同济大学计算机系

人工智能课程实验报告

Horn 子句归结

授课老师 武妍

姓名学号 孙文丽 1752844、刘兵 1752397、李辉 1652286

专　　业 计算机科学与技术

一、 实验概述

（一）、实验目的

熟悉和掌握归结原理的基本思想和基本方法，通过实验培养学生利用逻辑方法表示知识，并掌握采用机器推理来进行问题求解的基本方法。

（二）、实验内容

1. 对所给问题进行知识的逻辑表示，转换为子句，对子句进行归结求解。
2. 选用一种编程语言，在逻辑框架中实现 Horn 子句的归结求解。
3. 对下列问题用逻辑推理的归结原理进行求解，要求界面显示每一步的求解过程。

破案问题：在一栋房子里发生了一件神秘的谋杀案，现在可以肯定以下几点事实：

(a)在这栋房子里仅住有A,B,C三人；

(b)是住在这栋房子里的人杀了A；

(c)谋杀者非常恨受害者；

(d)A所恨的人，C一定不恨；

(e)除了B以外，A恨所有的人；

(f)B恨所有不比A富有的人；

(g)A所恨的人，B也恨；

(h)没有一个人恨所有的人；

(i)杀人嫌疑犯一定不会比受害者富有。

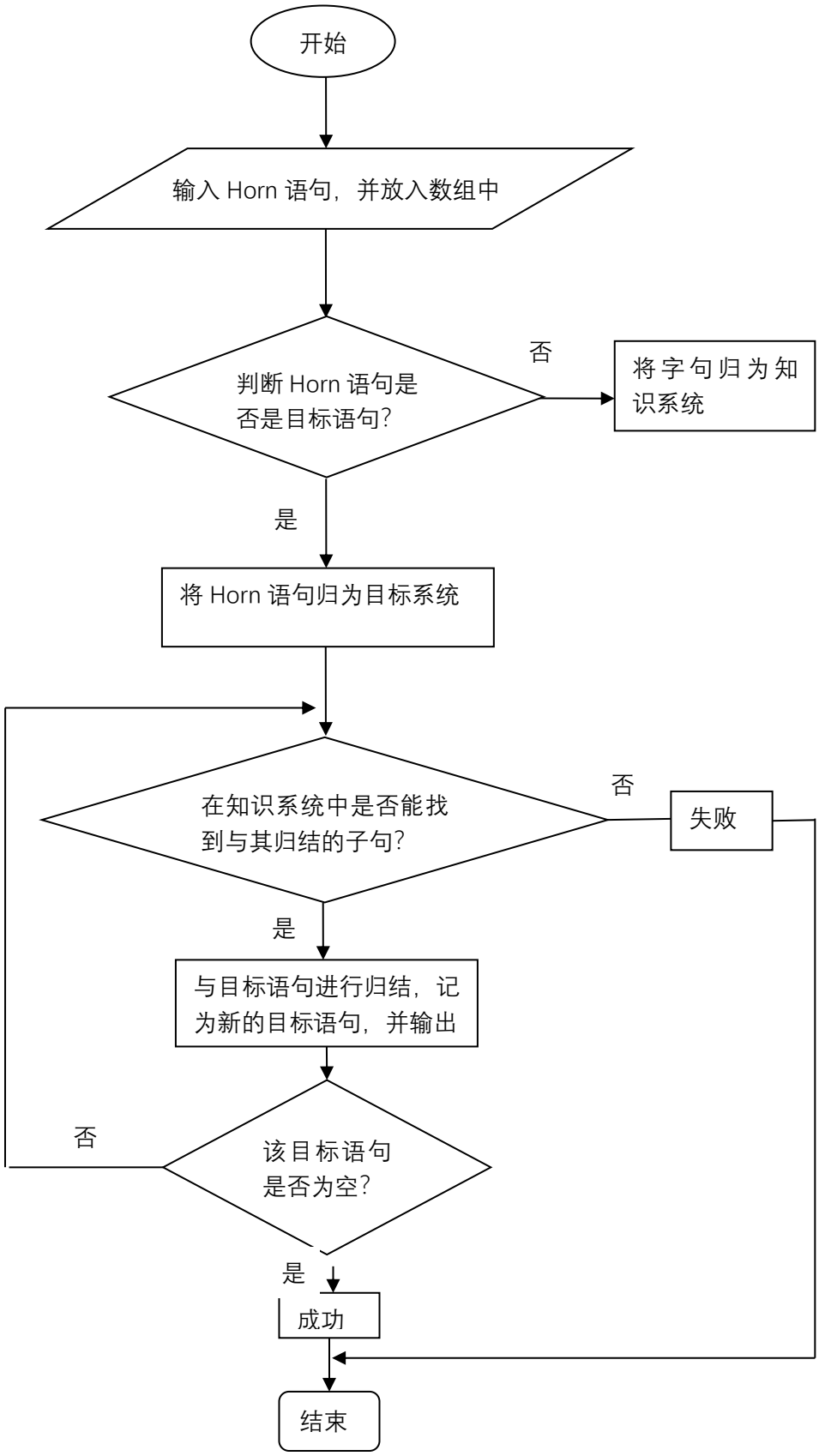
为了推理需要，增加如下常识：(j)A不等于B。

问：**谋杀者是谁？**

二、 实验方案设计

(1) 总体设计思路与总体架构

- 输入事实/过程
- 将一阶逻辑语言化为子句
- 输入目标
- 调用归结法证明
- 利用置换合一，匹配目标和过程的后件，根据前件更新目标
- 目标变成空语句，结论成立
- 输出证明过程


开始
输入 Horn 语句，并放入数组中
判断 Horn 语句是否是目标语句?
否
将字句归为知识系统
是
将 Horn 语句归为目标系统
在知识系统中是否能找到与其归结的子句?
否
失败
是
与目标语句进行归结，记为新的目标语句，并输出
该目标语句是否为空?
否
是
成功
结束

(2) 核心算法及基本原理：

采用的数据结构：线性结构、数组（Python 代码里面分别用到列表和元组存储 Horn 子句和逻辑语句的函数表示）

算法：本实验并没有涉及到很复杂的算法，其过程是模拟手写推导过程，需要将 Horn 子句，输入执行 main 文件中，最主要的两个函数是 mgu(self, other, limit=None)（实现置换合一）和 resolution(self, exp: Expression)（子句归结），都是通过遍历所有子句（从数组里面取出），尝试所有可能的表达式/单元，进行置换合一和归结，其中，子句归结过程主要通过递归证明置换后的新目标，返回一条证明步骤。

关于 Horn 子句的解释：

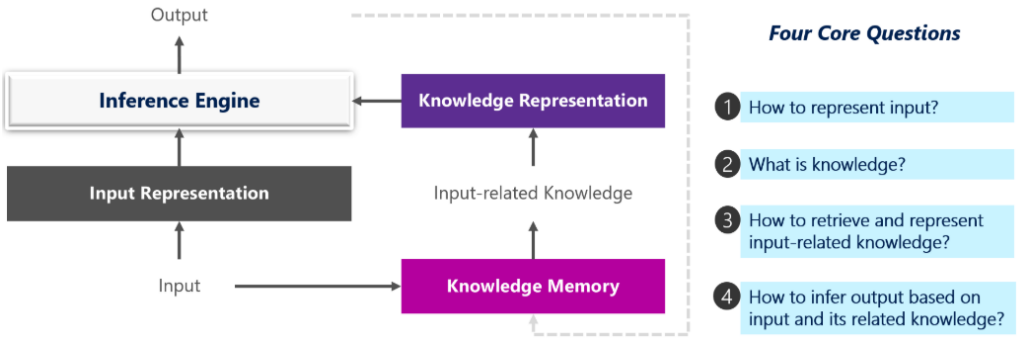
Horn子句$P \vee \sim Q_1 \vee \sim Q_2 \vee \ldots \vee \sim Q_n$通常表示为
$P \leftarrow Q_1, Q_2, \ldots, Q_n$。
很显然，Horn子句必取下列四种形式之一：
(1)$P \leftarrow Q_1, Q_2, \ldots, Q_n$ $(n \neq 0)$
称为过程,P称为过程名,$\{Q_1,\ldots,Q_n\}$称为过程体,Q_i为过程调用
(2)P← (上式中n=0)----事实
(3)←$Q_1,Q_2,\ldots,Q_n$ (n≠0)----目标,全部由过程调用组成,
常用来表示询问
(4) □----停机语句,表示程序执行（成功）终止。

网络查找的解释

Horn 子句可以表示出事件之间的逻辑关系，通过结合 Python 高效的处理正则语言的功能，可以简化证明过程。

关于机器推理：

机器推理（Machine Reasoning），是指基于已有知识对未见问题进行理解和推断，并得出问题对应答案的过程。根据该定义，机器推理涉及 4 个主要问题：（1）如何对输入进行理解和表示？（2）如何定义知识？（3）如何抽取和表示与输入相关的知识？（4）基于对输入及其相关知识的理解，如何推断出输入对应的输出？下图给出机器推理的整体框架：



机器推理整体框架

本次子句归结实验，总体上是依照机器推理的思路框架展开，可归类为机器推理里的基于推理的事实检测，具体推理步骤在下面依次进行展开。

(3) 模块设计：本实验的具体模块设计

两个 py 文件：

main.py 为入口文件，用于输入输出，控制流程

init.py 包含 3 个类，分别为 **Unit**(过程体或函数)，**Expression**（事实，过程，目标），**Engine**（归结法核心）

(4) 其他创新内容或优化算法

创新点：开始在处理非语句的时候考虑在肯定语句前加上“<-”来表示否定，例如：“A 所恨的人，C 一定不恨。”表示为“<-hate(C,*X)<-hate(A,*X)”结果这样实现起来会很麻烦，因为一个 Horn 语句中有两个“<-”，给正则语句分析造成了混淆，最后考虑在原肯定语句的基础上，在前面加上 n 表否定，如“nhate(C,*X)<-hate(A,*X)”，事实上还是一个肯定句，但语义是否定的，通过这种方法，使得证明步骤的实现方便了很多。

三、 实验过程

(1) 环境说明：

操作系统：windows10

开发语言：Python 3.7

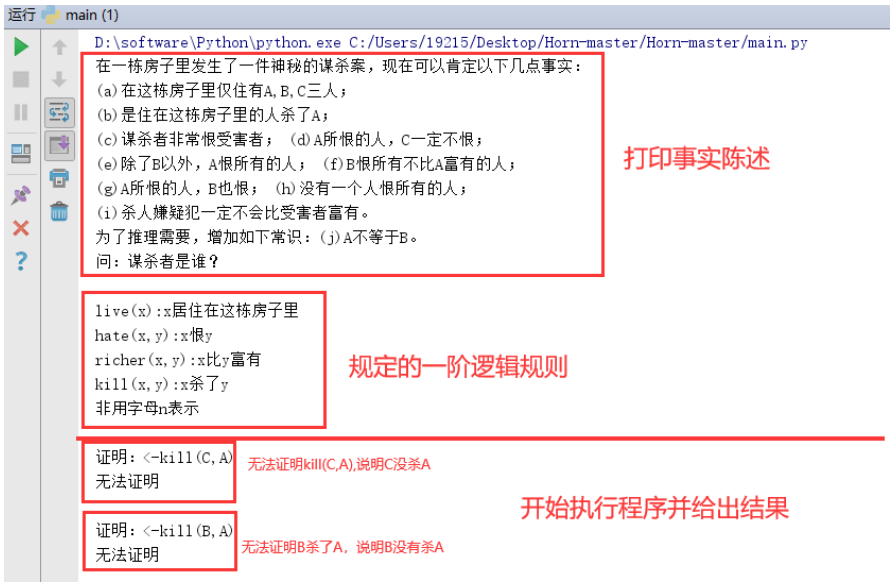
开发环境：PyCharm Professional 2017

核心使用库：re、copy

(2) 主要函数清单

函数名	作用域	功能
def is_variable(cls, arg: str) -> bool	Unit 类	判断事实或目标集中是否含有变量
def update_args(self, mapping: dict)	Unit 类	在置换的过程中替换变量
def mapping_args(self, other)	Unit 类	寻找可置换的过程（函数） self 和 other，分别需要修改什么变量，以 other 的为主
def __eq__(self, other)	Unit 类	判断当前事实和目标是否相等
def __str__(self)	Unit 类和 Expression 类	将事实集转换为字符串
def update_args(self, mapping: dict)	Expression 类	用 mapping 中的置换对，更新表达式
def remove_all_from_body(self, unit: Unit)	Expression 类	清除表达式中所有 unit
def clear_same_unit(self)	Expression 类	移除相同的 unit
def mgu(self, other, limit=None)	Expression 类	置换合一，利用过程更新目标 self 和 other 的归结结果 self 是目标（待证明）
def resolution(self, exp: Expression)	Engine 类	实现子句归结的过程
def proof(self, exp: str)	Engine 类	返回归结过程，存到数组中，如果证明失败，则返回 None

(3) 实验结果展示


运行 main (1)
D:\software\Python\python.exe C:/Users/19215/Desktop/Horn-master/Horn-master/main.py
在一栋房子里发生了一件神秘的谋杀案，现在可以肯定以下几点事实：
(a)在这栋房子里仅住有A,B,C三人；
(b)是住在这栋房子里的人杀了A；
(c)谋杀者非常恨受害者； (d)A所恨的人，C一定不恨；
(e)除了B以外，A恨所有的人； (f)B恨所有不比A富有的人；
(g)A所恨的人，B也恨； (h)没有一个人恨所有的人；
(i)杀人嫌疑犯一定不会比受害者富有。
为了推理需要，增加如下常识：(j)A不等于B。
问：谋杀者是谁？
打印事实陈述
live(x):x居住在这栋房子里
hate(x,y):x恨y
richer(x,y):x比y富有
kill(x,y):x杀了y
非用字母n表示
规定的一阶逻辑规则
证明：<-kill(C,A)
无法证明
无法证明kill(C,A),说明C没杀A
开始执行程序并给出结果
证明：<-kill(B,A)
无法证明
无法证明B杀了A，说明B没有杀A

注：

前面带“*”的是变量，A,B,C 是常量，谓词前加字母'n'表示谓词取非，形式上仍然是肯定形式。

实验结论：

通过程序执行结果可知，最终 kill(A,A)的证明过程得到了空子句，说明是 **A 自杀**，而非他杀，得到的结果与手写推导结果一致。

四、 总结

(1) 实验中存在的问题及解决方案

开始在处理非语句的时候考虑在肯定语句前加上“<-”来表示否定，例如：“A

所恨的人，C 一定不恨。”表示为“<-hate(C,*X)<-hate(A,*X)”结果这样实现起来会很麻烦，因为一个 Horn 语句中有两个“<-”，给正则语句分析造成了混淆，最后考虑在原肯定语句的基础上，在前面加上 n 表否定，如“nhate(C,*X)<-hate(A,*X)”，事实上还是一个肯定句，但语义是否定的，通过这种方法，使得原来的混淆问题得到很好的解决。

(2) 心得体会

通过本次实验，使我们加深了对子句归结求解过程的理解，并且掌握了基于 Horn 子句归结的方法。结合机器推理的过程进行实验，提高了我们用逻辑方法表示知识的能力。同时，过程中的合作也是很关键的，只有每位组员融入到实验中来，才能达到实验效果，提高每位组员对所学知识的掌握和提升。

(3) 后续改进方向

本次实验是机器推理的一个简单开端，基于已知知识对未见问题进行推导求解。不过本实验只能达到先做出假设，再进行论证过程，如果最后归结结果为空，则假设为真，依照这种方法，需要作出三次假设，进行三次推导。这个过程还是比较复杂的，如果一个房子里有 10 个人，就要作出 10 次假设和推导，显然不利于实验证明的简洁性。所以后续可以考虑引入基于问题求解的知识推导体系 。即采用 $Answer(u) \vee kill(u,A)$这种形式，将其取反，加入到字句集中，这样，就能达到一步求解的目的。

(4) 总结

本实验以 Horn 子句定义一阶逻辑表示的知识，通过正则表达式对输入进行理解和表示，对于每一条子句，程序都会递归的寻找子句集中是否存在满足归结条件的子句，若存在，通过置换合一，进行子句归结，若在过程中得到空子句，则论证了假设的正确性。这就是机器推理过程在本实验中的具体表示和实现。

参考书籍：

人工智能：一种现代的方法（第三版）（罗素）

离散数学（第 3 版）（屈婉玲、耿素云、张立昂）

《算法设计与分析》屈婉玲著，清华大学出版社

Python 编程：从入门到实践（[美]埃里克.马瑟斯）

参考网址：

https://blog.csdn.net/qq_36306833/article/details/82978607

https://wenku.baidu.com/view/dfae2160eefdc8d376ee32e5.html

成员分工与自评：

成员	分工
孙文丽	归结过程实现，代码编写
刘兵	PPT 制作、汇报实验
李辉	总结实验，撰写实验报告

1、 孙文丽

本次实验我主要负责编写代码，实现 Horn 子句的归结求解。核心程序采用面向对象的程序设计方法，实现归结功能，返回证明过程。首先，需要将破案问题化为 horn 子句的形式，再根据语义对过程和目标进行否定等操作，将事实、过程、目标作为输入写入主程序。归结的过程采用枚举的方式，寻找和目标函数名（过程名）一致的过程，然后进行置换合一得出新的目标，重复上述步骤，当目标与事实一致时停止归结。遇到的问题是在形式化的阶段，如果不变换目标和过程，用归结法无法证明。很开心能够负责代码部分，因为有组员的帮助，所以实验完成得比较顺利。

2、 刘兵

在本次子句归结实验中，我负责汇报 PPT 制作，积极参与了本次实验。通过对本次实验结果的整合，我加深了对归结原理的理解，熟悉了如何利用逻辑方法表示知识，使用机器推理来进行问题求解的方法。

3、 李辉

本次实验我负责实验总结，撰写实验报告，以及配合组员共同完成实验。写报告的过程中，对 Horn 子句形式有了清晰的了解，Horn 子句能够表达一阶逻辑，但形式又不同于课程中的一阶逻辑表达形式。在 python 中，对 Horn 子句的形式有着严格的要求，

python 高效的正则处理能力给基于 Horn 子句的一阶逻辑推导带来了极大的方便。同时，对一阶逻辑推导及其具体的置换合一和归结过程有了更进一步的了解。也了解了关于机器推理的一些基本知识，本次实验是对机器推理的简单应用，是机器推理的一个很小的应用方面，基于已知事实进行问题检测。机器推理在人工智能领域有着举足轻重的作用，在语义分析、自然语言推理等方面有着很大的发展空间。