组会汇报

陈钶杰
专业:计算数学

November 28, 2023

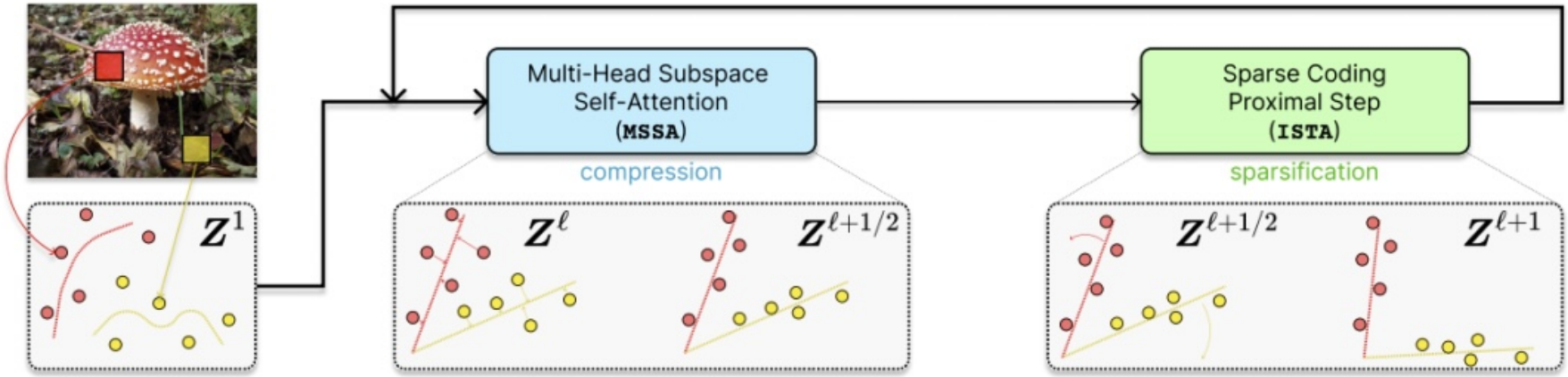
目录

这周主要做的事情

1. 相关文献的阅读.
2. 修改MLC模型,创建一个数字加减乘除的任务,进行训练以及评估测试结果.
3. 在原有的lstm模型基础上添加了注意力机制查看是否有明显效果提升.

NLP相关知识

- 神经网络的学习可能只是对数据集的压缩.
  图像说明：



- 关于提示的中间步骤中，他们使用了过程奖励模型(PRM)，为每一个推理步骤分配一个参数.

例如

The denominator of a fraction is 7 less than 3 times the numerator. If the fraction is equivalent to $2/5$, what is the numerator of the fraction? (Answer: 14)

Let's call the numerator x.

So the denominator is 3x-7.

We know that x/(3x-7) = 2/5.

So 5x = 2(3x-7).

5x = 6x - 14.

So x = 7.

模型部署说明

- 实现简单的加减乘运算的任务
- 任务数据集简介:

```
*SUPPORT*

IN: sum 8 OUT: 1 1 1 1 1 1 1 1
IN: sub 3 OUT: -1 -1 -1
IN: sum 2 sub 7 OUT: 1 1 -1 -1 -1 -1 -1 -1 -1
IN: sub 7 sum 2 OUT: -1 -1 -1 -1 -1 -1 -1 1 1
IN: 8 prod 2 OUT: 8 8
IN: sum 3 sub 7 sub 8 OUT: 1 1 1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1
IN: sum 2 sub 8 sum 8 OUT: 1 1 -1 -1 -1 -1 -1 -1 -1 -1 1 1 1 1 1 1 1 1
IN: 2 prod 8 sum 3 OUT: 2 2 2 2 2 2 2 2 1 1 1
IN: 7 prod 3 sub 2 OUT: 7 7 7 -1 -1
IN: 2 prod 8 sum 2 OUT: 2 2 2 2 2 2 2 2 1 1 1 1 1 1 1 1 1
IN: 3 prod 2 sub 7 OUT: 3 3 -1 -1 -1 -1 -1 -1 -1 1 1 1 1 1 1 1 1
IN: 3 prod 8 sum 7 OUT: 3 3 3 3 3 3 3 3 1 1 1 1 1 1 1 -1 -1
IN: 2 prod sum 7 OUT: 1 1 1 1 1 1 1 1 1 1 1 1 1 1
IN: 8 prod sub 3 OUT: -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1

*QUERY*

IN: sum 2 OUT: 1 1
IN: sub 7 OUT: -1 -1 -1 -1 -1 -1 -1
IN: sum 8 sub 3 OUT: 1 1 1 1 1 1 1 1 -1 -1 -1
IN: sub 3 sum 8 OUT: -1 -1 -1 1 1 1 1 1 1 1
IN: 3 prod 7 OUT: 3 3 3 3 3 3 3
IN: sum 7 sub 3 sum 2 OUT: 1 1 1 1 1 1 1 -1 -1 -1 1 1
IN: 3 prod 8 sum 7 OUT: 3 3 3 3 3 3 3 3 1 1 1 1 1 1 1
IN: 7 prod 2 sub 3 OUT: 7 7 -1 -1 -1
IN: 7 prod sum 3 OUT: 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
IN: 2 prod sub 8 OUT: -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 -1

*GRAMMAR*

sum -> 1
sub -> -1
2 -> 2
8 -> 8
3 -> 3
7 -> 7
u1 2 -> [u1] [u1]
u1 8 -> [u1] [u1] [u1] [u1] [u1] [u1] [u1] [u1]
u1 3 -> [u1] [u1] [u1]
u1 7 -> [u1] [u1] [u1] [u1] [u1] [u1] [u1]
2 prod u1 -> [u1] [u1]
8 prod u1 -> [u1] [u1] [u1] [u1] [u1] [u1] [u1] [u1]
3 prod u1 -> [u1] [u1] [u1]
7 prod u1 -> [u1] [u1] [u1] [u1] [u1] [u1] [u1]
```

模型评估说明

- 泛化能力测试:

```
*SUPPORT*
IN: sum two OUT: x x
IN: sub six OUT: x x x x x x
IN: sum two sub seven OUT: x x y y y y y y y
IN: sub seven sum two OUT: y y y y y y y x x
IN: d my_prod two OUT: d d
IN: sum six sub seven sub seven OUT: x x x x x x y y y y y y y x x x x x x
IN: sum two sub seven sum seven OUT: x x y y y y y y y x x x x x x
IN: a my_prod seven sum six OUT: a a a a a a a x x x x x x
IN: d my_prod six sub two OUT: d d d d d d y y
IN: b my_prod seven sum two OUT: b b b b b b x x
IN: c my_prod two sub seven OUT: c c y y y y y y y
IN: c my_prod seven sum seven OUT: c c c c c c c x x x x x x x
IN: a my_prod sum seven OUT: a a a a a a a a a a a a a a a
IN: d my_prod sub two OUT: d d d d d d d d d d d d d d d

*QUERY*
IN: sum three OUT: x x x
IN: sub seven OUT: y y y y y y y
IN: sum seven sub six OUT: x x x x x x x y y y y y y
IN: sub six sum seven OUT: y y y y y y x x x x x x x
IN: c my_prod seven OUT: c c c c c c c
IN: sum seven sub six sum two OUT: x x x x x x x y y y y y y x x
IN: c my_prod seven sum seven OUT: c c c c c c c x x x x x x x
IN: a my_prod two sub six OUT: y y y y y y y y y y y y y
IN: b my_prod sum four OUT: x x x x x x x x x x x x x x x x x
IN: a my_prod sub seven OUT: y y y y y y y y y y y y y y y

*GRAMMAR*
sum -> x
sub -> y
a -> a
b -> b
c -> c
d -> d
u1 two -> [u1] [u1]
u1 four -> [u1] [u1] [u1] [u1]
u1 six -> [u1] [u1] [u1] [u1] [u1] [u1]
u1 seven -> [u1] [u1] [u1] [u1] [u1] [u1] [u1]
u1 my_prod two -> [u1] [u1]
u1 my_prod four -> [u1] [u1] [u1] [u1]
u1 my_prod six -> [u1] [u1] [u1] [u1] [u1] [u1]
u1 my_prod seven -> [u1] [u1] [u1] [u1] [u1] [u1] [u1]
```

模型结果说明

- 目前做的只是10以内的加减乘的数学方法,通过一定的微调可以保证准确性
- 当数据变多以后就不确定这样做是否可以继续有效?
- 还有一些bug需要修改.

```
support items;
   sum 2 -> 1 1
retrieval items; 0.0 % correct
   sum 2 -> 1 1 1 1 1 1 1 1 1 1 (** target: 1 1)
```

Table: 准确率结果(模型)

模型及参数	lstm	lstm(添加注意力机制)
训练1	30.99%	31.68%
训练2	32.85%	31.42%
训练3	32.09%	28.96%
训练4	31.63%	32.58%
训练5	32.36%	30.95%

1. 在解码器中引入了注意力机制
2. 加入注意力机制以后变得更加相比没加之前稳定性有变差.

下一步计划

1. 将简单的加减乘任务尝试在数值上更加复杂,探究其泛化能力.
2. 更换注意力机制的变体以及在不同编码器上进行测试查看效果.

谢谢老师和同学们的聆听!