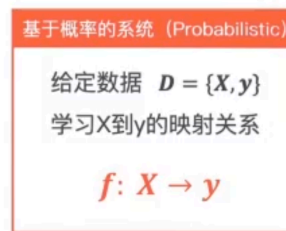


Two Main Branches of Learning



数据量：
少量数据/信号数据 \Rightarrow 专家系统
大量的数据 \Rightarrow 概率统计系统

1. expert systems

专家系统 = 推理引擎 + 知识 (类似于程序 = 数据结构+算法)

- 利用**知识和推理**来解决决策问题 (Decision Making Problem)
- 全球第一个专家系统叫做DENDRAL, 由斯坦福大学学者开发于70年代

核心：知识库

expert 把专家的经验转换成计算机能识别的数据。

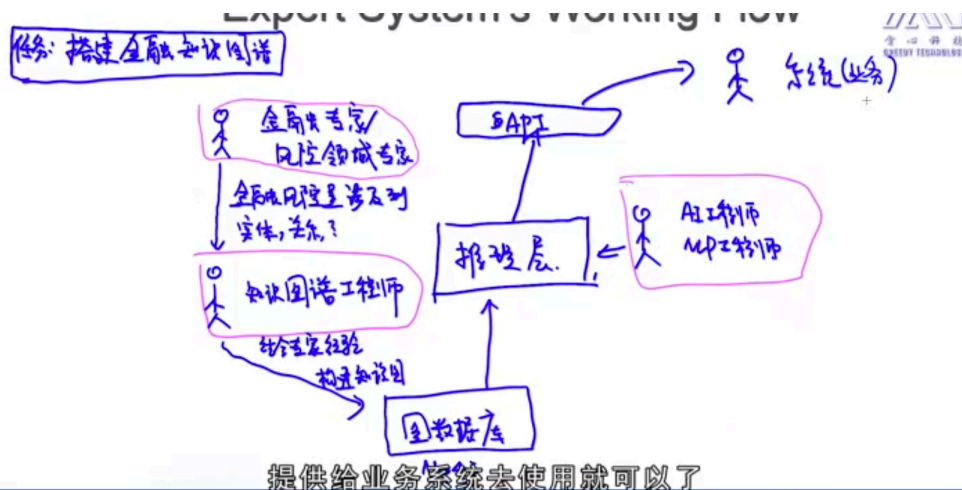
由此，**knowledge engineer**。能够 encode expertise。把能够识别的知识放到 knowledge base 里面。

System engineer/algorithm engineer, 让专家系统体现 AI 的功能，负责推理引擎的部分。

1.1 working flow

任务：搭建金融知识图谱

目的：风控



1.2 properties of expert systems

处理不确定性

知识的表示（知识图谱）

可解释性（深度学习的网络其实可解释性很差）

可以做知识推理（可以在上层搭建一个知识引擎）

2. logical inference

Given : A Rule base contains following Rule set

<u>Facts</u>	Rule 1: If A and C	Then F
	Rule 2: If A and E	Then G
	Rule 3: If B	Then E
	Rule 4: If G	Then D
Problem : Prove		
If A and B true		Then D is true

--》专家系统需要大量的离散数学的知识
解决该证明问题--> 两种方法

2.1 forward training

每次都循环所有的 rule，一条一条的 check 能不能获得新的知识

① 1st Iteration

Rule1 $\rightarrow \{ \}$
Rule2 $\rightarrow \{ \}$
Rule3 $\rightarrow \{ E \} \Rightarrow \text{true}$

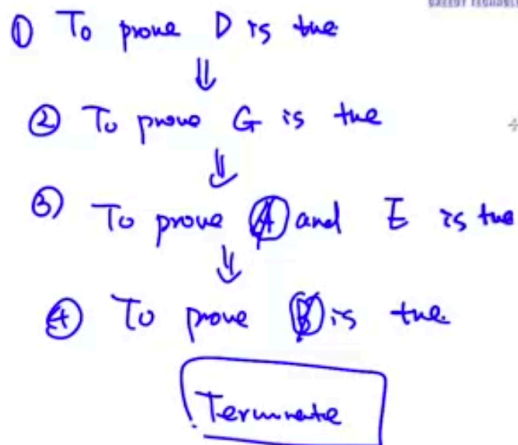
② 2nd iteration

Rule1 $\rightarrow \{ E \} \Rightarrow \text{true}$
Rule2 $\rightarrow \{ E, G \} \Rightarrow \text{true}$
Rule3 $\rightarrow \{ E, G \} \Rightarrow \text{true}$
Rule4 $\rightarrow \{ E, G, D \} \Rightarrow \text{true}$

Terminated

2.2 backward training

反方向证明，为了证明 D 是正确的，所以我们需要证明 AB 是正确的。
所以现在，已知 D 正确。

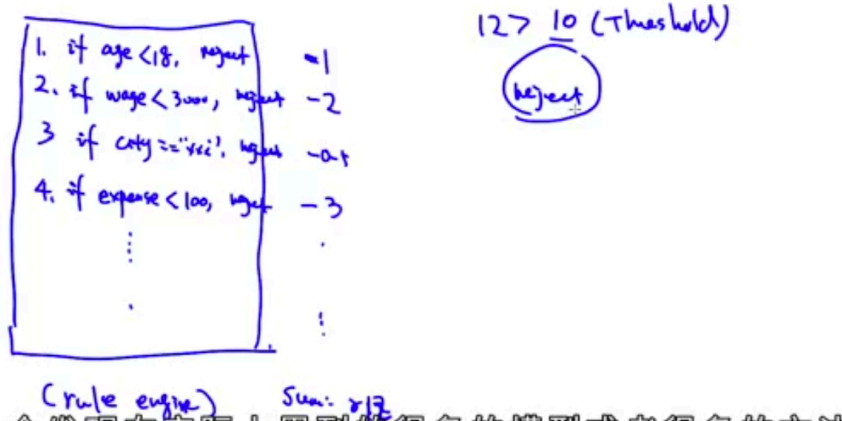


3. drawbacks of expert systems

设计大量的规则 design lots of rules
 需要领域专家来主导 heavily rely on domain expert
 可移植性差 limited transferability to other domain
 学习能力差 inability to learn
 人能考虑的范围是有限的 human capacity is limited

4. case study: risk control

问题：根据用户的信息，决定要不要放贷



5. some interesting problems

5.1 Logical inference (above)
 Conflict resolution
 Minimum size of rules