

# 《BERT 与逻辑的结合》

YKY

September 28, 2021

- 我比较熟悉 经典逻辑 AI，写过 逻辑引擎
- 但我没有 BERT/GPT 的实战经验
- 今天我们考虑一下 结合 BERT/GPT 和 逻辑引擎 的可能，有什么优势？

## 0. 我的策略

- 将 BERT/GPT 解释为一种 逻辑 / 符号演算的系统
- 将 逻辑结构 impose 到新的 BERT/GPT 模型  
(它不再是语言模型，而是逻辑模型)
- 利用我们对逻辑 AI 的理解，  
改良这新的模型，  
从逻辑角度理解参数的意义
- 如果不这样做，BERT/GPT 仍然是 “black box”，  
那就很难想出改良的思路

# 1. 逻辑 + BERT/GPT 混合的好处

似乎有一些优势，但我暂时未能提出一个很有说服力的说法：

- Long-term memory as a separate module
- Explicitly edit memories
- Make the BERT/GPT model more transparent

## 2. List of things desirable regardless of logicalization

由于我比较熟悉 logic，我不知有没有不用 logic 的更好方法.... 各位同学可以想想。

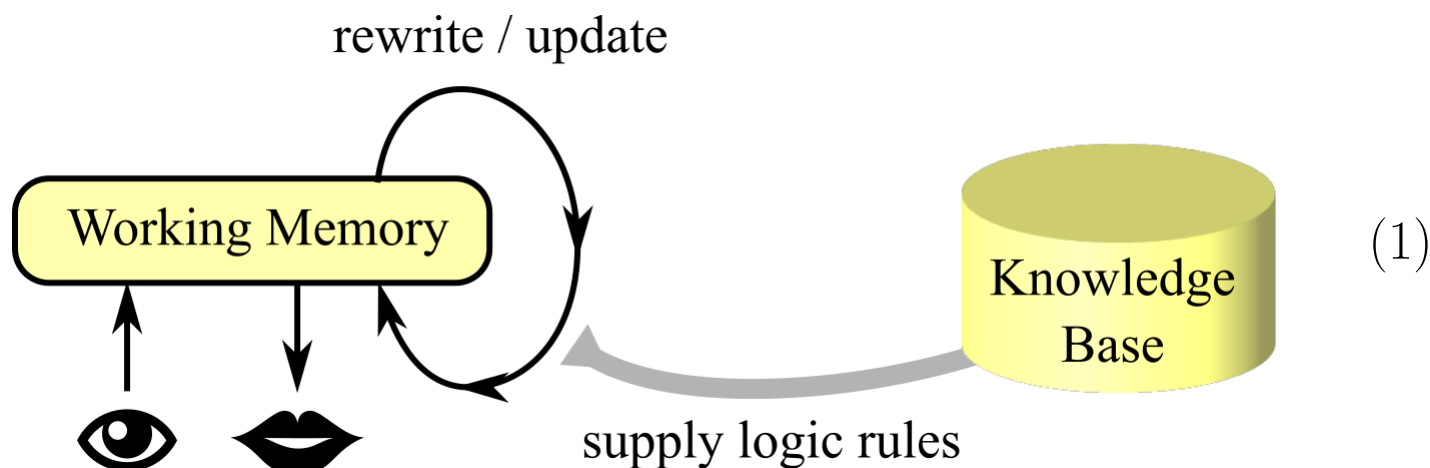
- 希望 BERT/GPT 的 inference 更 robust
- BERT/GPT 直接学习知识的能力 (“learn by being told”) 那就可以将这 AI 放到人类工作环境，透过工作教导 AI 似乎需要「长期记忆」(LTM) 但如果这 LTM 不是一个独立 module, 可能比较麻烦?

## 3. Transformer 的 equi-variance

(见 知乎 文章)

## 4. Logic AI 的基本架构

经典 logic-based AI 的架构，其实很简单的：



借用 **认知科学** 术语，Working Memory = 系统的 **状态** (state), 例如：

我很肚饿  $\wedge$   
冰箱没有食物  $\wedge$   
现在是午夜 3 点  $\wedge$   
商店已经打烊  $\wedge$   
....  $\wedge$  ....

换句话说，状态 是一堆 逻辑命题 的 **集合**

Working Memory 的 状态更新 是靠 逻辑 rules 的作用，  
例如： $\forall x. \text{human}(x) \rightarrow \text{mortal}(x)$

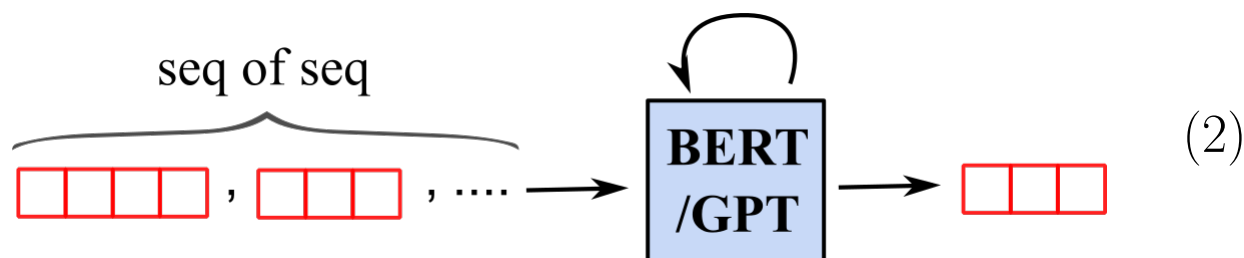
## 5. 神经科学 带来的的启发

- 一直以来，人们觉得 大脑的 KR (knowledge representtation, 知识表述) 跟符号逻辑 肯定是大相迳庭的

## 6. Seq-seq-2-seq

我提出：BERT/GPT 可能是一种 “seq-seq  $\rightarrow$  seq” architecture.

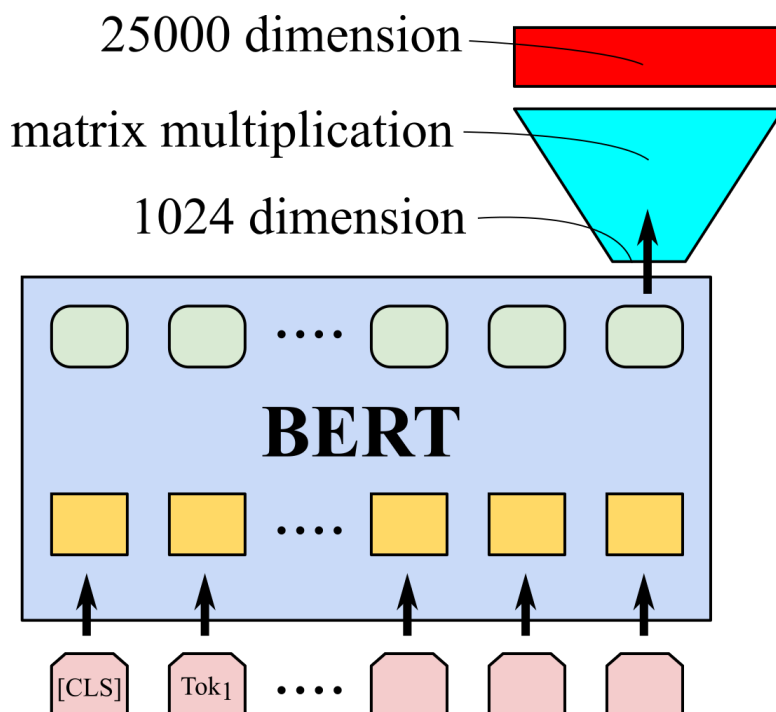
- 逻辑 与 自然语言 之间大约有这样的对应：  
句子  $\approx$  命题，词语  $\approx$  概念，  
命题 = 多个概念的 concatenation
- 从 强化学习的角度看：  
状态 (state) = 命题集合，  
transition function: 命题集合  $\rightarrow$  命题集合
- 命题集 = sequence of 命题，  
命题 = sequence of concepts，  
所以 状态 = 命题集 = sequence of sequences (seq-seq)
- transition fn: seq-seq  $\rightarrow$  seq-seq



- 每次状态更新时，我们可以只增添一个命题，  
「遗忘」另一个命题
- 因此 transition fn 只需是 seq-seq  $\rightarrow$  seq

## 7. 强化学习的考虑

- 从 强化学习 的角度看，  
每个 iteration 要输出一个 **命题** = 几个**词语**
- 这 输出 对应于 强化学习的 **actions**
- 换句话说，每个 action = 一个命题 = 几个词语
- 所以，我们需要输出 在 actions 之上的 **概率分布**  
(而不仅仅是一个 action)
- 数学上 这是  $\{\text{所有可能命题}\} \rightarrow \mathbb{R}$  的空间  $= \mathbb{R}^{|X|}$
- 这个空间异常大，我初时觉得 没有希望在计算机上表达
- 但 Dr 肖达 解释了一个很有效率的方法，  
用 矩阵乘法 将输出 由 1024 维 **扩张**到 25000 维：



(3)

- 但这个做法，其实输出的 只有 1024 个 **独立**的份量

例如,「天气很热,我在家中整天\_\_\_\_\_」

- 流汗
- 吃冰淇淋
- 喝冰水
- 不穿衣服
- 开冷气....

「女朋友说分手,我觉得\_\_\_\_\_」

- 很伤心
- 如释重负
- 很气愤
- 很妒忌....

「电脑的键盘没反应,可能是因为\_\_\_\_\_」

- 未插线
- 电线断了
- 档机了
- 视窗未 active ....

考虑这些例子,我暂时不清楚 1024 维 够不够用。

以 1024-dim 表示所有 **概念** 是足够的 (cf. Word2Vec)

但未知它能不能够 表示所有常见的 **multi-modal** 概率分布。

## 8. BERT/GPT 是符号演算系统

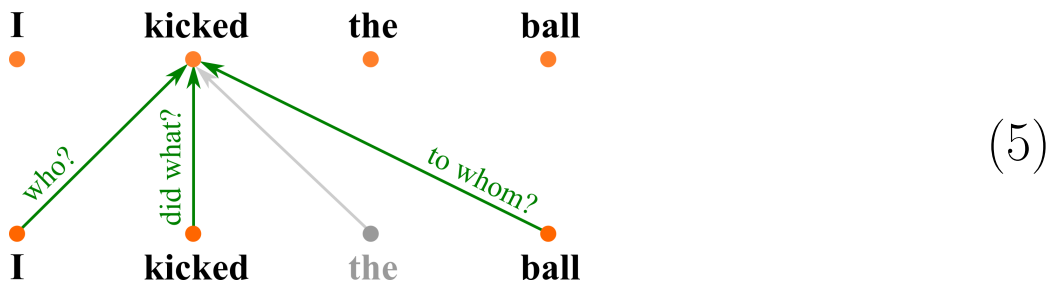
Few-shot generalization.

# 9. Variable binding

量词  $\forall X$  将 变量  $X$  「捆绑」，意思是在  $\forall X$  的 **有效范围** (scope) 内，所有  $X$ 's 的取值 必须是一**样**的：

$$\forall X, Y, Z. \text{ grandfather}(\textcolor{red}{X}, \textcolor{red}{Z}) \leftarrow \text{father}(\textcolor{red}{X}, \textcolor{red}{Y}) \wedge \text{father}(\textcolor{red}{Y}, \textcolor{red}{Z}) \quad (4)$$

用 self-Attention 表达，即是 将 输入层 的一个 token “copy” 到 输出层的 另一个 token 位置：



# 10. Relation algebra

Relation algebra 似乎是一种更 接近 自然语言 的 逻辑形式：

$$\begin{matrix} F & \circ & F & = & G \\ \text{爸爸} & \text{的} & \text{爸爸} & \text{是} & \text{爷爷} \end{matrix} \quad (6)$$

$$\begin{matrix} a & F & b \\ \text{Albert} & \text{是 爸爸 of} & \text{Bob} \\ b & F & c \\ \text{Bob} & \text{是 爸爸 of} & \text{Charles} \end{matrix} \quad (7)$$

$$\begin{array}{ccccc} a & & (F & \circ & F) & & c \\ \text{Albert} & \text{是} & \text{爸爸} & \text{of} & \text{爸爸} & \text{of} & \text{Charles} \end{array} \quad (8)$$

$$\begin{array}{ccccc} a & & G & & c \\ \text{Albert} & \text{是} & \text{爷爷} & \text{of} & \text{Charles} \end{array} \quad (9)$$

其实上例中 (7)  $\Rightarrow$  (8) 混合了 predicate logic.

Relation algebra 只描述 **关系** 之间的代数，  
但不涉及 **关系内** 的元素。

这个例子表明 BERT/GPT 用的可能是一种更为 flexible 的 **rewriting system** (改写系统)。

注意：relation algebra 不同于 relational algebra, 后者是描述 database 用。

## 11. 自动产生 / 运行 代码

我们可以**生成**一些 programming problems,  
而且 100% 肯定 答案是正确的。

逻辑上的  $\forall$  **泛化** 可以保证  
学习出来的知识是正确的，  
因为它很有可能是 **minimum description length** (MDL).

但逻辑引擎的问题是：  
它没有很效率的 **learning algorithm** (不像 深度学习)

问题是 BERT/GPT 可以怎样「帮助」逻辑引擎 的学习...?