**GPT-BASE**

**引言：**

Generative Pre-trained Transformer（GPT）系列是由OpenAI提出的预训练语言模型，与Google的BERT模型一样，它们都是以Transformer为核心结构的模型，BERT与早期的GPT在面对不同的NLP任务时都会采用微调的方式来适应下游任务，而随着GPT系列的不断发展，GPT展现出了之前从未有过的强大能力，既只需要简单的上下文学习，就可以让模型适应各种下游任务，这种范式的出现，立即引发了NLP领域的地震，随着ChatGPT以及GPT4的发布，各种振奋人心的成果，不断“涌现”出来。

**GPT系列**

1. **简介：**
   1. 基于文本预训练的GPT-1，GPT-2，GPT-3三代模型都是采用的以Transformer为核心结构的模型（图1），不同的是模型的层数和词向量长度等超参，其中GPT3采用了更大的数据与参数来进行语言模型建模任务；
   2. 论文名称：《Language Models are Few-Shot Learners》；

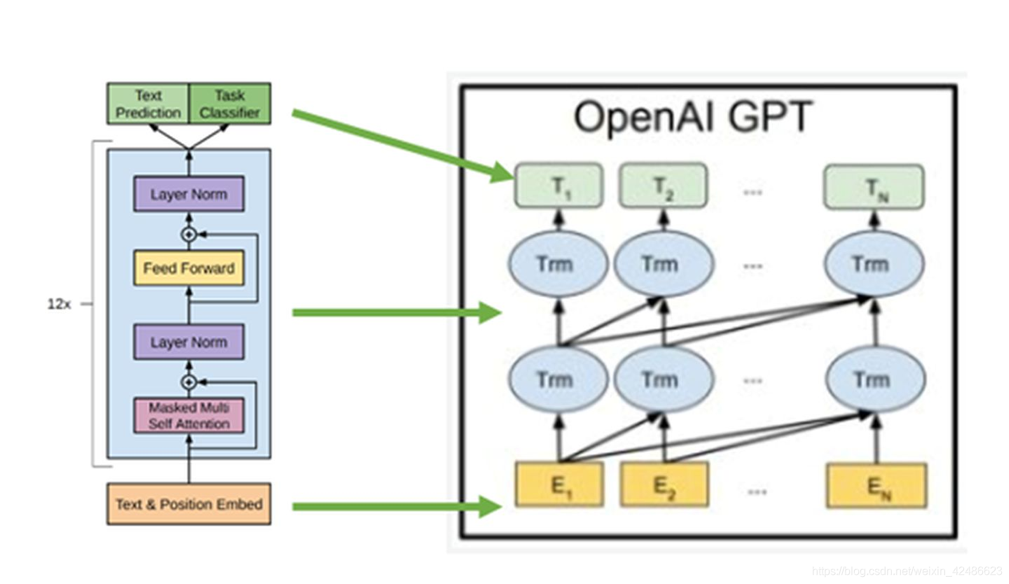
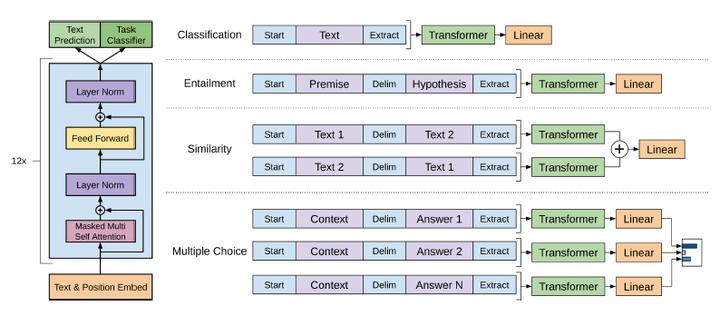


图1

1. **GPT-1**
   1. 在GPT-1（ELMO）之前，NLP任务需要大量的标注样本进行领域任务的训练，而这样做的缺陷为：
      1. 需要大量高质量的标注数据，而这些数据获取的成本是非常高的。
      2. 无法将现有模型泛化到训练数据集以外的任务。
   2. GPT-1通过Transformer结构打破了这个现状。
      1. 通过无监督训练，解决需要大量高质量标注数据的问题，降低数据成本。
      2. 通过大量不同类型混杂语料进行预训练，解决训练任务的泛化问题。
   3. GPT-1原理（如图所示）：



* + 1. GPT-1是一个典型的自回归语言模型，当我们给定一个无标签序列时，模型的优化目标为最大化似然函数：



其中 K是滑动窗口大小，P是条件概率，是模型参数。

* + 1. GPT-1使用了12个transformer块的结构作为解码器，每个transformer块是一个多头的自注意力机制，然后通过全连接得到输出的概率分布。
  1. 有监督微调
     1. 当进行完无监督训练后，我们将它的值直接应用到有监督任务中。对于一个有标签的数据集 C ，每个实例有 m 个输入token： 它对于的标签 y 组成。首先将这些token输入到训练好的预训练模型中，得到最终的特征向量 。然后再通过一个全连接层得到预测结果 y：



* + 1. 有监督的目标则是最大化y的值：



* + 1. 最后把有监督与无监督合并在一起：



* 1. 预训练数据
     1. GPT-1使用了包含 7,000本没有发布的书籍BooksCorpus数据集来进行训练。

1. GPT-2
   1. GPT2的架构与GPT-1基本一致，但扩充了大量的数据进行预训练，相较于GPT-1(117M)的参数量，GPT-2来到了（1.5B）的量级，是GPT-1的十倍之上，而扩大的参数量来验证，GPT2在zero-shot上的能力。
2. GPT-3：
   1. GPT-3是一个非常强大的语言模型，仅仅需要zero-shot或few-shot，就可以在下游任务表现的非常好。除了常见的NLP任务外，GPT-3在一些较为困难的任务上也有很惊艳的表现，例如：
      1. 甚至编写代码，进行推理题目，进行数据计算，而这些强大的能力的能力则依赖于GPT-3巨大的 175B的参数量， 45TB的训练数据以及高达 1,200万美元的训练费用。
   2. **预训练数据：**
      1. 预训练数据集为Books 1，Books 2，WebText2，Common Crawl（filtered）具体如下表所示：

|  |  |  |
| --- | --- | --- |
| 数据集 | 规模 | 语言 |
| Books1 | 2.2G | 英语 |
| Books3 | 37GB | 英语 |
| Wikipedia | 0.55G | 英语 |
| OpenWebText | 40G | 英语 |
| Common Crawl(filtered) | 863G | 75%的英语剩下为其它语言 |

其中WebText2的样例如下：

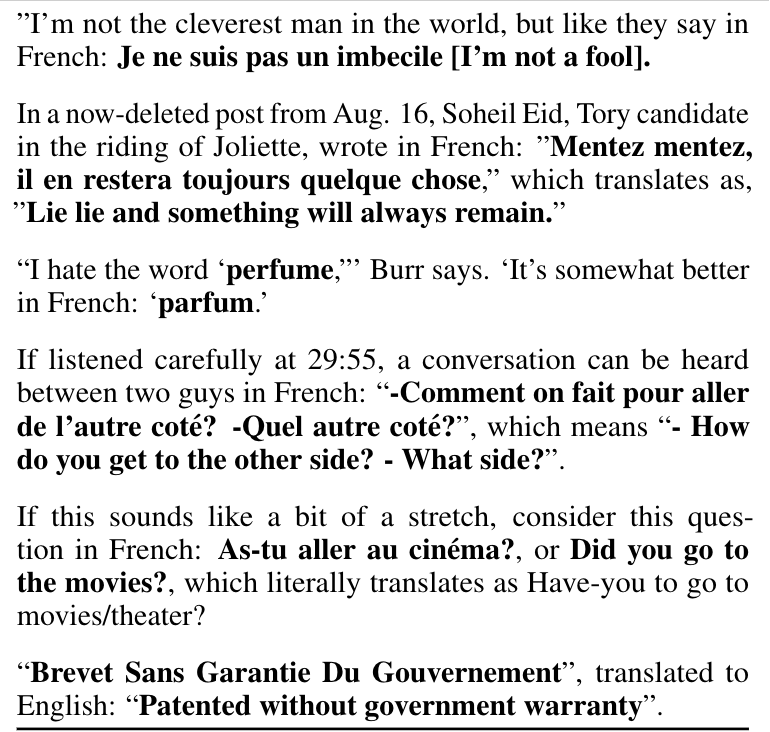


图3

* 1. **GPT3的模型结构及参数量：**
     1. 模型使用了Transformer decoder端，但在结构上做出了如下变化
     2. Layer Normalization移动到了每一块的输入部分，在每个self-attention之后额外添加了一个Layer Normalization；
     3. 将残差层的初始化值用 1/N 进行缩放，其中 N 是残差层的个数；
     4. 使用了Sparse Self Attention结构（相对距离不超过k的注意力都设为0）
     5. 模型的参数量如图4：

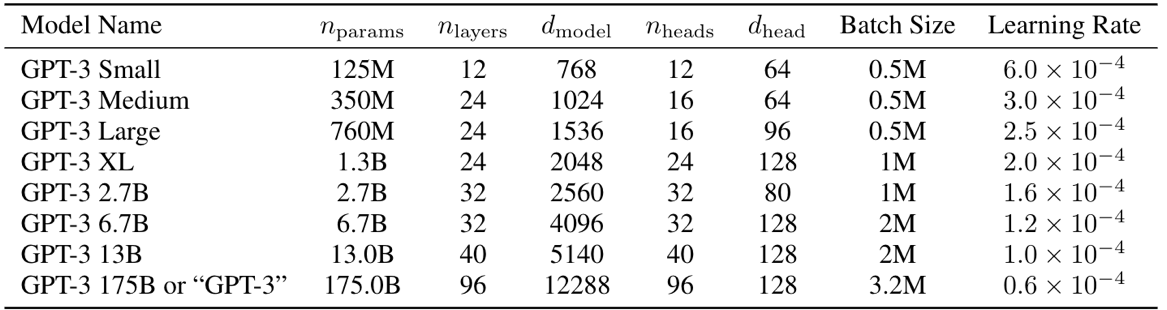


图4

* 1. **GPT3所展现的能力：**
     1. GPT3不需要进行Finetune，其利用超大参数并构造prompt来进行下游任务，并且模型不进行梯度的更新，其中下游任务分为： zeros-shot，one-shot，few-shot的形式，如（图5）

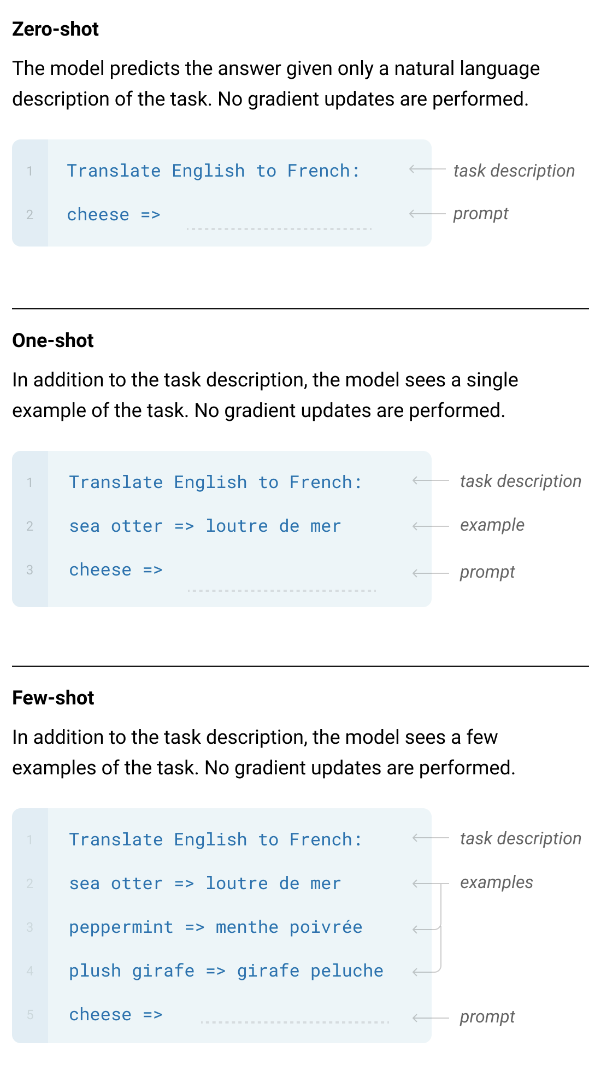


图5

**InstructGPT**

1. **简介:**
   1. InstructGPT是基于GPT3的模型，其目的是消除生成内容中不真实的，有毒的，或者是没有帮助的答案（来自于预训练数据集），例如：
      1. 对面那个黑鬼（这句话就充满了歧视）
   2. 通过来提升模型生成内容的安全性和可控性；
   3. 论文名称：《Training language models to follow instructions with human feedback》
2. **预训练数据：**
   1. 有两部分组成一部分来自于OpenAI的用户的数据，一部分来自于标注工人编写；
      1. 用户数据超过 96% 是英语，还有一小部分其他语言；
      2. 收集了200个问题，并根据用户ID进行划分；
      3. 用户数据的分布如图6：

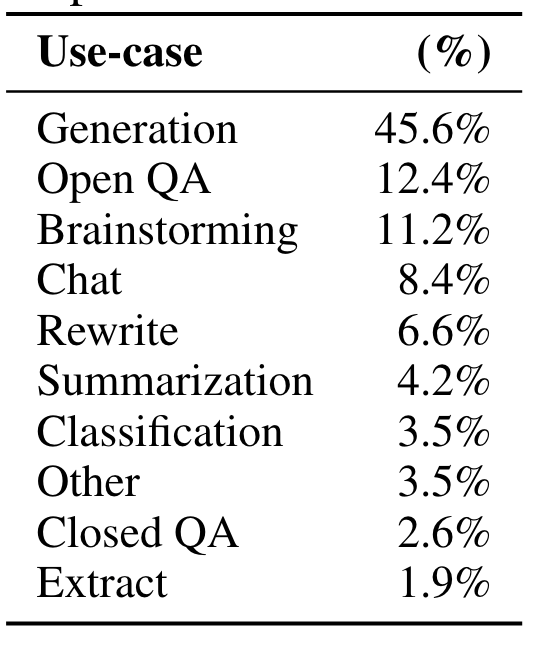


图6

* 1. 标注人员标注用户数据；
  2. 标注涉及的任务包括：Generation, open QA, closedQA, brainstorming, chat, rewriting, summarization, classification, extraction, or other；
  3. 具体的数据大小如下：

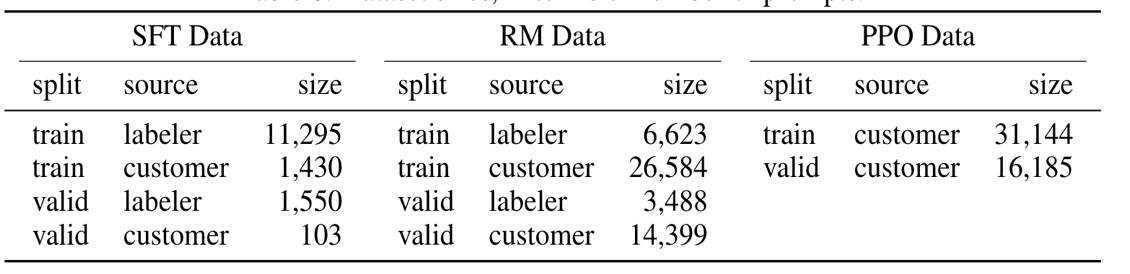


图7

1. **来源与构造方式：**
   1. SFT数据集：
      1. 来自OpenAI的用户数据与40个标注人员标注的样本数据。标注的样例如下：

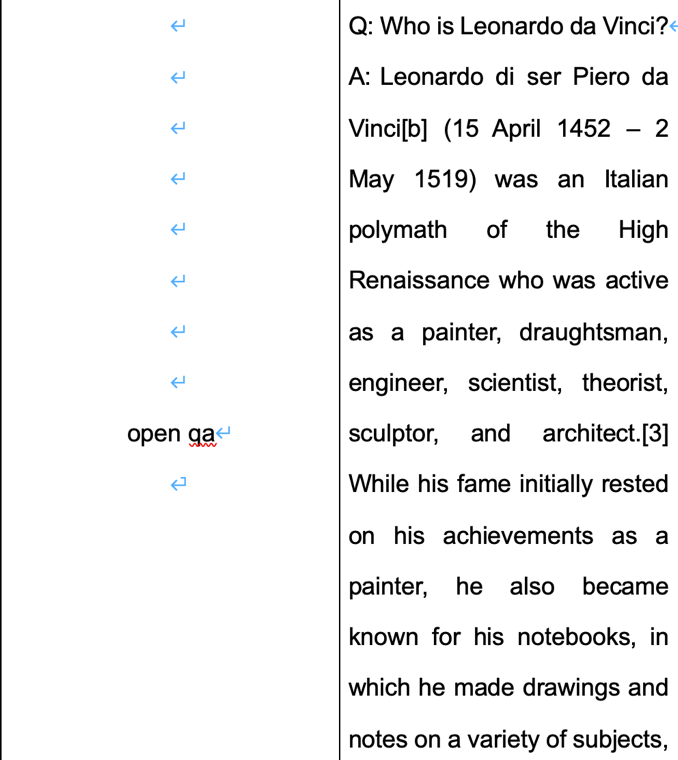


图8

* 1. RM数据集：
     1. Prompt来源：

1. 来自OpenAI 51094条用户数据；
   * 1. Response来源：
2. 使用SFT数据对GPT3进行微调；
3. 把51094条用户数据输入到微调过后的模型
4. 利用beam search生成k个答案（k的范围为0~9），并让标注人员根据生成数据的质量对这些生成内容进行排序，如(图9)：

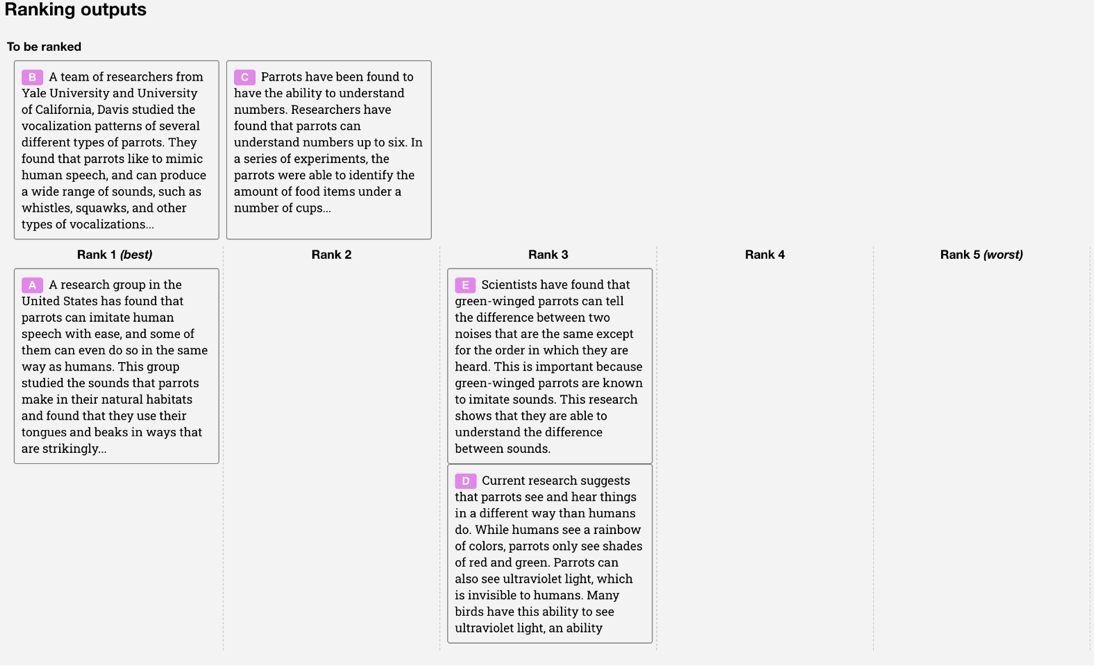


图9

* 1. PPO数据集：
     1. 来自OpenAI的47329条用户数据，此数据只包含prompt，形式如下：

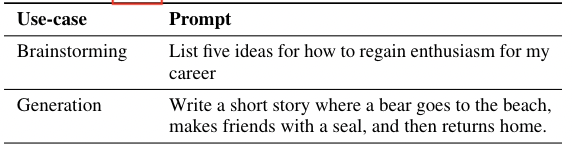


图10

1. **模型结构：**
   1. 模型的整体流程如下：

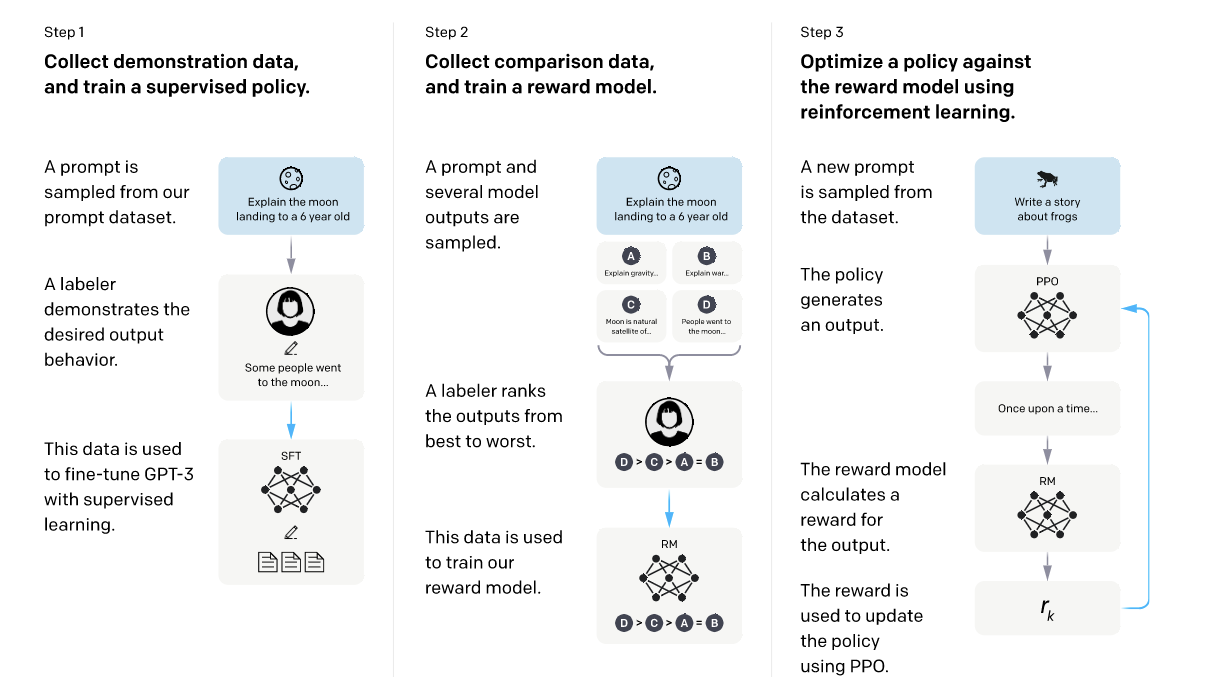


图11

* 1. SFT模型：
     1. 同GPT3模型，都是Transformers的decoder端。
  2. RM模型：
     1. 基础模型为6B大小的SFT模型，选择6B模型的原因是：

1. 6B模型节省了大量计算；
2. 175B 训练可能不稳定，不太适合用做RL期间的奖励函数；(引用自李沐)
   * 1. RM模型结构如图12所示

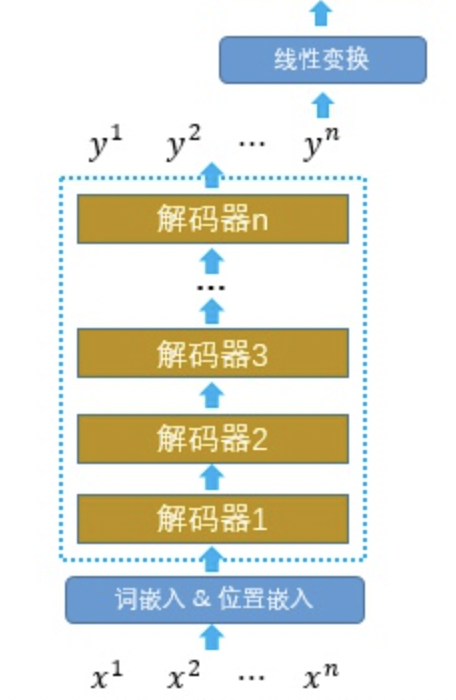
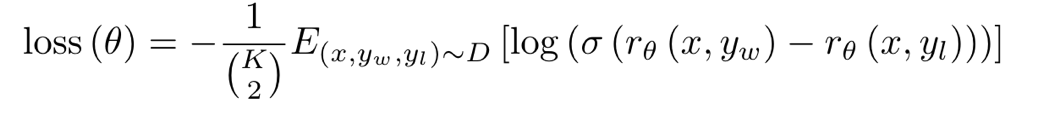


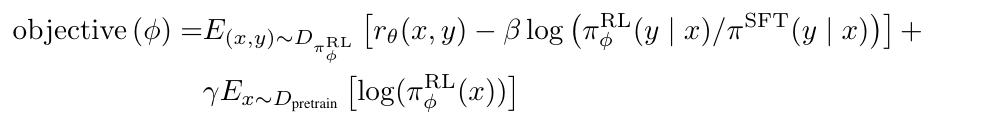
图12

* + 1. 模型的输入：每个prompt与个reponse做为一条样本；

1. 这里假设k=3，那么输入可能是{prompt:[(a,b),(a,c),(b,c)]}，其中a>b，a>c，b>c；
   * 1. 模型输出：线性层输出的值；
     2. 损失函数为：



* + 1. 损失函数的目标：最大化标注者喜欢跟不喜欢的response之间的差值；
    2. 其中是Prompt，是所对应的句子，是和 在参数为的RM模型下的奖励值， 是标注者更喜欢的响应结果，是标注者不喜欢的响应结果，是整个训练数据集。
  1. PPO模型：
     1. 基础模型为SFT模型；
     2. 损失函数如下：



* 1. 损失函数的目标是：生成的reponse分数值较高；
  2. 加入KL惩罚项来确保PPO模型的输出和SFT的输出差距不会很大；
  3. 加入了通用的语言模型目标，来解决通用NLP任务上性能的大幅下降的问题。

1. **实验:**
   1. 文章主要在（Helpful、Honest、Harmless）上对于模型做衡量；
   2. 使用只包含来自用户的3196条prompt数据经过下图模型，并进行人工评估判断是否对人有帮助；

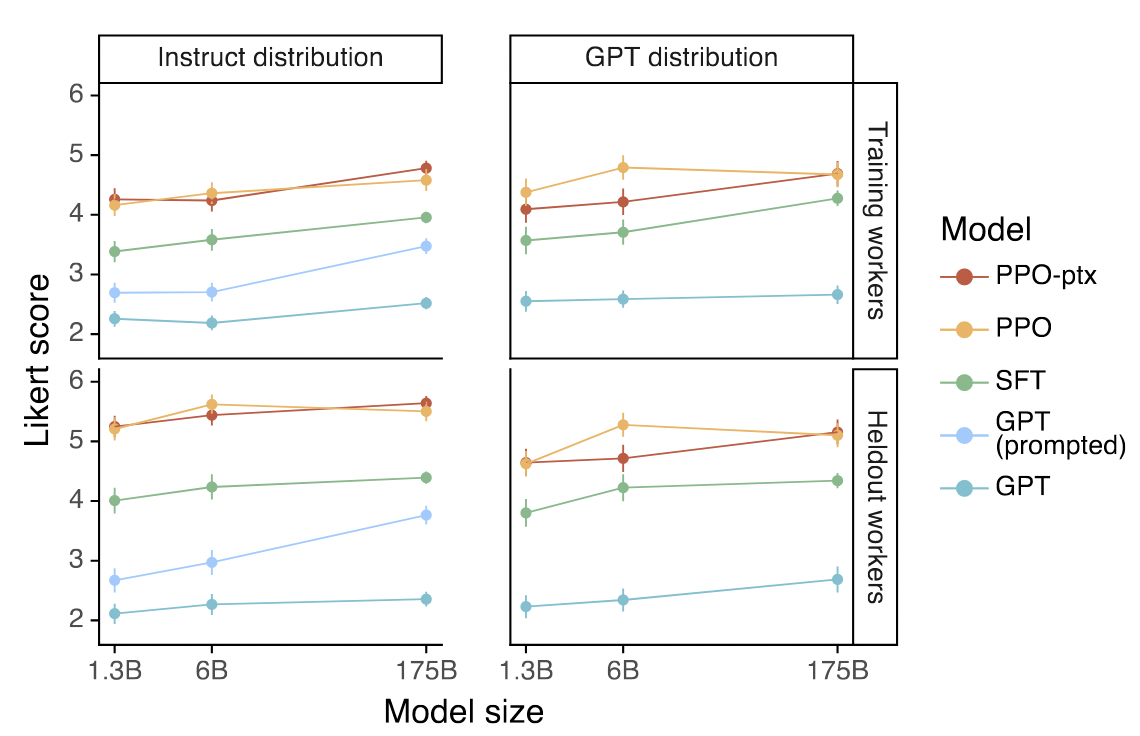


图13

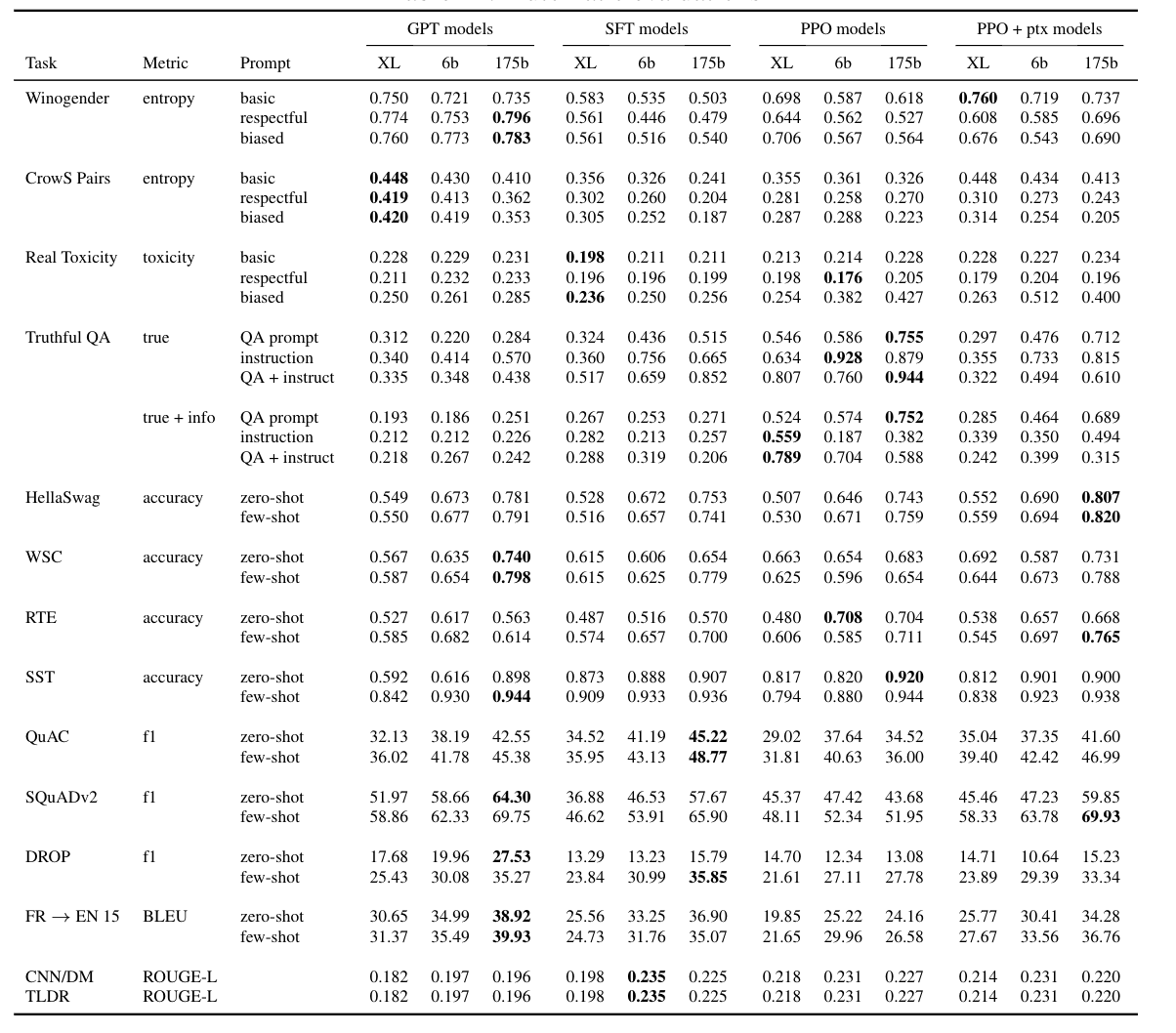
* 1. heldout workers是未参与标注的人员的评估，Trainingworkers是参与标注的人员的评估；
  2. 左侧是人工评分的平均值；
  3. 可以看出不管是SFT、PPO、PPO-ptx都要比GPT3要好
  4. 再多个数据集上的实验

图14

* + 1. TruthfulQA 基准测试中，InstructGPT 生成真实且信息丰富的答案频率大约是 GPT-3 的两倍；
    2. RealToxicity 基准测试中，InstructGPT 与 GPT-3 相比毒性输出比 GPT-3有所减少，但在Winogender与CrowS Pairs的测试中，偏见没有改变
    3. 与GPT3相比模型在一些公共的NLP 数据集上性能变差，特别是 SQuAD、DROP、HellaSwag和 WMT 2015，这是因为InstructGPT着重优化了模型输出的帮助性等问题，从而降低了模型的通用性，但是通过加入语言模型的损失（PPO-ptx）项后，提升了PPO模型在这些通用数据集上的效果；

1. **优势：**
   1. InstructGPT不会像GPT-3一样捏造事实；
   2. InstructGPT比GPT-3在对人有帮助的回答上有些许提升;
2. **缺陷：**
   1. InstructGPT会降低模型在通用NLP任务上的效果；
   2. 有时候InstructGPT会给出一些荒谬的答案；
   3. 模型对指示非常敏感；
   4. 模型对简单概念的过分解读；
   5. 对有害的指示可能会输出有害的答复。