

Abstract

引入 backward 推理去验证候选答案

FOBAR, 将前向推理和后向推理结合起来验证, 在数学数据集和三个大模型上达到了比较好的性能。

Introduction

自一致性 (Self-Consistency) 是一种简单有效的方法提高CoT。

使用温度采样, 对一组不同的推理链可能会导致多个候选答案, 选择得票最多的那个作为最终答案。

- 这篇文章中实验表示简单采样更多推理路径不会导致测试精度的提高。
- 应该使用更可靠的验证者。

使用反向推理验证答案

- 对于每个候选答案A, 随机掩码一个问题中的数字为x, 设计一个template 如果答案是A,那么x的值是多少? 由于x已知, 检测掩码x是否被正确预测。
- 一个正确答案比错误答案更可能帮助预测掩码x。

把 the vote of A定义为 成功预测出x值的链数 (即A只为答案, 对一个问题进行多次思维链推导), 即估计出反

向概率 $\mathbb{P}_B(\hat{A}_c)$ 同时结合上前向得到A的概率: $\mathbb{P}(\hat{A}_c) \propto (\mathbb{P}_F(\hat{A}_c))^\alpha (\mathbb{P}_B(\hat{A}_c))^{1-\alpha}$.

Candidate answers generated by Self-Consistency: 21 (16 times), 23 (24 times)

Forward probability: $\mathbb{P}_F(21) = 16/40$, $\mathbb{P}_F(23) = 24/40$

这就是使用自一致性 (Self-Consistency) 得到的前向概率

Backward question (with answer 21): The sum of three consecutive odd numbers is **x**. What is the smallest of the three numbers? If we know the answer to the above question is **21**, what is the value of unknown variable **x**?

(we sample 10 backward chains and all predict **x = 69**, thus, **10** correct backward chains)

Backward question (with answer 23): The sum of three consecutive odd numbers is **x**. What is the smallest of the three numbers? If we know the answer to the above question is **23**, what is the value of unknown variable **x**?

(we sample 10 backward chains and all predict **x = 75**, thus, **no** correct backward chains)

Backward probability: $\mathbb{P}_B(21) = 10/(10 + \epsilon)$, $\mathbb{P}_B(23) = \epsilon/(10 + \epsilon)$

($\epsilon = 10^{-8}$)

这是利用反向预测得到x值得概率, 最后用超参数alpha调整这两种概率。

Related Work

RCOT、MetaMath (使用Backward reasoning 去增强问题以进行微调)

Method

Forward reasoning

$$\mathbb{P}_F(\hat{A}_c) = \frac{1}{M_F} \sum_{i=1}^{M_F} \mathbb{I}(A_i = \hat{A}_c),$$

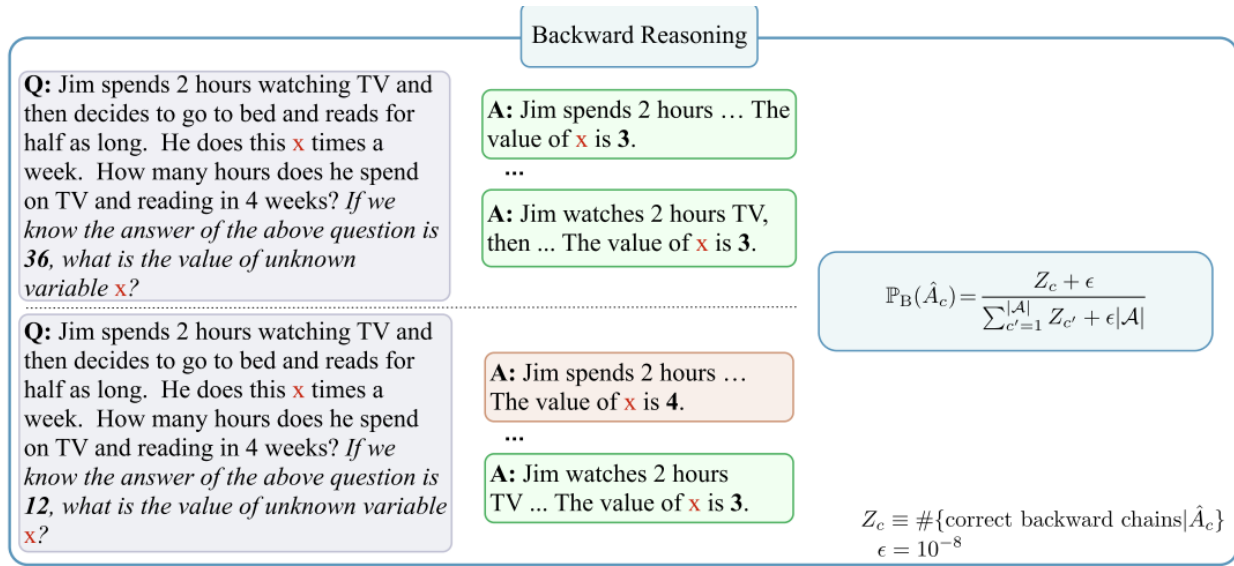
就是用多数投票的方法。

Backward reasoning

用这样的templete

Template For Creating Backward Question

$\mathcal{T}(\hat{A}_c) =$ *If we know the answer to the above question is $\{\hat{A}_c\}$, what is the value of unknown variable x ?*



防止分母为0

FOBAR (FORward and BACKward Reasoning)

$$\mathbb{P}(\hat{A}_c) \propto (\mathbb{P}_F(\hat{A}_c))^\alpha (\mathbb{P}_B(\hat{A}_c))^{1-\alpha},$$

使用

这样的公式就不需要训练了。

Extension to Non-Mathematical Tasks

把猜数值的方法改成让大模型给出一些问题中的关键词，然后猜word。

Experiment

CoT or ComplexCoT