

“Easy” presentation

The logic route to strong AI

YKY

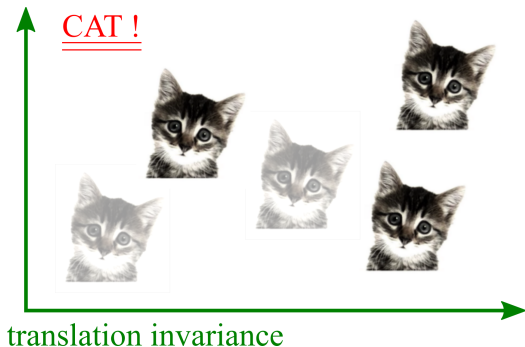
March 17, 2021

Table of contents

- 1 The success of CNN in computer vision
- 3 Symmetry and inductive bias
- 4 Richard Sutton's view
- 6 Doubts about logicism
- 7 Structure of logic
- 8 Symmetric neural networks
- 10 For example: symmetric NN for object recognition
- 11 Logicalization of BERT
- 12 Application: content-addressable long-term memory
- 13 Application: knowledge graphs
- 14 AGI 理论的突破

CNN 在机器视觉中的成功

- 在几何学上，视觉 具有 平移 不变性：



(1)

- Convolution 是一种具有平移不变性的运算：

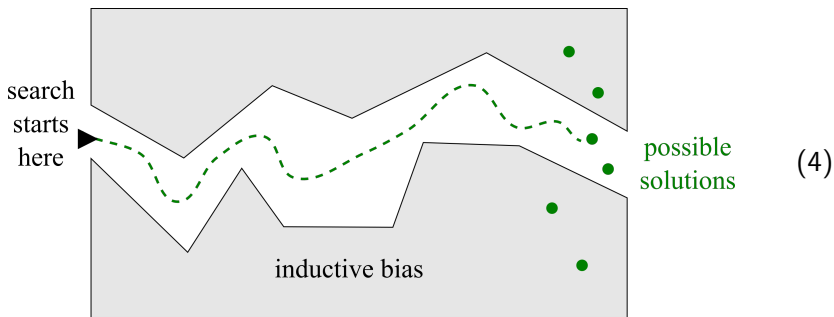
$$(T_x \circ f) * g = T_x \circ (f * g) \quad (2)$$

- Yann LeCun 等人 利用 CNN 的 对称性 加快了学习速度，成功地解决了机器视觉 的问题



Symmetry and inductive bias

- 在数学上, 对称性 经常能简化计算, 所以数学家 特别喜欢 对称
- 在机器学习中, 经常要引入 归纳偏好 (inductive bias), 缩小 搜寻空间:



- 往往如果 归纳偏好 选对了, 可以在短时间内找到答案, 否则问题是不可解的 (intractable)

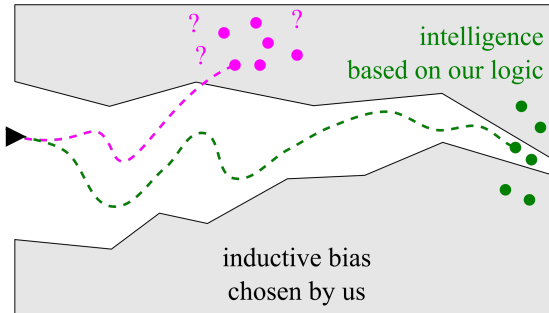
Richard Sutton 的观点

- Richard Sutton 认为，我们只需在 强化学习 的框架下 增加算力，就可以找到 strong AI



- 我們選擇的只是众多 形式逻辑 之中可能的一种：

intelligence based on "alternative" logics



(6)

- 这不只是一个「空想」的问题；事实上，世界各地的实验室 已经开始了对 AGI 不同形式的搜索！

对 逻辑主义 的质疑

- 很多人怀疑：人脑真的用 形式逻辑 思考吗？



(7)

其实人脑比我们想像中更接近逻辑

Structure of logic

- 我的想法是：在深度学习中引入 **逻辑** 的对称性，解决 strong AI 问题
- 因为人的思维 具有 **逻辑** 的结构，这个 inductive bias 可以帮助我们快速找到 the solution to strong AI
- 逻辑结构很复杂，但最粗略的 symmetry 是 命题的 **可交换律** (commutativity, or permutation invariance):

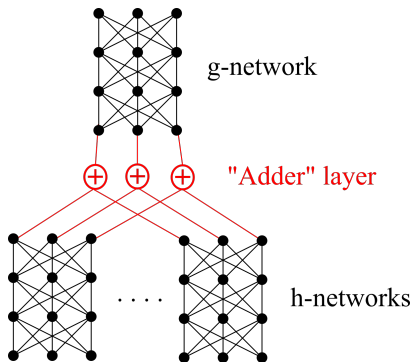
$$\begin{aligned} A \wedge B &\equiv B \wedge A \\ \text{下雨} \wedge \text{失恋} &\equiv \text{失恋} \wedge \text{下雨} \end{aligned} \tag{8}$$

- 它的重要性类似於 视觉中的 平移不变性
- 另一种讲法是：它将智能系统的 **思维状态** (mental state) 分拆成 一粒粒独立的 **命题** (propositions)

Symmetric neural networks

- Permutation invariance can be handled by **symmetric** neural networks
- 我浪费了两年时间试图解决这问题，却发现现在 3 年前已经有两篇论文解决了 [PointNet 2017] [DeepSets 2017]，而且数学水平比我高很多
- Any symmetric function can be represented by the following form (a special case of the Kolmogorov-Arnold representation of functions):

$$f(x, y, \dots) = g(h(x) + h(y) + \dots) \quad (9)$$



(10)

- Sym NN gives a powerful boost in efficiency $\propto n!$ where $n = \#inputs$
- The code for Sym NN is just a few lines of Tensorflow:

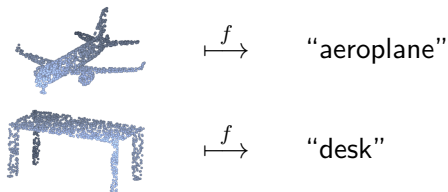
```
h = Dense(3, activation='tanh')
ys = []
for i in range(9):
    ys.append( h(xs[i]) )
y = Keras.stack(ys, axis=1)
Adder = Lambda(lambda x: Keras.sum(x, axis=1))
y = Adder(y)
g = Dense(3)
output = g(y)
```

(11)

- Very easy to adopt this to existing models such as BERT and reinforcement learning
- I have successfully tested it on the game of TicTacToe:
<https://github.com/Cybernetic1/policy-gradient>

For example: symmetric NN for object recognition

- Imagine objects represented as **point clouds**:

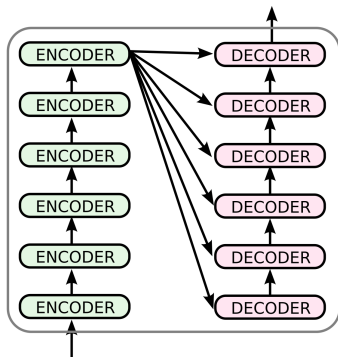


- It does not matter **in what order** the points are in a sequence; the function $f(x_1, \dots, x_n)$ is symmetric in its arguments (the points)
- Permutation invariance is **essential** for this to work

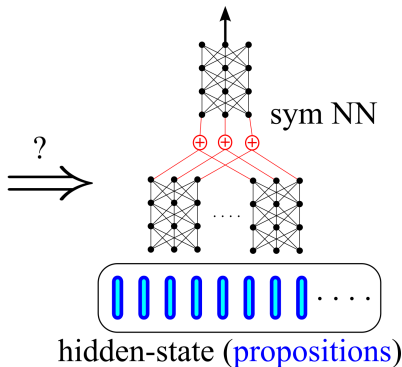
应用：BERT 的逻辑化

- 类似地，可以将 BERT 的 隐状态 变成 “set of propositions” 的形式，方法是将 原来的 decoder 变成 sym NN：

original BERT / Transformer



logic BERT ?

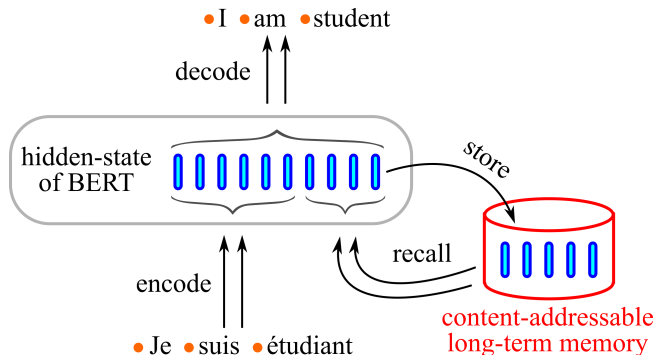


(12)

- 原来的 encoder 可以照旧使用，因为后半部改变了，error propagation 会令 representation 也改变
- 当然，这个想法有待实验证实 😊

应用：content-addressable long-term memory

- 以前 BERT 的隐状态 没有逻辑结构，我们不是很清楚它的内容是什么；逻辑化之后，BERT 内部的命题可以储存在 **长期记忆** 中：



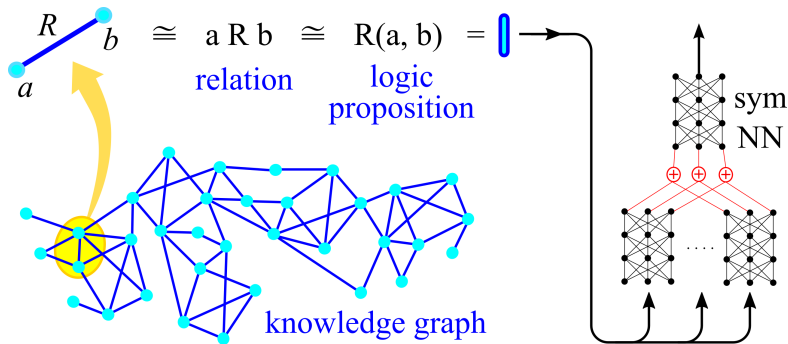
(13)

例如：「太阳是热的」、「水向下流」是经常正确的命题

- 这种系统 已非常接近 strong AI，而这是有赖 **逻辑化** 才能做到的
- Content-addressable memory 的想法来自 Alex Graves *et al* 的 Neural Turing Machine [2014]

应用：知识图谱 (knowledge graphs)

- 知识图谱 不能直接输入神经网络，它必需分拆成很多 edges，每个 edge 是一个 **关系**，也是一个 **逻辑命题**；也可以说 “graphs are isomorphic to logic”



- 而这些 edges 似乎必需用 **symmetric** NN 处理，因为它们是 permutation invariant

AGI 理论的突破

- 最近我成功地描述了 strong AI 和 逻辑之间的数学关系
这种数学关系是永恒不变的，它可以指导日后 AGI 的发展
请参看附件：《再谈一次 AI 的逻辑结构》
- 关于 Logic BERT 与 attention 的详细理论
可参看附件《Logic BERT》

References

多谢收看 😊

- [1] Alex Graves, Greg Wayne, and Ivo Danihelka. “Neural Turing Machines”. In: *CoRR* abs/1410.5401 (2014). arXiv: 1410.5401. URL: <http://arxiv.org/abs/1410.5401>.
- [2] Qi et al. “Pointnet: Deep Learning on Point Sets for 3D Classification and Segmentation”. In: *CVPR* (2017). <https://arxiv.org/abs/1612.00593>.
- [3] Zaheer et al. “Deep sets”. In: *Advances in Neural Information Processing Systems* 30 (2017), pp. 3391–3401.

Illustration credits:

- Translation invariance, from Udacity Course 730, Deep Learning (L3 Convolutional Neural Networks ▷ Convolutional Networks)