## 一、Model

### 1）Autoregressive Models

**I）Original GPT**

[Improving Language Understanding by Generative Pre-Training](https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf)

普通的transformer模型，基于Book Corpus训练；

**II）GPT-2**

[Language Models are Unsupervised Multitask Learners](https://d4mucfpksywv.cloudfront.net/better-language-models/language_models_are_unsupervised_multitask_learners.pdf)

普通的transformer模型，模型参数更大，在GPT基础上，多使用WebText训练；

**III）CTRL**

[CTRL: A CONDITIONAL TRANSFORMER LANGUAGE MODEL FOR CONTROLLABLE GENERATION](https://arxiv.org/pdf/1909.05858.pdf)

引入先验信息，可以控制不同风格文本的生成，后续的模型有PPLM等；

**IV）Transformer-XL**

[Transformer-XL: Attentive Language Models Beyond a Fixed-Length Context](https://arxiv.org/pdf/1901.02860.pdf)

引入recurrent segment用于编码更长的文本；此外引入相对位置编码，相比最原始的绝对位置编码，对建模有帮助；

**V）Reformer**

[REFORMER: THE EFFICIENT TRANSFORMER](https://arxiv.org/pdf/2001.04451.pdf)

引入Axial位置编码，使得长文本情况下，压缩位置编码矩阵；引入LSH Attention使得Attention的复杂度从O(n\*n)到O(n\*logn)；此外，利用ResNet特殊结构，引入reversible transformer，压缩显存大小；

**VI）XLNet**

[XLNet: Generalized Autoregressive Pretraining for Language Understanding](https://arxiv.org/pdf/1906.08237.pdf)

通过Attention Mask引入下文信息，使得语言模型变成全排列语言模型；此外，引入Transformer-XL结构可以建模更长的文本；

### 2）Autoencoding Models

**I）BERT**

[BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding](https://arxiv.org/pdf/1810.04805.pdf)

在输入引入[mask]，只预测该词语；此外引入next sentence prediction loss，做多目标训练；

**II）RoBERTa**

[RoBERTa: A Robustly Optimized BERT Pretraining Approach](https://arxiv.org/pdf/1907.11692.pdf)

相比BERT，采用了更大的batch size，更多的训练数据，更长的训练时间。添加dynamic mask机制，去掉next sentence prediction loss。目前工业界比较使用的一个版本，因为从NLP下游的任务应用来看，该版本由于使用了更大的训练数据和更大的batch，多数效果基本是benchmark。

**III）ALBERT**

[ALBERT: A Lite BERT for Self-supervised Learning of Language Representations](https://arxiv.org/abs/1909.11942)

对参数矩阵做矩阵分解；每层transformer共享（可以减少训练的GPU显存）；next sentence prediction改变为sentence order prediction；

**IV）DistilBERT**

[DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter](https://arxiv.org/abs/1910.01108)

去掉next sentence prediction loss；采用蒸馏的方式，使用正常的BERT模型作为Teacher，DistilBERT作为Student；

**V）XLM**

[Cross-lingual Language Model Pretraining](https://arxiv.org/abs/1901.07291)

输入的embedding包括：token embedding、position embedding和language embeddin（替换原先的sentence embedding）。采用CLM、MLM、MLM/TLM分别预训练模型，MLM/TLM：使用MLM学习源语言和目标语言的上下文关系，使用TLM学习学习源语言和目标语言词语的对齐关系。用于下游的任务：语言模型、词语分类、句子分类、问答。

在机器翻译任务上，采用预训练方法初始化encoder或者decoder，在无监督和有监督机器翻译上，BLEU都会有提升；

从多语种角度来看，采用预训练的方法，对低资源语种下游任务都会有明显的帮助。

**VI）XLM-RoBERTa**

[Unsupervised Cross-lingual Representation Learning at Scale](https://arxiv.org/abs/1911.02116)

引入RoBERTa的trick继续预训练原先XLM paper里面的模型；采用超过100+种语言，不再使用language embedding，让模型自己去学习区分语言。用于下游的任务：语言模型、词语分类、句子分类、问答。

**VII）FlauBERT**

[FlauBERT: Unsupervised Language Model Pre-training for French](https://arxiv.org/abs/1912.05372)

**VIII）ELECTRA**

[ELECTRA: Pre-training Text Encoders asDiscriminators Rather Than Generators](https://arxiv.org/abs/2003.10555)

**VIV）Longformer**

[Longformer: The Long-Document Transformer](https://arxiv.org/abs/2004.05150)

使用slide window的attention学习局部上下文信息，使用global attention学习整体的信息。基于slide window的attention采用GPU程序实现复杂度很高，作者开源了基于TVM的cuda程序，比naive的实现方式快了6倍。

### 3）Sequence-to-Sequence Models

**I）BART**

[BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension](https://arxiv.org/abs/1910.13461)

采用seq2seq的结构去做预训练，文本的输入添加不同的噪声函数变换（掩码、删除、句子排列变化、文档旋转、文本填充），通过encoder编码被破坏的文本，decoder采用自回归解码器来恢复原始的文档。相比传统bert模型，参数矩阵上，多了encoder和decoder层之间的cross-attention。

**II）Pegasus**

[PEGASUS: Pre-training with Extracted Gap-sentences forAbstractive Summarization](https://arxiv.org/pdf/1912.08777.pdf)

**III）MarianMT**

[Marian: Fast Neural Machine Translation in C++](https://arxiv.org/abs/1804.00344)

**IV）T5**

[Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer](https://arxiv.org/abs/1910.10683)

**V）MBart**

[Multilingual Denoising Pre-training for Neural Machine Translation](https://arxiv.org/abs/2001.08210)

### 4）Multimodal Models

**I）MMBT**

[Supervised Multimodal Bitransformers for Classifying Images and Text](https://arxiv.org/abs/1909.02950)

### 5）Retrieval-based Models

**I）DPR**

[Dense Passage Retrieval for Open-Domain Question Answering](https://arxiv.org/abs/2004.04906)

## 二、Transformer框架

### 1）local and fixed patterns

[Image Transformer](https://arxiv.org/abs/1802.05751)（2018）

### 2）utilizing global memory

[Set Transformer - A Framework for Attention-based Permutation-Invariant Neural Networks](https://arxiv.org/abs/1810.00825)（2019）

### 3）combinations of patterns

[Sparse Transformers - Generating Long Sequences with Sparse Transformers](https://arxiv.org/abs/1904.10509)（2019）

[Axial Transformers - Axial Attention in Multidimensional Transformers](https://arxiv.org/abs/1912.12180)（2019）

### 4）Sparse Transformer family

[Longformer - The Long-Document Transformer](https://arxiv.org/abs/2004.05150)（2020）

[Extended Transformer Construction - Encoding Long and Structured Inputs in Transformers](https://arxiv.org/abs/2004.08483?url=girls.copypasteads.com)（2020）

### 5）incorporate learnable patterns

[Routing Transformers - Efficient Content-Based Sparse Attention with Routing Transformers](https://arxiv.org/abs/2003.05997)（2020）

[Reformer - The Efficient Transformer](https://arxiv.org/abs/2001.04451)（2020）

[Sinkhorn Transformers - Sparse Sinkhorn Attention](https://arxiv.org/abs/2002.11296)（2020）

### 6）lowrank factorization approaches

[Linformer - Self-Attention with Linear Complexity](https://arxiv.org/abs/2006.04768)（2020）

[Synthesizer - Rethinking Self-Attention in Transformer Models](https://arxiv.org/abs/2005.00743)（2020）

### 7）kernel approaches

[Performer -Masked Language Modeling for Proteins via Linearly Scalable Long-Context Transformers](https://arxiv.org/abs/2006.03555)（2020）

[Linear Transformers - Transformers are RNNs: Fast Autoregressive Transformers with Linear Attention](https://arxiv.org/pdf/2006.16236.pdf)（2020）

### 8）segment-based recurrence

[Transformer-xl - Attentive Language Models Beyond a Fixed-Length Context](https://arxiv.org/abs/1901.02860)（2019）

[Compressive Transformers - Compressive Transformers for Long-Range Sequence Modelling](https://arxiv.org/abs/1911.05507)（2020）