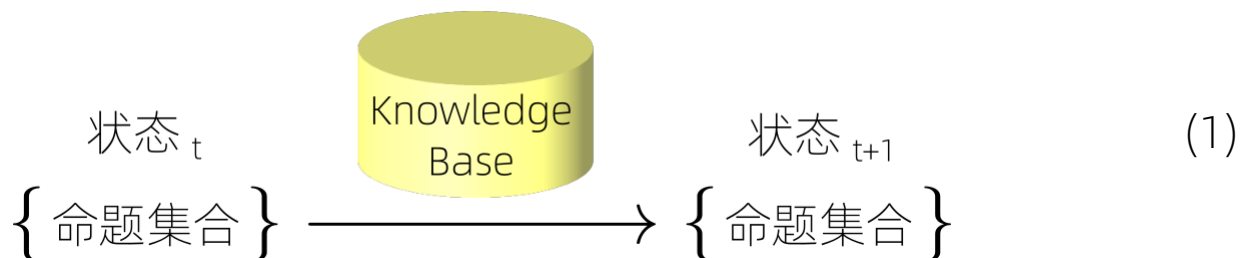


①

# 逻辑与深度学习的关系

这是经典逻辑 AI 的最基本运作模式：



它其实包含了两个算法：

- **matching** (unification):  
 逻辑 rules 是包含变量的条件命题，  
 例如  $\forall x. \text{是人}(x) \Rightarrow \text{会死}(x).$   
 Unification 判定一条 rule 是否可以 apply 到某逻辑命题上，  
 例如：是人(苏格拉底) 可以跟上式的左边 unify.  
 Matching 的结果是得到一推 instantiated (特例化，即不包含变量) 的命题。
- **forward- or backward-chaining** (resolution):  
 由已知事实 推导出新结论，或反过来，判断某给定的新结论是否成立。  
 例如：是人(苏格拉底)  $\Rightarrow$  会死(苏格拉底)  $\wedge$  是人(苏格拉底)  
 可以推出：会死(苏格拉底)。

深度学习的特点，就是将

$$\text{状态}_t \vdash \text{状态}_{t+1} \quad (2)$$

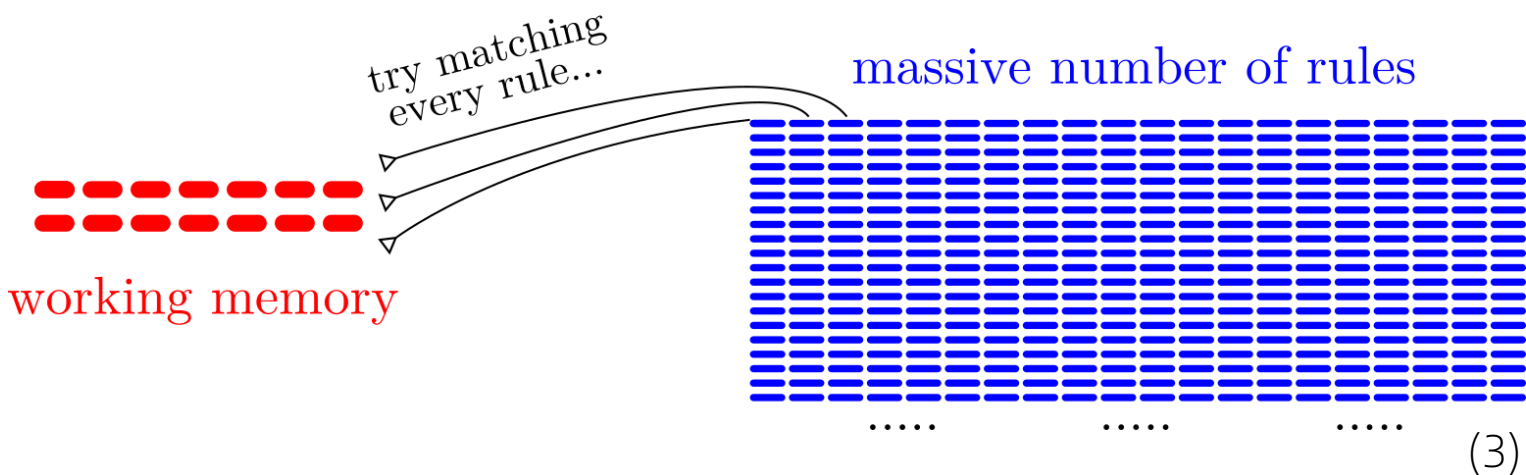
的逻辑推导过程，通通纳入进去一个非常复杂的非线性函数 (= 深度神经网络) 里面。这样做以后，上述的逻辑结构被 “mingled” 在一起，以至于很难分辨了。但也正是由于这种「大杂烩」，深度神经网络 将一套复杂的组合算法压缩成数量不算太多的一层层参数。它同时可以做 learning 和 inference 这两个动作。这种简单粗暴的方法，其实非常有效率，要超越它的速度并不容易！

我们知道 (或推测) 一个智能系统 应该具有 符号逻辑的结构。这点知识可不可以用来 约束/加速 深度神经网络？答案似乎是有可能的。现时 state-of-the-art 处理视觉的 CNN 和处理文字的 BERT，它们都有内部结构，**而不是 fully-connected**，而且这内部结构 对应于 被处理的资料的结构。因此我们有理由相信，逻辑结构 可以用来约束 深度神经网络的结构，达到加速。

②

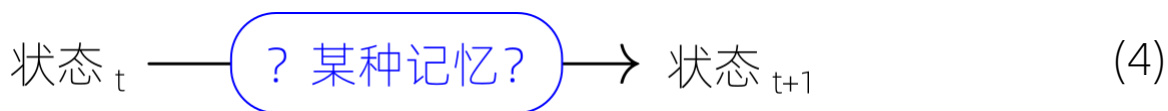
接下来我们详细一点看逻辑系统的结构：

Knowledge Base 里面有很多 rules，系统要将这些 rules 逐一 match with 系统状态 (= working memory) 里面的命题：

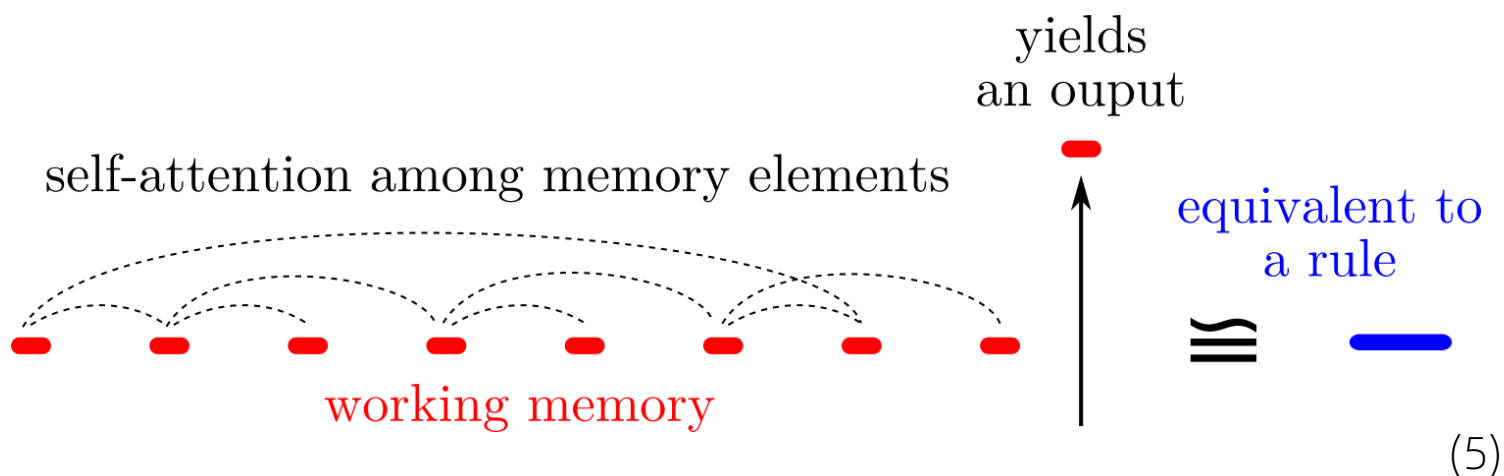


成功 matched 的 rules 可以导出新的结论，加进 working memory 的状态里面。

这个复杂的操作，完全被一个神经网络取代。或者可以更抽象地说：



而以 Transformer 来说，它是一种 输入元 之间 的记忆体（这记忆就储存在 Q, K, V 矩阵里），而它 **implicitly** 做到了 rules 的作用：



换句话说，Transformer 内部有某种（扭曲了的）逻辑 rules 的结构。那么很自然的问题就是：能否发掘更多 逻辑 / 逻辑系统 的结构？也就是说，公式 (4) 可以有怎样的代数结构约束？这个问题 可以参考 范畴逻辑 的理论，还有 经典 logic-based AI 系统的理论。

我们希望 勾画出公式 (4) 需要具备的代数约束，但暂时先用文字描述比较容易：

- 状态是 **颗粒化** 的，它是某集合的元素，元素之间可交换，也就是 Transformer 的 equivariance. （注意：Transformer 有 equivariance，但 equivariance 未必一定要用 Transformer 实现）
- **深度结构**：例如多层网络，每层是函数的复合 (composition). Transformer 也用了深度结构。
- 逻辑 包括了 **命题** 层次 和 **命题内部** 层次 的 颗粒化。后者是 **谓词** (predicate) 逻辑的结构，例如： *loves(John, Mary)*，也可以简单地将它视为 **代数元** 之间的 **乘积**，例如： *John · loves · Mary*，后者也叫做 “word”. （不同类别的代数元之间不一定容许乘积，因此有 groupoid 的概念，但暂时来说这细节不重要。）现时重点是如何将 这两层的 颗粒化 结构 施加到深度神经网络上。
- 逻辑推导 每步只产生 **一个** 新的结论（或其概率分布），然后这个新的结论，再加入到旧的状态中，作为一个命题集合的元素，而旧状态也要 **遗忘** 一些命题，否则需要无限记忆。这跟 Transformer 每次输出一列的 tokens 有点不同（虽然我们不太肯定 Transformer tokens 究竟对应于命题 还是 谓词 / 原子概念）。
- 逻辑 rule 通常只跟某几个前提有关，其它前提是 **无关** 的，例如： *眼睛好看 ∧ 鼻子好看 ∧ 嘴巴好看 ⇒ 帅*，跟 *有钱* 或 *穷* 无关。Transformer 的 **softmax** 结构似乎也可以排除一些无关的 tokens 的影响。
- (可能还有其他的结构特征.....)



⑤

在 范畴逻辑 里面有 **Beck-Chevalley** 条件和 **Frobenius** 条件，或许是我们所需的对称性？但细看之后，发觉还是不能解决问题..... For completeness, 我还是描述一下，没兴趣的可以略过。

首先考虑比较容易明白的 **Frobenius** 条件。在逻辑上，它等于说：

$$\exists x. [\phi \wedge \psi(x)] \equiv \phi \wedge \exists x. \psi(x). \quad (11)$$

由于 经典逻辑 AI 普遍使用  $\forall$  而忽略  $\exists$ ，我将上式改写成：

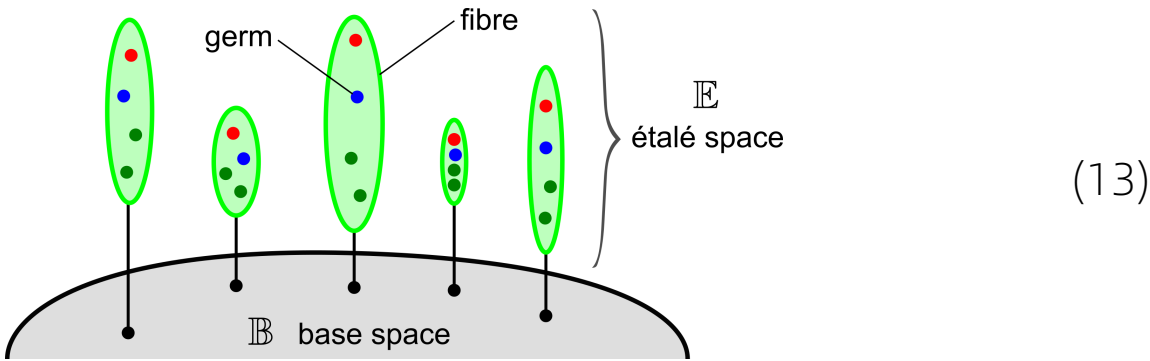
$$\forall x. [\phi \vee \psi(x)] \equiv \phi \vee \forall x. \psi(x). \quad (12)$$

但问题是，(12) 式的左边和右边，其对应的神经网络 (6) 是一样的（看不出有分别）。也就是说这个差别可能太 subtle 了，它并不影响我们实际 implement 的神经网络。

在逻辑里，任何变量 例如  $x, y$  等，必须被  $\forall$  或  $\exists$  quantify，否则不成为合法的句子。所以 表达 谓词结构的对称性，也很可能要涉及  $\forall$  或  $\exists$ 。

以前说过，谓词逻辑 带来 **fibration** 或 **indexing** 结构。Beck-Chevalley 和 Frobenius 条件 基本上是说，这 纤维结构 是 “preserved by re-indexing functors”.

这是 fibration 结构的示意图：



这整个结构 叫 **bundle**，而 **sheaf** 是 bundle 加上某个特殊的 拓扑结构。  
在  $(A, f)$  和  $(B, g)$  两个 bundle 之上可以定义 **fibred product** of  $A$  and  $B$  over  $I$ , 记作  $A \times_I B$ :

$$\begin{array}{ccc} A \times B & \xrightarrow{q} & B \\ \downarrow p & \searrow h & \downarrow g \\ A & \xrightarrow{f} & I \end{array}$$

(14)

其中  $h = f \circ p = g \circ q$ . 这也是一个 **pullback**.

**Beck-Chevalley** 条件是说 下面这幅图 commute:

$$\begin{array}{ccc} K \times J & \xrightarrow{u \times id} & I \times J \\ \pi \downarrow & & \downarrow \pi \\ K & \xrightarrow{u} & I \end{array}$$

(15)

其中  $\pi$  就是代表 量词  $\forall$  或  $\exists$  的 投影，它们是 weakening map  $\pi^*$  的伴随映射。

Beck-Chevalley 条件并不完全是空洞的；它有可能不成立。有一个反例是 Pitts 提出的：考虑  $X \times Y$ ，其中  $X = Y = \mathbb{N} \cup \{\infty\}$  亦即 自然数加上  $\infty$  作为 top element；但  $Y$  是用 discrete order，亦即所有 order 都是  $=$ .  $A$  是  $X \times Y$  上的关系： $A = \{(x, y) \in \mathbb{N} \times \mathbb{N} \mid x \leq y\}$ . 那么  $\exists y.(x, y) \in A$  会是整个  $X$  集合。如果考虑 DCPO 范畴，我们要求 fibration of Scott-closed subsets (ordered by inclusion) over DCPO.  $\exists y.A$  的 Scott closure 的条件是 它是一个 lower set closed under directed joins; 而这个 Scott closure 条件似乎不成立，因而导致 图 (15) 不 commute. (我对 Scott closure 的细节不太理解)

Lawvere 的工作将 Beck-Chevalley 条件变得更一般化：「简单」的  $\forall$  和  $\exists$  量词 是 weakening functor  $\pi^*$ （基于笛卡尔积）的伴随函子，但 Lawvere 将它扩充到任何 **substitution** functor  $u^*$ .