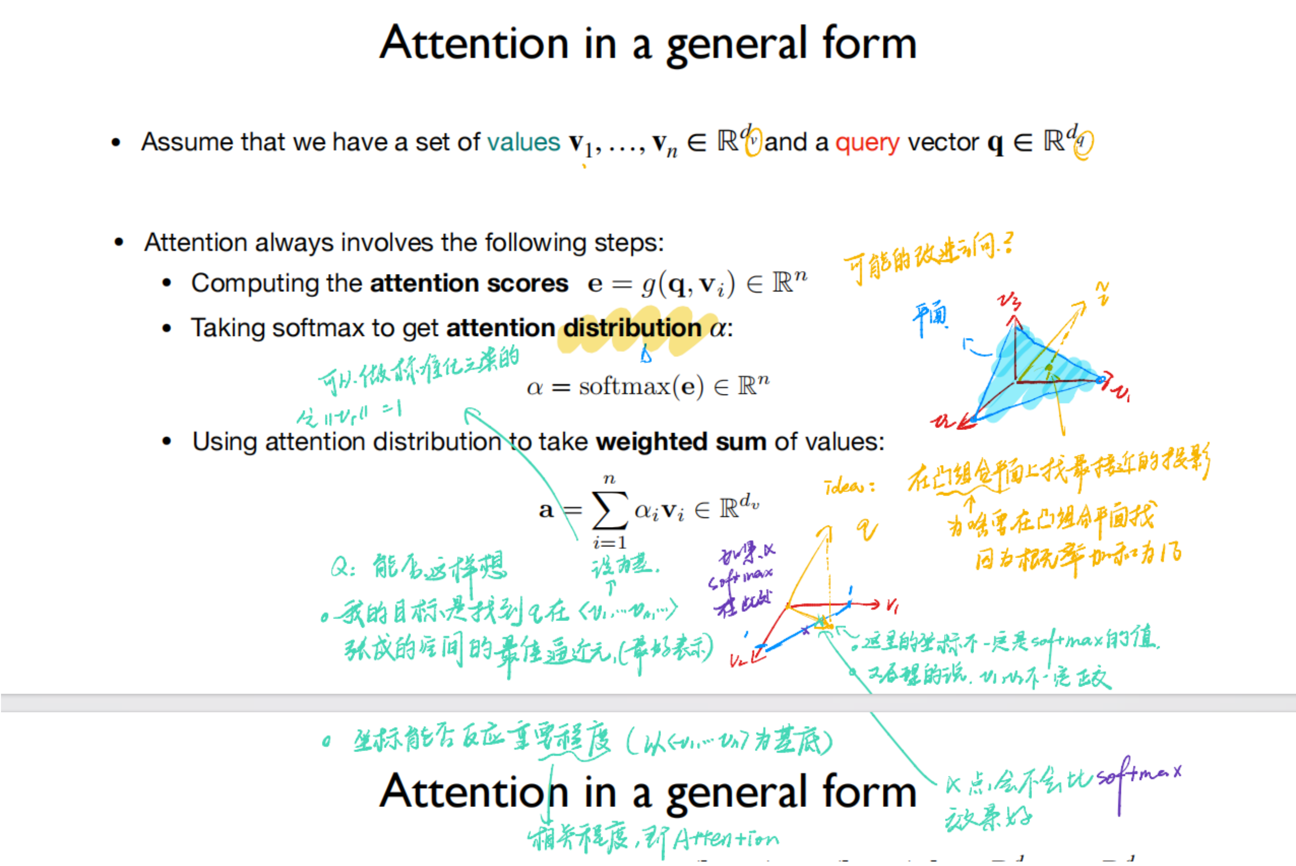
Two Topics

Attention

Q: Why attention work?



Attention in a general form

- Assume that we have a set of values $\mathbf{v}_1, \ldots, \mathbf{v}_n \in \mathbb{R}^{d_v}$ and a query vector $\mathbf{q} \in \mathbb{R}^{d_q}$
- Attention always involves the following steps:
  - Computing the **attention scores** $\mathbf{e} = g(\mathbf{q}, \mathbf{v}_i) \in \mathbb{R}^n$
  - Taking softmax to get **attention distribution** α:

$$\alpha = \mathrm{softmax}(\mathbf{e}) \in \mathbb{R}^n$$

  - Using attention distribution to take **weighted sum** of values:

$$\mathbf{a} = \sum_{i=1}^{n} \alpha_i \mathbf{v}_i \in \mathbb{R}^{d_v}$$

可以做标准化之类的

令 $\|v_i\| = 1$

可能的改进方向？

Idea: 在凸组合平面上找最接近的投影

为啥要在凸组合平面找

因为概率加和为1

Q: 能否这样想

设为基.

- 我的目标是找到 q 在 $\langle v_1, \cdots v_n, \cdots \rangle$ 张成的空间的最佳逼近元.(带权表示)

如果α softmax 在此处

- 这里的坐标不一定是softmax的值.
- 又合理的说. v_1, v_2 不一定正交

- 坐标能否反应重要程度 (以 $\langle v_1, \cdots v_n \rangle$ 为基底)

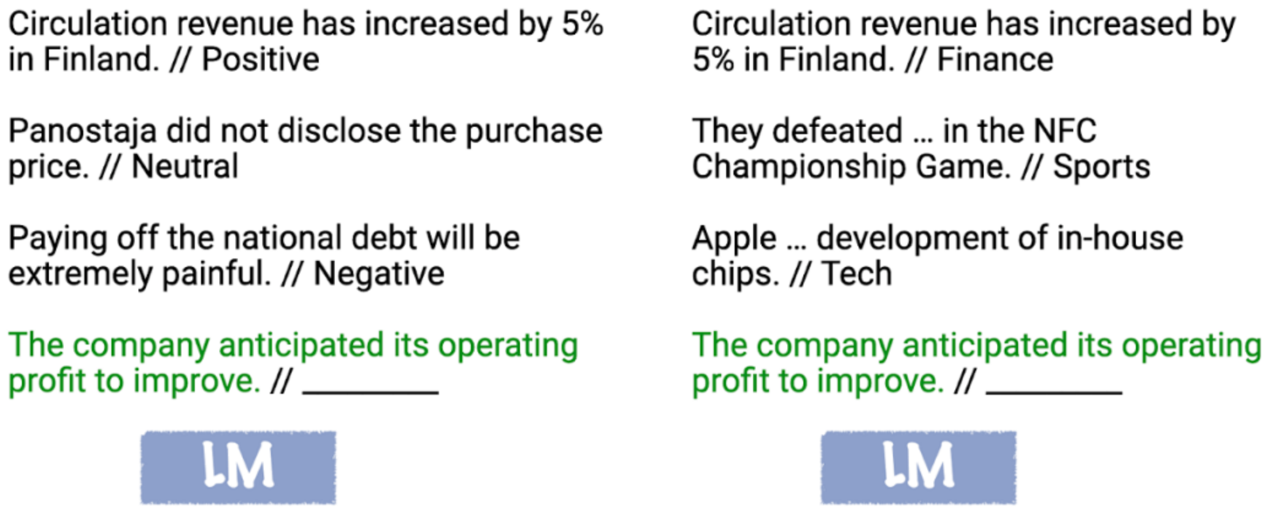
相关程度, 即Attention

α点,会不会比softmax 效果好

Attention in a general form

In context learning

Q:In-context learning is a mysterious emergent behavior in large language models (LMs) where the LM performs a task just by conditioning on input-output examples, without optimizing any parameters.



Circulation revenue has increased by 5% in Finland. // Positive

Panostaja did not disclose the purchase price. // Neutral

Paying off the national debt will be extremely painful. // Negative

The company anticipated its operating profit to improve. // __________

Circulation revenue has increased by 5% in Finland. // Finance

They defeated ... in the NFC Championship Game. // Sports

Apple ... development of in-house chips. // Tech

The company anticipated its operating profit to improve. // __________

Bayesian inference view

- **Pretraining distribution** (p): Our main assumption on the structure of pretraining documents is that a document is generated by first sampling a latent concept, and then the document is generated by conditioning on the latent concept. We assume that the pretraining data and LM are large enough that the LM fits the pretraining distribution exactly. Because of this, we will use p to denote both the pretraining distribution and the probability under the LM.
- **Prompt distribution**: In-context learning prompts are lists of IID (independent and identically distributed) training examples concatenated together with one test input. Each example in the prompt is drawn as a sequence conditioned on the same prompt concept, which describes the task to be learned.

The process of "locating" learned capabilities can be viewed as Bayesian inference of a prompt concept that every example in the prompt shares. If the model can infer the prompt concept, then it can be used to make the correct prediction on the test example. Mathematically, the prompt provides evidence for the model (p) to sharpen the posterior distribution over concepts, $p(concept \mid prompt)$. If $p(concept \mid prompt)$ is concentrated on the prompt concept, the model has effectively "learned" the concept from the prompt.

$$p(\text{output}|\text{prompt}) = \int_{\text{concept}} p(\text{output}|\text{concept}, \text{prompt}) p(\text{concept}|\text{prompt}) d(\text{concept})$$

Ideally, $p(concept \mid prompt)$ concentrates on the prompt concept with more examples in the prompt so that the prompt concept is "selected" through marginalization.

Note

In-context-learning（few-shot）通过prompt中给出的几个例子帮助LLM**定位**需要解决问题对应在**预训练数据集**的**concept**（latent concept）,并基于concept去解决对应的问题。

第一步: 通过例子学习问题关于concept的分布,也就是所谓的**定位**

$p(concept|example\ in\ prompt)$

第二步:基于prompt给出output

$p(output|example\ in\ prompt) = \int p(output|concept, example\ in\ prompt) p(concept|example\ in\ prompt) d(concep$

```
可能可以考虑的实验设计
Pretrain data(预训练数据):
{
    一大堆不同分布(正态,卡方,泊松)的数据,形式:
    {
        "x": ... ,
        "distribution category:"...,
        "P(x)":...
    }
}

prompt for In-context-learning:
Idea:定位其实是定位对应任务的分布

1.
以(x,P(x))的形式给出一些在预训练集中出现过的
    {
        "x": ... ,
        "distribution category:" 固定比如正态
        "P(x)":...
    }
的样例,然后问x_0 = ...时,【P(x_0)是多少】,【distribution category是什么】(没见过的任务)
---
2.
在上述案例的基础上加上随意标注的distribution category:(x,P(x),distribution category)
然后问x_0 = ...时,【P(x_0)是多少】,【distribution category是什么】(没见过的任务),
看结果是否收到影响
---
3.
看不用预训练数据集中出现过的数据,是否会更难定位到问题
给出不在预训练数据集中但同属正态分布的样例(x,P(x)),
 的样例,然后问x_0 = ...时,【P(x_0)是多少】,可以问很多组,然后对比和1的准确性
(如果只是locate,感觉用预训练数据中出现过的数据集效果最好, 为了极端,可以在预训练数据集中使用集中在2sigma内的样本点,但是In context主要提供3 -sigma外的样本点)
---
4.
在的基础上混入一些其他分布的数据,实验不同的掺杂比例,看能否正确分类和做预测
```

说明预训练数据集提供的universal knowledge内知识自己的分布也非常重要,但是In-context learning貌似在没见过的任务的导航也可以表现很好,说明他的作用可能是帮助重塑universal knowledge 内含的分布