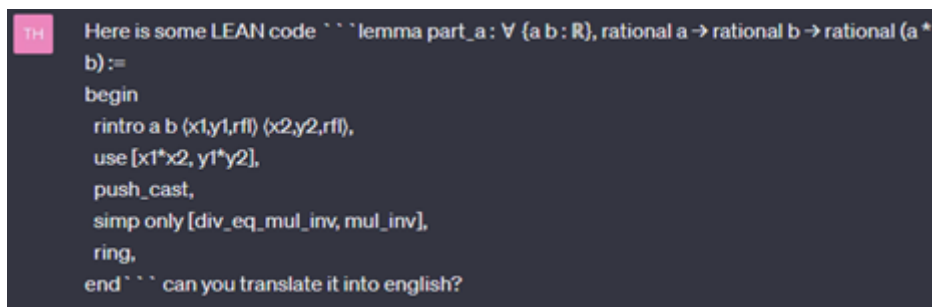


Motivation

The advent of ChatGPT has facilitated a wealth of technologies driven by language. With its vast knowledge base, it has been used as an assistant, a customer support agent, and a teacher. This last application has been pioneered by Khan academy, who have utilized GPT4 to allow its user access to a virtual tutor who can guide them through concepts and give them hints to answer questions. Learning about this application of large language models I thought it may be possible to capitalize on the formalism of LEAN and the abilities of ChatGPT to create an environment where students could explore higher level mathematics in an interactive fashion. I have since conducted some investigations which have motivated some potential next steps of the project.

Translation

Translating LEAN Code into Proofs



```
TH Here is some LEAN code ``` lemma part_a : ∀ (a b : ℝ), rational a → rational b → rational (a * b) := begin   rintro a b (⟨x1,y1,rfl⟩) (⟨x2,y2,rfl⟩),   use [x1*x2, y1*y2],   push_cast,   simp only [div_eq_mul_inv, mul_inv],   ring, end``` can you translate it into english?
```

Firstly, I took a piece of LEAN code and ask ChatGPT to translate it. Note that I did not include any of the libraries or definitions made before the code block. Despite this it seems that ChatGPT was able to understand the code, the definition of `rational`, and output an explanation of what the code was trying to achieve.



The given code is written in Lean, which is a proof assistant programming language. It defines a lemma named "part_a" and proves a specific statement about rational numbers. Here's an English translation of the code:

The lemma states that for any real numbers a and b , if a is rational and b is rational, then their product ($a * b$) is also rational.

To prove this, the proof begins by introducing variables a and b and assuming that a is a rational number. The notation " $\langle x1, y1, rfl \rangle$ " indicates that a can be represented as the fraction $x1 / y1$, where $x1$ and $y1$ are integers. Similarly, the assumption that b is rational is represented by " $\langle x2, y2, rfl \rangle$ " where $b = x2 / y2$.

Next, the proof uses the "use" keyword to specify the values of $x1x2$ and $y1y2$ as the numerator and denominator of the rational number $a * b$, respectively. This step essentially shows that the product of two rational numbers can be expressed as a fraction.

The "push_cast" tactic is then applied to simplify the goal by converting various operations and functions involving integers and rationals into their corresponding forms.

The proof continues by using the "simp only" tactic to further simplify the goal. It applies specific simplification rules related to division (div_eq_mul_inv) and multiplicative inverse (mul_inv).

Finally, the "ring" tactic is used to complete the proof. The "ring" tactic is a powerful tool in Lean that automatically solves equations involving addition, subtraction, multiplication, and their identities.

In summary, the given code proves that if two real numbers a and b are rational, then their product ($a * b$) is also a rational number.

From this I tried to prompt ChatGPT to format its answer in a more conventional way, such that it is easier to read and resembles other statements made in mathematical textbooks. To do this I asked it to remove any references to the LEAN syntax and structure it so that the statement of the lemma is followed by its proof. There are still some structure and formatting issues which are to be resolved.

TH

Great, can you rewrite this with the statement of the lemma followed by its proof? Do not include any references to LEAN tactics in your answer.



Certainly! Here's the rewritten version of the lemma statement and its proof without any references to Lean tactics:

Lemma: For any real numbers a and b , if a is a rational number and b is a rational number, then their product ($a * b$) is also a rational number.

Proof: Let a and b be real numbers. Assume that a is a rational number, which means it can be expressed as the fraction x_1 / y_1 , where x_1 and y_1 are integers. Similarly, assume that b is a rational number, represented by the fraction x_2 / y_2 , where x_2 and y_2 are integers.

To prove that the product of two rational numbers is also rational, we can express their product ($a * b$) as a fraction. Let's define the numerator and denominator of this fraction as x_1x_2 and y_1y_2 , respectively.

Now, we need to show that the numerator x_1x_2 and denominator y_1y_2 are integers. Since x_1 and x_2 are integers (as x_1/y_1 and x_2/y_2 represent rational numbers), their product x_1x_2 is also an integer. Similarly, y_1 and y_2 being integers implies that y_1y_2 is an integer.

Therefore, we have expressed the product of the rational numbers a and b as the fraction $(x_1x_2) / (y_1y_2)$, where the numerator and denominator are both integers. This demonstrates that the product ($a * b$) is rational.

Hence, we have proved that for any real numbers a and b , if a is a rational number and b is a rational number, then their product ($a * b$) is also a rational number.

Next, I wanted to see if ChatGPT could perform a similar role to that of a graduate teaching assistant for an undergraduate. Despite being a simple example, ChatGPT was able to give a hint on how to prove this lemma and provide it in the desired context. Therefore, it may be possible to use ChatGPT to translate the proofs written in LEAN to a more familiar format. Then the student could query certain components of the output to increase their understanding. For example, they could ask ChatGPT to elaborate on a particular component, or they could ask ChatGPT to give a partial proof and provide hints on how to proceed.

TH

I am a student trying to prove this lemma. Could you give me a hint on how to start the proof?



Certainly! When starting the proof, it's often helpful to consider the definition of rational numbers and how they can be expressed as fractions. Here's a hint to get you started:

Hint: Recall that a rational number is a number that can be expressed as the ratio of two integers. To prove that the product of two rational numbers is also rational, you can begin by assuming that a and b are rational numbers and express them as fractions. Then, consider how you can manipulate the fractions and perform the multiplication to show that the product is also a ratio of two integers.

Try starting with the assumption that a is a rational number and can be expressed as a fraction x_1 / y_1 , where x_1 and y_1 are integers. Similarly, assume that b is a rational number represented by the fraction x_2 / y_2 , where x_2 and y_2 are integers. From there, think about how you can manipulate these fractions to express the product $(a * b)$ as a fraction and show that its numerator and denominator are both integers.

Remember to make use of properties of multiplication and the fact that integers are closed under multiplication. Good luck with your proof!

Translating Proofs into LEAN Code

Conversely, I wanted to see if ChatGPT could go the other way. When presented with a proof written by a student could ChatGPT translate it into LEAN code? I am not too familiar with LEAN so cannot comment on whether the above code would work in proving the lemma. Comparing it to the original code it is substantially longer. Perhaps with more refined prompting, such as stating the libraries it should use, it may be possible to use ChatGPT in this direction. Note that ChatGPT does provide comments on its code so it could be used as a guide to students on how to translate their proofs into LEAN code.



Certainly! Here's a translation of the proof into Lean code:

lean

Copy code

```
lemma part_a : ∀ {a b : ℝ}, rational a → rational b → rational (a * b) :=
begin
  intros a b ha hb,
  rcases ha with ⟨x1, y1, ha⟩,
  rcases hb with ⟨x2, y2, hb⟩,
  use [x1 * x2, y1 * y2],
  split,
  { show (a * b) * (y1 * y2) = (x1 * x2),
    rw [ha, hb],
    ring },
  { intro h,
    apply_fun (λ (v : ℝ), v * (y1 * y2)) at h,
    rw [ha, hb] at h,
    rw mul_assoc at h,
    rw mul_comm y1 at h,
    rw ←mul_assoc at h,
    rw mul_assoc at h,
    rw mul_comm y2 at h,
    rw ←mul_assoc at h,
    rw mul_assoc at h,
    have key : x1 * y1 * (x2 * y2) = x1 * x2 * (y1 * y2), by ring,
    rw key at h,
    exact h }
end
```

In this translation, the proof starts with the `intros` tactic to introduce the variables and assumptions. The assumptions `ha` and `hb` are then matched with the form `⟨x1, y1, ha⟩` and `⟨x2, y2, hb⟩` using the `rcases` tactic to extract the numerator and denominator of `a` and `b`.

The `use` tactic is used to specify the values of `x1 * x2` and `y1 * y2` as the numerator and denominator of the rational number `a * b`. The proof then proceeds to split into two subgoals using the `split` tactic.

In the first subgoal, we need to show that `(a * b) * (y1 * y2) = (x1 * x2)`. This is achieved by rewriting `ha` and `hb` using the `rw` tactic and applying the `ring` tactic to simplify the equation.

In the second subgoal, we assume the equation `(a * b) * (y1 * y2) = (x1 * x2)` and aim to prove that `x1 * y1 * (x2 * y2) = x1 * x2 * (y1 * y2)`. This is done by applying a series of rewriting steps using the `rw` tactic and the `ring` tactic to manipulate the equation.

Finally, the proof is concluded using the `end` keyword.


Note that the code assumes that you have defined the `rational` type and relevant operations in Lean or imported the necessary libraries.

Understanding the Output

Previously, we saw that ChatGPT was able to infer what the type `rational` was without being told its definition. In the following prompts we see that it inferred what the type was by drawing similarities to the type `rat`. On the one hand, this shows that ChatGPT can make reasonable inferences about the code it is presented. Which would make translating proofs from LEAN easier as we wouldn't have to worry about providing all the necessary definitions. On the other hand, if the inferences it makes are wrong this could lead to proofs which are false or rely on facts that have yet to be proven. Note also how ChatGPT is easily manipulated to thinking it made a mistake, work would have to be done here to ensure that the model doesn't generate these sorts of outputs when working with a student.

TH

How did you know what a ``rational`` type was when it was defined in the original LEAN code?



Apologies for the confusion. In the original LEAN code you provided, there was no explicit definition or reference to a ``rational`` type. My earlier response was based on the assumption that the ``rational`` type was defined elsewhere in your Lean environment or imported from relevant libraries.

However, if the ``rational`` type is not defined, you would need to define it before using it in the lemma and proof. In Lean, the ``rat`` type is commonly used to represent rational numbers. You can define the ``rational`` type and its properties in the following way:

lean

Copy code

```
import data.rat

lemma part_a : ∀ {a b : rat}, rat.rec_on a (λ x1 y1 _, ∀ b, rat.rec_on b (λ
  (λ _ h _ _ , false.elim (rat.not_zero_of_rat_eq (h rfl))))
  (λ _ _ _ _ , ∀ b, rat.rec_on b (λ x2 y2 _ h, (x1 * x2, y2, rfl))))
  b))
```

This code imports the necessary ``rat`` library and defines the ``part_a`` lemma using the ``rat.rec_on`` function, which performs recursive computations on rational numbers. The proof is left as an exercise to complete.

Please note that this code assumes the ``rat`` library is available in your Lean environment. If it's not imported by default, you may need to import it explicitly or modify the code accordingly.

TH

How did you know that ``rational`` in the original LEAN code referred to the ``rat`` type?



Apologies for the confusion caused by my previous response. Upon reviewing the original Lean code you provided, I realize that my assumption about ``rational`` referring to the ``rat`` type was incorrect.

In Lean, the ``rational`` type is usually used to represent rational numbers, and it is typically defined in the ``data.rat`` module. However, the code you provided does not directly correspond to the standard definition of rational numbers in Lean.

To accurately translate the original Lean code you provided into English, I would need more information about the specific context and any custom definitions or assumptions that may be in place. It appears that the original code defines a lemma named ``part_a`` and performs calculations on rational numbers. However, without further details or a more complete understanding of the surrounding code, it is difficult to provide an accurate translation.

If you can provide additional information about the context or any relevant definitions, I will do my best to assist you further.

Formatting

To make the translations we saw above more readable I asked ChatGPT to translate them into LaTeX. Executing this in the naïve way generates an output that renders into something that is no improvement on the standard output we saw above.

TH I have the following LEAN code `` lemma part_b : ¬ ∀ {a b : ℝ}, irrational a → irrational b → irrational (a * b) :=
begin
 push_neg,
 refine (real.sqrt 2, real.sqrt 2, irrational_sqrt_two, irrational_sqrt_two, _),
 rw irrational_iff_ne_rational,
 push_neg,
 use [2, 1],
 norm_num,
end `` Could you translate it into Latex?



Certainly! Here's the translated LEAN code in LaTeX:

```
\begin{lstlisting}[language=Lean]
\begin{lemma}
part_b : $\neg$ \forallall {a\ b : \mathbb{R}}, $ irrational $a \rightarrow$ irrational $b$
\rightarrow$ irrational $(a \cdot b)$ :=
begin
  push_neg,
  refine $\langle$ \sqrt{2}, \sqrt{2}, \text{irrational\_sqrt\_two}, \text{irrational\_sqrt\_two}, _
  \rangle$,
  rw irrational\_iff\_ne\_rational,
  push_neg,
  use $\{2, 1\}$,
  norm\_num,
end
\end{lemma}
\end{lstlisting}
```

Note that you may need to include the appropriate packages and preamble in your LaTeX document to ensure that the mathematical symbols and notation are rendered correctly.

[language=Lean]

Lemma 3. $part_b : \neg \forall a\ b : \mathbb{R}, \text{irrational } a \rightarrow \text{irrational } b \rightarrow \text{irrational } (a \cdot b) :=$
 $begin\ push_neg, refine(\sqrt{2}, \sqrt{2},$

If instead we provide ChatGPT with an initial example, then results significantly improve. Specifically, I first translated a different proof from LEAN into Latex and gave this example to ChatGPT. I then took the same LEAN code I asked it to translate originally and asked it to translate it again. Ignore the inconvenient format, which could be fixed through better prompting. We can render this Latex code to generate a plausible proof written by a student.



```

TH I have translated the follows LEAN code
lemma part_a : ∀ (a b : ℝ), rational a → rational b
→ rational (a * b) :=
begin
  rintro a b (x1,y1,rfl) (x2,y2,rfl),
  use [x1*x2, y1*y2],
  push_cast,
  simp only [div_eq_mul_inv, mul_inv],
  ring,
end
-- into LaTeX as
\begin{lemma}
  For all rational numbers  $a$  and  $b$ , their product is also a rational number.
\end{lemma}
\begin{proof}
  Let  $a = \frac{x_1}{y_1}$  and  $b = \frac{x_2}{y_2}$  for  $x_1, y_1, x_2, y_2 \in \mathbb{Z}$ . We
  can do this as  $a$  and  $b$  are rational numbers. It follows,  $ab = \frac{x_1 x_2}{y_1 y_2}$ 
  which implies that  $ab$  is also rational because  $x_1 x_2, y_1 y_2 \in \mathbb{Z}$ .
\end{proof}
-- Can you do the same for the LEAN code
lemma part_b : ¬ ∀ (a b : ℝ),
irrational a → irrational b → irrational (a * b) :=
begin
  push_neg,
  refine (real.sqrt 2, real.sqrt 2, irrational_sqrt_two, irrational_sqrt_two, _),
  rw irrational_iff_ne_rational,
  push_neg,
  use [2, 1],
  norm_num,
end

```

Lemma 1. *For all rational numbers a and b , their product is also a rational number.*

Proof. Let $a = \frac{x_1}{y_1}$ and $b = \frac{x_2}{y_2}$ for $x_1, y_1, x_2, y_2 \in \mathbb{Z}$. We can do this as a and b are rational numbers. It follows, $ab = \frac{x_1 x_2}{y_1 y_2}$ which implies that ab is also rational because $x_1 x_2, y_1 y_2 \in \mathbb{Z}$. \square

 Sure! Here's the translation of the second LEAN code into LaTeX:

```
\begin{lemma}
There exist irrational numbers  $a$  and  $b$  such that their product  $ab$  is rational.
\end{lemma}
\begin{proof}
We will prove this by contradiction. Assume that for all irrational numbers  $a$  and  $b$ , their
product  $ab$  is also irrational.



swift Copy code



```
We will provide a counterexample to this assumption. Let $a = \sqrt{2}$ and
Now, let's consider their product ab . We have $ab = \sqrt{2} \cdot \sqrt{2}$
Hence, we have found an example where a and b are irrational numbers, bu
```



\end{proof}



Please note that in the translation, I used the assumption that  $\sqrt{2}$  is irrational, as mentioned in the LEAN code.


```

Lemma 2. *There exist irrational numbers a and b such that their product ab is rational.*

Proof. We will prove this by contradiction. Assume that for all irrational numbers a and b , their product ab is also irrational. We will provide a counterexample to this assumption. Let $a = \sqrt{2}$ and $b = \sqrt{2}$. It is known that $\sqrt{2}$ is irrational, so we have a, b as irrational numbers.

Now, let's consider their product ab . We have $ab = \sqrt{2} \cdot \sqrt{2} = 2$. Since 2 is a rational number, this contradicts our assumption that the product of irrational numbers is always irrational.

Hence, we have found an example where a and b are irrational numbers, but their product ab is rational. Therefore, the statement that for all irrational numbers a and b , their product ab is irrational is false. \square

Next Steps

1. Refine ChatGPT to create readable proofs from LEAN code.
 1. Take some number of proofs written in LEAN and write them in Latex with a structure standard consistent with mathematical texts.
 2. Fine-tune the GPT model on these examples.
 3. Implement this refined model with appropriate prompting to translate other LEAN code.
2. Use these translated proofs to create an interactive environment where students can explore the proofs of theorems with ChatGPT playing the role of a GTA.
3. Look into ways to improve the ability of ChatGPT to translate proofs written in English into LEAN code.