实验报告

141242026 刘驭壬

1. 实验环境

python 3.5

2. 实验进度

我独立完成了作业要求的所有内容（未参考任何他人和网上的代码）：

- 实现正向推理和反向推理
- 得到了问题要求的答案：长颈鹿
- 两种方法均输出了推理过程

3. 实验结果

题目答案为长颈鹿，推理过程记录在 generate.txt 和 validate.txt 中。

4. 问题求解

问题的输入：第一行为动物的种类，第二行为推理规则的数量 k，接下来 k 行为所有的推理规则，#划分了条件和结论。接下来 1 行表示需要回答问题的数量，之后是问题的条件。

问题的输出：是哪一种动物（如果均不是则不输出）与推理过程。

```
tiger leopard zebra giraffe penguin ostrich seagull
15
mammal carnivore tawny spotty#leopard
mammal carnivore tawny blackstrip#tiger
ungulate longneck longleg spotty#giraffe
ungulate blackstrip#zebra
bird longneck longleg nfly blackwhite#ostrich
bird swim nfly blackwhite#penguin
bird swim fly#seagull
mammal ruminate#ungulate
hairy#mammal
milk#mammal
feather#bird
fly egg#bird
eatmeat#carnivore
laniarii claw eyeforward#canivore
mammal hoof#ungulate
1
spotty longneck longleg milk hoof
```

算法实现：
我用 python 实现了 ppt 上的两页伪代码，没什么特别要说的地方，详细见代码。但是实现过程中我发现反向推理的伪代码是错误。因为参数 X 可能会是一个列表。需要添加一个 for 循环判断是否列表中的每个元素都满足。以及实现过程中我省略了以下这种情况，这对算法正确性没有影响。

else if C is false in the database

then do nothing

数据驱动：正向推理

```
procedure generate;
begin
 identify the set S of applicable rules;
 while S is non-empty
  do begin
   select a rule R from S;
  apply R;
   if the problem is solved by the application of R
   then indicate SUCCESS
   else call 'generate' recursively
  remove R from S and undo the effect of applying R
 end
end;
```

本质是深度搜索！

N能被12整除，也能被20整除

迭代次数1：匹配产生式，冲突集合R1和R2，选择R1执行

迭代次数2：在工作内存中产生事实“能被6整除”，匹配产生式，应用规则R3

迭代次数3：失败，回溯，将事实“能被6整除”从工作内存中去除。应用规则R2……

目标驱动：反向推理

```
function validate(X:expression): boolean;
 var result:boolean:
 begin
  result:=false;
  identify the set of applicable rules S which have X on the
right-hand side;

 While (result = false) and (S is non-empty)
 do begin
  select and remove a rule R from S;
  C:=the condition part of R;
  if C is true in the database
    then result:=true
   else if C is false in the database
       then do nothing
    else if validate (C) is true
       then result:=true
 end;
 validate:=result
end;
```

本质也是深度搜索！

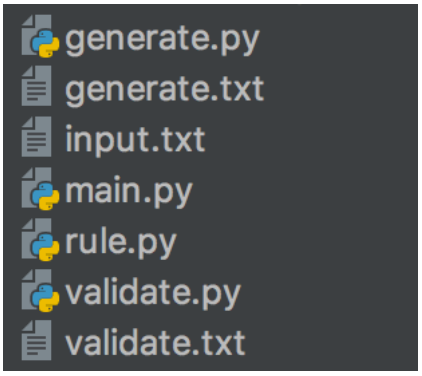
N能被5整除

迭代次数1：匹配产生式，选择R4执行

迭代次数2：匹配“能被10整除”，应用规则R2

迭代次数3：在数据库中匹配了R2的左部“能被20整除”

文件结构如下图：



generate.py 中实现了正向推理
generate.txt 记录了正向推理的过程
input.txt 是输入文件
main.py 是主函数
rule.py 实现了 Rule 的数据结构
validate.py 实现了反向推理
validate.txt 记录了反向推理的过程