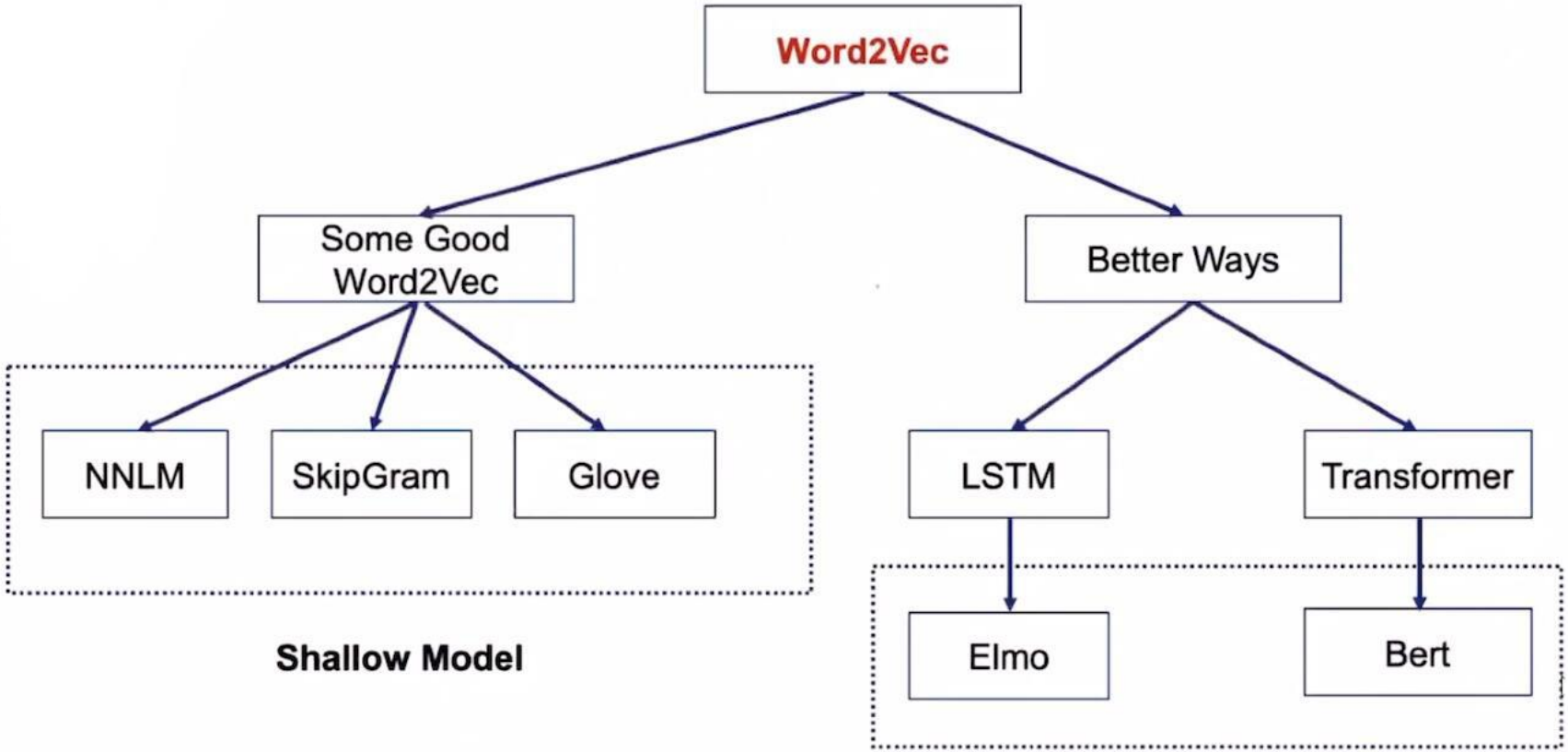
2021年度暑期强化课程

预训练模型：深度词表示模型

授课人：曹亚男

中国科学院信息工程研究所
INSTITUTE OF INFORMATION ENGINEERING,CAS

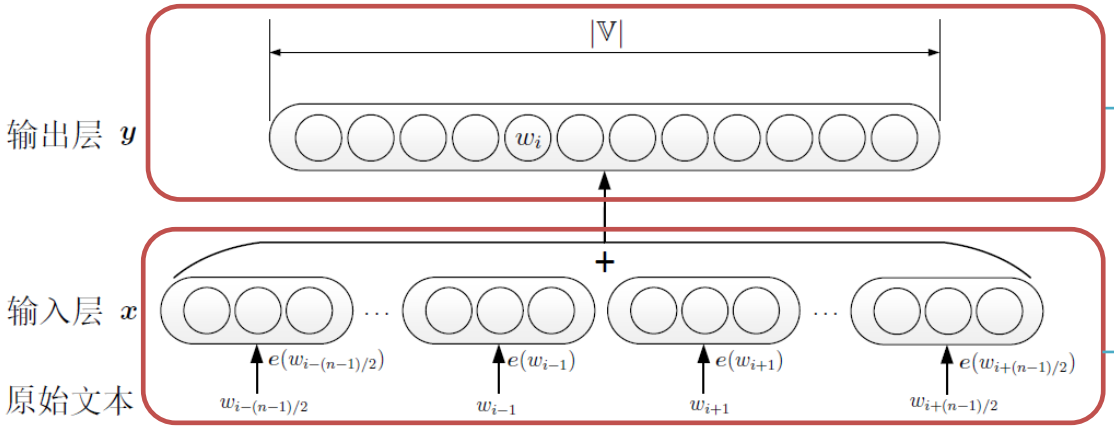
词向量模型


Word2Vec
Some Good Word2Vec
Better Ways
NNLM
SkipGram
Glove
LSTM
Transformer
Elmo
Bert
Shallow Model
Deep Model

5. 深层词嵌入表示模型

CBOW模型

- CBOW借鉴了C&W使用上下文作为输入的思想，并在NNLM基础上进行了简化



CBOW模型结构图

根据上下文的表示，直接对目标词进行预测

$$P(w|c)=\frac{\exp(e'(w)^T x)}{\sum_{w'\in V}\exp(e'(w)^T x)}$$

对于一段长度为n的训练样本，输入为

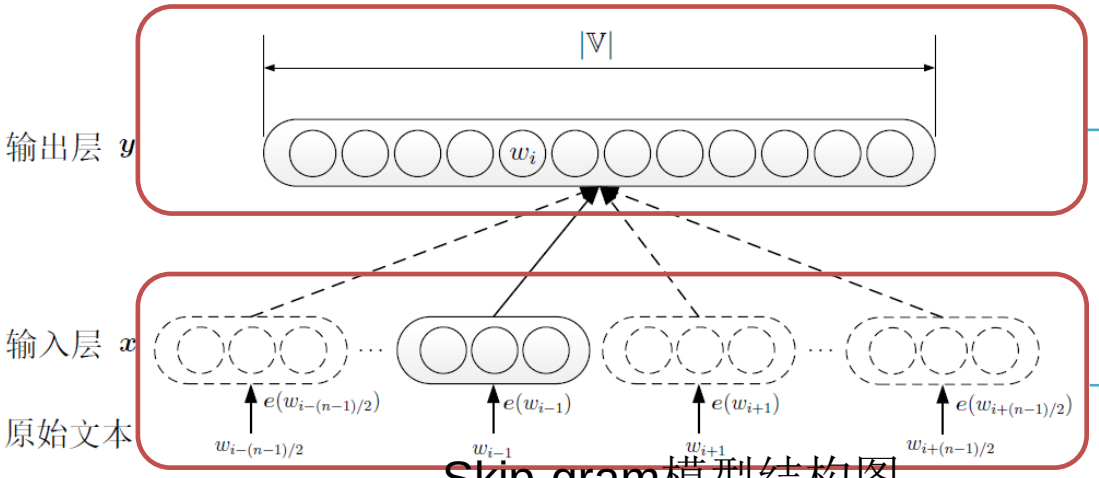
$$x=\frac{1}{n-1}\sum_{w_j\in c}e(w_j)$$

- CBOW的优化目标为最大化：$\sum_{(w,c)\in D}\log P(w|c)$
- 与NNLM的区别：1. 去除上下文的词序信息；2. 无隐藏层，NN结构转化为log线性结构

Tomas Mikolov：Efficient estimation of word representations in vector space，2013

Skip-gram模型

- Skip-gram与CBOW模型类似，模型中没有隐藏层；但模型的输入和优化目标不同



Skip-gram模型结构图

根据w_j的表示，对目标词进行预测

$$P(w \mid w_j) = \frac{\exp(e'(w)^T e(w_j))}{\sum_{w' \in V} \exp(e'(w')^T e(w_j))}$$

从目标词w的上下文中选择一个词w_j，将其作为模型的输入x

- Skip-gram的优化目标是最大化：$\sum_{(w,c)\in D} \sum_{w_j \in c} \log P(w \mid w_j)$
- 更多优化和提升：负采样、二次采样、层次softmax……

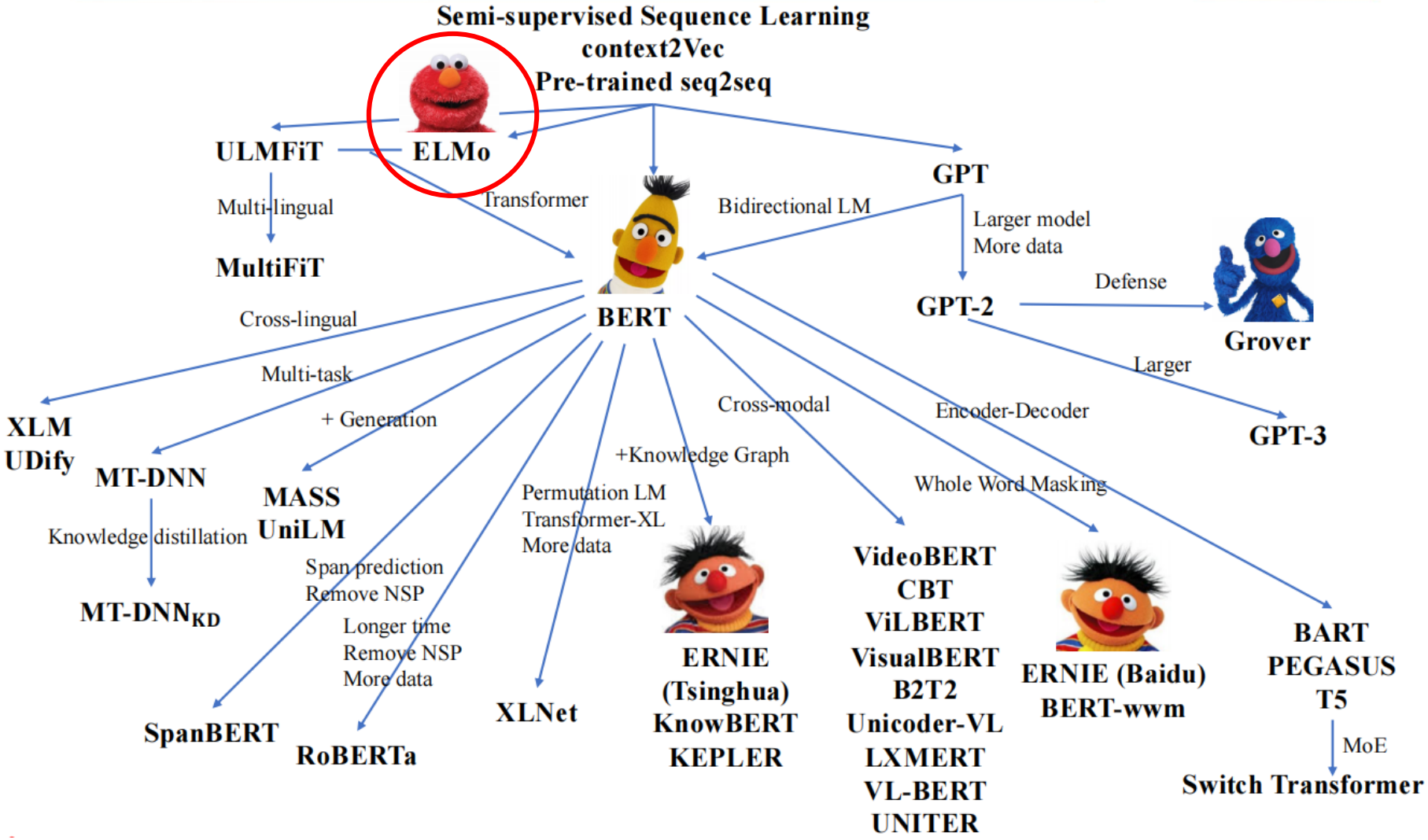
Tomas Mikolov：Efficient estimation of word representations in vector space，2013

浅层模型的缺点

- word2vec无法捕获一词多义的情况，无法根据下游任务进行调整
- 如果一个词没有根据下游任务改变自己的能力，就需要设计复杂模型在下游任务里使用词向量来展示不同层面的特征
- 从一开始就让词向量拥有可以根据不同下游任务而变换的能力？

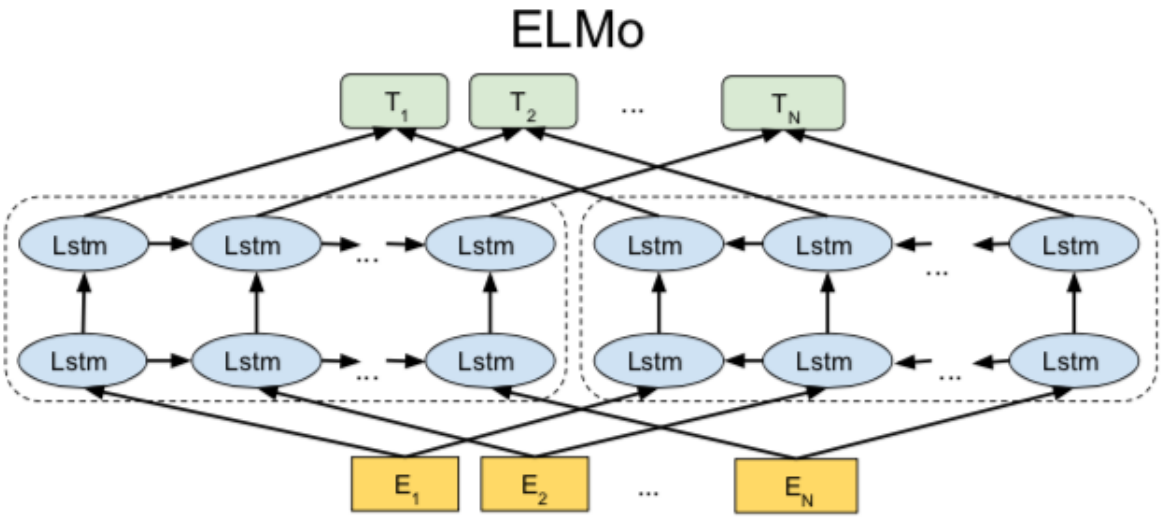
5. 深层词嵌入表示模型

预训练模型家族


Semi-supervised Sequence Learning
context2Vec
Pre-trained seq2seq
ELMo
ULMFiT
Multi-lingual
MultiFiT
Transformer
BERT
Bidirectional LM
GPT
Larger model
More data
GPT-2
Defense
Grover
Larger
GPT-3
Cross-lingual
XLM
UDify
Multi-task
MT-DNN
Knowledge distillation
MT-DNN$_{KD}$
+ Generation
MASS
UniLM
Span prediction
Remove NSP
SpanBERT
Longer time
Remove NSP
More data
RoBERTa
Permutation LM
Transformer-XL
More data
XLNet
+Knowledge Graph
ERNIE
(Tsinghua)
KnowBERT
KEPLER
Cross-modal
VideoBERT
CBT
ViLBERT
VisualBERT
B2T2
Unicoder-VL
LXMERT
VL-BERT
UNITER
Encoder-Decoder
Whole Word Masking
ERNIE (Baidu)
BERT-wwm
BART
PEGASUS
T5
MoE
Switch Transformer

ElMo

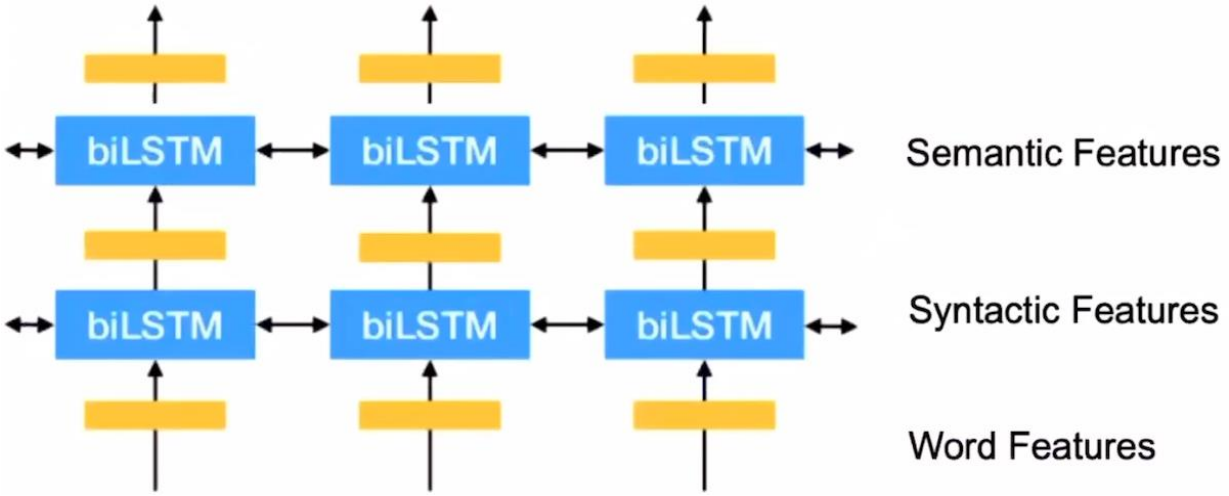
- Elmo使用双向的LSTM语言模型，由一个前向和一个后向语言模型构成，目标函数是取这两个方向语言模型的最大似然



- 前向语言模型：$p(t_1, t_2, \ldots, t_N) = \prod_{}^{N} p(t_k | t_1, t_2, \ldots, t_{k-1})$
- 后向语言模型：$p(t_1, t_2, \ldots, t_N) = \prod_{}^{N} p(t_k | t_{k+1}, t_{k+2}, \ldots, t_N)$
- 目标函数：$\sum_{k=1}^{N} (logp(t_k | t_1, t_2, \ldots, t_{k-1}) + logp(t_k | t_{k+1}, t_{k+2}, \ldots, t_N))$

ELMo预训练

- ELMo给原始词向量层和每个LSTM隐层都设置了一个可训练参数，每层BiLSTM学习不同特征，不同下游任务对每层的权重也不同
- 可以将各层的输出作为最后的输出，或将各层的输出进行组合作为最后的输出，将学习到的向量表示 $[\mathbf{x}_k; \mathbf{ELMo}_k^{task}]$ 输出给下游任务



$$R_k = \{\mathbf{x}_k^{LM}, \overrightarrow{\mathbf{h}}_{k,j}^{LM}, \overleftarrow{\mathbf{h}}_{k,j}^{LM} \mid j = 1, \ldots, L\}$$
$$= \{\mathbf{h}_{k,j}^{LM} \mid j = 0, \ldots, L\},$$

$$\mathbf{ELMo}_k^{task} = E(R_k; \Theta^{task}) = \gamma^{task} \sum_{j=0}^{L} s_j^{task} \mathbf{h}_{k,j}^{LM}.$$

Matthew E. Peters, et.al., Deep contextualized word representations,2018. NAACL Best Paper

ELMo实验效果

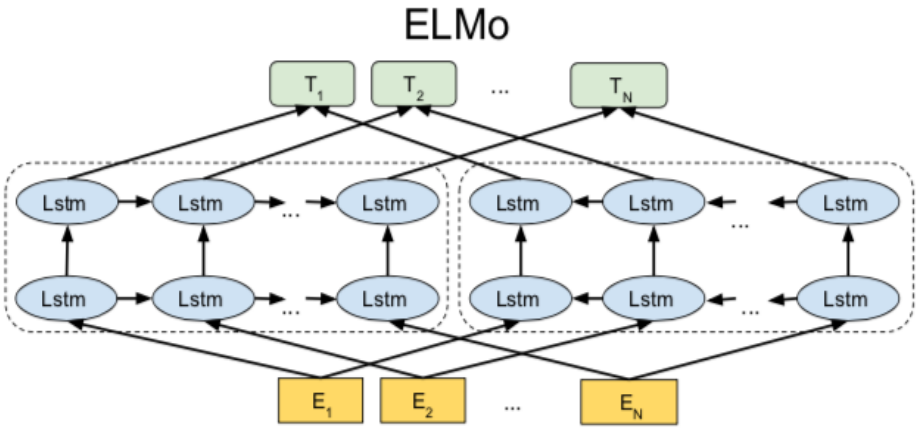
Task	Previous SOTA		Our baseline	ELMo + baseline	Increase (absolute/ relative)
SQuAD	Liu et al. (2017)	84.4	81.1	85.8	4.7 / 24.9%
SNLI	Chen et al. (2017)	88.6	88.0	88.7 ± 0.17	0.7 / 5.8%
SRL	He et al. (2017)	81.7	81.4	84.6	3.2 / 17.2%
Coref	Lee et al. (2017)	67.2	67.2	70.4	3.2 / 9.8%
NER	Peters et al. (2017)	91.93 ± 0.19	90.15	92.22 ± 0.10	2.06 / 21%
SST-5	McCann et al. (2017)	53.7	51.4	54.7 ± 0.5	3.3 / 6.8%

Table 1: Test set comparison of ELMo enhanced neural models with state-of-the-art single model baselines across six benchmark NLP tasks. The performance metric varies across tasks – accuracy for SNLI and SST-5; F_1 for SQuAD, SRL and NER; average F_1 for Coref. Due to the small test sizes for NER and SST-5, we report the mean and standard deviation across five runs with different random seeds. The "increase" column lists both the absolute and relative improvements over our baseline.

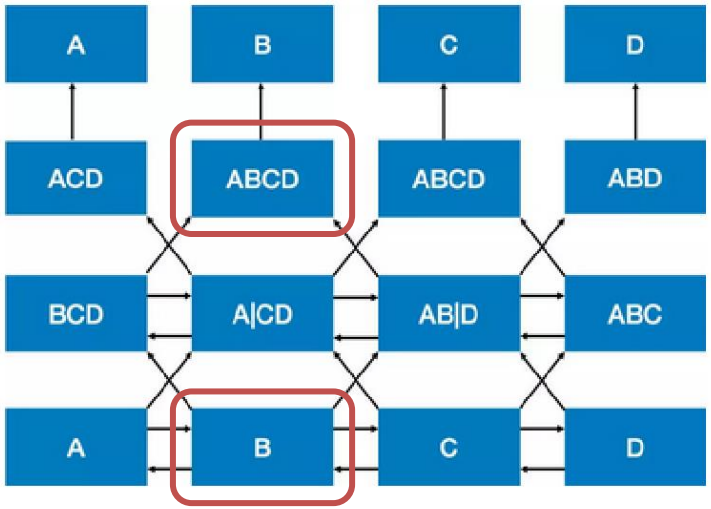
- 在问答、文本蕴含、语义角色标注、指代消解、命名实体识别、情感分析6个任务上达到了最好结果

ELMo的几个问题

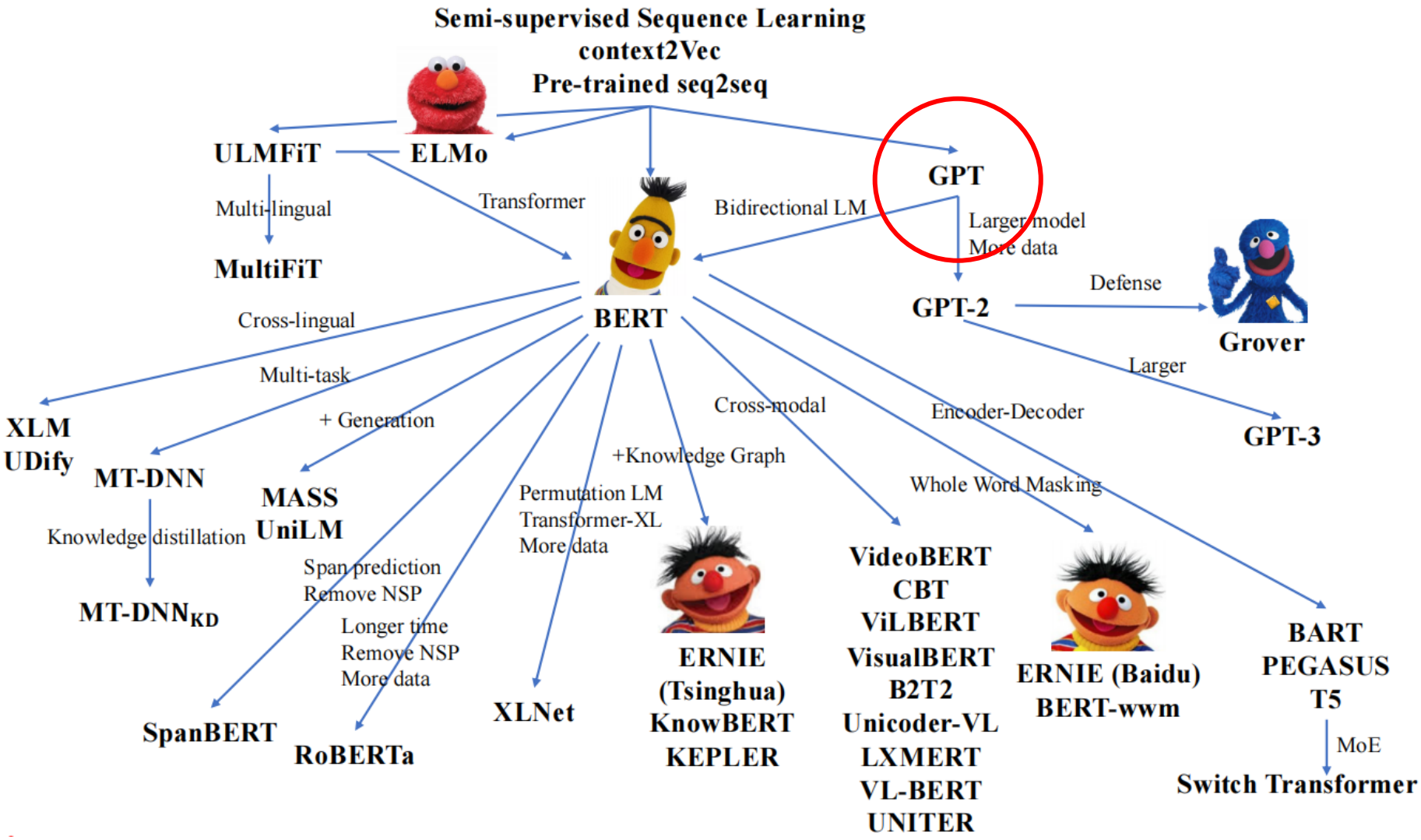
- **不完全双向**：前向和后向LSTM两个模型分别训练，得到的隐层向量直接拼接得到结果向量；Loss function是前向和后向的loss function直接相加，并非完全同时的双向计算
- **自己看见自己**：要预测的下一个词在给定的序列中已经出现，而传统语言模型的数学原理决定了它的单向性
- **使用LSTM**：无法并行化、特征提取能力不够



$$\sum_{k=1}^{N}(logp(t_k|t_1,t_2,\ldots,t_{k-1})+logp(t_k|t_{k+1},t_{k+2},\ldots,t_N))$$

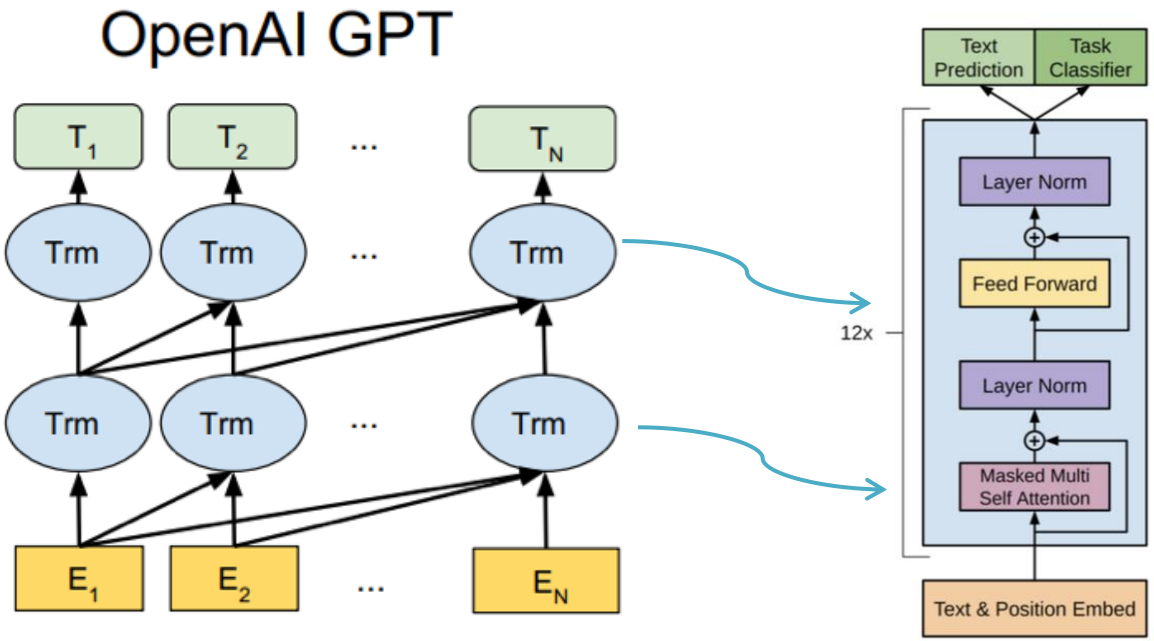


预训练模型家族


Semi-supervised Sequence Learning
context2Vec
Pre-trained seq2seq
ULMFiT
ELMo
GPT
Multi-lingual
MultiFiT
Transformer
Bidirectional LM
Larger model
More data
Defense
GPT-2
Grover
Cross-lingual
Multi-task
BERT
Larger
GPT-3
XLM
UDify
MT-DNN
+ Generation
Cross-modal
Encoder-Decoder
+Knowledge Graph
MASS
UniLM
Permutation LM
Transformer-XL
More data
Whole Word Masking
Knowledge distillation
MT-DNN$_{KD}$
Span prediction
Remove NSP
Longer time
Remove NSP
More data
VideoBERT
CBT
ViLBERT
VisualBERT
B2T2
Unicoder-VL
LXMERT
VL-BERT
UNITER
ERNIE (Baidu)
BERT-wwm
BART
PEGASUS
T5
MoE
Switch Transformer
SpanBERT
RoBERTa
XLNet
ERNIE
(Tsinghua)
KnowBERT
KEPLER

GPT

- 使用transformer作为RNN的替代结构，仅采用decoder部分Mask Multi-Head Attention结构
- 只采用上文词来进行预测当前词，不再考虑下文词
- 预训练阶段的目标是最大化语言模型 $L_1(\mathcal{U}) = \sum \log P(u_i|u_{i-k}, \ldots, u_{i-1}; \Theta)$



Radford, Alec and Karthik Narasimhan. “Improving Language Understanding by Generative Pre-Training.” (2018).

GPT

- 在Fine-tune阶段，将预训练模型提供给下游的任务，预训练模型与下游任务模型联合优化 $L_3(\mathcal{C}) = L_2(\mathcal{C}) + \lambda * L_1(\mathcal{C})$

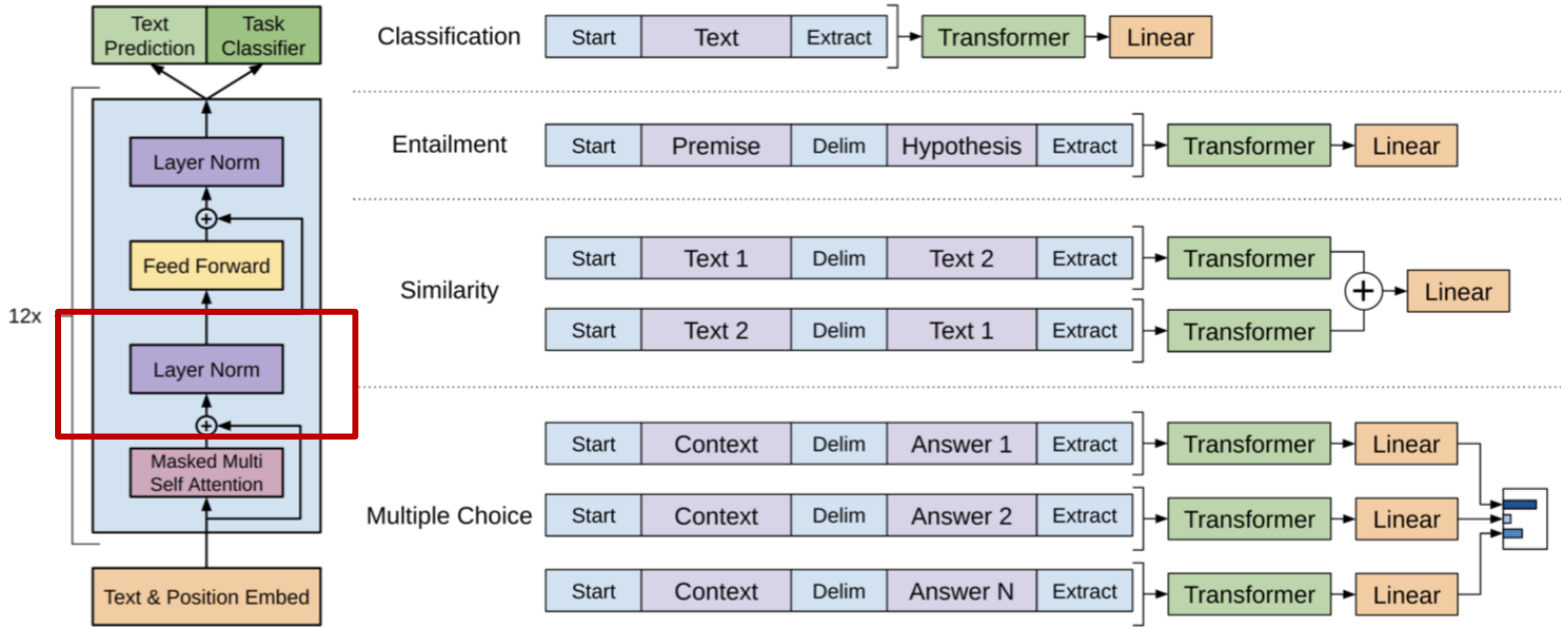


Figure 1: **(left)** Transformer architecture and training objectives used in this work. **(right)** Input transformations for fine-tuning on different tasks. We convert all structured inputs into token sequences to be processed by our pre-trained model, followed by a linear+softmax layer.

Radford, Alec and Karthik Narasimhan. “Improving Language Understanding by Generative Pre-Training.” (2018).

GPT实验效果

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.

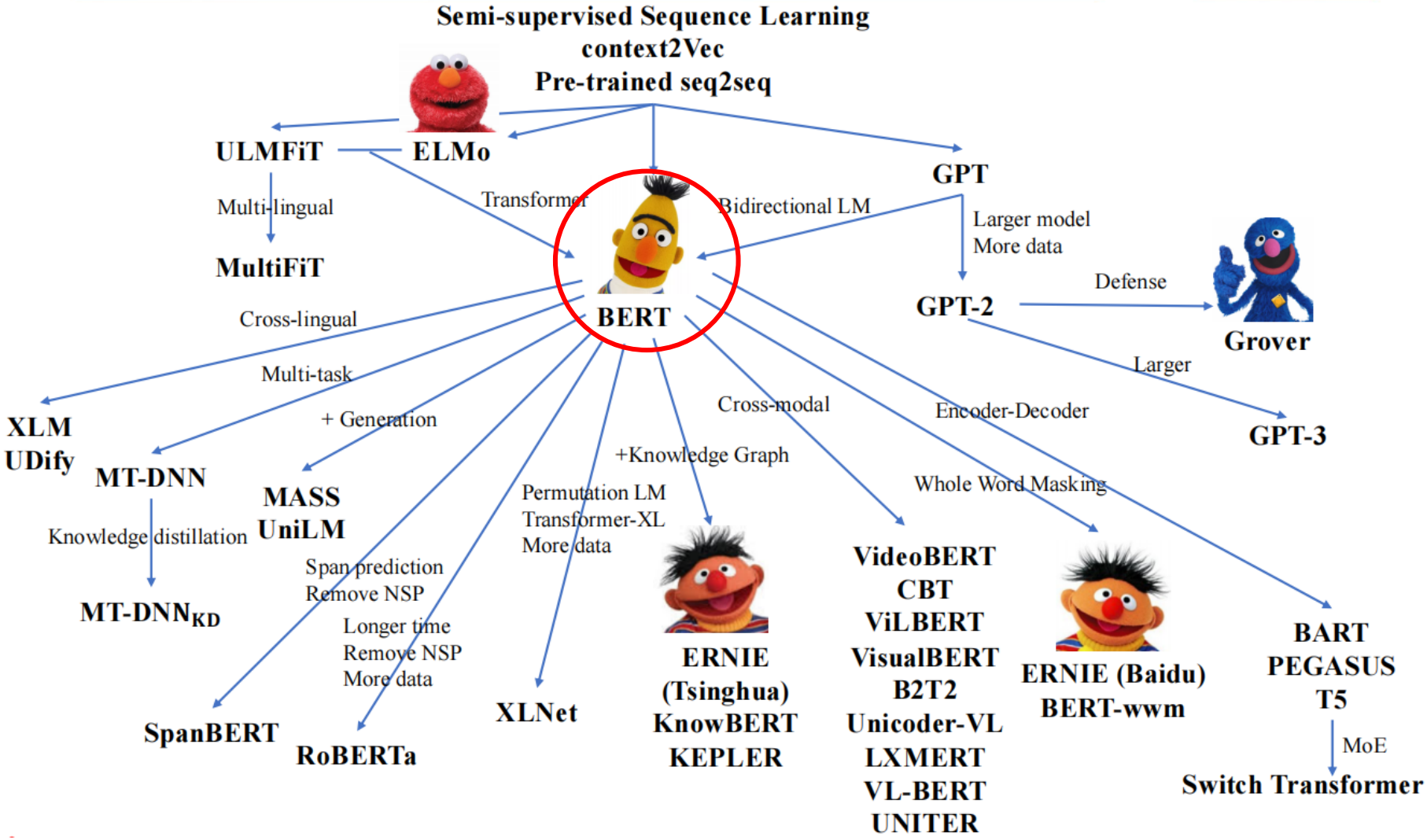
Method	Story Cloze	RACE-m	RACE-h	RACE
val-LS-skip [55]	76.5	-	-	-
Hidden Coherence Model [7]	77.6	-	-	-
Dynamic Fusion Net [67] (9x)	-	55.6	49.4	51.2
BiAttention MRU [59] (9x)	-	60.2	50.3	53.3
Finetuned Transformer LM (ours)	**86.5**	**62.9**	**57.4**	**59.0**

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**

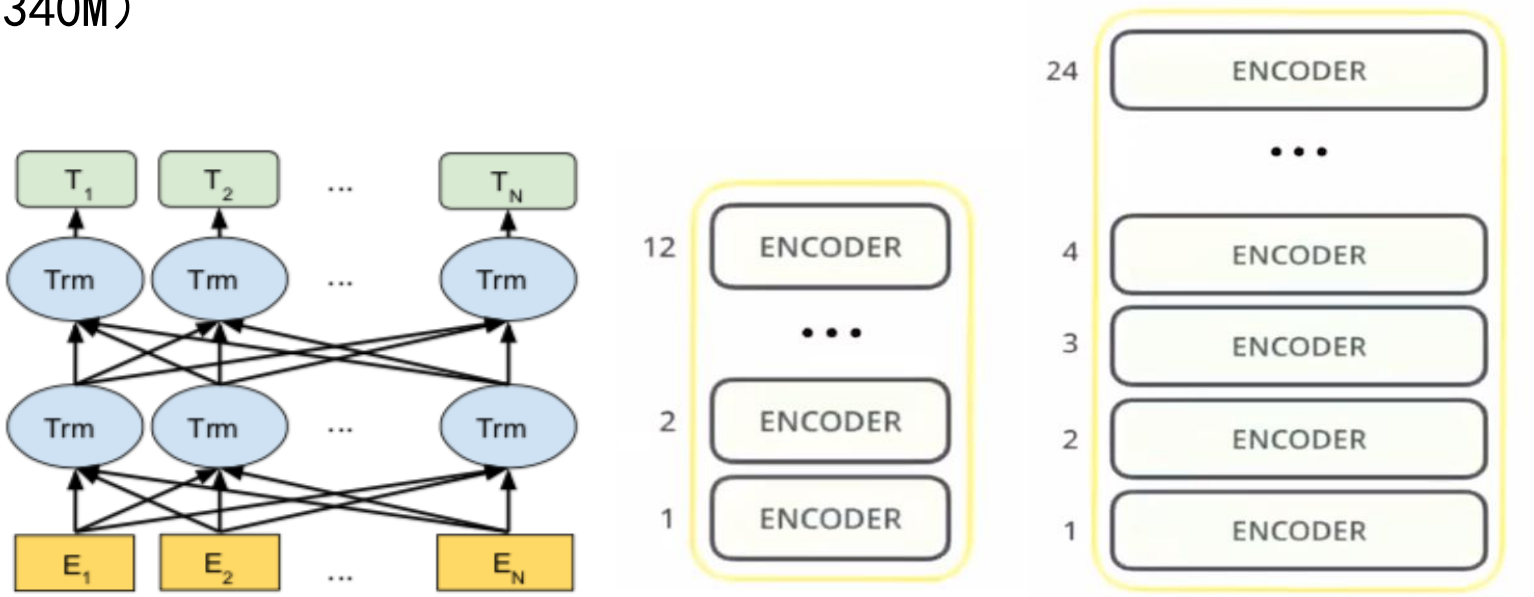
- 在9个任务上达到了最好结果
- 文本生成任务效果提升显著
- 分类任务效果较差

预训练模型家族


Semi-supervised Sequence Learning
context2Vec
Pre-trained seq2seq
ULMFiT
ELMo
GPT
Multi-lingual
MultiFiT
Transformer
Bidirectional LM
Larger model
More data
Defense
GPT-2
Grover
Cross-lingual
BERT
Multi-task
Larger
GPT-3
XLM
UDify
MT-DNN
+ Generation
Cross-modal
Encoder-Decoder
+Knowledge Graph
Whole Word Masking
Permutation LM
Transformer-XL
More data
MASS
UniLM
Knowledge distillation
MT-DNN$_{KD}$
Span prediction
Remove NSP
Longer time
Remove NSP
More data
SpanBERT
RoBERTa
XLNet
ERNIE
(Tsinghua)
KnowBERT
KEPLER
VideoBERT
CBT
ViLBERT
VisualBERT
B2T2
Unicoder-VL
LXMERT
VL-BERT
UNITER
ERNIE (Baidu)
BERT-wwm
BART
PEGASUS
T5
MoE
Switch Transformer

Bert模型

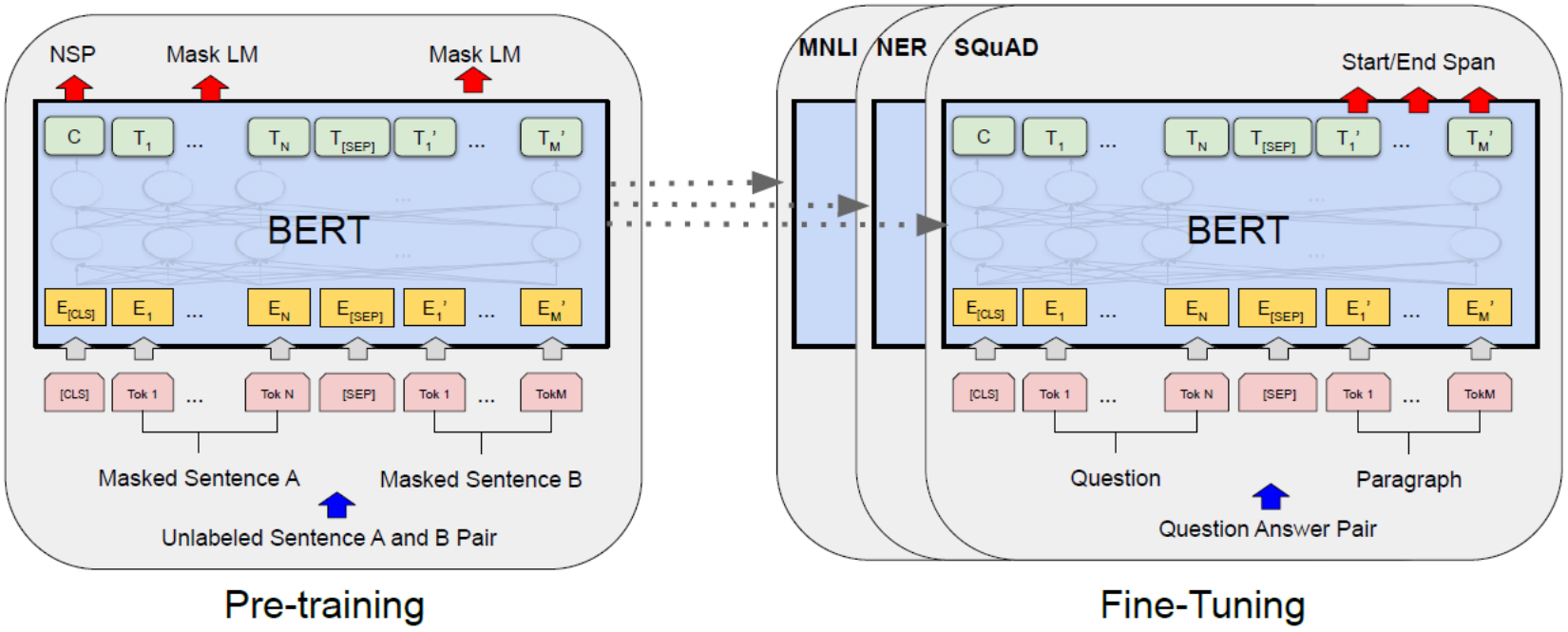
- 使用堆叠的双向Transformer Encoder，在所有层中共同依赖于左右上下文，不用Decoder
- 基础版是12个Encoder（768隐藏单元，12个head，总参数量110M）
- 高级版24个Encoder（24层，1024隐藏单元，16个head，总参数量340M）



BERT　　$BERT_{BASE}$　　$BERT_{Large}$

Bert

- Pre-training和Fine-Tuning阶段使用相同的模型结构
- 同样的Pre-trainng模型参数可用于初始化不同下游任务模型
- 预训练在微调时可以仅通过使用一个额外的输出层就可以达到最好的效果



Pre-Training策略

1. 看不到自己

MLM：随机屏蔽（masking）部分输入token，然后只预测那些被屏蔽的token

2. 建模句间关系

预训练句子关系模型，使用二元分类任务预训练句子关系模型

输入和输出

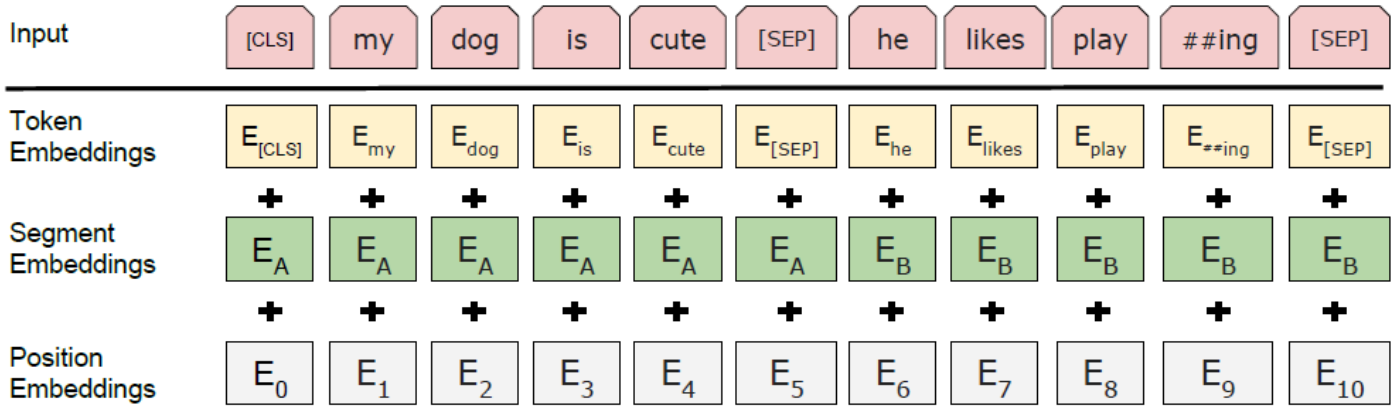
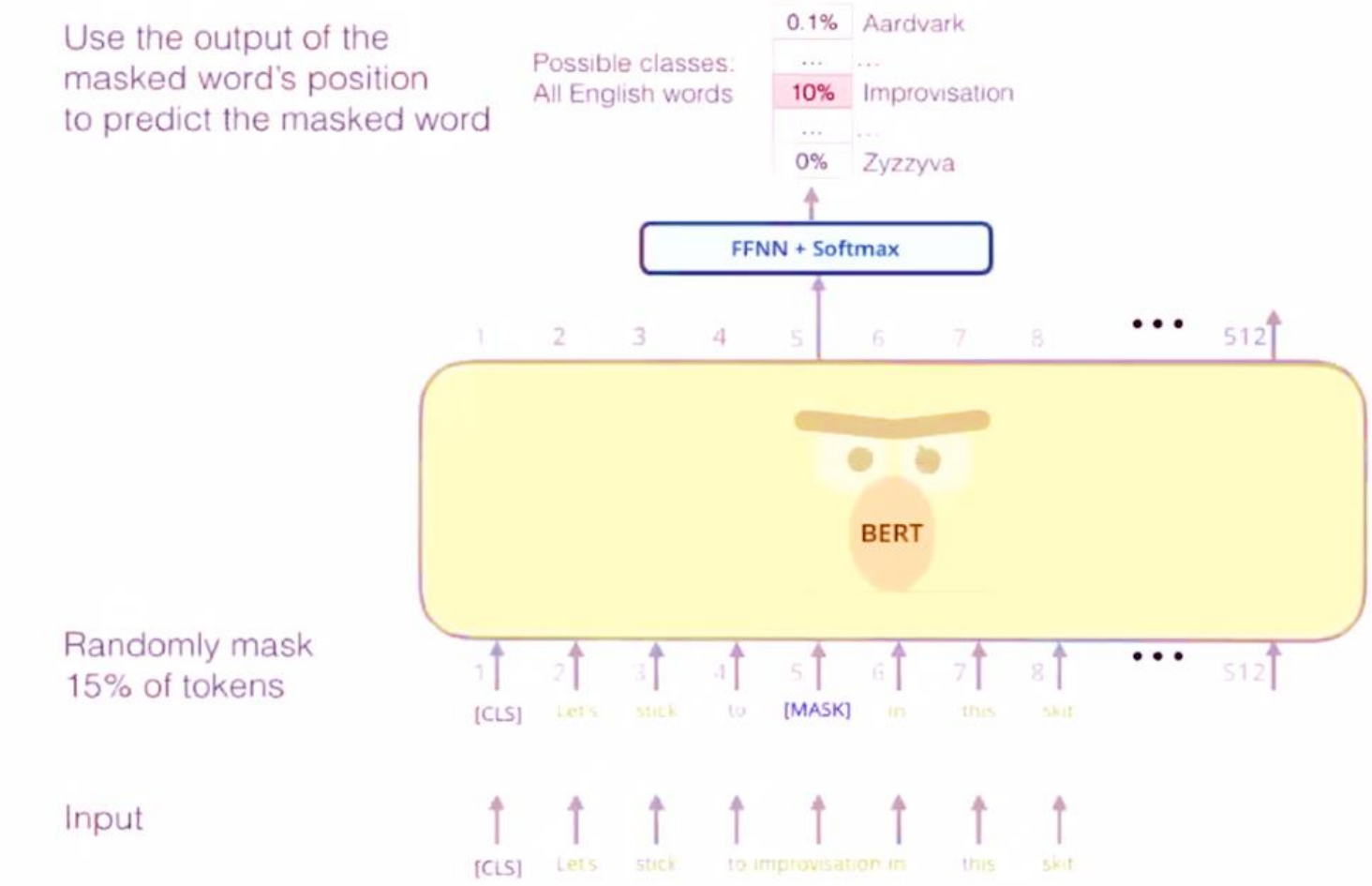


Figure 2: BERT input representation. The input embeddings are the sum of the token embeddings, the segmentation embeddings and the position embeddings.

- 为了能够处理多类下游任务，输入是一个句子或一个句对（例如<Question, Answer>），此处的句子指**任意长度的连续文本片段**，而不是语言学上所指的句子
- 每个 sequence 以 [CLS] 开头，最长 512 个字符，最终 [CLS] 的隐层状态在分类任务代表了整个 sequence
- 句子对之间加一个 [SEP]，用A/B embedding 区分一个词所在的句子

Bert：Masked MLM

- 与训练语言模型类似，由编码器来提供输入，用被掩盖位置的最终隐藏状态预测被掩盖的单词



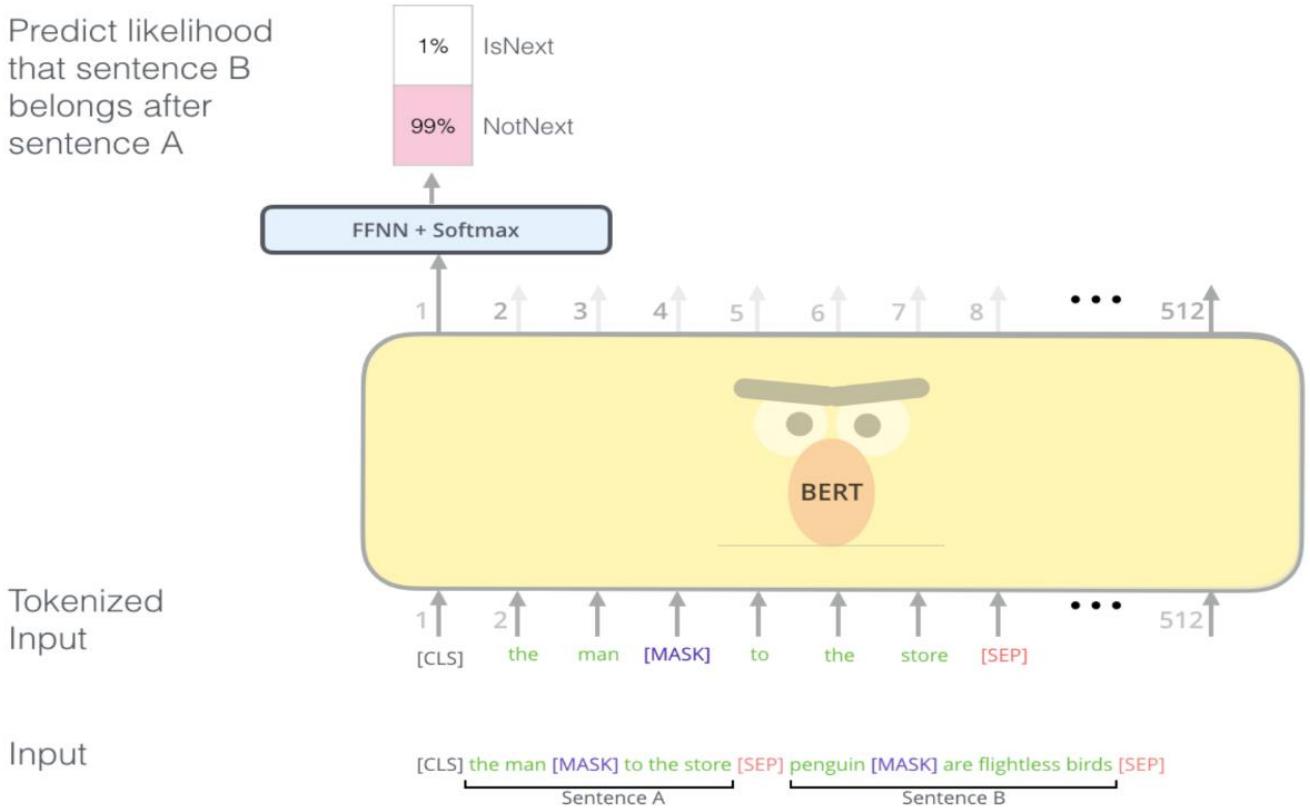
Bert：Masked MLM

- 问题1：[MASK]标记在微调期间从未出现过，预训练和微调之间不匹配
  - ✓ 随机挑选15%的词进行遮盖，其中80%直接替换成[MASK]，10%被随机替换成另一个语料中的词，10%不做任何处理
  - ✓ Transformer encoder不知道它将被要求预测哪些单词或哪些单词已被随机单词替换，因此它被迫保持每个输入token的分布式上下文表示

- 问题2：每批次数据中只有15%的标记被预测，模型需要更多的预训练步骤来收敛
  - ✓ Transformer确实比从左到右的模型（预测每个标记）稍微慢一点，但是Transformer 模型的实验效果远远超过了它增加的预训练模型的成本

Bert训练：Next Sentence Prediction

- 二元分类任务：给定输入句对<A, B>，预测B是否是A的下一个句子
- 当选择句子A和B作为预训练样本时，B有50%的可能是A的下一个句子，也有50%的可能是来自语料库的随机句子



The second task BERT is pre-trained on is a two-sentence classification task. The tokenization is oversimplified in this graphic as BERT actually uses WordPieces as tokens rather than words --- so some words are broken down into smaller chunks.

Bert效果

- GLUE：MNLI精度达86.7%

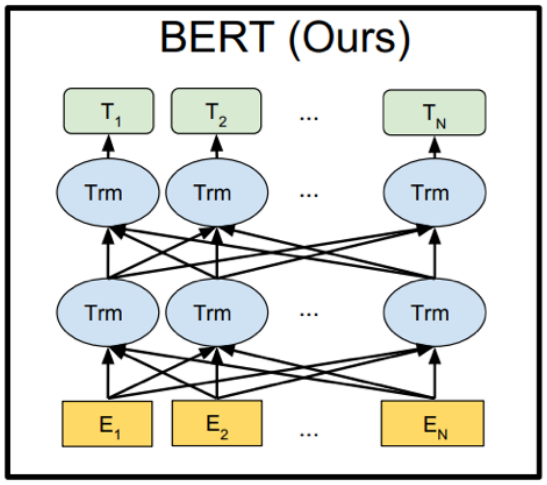
System	MNLI-(m/mm) 392k	QQP 363k	QNLI 108k	SST-2 67k	CoLA 8.5k	STS-B 5.7k	MRPC 3.5k	RTE 2.5k	**Average** -
Pre-OpenAI SOTA	80.6/80.1	66.1	82.3	93.2	35.0	81.0	86.0	61.7	74.0
BiLSTM+ELMo+Attn	76.4/76.1	64.8	79.8	90.4	36.0	73.3	84.9	56.8	71.0
OpenAI GPT	82.1/81.4	70.3	87.4	91.3	45.4	80.0	82.3	56.0	75.1
$\text{BERT}_{\text{BASE}}$	84.6/83.4	71.2	90.5	93.5	52.1	85.8	88.9	66.4	79.6
$\text{BERT}_{\text{LARGE}}$	**86.7/85.9**	**72.1**	**92.7**	**94.9**	**60.5**	**86.5**	**89.3**	**70.1**	**82.1**

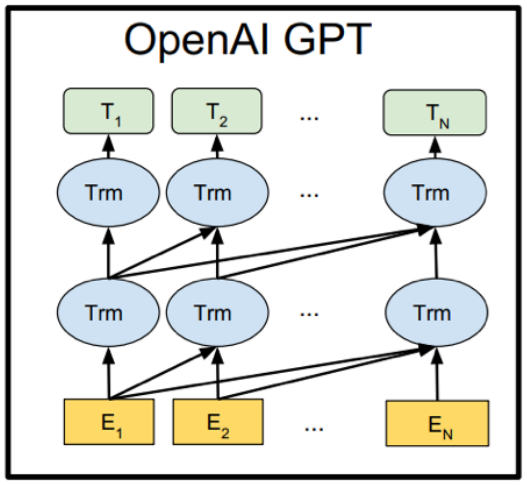
- SQuAD 问答F1得分93.2分（提升1.5分），超过人类表现2.0分

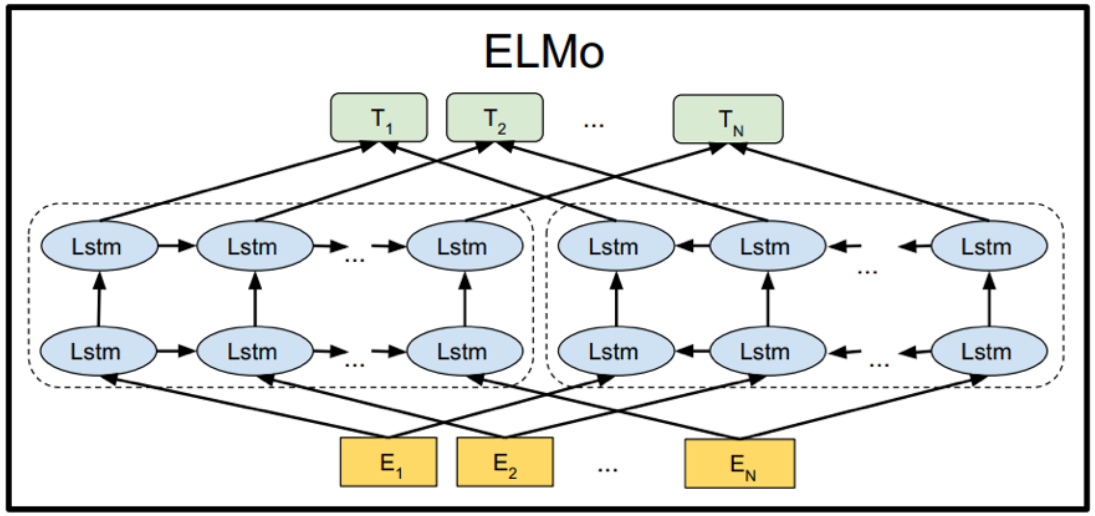
System	Dev EM	Dev F1	Test EM	Test F1
Top Leaderboard Systems (Dec 10th, 2018)				
Human	-	-	82.3	91.2
#1 Ensemble - nlnet	-	-	86.0	91.7
#2 Ensemble - QANet	-	-	84.5	90.5
Published				
BiDAF+ELMo (Single)	-	85.6	-	85.8
R.M. Reader (Ensemble)	81.2	87.9	82.3	88.5
Ours				
$\text{BERT}_{\text{BASE}}$ (Single)	80.8	88.5	-	-
$\text{BERT}_{\text{LARGE}}$ (Single)	84.1	90.9	-	-
$\text{BERT}_{\text{LARGE}}$ (Ensemble)	85.8	91.8	-	-
$\text{BERT}_{\text{LARGE}}$ (Sgl.+TriviaQA)	**84.2**	**91.1**	**85.1**	**91.8**
$\text{BERT}_{\text{LARGE}}$ (Ens.+TriviaQA)	**86.2**	**92.2**	**87.4**	**93.2**

- 刷新11项自然语言处理任务榜单：
  - ✓ 句子关系判断及分类任务
  - ✓ 抽取式任务（SQuAD）
  - ✓ 序列标注：命名实体识别
  - ✓ 分类任务：SWAG
  - ？ 文本生成任务

ElMo vs. BERT vs. GPT







- RNN vs Transformer
- 双向 vs 单向
- 生成 vs 预测

参考文献

- Radford, Alec and Karthik Narasimhan. Improving Language Understanding by Generative Pre-Training. 2018.
- Matthew E. Peters, et.al., Deep contextualized word representations,2018.
- Jacob Devlin, Ming-Wei Chang, et.al.: Pre-training of Deep Bidirectional Transformers for Language Understanding. CoRR abs/1810.04805 2018
- Pre-Trained Models: Past, Present and Future. arXiv:2106.07139v2 [cs.AI] 15 Jun 2021

欢迎加入DL4NLP！

中国科学院信息工程研究所
INSTITUTE OF INFORMATION ENGINEERING,CAS