Are language models aware of the road not taken? Token-level uncertainty and hidden state dynamics

Amir Zur Eric Bigelow Atticus Geiger Ekdeep Singh Lubana

Pr(Ai)²R Group CBS-NTT, Harvard University Psychology, Harvard University

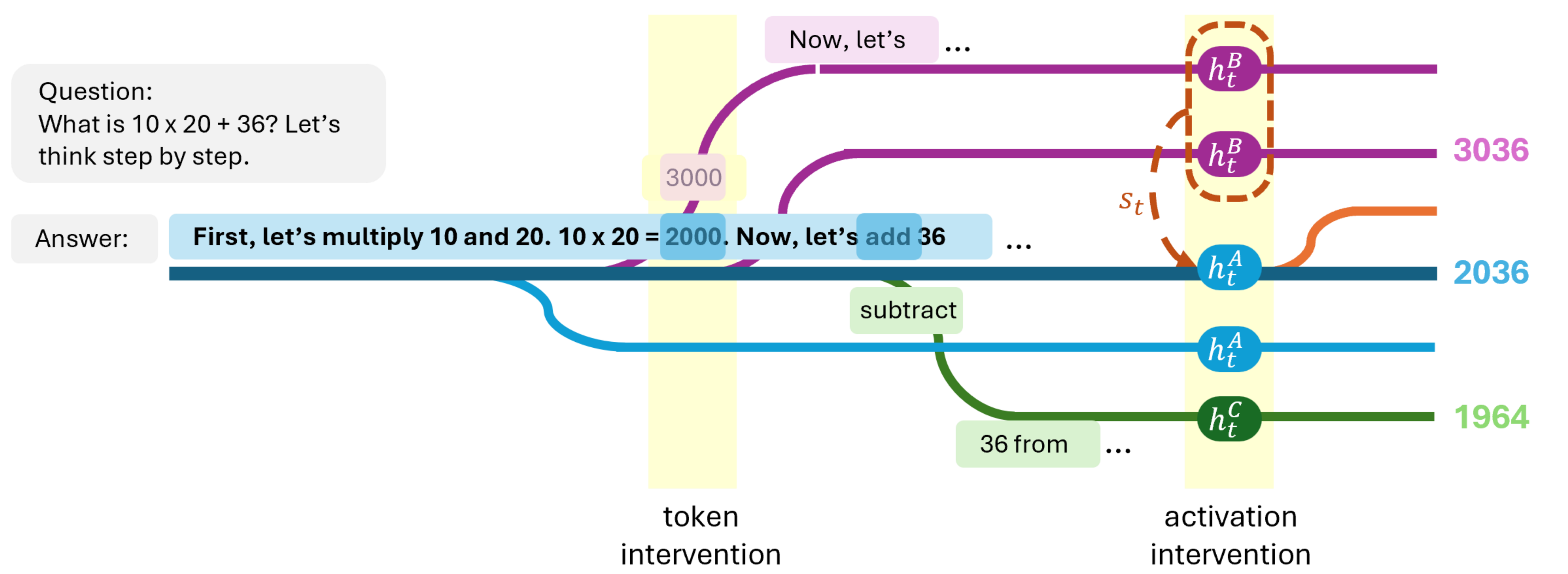
check out our interactive website!

Estimating Outcome Distribution

through forking paths analysis

Taking all paths into account

- When evaluating an LLM, it's important to understand **the full distribution of answers** that it may output!
- For example, outputting harmful texts to a question 1% of the time is still very harmful behavior!
- With **forking paths analysis**, we consider alternate tokens at each generation step to **estimate an LLM's outcome distribution**.



Do models know where they're going?

- Estimating the outcome distribution is **computationally expensive**, requiring generating millions of tokens with forking paths analysis.
- **Inspecting model internals** may be a way forward! We find that **steering vectors** and **linear probes** predict a model's distribution.

Steering Outcome Distribution

through difference-in-means vectors

Question
A book was sold for 27.50 with a 10% profit. If it's sold for 25.75, then what is the percent of profit and loss?

Answer
To find the cost price (CP) of the book, we can use the formula:
CP = Selling Price (SP) / (1 + Profit Percentage)
CP = 27 . 50 / (1 + 0.10) = 25.00
Now, we can calculate the loss as follows:
Loss = CP - SP
...
The correct answer is D) 3% Loss .

Uncertain LLMs are easier to steer

- Answers can contain **forking tokens** that, upon sampling, **drastically change the outcome distribution**.
- Turns out, **steering success** of difference-in-means **significantly correlates** with **the underlying outcome probability**!
- Since **steering success** and **outcome probability** have the **same changepoints**, steering might reveal when models make decisions during generation.

Significant correlation between steering success from answer A to B & the underlying probability of answer B

Predicting Outcome Distribution

through linear probes

LLMs seem to know where their paths will lead

- Training a **linear probe** on the internal states of an LLM, we can **accurately predict** its outcome distribution over each token position.
- The information in the **hidden states** is **more predictive than** the information in **the text alone**. An equally-sized model with a different architecture doesn't get the same accuracy.