Can ChatGPT Write Mathematical Proofs

Motivation

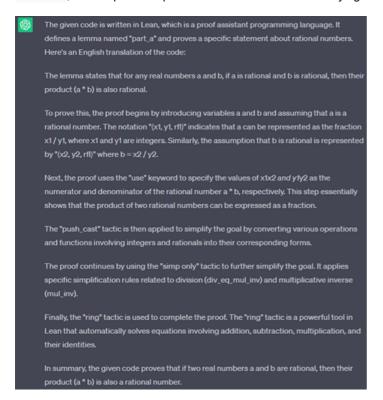
The advent of ChatGPT has facilitated a wealth of technologies driven by language. With its vast knowledge base, it has been used as an assistant, a customer support agent, and a teacher. This last application has been pioneered by Khan academy, who have utilized GPT4 to allow its user access to a virtual 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

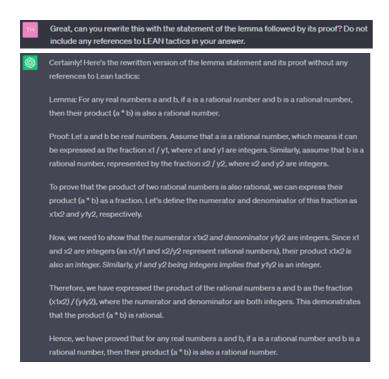
Translating LEAN Code into Proofs

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

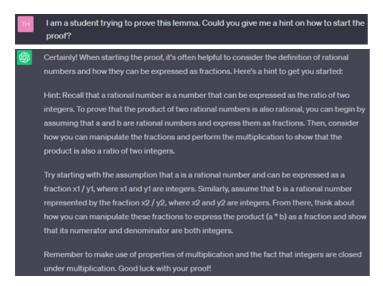
Firstly, I took a piece of LEAN code and ask ChatGPT to translate it. Note that I did not include any of the libraries or definitions made before the code block. Despite this it seems that ChatGPT was able to understand the code, the definition of rational, and output an explanation of what the code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



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.



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.



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 promoting, 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.

```
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 },
             apply_fun (\lambda (v : \mathbb{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_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,
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 'xw' tactic and applying the 'xing' 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
```

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.



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.

```
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{\text{lease} begin{\text{lease} begin{\text{lease} begin{\text{push} begin{\text{lease} begin{\text{push} begin{\text{lease} begin{\text{push} begin{\text{lease} begin{\text{push} begin} begin{\text{push
```

[language=Lean]

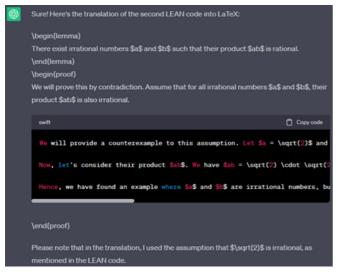
Lemma 3. $part_b : \neg \forall a \ b : \mathbb{R}$, $irrational \ a \rightarrow irrational \ b \rightarrow irrational \ (a \cdot b) := begin <math>push_neg, refine(\sqrt{2}, \sqrt{2},$

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.

```
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],
 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, $ab=\frac{x_1x_2}{y_1y_2}$
which implies that ab\ is also rational because x_1x_2,y_1y_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, _),
rwirrational_iff_ne_rational,
use [2, 1],
```

 $\textbf{Lemma 1.} \ \textit{For all rational numbers a and b, their product is also a rational number.}$

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, $ab=\frac{x_1x_2}{y_1y_2}$ which implies that ab is also rational because $x_1x_2,y_1y_2\in\mathbb{Z}$.



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.

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.