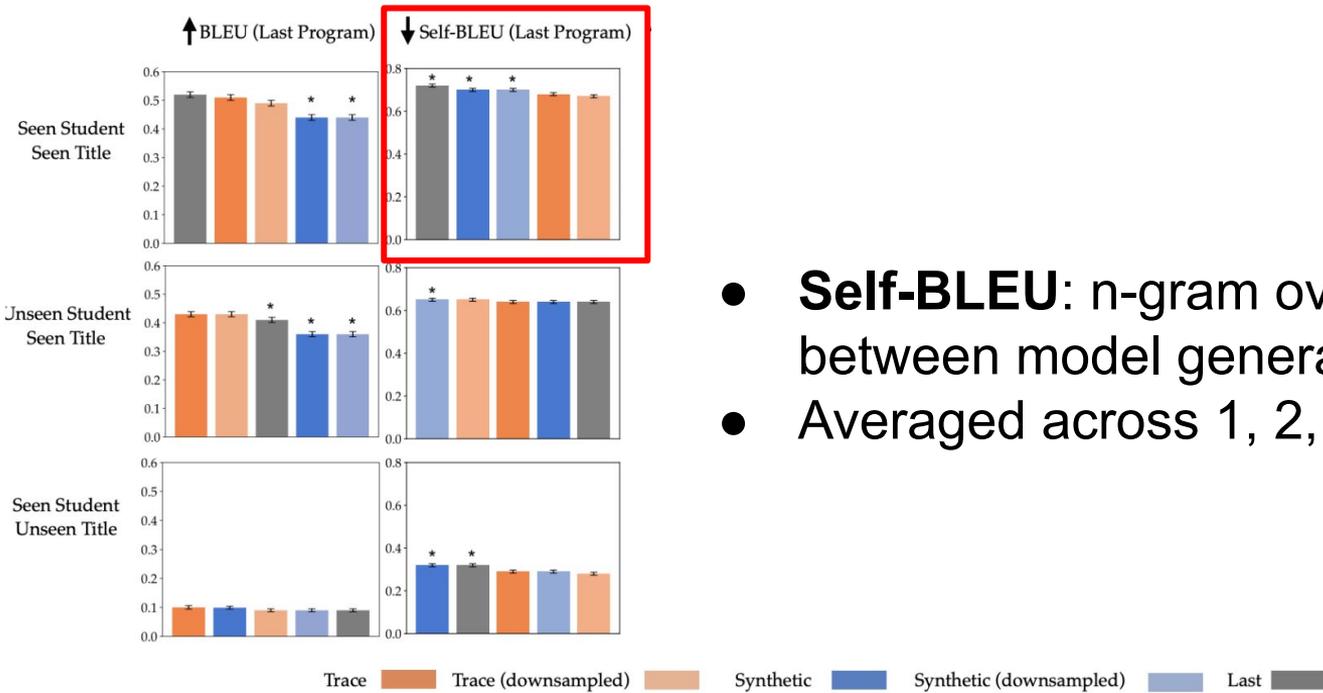
# Evaluation Methodology

- **Behavioral:** Conditional generation of an entire program trace given program title and student embedding
  - 200 classroom assignments, random 50 students
- **Representational:** Linear probe over last layer embeddings to predict different properties
- **Test Sets:** (seen/unseen) students + (seen/unseen) titles
- **Downsampling:** control for number of tokens between `trace/synthetic` and `last`



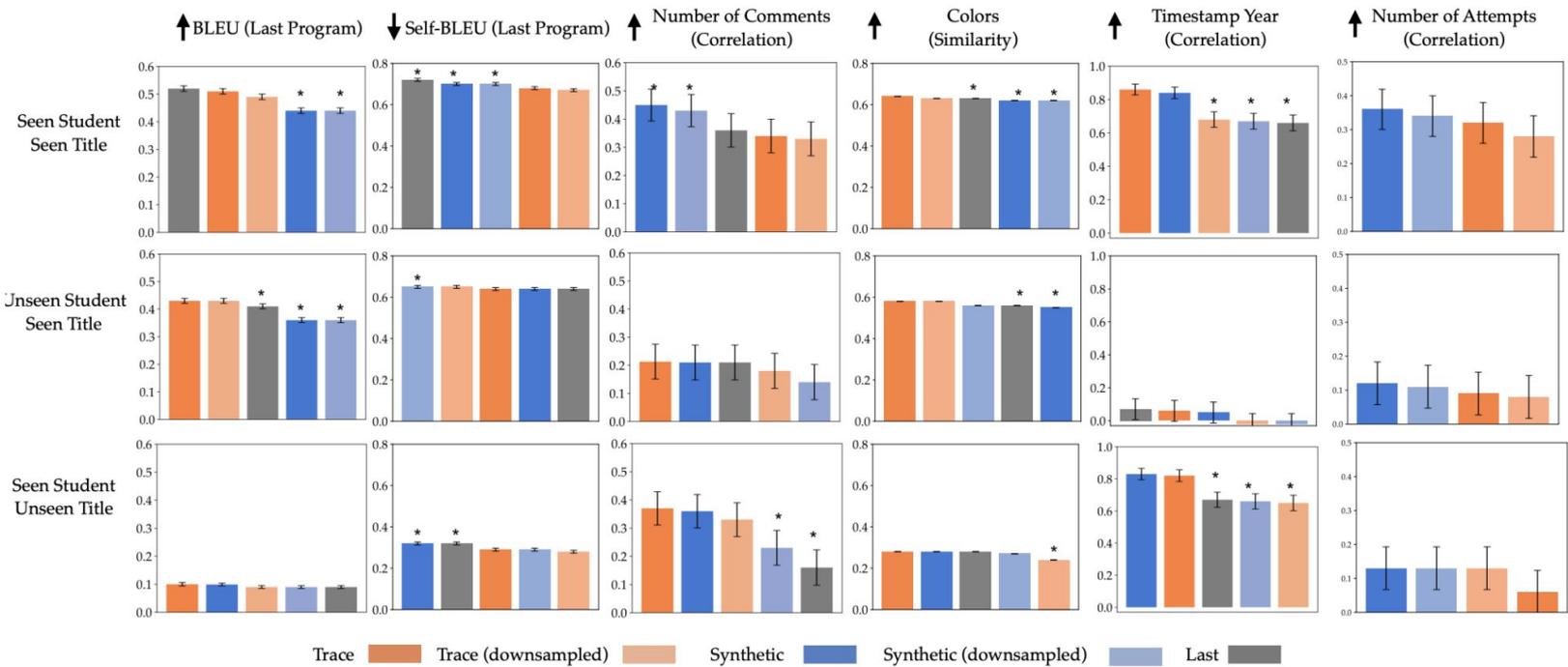
- **BLEU**: n-gram overlap similarity of predicted program to ground truth for user/title
- Averaged across 1, 2, 3, 4-n grams

- Modeling real student traces leads to more accurate last programs ( **trace** >> **synthetic**) for unseen students ( **trace** >> **last**)

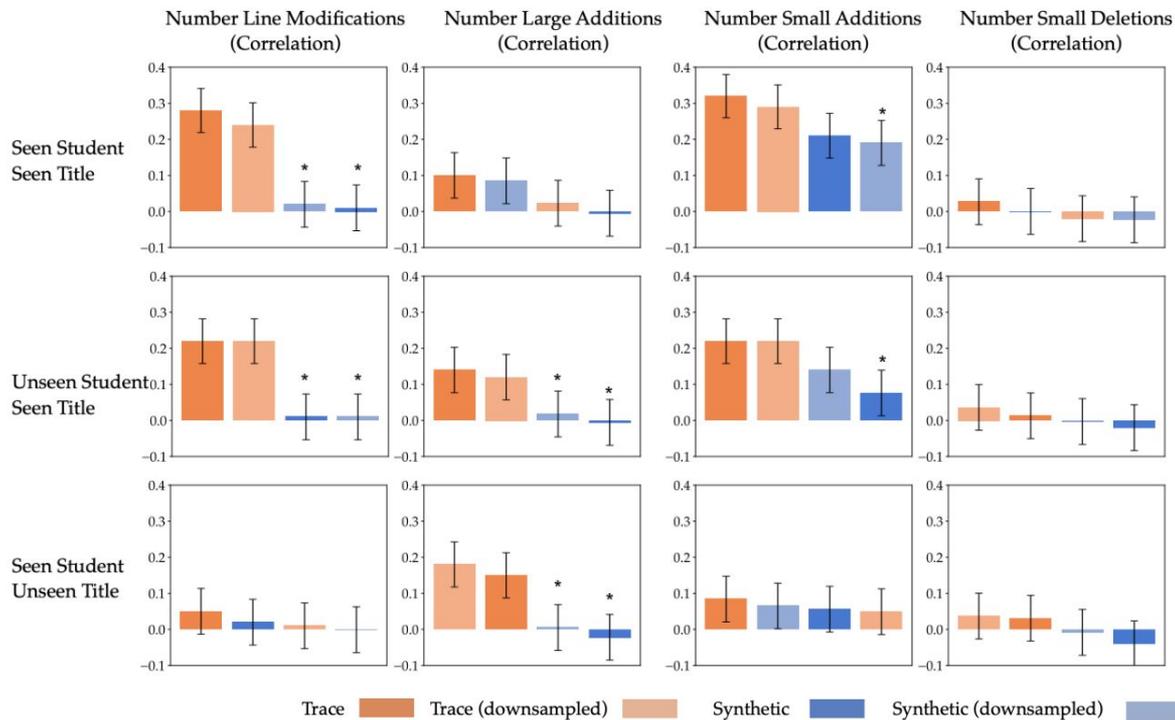


- **Self-BLEU**: n-gram overlap similarity between model generations
- Averaged across 1, 2, 3, 4-n grams

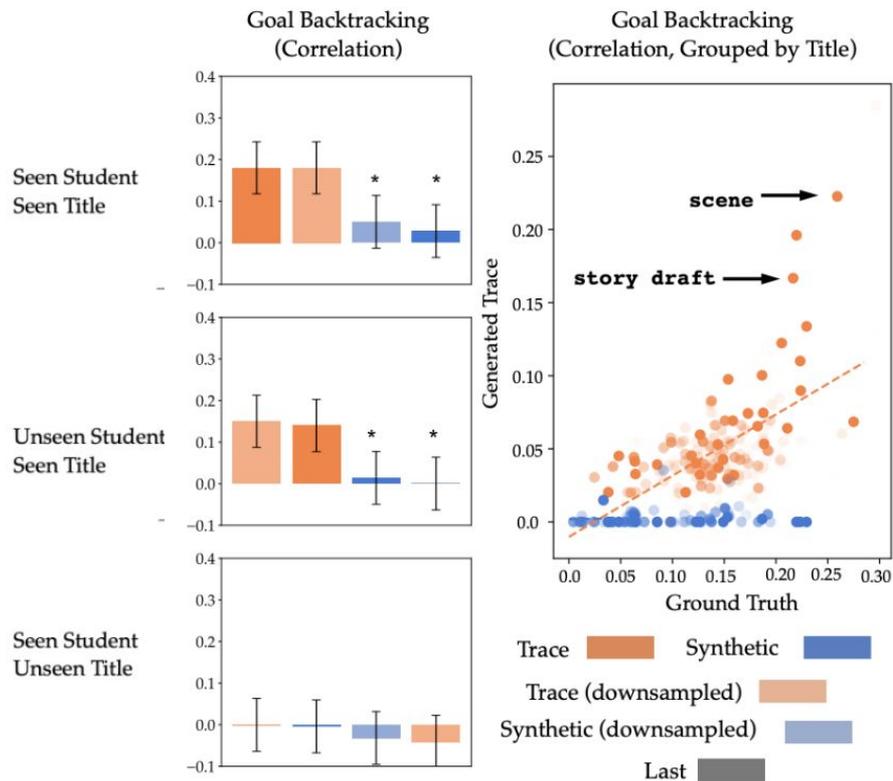
- Modeling real student traces leads to more diverse last programs ( `trace` >> `synthetic/last` ) for seen students and titles



- Title carries a lot of useful information about program behavior
- Student embedding captures time / comment behavior



- Synthetic traces don't fully capture richness of student edit types (**trace >> synthetic**)

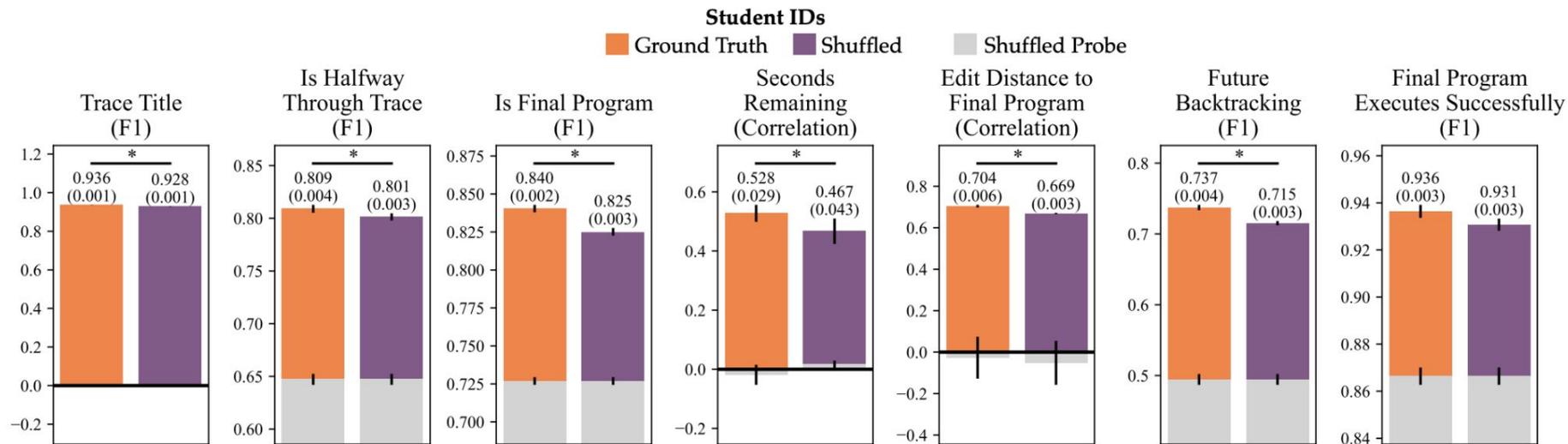


**Goal Backtracking:** average likelihood a student's edit leads to an *increase* in edit distance between current and final program

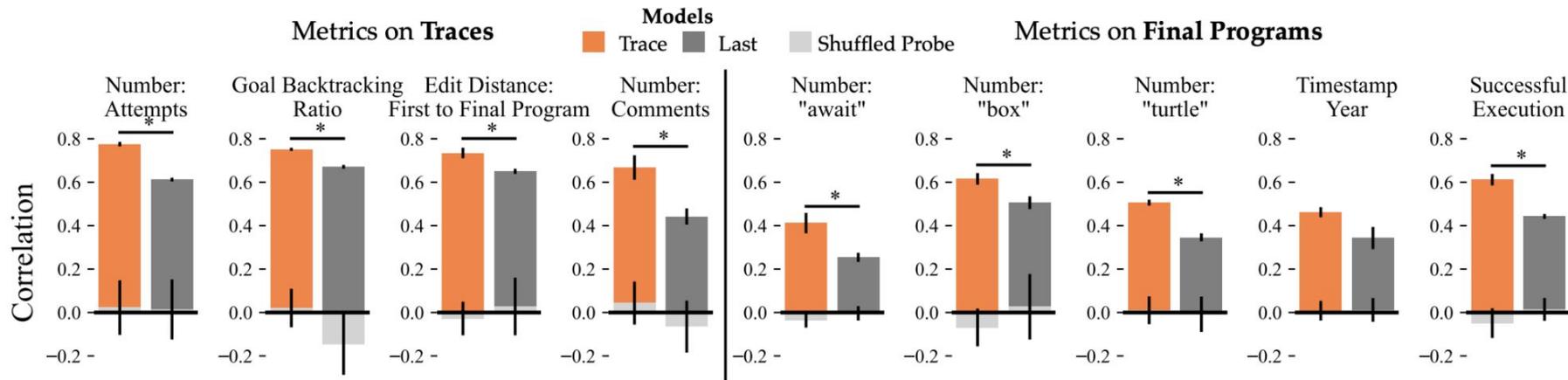
- Backtracking correlation increases for titles that are seen more

# Evaluation Methodology

- **Behavioral:** Conditional generation of an entire program trace given program title and student embedding
  - 200 classroom assignments, random 50 students
- **Representational:** Linear probe over last layer embeddings to predict different properties
- **Test Sets:** (seen/unseen) students + (seen/unseen) titles
- **Downsampling:** control for number of tokens between trace/synthetic and last

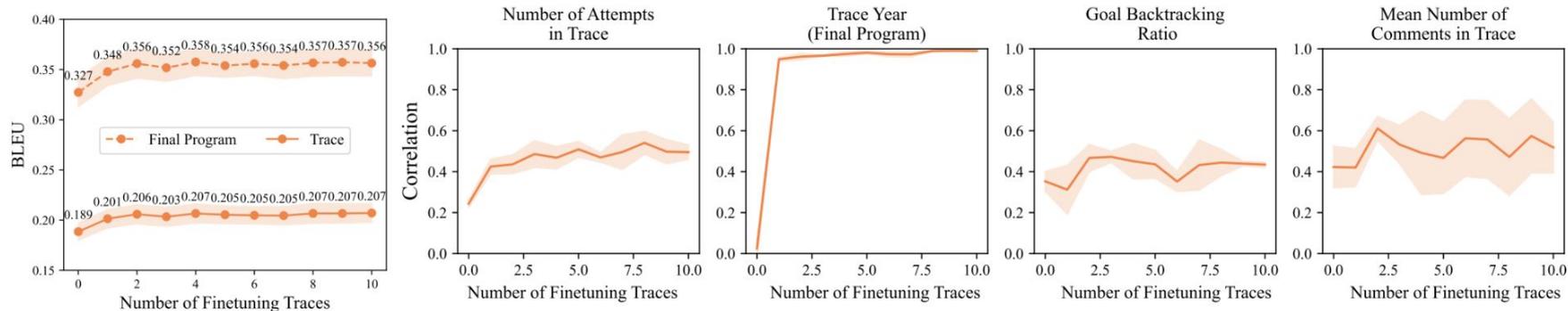


- Code embeddings capture variety of useful properties (compare with gray bars)
- Shuffling student ID worsens performance of code embeddings, showing importance of personalization



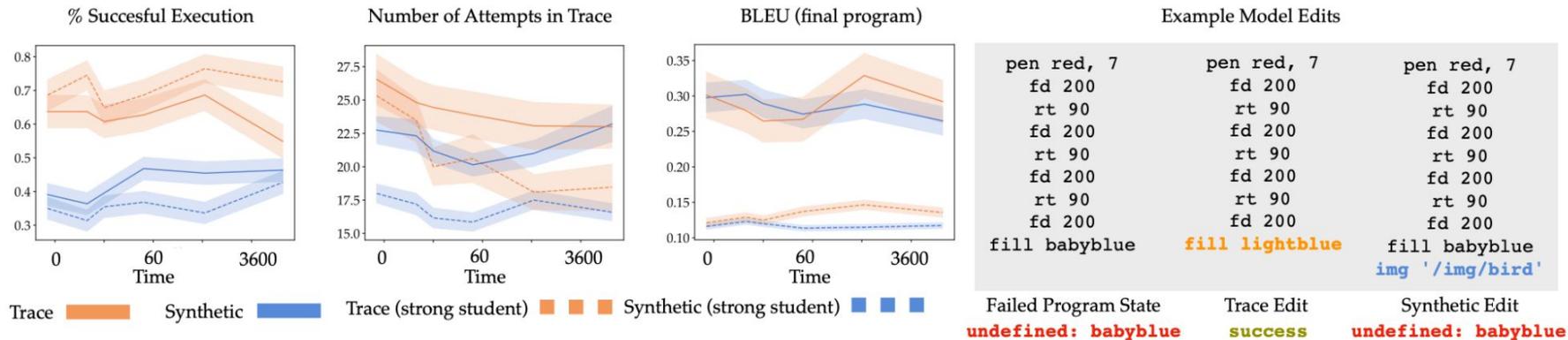
- Student embeddings from **trace** model outperform those from **last**
- E.g. *training on student reasoning helps the model better represent if a student will write correct code for a new assignment*

*What happens if we see a completely new student?*

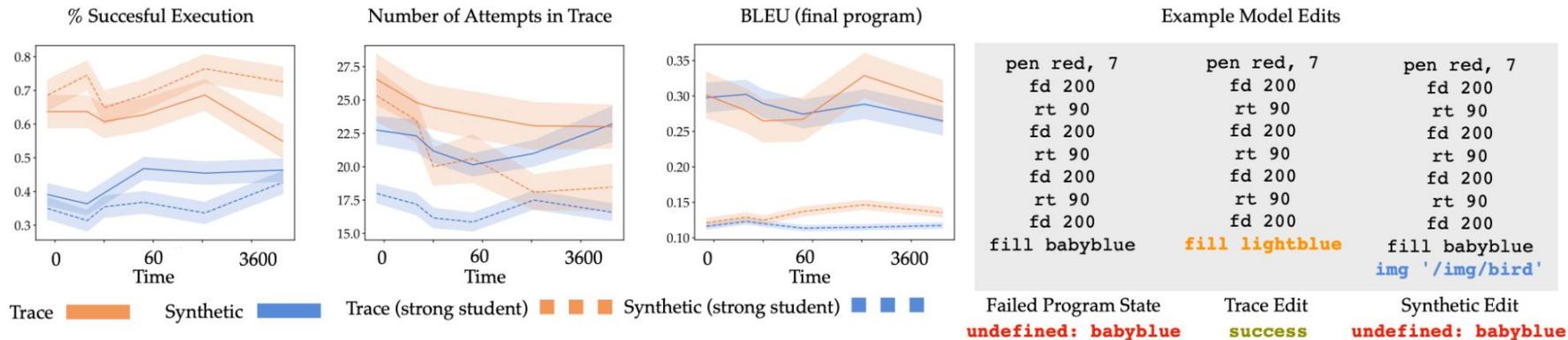


- *What happens if we see a completely new student?*
- **Frozen fine-tuning:** only update the student embedding on a few sample traces from new student
- Fine-tuning improves ability to personalize to a brand new student, but plateaus after ~3 new traces.

*How can trace models be used to help students?*



- Error recovery: conditionally generate trace given a sub-trace leading to an incorrect state
- **trace** >> **synthetic** at error recovery



- Error recovery: conditionally generate trace given a sub-trace leading to an incorrect state
- **trace** >> **synthetic** at error recovery
- Using a strong student embedding improves **trace** accuracy
  - As expected, harms similarity with ground truth of original user
  - Student embeddings → form of controllability
- Increasing “thinking time” (via prefix) leads to shorter trace lengths

# Future Work

- **Generative Populations:** Create synthetic “classroom” of different students
  - E.g. What mistakes might a student make for a program?
  - E.g. If I set a limit of 5 minutes, how many students will progress?
- **Exploration:** Dataset has many examples like “untitled” where students just randomly play around with the environment – what topics to students tend to self-learn? How critical are assignments?
- **Synthetic Design:** How can we create synthetic data with more “human-like” behavior?

<https://github.com/meghabyte/pencilcode-public>