「GPT1」Improving Language Understanding by Generative Pre-Training

在发表《Learning to Generate Reviews and Discovering Sentiment》之后，GPT1 紧接着被 OpenAI 发表。其实我感觉 GPT1 的工作其实是刚开说的这篇文章的一个延伸，不同点在于：

1、将原有的 LSTM 结构替换为了 Google 的 Transformer Decoder

2、将任务从情感分析领域拓展到了更多的 NLP 领域

3、提出了无监督学习的生成表征可以 zero-shot 应用于下游任务，效果还不错

Google 的 Transformer 发表让 OpenAI 在无监督这条路上更容易走了下去，GPT1 主旨就是使用了无监督预训练+多任务监督微调的形式。首先作者便阐述了不准备使用单词级进行文本编码，给出了两个原因：

1、First, it is unclear what type of optimization objectives are most effective at learning text representations that are useful for transfer.

2、Second, there is no consensus on the most effective way to transfer these learned representations to the target task

随后便介绍了打算用无监督进行前期预训练，模型采用 Transformer Decoder ，数据集使用了 BooksCorpus dataset 和 1B 的 Word Benchmark，学习率 2.5e-4 + cosine warmup，Adam 的优化器，epoch 100以及 batchsize 64，最大长度512个 tokens，并用了 L2 正则化。数据集使用了 ftfy 进行清洗，并用 spacy 作为 tokenizer。预训练这里没有给出特别多的细节，只给了公式，因为具体的训练细节在开头给的文章中已经介绍过了。

$$L_1(\mathcal{U}) = \sum_i \log P(u_i|u_{i-k}, \ldots, u_{i-1}; \Theta)$$

无监督预训练后作者针对不同的任务进行 fine-tune，这里 fine-tune 是根据不同任务的数据进行特定的无监督+监督学习模式，其中监督 loss为：

$$L_2(\mathcal{C}) = \sum_{(x,y)} \log P(y|x^1, \ldots, x^m).$$

那么 fine-tune 阶段最终是两个 loss 进行相加，并带有一个超参：

$$L_3(\mathcal{C}) = L_2(\mathcal{C}) + \lambda * L_1(\mathcal{C})$$

作者通过图跟我们说明了不同任务的数据构造以及模型的结构：

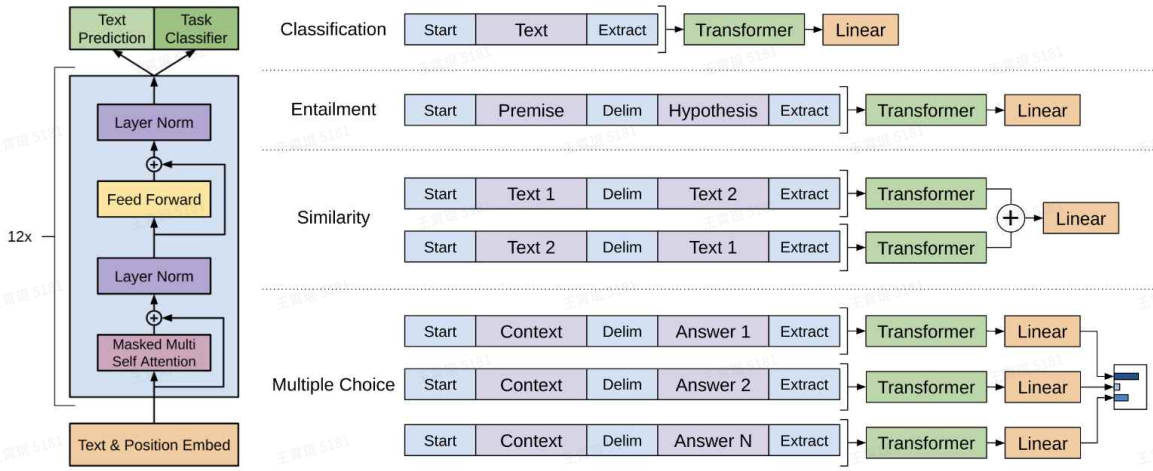

Figure 1: **(left)** Transformer architecture and training objectives used in this work. **(right)** Input transformations for fine-tuning on different tasks. We convert all structured inputs into token sequences to be processed by our pre-trained model, followed by a linear+softmax layer.

在图中可以看到作者在做无监督和 fine-tune 时，训练数据构造是不一样的，无监督是直接一连串的文本输入，占满最大的 512 token 位，但是在 fine-tune 时每条数据被加上了一些 special tokens「Start」、「Delim」、「Extract」，作者也解释了为什么要这么做：Since our pre-trained model was trained on contiguous sequences of text, we require some modifications to apply it to these tasks，说白了就是要让模型知道当前是在让它学习到一些特殊知识，就是给它构造一种 prompt，只不过现在这个 prompt 概念还没有被提出来，这里作者阐述了几个任务是怎么构造 prompt 的：

> **Textual entailment** For entailment tasks, we concatenate the premise p and hypothesis h token sequences, with a delimiter token ($) in between.
>
> **Similarity** For similarity tasks, there is no inherent ordering of the two sentences being compared. To reflect this, we modify the input sequence to contain both possible sentence orderings (with a delimiter in between) and process each independently to produce two sequence representations h_l^m which are added element-wise before being fed into the linear output layer.
>
> **Question Answering and Commonsense Reasoning** For these tasks, we are given a context document z, a question q, and a set of possible answers $\{a_k\}$. We concatenate the document context and question with each possible answer, adding a delimiter token in between to get $[z; q; \$; a_k]$. Each of these sequences are processed independently with our model and then normalized via a softmax layer to produce an output distribution over possible answers.

为什么 fine-tune 要采用无监督+有监督一块学这一套，作者发现一个是能够改变监督模型的泛化性，另一个就是能够加速模型的收敛。不同 fine-tune 的数据集如下，dropout 率0.1，学习率6.25e-5 并采用线性 warm-up，batch-size 为32，epoch为3，loss 超参设置为0.5。

Table 1: A list of the different tasks and datasets used in our experiments.

Task	Datasets
Natural language inference	SNLI [5], MultiNLI [66], Question NLI [64], RTE [4], SciTail [25]
Question Answering	RACE [30], Story Cloze [40]
Sentence similarity	MSR Paraphrase Corpus [14], Quora Question Pairs [9], STS Benchmark [6]
Classification	Stanford Sentiment Treebank-2 [54], CoLA [65]

接下来就是不同任务的实验对比效果了，总结下来就是我们的最好，我这里只贴图就不分析了

Table 2: Experimental results on natural language inference tasks, comparing our model with current state-of-the-art methods. 5x indicates an ensemble of 5 models. All datasets use accuracy as the evaluation metric.

Method	MNLI-m	MNLI-mm	SNLI	SciTail	QNLI	RTE
ESIM + ELMo [44] (5x)	-	-	89.3	-	-	-
CAFE [58] (5x)	80.2	79.0	89.3	-	-	-
Stochastic Answer Network [35] (3x)	80.6	80.1	-	-	-	-
CAFE [58]	78.7	77.9	88.5	83.3		
GenSen [64]	71.4	71.3	-	-	82.3	59.2
Multi-task BiLSTM + Attn [64]	72.2	72.1	-	-	82.1	**61.7**
Finetuned Transformer LM (ours)	**82.1**	**81.4**	**89.9**	**88.3**	**88.1**	56.0

Table 3: Results on question answering and commonsense reasoning, comparing our model with current state-of-the-art methods.. 9x means an ensemble of 9 models.

Method	Story Cloze	RACE-m	RACE-h	RACE
val-LS-skip [55]	76.5	-	-	-
Hidden Coherence Model [7]	77.6	-	-	-
Dynamic Fusion Net [67] (9x)	-	55.6	49.4	51.2
BiAttention MRU [59] (9x)	-	60.2	50.3	53.3
Finetuned Transformer LM (ours)	**86.5**	**62.9**	**57.4**	**59.0**

Table 4: Semantic similarity and classification results, comparing our model with current state-of-the-art methods. All task evaluations in this table were done using the GLUE benchmark. (*mc*= Mathews correlation, *acc*=Accuracy, *pc*=Pearson correlation)

Method	**Classification**		**Semantic Similarity**			**GLUE**
	CoLA (mc)	SST2 (acc)	MRPC (F1)	STSB (pc)	QQP (F1)	
Sparse byte mLSTM [16]	-	**93.2**	-	-	-	-
TF-KLD [23]	-	-	**86.0**	-	-	-
ECNU (mixed ensemble) [60]	-	-	-	81.0	-	-
Single-task BiLSTM + ELMo + Attn [64]	35.0	90.2	80.2	55.5	66.1	64.8
Multi-task BiLSTM + ELMo + Attn [64]	18.9	91.6	83.5	72.8	63.3	68.9
Finetuned Transformer LM (ours)	**45.4**	91.3	82.3	**82.0**	**70.3**	**72.8**

之后作者做了几个方面的分析：一个是发现经过无监督预训练后不同层参数加载到监督学习的层上效果会变好，说明无监督预训练是有作用的。另一个就是无监督预训练后直接不微调进行 zero-shot 后不同任务的效果也还不错，作者主要使用生成的特性来进行指标评估，并根据任务对生成文本进行不同的限制，比如情感分析每个 prompt 加一个 very，并只取 positive 和 negative 两个词的概率，选最大的作为输出：

For CoLA (linguistic acceptability), examples are scored as the average token log-probability the generative model assigns and predictions are made by thresholding. For SST-2 (sentiment analysis), we append the token *very* to each example and restrict the language model's output distribution to only the words *positive* and *negative* and guess the token it assigns higher probability to as the prediction. For RACE (question answering), we pick the answer the generative model assigns the highest average token log-probability when conditioned on the document and question. For DPRD [46] (winograd schemas), we replace the definite pronoun with the two possible referrents and predict the resolution that the generative model assigns higher average token log-probability to the rest of the sequence after the substitution.

作者发现生成式是可以应对不同任务的：We observe the performance of these heuristics is stable and steadily increases over training suggesting that generative pretraining supports the learning of a wide variety of task relevant functionality，并同时发现对于越大的数据集，fine-tune 期间的辅助无监督预训练的效果越好：the trend suggests that larger datasets benefit from the auxiliary objective but smaller datasets do not，这位之后的 GPT 埋下了伏笔。

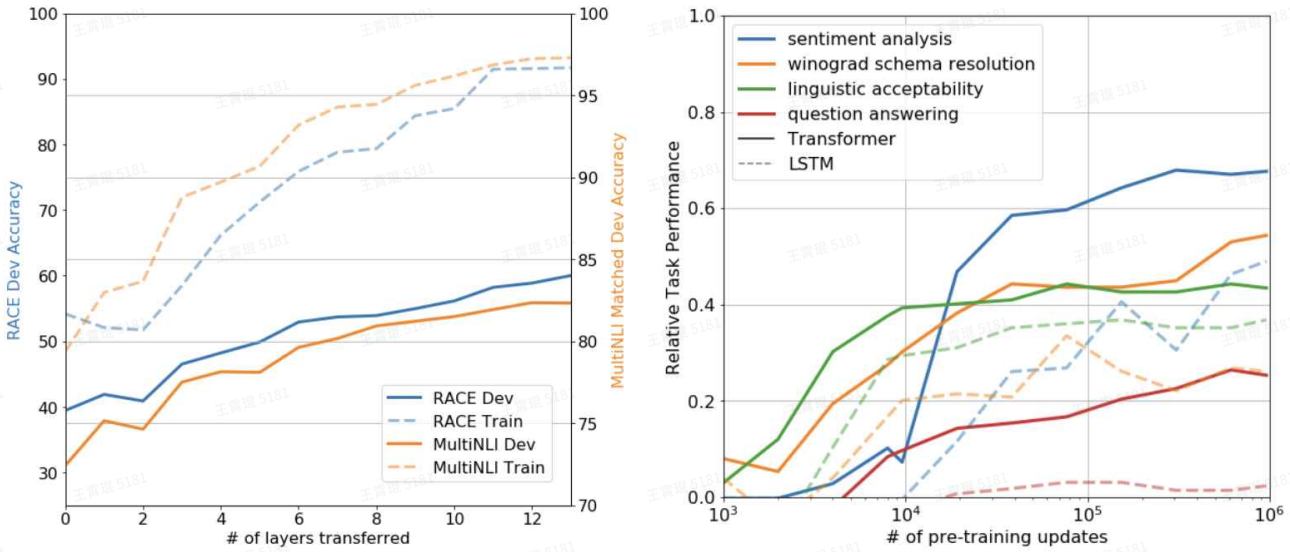

Figure 2: (**left**) Effect of transferring increasing number of layers from the pre-trained language model on RACE and MultiNLI. (**right**) Plot showing the evolution of zero-shot performance on different tasks as a function of LM pre-training updates. Performance per task is normalized between a random guess baseline and the current state-of-the-art with a single model.

Table 5: Analysis of various model ablations on different tasks. Avg. score is a unweighted average of all the results. (*mc*= Mathews correlation, *acc*=Accuracy, *pc*=Pearson correlation)

Method	Avg. Score	CoLA (mc)	SST2 (acc)	MRPC (F1)	STSB (pc)	QQP (F1)	MNLI (acc)	QNLI (acc)	RTE (acc)
Transformer w/ aux LM (full)	74.7	45.4	91.3	82.3	82.0	**70.3**	**81.8**	**88.1**	**56.0**
Transformer w/o pre-training	59.9	18.9	84.0	79.4	30.9	65.5	75.7	71.2	53.8
Transformer w/o aux LM	**75.0**	**47.9**	**92.0**	**84.9**	**83.2**	69.8	81.1	86.9	54.4
LSTM w/ aux LM	69.1	30.3	90.5	83.2	71.8	68.1	73.7	81.1	54.6