Learning to Learn Courses

Iddo Drori

Massachusetts Institute of Technology

Columbia University in the City of New York

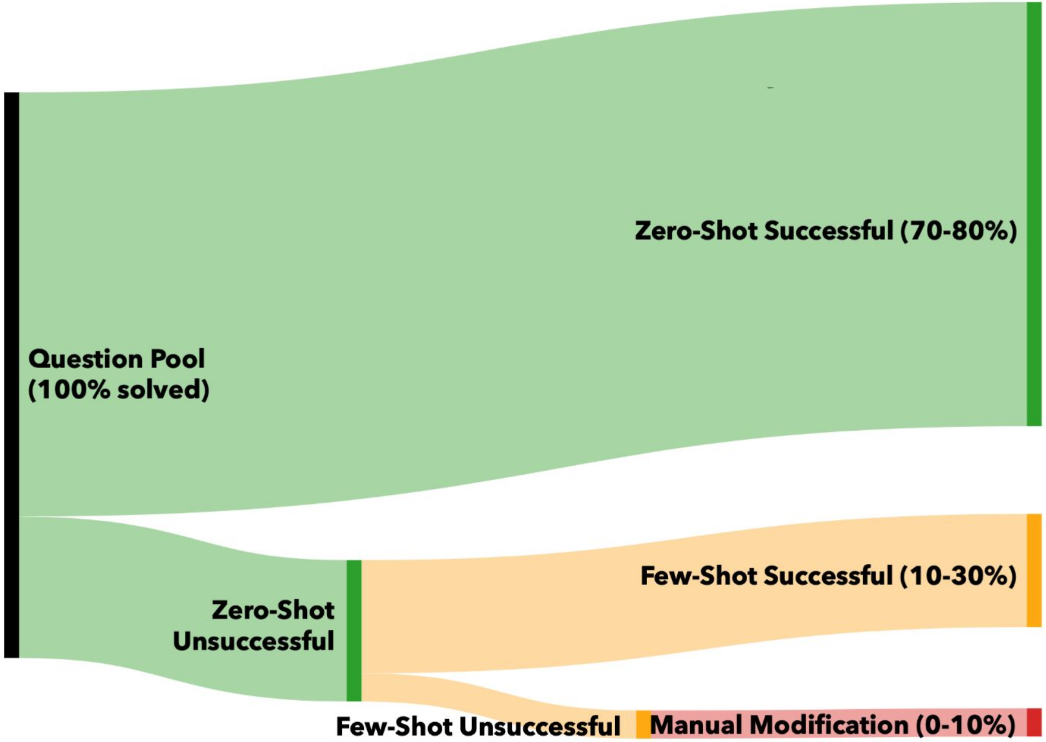
Our Result: Neural Networks Solve Undergraduate Math

1. **Problem**: Solve, explain, and generate undergrad Math course questions

2. **Methods**: Program synthesis and few-shot learning with Transformer pre-trained on text and fine-tuned on code, Codex

3. **Result**: 80-100% automatic accuracy on MIT undergraduate math courses questions by few-shot learning.

Our Result: 80–100% Automatic Accuracy on Undergraduate Mathematics Courses at MIT



Zero-Shot:

input is:

- no change to original prompt
- add "write a program"
- add "using sympy"
- add "using simulations"

Few-Shot:

If zero-shot does not work, perform few-shot using (question, code) pairs:

1. Embed all questions
2. Compute pairwise similarity between embeddings
3. Rank the questions that were zero-shot to select examples for few-shot

Previous Misconception 10/21: Neural Networks cannot do Math

IEEE Spectrum

7 Revealing Ways AIs FAIL

Neural networks can be disastrously brittle, forgetful, and surprisingly bad at math BY CHARLES Q. CHOI

ARTIFICIAL INTELLIGENCE systems can perform more quickly, accurately, reliably, and impartially than humans on a wide range of problems, from detecting cancer to deciding who receives an interview for a job. But AIs have also suffered numerous, sometimes deadly, failures. And the increasing ubiquity of AI means that failures can affect not just individuals but millions of people.

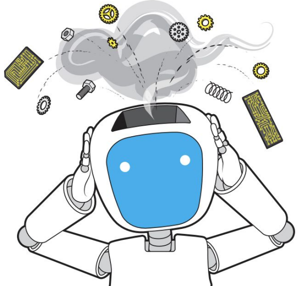
Increasingly, the AI community is cataloging these failures with an eye toward monitoring the risks they may pose. "There tends to be very little information for users to understand how these systems work and what it means to them," says Charlie Pownall, founder of the AI, Algorithmic and Automation Incident & Controversy Repository. "I think this directly impacts trust and confidence in these systems. There are lots of possible reasons why organizations are reluctant to get into the nitty-gritty of what exactly happened in an AI incident or controversy, not the least being potential legal exposure, but if looked at through the lens of trustworthiness, it's in their best interest to do so."

Part of the problem is that the neural network technology that drives many AI systems can break down in ways that remain a mystery to researchers. "It's unpredictable which problems artificial intelligence will be good at, because we don't understand intelligence itself very well," says computer scientist Dan Hendrycks at the University of California, Berkeley.

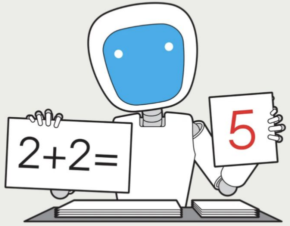
Here are seven examples of AI failures and what current weaknesses they reveal about artificial intelligence. Scientists are discussing possible ways to deal with some of these problems; others currently defy explanation or may, philosophically speaking, lack any conclusive solution altogether.

1) BRITTLENESS

Take a picture of a school bus. Flip it so it lays on its side, as it might be found in the case of an accident



7) Math



CHRIS PHILPOT

Although conventional computers are good at crunching numbers, AIs "are surprisingly not good at mathematics at all," Berkeley's Hendrycks says. "You might have the latest and greatest models that take hundreds of GPUs to train, and they're still just not as reliable as a pocket calculator."

For example, Hendrycks and his colleagues trained an AI on hundreds of thousands of math problems with step-by-step solutions. However, when tested on 12,500 problems from high school math competitions, "it only got something like 5 percent accuracy," he says. In comparison, a three-time International Mathematical Olympiad gold medalist attained 90 percent success on such problems "without a calculator," he adds.

Neural networks nowadays can learn to solve nearly every kind of problem "if you just give it enough data and enough resources, but not math," Hendrycks says. Many problems in science require a lot of math, so this current weakness of AI can limit its application in scientific research, he notes.

It remains uncertain why AI is currently bad at math. One possibility is that neural networks attack problems in a highly parallel manner like human brains, whereas math problems typically require a long series of steps to solve, so maybe the way AIs process data is not as suitable for such tasks, "in the same way that humans generally can't do huge calculations in their head," Hendrycks says. However, AI's poor performance on math "is still a niche topic: There hasn't been much traction on the problem," he adds.

Recent Progress 7/21: Solving Machine Learning Problems

1. Transformer pre-trained on text and a GNN solves ML course
2. Only handles numeric answers
3. Week of curation & training for one course
4. Overfitting
5. Does not scale up to multiple courses

Proceedings of Machine Learning Research 157, 2021 ACML 2021

Solving Machine Learning Problems

Sunny Tran SUNNYT@MIT.EDU
MIT EECS
Pranav Krishna PKRISHNA@MIT.EDU
MIT EECS
Ishan Pakuwal IPAKUWAL@MIT.EDU
MIT EECS
Prabhakar Kafle PKAFLE@MIT.EDU
MIT EECS
Nikhil Singh NSINGH1@MIT.EDU
MIT Media Lab
Jayson Lynch JAYSONL@MIT.EDU
University of Waterloo
Iddo Drori IDRORI@MIT.EDU
MIT EECS

Editors: Vineeth N Balasubramanian and Ivor Tsang

Abstract

Can a machine learn Machine Learning? This work trains a machine learning model to solve machine learning problems from a University undergraduate level course. We generate a new training set of questions and answers consisting of course exercises, homework, and quiz questions from MIT's 6.036 Introduction to Machine Learning course and train a machine learning model to answer these questions. Our system demonstrates an overall accuracy of 96% for open-response questions and 97% for multiple-choice questions, compared with MIT students' average of 93%, achieving grade A performance in the course, all in real-time. Questions cover all 12 topics taught in the course, excluding coding questions or questions with images. Topics include: (i) basic machine learning principles; (ii) perceptrons; (iii) feature extraction and selection; (iv) logistic regression; (v) regression; (vi) neural networks; (vii) advanced neural networks; (viii) convolutional neural networks; (ix) recurrent neural networks; (x) state machines and MDPs; (xi) reinforcement learning; and (xii) decision trees. Our system uses Transformer models within an encoder-decoder architecture with graph and tree representations. An important aspect of our approach is a data-augmentation scheme for generating new example problems. We also train a machine learning model to generate problem hints. Thus, our system automatically generates new questions across topics, answers both open-response questions and multiple-choice questions, classifies problems, and generates problem hints, pushing the envelope of AI for STEM education.

Keywords: Learning to learn; Machine Learning course; Transformers; Graph neural network; Expression trees; Encoder-decoder architecture.

Recent Progress 7/21: Solving Machine Learning Problems

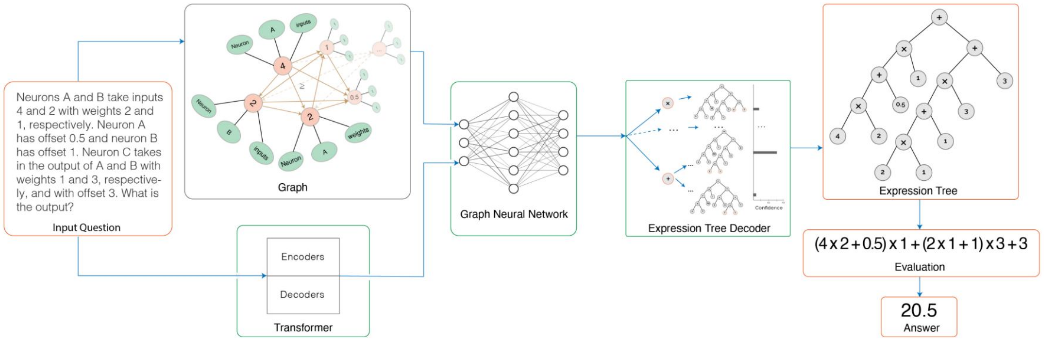


Figure 1: Architecture for solving Machine Learning questions. We create the question text embedding by a Transformer and also a graph representing the relationship between words and values and among values. These are passed through a graph neural network to create a vector in latent space which is then decoded to generate expression trees. The process of inputting the same question into the model is repeated a fixed number of times, resulting in multiple outcomes due to dropout in the GNN. The expression tree with the highest confidence is parsed to get the expression and the final value.

Recent Progress 7/21: Solving Machine Learning Problems

Tran Krishna Pakuwal Kafle Singh Lynch Drori

Figure 2 depicts self-contained hints and the progression of providing hints to the learner until a full solution is be achieved.

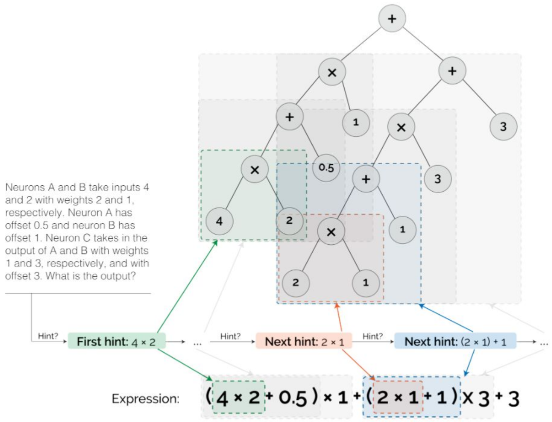


Figure 2: Generating successive hints for a machine learning problem. The expression tree is traversed in a binary search tree order starting from the left-most non-leaf node to give the sub-expression rooted at each non-leaf node as hint.

Recent Progress 7/21: Solving Machine Learning Problems

Tran Krishna Pakuwal Kafle Singh Lynch Drori

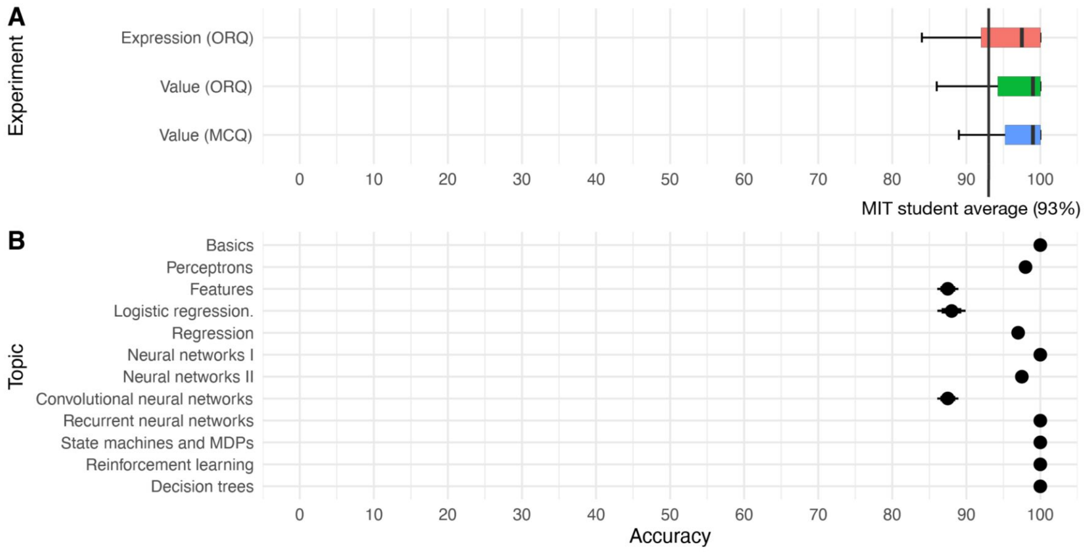


Figure 3: Accuracy distributions across **(A)** evaluation methods and **(B)** topics. The line overlay indicates the average student score.

Recent Progress 7/21: Solving Machine Learning Problems

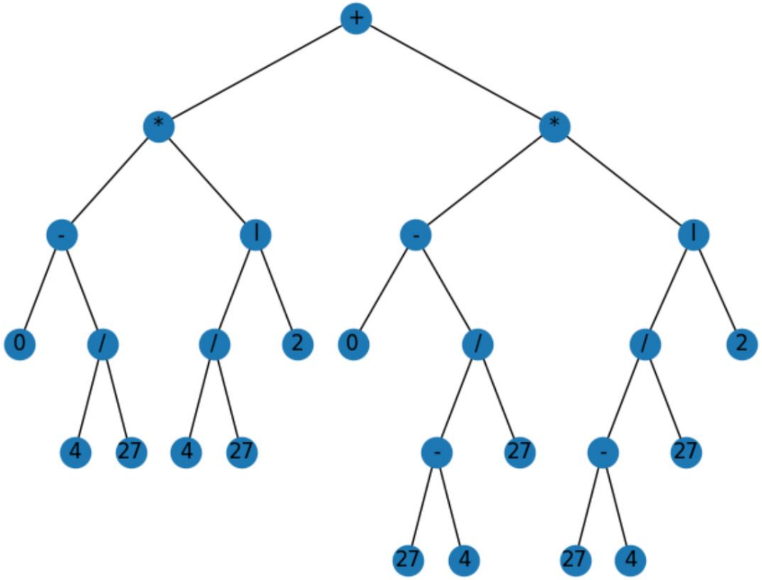


Figure 4: An automatically generated expression tree for the question "What is the entropy of the left side of a region containing 27 points where the plane has 45 points in total and 4 points on the left are positive?" Note "l" denotes the log operator.

Recent Progress 11/21: From 10% to 67% Accuracy

Comparing Learning Approaches for Solving Intro to Astronomy

Avi Shporer
MIT

Sunny Tran
MIT

Nikhil Singh
MIT

Brandon Kates
Cornell University

Jayson Lynch
University of Waterloo

Nakul Verma
Columbia University

Iddo Drori
MIT & Columbia University

- Order of magnitude improvement on sample course: as is prompt, automatic, works.
- From 10% correct using Transformer pre-trained on text to 67% correct by program synthesis using Codex, Transformer pre-trained on text and fine-tuned on code, easily scales up to many courses.
- Specialized model does better but overfits and doesn't scale to any other course.

Approach	Accuracy
Our specialized model (shown in Figure 2)	**92%**
Our generalized use of Codex [3] (shown in Figure 3)	**67%**
GPT-3 [2]	10%
Jurassic-1 [22]	10%
Wolfram Alpha [11]	0%
SMLP [27]	0%

Table 1: Comparison of accuracy of solving Introduction to Astronomy course questions using different approaches. Our specialized approach achieves 92% accuracy. Our generalized approach of turning the questions into the task of writing programs to solve the questions, and synthesizing the programs using OpenAI Codex, achieves 67%. GPT-3 and Jurassic-1 achieve 10%. Wolfram Alpha and SMLP completely fail to answer Astronomy questions.

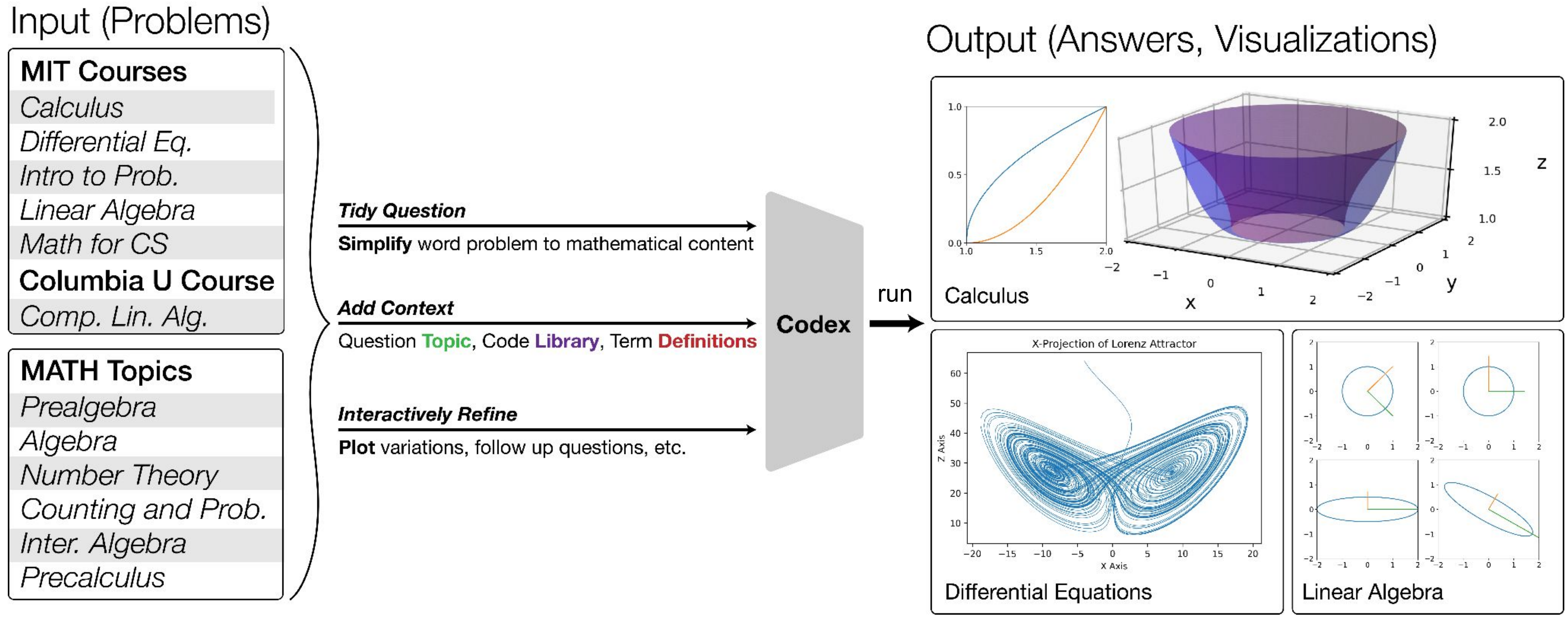
Recent Progress 7/21: Solving Machine Learning Problems

Week	Topic	Exp. Acc. ORQ	Acc. ORQ	Acc. MCQ
1	Basics	1.00	1.00	1.00
2	Perceptrons	0.98	0.98	0.98
3	Features	0.86	0.86	0.89
4	Logistic regression	0.86	0.86	0.90
5	Regression	0.97	0.97	0.97
6	Neural networks I	1.00	1.00	1.00
7	Neural networks II	0.97	0.97	0.98
8	Convolutional neural networks	0.84	0.86	0.89
9	Recurrent neural networks	1.00	1.00	1.00
10	State machines and MDPs	0.94	1.00	1.00
11	Reinforcement learning	1.00	1.00	1.00
12	Decision trees	1.00	1.00	1.00
	Overall average over topics	0.95	0.96	0.97

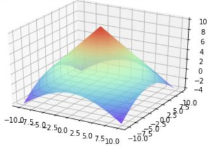
Table 1: Accuracy achieved using our system for each topic taught in MIT's Introduction to Machine Learning course, 6.036. Our system demonstrates an overall average expression accuracy (percent of correct expressions) of 95% and value accuracy (percent of correct values) of 96% for open response questions (ORQ), and accuracy (percent of correct values) of 97% for multiple-choice questions (MCQ), achieving grade A performance in real-time.

12/31/21: A Neural Network Solves and Generates Mathematics Problems by Program Synthesis: Calculus, Differential Equations, Linear Algebra, and More

Iddo Drori, Sunny Tran, Roman Wang, Newman Cheng, Kevin Liu, Leonard Tang, Elizabeth Ke, Nikhil Singh, Taylor L. Patti, Jayson Lynch, Avi Shporer, Nakul Verma, Eugene Wu, Gilbert Strang



Example Questions and Solutions

ID	Course	Question	Solution
1	18.01 Single Variable Calculus	A bacteria population is 4000 at time $t = 0$ and its rate of growth is $1000 * 2^t$ bacteria per hour after t hours. What is the population after one hour?	$4000 + \frac{1000}{\log(2)}$
2	18.02 Multi-variable Calculus	Describe the graph of the function f: $f(x, y) = 10 - \sqrt{x^2 + y^2}$	
3	18.03 Differential Equations	Find general solutions of the differential equations. If an initial condition is given, find the corresponding particular solution. Throughout, primes denote derivatives with respect to x. $y' + y = 2$, $y(0) = 0$	$y(x) = 2(1 - e^{-x})$
4	18.05 Introduction to Probability and Statistics	Calculate the probability of getting a three-of-a-kind poker hand.	0.021128
5	18.06 Linear Algebra	Find a combination $x_1 w_1 + x_2 w_2 + x_3 w_3$ that gives the zero vector with $x_1 = 1$. w_1 is the vector (1;2;3). w_2 is the vector $(4; 5; 6)$. w_3 is the vector $(7; 8; 9)$.	$x_1 = 1, x_2 = -2, x_3 = 1$
6	6.042 Mathematics for Computer Science	Find a number $x \in \{0, 1, ..., 112\}$ such that $11x \equiv 1$ (mod 113).	72
7	COMS3251 Computational Linear Algebra	Given a d-dimensional non-zero vector v, compute the rank of the matrix vv'	1
8	MATH Prealgebra	What is the greatest common factor of 84, 112 and 210?	14
9	MATH Algebra	Let N, O be functions such that $N(x) = 2\sqrt{x}$, and $O(x) = x^2$. What is $N(O(N(O(N(O(3))))))$?	24
10	MATH Number Theory	How many four-digit numbers whose digits add up to 9 are divisible by 11?	0
11	MATH Counting and Probability	A standard six-sided fair die is rolled four times. The probability that the product of all four numbers rolled is a perfect square is $\frac{m}{n}$, where m and n are relatively prime positive integers. Find $m + n$.	187
12	MATH Intermediate Algebra	Given that $x^2 + y^2 = 14x + 6y + 6$, find the largest possible value of $3x + 4y$.	73
13	MATH Precalculus	If the six solutions of $x^6 = -64$ are written in the form $a + bi$, where a and b are real, find the product of those solutions with $a > 0$.	4

12/31/21: A Neural Network Solves and Generates Mathematics Problems by Program Synthesis: Calculus, Differential Equations, Linear Algebra, and More

Iddo Drori, Sunny Tran, Roman Wang, Newman Cheng, Kevin Liu, Leonard Tang, Elizabeth Ke, Nikhil Singh, Taylor L. Patti, Jayson Lynch, Avi Shporer, Nakul Verma, Eugene Wu, Gilbert Strang

1. Program synthesis using Codex pre-trained on text and fine-tuned on code
2. Executing programs results in answer
3. Handle answers with multiple modalities: numeric, text, plots..
4. Few shot learning in seconds
5. Verify that does not overfit by solving a new course
6. Scales up to multiple courses

Our Approach: Program Synthesis

1. Turn question into programming task

   (i) As is; (ii) Add context; (iii) Few shot learning

2. Automatically generate program using a Transformer, OpenAI Codex, pre-trained on text and fine-tuned on code

3. Execute program to obtain and evaluate answers

4. Automatically explain correct solution using Codex

Our Approach: Program Synthesis

5 Automatically generate new questions that are qualitatively indistinguishable from human-written questions.

Scaling up to Many Courses

- 32 courses
- 13 departments
- 2 schools (eng. science)
- 8 universities

ID	University	Department	Course	Number
1	MIT	Aeronautics and Astronautics	Unified Engineering 1-4	16.01-4
2	MIT	Aeronautics and Astronautics	Estimation and Control of Aerospace Systems	16.30
3	MIT	Aeronautics and Astronautics	Intro to Propulsion Systems	16.50
4	MIT	Materials Science and Engineering	Fundamentals of Materials Science	3.012
5	MIT	Materials Science and Engineering	Mathematics for Materials Scientists and Engineers	3.016
6	MIT	Materials Science and Engineering	Introduction to Solid-State Chemistry	3.091
7	MIT	Chemical Engineering	Chemical and Biological Reaction Engineering	10.37
8	MIT	Chemistry	Principles of Chemical Science	5.111
9	MIT	IDSS	Statistical Thinking & Data Analysis	IDS.013(J)
10	MIT	Electrical Engineering & Computer Science	Signal Processing	6.003
11	MIT	Electrical Engineering & Computer Science	Introduction to Machine Learning	6.036
12	MIT	Electrical Engineering & Computer Science	Mathematics for Computer Science	6.042
13	MIT	Physics	Introduction to Astronomy	8.282
14	MIT	Nuclear Science and Engineering	Introduction to Nuclear Engineering and Ionizing Radiation	22.01
15	MIT	Economics	Principles of Microeconomics	14.01
16	MIT	Mechanical Engineering	Hydrodynamics	2.016
17	MIT	Mechanical Engineering	Nonlinear Dynamics I: Chaos	2.050J
18	MIT	Mechanical Engineering	Information & Entropy	2.110J
19	MIT	Mechanical Engineering	Marine Power and Propulsion	2.611
20	MIT	Mathematics	Single Variable Calculus	18.01
21	MIT	Mathematics	Multi-variable Calculus	18.02
22	MIT	Mathematics	Differential Equations	18.03
23	MIT	Mathematics	Introduction to Probability and Statistics	18.05
24	MIT	Mathematics	Linear Algebra	18.06
25	MIT	Mathematics	Theory of Numbers	18.781
26	Harvard	Statistics	Probability	STATS110
27	Princeton	Mathematics	Calculus II	MAT104
28	Cornell	Computer Science	Computer Architecture	CS4420
29	Columbia	Computer Science	Computational Linear Algebra	COMS3251
30	Yale	Mathematics	Fundamentals of Physics	PHYS200
31	UPenn	Mathematics	Calculus	MATH110
32	Brown	Mathematics	Intermediate Calculus	MATH0180

Turning Questions Into Programming Tasks

- Question

  Find the distance between the points (-2, 4) and (3, -8)

- Program synthesis task

  Write a program that finds the distance between the points (-2, 4) and (3, -8)

Program Synthesis Results in a Program

"""

Write a program to compute the reduced row echelon form (rref) of a matrix

Compute the rref of the matrix [-1,2,1,0;2,1,0,-1;5,0,-2,6]

"""

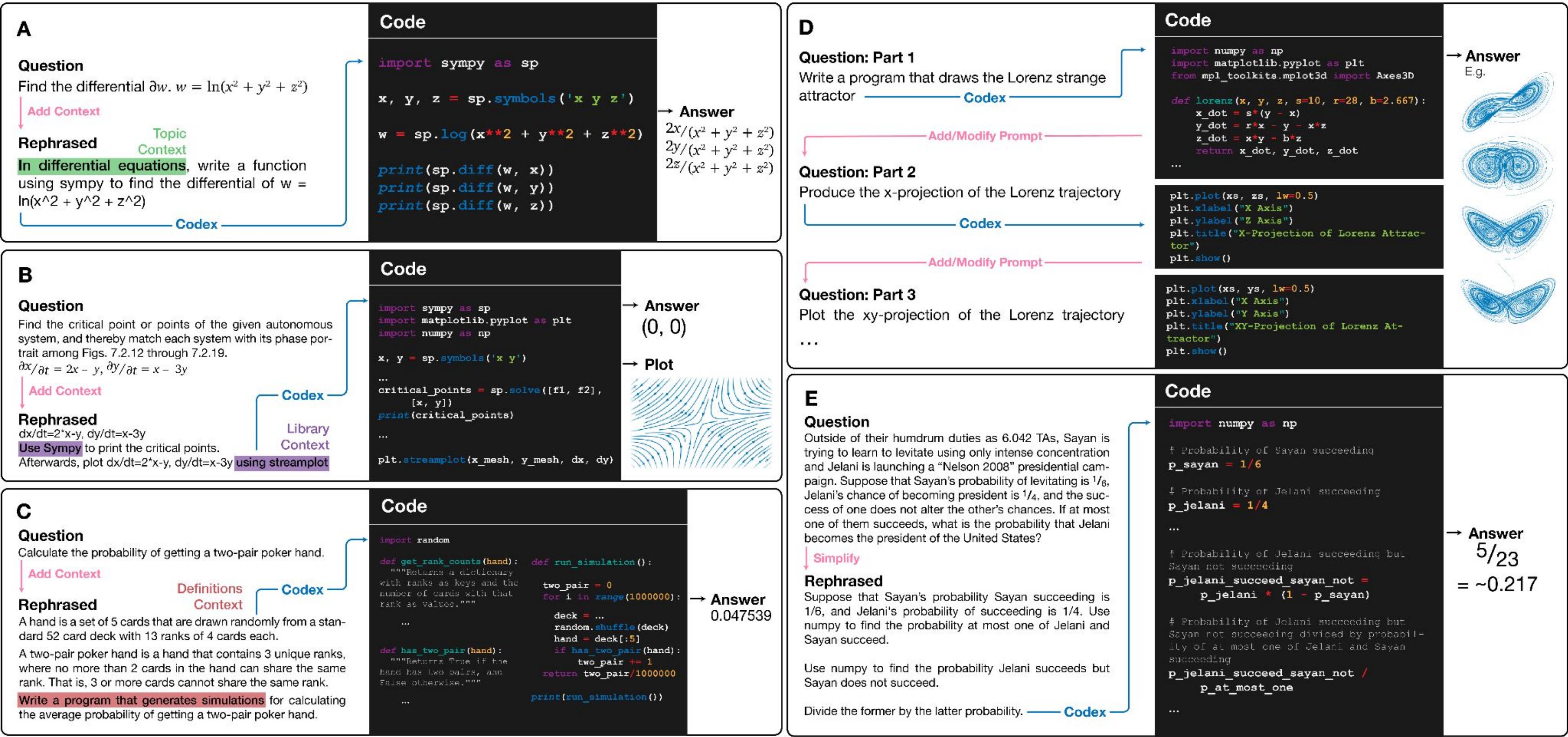
```
def rref(matrix):
    if not matrix: return
    num_rows = len(matrix)
    num_cols = len(matrix[0])

    # Start at the last column and work backwards.
    for col in range(num_cols - 1, -1, -1):

        # Find the row with the leading non-zero entry in this column.
        for row in range(num_rows):
            if matrix[row][col] != 0: break

        # If there is no leading non-zero entry, then all entries are zero.
        if row == num_rows: continue

        # Swap the current row with the one that has the leading non-zero entry.
        matrix[row], matrix[num_rows - 1] = matrix[num_rows - 1], matrix[row]

        # Eliminate all other entries in this column.
        for r in range(num_rows):
            if r == num_rows - 1: continue  # skip pivot row; already done above

            # Eliminate current row of all other entries in this column.
            multiplier = matrix[r][col] / matrix[num_rows - 1][col]  # get multiplier to eliminate value at [r][c] from rest of row (except pivot)
            for c in range(col, num_cols):
                matrix[r][c] -= multiplier * matrix[num_rows - 1][c]

    return matrix

if __name__ == "__main__":
    print(rref([[1, 2, 3], [4, 5, 6], [7, 8, 9]]))
```

Running the Program Solves the Question

```
def rref(matrix):
    if not matrix: return
    num_rows = len(matrix)
    num_cols = len(matrix[0])

    # Start at the last column and work backwards.
    for col in range(num_cols - 1, -1, -1):

        # Find the row with the leading non-zero entry in this column.
        for row in range(num_rows):
            if matrix[row][col] != 0: break

        # If there is no leading non-zero entry, then all entries are zero.
        if row == num_rows: continue

        # Swap the current row with the one that has the leading non-zero entry.
        matrix[row], matrix[num_rows - 1] = matrix[num_rows - 1], matrix[row]

        # Eliminate all other entries in this column.
        for r in range(num_rows):
            if r == num_rows - 1: continue  # skip pivot row; already done above

            # Eliminate current row of all other entries in this column.
            multiplier = matrix[r][col] / matrix[num_rows - 1][col]  # get multiplier to eliminate value at [r][c] from rest of row (except pivot)
            for c in range(col, num_cols):
                matrix[r][c] -= multiplier * matrix[num_rows - 1][c]

    return matrix

if __name__ == "__main__":
    print(rref([[1, 2, 3], [4, 5, 6], [7, 8, 9]]))
```

```
[[0.0, 8.0, -21.0], [0.0, 0.0, -12.0], [1, 0.0, 3.0]]
```

Add Question or Program Context: Available to Students



500+ Examples

- In topic and subtopic write a program that answers question
- Write a program using specific library that answers question
- Write a program that generates simulations that answers question

… emerging property: simple usage while generalizing

Automatic Accuracy on Mathematics Courses

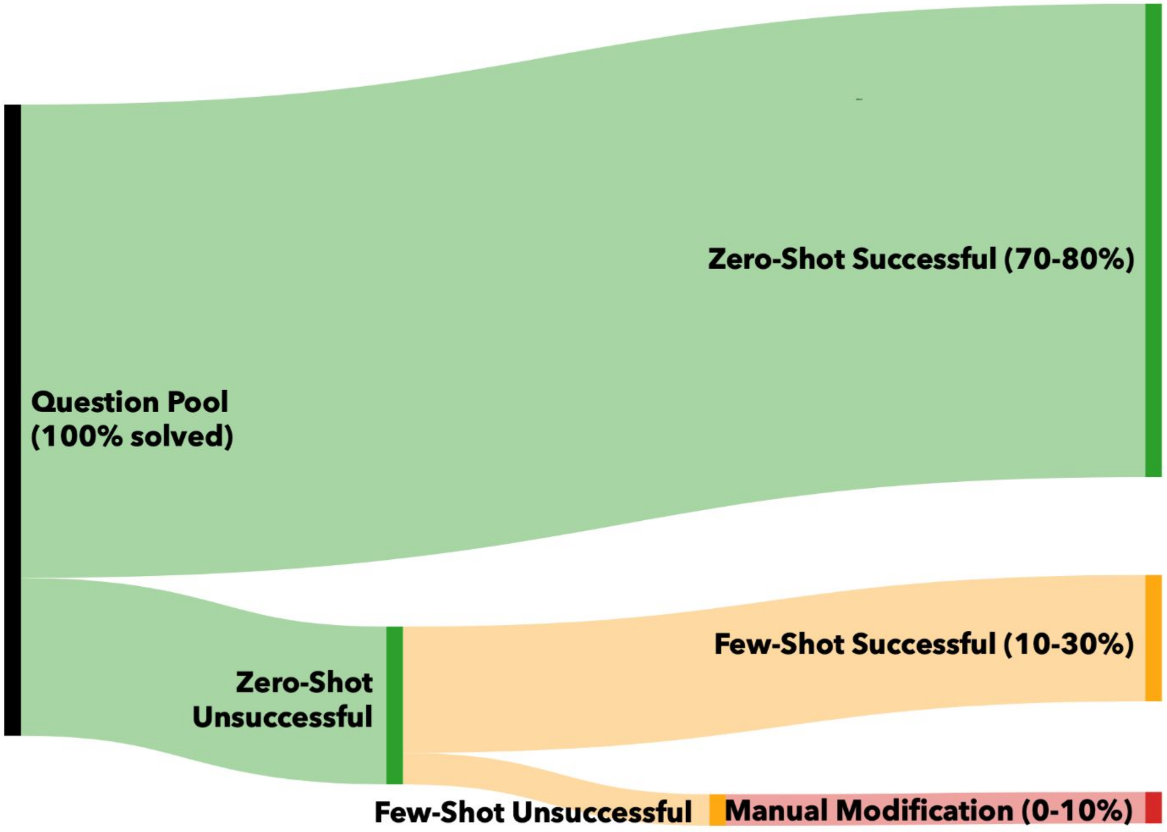
- GPT-3: only 10-20% automatic accuracy, and only text output.

- Accuracy for questions as-is or automatically turned into programming tasks:

  Zero-shot learning using Codex: 70-80% automatic accuracy

  Few-shot learning with (question, program) pairs using Codex: 80-100% automatic accuracy

90–100% Automatic Accuracy on Mathematics Courses



Zero-Shot:

input is:

- no change to original prompt
- add "write a program"
- add "using sympy"
- add "using simulations"

Few-Shot:

If zero-shot does not work, perform few-shot using (question, code) pairs:

1. Embed all questions
2. Compute pairwise similarity between embeddings
3. Rank the questions that were zero-shot to select examples for few-shot

Example Course: 18.05

Id	Course	Topic	Original Proble	Solution	Codex Input	Codex Code	Codex Code Run Output	Codex Code Explanatior	Libraries	Output Evalu	Solution Type	GPT-3 Response	GPT-3 Evaluation	Nearest Questions
1	Introduction to Pro	Probability	Calculate the prc	0.021128	Create a card de	import random		1. Create a deck of cards.	random	Correct	Automatic Zero-shot	There are 13 possible \	Incorrect	[25, 5, 2, 3, 4, 8, 6, 7, 2
2	Introduction to Pro	Probability	In class we work	0.5833333333	Given non-transi	import random		1. Define a function roll tha	random	Correct	Automatic Zero-shot	The probability that whi	Incorrect	[3, 6, 7, 8, 9, 18, 24, 17,
3	Introduction to Pro	Probability	In class we work	0.6944444444	In class we work	import random		1. Define a function that ta	random	Correct	Automatic Zero-shot	The probability that gre	Incorrect	[2, 6, 7, 8, 9, 24, 18, 21,
4	Introduction to Pro	Conditional Prob	Mr. Jones has tw	0.5	Mr. Jones has tw	import random		1. We start by initializing th	random	Correct	Automatic Zero-shot	There is a 1/4 chance t	Incorrect	[5, 24, 7, 25, 2, 3, 6, 8,
5	Introduction to Pro	Conditional Prob	Mr. Smith has tw	0.3333333333	Mr. Smith has tw	import random		1. We start by initializing th	random	Correct	Automatic Zero-shot	There is a 50% chance	Incorrect	[4, 24, 7, 6, 25, 8, 2, 3,
6	Introduction to Pro	Discrete Randon	Let X be the resu	1.118	Let X be the resu	import numpy as np		1. It's defining a function c	numpy	Correct	Automatic Zero-shot	The standard deviation	Incorrect	[7, 8, 9, 24, 2, 3, 18, 17,
7	Introduction to Pro	Discrete Randon	Let X be the resu	1.708	Let Y be the resu	import numpy as np		1. def std_dev(n):	numpy	Correct	Automatic Zero-shot	The standard deviation	Incorrect	[6, 8, 9, 24, 2, 3, 18, 17,
8	Introduction to Pro	Discrete Randon	Let X be the resu	1.021	Let X be the resu	import random		1. def roll_die(sides):	random	Correct	Automatic Zero-shot	The standard deviation	Incorrect	[6, 7, 9, 2, 24, 3, 18, 17,
9	Introduction to Pro	Discrete Randon	Let X be the resu	55	Let X be the resu	import random		1. It's creating a class calle	random	Correct	Automatic Zero-shot	Your expected total gai	Incorrect	[6, 7, 8, 22, 21, 2, 3, 24,
10	Introduction to Pro	Counting, Contir	Boxes of Raisin	750	Boxes of Raisin	def f(h):		1. Define a function f(h) that returns the der		Correct	Automatic Zero-shot	There are 1200 raisins	Incorrect	[11, 17, 18, 21, 22, 23,
11	Introduction to Pro	Probability, Cont	Boxes of Raisin	0.4666666667	Boxes of Raisin	def f(h):		1. f(h) is the density functio	scipy	Correct	Automatic Few-shot (1)	The probability that a ra	Incorrect	[10, 21, 22, 20, 12, 19,
12	Introduction to Pro	Exponential Ran	Suppose we are	0.1	Suppose we are	import numpy as np		1. Simulate T1, T2, and T3	numpy	Correct	Automatic Zero-shot	The expected time befo	Incorrect	[18, 17, 8, 6, 7, 19, 20,
13	Introduction to Pro	Central Limit The	To head the new	0.16	50% of the popu	# The population proportion is 0.5		1. The population proporti	stats	Correct	Automatic Zero-shot	The central limit theore	Incorrect	[14, 18, 20, 17, 6, 7, 8,
14	Introduction to Pro	Central Limit The	To head the new	0.0145	Suppose that 30	# The population proportion is 0.3		1. The population proporti	stats	Correct	Automatic Zero-shot	The central limit theore	Incorrect	[13, 20, 18, 6, 7, 8, 17,
15	Introduction to Pro	Uniform Randon	Alice and Bob ar	0.5	Alice and Bob ar	import numpy as np		1. Generate a random nur	numpy	Correct	Automatic Zero-shot	The probability that Alic	Correct	[16, 20, 19, 2, 13, 3, 6,
16	Introduction to Pro	Uniform Randon	Alice and Bob ar	0.4375	Alice and Bob ar	import numpy as np		1. Generate n random nur	numpy	Correct	Automatic Zero-shot	The probability that Alic	Incorrect	[15, 20, 19, 13, 14, 12,
17	Introduction to Pro	Uniform Randon	Suppose we hav	1.2	Suppose we hav	import numpy as np		1. It's creating a class calle	numpy	Correct	Automatic Zero-shot	The maximum likelihoo	Correct	[18, 24, 6, 7, 19, 8, 20,
18	Introduction to Pro	Uniform Randon	Suppose we hav	10.5	Suppose we hav	import numpy as np		1. It's initializing the class	numpy	Correct	Automatic Zero-shot	The maximum likelihoo	Correct	[17, 24, 7, 6, 8, 19, 20,
19	Introduction to Pro	Null Hypothesis	One generates a	0.1	One generates a	import numpy as np		1. Generate a random nur	numpy	Correct	Automatic Zero-shot	The probability of a typ	Incorrect	[20, 18, 17, 6, 7, 24, 8,
20	Introduction to Pro	Null Hypothesis	One generates a	0.72	One generates a	import numpy as np		1. Generate a random nur	Numpy	Correct	Automatic Zero-shot	The probability of a typ	Incorrect	[19, 18, 17, 6, 7, 24, 8,
21	Introduction to Pro	Prior Odds	You have a draw	0.2	You have a draw	# Probability of choosing a 0.3 coin		1. We have a class called	\	Correct	Automatic Zero-shot	There are 50 coins in th	Correct	[22, 9, 6, 7, 17, 8, 18, 1
22	Introduction to Pro	Prior Odds	You have a draw	0.6	You have a draw	# Solution		1. We have a class called	\	Correct	Automatic Zero-shot	The odds of choosing a	Incorrect	[21, 9, 6, 7, 18, 8, 17, 1
23	Introduction to Pro	Confidence Inter	Suppose μ is the	155	A gaussian distri	import math		1. Import the math module	Math	Correct	Automatic Zero-shot	You would need to mea	Incorrect	[20, 19, 17, 18, 14, 11,
24	Introduction to Pro	Joint Distribution	Suppose X and Y	1.714285714	Suppose X and Y	from sympy import integrate, symbols, simp		1. Define the variables x a	sympy	Correct	Automatic Zero-shot	c = 1/3	Incorrect	[17, 18, 7, 6, 8, 19, 20,
25	Introduction to Pro	Probability	Calculate the prc	0.047539	Using simulation	import random		1. Create a deck of cards.	random	Correct	Automatic Few-shot (1)	There are 13 possible p	Incorrect	[1, 5, 4, 24, 7, 8, 6, 2, 3,

Few-Shot Learning Method

- Embed all questions
- Compute pairwise similarity between embeddings
- For questions that cannot be zero-shot, rank other questions that were zero-shot, by similarity to their embeddings, and use that ranking to select the (question,code) examples for few-shot.

Our Approach: Program Synthesis

- Why does this work so well?

Does GPT-3 Do Math? No (fails <7% on MATH dataset)

Q: What is the magnitude of the vector [1, 2, 1]?

GPT-3:
The magnitude of the vector is the square root of the sum of the squares of the three numbers, or about 2.77.

OR

GPT-3: **The magnitude of the vector is sqrt(3).**

Does Codex Do Math? Yes (solves MATH dataset)

Programming task: Write a program that computes the magnitude of the vector [1, 2, 1]

Codex:
```
import math

def magnitude(x, y, z):
    return math.sqrt(x**2 + y**2 + z**2)

print(magnitude(1, 2, 1))
```

```
2.449489742783178
```

- Codex is pre-trained on text and fine-tuned on code.
- Math and STEM problems may be represented as expression trees and any programs may be represented as a trees.
- Not overfitting: Solving a new course COMS3251 not online.

Generating New Questions

- Few shot learning using Codex
- Indistinguishable by students from human written questions

ID	Course	Machine-generated question	Closest question in the dataset	Similarity
1	18.01 Single-Variable Calculus	Find the area of the region bounded by the curve and the x-axis. $y = x^2 \sin(x), 0 \leq x \leq \pi$	Find the area of the region under the given curve from 1 to 2. $y = (x^2+1)/(3x - x^2)$	0.61
2	18.02 Multi-Variable Calculus	Find $a \times b$. $a = \langle 9, -2, 1 \rangle, b = \langle -2, 1, 1 \rangle$	Find $a \times b$. $a = \langle 5, -1, -2 \rangle, b = \langle -3, 2, 4 \rangle$	0.87
3	18.03 Differential Equations	Use the method of separable variables to solve the initial-value problem $\frac{dy}{dx} = 5e^x, y(2) = 12$ when $x = 2$	Separate variables and use partial fractions to solve the initial value problems. Use either the exact solution or a computer-generated slope field to sketch the graphs of several solutions of the given differential equation, and highlight the indicated particular solution. $f'(x) = 3f(x)(5 - f(x)), f(0) = 8$	0.21
4	18.05 Introduction to Probability and Statistics	Let X be a uniformly distributed random variable over the interval $[0, 1)$. Find $\mathbb{E}[X^2]$	Let X be the result of rolling a fair 4-sided die. Let Y be the result of rolling a fair 6-sided die. You win $2X$ dollars if $X > Y$ and lose 1 dollar otherwise. After playing this game 60 times, what is your expected total gain?	0.29
5	18.06 Linear Algebra	Write a Matlab code to determine if the given matrix $A = [1, 1; 4, 4]$ is positive semidefinite and if it is negative semidefinite.	Find $A'A$ if the columns of A are unit vectors, all mutually perpendicular.	0.21
6	6.042 Mathematics for Computer Science	A student is taking a test consisting of n multiple-choice questions. Each question has five possible answers, and only one of them is correct. The student knows that the probability that any particular question is answered correctly is $\frac{1}{5}$. Let X be the number of questions answered correctly by the student. What is $\mathbb{E}(X)$?	MIT students sometimes delay laundry for a few days. Assume all random values described below are mutually independent. A busy student must complete 3 problem sets before doing laundry. Each problem set requires 1 day with probability $\frac{2}{3}$ and 2 days with probability $\frac{1}{3}$. Let B be the number of days a busy student delays laundry. What is $\mathbb{E}(B)$?	0.47
7	COMS3251 Computational	Find a combination of the vectors $\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$ that gives the vector $\begin{bmatrix} 1 & 2 & 3 \end{bmatrix}$.	Find a combination of the vectors $\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$ that give the zero vector.	0.90
8	MATH Pre-Algebra	How many four-digit positive integers are there with hundreds digit 2?	How many four-digit positive integers are there with thousands digit 2?	0.90
9	MATH Algebra	Find the distance between the points $(0, 0)$ and $(3, 4)$.	Find the distance between the points $(0, 4)$ and $(3, 0)$.	0.99
10	MATH Number Theory	Find the smallest positive integer n such that n^2 is divisible by 2^{10} and n^3 is divisible by 3^{10}.	How many four-digit numbers whose digits add up to 9 are divisible by 11?	0.25
11	MATH Counting and Probability	How many ways are there to divide a set of 10 objects into two sets of equal size?	Compute $\binom{8}{4}$.	0.12
12	MATH Intermediate Algebra	Let x and y be positive real numbers such that $x^2 + y^2 = 1$. Find the maximum value of xy.	Given that $x^2 + y^2 = 14x + 6y + 6$, find the largest possible value of $3x + 4y$.	0.59
13	MATH Precalculus	Let A be the matrix $\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$ Find the determinant of $A^2 + A^3$.	If $\det(A) = 2$ and $\det(B) = 12$, then find $\det(AB)$.	0.41

Generating New Questions

```
"""
1. If v+w = (5,1) and v-w=(1,5), compute and draw the vectors v and w
2. Four of the eight corners of a unit cube are (0,0,0), (1,0,0), (0,1,0), (0,0,1). Find the coordinates of the center point of the cube.
3. If v = (1,2) draw all vectors w = (x,y) in the xy plane with dot(v,w) = x+ 2*y = 5. Why do those w's lie along a line? Which is the shortest w?
4. Find the linear combination 3s1 + 4s2 + 5s3 = b. Then write b as a matrix-vector multiplication Sx, with 3, 4, 5 in x. Compute the three dot products (row of S)· x: s_1 = [1;1;1], s_2 = [0;1;1], s_3 = [0;0;1]
5. Find a combination x_1*w_1+x_2*w_2+x_3*w_3 that gives the zero vector with x_1 = 1. w_1 is the vector (1;2;3), w_2 is the vector (4;5;6), w_3 is the vector (7;8;9)
6. What 2 by 2 matrix R rotates every vector through 45 degrees? Example: the vector [1,0] goes to [sqrt(2)/2, sqrt(2)/2].
7. The paraboloa y=a+bx+cx^2 goes through the points (x,y) = (1,4) and (2,8) and (3,14). Find and solve a matrix equation for the unknowns (a,b,c)
8. Find a matrix that has A^2 does not equal 0 but A^3 = 0
9. Find the inverses (directly or from the 2 by 2 formula) of A, B, C : A = [0,4;3,0], B = [2,0;4,2], and C = [3,4;5,7].
10. Solve for the first column (x, y) and second column (t, z) of A^-1: [10,20;20,50]@[x;y] = [1;0] and [10,20;20,50]@[t;z] = [0;1].
11. Find singular matrices A and B such that A+B is invertible.
12. Construct a 2 by 2 matrix whose nullspace equals its column space. This is possible.
13. Show that v1, v2, v3 are independent but v1,v2,v3,v4 are dependent. v1 = [1;0;0], v2 = [1;1;0], v3 = [1;1;1], v4 = [2;3;4]
14. Find A'A if the columns of A are unit vectors, all mutually perpendicular.
15. Draw the projection of b onto a: b=[1;1] and a=[1;-1].
16. What linear combination of (1,2,-1) and (1,0,1) is closest to b=(2,1,1)?
17. Find the best line C+Dt to fit b=4,3,-1,0,0 at times t=-2,-1,0,1,2.
18. If a 4 by 4 matrix has det(A)=1/2, find det(2A), det(-1A), det(A^2) and det(A^-1).
19. Find the area of the parallelogram with edges v=(3,2) and w=(1,4).
20. Find the eigenvalues of A and B (easy for triangular matrices) and A + B: A = [3,0;1,1], B = [1,1;0,3], A+B = [4,1;1,4].
21. Find the eigenvalues and eigenvectors for both of these Markov matrices A and A∞. A = [.6, .2; .4, .8]. A∞ = [1/3, 1/3; 2/3, 2/3].
22. Find a symmetric matrix [1,b;b,1] that has a negative eigenvalue.
23. For which numbers b is the following matrices positive definite? S=[1,b;b,9], S=[2,4;4,c], S=[c,b;b,c].
24. Find the eigenvalues and the singlular values of this 2 by 2 matrix A=[2,1;4,2]. The eigenvectors (1,2) and (1,-2) of A are not orthogonal.
25. Suppose A0 holds these 2 measurements of 5 samples: A0 = [5,4,3,2,1;-1,1,0,1,-1]. Find the average of each row and subtract it to produce the centered matrix A.
Compute the sample covariance matrix S=AA'/(n-1) and find its eigenvalues. What line through the origin is closest to the 5 samples in the columns of A?
26. Suppose a linear T transforms (1,1) to (2,2) and (2,0) to (0,0). Find T(v), when v=[2,2], v=(3,1), v=(-1,1), v=(a,b).
27. We know: 1/3 of all integers are divisible by 3 and 1/7 of integers are divisible by 7. What fraction of integers will be divisible by 3 or 7 or both ?
28. Sample again from 1 to 1000 but look at the last digit of the sample squared. That square could end with x = 0, 1, 4, 5, 6, or 9. What are the probabilities p0, p1, p4, p5, p6, p9?
29. If all 24 samples from a population produce the same age x = 20, what are the sample mean μ and the sample variance S^2?
30. Find the average A_N of a million random 0-1 samples! What is X = (A_N - 1/2)/(2*sqrt(N))?
31.
```

Generating Explanations

```
"""
Compute the 7th element in the fibonacci series
"""


def fibonacci(n):
    if n == 0:
        return 0
    elif n == 1:
        return 1
    else:
        return fibonacci(n-1) + fibonacci(n-2)

print(fibonacci(7))

"""
Starting at the beginning of the fibonacci function, please provide a line by line explanation of the above.
"""


"""
The fibonacci function takes in a number n and returns the nth element in the fibonacci series.

The first two lines of the function check to see if the number is 0 or 1. If it is either of these, the function returns 0 or 1.

If the number is neither of these, the function calls itself and adds the previous two numbers in the series to get the next number.

"""
```

Student Survey

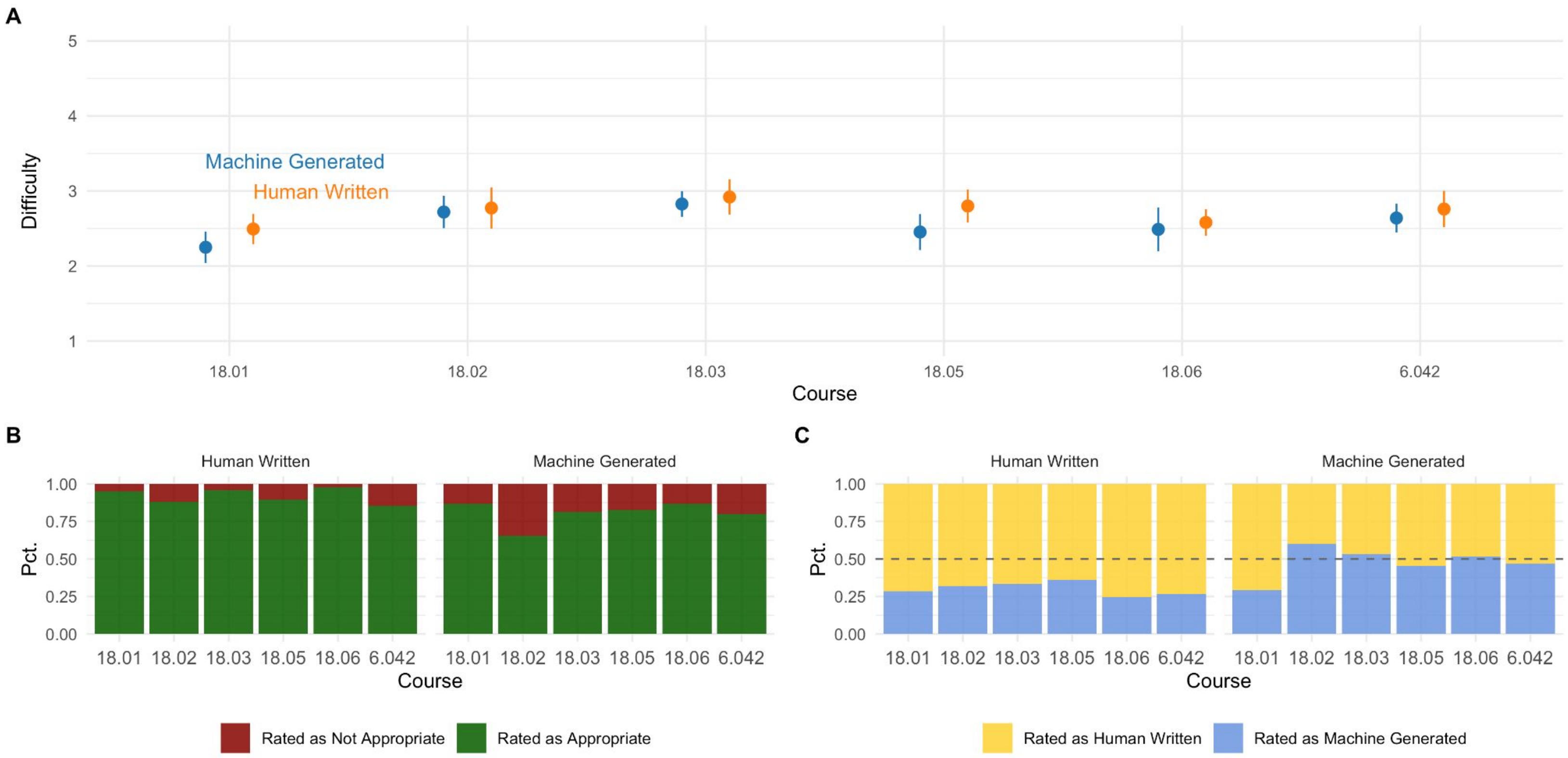
Course Name:

Introduction to Probability and Statistics

For each of the following questions, indicate whether you think the question is Human Written or Machine Generated, whether it is appropriate for the above course, as well as its level of difficulty between 1-5 where 1 is easiest, 2 is below average, 3 is average for the course, 4 is above average, and 5 in hardest.

	Human vs. Machine Generated		Appropriate for Course		Difficulty (1=Easiest;5=Hardest)				
	Human Written	Machine Generated	Appropriate	Not Appropriate	1	2	3	4	5
Boxes of Raisin Bran cereal are 30cm tall. Due to settling, boxes have a higher density of raisins at the bottom (h = 0) than at the top (h = 30). Suppose the density (in raisins per cm of height) is given by f(h) = 40 – h. What is the probability that a random raisin is in the top third of the box?	○	○	○	○	○	○	○	○	○

Student Survey Results



Pipeline Examples

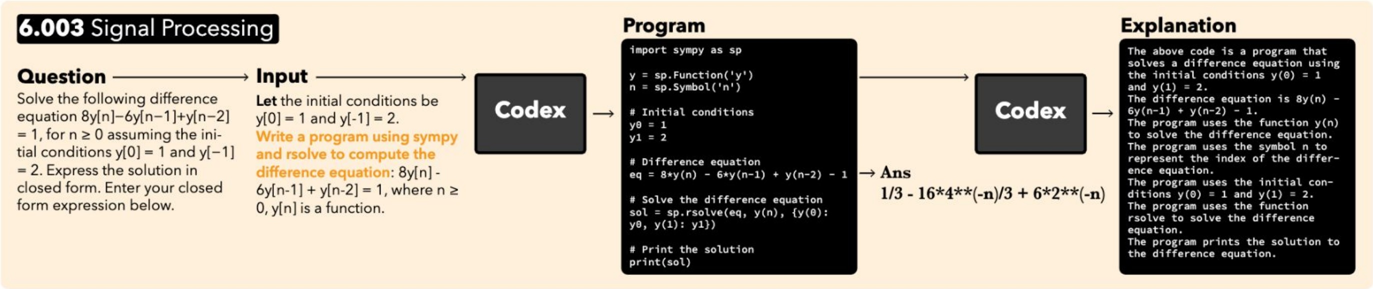


Figure 2: MIT 6.003 Signal Processing workflow: The question is solved as is and the prompt adds programming context to use the symbolic math sympy package to produce code snippets that generate answers in the form of a symbolic mathematical equation.

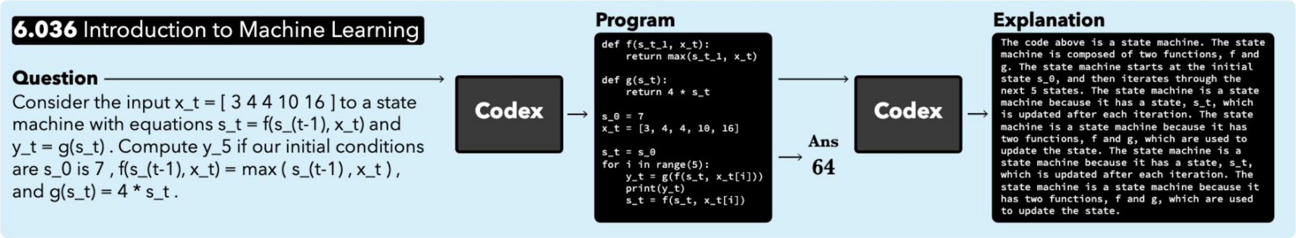
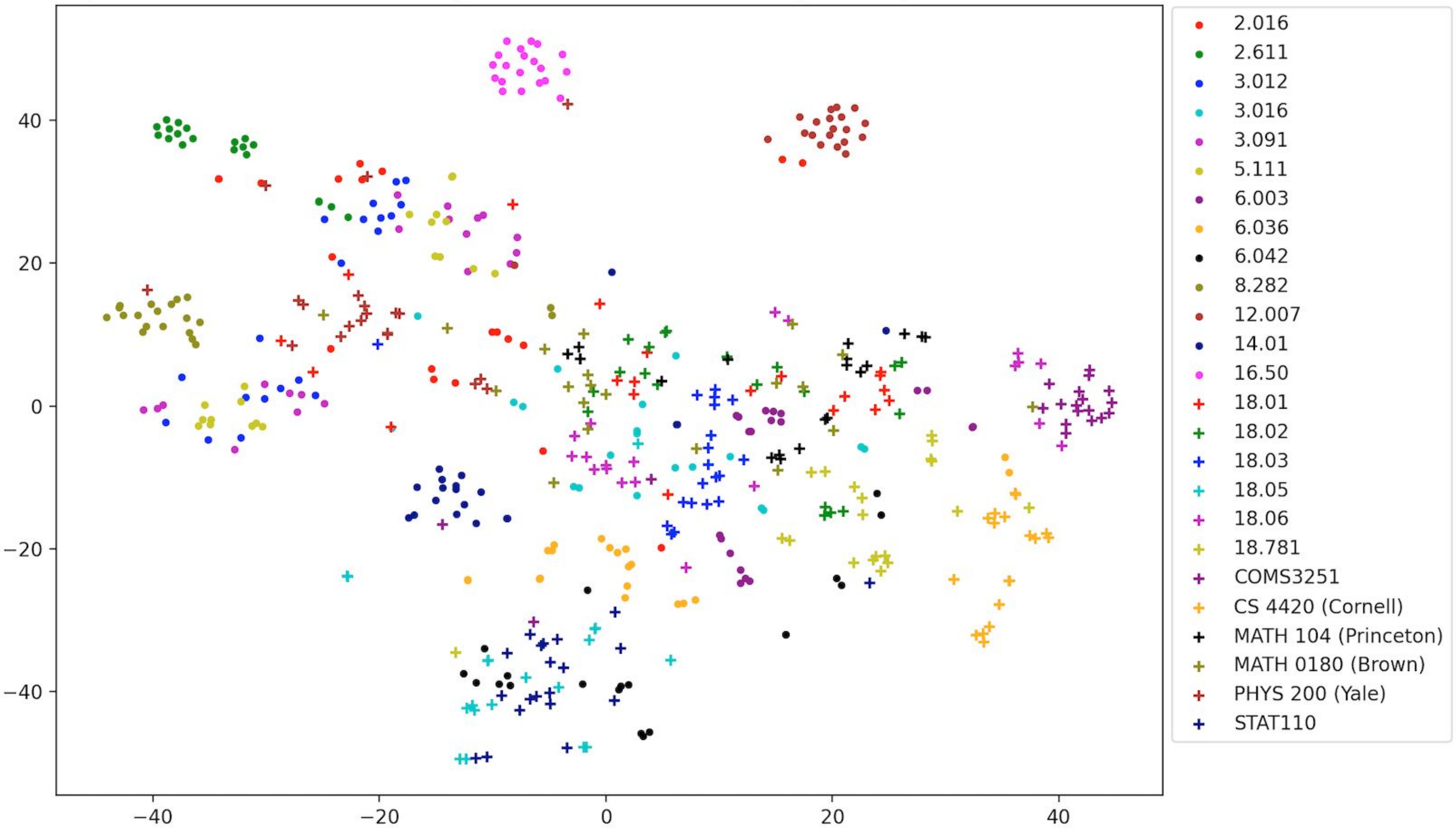


Figure 3: MIT 6.036 Introduction to Machine Learning workflow: The question is solved as is and is able to iteratively plug-in values in a recurrent neural network.

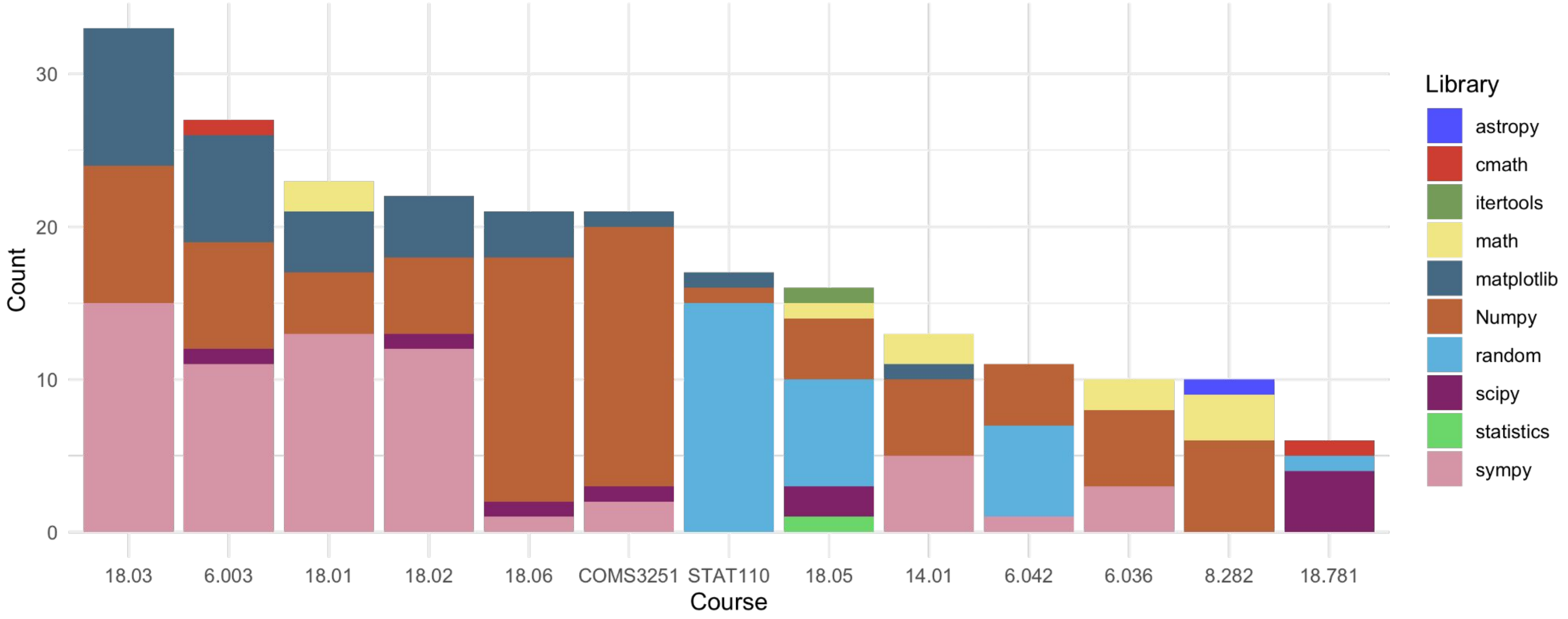
Current Courses

ID	University	Department	Course	Number
1	MIT	Aeronautics and Astronautics	Unified Engineering 1-4	16.01-4
2	MIT	Aeronautics and Astronautics	Estimation and Control of Aerospace Systems	16.30
3	MIT	Aeronautics and Astronautics	Intro to Propulsion Systems	16.50
4	MIT	Materials Science and Engineering	Fundamentals of Materials Science	3.012
5	MIT	Materials Science and Engineering	Mathematics for Materials Scientists and Engineers	3.016
6	MIT	Materials Science and Engineering	Introduction to Solid-State Chemistry	3.091
7	MIT	Chemical Engineering	Chemical and Biological Reaction Engineering	10.37
8	MIT	Chemistry	Principles of Chemical Science	5.111
9	MIT	IDSS	Statistical Thinking & Data Analysis	IDS.013(J)
10	MIT	Electrical Engineering & Computer Science	Signal Processing	6.003
11	MIT	Electrical Engineering & Computer Science	Introduction to Machine Learning	6.036
12	MIT	Electrical Engineering & Computer Science	Mathematics for Computer Science	6.042
13	MIT	Physics	Introduction to Astronomy	8.282
14	MIT	Nuclear Science and Engineering	Introduction to Nuclear Engineering and Ionizing Radiation	22.01
15	MIT	Economics	Principles of Microeconomics	14.01
16	MIT	Mechanical Engineering	Hydrodynamics	2.016
17	MIT	Mechanical Engineering	Nonlinear Dynamics I: Chaos	2.050J
18	MIT	Mechanical Engineering	Information & Entropy	2.110J
19	MIT	Mechanical Engineering	Marine Power and Propulsion	2.611
20	MIT	Mathematics	Single Variable Calculus	18.01
21	MIT	Mathematics	Multi-variable Calculus	18.02
22	MIT	Mathematics	Differential Equations	18.03
23	MIT	Mathematics	Introduction to Probability and Statistics	18.05
24	MIT	Mathematics	Linear Algebra	18.06
25	MIT	Mathematics	Theory of Numbers	18.781
26	Harvard	Statistics	Probability	STATS110
27	Princeton	Mathematics	Calculus II	MAT104
28	Cornell	Computer Science	Computer Architecture	CS4420
29	Columbia	Computer Science	Computational Linear Algebra	COMS3251
30	Yale	Mathematics	Fundamentals of Physics	PHYS200
31	UPenn	Mathematics	Calculus	MATH110
32	Brown	Mathematics	Intermediate Calculus	MATH0180

Visualization of Course Questions



Programming Libraries Usage

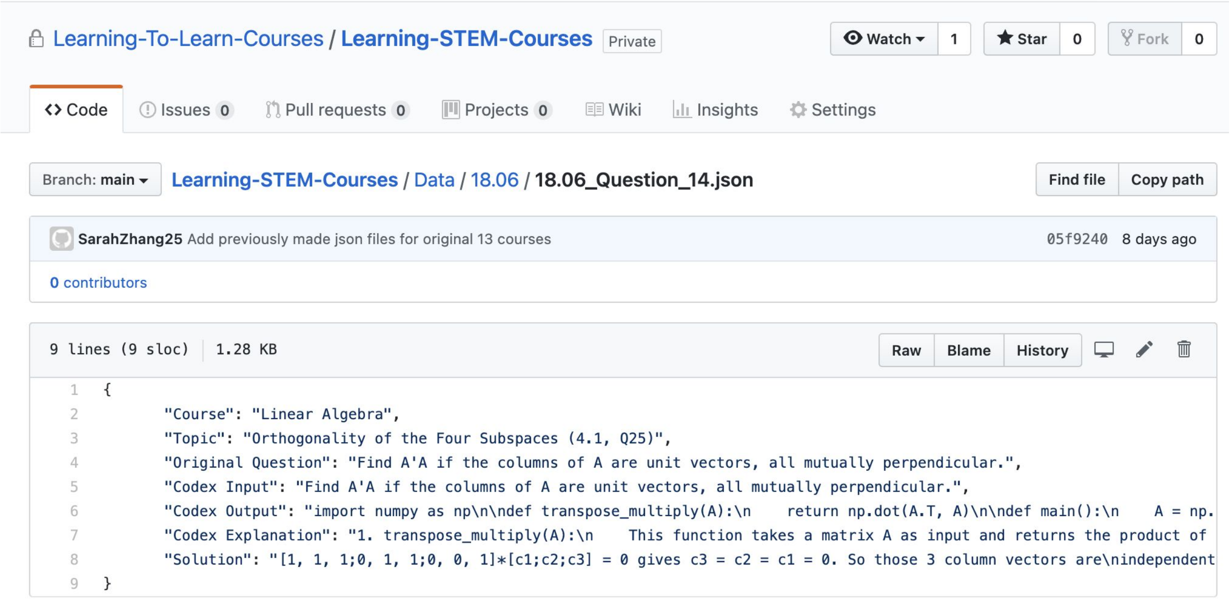


Limitations

- Input images
- Multi-page proofs
- Theoretical complexity

Data and Code

- Private MIT repo
- Public release upon publication



Conclusions

- Automatically solve, explain, and generate university-level courses questions
- Students rated machine-generated questions as equally likely to be human-written as machine-generated
- Scaling up to hundreds of courses
- Reproducibility is 100% by examples; Automation will not be 100% since we may always come up with a harder question.
- Impact on higher education

Benefits for Higher Education

- Advance residential learning
- Enable self-paced learning: mapping points of confusion, reinforcing learning, a cognitive assistant for students
- Serve as a tool for curriculum design and evaluation
- Allow automatic content generation at scale

Learning to Learn Courses Team

Fall 2020

Iddo Drori, MIT
Sunny Tran, MIT
Alexander Gu, MIT

Spring 2021

Iddo Drori, MIT
Sunny Tran, MIT
Prabhakar Kafle, MIT
Pranav Krishna, MIT
Ishan Pakuwal, MIT
Nikhil Singh, MIT

Fall 2021

Iddo Drori, MIT
Elizabeth Ke, MIT
Kevin Liu, MIT
Pranav Krishna, MIT
Sunny Tran, MIT
Nikhil Singh, MIT
Avi Shporer, MIT
Gilbert Strang, MIT
Leonard Tang, Harvard
Taylor Patti, Harvard
Newman Cheng, Columbia University
Roman Wang, Columbia University
Nakul Verma, Columbia University
Eugene Wu, Columbia University
Jayson Lynch, University of Waterloo

Spring 2022

Iddo Drori, MIT and Columbia University
Michelle He, MIT
Linda Chen, MIT
Sarah Zhang, MIT
Reece Shuttleworth, MIT
Muhender Rajvee, MIT
Albert Lu, MIT
Nikhil Singh, MIT
Avi Shporer, MIT
Tonio Buonassisi, MIT
Gilbert Strang, MIT
Leonard Tang, Harvard
Taylor Patti, Harvard and NVidia
Yann Hicke, Cornell University
Ethan Prihar, WPI
Bo Feng, Columbia University
Newman Cheng, Columbia University
Nakul Verma, Columbia University
Eugene Wu, Columbia University

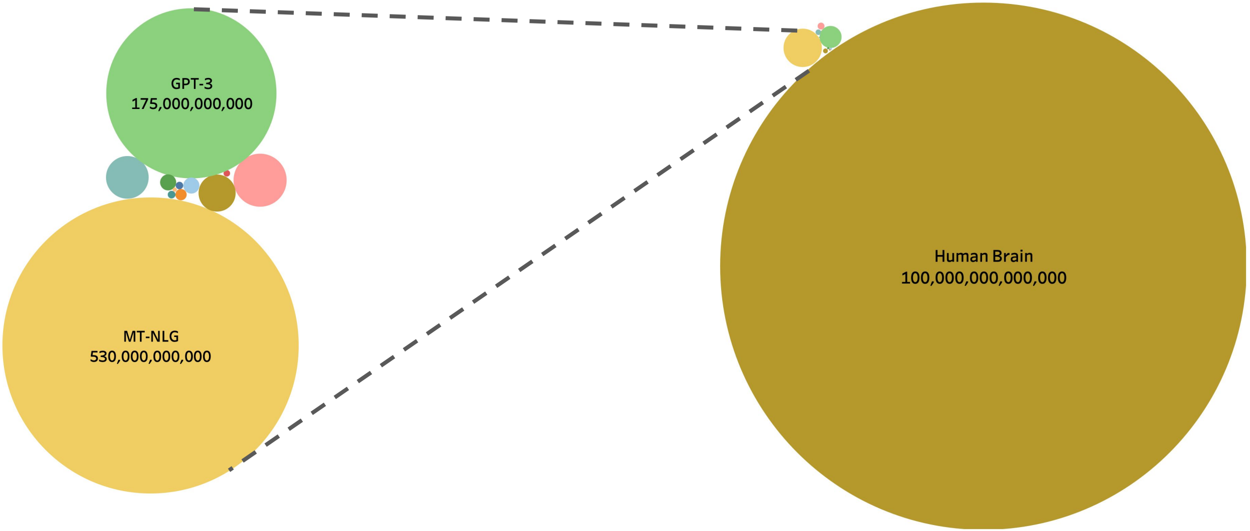
Learning to Learn Courses

Iddo Drori

Massachusetts Institute of Technology

Columbia University in the City of New York

Transformer Parameters vs. Brain Connections



- Transformers currently have only 3 orders of magnitude less parameters than the number of connections in the human brain.
- Number of parameters is growing by x10 each year, expected to surpass the number of human brain connections in three years.

Future Work

- Scaling to hundreds of courses.
- Proofs.
- Codex writing its own code.
- Training populations of Transformers that compete and collaborate.

www.dlbook.org

Contact idrori@mit.edu or idrori@cs.columbia for the entire book (if you would like to use the book for teaching)