How does GPT-2 compute greater-than?: Interpreting mathematical abilities in a pre-trained language model

The Task
We want to study how language models do math, using circuits. So, we study a small LM, GPT-2 small, on a simple math task:

Input: The war lasted from the year 1741 to the year 17

GPT-2 Small:

Finding and Testing the Greater-Than Circuit

The main contributors to the logits are MLPs 8-11, and a set of attention heads that bring information from other positions. We patch all non-circuit edges; model performance remains the same!

Circuit Semantics
To understand circuit semantics, we apply the logit lens to components, multiplying their outputs by the unembedding matrix.

- MLPs upweight the correct years.
- Attention heads identify the start year, YY.

Generalization
We test if GPT-2 exhibits greater-than behavior in other contexts. In some contexts, it does, using the same circuit; in others, it does not.

Conclusions
- Using path patching / causal ablations, we successfully found a circuit, and causally proved that it was responsible for the task at hand.
- Our circuit generalizes to some extent: it is responsible for greater-than in multiple scenarios.

Behaviors supported by our circuit:
- The price of that [luxury good] ranges from 17[YY] to 17
- 1599, 1607, 1633, 1679, 17[YY], 17
- The [event] ended in the year 17[YY] and started in the year 17
- The [event] lasted from the year 7[YY] BC to the year 7

Behaviors not supported by our circuit:
- 17[YY] is smaller than 17
- 1799, 1753, 1733, 1701, 16[YY], 16
- 1695, 1697, 1699, 1701, 1703, 17

However, GPT-2 cannot perform other mathematical tasks, despite apparent rich number representations.

We hypothesize that our circuit lies between generalization and memorization, because our circuit:
- performs greater-than across contexts
- does not learn generalized math knowledge
- may have memorized the greater-than response