Improving Language Understanding by Generative Pre-Training
通过生成预训练提高语言理解

Alec Radford
OpenAI
alec@openai.com

Karthik Narasimhan
OpenAI
karthikn@openai.com

Tim Salimans
OpenAI
tim@openai.com

Ilya Sutskever
OpenAI
ilyasu@openai.com

Abstract

Natural language understanding comprises a wide range of diverse tasks such as textual entailment, question answering, semantic similarity assessment, and document classification. Although large unlabeled text corpora are abundant, labeled data for learning these specific tasks is scarce, making it challenging for discriminatively trained models to perform adequately. We demonstrate that large gains on these tasks can be realized by generative pre-training of a language model on a diverse corpus of unlabeled text, followed by discriminative fine-tuning on each specific task. In contrast to previous approaches, we make use of task-aware input transformations during fine-tuning to achieve effective transfer while requiring minimal changes to the model architecture. We demonstrate the effectiveness of our approach on a wide range of benchmarks for natural language understanding. Our general task-agnostic model outperforms discriminatively trained models that use architectures specifically crafted for each task, significantly improving upon the state of the art in 9 out of the 12 tasks studied. For instance, we achieve absolute improvements of 8.9% on commonsense reasoning (Stories Cloze Test), 5.7% on question answering (RACE), and 1.5% on textual entailment (MultiNLI).

自然语言理解包括一系列多样化的任务，如文本蕴含、问答、语义相似性评估和文档分类。尽管大量未标记的文本语料库丰富，但用于学习这些特定任务的标记数据稀缺，这使得通过判别训练的模型难以表现良好。我们证明，通过在多样化的未标记文本语料库上对语言模型进行生成预训练，然后在每个特定任务上进行判别微调，可以在这些任务上实现显著的提升。与之前的方法相比，我们在微调过程中利用任务感知的输入转换，以实现有效的迁移，同时对模型架构的改动最小。我们在自然语言理解的广泛基准测试中展示了我们方法的有效性。我们的通用任务无关模型在使用专门为每个任务设计的架构的判别训练模型中表现更佳，在研究的 12 个任务中，有 9 个任务显著提高了当前的最佳水平。例如，我们在常识推理 (故事完形填空测试) 上实现了 8.9% 的绝对提升，在问答 (RACE) 上提升了 5.7%，在文本蕴含 (MultiNLI) 上提升了 1.5%。

1 Introduction

The ability to learn effectively from raw text is crucial to alleviating the dependence on supervised learning in natural language processing (NLP). Most deep learning methods require substantial amounts of manually labeled data, which restricts their applicability in many domains that suffer from a dearth of annotated resources [61]. In these situations, models that can leverage linguistic information from unlabeled data provide a valuable alternative to gathering more annotation, which can be time-consuming and expensive. Further, even in cases where considerable supervision is available, learning good representations in an unsupervised fashion can provide a significant performance boost. The most compelling evidence for this so far has been the extensive use of pre-trained word embeddings [10, 39, 42] to improve performance on a range of NLP tasks [8, 11, 26, 45].

从原始文本中有效学习的能力对于减轻自然语言处理 (NLP) 中对监督学习的依赖至关重要。大多数深度学习方法需要大量手动标记的数据，这限制了它们在许多缺乏注释资源的领域的适用性 [61]。在这些情况下，能够利用未标记数据中的语言信息的模型为收集更多注释提供了有价值的替代方案，而收集注释可能既耗时又昂贵。此外，即使在有相当监督的情况下，以无监督的方式学习良好的表示也可以显著提升性能。迄今为止，最有力的证据是广泛使用预训练的词嵌入 [10, 39, 42] 来提高一系列 NLP 任务的性能 [8, 11, 26, 45]。

Leveraging more than word-level information from unlabeled text, however, is challenging for two main reasons. First, it is unclear what type of optimization objectives are most effective at learning text representations that are useful for transfer. Recent research has looked at various objectives such as language modeling [44], machine translation [38], and discourse coherence [22], with each method outperforming the others on different tasks. [1] Second, there is no consensus on the most effective way to transfer these learned representations to the target task. Existing techniques involve a combination of making task-specific changes to the model architecture [43, 44], using intricate learning schemes [21] and adding auxiliary learning objectives [50]. These uncertainties have made it difficult to develop effective semi-supervised learning approaches for language processing.

然而，从未标记文本中利用超过词级别的信息面临两个主要挑战。首先，目前尚不清楚哪些类型的优化目标在学习对迁移有用的文本表示方面最有效。最近的研究考察了各种目标，如语言建模 [44]、机器翻译 [38] 和话语连贯性 [22]，每种方法在不同任务上都优于其他方法 [1] 。其次，对于如何将这些学习到的表示有效地转移到目标任务上没有共识。现有技术涉及对模型架构进行任务特定的更改 [43, 44]、使用复杂的学习方案 [21] 和添加辅助学习目标 [50] 的组合。这些不确定性使得开发有效的半监督学习方法用于语言处理变得困难。

In this paper, we explore a semi-supervised approach for language understanding tasks using a combination of unsupervised pre-training and supervised fine-tuning. Our goal is to learn a universal representation that transfers with little adaptation to a wide range of tasks. We assume access to a large corpus of unlabeled text and several datasets with manually annotated training examples (target tasks). Our setup does not require these target tasks to be in the same domain as the unlabeled corpus. We employ a two-stage training procedure. First, we use a language modeling objective on the unlabeled data to learn the initial parameters of a neural network model. Subsequently, we adapt these parameters to a target task using the corresponding supervised objective.

在本文中，我们探索了一种用于语言理解任务的半监督方法，结合了无监督预训练和监督微调。我们的目标是学习一种通用表示，可以在适应性较小的情况下转移到广泛的任务上。我们假设可以访问大量未标记文本的语料库和几个带有手动注释训练示例 (目标任务) 的数据集。我们的设置不要求这些目标任务与未标记语料库在同一领域。我们采用两阶段的训练程序。首先，我们在未标记数据上使用语言建模目标来学习神经网络模型的初始参数。随后，我们使用相应的监督目标将这些参数适应到目标任务上。

For our model architecture, we use the Transformer [62], which has been shown to perform strongly on various tasks such as machine translation [62], document generation [34], and syntactic parsing [29]. This model choice provides us with a more structured memory for handling long-term dependencies in text, compared to alternatives like recurrent networks, resulting in robust transfer performance across diverse tasks. During transfer, we utilize task-specific input adaptations derived from traversal-style approaches [52], which process structured text input as a single contiguous sequence of tokens. As we demonstrate in our experiments, these adaptations enable us to fine-tune effectively with minimal changes to the architecture of the pre-trained model.

对于我们的模型架构，我们使用 Transformer [62]，该模型在机器翻译 [62]、文档生成 [34] 和句法解析 [29] 等各种任务上表现出色。与递归网络等替代方案相比，这种模型选择为处理文本中的长期依赖关系提供了更结构化的记忆，从而在不同任务之间实现了强大的迁移性能。在迁移过程中，我们利用从遍历式方法 [52] 中派生的任务特定输入适配，这些方法将结构化文本输入处理为一系列连续的标记。正如我们在实验中所展示的，这些适配使我们能够在对预训练模型的架构进行最小更改的情况下有效地进行微调。

We evaluate our approach on four types of language understanding tasks - natural language inference, question answering, semantic similarity, and text classification. Our general task-agnostic model outperforms discriminatively trained models that employ architectures specifically crafted for each task, significantly improving upon the state of the art in 9 out of the 12 tasks studied.

我们在四种语言理解任务上评估我们的方法——自然语言推理、问答、语义相似性和文本分类。我们的通用任务无关模型在 12 个研究任务中的 9 个任务上显著超越了采用专门为每个任务设计的架构的判别训练模型，显著提高了当前的技术水平。

[11] https://gluebenchmark.com/leaderboard

For instance, we achieve absolute improvements of 8.9% on commonsense reasoning (Stories Cloze Test) [40], 5.7% on question answering (RACE) [30], 1.5% on textual entailment (MultiNLI) [66] and 5.5% on the recently introduced GLUE multi-task benchmark [64]. We also analyzed zero-shot behaviors of the pre-trained model on four different settings and demonstrate that it acquires useful linguistic knowledge for downstream tasks.

例如，我们在常识推理 (故事填空测试) [40] 上实现了 8.9% 的绝对提升，在问答 (RACE) [30] 上提升了 5.7%，在文本蕴含 (MultiNLI) [66] 上提升了 1.5%，在最近引入的 GLUE 多任务基准 [64] 上提升了 5.5%。我们还分析了预训练模型在四种不同设置下的零样本行为，并证明它获得了对下游任务有用的语言知识。

2 Related Work 相关工作

Semi-supervised learning for NLP Our work broadly falls under the category of semi-supervised learning for natural language. This paradigm has attracted significant interest, with applications to tasks like sequence labeling [24, 33, 57] or text classification [41, 70]. The earliest approaches used unlabeled data to compute word-level or phrase-level statistics, which were then used as features in a supervised model [33]. Over the last few years, researchers have demonstrated the benefits of using word embeddings [11, 39, 42], which are trained on unlabeled corpora, to improve performance on a variety of tasks [8, 11, 26, 45]. These approaches, however, mainly transfer word-level information, whereas we aim to capture higher-level semantics.

半监督学习用于自然语言处理我们的工作大致属于自然语言的半监督学习范畴。这个范式引起了广泛的关注，应用于序列标注 [24, 33, 57] 或文本分类 [41, 70] 等任务。最早的方法使用未标记的数据来计算词级或短语级统计数据，然后将其作为特征用于监督模型 [33]。在过去几年中，研究人员展示了使用在未标记语料库上训练的词嵌入 [11, 39, 42] 的好处，以提高各种任务的性能 [8, 11, 26, 45]。然而，这些方法主要传递词级信息，而我们的目标是捕捉更高层次的语义。

Recent approaches have investigated learning and utilizing more than word-level semantics from unlabeled data. Phrase-level or sentence-level embeddings, which can be trained using an unlabeled corpus, have been used to encode text into suitable vector representations for various target tasks [28, 32, 1, 36, 22, 12, 56, 31].

最近的方法研究了从未标记数据中学习和利用超过词级语义的能力。可以使用未标记语料库训练的短语级或句子级嵌入，已被用于将文本编码为适合各种目标任务的向量表示 [28, 32, 1, 36, 22, 12, 56, 31]。

Unsupervised pre-training Unsupervised pre-training is a special case of semi-supervised learning where the goal is to find a good initialization point instead of modifying the supervised learning objective. Early works explored the use of the technique in image classification [20, 49, 63] and regression tasks [3]. Subsequent research [15] demonstrated that pre-training acts as a regularization scheme, enabling better generalization in deep neural networks. In recent work, the method has been used to help train deep neural networks on various tasks like image classification [69], speech recognition [68], entity disambiguation [17] and machine translation [48].

无监督预训练无监督预训练是半监督学习的一种特殊情况，其目标是找到一个良好的初始化点，而不是修改监督学习目标。早期的工作探索了该技术在图像分类 [20, 49, 63] 和回归任务 [3] 中的应用。后续研究 [15] 证明预训练作为一种正则化方案，能够在深度神经网络中实现更好的泛化。在最近的工作中，该方法已被用于帮助训练深度神经网络，处理各种任务，如图像分类 [69]、语音识别 [68]、实体消歧 [17] 和机器翻译 [48]。

The closest line of work to ours involves pre-training a neural network using a language modeling objective and then fine-tuning it on a target task with supervision. Dai et al. [13] and Howard and Ruder [21] follow this method to improve text classification.

与我们工作最接近的研究涉及使用语言建模目标对神经网络进行预训练，然后在目标任务上进行监督微调。Dai 等 [13] 和 Howard 与 Ruder [21] 采用这种方法来改善文本分类。

However, although the pre-training phase helps capture some linguistic information, their usage of LSTM models restricts their prediction ability to a short range. In contrast, our choice of transformer networks allows us to capture longer-range linguistic structure, as demonstrated in our experiments. Further, we also demonstrate the effectiveness of our model on a wider range of tasks including natural language inference, paraphrase detection and story completion. Other approaches [43, 44, 38] use hidden representations from a pre-trained language or machine translation model as auxiliary features while training a supervised model on the target task. This involves a substantial amount of new parameters for each separate target task, whereas we require minimal changes to our model architecture during transfer.

> 然而，尽管预训练阶段有助于捕捉一些语言信息，但他们使用 LSTM 模型限制了其预测能力在短范围内。相比之下，我们选择的 transformer 网络使我们能够捕捉更长范围的语言结构，正如我们的实验所示。此外，我们还展示了我们的模型在更广泛的任务上的有效性，包括自然语言推理、释义检测和故事完成。其他方法 [43, 44, 38] 使用来自预训练语言或机器翻译模型的隐藏表示作为辅助特征，同时在目标任务上训练监督模型。这涉及到每个单独目标任务的大量新参数，而我们在迁移过程中对模型架构的更改要求最小。

Auxiliary training objectives Adding auxiliary unsupervised training objectives is an alternative form of semi-supervised learning. Early work by Collobert and Weston [10] used a wide variety of auxiliary NLP tasks such as POS tagging, chunking, named entity recognition, and language modeling to improve semantic role labeling. More recently, Rei [50] added an auxiliary language modeling objective to their target task objective and demonstrated performance gains on sequence labeling tasks. Our experiments also use an auxiliary objective, but as we show, unsupervised pre-training already learns several linguistic aspects relevant to target tasks.

> 辅助训练目标添加辅助无监督训练目标是半监督学习的另一种形式。Collobert 和 Weston [10] 的早期工作使用了多种辅助自然语言处理任务，如词性标注、分块、命名实体识别和语言建模，以改善语义角色标注。最近，Rei [50] 在其目标任务目标中添加了辅助语言建模目标，并在序列标注任务上展示了性能提升。我们的实验也使用了辅助目标，但正如我们所示，无监督预训练已经学习了与目标任务相关的多个语言方面。

3 Framework **框架**

Our training procedure consists of two stages. The first stage is learning a high-capacity language model on a large corpus of text. This is followed by a fine-tuning stage, where we adapt the model to a discriminative task with labeled data.

> 我们的训练过程分为两个阶段。第一阶段是在大规模文本语料库上学习一个高容量的语言模型。接下来是微调阶段，我们将模型适应于带标签的数据的判别任务。

3.1 Unsupervised pre-training **无监督预训练**

Given an unsupervised corpus of tokens $\mathcal{U}=\{u_1,\ldots,u_n\}$, we use a standard language modeling objective to maximize the following likelihood:

> 给定一个无监督的标记语料库 $\mathcal{U}=\{u_1,\ldots,u_n\}$ ，我们使用标准的语言建模目标来最大化以下似然性:

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

where k is the size of the context window, and the conditional probability P is modeled using a neural network with parameters Θ . These parameters are trained using stochastic gradient descent [51].

> 其中 k 是上下文窗口的大小，条件概率 P 使用具有参数 Θ 的神经网络进行建模。这些参数使用随机梯度下降 [51] 进行训练。

In our experiments, we use a multi-layer Transformer decoder [34] for the language model, which is a variant of the transformer [62]. This model applies a multi-headed self-attention operation over the input context tokens followed by position-wise feedforward layers to produce an output distribution over target tokens:

> 在我们的实验中，我们使用多层 Transformer 解码器 [34] 作为语言模型，它是 transformer [62] 的一种变体。该模型对输入上下文标记应用多头自注意力操作，然后通过逐位置前馈层生成目标标记的输出分布:

$$h_0 = UW_e + W_p$$

$$h_l = \text{ transformer\_block } (h_{l-1}) \forall i \in [1, n] \tag{2}$$

$$P(u) = \text{softmax}\left(h_n W_e^T\right)$$

where $U = (u_{-k}, \ldots, u_{-1})$ is the context vector of tokens, n is the number of layers, W_e is the token embedding matrix, and W_p is the position embedding matrix.

> 其中 $U = (u_{-k}, \ldots, u_{-1})$ 是标记的上下文向量，n 是层数，W_e 是标记嵌入矩阵，W_p 是位置嵌入矩阵。

3.2 Supervised fine-tuning **监督微调**

After training the model with the objective in Eq. 1, we adapt the parameters to the supervised target task. We assume a labeled dataset $\mathcal{C}$, where each instance consists of a sequence of input tokens, $x^1, \ldots, x^m$, along with a label y . The inputs are passed through our pre-trained model to obtain the final transformer block’ s activation h_l^m , which is then fed into an added linear output layer with parameters W_y to predict y :

> 在用公式 1 中的目标训练模型后，我们将参数调整到监督目标任务。我们假设有一个标记数据集 $\mathcal{C}$ ，其中每个实例由一系列输入标记 $x^1, \ldots, x^m$ 和一个标签 y 组成。输入通过我们预训练的模型传递，以获得最终 transformer 块的激活 h_l^m ，然后将其输入到一个附加的线性输出层，该层的参数为 W_y ，以预测 y :

$$P\left(y \mid x^1, \ldots, x^m\right) = \text{softmax}\left(h_l^m W_y\right). \tag{3}$$

This gives us the following objective to maximize:

> 这给我们提供了以下需要最大化的目标:

$$L_2(\mathcal{C}) = \sum_{(x,y)} \log P\left(y \mid x^1, \ldots, x^m\right). \tag{4}$$

We additionally found that including language modeling as an auxiliary objective to the fine-tuning helped learning by (a) improving generalization of the supervised model, and (b) accelerating convergence. This is in line with prior work [50, 43], who also observed improved performance with such an auxiliary objective. Specifically, we optimize the following objective (with weight λ):

> 我们还发现，将语言建模作为微调的辅助目标有助于学习，具体表现为 (a) 改善监督模型的泛化能力，以及 (b) 加速收敛。这与之前的研究 [50, 43] 一致，他们也观察到使用这样的辅助目标可以提高性能。具体而言，我们优化以下目标 (权重为 λ):

$$L_3(\mathcal{C}) = L_2(\mathcal{C}) + \lambda * L_1(\mathcal{C}) \tag{5}$$

Overall, the only extra parameters we require during fine-tuning are W_y , and embeddings for delimiter tokens (described below in Section 3.3).

总体而言，我们在微调过程中所需的唯一额外参数是 W_y ，以及分隔符标记的嵌入 (在第 3.3 节中描述)。

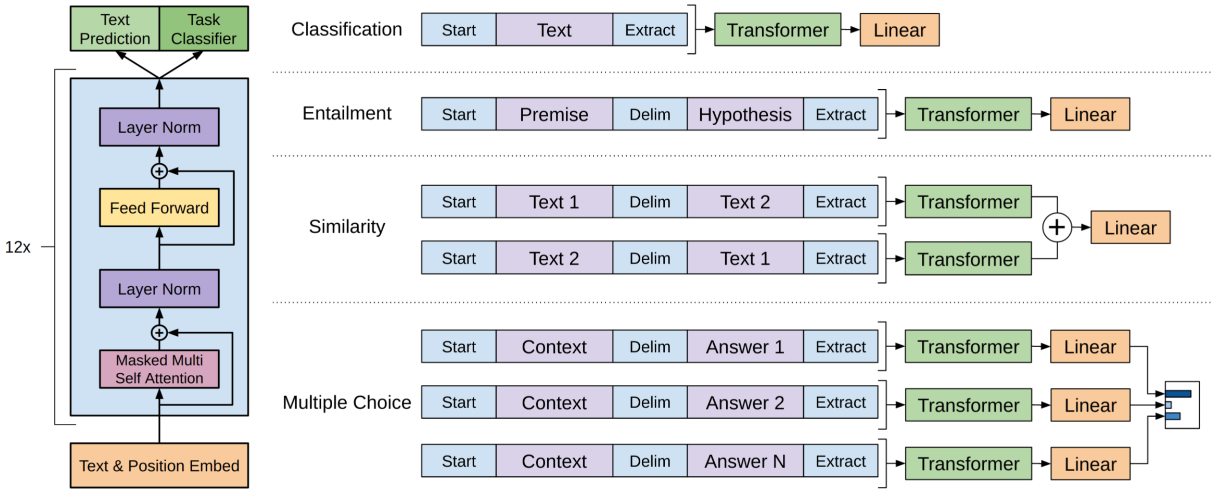

Figure 1: (left) Transformer architecture and training objectives used in this work. (right) Input transformations for fine-tuning on different tasks. We convert all structured inputs into token sequences to be processed by our pre-trained model, followed by a linear+softmax layer.
图 1: (左) 本研究中使用的 transformer 架构和训练目标。(右) 用于在不同任务上微调的输入转换。我们将所有结构化输入转换为 token 序列，以便由我们预训练的模型处理，随后是一个线性 +softmax 层。

3.3 Task-specific input transformations **任务特定输入转换**

For some tasks, like text classification, we can directly fine-tune our model as described above. Certain other tasks, like question answering or textual entailment, have structured inputs such as ordered sentence pairs, or triplets of document, question, and answers. Since our pre-trained model was trained on contiguous sequences of text, we require some modifications to apply it to these tasks. Previous work proposed learning task specific architectures on top of transferred representations [44]. Such an approach re-introduces a significant amount of task-specific customization and does not use transfer learning for these additional architectural components. Instead, we use a traversal-style approach [52], where we convert structured inputs into an ordered sequence that our pre-trained model can process. These input transformations allow us to avoid making extensive changes to the architecture across tasks. We provide a brief description of these input transformations below and Figure 1 provides a visual illustration. All transformations include adding randomly initialized start and end tokens $(\langle s\rangle, \langle e\rangle)$.

对于某些任务，如文本分类，我们可以直接对我们的模型进行微调，如上所述。其他某些任务，如问答或文本蕴含，具有结构化输入，例如有序的句子对，或文档、问题和答案的三元组。由于我们的预训练模型是在连续的文本序列上训练的，因此我们需要进行一些修改以将其应用于这些任务。之前的工作提出了在转移表示之上学习任务特定架构的方案 [44]。这种方法重新引入了大量的任务特定定制，并且没有对这些额外的架构组件使用迁移学习。相反，我们使用了一种遍历式的方法 [52]，将结构化输入转换为我们的预训练模型可以处理的有序序列。这些输入转换使我们能够避免在任务之间进行广泛的架构更改。我们在下面简要描述这些输入转换，图 1 提供了可视化说明。所有转换都包括添加随机初始化的开始和结束标记 $(\langle s\rangle, \langle e\rangle)$ 。

Textual entailment For entailment tasks, we concatenate the premise p and hypothesis h token sequences, with a delimiter token ($) in between.

文本蕴含对于蕴含任务，我们将前提 p 和假设 h token 序列连接起来，中间用分隔符 token($) 分隔。

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.

相似性对于相似性任务，比较的两个句子没有固有的顺序。为了反映这一点，我们修改输入序列，使其包含两种可能的句子顺序 (中间用分隔符分隔)，并独立处理每个序列以生成两个序列表示 h_l^m，然后在输入线性输出层之前逐元素相加。

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.

问答和常识推理对于这些任务，我们给定一个上下文文档 z 、一个问题 q 和一组可能的答案 $\{a_k\}$ 。我们将文档上下文和问题与每个可能的答案连接起来，中间添加一个分隔符 token 以得到 $[z; q; \$; a_k]$ 。这些序列中的每一个都与我们的模型独立处理，然后通过 softmax 层进行归一化，以生成对可能答案的输出分布。

4 Experiments 实验

4.1 Setup 设置

Unsupervised pre-training We use the BooksCorpus dataset [71] for training the language model. It contains over 7,000 unique unpublished books from a variety of genres including Adventure, Fantasy, and Romance. Crucially, it contains long stretches of contiguous text, which allows the generative model to learn to condition on long-range information. An alternative dataset, the 1 B Word Benchmark, which is used by a similar approach, ELMo [44], is approximately the same size but is shuffled at a sentence level - destroying long-range structure. Our language model achieves a very low token level perplexity of 18.4 on this corpus.

无监督预训练我们使用 BooksCorpus 数据集 [71] 来训练语言模型。它包含超过 7,000 本独特的未出版书籍，涵盖冒险、幻想和浪漫等多种类型。至关重要的是，它包含长篇连续文本，这使得生成模型能够学习对长距离信息进行条件化。一个替代数据集 1 B Word Benchmark，采用类似的方法 ELMo [44]，大小大致相同，但在句子级别进行了洗牌 - 破坏了长距离结构。我们的语言模型在该语料库上实现了非常低的标记级别困惑度 18.4。

Table 1: A list of the different tasks and datasets used in our experiments.
我们实验中使用的不同任务和数据集的列表。

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]

Model specifications Our model largely follows the original transformer work [62]. We trained a 12-layer decoder-only transformer with masked self-attention heads (768 dimensional states and 12 attention heads). For the position-wise feed-forward networks, we used 3072 dimensional inner states. We used the Adam optimization scheme [27] with a max learning rate of 2.5e-4. The learning rate was increased linearly from zero over the first 2000 updates and annealed to 0 using a cosine schedule.

模型规格 我们的模型在很大程度上遵循了原始的 transformer 工作 [62]。我们训练了一个 12 层的仅解码器 transformer，具有掩蔽自注意力头 (768 维状态和 12 个注意力头)。对于逐位置前馈网络，我们使用了 3072 维的内部状态。我们使用了 Adam 优化方案 [27]，最大学习率为 2.5e-4。学习率在前 2000 次更新中从零线性增加，并使用余弦调度降至 0。

We train for 100 epochs on minibatches of 64 randomly sampled, contiguous sequences of 512 tokens. Since layernorm [2] is used extensively throughout the model, a simple weight initialization of $N(0, 0.02)$ was sufficient. We used a bytepair encoding (BPE) vocabulary with 40,000 merges [53] and residual, embedding, and attention dropouts with a rate of 0.1 for regularization. We also employed a modified version of L2 regularization proposed in [37], with $w = 0.01$ on all non bias or gain weights. For the activation function, we used the Gaussian Error Linear Unit (GELU) [18]. We used learned position embeddings instead of the sinusoidal version proposed in the original work. We use the ftfy library [2] to clean the raw text in BooksCorpus, standardize some punctuation and whitespace, and use the spaCy tokenizer. [3]

我们在 64 个随机采样的连续 512 个标记的序列的小批量上训练了 100 个周期。由于 layernorm [2] 在整个模型中被广泛使用，因此简单的权重初始化 $N(0, 0.02)$ 就足够了。我们使用了 40,000 次合并的字节对编码 (BPE) 词汇 [53]，并对残差、嵌入和注意力进行了 0.1 的丢弃率以进行正则化。我们还采用了 [37] 中提出的修改版 L2 正则化，对所有非偏置或增益权重应用 $w = 0.01$ 。对于激活函数，我们使用了高斯误差线性单元 (GELU) [18]。我们使用了学习到的位置嵌入，而不是原始工作中提出的正弦版本。我们使用 ftfy 库 [2] 清理 BooksCorpus 中的原始文本，标准化一些标点符号和空格，并使用 spaCy 分词器。[3]

Fine-tuning details Unless specified, we reuse the hyperparameter settings from unsupervised pre-training. We add dropout to the classifier with a rate of 0.1 . For most tasks, we use a learning rate of 6.25e-5 and a batchsize of 32. Our model finetunes quickly and 3 epochs of training was sufficient for most cases. We use a linear learning rate decay schedule with warmup over 0.2% of training. λ was set to 0.5 .

微调细节 除非另有说明，我们重用了无监督预训练的超参数设置。我们在分类器中添加了 0.1 的丢弃率。对于大多数任务，我们使用 6.25e-5 的学习率和 32 的批量大小。我们的模型微调速度很快，3 个周期的训练对于大多数情况来说是足够的。我们使用线性学习率衰减调度，并在 0.2% 的训练上进行预热。λ 被设置为 0.5。

4.2 Supervised fine-tuning **监督微调**

We perform experiments on a variety of supervised tasks including natural language inference, question answering, semantic similarity, and text classification. Some of these tasks are available as part of the recently released GLUE multi-task benchmark [64], which we make use of. Figure 1 provides an overview of all the tasks and datasets.

我们在多种监督任务上进行实验，包括自然语言推理、问答、语义相似性和文本分类。其中一些任务作为最近发布的 GLUE 多任务基准的一部分可用 [64]，我们对此进行了利用。图 1 提供了所有任务和数据集的概述。

Natural Language Inference The task of natural language inference (NLI), also known as recognizing textual entailment, involves reading a pair of sentences and judging the relationship between them from one of entailment, contradiction or neutral. Although there has been a lot of recent interest [58, 35, 44], the task remains challenging due to the presence of a wide variety of phenomena like lexical entailment, coreference, and lexical and syntactic ambiguity. We evaluate on five datasets with diverse sources, including image captions (SNLI), transcribed speech, popular fiction, and government reports (MNLI), Wikipedia articles (QNLI), science exams (SciTail) or news articles (RTE).

自然语言推理自然语言推理 (NLI) 任务，也称为识别文本蕴含，涉及阅读一对句子并判断它们之间的关系，可能是蕴含、矛盾或中立。尽管最近对此有很多兴趣 [58, 35, 44]，但由于存在词汇蕴含、共指以及词汇和句法歧义等多种现象，该任务仍然具有挑战性。我们在五个来源多样的数据集上进行评估，包括图像标题 (SNLI)、转录语音、流行小说和政府报告 (MNLI)、维基百科文章 (QNLI)、科学考试 (SciTail) 或新闻文章 (RTE)。

[22] https://ftfy.readthedocs.io/en/latest/

[33] https://spacy.io/

Table 2 details various results on the different NLI tasks for our model and previous state-of-the-art approaches. Our method significantly outperforms the baselines on four of the five datasets, achieving absolute improvements of upto 1.5% on MNLI, 5% on SciTail, 5.8% on QNLI and 0.6% on SNLI over the previous best results. This demonstrates our model's ability to better reason over multiple sentences, and handle aspects of linguistic ambiguity. On RTE, one of the smaller datasets we evaluate on (2490 examples), we achieve an accuracy of 56%, which is below the 61.7% reported by a multi-task biLSTM model. Given the strong performance of our approach on larger NLI datasets, it is likely our model will benefit from multi-task training as well but we have not explored this currently.

表 2 详细列出了我们模型和之前的最先进方法在不同 NLI 任务上的各种结果。我们的方法在五个数据集中的四个上显著优于基线，在 MNLI 上实现了最高 1.5% 的绝对提升，在 SciTail 上为 5%，在 QNLI 上为 5.8%，在 SNLI 上为 0.6%，超越了之前的最佳结果。这证明了我们模型在多个句子上更好地推理的能力，并处理语言歧义的各个方面。在我们评估的较小数据集 RTE(2490 个示例) 上，我们达到了 56% 的准确率，低于多任务 biLSTM 模型报告的 61.7%。考虑到我们的方法在较大 NLI 数据集上的强劲表现，我们的模型可能也会从多任务训练中受益，但我们目前尚未对此进行探索。

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.
自然语言推理任务的实验结果，将我们的模型与当前最先进的方法进行比较。5x 表示 5 个模型的集成。所有数据集使用准确率作为评估指标。

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.
表 3: 关于问答和常识推理的结果，将我们的模型与当前最先进的方法进行比较。9x 表示 9 个模型的集成。

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

Question answering and commonsense reasoning Another task that requires aspects of single and multi-sentence reasoning is question answering. We use the recently released RACE dataset [30], consisting of English passages with associated questions from middle and high school exams. This corpus has been shown to contain more reasoning type questions that other datasets like CNN [19] or SQuaD [47], providing the perfect evaluation for our model which is trained to handle long-range contexts.

问答和常识推理 另一个需要单句和多句推理方面的任务是问答。我们使用最近发布的 RACE 数据集 [30]，该数据集包含来自中学和高中考试的英文段落及相关问题。该语料库已被证明包含比其他数据集 (如 CNN [19] 或 SQuaD [47]) 更多的推理类型问题，为我们的模型提供了完美的评估，该模型经过训练以处理长距离上下文。

In addition, we evaluate on the Story Cloze Test [40], which involves selecting the correct ending to multi-sentence stories from two options. On these tasks, our model again outperforms the previous best results by significant margins - up to 8.9% on Story Cloze, and 5.7% overall on RACE. This demonstrates the ability of our model to handle long-range contexts effectively.

此外，我们还在故事闭合测试 [40] 上进行评估，该测试涉及从两个选项中选择多句故事的正确结尾。在这些任务中，我们的模型再次以显著的优势超越了之前的最佳结果——在故事闭合测试中高达 8.9%，在 RACE 上整体提高了 5.7%。这证明了我们的模型有效处理长距离上下文的能力。

Semantic Similarity Semantic similarity (or paraphrase detection) tasks involve predicting whether two sentences are semantically equivalent or not. The challenges lie in recognizing rephrasing of concepts, understanding negation, and handling syntactic ambiguity. We use three datasets for this task - the Microsoft Paraphrase corpus (MRPC) [14] (collected from news sources), the Quora Question Pairs (QQP) dataset [9], and the Semantic Textual Similarity benchmark (STS-B) [6]. We obtain state-of-the-art results on two of the three semantic similarity tasks (Table 4) with a 1 point absolute gain on STS-B. The performance delta on QQP is significant, with a 4.2% absolute improvement over Single-task BiLSTM + ELMo + Attn.

语义相似性语义相似性 (或释义检测) 任务涉及预测两个句子是否在语义上等价。挑战在于识别概念的重新表述、理解否定以及处理句法歧义。我们为此任务使用三个数据集——微软释义语料库 (MRPC) [14](收集自新闻来源)、Quora 问题对 (QQP) 数据集 [9] 和语义文本相似性基准 (STS-B) [6]。我们在三个语义相似性任务中的两个任务上获得了最先进的结果 (表 4)，在 STS-B 上实现了 1 分的绝对提升。QQP 的性能差异显著，相较于单任务 BiLSTM + ELMo + Attn 提升了 4.2% 的绝对值。

Classification Finally, we also evaluate on two different text classification tasks. The Corpus of Linguistic Acceptability (CoLA) [65] contains expert judgements on whether a sentence is grammatical or not, and tests the innate linguistic bias of trained models. The Stanford Sentiment Treebank (SST-2) [54], on the other hand, is a standard binary classification task. Our model obtains an score of 45.4 on CoLA, which is an especially big jump over the previous best result of 35.0, showcasing the innate linguistic bias learned by our model. The model also achieves 91.3% accuracy on SST-2, which is competitive with the state-of-the-art results. We also achieve an overall score of 72.8 on the GLUE benchmark, which is significantly better than the previous best of 68.9.

分类最后，我们还在两个不同的文本分类任务上进行评估。语言可接受性语料库 (CoLA) [65] 包含专家对句子是否语法正确的判断，并测试训练模型的内在语言偏见。斯坦福情感树库 (SST-2) [54] 则是一个标准的二分类任务。我们的模型在 CoLA 上获得了 45.4 的分数，这比之前的最佳结果 35.0 有了特别大的跃升，展示了我们的模型所学习的内在语言偏见。该模型在 SST-2 上也达到了 91.3% 的准确率，与最先进的结果相当。我们在 GLUE 基准上也取得了 72.8 的整体得分，显著优于之前的最佳 68.9。

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)
语义相似性和分类结果，将我们的模型与当前最先进的方法进行比较。此表中的所有任务评估均使用 GLUE 基准进行。(mc= Mathews 相关性，*acc* = 准确率，*pc* = Pearson 相关性)

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

Overall, our approach achieves new state-of-the-art results in 9 out of the 12 datasets we evaluate on, outperforming ensembles in many cases. Our results also indicate that our approach works well across datasets of different sizes, from smaller datasets such as STS-B (5.7k training examples) - to the largest one - SNLI (≈ 550k training examples).

总体而言，我们的方法在我们评估的 12 个数据集中有 9 个达到了新的最先进结果，在许多情况下超越了集成方法。我们的结果还表明，我们的方法在不同大小的数据集上表现良好，从较小的数据集如 STS-B(5.7k 训练样本) 到最大的 SNLI(≈ 550k 训练样本)。

5 Analysis 分析

Impact of number of layers transferred We observed the impact of transferring a variable number of layers from unsupervised pre-training to the supervised target task. Figure 2(left) illustrates the performance of our approach on MultiNLI and RACE as a function of the number of layers transferred. We observe the standard result that transferring embeddings improves performance and that each transformer layer provides further benefits up to 9% for full transfer on MultiNLI. This indicates that each layer in the pre-trained model contains useful functionality for solving target tasks.

转移层数的影响我们观察了从无监督预训练到监督目标任务转移可变层数的影响。图 2(左) 展示了我们的方法在 MultiNLI 和 RACE 上的表现与转移层数的关系。我们观察到标准结果，即转移嵌入可以提高性能，并且每个 transformer 层在 MultiNLI 上的完全转移提供了高达 9% 的额外收益。这表明预训练模型中的每一层都包含解决目标任务的有用功能。

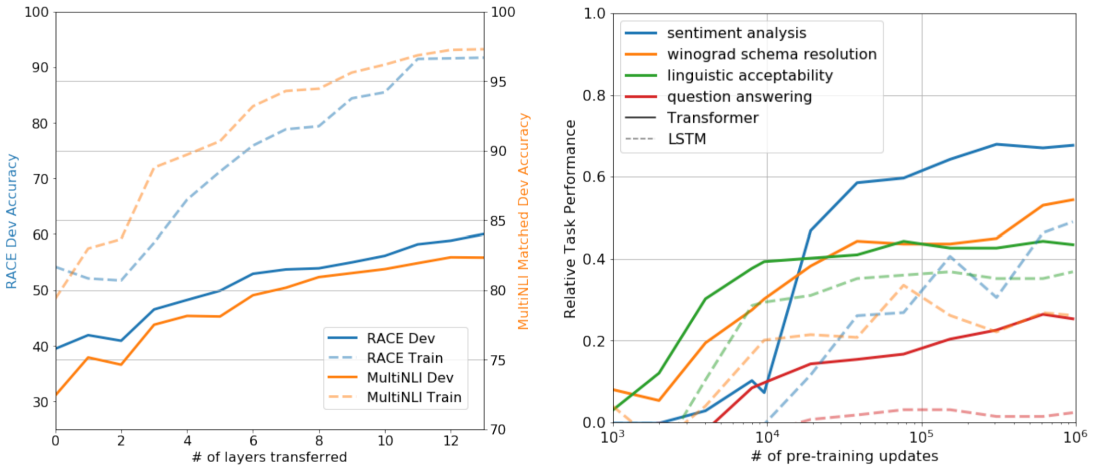

Figure 2: (left) Effect of transferring increasing number of layers from the pre-trained language model on RACE and MultiNLI. (right) Plot showing the evolution of zero-shot performance on different tasks as a function of LM pre-training updates. Performance per task is normalized between a random guess baseline and the current state-of-the-art with a single model.
(左) 从预训练语言模型转移增加层数对 RACE 和 MultiNLI 的影响。(右) 图示显示了不同任务的零-shot 性能随语言模型预训练更新的演变。每个任务的性能在随机猜测基线和当前最先进的单一模型之间进行了归一化。

Zero-shot Behaviors We'd like to better understand why language model pre-training of transformers is effective. A hypothesis is that the underlying generative model learns to perform many of the tasks we evaluate on in order to improve its language modeling capability and that the more structured attentional memory of the transformer assists in transfer compared to LSTMs. We designed a series of heuristic solutions that use the underlying generative model to perform tasks without supervised finetuning.

零样本行为我们希望更好地理解为什么语言模型的 transformer 预训练是有效的。一个假设是，基础生成模型学习执行我们评估的许多任务，以提高其语言建模能力，并且与 LSTM 相比，transformer 的更结构化的注意力记忆有助于转移。我们设计了一系列启发式解决方案，利用基础生成模型在没有监督微调的情况下执行任务。我们在图 2(右) 中可视化了这些启发式解决方案在生成预训练过程中的有效性。

We visualize the effectiveness of these heuristic solutions over the course of generative pre-training in Fig 2(right). 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. We also observe the LSTM exhibits higher variance in its zero-shot performance suggesting that the inductive bias of the Transformer architecture assists in transfer.

我们观察到这些启发式方法的性能稳定，并在训练过程中稳步提高，这表明生成预训练支持学习各种任务相关的功能。我们还观察到 LSTM 在其零-shot 性能中表现出更高的方差，这表明 Transformer 架构的归纳偏差有助于转移。

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) 表 5: 对不同任务的各种模型消融分析。平均分是所有结果的无权平均。(mc = Mathews 相关性，acc = 准确率，pc = Pearson 相关性)

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

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.

对于 CoLA(语言可接受性)，示例的评分为生成模型分配的平均标记对数概率，预测通过阈值进行。对于 SST-2(情感分析)，我们在每个示例中附加标记 very，并将语言模型的输出分布限制为仅包含单词 positive 和 negative，猜测其分配更高概率的标记作为预测。对于 RACE(问答)，我们选择生成模型在给定文档和问题时分配的最高平均标记对数概率的答案。对于 DPRD [46](Winograd 语法)，我们用两个可能的指代词替换定冠词，并预测生成模型在替换后分配给序列其余部分的更高平均标记对数概率的解析。

Ablation studies We perform three different ablation studies (Table 5). First, we examine the performance of our method without the auxiliary LM objective during fine-tuning. We observe that the auxiliary objective helps on the NLI tasks and QQP. Overall, the trend suggests that larger datasets benefit from the auxiliary objective but smaller datasets do not. Second, we analyze the effect of the Transformer by comparing it with a single layer 2048 unit LSTM using the same framework. We observe a 5.6 average score drop when using the LSTM instead of the Transformer. The LSTM only outperforms the Transformer on one dataset - MRPC. Finally, we also compare with our transformer architecture directly trained on supervised target tasks, without pre-training. We observe that the lack of pre-training hurts performance across all the tasks, resulting in a 14.8% decrease compared to our full model.

消融研究 我们进行了三项不同的消融研究 (表 5)。首先，我们检查在微调过程中不使用辅助 LM 目标时我们方法的性能。我们观察到辅助目标对 NLI 任务和 QQP 有帮助。总体趋势表明，较大的数据集受益于辅助目标，但较小的数据集则没有。其次，我们通过将 Transformer 与使用相同框架的单层 2048 单元 LSTM 进行比较，分析 Transformer 的效果。我们观察到使用 LSTM 而不是 Transformer 时平均得分下降了 5.6。LSTM 仅在一个数据集 - MRPC 上优于 Transformer。最后，我们还与直接在监督目标任务上训练的 Transformer 架构进行比较，而没有预训练。我们观察到缺乏预训练对所有任务的性能造成了损害，导致与我们的完整模型相比出现了 14.8% 的下降。

6 Conclusion **结论**

We introduced a framework for achieving strong natural language understanding with a single task-agnostic model through generative pre-training and discriminative fine-tuning. By pre-training on a diverse corpus with long stretches of contiguous text our model acquires significant world knowledge and ability to process long-range dependencies which are then successfully transferred to solving discriminative tasks such as question answering, semantic similarity assessment, entailment determination, and text classification, improving the state of the art on 9 of the 12 datasets we study. Using unsupervised (pre-)training to boost performance on discriminative tasks has long been an important goal of Machine Learning research. Our work suggests that achieving significant performance gains is indeed possible, and offers hints as to what models (Transformers) and data sets (text with long range dependencies) work best with this approach. We hope that this will help enable new research into unsupervised learning, for both natural language understanding and other domains, further improving our understanding of how and when unsupervised learning works.

我们提出了一个框架，通过生成预训练和判别微调，实现强大的自然语言理解，使用单一的任务无关模型。通过在包含长段连续文本的多样语料库上进行预训练，我们的模型获得了显著的世界知识和处理长程依赖的能力，这些能力随后成功转移到解决判别任务，如问答、语义相似性评估、蕴涵判断和文本分类，提高了我们研究的 12 个数据集中的 9 个的最新水平。利用无监督 (预) 训练来提升判别任务的性能长期以来一直是机器学习研究的重要目标。我们的工作表明，确实可以实现显著的性能提升，并提供了关于哪些模型 (Transformer) 和数据集 (具有长程依赖的文本) 最适合这种方法的线索。我们希望这将有助于推动对无监督学习的新研究，无论是针对自然语言理解还是其他领域，进一步提高我们对无监督学习何时以及如何有效的理解。