PART Ⅱ
ARTIFICIAL INTELLIGENCE AS REPRESENTATION AND SEARCH

A PROPOSAL FOR DARTMOUTH SUMMER RESEARCH PROJECT ON ARTIFICIAL INTELLIGENCE

§ 2. The predicate calculus

2.0 Introduction

2.1 The Propositional Calculus

2.2 The Predicate Calculus

2.3 Using Inference Rules to Produce Predicate Calculus Expressions

2.4 Application: A Logic-Based Financial Advisor

2.0 Introduction

- Predicate calculus is a representation language for artificial intelligence (AI).
- Its advantages include:
  - a well-defined （明确定义的） formal semantic（形式语义）
  - sound（可靠的） and complete（完备的） inference rules.

2.1 The Propositional Calculus

2.1.1 Symbols and Sentences

- **Definition**

propositional calculus（命题演算）symbols :

- **Propositional symbols: P, Q, R, S…**
- **Truth（真值） symbols: truth, false**
- **Connectives（连接词）:**

  $\wedge$, $\vee$, $\neg$, $\rightarrow$, $\equiv$

- **Definition**

  propositional calculus sentences

  - Every propositional symbol and truth symbol is a sentence.（true, P, Q …)
  - The **negation**（非，否定）of a sentence is a sentence. （¬P, ¬true)
  - The **conjunction**（与，合取）of two sentences is a sentence. （P∧¬P)
  - The **disjunction**（或，析取）of two sentences is a sentence. （P∨¬P)

- The **implication**（蕴含）of one sentence form another sentence is a sentence. ($P \rightarrow Q$)
- The **equivalence**（等价）of two sentences is a sentence. （ $P \wedge Q \equiv R$)

Legal sentences are also called **well-formed formulas**（合适公式）or **WFFs**.

- The symbols ( ) and [ ] are used to group symbols into sub expressions and so to control their order of evaluation and meanings.
- For example:

  $(P \vee Q) \equiv R$ 与 $P \vee (Q \equiv R)$ 含义完全不同。

- An expression（表达式） is a sentence,or well-formed formula,of the propositional calculus if and only if it can be formed of legal symbols through some sequence of these rules.
- Example: a well-formed sentence

  $((P \wedge Q) \rightarrow R) \equiv \neg P \vee \neg Q \vee R$

2.1.2 The semantics of the propositional calculus

- **Interpretation（解释）: is the mapping（映射）from the propositional symbols into the set { F， T }**

Definition
propositional calculus semantics :

- The symbol true is always assigned（指派） T, and the symbol false is assigned F;
- The truth assignment（真值指派） of negation, ¬P, where P is any propositional symbol, is F if the assignment to P is T; and is T if the assignment to P is F;
- The truth assignment of conjunction, is T only when both conjuncts have truth value T; otherwise it is F;

- **The truth assignment of disjunction, is F only when both disjuncts have truth value F; otherwise it is T;**
- **The truth assignment of implication, is F only when the premise is T and the truth value of the consequent is F; otherwise it is T;**
- **The truth assignment of equivalence, is T only when both expressions have the same truth assignment for all possible interpretations; otherwise it is F.**

- **truth tables（真值表）：**

The truth assignments（真值指派） of compound propositions（复合命题）are often described by truth tables.

P	Q	$\neg P$	$\neg P \vee Q$	$P \rightarrow Q$	$(\neg P \vee Q) \equiv (P \rightarrow Q)$
T	T	F	T	T	T
T	F	F	F	F	T
F	T	T	T	T	T
F	F	T	T	T	T

- Some laws:
  - $\neg(\neg P) \equiv P$
  - $(P \vee Q) \equiv (\neg P \rightarrow Q)$
  - The contrapositive law（换质换位律）:

    $(P \rightarrow Q) \equiv (\neg Q \rightarrow \neg P)$
  - De morgan’s law（德摩根定律）:

    $\neg(P \vee Q) \equiv (\neg P \wedge \neg Q)$
  - De morgan’s law （德摩根定律） :

    $\neg(P \wedge Q) \equiv (\neg P \vee \neg Q)$
  - The commutative law （交换律） :

    $(P \vee Q) \equiv (Q \vee P)$

- **Some laws:**
  - **The commutative law（交换律）：**

    $$(P \wedge Q) \equiv (Q \wedge P)$$

  - **The associative law（结合律）：**

    $$((P \wedge Q) \wedge R) \equiv (P \wedge (Q \wedge R))$$

  - **The associative law （结合律）：**

    $$((P \vee Q) \vee R) \equiv (P \vee (Q \vee R))$$

  - **The distributive law（分配律）：**

    $$P \wedge (Q \vee R) \equiv (P \wedge Q) \vee (P \wedge R)$$

  - **The distributive law （分配律）：**

    $$P \vee (Q \wedge R) \equiv (P \vee Q) \wedge (P \vee R)$$

命题的局限性

- 命题逻辑的表示方法有较大的局限性，它无法描述客观事物的结构及逻辑特征，也不能把不同事物的共同特征表述出来。

例如："杨青是教师"和"李文是教师"，用命题逻辑表示时，无法把两人都是教师这一共同特征表示出来。

思考：试比较命题逻辑和谓词逻辑！

2.2 The Predicate Calculus

- In propositional calculus（命题演算）, P denotes（表示） the entire sentence:

  "it rained on Tuesday."

- We can create a predicate "weather" that describes the relationship between a date and weather:

  weather(tuesday, rain)

  weather(X, rain)

2.2.1 The syntax of predicates and sentences

- **Definition Predicate calculus symbols**

The alphabet（字符表） that makes up the symbols of the predicate calculus consists of :

1. The set of letters, both upper and lower case
2. The set of digits：0, 1, … 9
3. The underscore：_

Symbols in the predicate calculus begin with a letter and are followed by any sequence of these legal characters.

- Legitimate（合法的） predicate calculus symbols :

  George fire3 torm_and bill

- Not legal symbols :

  3jack ab% “no blanks allowed”

- 注意（1）提高可读性。（2）谓词演算的符号可以表示变量、常量、函数或谓词。

- Definition symbols and terms（项）

predicate calculus symbols include:

- Truth symbols true and false (reserved symbols)
- Constant symbols are symbol expressions having the first character lowercase.
- Variable symbols are symbol expressions beginning with an uppercase character.
- Function symbols are symbol expressions having the first character lowercase.

- A function expressions consists of a function constant of arity（元数） n, followed by n terms t_1, t_2, …. t_n, enclosed in parentheses and separated by commas.
- A predicate calculus term is either a constant, variable, or function expression.

For example:

- **Constant: cat, tree, tall**
- **Variable: X, George**
- **Function (arity):**

  plus(2, 3), times(6, 9),

  father(x), mother(sarah)

- **Definition: Predicates and atomic sentence(原子语句):**

  - **Predicate symbols are symbols beginning with a lowercase letter.**
  - **An atomic sentence is a predicate constant of arity n, followed by n terms, t1,t2…tn, enclosed in parentheses and separated by commas.**
  - **The truth value, true and false, are also atomic sentences.**

- Examples of predicates :

  likes(george, kate);   likes(X, george)

  likes(george, sarah, tuesday);

  likes(X, X);

  friends(father(david), father(andrew));

- the variable quantifiers（量词） ∀and ∃

  First order predicate calculus includes two symbols, ∀and ∃, that constrain(约束) the meaning of a sentence containing a variable.

- Example:

  ∀X likes(X,icecream) ;

  ∃Yfriends(Y,peter);

- **Definition**

  predicate calculus sentences

- **Every atomic sentence is a sentence.**
- **If s is a sentence, then so is its negation, ¬s.**
- **If s1 and s2 are sentences, then so is their conjunction, s1∧s2.**
- **If s1 and s2 are sentences, then so is their disjunction, s1∨s2.**

- **If s1 and s2 are sentences, then so is their implication, s1→s2.**
- **If s1 and s2 are sentences, then so is their equivalence, s1≡s2.**
- **If X is a variable and S is a sentence, then ∀XS is a sentence.**
- **If X is a variable and S is a sentence, then ∃XS is a sentence.**
- 谓词演算中的语句由以上方式归纳定义。

- **Example:**

  Let plus be function symbols of artiy 2 and let equal be predicate symbols with arity 2.

  plus (two, three);

  equal (plus (two, three), seven);

- **A recursive algorithm, verify_sentence, suggest a method for verifying that an expression is a sentence.**

2.2.2 A semantics for the predicate calculus

- The truth（真值） of expressions depends on the mapping of constants, variable, predicates, and functions into objects and relations in the domain of discourse（论域）.
- The truth of relationships in the domain determines the truth of the corresponding expressions.
- For example:

  friends(george, susie)

  friends (george, kate)

- **Definition Interpretation（解释）**

Let the domain D be a nonempty set.

An interpretation over D is an assignment （指派） of the entities of D to each of the constant, variable, predicate, and function symbols of a predicate calculus expression, such that:

1. Each constant is assigned an element of D.
2. Each variable is assigned to an nonempty subset of D; these are allowable substitutions （置换） for that variable.

3. Each function f of arity m is defined on m arguments of D and defines a mapping from D^m into D

4. Each predicate p of arity n is defined on n arguments from D and defines a mapping form D^n into {T, F}

- Given an interpretation, the meaning of an expression is a truth value assignment over the interpretation.

- **Definition**

truth value of predicate calculus expressions :

Assume an expression E and an interpretation I for E over a nonempty domain D. The truth value for E is determined by :

1. The value of a constant is the element of D it is assigned to by I.

2. The value of a variable is the set of elements of D it is assigned to by I.

3. The value of a function expression is that element of D obtained by evaluating the function for the parameter values assigned by the interpretation.

4. The value of truth symbol “true” is T and ”false” is F.

5. **The value of an atomic sentence is either Tor F, as determined by the interpretation I.**
6. **The value of the negation of a sentence is T if the value of the sentence is F and is F if the value of the sentence is T.**
7. **The value of the conjunction of two sentences is T if the value of both sentences is T and is F otherwise.**

8. **The value of the disjunction of two sentences is F if the value of both sentences is F and is T otherwise.**

9. **The value of the implication of two sentences is F only when the premise sentence is T and the consequent sentence is F; otherwise it is T.**

10. **The value of the equivalence of two sentences is T only when both sentences have the same truth assignment for all possible interpretations; otherwise it is F**

11. The value of $\forall x$ S is T if S is T for all assignments to X under I, and it is F otherwise.

12. The value of $\exists X$ S is T if there is an assignment to X in the interpretation under which S is T, otherwise it is F.

例1 设D={1，2}，求公式A= $(\forall x)(\exists y)P(x,y)$在D上的解释，并指出解释下公式A的真值。

解: 设对谓词P(x,y)在个体域D上的真值指派为：

$$P(1,1)=T,\ P(1,2)=F,\ P(2,1)=T,\ P(2,2)=F$$

在此解释下，因为x=1时有y=1使P(x,y)的真值为T， x=2时也有y=1使P(x,y)的真值为T，所以在此解释下公式A的真值为T。

还可以为公式A中的谓词指派另一组真值，设为：

$$P(1,1)=T,\ P(1,2)=T,\ P(2,1)=F,\ P(2,2)=F$$

对D中的所有x(即x=1与x=2)不存在一个y使得公式A的真值为T，所以在此解释下公式A的真值为F。

例2 设D={1，2}，求公式B= $(\forall x)(P(x) \rightarrow Q(f(x),b))$在D上的一个解释，并指出B在此解释下的真值。

解:设对b指派D中的一个元素为b=1，对函数f(x)指派到D的映射为f(1)=2, f(2)=1。对谓词指派的真值为：

$P(1)=F, P(2)=T, Q(1,1)=T, Q(2,1)=F$

当x=1时，有 $P(1)=F, Q(f(1),1)=Q(2,1)=F$

所以$P(1) \rightarrow Q(f(1),1)$的真值为T。

当x=2时，有$P(2)=T, Q(f(2),1)=Q(1,1)=T$

所以$P(2) \rightarrow Q(f(2),1)$的真值也为T。

即对个体域D中的所有x都有$P(x) \rightarrow Q(f(x),b)$的真值为T，所以公式B在此解释下的真值为T。

- Dummy（哑元）： a variable only take a place, likes(george, x) ： x is a dummy

  ——仅仅是占位符，改变名字不会改变含义。

- Quantification（量化） of variables ：

  Variables must be quantified in either of two ways:

  universally or existentially.

- Free variable ： not be quantified

- Closed（封闭） expression ： all variables be quantified

- Undecidable（不可判定） ：when the domain is infinite, we can not decide the truth value of $\forall X p(X)$, …

- **Several relationships between negation and universal and existential quantifiers:**

for predicates p and q and variable X and Y:

$$\neg\exists X p(X) \equiv \forall X \neg p(X);$$

$$\neg\forall X p(X) \equiv \exists X \neg p(X);$$

$$\exists X p(X) \equiv \exists Y p(Y);$$

$$\forall X p(X) \equiv \forall Y p(Y);$$

$$\forall X(p(X) \wedge q(X)) \equiv \forall X p(X) \wedge \forall Y q(Y);$$

$$\exists X(p(X) \vee q(X)) \equiv \exists X p(X) \vee \exists Y q(Y);$$

注意：∀对∨，∃对∧无分配等值式

基本等值式

第一组 命题逻辑中16组基本等值式的代换实例（P62）

例如，$\neg\neg\forall xF(x)\Leftrightarrow\forall xF(x)$（$\neg\neg P\Leftrightarrow P$的代换实例）

$\forall xF(x)\rightarrow\exists yG(y) \Leftrightarrow \neg\forall xF(x)\vee\exists yG(y)$（$P\rightarrow Q$）

第二组

(1) 消去量词等值式

设$D=\{a_1, a_2, \ldots, a_n\}$

① $\forall xA(x) \Leftrightarrow A(a_1)\wedge A(a_2)\wedge\ldots\wedge A(a_n)$

② $\exists xA(x) \Leftrightarrow A(a_1)\vee A(a_2)\vee\ldots\vee A(a_n)$

(2) 量词否定等值式

① $\neg\forall xA(x) \Leftrightarrow \exists x\neg A(x)$

② $\neg\exists xA(x) \Leftrightarrow \forall x\neg A(x)$

(3) 量词辖域收缩与扩张等值式.

$A(x)$ 是含 x 自由出现的公式，B 中不含 x 的自由出现

① $\forall x(A(x)\vee B) \Leftrightarrow \forall xA(x)\vee B$

② $\forall x(A(x)\wedge B) \Leftrightarrow \forall xA(x)\wedge B$

③ $\forall x(A(x)\to B) \Leftrightarrow \exists xA(x)\to B$

④ $\forall x(B\to A(x)) \Leftrightarrow B\to\forall xA(x)$

关于存在量词的：

① $\exists x(A(x)\vee B) \Leftrightarrow \exists xA(x)\vee B$

② $\exists x(A(x)\wedge B) \Leftrightarrow \exists xA(x)\wedge B$

③ $\exists x(A(x)\to B) \Leftrightarrow \forall xA(x)\to B$

④ $\exists x(B\to A(x)) \Leftrightarrow B\to\exists xA(x)$

- **Definition**

  first-order predicate calculus

  （一阶谓词演算）

  First-order predicate calculus allows quantified（量化） variables to refer to objects in the domain and not to predicates or functions.

- **For example:**

  ∀(likes) likes(george, kate)

- Representation examples

- If it doesn't rain on Monday,Tom will go to the mountains.

  $\neg weather(rain, monday) \rightarrow go(tom, mountains)$

- All basketball player are tall.

  $\forall X(basketballplayer(X) \rightarrow tall(X))$

- Nobody likes taxes.

  $\neg \exists X\ likes(X, taxes)$

思考?

（1）自然语言中有哪些连词?汉语中常用连词应如何表示?

（2）自然语言中有哪些量词?汉语中常用量词应如何表示?

（3）从思考题（1）（2）中得出了什么结论?

存在问题！

谓词表示越细，推理越慢、效率越低，但表示清楚。实际中是要折衷的。

练习：试用一阶谓词逻辑表示下列语句：

如果一个计算机系统能够完成这样一个任务：当该任务由人完成时，要求人具有智能，则该计算机系统具有智能。

思考：一阶逻辑的知识表示方法如何推理？

2.3 Using Inference Rules to Produce Predicate Calculus Expressions

- Definition : satisfy（满足）, model（模型）, valid（有效）, inconsistant （不一致）

  For a predicate calculus expression X and an interpretation I:

- If X has a value of T under I and a particular variable assignment, then I is said to satisfy X;
- If I satisfies X for all variable assignments,then I is a model of X;

- X is satisfiable if and only if there exist an interpretation and variable assignment that satisfy it; otherwise,it is unsatisfiable.
- A set of expressions is satisfiable if and only if there exist an interpretation and variable assignment that satisfy every element.

- If a set of expressions is not satisfiable ,it is said to be **inconsistent**.
- If X has a value T for all possible interpretation,X is said to be **valid**.
- For example:

$\exists X(p(X) \wedge \neg p(X))$ **inconsistent**

$\forall X(p(X) \vee \neg p(X))$ **valid**

Definition : logically follows（逻辑派生）

sound（可靠的）, and complete(完备的)

- **A predicate calculus expression X logically follows from a set of predicate calculus expressions S if every interpretation and variable assignment that satisfies S also satisfy X.**

- **An inference rule is sound** if every predicate calculus expression produced by the rule from a set S of predicate calculus expressions also logically follows from S.

- **An inference rule is complete** if,given a set S of predicate calculus expressions,the rule can infer every expression that logically follows from S.

- **Definition**

MODUS PONENS（取式假言推理），MODUS TOLLENS（拒式假言推理）, AND ELIMINATION（与消除）

- **If the sentences P and P→Q are known to be true,then modus ponens let us infer Q.**

- Under the inference rule **modus tollens** ,if $P \rightarrow Q$ is known to be true and Q is known to be false,we can infer that P is false, $\neg P$ is true.
- **and elimination** allows us infer the truth of either of the conjuncts from the truth of a conjunctive sentence.For instance, $P \wedge Q$ lets us conclude P and Q are true.

- **Definition**

AND INTRODUCTION（与引入）， UNIVERSAL INSTANTIATION（全称例化）

- **and introduction lets us infer the truth of a conjunction form the truth of its conjuncts.For instance, if P and Q are true,then P∧Q is true.**
-

universal instantiation states that if any universally quantified variable in a true sentence, say p(X), is replaced by an appropriate term from the domain, the result is a true sentence. Thus, if a is from the domain of X, $\forall X p(X)$ lets us infer p(a).

- **Example**

"All man are mortal and Socrates is man ;

therefore Socrates is mortal."

- "all man are mortal "

  $\forall X(man(X) \rightarrow mortal(X));$

- "Socrates is man "

  $man(Socrates);$

  $man(Socrates) \rightarrow mortal(Socrates);$

  $mortal(socrates);$

2.3.2 Unification（合一）

- Unification is an algorithm for determining the substitutions（代换） needed to make two predicate calculus expressions match.
- An essential aspect of this form is the requirement that all variables be universally quantified（全称量化）. So existentially quantified（存在量化） variables should be eliminated.
- ∃X parent(X, tom) ⟹ parent(bob, tom)
- ∀X∃Y mother(X, Y) ⟹ ∀X mother(X, f(X))

例：将下列公式G斯柯伦化（**skolemization**）。

$$(\forall x)(\exists y)(\exists z)((\sim P(x,y)\vee R(x,y,z))\wedge(Q(x,z)\vee R(x,y,z)))$$

解：G的Skolem标准型如下：

$$(\forall x)((\sim P(x,f(x))\vee R(x,f(x),g(x)))$$
$$\wedge(Q(x,g(x))\vee R(x,f(x),g(x))))$$

- **Instance**

  foo(X, a, goo(Y)) can generate

  foo(fred, a, goo(Z))

  by legal substitution: { fred / X, Z / Y }

- { X / Y } indicates that X is substituted for the variable Y in the original expression.

- { p(X) / X } is illegal

  X can not be replaced by p(X) : p(p(p(p(…X)))

  p(X) and X can not be unified :

  p(p(X)) p(X)

The composition of unification subsitutions

- If S and S’ are two substitution sets, then the composition of S and S’(written SS’) is obtained by applying S’ to the elements of S and adding the result to S.

- For example

  S1={X/Y,W/Z}, S2={V/X}, S3={a/V,f(b)/w}

  S2S3={a/X, a/V, f(b)/W}=s4

  S1S4={a/Y, f(b)/Z, a/X, a/V, f(b)/W}

- **定义1** ：代换是形如$\{t_1/x_1, t_2/x_2, \ldots, t_n/x_n\}$的有限集合。其中

（1）$t_1, t_2, \ldots, t_n$是项；$x_1, x_2, \ldots, x_n$是互不相同的变元；t_i/x_i表示用t_i代换x_i。

（2）不允许t_i与x_i相同，也不允许变元x_i循环地出现在另一个t_j中。

- 例如：$\{g(y)/x, f(x)/y\}$不是一个代换。

$\{g(a)/x, f(x)/y\}$是一个代换。

举例说明$\{a/x\}$、$\{y/x\}$、$\{f(a)/y\}$、$\{x/x, a/y\}$、$\{a/x, b/x\}$、$\{y/x, a/y\}$、$\{a/x, a/y\}$ 是代换吗？

代换可以作用于某个谓词公式上，也可以作用于某个项上。

例如：如果 $\theta=\{c/x, f(d)/y, t/z\}$，

$P=Q(x, y, z)$，

$u=g(x, y)$

则 $P\theta=Q(c, f(d), t)$，

$u\theta=g(c, f(d))$。

- **定义2** 设 $\theta=\{t_1/x_1, t_2/x_2, \ldots, t_n/x_n\}$

$$\lambda=\{u_1/y_1, u_2/y_2, \ldots, u_m/y_m\}$$

是两个代换，则此两个代换的复合也是一个代换，它是从$\{t_1\lambda\ /x_1, t_2\lambda\ /x_2, \ldots, t_n\lambda\ /x_n, u_1/y_1, u_2/y_2, \ldots, u_m/y_m\}$中删去如下两个元素：

$$t_i\ \lambda/x_i \qquad 当 t_i\ \lambda=x_i$$

$$u_i/y_i \qquad 当 y_i \in \{x_1, x_2, \ldots, x_n\}$$

后剩下的元素所构成的集合，记为$\theta\circ\lambda$ 。

- 例：设有代换θ和λ分别为：

$$\theta=\{f(y)/x, z/y\}$$

$$\lambda=\{a/x, b/y, y/z\}$$

求$\theta\circ\lambda$

解：$\theta\circ\lambda=\{f(y)\cdot\lambda/x,\ z\cdot\lambda/y,\ a/x,\ b/y,\ y/z\}$

$$=\{f(b)/x,\ y/y,\ a/x,\ b/y,\ y/z\}$$

先删除a/x和b/y, 再删除y/y,

得：

$$\theta\circ\lambda=\{f(b)/x,\ y/z\}$$

- 定义3：设有公式集F={F_1，F_2，...，F_N}，若存在一个代换λ使得

$$F_1\lambda = F_2\lambda = \ldots = F_N\lambda$$

则称λ为公式集F的一个合一（代换），且称F_1，F_2，... F_N 是可合一的。

- 判断公式集{P(a, x)，P(b, y)}、{P(a, x)，P(x, y)}、{P(y, c)，P(x, z)}可合一吗？

- 定义4：设σ是公式集F的一个合一，如果对任一个合一θ都存在一个代换λ，使得

$$\theta = \sigma \circ \lambda$$

则称σ是公式集F的最一般合一。

- 例1：E1=Q(a, y), E2=Q(z, f(b))是可合一的。

  θ={a/z, f(b)/y}是其最一般合一。

- 例2：E1=Q(y), E2=Q(f(z))是可合一的。

  θ={f(a)/y, a/z}不是其最一般合一。

  δ={f(z)/y} 是其最一般合一。

- 例3：E1=Q(y), E2=Q(z)是可合一的

  θ={y/z}和δ={z/y}都是其最一般合一。

- **Definition**

most general unifier（最一般合一，mgu）

- **If s is any unifier of expressions E, and g is the most general unifier of that set of expressions, then for s applied to E there exists another substitution s’ such that Es=Egs’ , where s = gs’ is the composition of g and s’**

The most general unifier of a set of expressions is **unique** except for alphabetic variations(字母变种); i.e., whether a variable is eventually called X or Y really does not make any difference to the generality of the resulting unifications.

We next present **pseudo-code** for a function, **unify**, that can compute the unifying substitutions (when this is possible) between two predicate calculus expressions.

Unify takes as arguments two expressions in the predicate calculus and returns either the most general set of unifying substitutions or the constant FAIL if no unification is possible.

It is defined as a **recursive function**: fist, it recursively attempts to unify the initial components(成分) of the expressions. If this succeeds, any substitutions returned by this unification are applied to the remainder of both expressions. These（作用后的结果） are then passed in a second recursive call to **unify**, which attempts to complete the unification.

The recursion stops when either argument is a symbol (a predicate, function name, constant, or variable) or the elements of the expression have all been matched.

List ——to simplify the manipulation of expressions

- examples——list syntax
  - p(a, b) ⇨ (p a b)
  - p(f(a), g(X, Y)) ⇨ (p (f a) (g X Y))
  - equal(eve, mother(cain))

    ⇩

  - (equal eve (mother cain))

- **List operations : first and rest**
  - first element of (p a b) : p
  - rest of (p a b) : (a b)
  - first element of (p (f a) (g X Y)) : p
  - rest of (p (f a) (g X Y)) : ((f a) (g X Y))
  - first element of (equal eve (mother cain)) :

    equal
  - rest of (equal eve (mother cain)) :

    (eve (mother cain))

```
Function unify(E1,E2)
 begin
   case
     both E1 and E2 are constants or the empty list:
   //recursion stops
          if E1=E2 then return {};
          else return fail;
     E1 is a variable:
          if E1 occurs in E2 then return fail
          else return {E2/E1};
     E2 is a variable:
          if E2 occurs in E1 then return fail
          else return {E1/E2};
     either E1 or E2 are empty then return fail
   //the list are of different size
```

```
otherwise:   //both E1 and E2 are list
    begin
        HE1:=first element of E1;
        HE2:=first element of E2;
        SUBS1:=unify(HE1,HE2);
        if SUBS1:=fail then return fail;
        TE1:=apply(SUBS1,rest of E1);
        TE2:=apply(SUBS1,rest of E2);
        SUBS2:=unify(TE1,TE2);
        if SUBS2:=fail then return fail;
            else return composition(SUBS1,SUBS2)
    end
  end
end
```

求最一般合一的算法

1）令$k=0$，$F_k=F$，$\sigma_k=\varepsilon$。这里，F是欲求其最一般合一的公式集，ε是空代换，它表示不做代换。

2）若F_k只含一个表达式，则算法停止，σ_k就是最一般合一。

3）找出F_k的差异集D_k。

4）若D_k中存在元素x_k和t_k，其中x_k是变元，t_k是项，且x_k不在t_k中出现，则置：

$$\sigma_{k+1}=\sigma_k \circ \{t_k/x_k\},$$

$$F_{k+1}=F_k\{t_k/x_k\},$$

$$k=k+1$$

然后转2）。

5）算法终止，F的最一般合一不存在。

- 例4 设有公式集 $F=\{P(a,x,f(g(y))),P(z,f(z),f(u))\}$ 求其最一般合一。
- 解：1）令 $\sigma_0=\varepsilon$ ， $F_0=F$ ，因 F_0 中有两个表达式，所以 σ_0 不是最一般合一。

  2）差异集 $D_0=\{a,z\}$

  3） $\sigma_1=\sigma_0\{a/z\}=\{a/z\}$，

  $F_1=\{P(a,x,f(g(y)),P(a,f(a),f(u))\}$

  4） $D_1=\{x,f(a)\}$

5） $\sigma_2 = \sigma_1\{f(a)/x\} = \{a/z, f(a)/x\}$

$F_2 = F_1\{f(a)/x\} = \{P(a, f(a), f(g(y))), P(a, f(a), f(u))\}$ 。

6） $D_2 = \{g(y), u\}$

7） $\sigma_3 = \sigma_2\{g(y)/u\} = \{a/z, f(a)/x, g(y)/u\}$

8） $F_3 = F_2\{g(y)/u\} = \{P(a, f(a), f(g(y)))\}$

- 因为F_3只含一个表达式，所以σ_3 就是最一般合一， 即最一般合一为：$\{a/z, f(a)/x, g(y)/u\}$

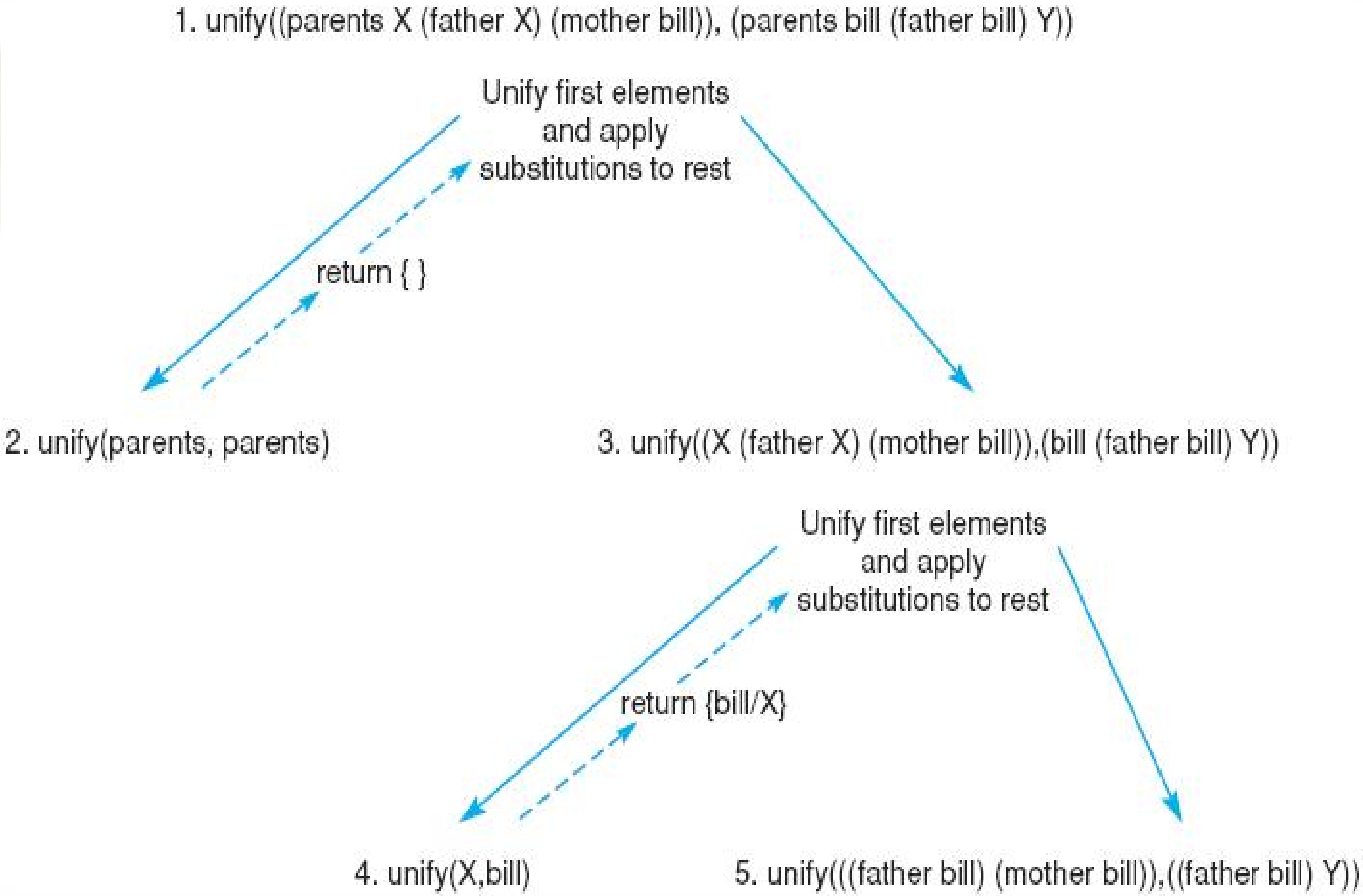
2.3.3 A Unification Example

- **Example:**

➢ **Unify ((parents X (father X) (mother bill)),**

(parents bill (father bill) Y))

Partial trace of the unification



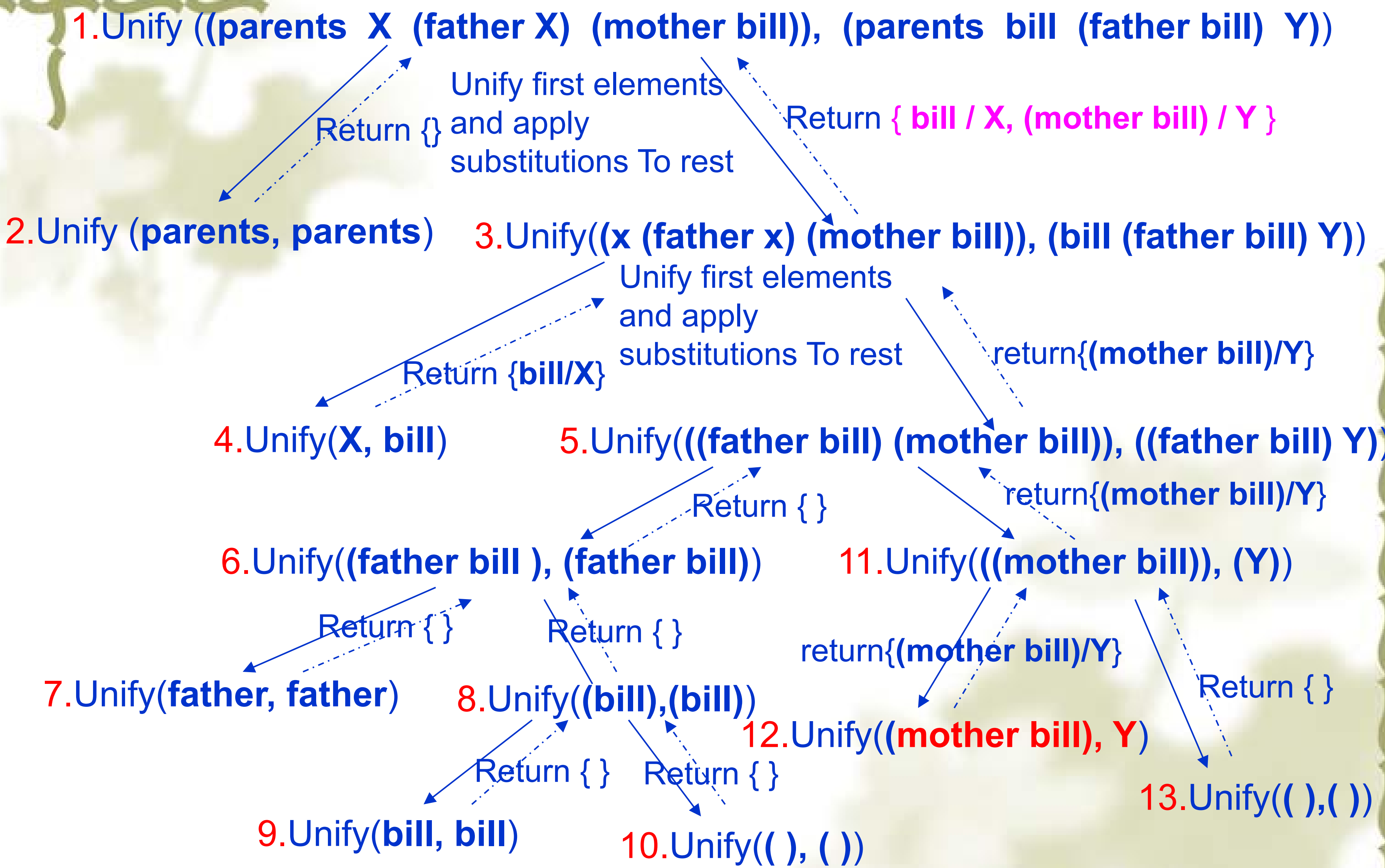


Figure 2.6 full trace of the unification

2.4 Application: A Logic-Based Financial Advisor

1. **Individual with an inadequate savings account should always make increasing the amount saved their first priority, regardless of their income.**
2. **Individual with an adequate savings account and an adequate income should consider a riskier but potentially more profitable investment in the stock market.**
3. **Individual with a lower income who already have an adequate savings account may want to consider splitting their surplus income between savings and stocks, to increase the cushion in savings while attempting to increase their income through stocks.**

1. **saving_account(inadequate)→investment(savings)**
2. **saving_account(adequate) ∧ income(adequate)→investment(stocks)**
3. **saving_account(adequate) ∧ income(inadequate)→investment(combination)**

4. $\forall X amount\_saved(X) \wedge \exists Y(dependents(Y) \wedge greater(X, minsavings(Y))) \rightarrow saving\_account(adequate)$

5. $\forall X amount\_saved(X) \wedge \exists Y(dependents(Y) \wedge \neg greater(X, minsavings(Y))) \rightarrow saving\_account(inadequate)$

6. $\forall X$ earnings(X, steady) $\wedge \exists Y$(dependents(Y) $\wedge$greater(X, minincome(Y))) $\rightarrow$income(adequate)

7. $\forall X$ earnings(X, steady) $\wedge \exists Y$(dependents(Y) $\wedge \neg$greater(X, minincome(Y))) $\rightarrow$income(inadequate)

8. $\forall X$ earnings(X, unsteady) $\rightarrow$ income(inadequate)

9. amount_saved(22000)

10. earnings(25000, steady)

11. dependents(3)

Where: minsavings(X) ≡ 5000*X

minincome(X) ≡ 15000 + (4000*X)

- **Step1**

earnings(25000, steady)∧ dependents(3) unifies with :

earnings(X, steady) ∧(dependents(Y)

under the substitution {25000/X, 3/Y},

this yeilds a new implication :

earnings(25000, steady) ∧(dependents(3)

∧¬greater(25000, minincome(3)))

→income(inadequate)

12. income(inadequate)

- **step 2 (similarly)**

  amount_saved(22000)∧ dependents(3)

  unifies with the first two element of the premise of assertion 4

  under the substitution {22000/X, 3/Y}

  yielding the implication :

  amount_saved(22000) ∧(dependents(3)

  ∧greater(22000,minsavings(3)))

  →saving_account(adequate)

13. saving_account(adequate)

- **Step 3**

From 3, 12, and 13, we can get the following conclusion:

14. Investment (combination)

1. Write predicate calculus expressions for the following English sentences:

i. Emma is a German shepherd, all German shepherds are dogs, therefore Emma is a dog.

ii. People who like cats don’t like dogs.

iii. Some English sentences cannot be represented with predicate calculus.

iv. If we have cheese, I will make a sandwich.

2. Jane Doe has three dependents, a steady income of $45,000 and $21,000 in her savings account. Add the appropriate predicates describing her situation to the general investment advisor of the example in Section 2.4 and perform the unifications and inferences needed to determine her suggested investment.

1. $savings\_account(inadequate) \rightarrow investment(savings).$
2. $savings\_account(adequate) \wedge income(adequate) \rightarrow investment(stocks).$
3. $savings\_account(adequate) \wedge income(inadequate)$
   $\rightarrow investment(combination).$
4. $\forall\ amount\_saved(X) \wedge \exists Y\ (dependents(Y) \wedge$
   $greater(X, minsavings(Y))) \rightarrow savings\_account(adequate).$
5. $\forall X\ amount\_saved(X) \wedge \exists Y\ (dependents(Y) \wedge$
   $\neg\ greater(X, minsavings(Y))) \rightarrow savings\_account(inadequate).$
6. $\forall X\ earnings(X, steady) \wedge \exists Y\ (dependents\ (Y) \wedge$
   $greater(X, minincome(Y))) \rightarrow income(adequate).$
7. $\forall X\ earnings(X, steady) \wedge \exists Y\ (dependents(Y) \wedge$
   $\neg\ greater(X, minincome(Y))) \rightarrow income(inadequate).$
8. $\forall X\ earnings(X, unsteady) \rightarrow income(inadequate).$
9. $amount\_saved(22000).$
10. $earnings(25000, steady).$
11. $dependents(3).$

Solution:

Jane Doe can be described by:

amount_saved(21000)

earnings(45000, steady)

dependents(4)

minsavings(4) ≡ 5000 * 4 = 20000

minincome(4) ≡ 15000 + (4000 * 4) = 31000

Unify the conjunction of earnings(45000, steady) and dependents(4) with first two components of the premise of rule 6, under the substitution {45000/X, 4/Y}, yielding the new implication:

第二章 作业

3

6

7

10