

Вивчення моделей подання знань в курсі інформатики вищого педагогічного навчального закладу

Подання знань як методологія моделювання і формалізації концептуальних знань, орієнтована на комп'ютерне опрацювання, є однією з основних і найважливіших тем, які відносяться до інженерії знань. Це обумовлено інтенсивним розвитком штучного інтелекту після 70-х років минулого століття, коли на відміну від досліджень, орієнтованих на одержання простих і універсальних висновків, які обмежували галузі застосування систем штучного інтелекту, появились підходи (що базувалися на використанні знань), які відкрили шлях до створення прикладних систем в різноманітних сферах людської діяльності. Саме тому історію досліджень з штучного інтелекту, за виключенням початкових її етапів, можна назвати історією досліджень методів подання знань [1].

Використання знань - це технологія одержання (вивід) розв'язку у відповідності з певною моделлю подання знань. Складність і різноманітність структур знань вимагають різних способів їх подання.

Особливість систем подання знань полягає в тому, що вони моделюють діяльність, яка здійснюється часто в неформальному виді. Якщо методи розв'язування обчислювальних задач ґрунтуються на строгих алгоритмах, обґрунтованість яких базується на поняттях збіжності та стійкості, то моделі подання знань мають справу з інформацією, яку одержують від експертів, яка часто носить якісний і до того ж суперечний характер. І тим не менш в силу специфіки функціонування комп'ютера подібна інформація повинна бути приведена до однозначного формалізованого виду.

В наш час моделі подання знань є предметом досліджень і розробок як в середовищі вузьких спеціалістів (в основному програмістів і математиків), так і спеціалістів різноманітних предметних галузей, які намагаються використовувати інтелектуальні системи в своїй діяльності.

Можна визначити кілька напрямів досліджень щодо подання знань в інтелектуальних системах. Так, дослідники з штучного інтелекту когнітивного напрямку (що займалися проблемами подання знань, яке орієнтоване в основному на можливості розв'язування задач) спиралися в своїх дослідженнях на принципи організації людської пам'яті і моделювання виводу на їх основі. Тут типовими моделями для подання знань є моделі, що базуються на використанні продукцій, семантичних мереж, фреймів.

Дослідники логічного напрямку з свого боку просунулися по шляху вивчення теоретичних методів подання знань. В результаті була створена модель подання знань на основі логіки предикатів першого порядку. При цьому серйозна увага приділялася поданню знань і генеруванню виводів на основі строгої теоретичної системи, що безумовно, призвело до розвитку систем, що будуються на знаннях. В основі методів використання знань лежить головним чином математична формалізація і логічна повнота. Навпаки, когнітивний підхід базується на розумінні процесу усвідомлення чого-небудь людиною, тому поданню знань в цьому випадку притаманна швидше виразність, ніж математична елегантність і строгість.

Слід зазначити, що останнім чином посилюються інтеграційні зв'язки між згаданими напрямками. Зокрема, було запропоновано декілька корисних концепцій, методи їх моделювання і практичного втілення, які базуються на впровадженні ідей логічного підходу в когнітивне подання знань, а також розширення рамок існуючої логіки (для надання їй більшої гнучкості) з врахуванням практичного застосування логічного подання знань. Тобто обидва ці підходи розвиваються в тісному взаємозв'язку.

Сьогодні ідеї штучного інтелекту диктують напрями науково-технічного прогресу. Тому вивчення питань інтелектуалізації комп'ютерів, основ штучного інтелекту повинно займати вагоме місце в курсі інформатики педагогічного ВНЗ і загальноосвітньої школи. Зазначимо, що початки штучного інтелекту включені в посібники з інформатики для вищих педагогічних навчальних закладів [2, 3], проте на сьогодні цей розділ потребує набагато ширшого розгляду.

Важливість вивчення моделей подання знань обумовлюється також тією роллю, яка належить моделюванню: на ідеї моделювання по суті базується будь-який метод наукового пізнання - як теоретичний (при якому використовуються різного роду знакові, абстрактні моделі, зокрема інформаційні та математичні), так і експериментальний (який використовує предметні моделі).

Вивчення моделювання, зокрема моделей подання знань та їх використання в інтелектуальних системах значною мірою сприятиме підвищенню рівня фундаментальної підготовки студентів, формуванню вмінь працювати із знаннями, вибирати адекватні моделі для їх подання та досліджувати ці моделі за допомогою комп'ютера. Очевидно, що зазначені вміння є важливими компонентами інформаційної та професійної культури майбутніх вчителів математики та інформатики.

В інтелектуальних системах, як зазначалося, для подання знань використовуються різноманітні засоби, зокрема: логіка предикатів, продукційні правила, семантичні мережі, фрейми,

об'єктно-орієнтоване програмування. Моделі подання знань, що базуються на них, відповідно називають: логічна, продукційна, фреймова, об'єктно-орієнтована моделі, модель семантичної мережі. Кожний з формалізмів подання знань має свої переваги й недоліки. Звичайно, вибір оптимального способу подання знань в кожному випадку в значній мірі залежить від характеру і складності тих задач, які доводиться розв'язувати. Проте незалежно від того, використовується чи ні логіка предикатів для подання знань, вона складає теоретичний фундамент в систематиці опису, а по своїй суті знаходиться в основі всіх способів подання знань. Використання логіки предикатів для автоматичного доведення теорем, подання знань, одержання виводів і операцій зі знаннями, аналізу міркувань розглядалося в [4]. Тут розглянемо інші моделі подання знань. Крім того, покажемо, яким чином реалізувати ці моделі засобами логіки предикатів першого порядку. Такий підхід у виченні моделей подання знань дозволяє не тільки глибше вивчити особливості кожної моделі подання знань та зв'язки між ними, а також доводити задачі до практичної реалізації, виводити відповіді на запити щодо розроблених баз знань з використанням методу резолюції, легко і природно переходити до реалізації задач на комп'ютері засобами логічного програмування.

Продукційні правила (правила продукції) є операторами вигляду “ситуація-дія” і в загальному випадку записуються так:

ЯКЩО P_1, P_2, \dots, P_n ТО B .

Тут P_1, P_2, \dots, P_n – умови застосовності продукції (посилки), B – висновок (дія). Записане правило означає, що “якщо всі умови від P_1 до P_n істинні, то B також істинне” або “якщо всі умови від P_1 до P_n виконуються, то слід виконати дію B ”.

Прикладом продукції є правило

ЯКЩО

(1) x є батьком y

(2) y є батьком z

ТО x є дідусям z ,

яке містить дві умови ($n=2$).

У випадку $n=0$ продукція описує знання, яке складається тільки з висновку, тобто факт. Прикладами такого знання є факти “Сергій є батьком Петра”, “Квадрат є ромбом”.

Знання про предметну галузь, подані продукційною моделлю, часто можна легко подати також за допомогою логіки предикатів.

Нехай, наприклад, вартість проїзду в автобусі визначається такими правилами:

ЯКЩО кількість зупинок менша 3

ТО вартість проїзду 20 коп.

ЯКЩО кількість зупинок не менша 3, але менша 5

ТО вартість проїзду 30 коп.

ЯКЩО кількість зупинок не менша 5
ТО вартість проїзду 50 коп.

Для побудови логічної моделі цієї галузі знань спочатку виділимо значущі сутності (кількість зупинок і вартість проїзду), введемо потрібні предикати і відповідні позначення для них:

МЕНШЕ (x, y) ; x менше y – $Q(x, y)$;

ВАРТІСТЬ (x, y) ; y є вартість за проїзд x зупинок – $P(x, y)$;
якщо y збігається з встановленою вартістю, то значення предиката істинне, інакше – хибне.

Подаючи продукції відповідними формулами логіки предикатів (і опускаючи квантори), матимемо

$Q(x, 3) \rightarrow P(x, 20)$;

$\neg Q(x, 3) \wedge Q(x, 5) \rightarrow P(x, 30)$;

$\neg Q(x, 5) \rightarrow P(x, 50)$.

Щоб теорія галузі знань була повною, треба до трьох її аксіом ще додати формули (факти), які задають в області визначення істинні значення предиката $Q(x, y)$: $Q(1, 2)$, $Q(1, 3)$, $Q(4, 5)$, $\neg Q(3, 3)$ тощо. Тоді можна буде формально виводити відповіді на різноманітні запити щодо цієї галузі знань.

Про можливість подання продукційних правил формулами логіки предикатів свідчать також приклади в [4].

В моделлях, які базуються на апараті семантичних мереж, в явній формі виділяються деякі інформаційні одиниці (поняття, сутності), що відповідають предметам, явищам, фактам чи процесам реального світу, між якими вказуються типи зв'язків (відношення). При цьому інформаційні одиниці подаються пойменованими вершинами (вузлами), а відношення – направленими пойменованими дугами. Дуга, під'єднана до єдиного вузла, задає його властивості. Модель у даному випадку подається у вигляді мережі (семантичної мережі) – орієнтованого графа, який складений з пойменованих вершин і дуг. Семантичні мережі здатні відображати структуру знань з усією складністю їх взаємозв'язків, пов'язувати в єдине ціле об'єкти і їх властивості.

На рис.1 наведено частину семантичної мережі, яка описує поняття “чотирикутник”.

Виводи на семантичних мережах визначаються через відношення між множинами дуг, які мають спільні вузли.

Важливою особливістю семантичних мереж є ієрархія. Вузол, який представляє певні знання, з'єднано дугами ”є” з вузлами вищого та нижнього рівнів (відношення виду і роду, відношення належності елемента класу). Спосіб подання знань семантичною мережею дозволяє легко робити виводи завдяки ієрархії успадкування. Екземпляр нижнього рівня містить атрибути (властивості), які має екземпляр поняття верхнього рівня (прототип). Саме ця властивість

називається успадкуванням атрибутів між рівнями ієрархії "ε". У семантичних мережах можна відобразити також відношення "ціле-частина".

Якщо розглядати семантичну мережу як опис відношень, які мають місце між поняттями, то її можна реалізувати за допомогою логіки предикатів. Покажемо, як це можна здійснити для мережі, поданої на рис.1.

Введемо предикати і позначення для них:

$P(x, y)$ – "x є y";

$Q(x)$ – "x має чотири сторони";

$R(x, y)$ – "x має паралельні протилежні y";

$S(x, y)$ – "x має всі прямі y";

$T(x, y)$ – "x має всі рівні y".

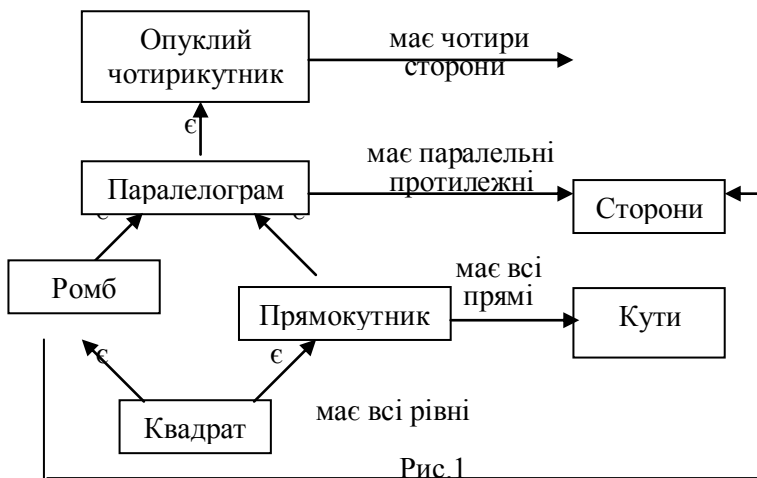


Рис.1

Структура мережі дає можливість записати такі істинні

формули:

P (квадрат, прямокутник);

P (квадрат, ромб);

P (ромб, паралелограм);

P (прямокутник, паралелограм);

P (паралелограм, опуклий_чотирикутник);

S (прямокутник, кути);

T (ромб, сторони);

R (паралелограм, сторони);

Q (опуклий_чотирикутник);

$\forall x \forall y \forall z (P(x, z) \wedge S(z, y) \rightarrow S(x, y))$;

$\forall x_1 \forall y_1 \forall z_1 (P(x_1, z_1) \wedge T(z_1, y_1) \rightarrow T(x_1, y_1))$;

- $\forall x_2 \forall y_2 \forall z_2 (P(x_2, z_2) \wedge R(z_2, y_2) \rightarrow R(x_2, y_2))$;
- $\forall x_3 \forall z_3 (P(x_3, z_3) \wedge Q(z_3) \rightarrow Q(x_3))$.

Зазначимо, що останні чотири формули забезпечують успадкування об'єктами нижчого рівня властивостей об'єктів вищого рівня.

Множина із 13 формул описує теорію предметної галузі, поданої на рис.1.

У відповідності з правилами побудови запису формул логіки предикатів, а також для спрощення записів, введемо позначення для предметних констант (значущих сутностей):

- опуклий_чотирикутник – a ;
- паралелограм – b ;
- квадрат – c ;
- прямокутник – b_1 ;
- ромб – b_2 ;
- сторони – d ;
- кути – e ;

Скориставшись рівносильностями

- $\forall x \forall y \forall z (P(x, z) \wedge S(z, y) \rightarrow S(x, y)) \equiv$
- $\equiv \forall x \forall y \forall z (\neg P(x, z) \vee \neg S(z, y) \vee S(x, y))$;
- $\forall x_1 \forall y_1 \forall z_1 (P(x_1, z_1) \wedge T(z_1, y_1) \rightarrow T(x_1, y_1)) \equiv$
- $\equiv \forall x_1 \forall y_1 \forall z_1 (\neg P(x_1, z_1) \vee \neg T(z_1, y_1) \vee T(x_1, y_1))$;
- $\forall x_2 \forall y_2 \forall z_2 (P(x_2, z_2) \wedge R(z_2, y_2) \rightarrow R(x_2, y_2)) \equiv$
- $\equiv \forall x_2 \forall y_2 \forall z_2 (\neg P(x_2, z_2) \vee \neg R(z_2, y_2) \vee R(x_2, y_2))$;
- $\forall x_3 \forall y_3 \forall z_3 (P(x_3, z_3) \wedge Q(z_3) \rightarrow Q(x_3)) \equiv$
- $\equiv \forall x_3 \forall y_3 \forall z_3 (\neg P(x_3, z_3) \vee \neg Q(z_3) \vee Q(x_3))$

і опустивши для спрощення записів квантори загальності, одержуємо теорію предметної галузі у вигляді множини диз'юнктив.

- | | |
|----------------|---|
| 1. $P(c, b_1)$ | 8. $R(b, d)$ |
| 2. $P(c, b_2)$ | 9. $Q(a)$ |
| 3. $P(b_2, b)$ | 10. $\neg P(x, z) \vee \neg S(z, y) \vee S(x, y)$ |
| 4. $P(b_1, b)$ | 11. $\neg P(x_1, z_1) \vee \neg T(z_1, y_1) \vee T(x_1, y_1)$ |
| 5. $P(b, a)$ | 12. $\neg P(x_2, z_2) \vee \neg R(z_2, y_2) \vee R(x_2, y_2)$ |
| 6. $S(b_1, e)$ | 13. $\neg P(x_3, z_3) \vee \neg Q(z_3) \vee Q(x_3)$ |
| 7. $T(b_2, d)$ | |

Така множина даватиме можливість виводити методом резолюції відповіді на запити щодо даної галузі знань. Наведемо приклади запитів і одержання відповідей на них.

Нехай, наприклад, задано запит “Чи існує чотирикутник, який має всі прямі кути, і якщо так, то який?” цей запит можна записати у вигляді формули $\exists u S(u, e)$

Щоб отримати відповідь, запишемо заперечення до запиту, додамо його до множини диз'юнктив, які представляють теорію галузі знань, і спробуємо довести суперечність одержаної множини.

$$14. \neg S(u, e) \vee Answer(u)$$

Застосовуючи резолюцію до множини диз'юнктив, дістанемо

$$15. \square \vee Answer(b_1) \quad (14,6); \theta = \{u := b_1\}$$

Інтерпретуючи результат, можемо сформулювати відповідь. Існує чотирикутник, який має всі прямі кути, і таким чотирикутником є, зокрема, прямокутник.

Можна вивести ще один розв'язок.

$$15'. \neg P(u, z) \vee S(z, e) \vee Answer(u) \quad (14,7); \theta = \{u := c, z := b_1\};$$

$$16'. \neg S(b_1, e) \vee Answer(c) \quad (15',1); \theta = \{u := c, z := b_1\};$$

$$17'. \square \vee Answer(c) \quad (16',6).$$

Одержаний результат вказує, що зазначену властивість має також квадрат.

Розглянемо ще запит: "Які чотирикутники мають паралельні протилежні сторони?" Це відповідає такій формулі $\exists u R(u, d)$.

Як і раніше до множини диз'юнктив, які описують теорію цієї галузі знань, допишемо заперечення запиту з предикатом *Answer*.

$$14. \neg R(u, d) \vee Answer(u)$$

Застосовуючи метод резолюції, дістаємо

$$15. \square \vee Answer(b) \quad (14,10; \theta = \{u := b\}).$$

Таким чином, виведена відповідь: протилежні паралельні сторони має паралелограм.

Можна вивести й інші розв'язки.

$$15'. P(u, z_2) \vee \neg R(z_2, d) \vee Answer(u) \quad (14,12; \theta = \{x_2 := u, y_2 := d\})$$

$$16'. \neg R(b_1, d) \vee Answer(c) \quad (15',1; \theta = \{u := c, z_2 := b_1\})$$

$$17'. \neg P(b_1, z_2) \therefore \vee \neg R(z_2, d) \vee Answer(c) \quad (16',12; \theta = \{x_2 := b_1, y_2 := d\})$$

$$18'. \neg R(b, d) \vee Answer(c) \quad (17',4; \theta = z_2 := b\})$$

$$19'. \square \vee Answer(c) \quad (18',8).$$

Відповідь: квадрат.

Теорія предметної галузі дозволяє вивести й інші альтернативні розв'язки.

$$16''. \neg R(b, d) \vee Answer(b_2) \quad (15',3; \theta = \{u := b_2, z_2 := b\})$$

$$17''. \square \vee Answer(b_2) \quad (16',8).$$

Відповідь: ромб.

$$16'''. \neg R(b, d) \vee Answer(b_1) \quad (15',4; \theta = \{u := b_1, z_2 := b\})$$

$$17'''. \square \vee Answer(b_1) \quad (16''',8).$$

Відповідь: прямокутник.

Отже, з бази знань про предметну галузь методом логічного виводу одержано, що протилежні сторони паралельні у паралелограмі, квадрата, ромба і прямокутника.

Запит “Які об’єкти (чотирикутники) перебувають у відношенні S ?” матиме вигляд $\exists u \exists v S(u, v)$?

Застосувавши метод резолюції, дістанемо:

1) $u = b_1, v = e$;

2) $u = c, v = e$.

А це означатиме, що кути всі прямі у прямокутника і квадрата.

На рис.2 зображено мережу, яка відповідає тексту “Дідусь сів у човен, посадив у човен козу, переплив на другий берег і взяв кошик з капустою”.

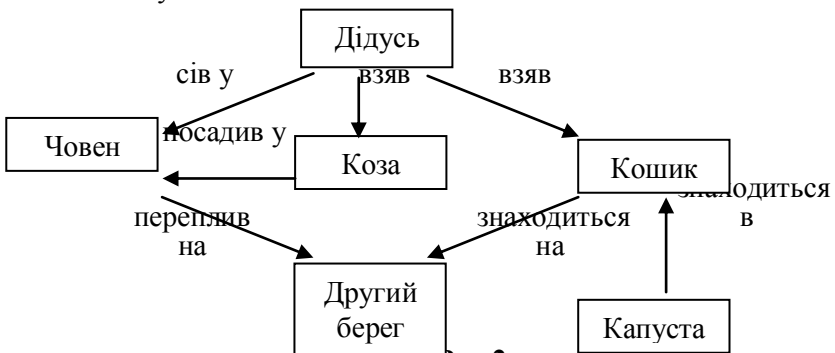


Рис.2

Знання, про які йдеться у тексті і які подані моделлю семантичної мережі, опишемо засобами логіки предикатів.

Спочатку введемо потрібні предикати, позначення для них та предметних констант:

$P(x, y)$ – “ x сів_у y ”;

$P_1(x, y)$ – “ x взяв y ”;

$Q(x, y)$ – “ x дідусь_посадив_у y ”;

$R(x, y)$ – “ x переплив_на y ”;

$R_1(x, y)$ – “ x знаходиться_на y ”;

$R_2(x, y)$ – “ x знаходиться_в y ”;

дідусь – d ;

човен – b ;

другий берег – c ;

коза – e_1 ;

кошик – e_2 ;

капуста – e_3 .

З наведеного тексту випливає, що істинними є такі конкретні реалізації введених предикатів:

$P(d, b)$ - “дідусь сів у човен”;
 $P_1(d, e_1)$ - “дідусь взяв козу”;
 $P_1(d, e_2)$ - “дідусь взяв кошик”;
 $Q(e_1, b)$ - “козу дідусь посадив у човен”;
 $R_2(e_3, e_2)$ - “капуста знаходиться в кошику”;
 $R(b, c)$ - “човен переплив на другий берег”;
 $R_1(e_2, c)$ - “кошик знаходиться на другому березі”.

Якщо обмежитися тільки цими фактами, то теорія предметної галузі не буде повною. Її слід доповнити такими очевидними правилами:

“Якщо x сів у y , а y переплив на z , то x переплив на z ”;

“Якщо x_1 знаходиться на y_1 , а z_1 знаходиться в x_1 , то z_1 знаходиться на y_1 ”,
яким відповідають формули:

$$\forall x \forall y \forall z (P(x, y) \wedge R(y, z) \rightarrow R(x, z)), \quad (1)$$

$$\forall x_1 \forall y_1 \forall z_1 (R_1(x_1, y_1) \wedge R_2(z_1, y_1) \rightarrow R_1(z_1, y_1)). \quad (2)$$

Отже, предметну галузь, описану текстом, можна подати такою базою знань:

- | | |
|----------------------|--|
| 1. $P(d, b)$; | 6. $R(b, c)$; |
| 2. $P_1(d, e_1)$; | 7. $\neg P(x, y) \vee \neg R(y, z) \vee R(x, z)$; |
| 3. $P_1(d, e_2)$; | 8. $R_1(e_2, c)$; |
| 4. $Q(e_1, b)$; | 9. $\neg R_1(x_1, y_1) \vee \neg R_2(z_1, x_1) \vee R_1(z_1, y_1)$. |
| 5. $R_2(e_3, e_2)$; | |

В наведеному списку формули 7) і 9) є диз'юнктивними нормальними формами формул (1) і (2) відповідно (квантори загальності за домовленістю опущено).

Проілюструємо, яким чином з наведеної бази знань можна виводити логічні висновки.

Сформулюємо, наприклад, такий запит: “Кого посадив дідусь у човен?”, якому відповідає формула $\exists u Q(u, b)$?

Для одержання відповіді до множини диз'юнктивів, які описують теорію предметної галузі, додамо заперечення запиту, тобто формулу $\forall u \neg Q(u, b)$.

Отже, десята у списку формула матиме вигляд

$$10. \neg Q(u, b) \vee Answer(u).$$

Для виведення відповіді застосуємо метод резолюції. Застосувавши резолюцію, одержимо

$$11. \square \vee Answer(e_1) \quad (10, 4); \quad \theta = \{u := e_1\}.$$

Отримана відповідь означає, що “у човен дідусь посадив козу”.

Тепер задамо запит: “Що знаходиться на другому березі?”
Цьому запиту відповідає формула

$$\exists u R_1(u, c)?$$

Для виведення відповіді до множини формул бази знань додаємо заперечення запиту з предикатом *Answer*, тобто формулу

$$10. \neg R_1(u, c) \vee \text{Answer}(u).$$

Резолюційний процес продовжуємо до одержання одиночного диз’юнкта *Answer*:

$$11. \square \vee \text{Answer}(e_2) \quad (10, 8); \quad \theta = \{u := e_2\}.$$

Отже, одержана така відповідь: “На другому березі знаходиться кошик”.

Можна вивести й інший розв’язок

$$11'. \neg R_1(x_1, c) \vee \neg R_2(u, x_1) \vee \text{Answer}(u)$$

$$(10, 9); \quad \theta = \{z_1 := u, y_1 := c\};$$

$$12'. \neg R_2(u, e_2) \vee \text{Answer}(u) \quad (11', 8) \quad \theta = \{x_1 := e_2\};$$

$$13'. \text{Answer}(e_3) \quad (12', 5); \quad \theta = \{u := e_3\}.$$

Таким чином, на запит одержана ще одна відповідь: “На другому березі знаходиться капуста”.

Інших відповідей на цей запит вивести не можна. Отже, на запит можна дати таку загальну відповідь: “На другому березі знаходяться кошик і капуста”.

Розглянемо ще один запит: “Чи дідусь переплив на другий берег?” Відповідь буде позитивна, якщо можна з бази знань вивести формулу $R(d, c)$.

Для доведення до множини диз’юнктів додамо заперечення запиту, тобто формулу

$$10. \neg R(d, c).$$

Доведення, як і раніше, проводимо методом резолюції.

$$11. \neg P(d, y) \vee \neg R(y, c) \quad (10, 7); \quad \theta = \{x := d, z := c\};$$

$$12. \neg R(b, c) \quad (11, 1); \quad \theta = \{y := -b\};$$

$$13. \square \quad (12, 6).$$

Оскільки виведено порожній диз’юнкт, то відповідь позитивна: “Так, дідусь переплив на другий берег”.

Фреймові моделі базуються на теорії фреймів, яка розроблена М.Мінським [4, 5].

Фрейм (від англійського слова *frame* – рамка, каркас) – це спеціальні інформаційні структури для подання знань про об’єкти, які можна описати деякою сукупністю понять і сутностей. Фрейм має певну внутрішню структуру, яка складається з елементів, які називаються слотами (від англ. *slot* – щілина). Кожний слот, в свою чергу, подається певною структурою даних, процедурою, або може бути зв’язаний з іншим фреймом.

Фрейми можуть бути зв'язані між собою деякими відношеннями і утворювати семантичну мережу. Основна відмінність від звичайних семантичних мереж полягає в тому, що фрейми можуть містити процедури, які управляють перетворенням інформації як у самому фреймі, так і в інших вузлах мережі.

На рис.3 подано приклад схеми класифікаційної ієрархічної структури.

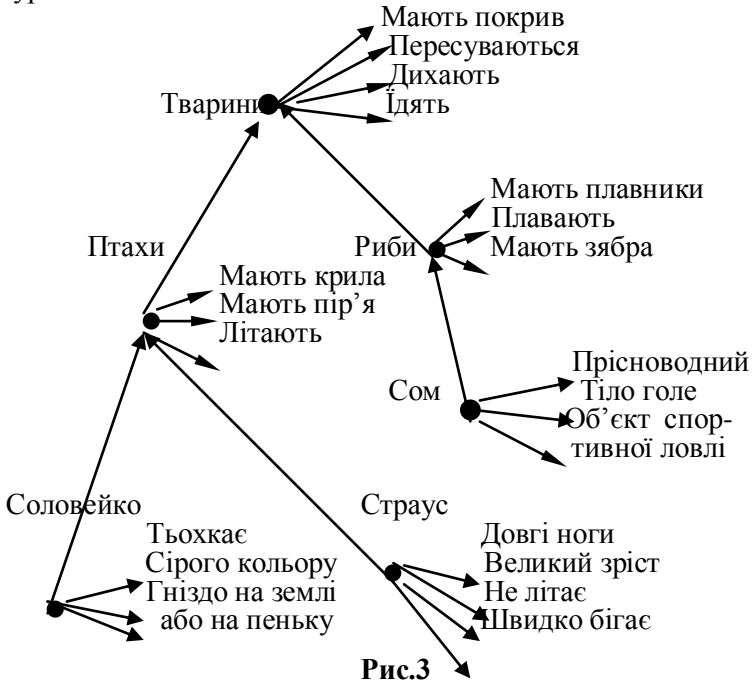


Рис.3

Ця структура базується на відношеннях “абстрактне-конкретне”. (Крім такого типу структур існують й інші, які базуються на відношеннях “ціле-частина”). Відношення “абстрактне-конкретне” характерні тим, що на верхніх рівнях, як показано на рис.3, розміщені абстрактні об’єкти (концепти), а на нижніх конкретні об’єкти, причому об’єкти нижніх рівнів успадковують атрибути (властивості) (якщо відповідні описи в них відсутні) об’єктів верхніх рівнів. Ці відношення ще називають відношеннями типу “є” (IS-A) або “різновид” (KIND-OF) (сом є риба, сом – різновид риби)). Наприклад, атрибут “тварина дихає” є спільним і для риб, і для сома, який знаходиться на найнижчому рівні.

Якби людська пам’ять мала подібну структуру, то можна було б систематизувати і запам’ятовувати подібні поняття, уникати зайвих складнощів стосовно інформації про атрибути і додавати нові поняття чи знання у відповідні позиції в цій ієрархії

(навчання). Тоді значно простіше було б виявляти суперечності в знаннях і управляти послідовністю, і якщо можна було б зрозуміти, що це риба навіть без використання конкретних знань (наприклад, що це лин), то значно підвищилася гнучкість системи, наприклад, появилася б можливість одержання висновків на основі знань про риб і тварин.

Відношення “ціле-частина” стосується структурованих об’єктів і показує, що об’єкт нижнього рівня є частиною (PART-OF) об’єкта верхнього рівня. У відношеннях “ціле-частина” атрибути не успадковуються. Наприклад, стіна є структурним елементом аудиторії, проте вона не є аудиторією, і тому об’єкт “стіна” не успадковує атрибутів “аудиторія”, який є об’єктом верхнього рівня.

У переважній більшості фреймових систем застосовуються відношення “абстрактне-конкретне”. Якщо значення слоту в одному з фреймів такої системи не задається, то воно переноситься (успадковується) з фрейма вищого рівня. Успадкування фреймами значення слотів здійснюється в тому випадку, якщо у фреймі є слот Різновид, в якому міститься ім’я фрейма вищого рівня.

Наведемо приклад фреймової моделі, яка описує поняття “науковий семінар”.

Фрейм: НАУКОВИЙ_СЕМІНАР

Різновид: ЗАХІД

Час: Вівторок_перед_третьою_середою_14.00

Місце: Кафедра_інформатики_НПУ_імені_Драгоманова

Фрейм: НАУКОВИЙ_СЕМІНАР_80

Різновид: НАУКОВИЙ_СЕМІНАР

Дата: 14.11.2000

Доповідачі: [Красюк Ю., Лотюк Ю.].

Тут згадують три фрейми – ЗАХІД, НАУКОВИЙ_СЕМІНАР, НАУКОВИЙ_СЕМІНАР_80. Фрейм ЗАХІД – найзагальніший (найвищого рівня), НАУКОВИЙ_СЕМІНАР – конкретніший, який описує вид ЗАХОДУ, а фрейм НАУКОВИЙ_СЕМІНАР_80 – найконкретніший, він описує конкретний НАУКОВИЙ_СЕМІНАР.

Оскільки фрейм НАУКОВИЙ_СЕМІНАР_80 має слот Різновид, в якому є посилання на фрейм НАУКОВИЙ_СЕМІНАР, то значення слотів Час і Місце фрейма НАУКОВИЙ_СЕМІНАР успадковуються фреймом НАУКОВИЙ_СЕМІНАР_80. Слот Різновид мови фреймів аналогічний до дуги “є” семантичної мережі в тому розумінні, що він подає включення в клас або відношення належності до класу.

Наведену фреймову модель предметної галузі неважко подати також засобами логіки предикатів. Це можна зробити, наприклад, так. Введемо такі предикати і позначення для них:

$P(x, y)$ - “ x є y ” (“ x – різновид y ”);

$Q(x, y, z)$ - “слот y фрейма x має значення z ” (x, y – імена фрейма і слота відповідно).

Введемо також позначення для предметних констант.

ЗАХІД - a ;

НАУКОВИЙ_СЕМІНАР - b ;

НАУКОВИЙ_СЕМІНАР_80 - c ;

Час - b_1 ;

Місце - b_2 ;

Дата - c_1 ;

Доповідачі - c_2 ;

Вівторок_перед_третьою_середою_14.00 - b_{11} ;

Кафедра_інформатики_НПУ_імені_Драгоманова - b_{21} ;

14.11.2000 - c_{11} ;

[Красюк Ю., Лотюк Ю.] - c_{21} .

Модель дає можливість записати такі істинні формули:

$P(b, a); \quad Q(b, b_2, b_{21});$

$P(c, b); \quad Q(c, c_1, c_{11});$

$Q(b, b_1, b_{11}); \quad Q(c, c_2, c_{21})$

На основі концепції успадкування властивостей можна записати формулу

$\forall x \forall y \forall z \forall u (P(x, y) \wedge Q(y, z, u) \rightarrow Q(x, z, u))$

{Якщо фрейм x є різновидом y , а слот z фрейма y має значення u , то слот z фрейма x має значення u }.

Сім останніх формул описують предметну галузь, яка була подана фреймовою моделлю.

Щоб мати можливість виводити відповіді на запити методом резолюції, необхідно подати теорію галузі у вигляді рівносильної множини диз'юнктив:

1. $P(b, a);$

2. $P(c, b);$

3. $Q(b, b_1, b_{11});$

4. $Q(b, b_2, b_{21});$

5. $Q(c, c_1, c_{11});$

6. $Q(c, c_2, c_{21});$

7. $\neg P(x, y) \vee \neg Q(y, z, u) \vee Q(x, z, u).$

У формулі 7) квантори загальності опущено.

Для прикладу виведемо відповідь на запит: “Коли відбулося вісімдесяте засідання наукового семінару?”
Символічно запит запишеться так:

$$\exists v Q(c, c_1, v) \wedge \exists w Q(c, b_1, w)?$$

Для відшукування відповіді запишемо заперечення запиту і допишемо його до множини диз'юнктив аксіом теорій.

$$8. \neg Q(c, c_1, v) \vee \neg Q(c, b_1, w) \vee Answer(v, w).$$

Застосовуючи резолюцію до множини диз'юнктив, одержуємо:

$$9. \neg Q(c, b_1, w) \vee Answer(c_{11}, w) \quad (8,5) \quad \theta = \{v := c_{11}\};$$

$$10. \neg P(c, y) \vee \neg Q(y, b_1, w) \vee Answer(c_{11}, w)$$

$$11. (9,7) \quad \theta = \{x := c, z := b_1, u := w\};$$

$$12. \neg Q(b, b_1, w) \vee Answer(c_{11}, w) \quad (10,2) \quad \theta = \{y := b\};$$

$$13. \square \vee Answer(c_{11}, b_{11}) \quad (11,3) \quad \theta = \{w := b_{11}\}.$$

Одержана відповідь означає, що дата і час проведення семінару – це значення констант c_{11} і b_{11} відповідно. Тобто, засідання семінару повинно відбутися 14.11.2000 (це як раз і буде вівторок перед третьою серединою) о 14.00 год.

Для одержання потрібної відповіді, звичайно, можна було б задати два запити: один на визначення дати, а інший – часу.

В об'єктно-орієнтованому програмуванні основними складовими одиницями програмами є об'єкти. Кожний об'єкт містить деяку структуру даних (або тип даних) і набір процедур (методів), які призначені для опрацювання цих даних. Об'єкти з однаковим властивостями і поведінкою об'єднуються в класи. Кожен об'єкт входить в один клас і називається екземпляром класу. Кожен клас має одного предка, який називається суперкласом (надкласом). Клас може мати одного або декілька нащадків, які називаються підкласами. Є клас, який не має суперкласу і є коренем дерева ієрархії класів. Кожний клас успадковує змінні екземпляра і методи свого суперкласу. Крім того, він може містити нові змінні екземпляра і методи, може перевизначати успадковані.

Єдиний спосіб виконати обчислення полягає в тому, щоб послати об'єкту повідомлення. Для кожного повідомлення є відповідний метод (тобто процедура), який виконує обчислення, необхідні для відповіді на повідомлення.

На рис.4 показана ієрархія класів програми, яка несе інформацію про види птахів.

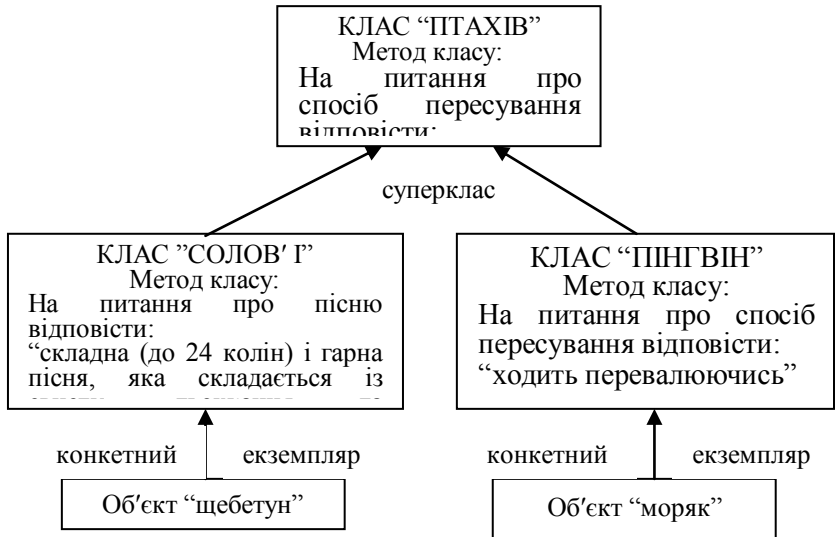


Рис.4

Якщо об'єкту "моряк" посилається повідомлення, в якому запитується про спосіб пересування, то воно опрацьовується методом класу ПІНГВІН і видається відповідь "ходить перевалюючись". Якщо послати таке саме повідомлення об'єкту "щебетун", то здійснюється звертання до класу СОЛОВ'І. Але оскільки в цьому класі відсутній метод для опрацювання повідомлення, то воно пересилається в клас ПТАХИ, який є суперкласом класу СОЛОВ'І. Повідомлення опрацьовується методом класу ПТАХИ і видається відповідь "літає".

Формалізм об'єктно-орієнтованого програмування можна трактувати як уточнення формалізму фреймів. У формалізмі фреймів не розрізняють види відношень клас/підклас і клас/конкретний екземпляр (тобто те саме відношення Різновид може існувати і як відношення між двома класами, і як відношення між класом і конкретним екземпляром). В об'єктно-орієнтованому програмуванні ці два види відношень розрізняються. Так, якщо б приклад з науковим семінаром було реалізовано мовою об'єктно-орієнтованого програмування, то НАУКОВИЙ_СЕМІНАР був би класом, а ЗАХІД був би суперкласом НАУКОВОГО_СЕМІНАРУ; НАУКОВИЙ_СЕМІНАР_80 був би конкретним екземпляром НАУКОВОГО_СЕМІНАРУ.

В об'єктно-орієнтованому програмуванні управління виконанням програми здійснюється за допомогою повідомлень між об'єктами, що майже аналогічно механізму управління

виводом за допомогою передачі повідомлень між фреймами. Проте на відміну від об'єктно-орієнтованої мови, яка має парадигми для універсального програмування, моделі подання знань фреймового типу мають парадигми для управління поданням знань і виводами, тобто за основними ідеями і конкретною реалізацією вони мають багато відмінностей.

ЛІТЕРАТУРА

1. Представление и использование знаний: Пер. с япон. / Под ред. Х.Уэно, М. Исидзука.-М.: Мир, 1989.- 220с.
2. Жалдак М.І., Рамський Ю.С. Інформатика: Навч. посібник.- К.:Вища шк., 1991.- 319 с.
3. Могилев А.В., Пак Н.И., Хённер Е.К. Информатика: Учеб. пособие для студ. пед. вузов.- М.: АСАДЕМА, 1999.-816 с.
4. Рамський Ю.С. Математична логіка і інформатика / В зб. наук. праць: Комп'ютерно–орієнтовані системи навчання.- К.: Компютер у школі та сім'ї.-1998.- с. 32-50.
5. Minsky M. F Framework for Representing Knowledge. In PCV,1975.- pp. 211-277.
6. Минский М. Фреймы для представления знаний: Пер. с англ. М.:Энергия, 1979.-151 с.