Open in app ↗

New: Navigate Medium from the top of the page, and focus more on reading as you scroll.

Okay, got it

lium

Leo Irakliotis

Dec 6, 2022 · 6 min read · Listen

Save

College teaching and AI

Artificial Intelligence will reshape the way we assess student learning in ways that no one has prepared us for.

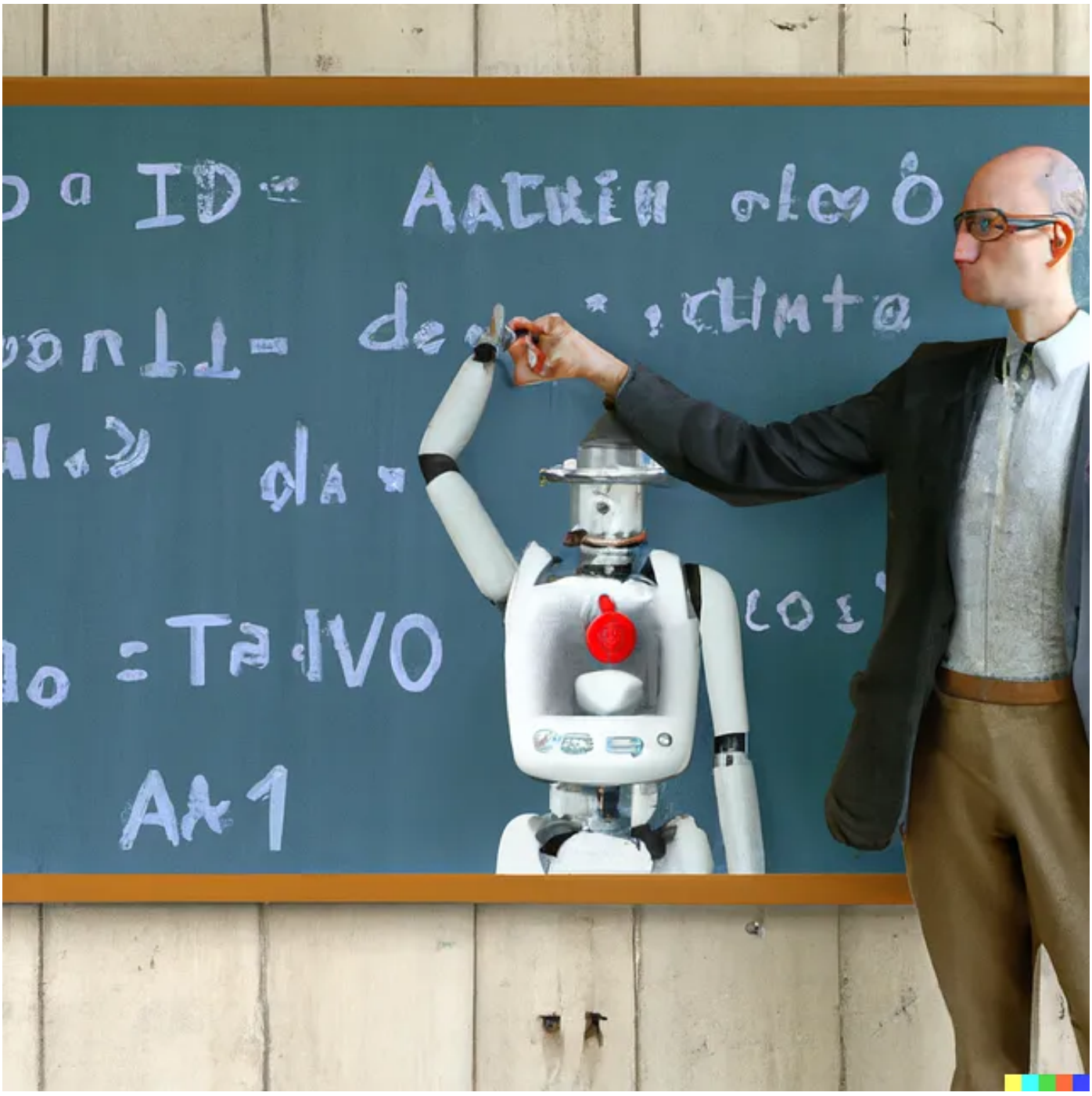

Image produced with OpenAI's Dall-e with the specification: "A postmodernist professor in front of an old fashion blackboard, teaching AI robots about neural networks"

The first thing that I tried with OpenAI's ChatGPT was the questions I was preparing for a final exam in data structures. It answered them accurately — though, not entirely correct. ChatGPT is *self*-described as

> *a language model trained by OpenAI. [It is] not a real person and [does] not have a physical form. [It is] a computer program that is designed to generate human-like responses to text input. [Its] purpose is to assist with a variety of tasks, including answering questions,*

> *providing information, and engaging in conversation. [It is] not capable of browsing the internet, so [it is] unable to provide information that is not part of [its] pre-existing knowledge base.*
> — ChatGPG response to “What are you?”

For years now, I’ve been giving students in my programming courses take-home exams. I stopped worrying about plagiarising a long time ago. I expect students to consult the vastness of the web and in fact, I encourage them to do so. At the same time, I expect them to *internalize* any code they find online and use. They must be able to describe to me how it works and why is it acceptable to them.

ChatGPT takes the game to a different level. Not only it provides an accurate answer, but it also provides a narrative that reflects an understanding of the solution. Students could use this narrative as proof of internalizing the code the AI found for them.

Here’s an example: One of the easier questions in the exam asks students to write a (Java) method that returns duplicate elements in an integer array. ChatGPT gave me the following code, comments, and all:

```
public static List<Integer> findDuplicates(int[] arr) {
  // Create a set to store the elements that have been seen
  Set<Integer> seen = new HashSet<>();
  // Create a list to store the duplicate elements
  List<Integer> duplicates = new ArrayList<>();
  // Loop through the array and check each element
  for (int i = 0; i < arr.length; i++) {
    int num = arr[i];
    // If the element has been seen before, it is a duplicate
    if (seen.contains(num)) {
      duplicates.add(num);
    } else {
      // If the element has not been seen before, add it to the set
      seen.add(num);
    }
  }
  // Return the list of duplicate elements
  return duplicates;
}
```

It also provided a narrative for the code — I'll omit it for brevity, but it's good. The comments in the code are also good. They read as pseudocode, demonstrating a good understanding of how the code works.

The code above was in response to the inquiry for a Java method that **returns** duplicate elements in an integer array. Had I asked the AI for a method to **find** duplicate elements, it would have given me a void method for printing the duplicate elements. I am impressed by the AI's ability to discern requirements.

The AI begins to falter when the requirement becomes a bit more specific: *write a Java method that returns an array with the positions of duplicate elements in an integer array:*

```
public static int[] findDuplicates(int[] arr) {
    Set<Integer> duplicates = new HashSet<>();
    for (int i = 0; i < arr.length; i++)
        for (int j = i + 1; j < arr.length; j++)
            if (arr[i] == arr[j])
                duplicates.add(j);
    return duplicates.stream().mapToInt(i -> i).toArray();
}  // Comments and single-line bracing removed for brevity by Leo
```

The method is accurate. It is also incorrect in the context of a first course in data structures. In my course, we steer clear of the `Stream` interface. A return statement like the one above is quite unexpected. Students may have a difficult time demonstrating their understanding of `stream` and `mapToInt` at this stage. But if they do, more power to them.

Finally, when I narrow down the specification to: *write a Java method that returns an integer array showing the index positions of duplicate elements in an integer array without using a `Set` or a `List` object and without using a `break` statement;* the AI kind of gives up — for now:

```
  public static int[] findDuplicates(int[] arr) {
    int numDuplicates = 0;
    for (int i = 0; i < arr.length; i++)
```

```
      for (int j = i + 1; j < arr.length; j++)
        if (arr[i] == arr[j])
          numDuplicates++;
          break;
    int[] duplicates = new int[numDuplicates];
    int index = 0;
    for (int i = 0; i < arr.length; i++) {
      for (int j = i + 1; j < arr.length; j++) {
        if (arr[i] == arr[j] && !contains(duplicates, i)) {
          duplicates[index] = i;
          index++;
    return duplicates;
  }  // Single-line bracing removed for brevity by Leo
```

Gotcha! I'll ask my students to *write a Java method that returns an integer array showing the index positions of duplicate elements in an integer array, or -1 if the element is not a duplicate, without using a* `Set` *or a* `List` *object and without using a* `break` *statement.* Not so fast says the AI:

```
public static int[] findDuplicates(int[] arr) {
  int[] duplicates = new int[arr.length];
  for (int i = 0; i < arr.length; i++) {
    duplicates[i] = -1;
  for (int i = 0; i < arr.length; i++) {
    for (int j = i + 1; j < arr.length; j++) {
      if (arr[i] == arr[j] && duplicates[i] == -1) {
        duplicates[i] = i;
  return duplicates;
}  // Single-line bracing removed for brevity by Leo
```

And then it has the cheek to tell me that

> *this method is less efficient than the previous ones because it has to iterate over the input array twice: once to initialize the array of duplicates, and once to find the duplicate elements. This can be avoided by using a* `Set` *or* `List` *object, as in the previous examples.*

The AI kept up with me, as the exam questions became more challenging. I tried to evade it by obfuscating the requirements e.g., return a string with the contents of a

linked list while traversing it forward and without using StringBuilder. Fine, said the AI, here's your method but you should know that it has:

> *$O(n^2)$ time complexity, where n is the number of elements in the linked list, because concatenating strings in Java has O(n) time complexity. For larger linked lists, a more efficient implementation that uses a StringBuilder or a similar class would be preferred.*

When I asked it to convert Chicago's southbound Red Line into a linked list, it told me that

> *It is not possible to convert the Chicago Transit Authority's (CTA) southbound Red Line train into a Java linked list, as a linked list is a data structure used in computer programming to store a linear collection of data, while the CTA Red Line is a physical train line that provides public transportation in Chicago.*

Adding insult to the injury, it proceeded by suggesting how to *represent* the stops on the southbound Red Line as a linked list.

```
LinkedList<String> redLineSouthbound = new LinkedList<>();

redLineSouthbound.add("Howard");
redLineSouthbound.add("Jarvis");
redLineSouthbound.add("Morse");
...
```

At this point, I am thinking that we, educators, are in trouble. AI has the potential to upend every teaching paradigm we have been relying on for years. Our knee-jerk reaction will be probably to forbid students from using AI in their assignments. This will not work for two reasons: first, it's very difficult to enforce. And second, what's the purpose of forbidding the use of a tool that students will be probably required to use in their careers?

A better strategy will be to integrate AI into teaching and assessing learning. This will require more frequent and more structured oral examinations, in small groups or 1–1 settings. In programming classes, it may require a more significant shift from coding to designing; with more emphasis on the structure and interpretation of computer

programs (hint, hint, wink, wink). Possibly abolishing the notion of homework and localizing the learning activity in the classroom. It will definitely require a conversation with our CTOs about licensing AI products across campus. Some retraining for ourselves won't go amiss, either. And the sooner we start preparing for an AI-infused higher education experience, the better off our students will be.

Higher Ed AI

Sign up for CrunchX

By CodeX

A weekly newsletter on what's going on around the tech and programming space Take a look.

Emails will be sent to lgreco@gmail.com. Not you?

Get this newsletter