В. П. Бурдаєв

ПОДАННЯ ЗНАНЬ У СИСТЕМІ "КАРКАС"

Анотація. *Представлені результати циклу досліджень в області моделювання сучасних інформаційних предметних областей – від формалізації поняття розшарування бази знань до реалізації та дослідження моделі ієрархічної функціональної системи динамічної предметної області. Розглянуто побудови ієрархічної функціональної системи для подання знань з наступними властивостями: зв'язністю, складністю, стійкістю. Дано опис методу розшарування бази знань і приклади реалізації засобами системи "КАРКАС" в медичній та еколого-економічній предметних областях.*
Ключові слова: *ієрархічна функціональна система, база знань, правила продукцій, фрейми, складність предметної області, фільтрація бази знань, тестування знань.*
Abstract. *The results of the research cycle in the field of modeling modern information subject areas are presented – from the formalization of the concept of the stratification of the knowledge base to the realization and research of the model of the hierarchical functional system of the dynamic domain. The construction of a hierarchical functional system for representation of knowledge with the following properties is considered: connectivity, complexity, stability. A description is given of the method of stratifying the knowledge base and examples of implementation by means of the "KARKAS" system in the medical and environmental-economic subject areas.*
Keywords: *hierarchical functional system, knowledge base, product rules, frames, domain complexity, knowledge base filtering, knowledge testing.*

Введення і постановка задачі. Дані (факти) – це інформація, що описує об'єкти, процеси і явища предметної області (ПрО). Знання – це сприйнята живою істотою інформації із зовнішнього світу. Знання залежить від особливостей життєвого досвіду суб'єкта – від процесу його навчання або самонавчання. Знання передаються між суб'єктами за допомогою мови представлення знань (природна мова). Сформулюємо наступні відмінності знань від даних: у знаннях найбільше значення мають не елементи знань, а взаємозв'язки між ними; знання містять інформацію про те, як їх використовувати.

Таким чином, знання являють собою результат розумової діяльності людини, спрямованої на узагальнення його досвіду, отриманого в результаті практичної діяльності.

До типових моделей подання знань відносяться: логічна, продукційна, фреймова і семантична мережа.

Головна мета даної роботи полягає у вирішенні завдання подання знань з використанням ієрархічної функціональної системи, заснованої на розшаруванні бази знань.

Основна частина. З позицій об'єктно-орієнтованого програмування базовим поняттям в онтології є клас, який характеризується властивостями і методами. Властивості класу задаються значеннями його полів, а методи вирішують певні завдання.

Практично всі моделі антологій, в тій чи іншій мірі, містять концепти (сутності, поняття, класи, об'єкти), властивості концептів (атрибути, слоти), відносини між концептами (зв'язки, залежності).

Об'єкти (а точніше цілі і підцілі системи) не існують окремо один від одного. Між ними є реальні відносини, і вони повинні бути відображені в моделі бази знань (БЗ) предметної області. При виділенні відносин акцент робиться на фіксацію зв’язківків та їх характеристик. Ставлення (зв'язок) являє собою з'єднання (взаємовідношення) між двома або більше об'єктами, яке формує фільтрацію БЗ. Кожний зв'язок реалізується через значення атрибутів об'єктів.

Деяку аналогію можна привести з побудовою розшарування. Трійка об'єктів (M, p, B) утворює розшарування, де $p : M \to B$ проекція, B – база розшарування, $X = p^{-1}(b)$ – шар розшарування [1]. Наприклад, маємо два домена атрибутів V_1 і V_2 і розглянемо тривіальне розшарування pr_2: $V_1 \times V_2 \to V_2$. Виділимо в V_2 об'єкт $b:=g_{11}$, який буде визначати мету досягнення під час функціонування системи. Тоді перетин вказаного розшарування можна представити у вигляді графіка $b \mapsto (b, s(b))$, де $b \in V_2$ і $pr_2 \circ s(b)=b$. А саме перетин інтерпретуємо, як правило (продукцію), наприклад, якщо атрибут a_1 приймає значення $v_{11} \in V_1$, тоді ціль b приймає значення g_{11} (правило 1: якщо $a_1:=v_{11}$ (антецедент), то $b:=g_{11}$ (консеквент),cf:=1) з коефіцієнтом впевненості cf рівним одиниці.

$$\begin{array}{c} V_1 \times V_2 \\ pr_2 \downarrow \uparrow s \\ V_2. \end{array}$$

Для створення правил з однією цілю $b:=g_{11}$ в антецеденті перебираємо всі можливі пари $a_1:=v_{1i}$ $(i=1,\ldots,m)$, отже, отримуємо m різних правил. Експерт серед

цих правил відбирає тільки ті, які вважає потрібними для досягнення цілі системи. Відібрані правила відрізняються один від одного коефіцієнтом впевненості cf, який приймає значення з діапазону від 0 по 1. Таку процедуру створення правил назвемо вертикальним збуренням правила 1. У разі, якщо домен V_2 складається з однієї цілі, то визначення вертикальних збурень збігається зі звичайним визначенням збурення правил.

Цей варіант подання правила прийняття рішення можна поширити на випадок, коли є n атрибутів, що характеризують цілі досягнення системи. База розшарування в цьому випадку інтерпретується як множина головних цілей системи.

Нехай кожен з n атрибутів предметної області приймає відповідно m_n значень, тоді для цільового об'єкта $g_0 \in V_n$ можна отримати число $m_n \times n$ всіляких правил для досягнення цілі.

$$V_1 \times_{\dots} \times V_n$$
$$pr_2 \downarrow \uparrow s$$
$$V_n.$$

Іншими словами, представлена повна база знань для визначення досягнення цілі функціонування системи. Ясно, що серед цих правил є ті, які не дозволяють досягти цілі системи. Тому експерт з цієї множини правил відбирає тільки ті правила, які відповідають цілі системи, а інші правила може помітити для видалення, і вони не будуть використовуватися далі в системі для досягнення цілі.

Якщо в деякому домені V_k експертом виділяється підціль g_k, то аналогічно міркуючи, як і вище, отримуємо, що розшарування з базою V_k являється другим рівнем для досягнення цілі системи.

$$\widetilde{V}_{\ell} \times V_1 \times_{\dots} V_{k\dots} \times V_n$$
$$pr_k \downarrow \uparrow s_k$$
$$V_1 \times_{\dots} V_{k\dots} \times V_n$$
$$pr_2 \downarrow \uparrow s$$
$$V_n.$$

Таким чином, будується ієрархічна функціональна система предметної області.

Коли побудована БЗ, її можна доповнювати іншими правилами. Для цього в системі "КАРКАС" є модуль перевірки правила що додається на несуперечливість його з правилами бази знань. Продовжуючи цей процес виділення підцілей, отримуємо складну систему бази знань.

В системі "КАРКАС" процес побудови бази знань здійснюється за допомогою ієрархічної функціональної системи. Приклад фрагмента такої системи, перед консультацією для визначення ризику виникнення ішемічної хвороби серця (ІХС) наведено на рис. 1.

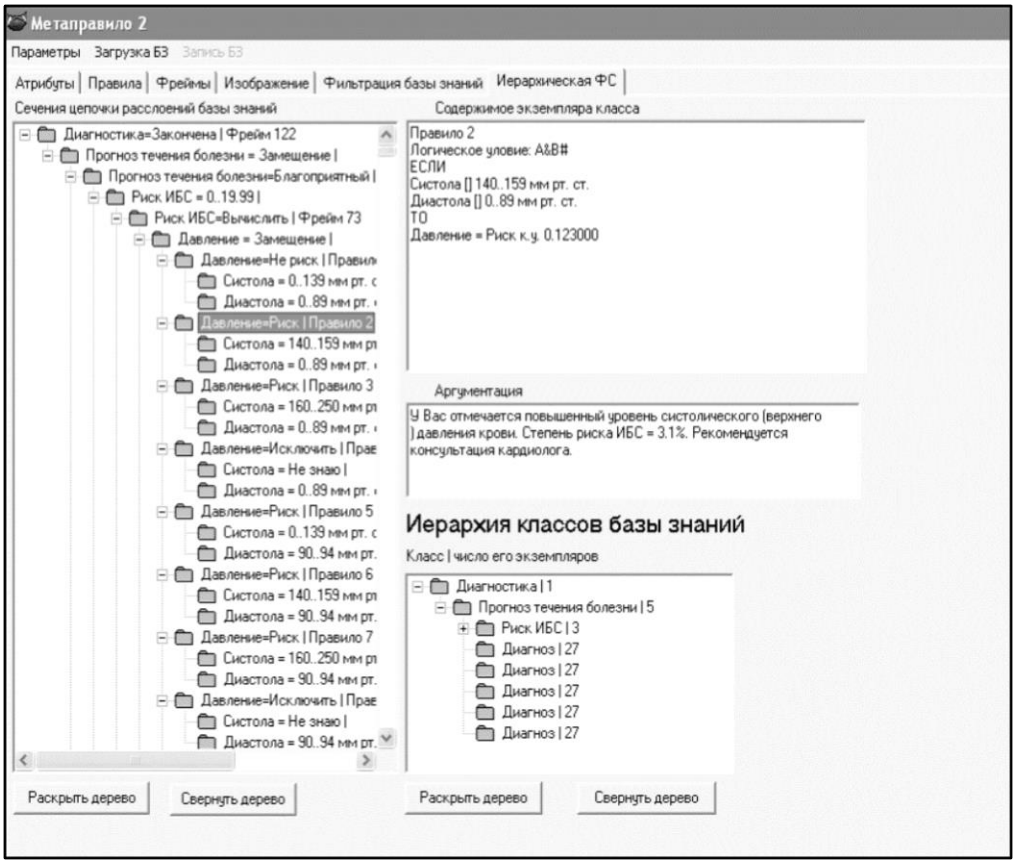

Рис. 1. Ієрархічна функціональна система для визначення ризику ІХС (перед консультацією)

Ієрархія класів в онтології будується шляхом виділення, так званого базового класу (основних цілей системи) що розташований на самому верхньому рівні ланцюжка розшарувань бази знань. Далі, виділяються підкласи, що знаходяться на наступному рівні і так далі. Базовий клас можна розглядати як логічну конструкцію моделі для подання множини об'єктів і зв’язків між ними як єдиного цілого (рис. 1).

Функціональна система (ФС) – це система, що сформована для досягнення заданого корисного результату (цільової функції) в процесі свого функціонування. Отже, її системоутворюючим фактором є конкретний результат ФС. Іншими словами, ціль розглядається як заданий результат, а обмеження – як ступінь свободи, що необхідна для досягнення результату.

Модель предметної області розглядається як ФС, в якій результат робить організуючий вплив на всі етапи формування онтології. Класи і зв'язки між ним можна розглядати як логічний конструкцію ФС.

Відмінна особливість ФС – в її відкритості, не автономності, не ізольованості від зовнішнього середовища. Математичною моделлю, що описує еволюцію таких систем, служать неавтономні диференційні рівняння. В системі "КАРКАС" модель ФС реалізована на основі використання онтології предметної області.

ФС можна розглядати як сукупність функцій з деяким набором операцій, що застосовуються до цих функцій. Роль функцій грають правила БЗ, а основні операції – це визначення умов застосовності правил. З іншого боку, ФС можна розглядати як систему висловлювань з логічними операціями над ними.

Важливою особливістю ФС є її контент предметної області. В роботі [1] досліджується математична модель динамічної предметної області у вигляді ієрархічної функціональної системи, в якій БЗ асоціюється з ланцюжком розшарувань баз знань, тобто являє собою перетин ланцюжка розшарувань БЗ.

Зауважимо, що при цьому кожний перетин ланцюжка розшарувань ФС має вигляд орієнтованого графа досліджуваного цільового об'єкта при фіксованому $b \in B$, іншими словами, стан ФС залежить від параметра b, що характеризує еволюції зовнішнього середовища. У деяких предметних областях параметр b можна розглядати як дискретну одиницю часу.

В системі "КАРКАС" формування орієнтованого графа і його зміна по параметру b відбувається динамічно під час роботи машини висновку при консультації користувача. Зміна перетину (правил і фреймів БЗ) прообразів ланцюжка розшарування бази знань (ієрархічної ФС), під час обробки їх машиною висновку в динаміці представлено на рис. 2.

ФС характеризується такими властивостями:

зв'язність – ланцюжок розшарувань бази знань;

складність – ієрархія рівнів локальних баз знань;

стійкість (динамічна поведінка системи) – структура орієнтованого графа ФС не змінюється при вертикальних збуренях правил. Іншими словами, здійснюється тільки зміна евристик (правил) локальних баз знань ланцюжка розшарувань, а база розшарування, яка інтерпретується як зовнішнє середовище, залишається незмінною.

Зв'язність ФС виражається в фільтрації бази знань. Нехай B_i – це локальна база знань, тобто містить правила продукції для визначення під цілі Gi, яка знаходиться на i – тому рівні в ієрархії ФС.

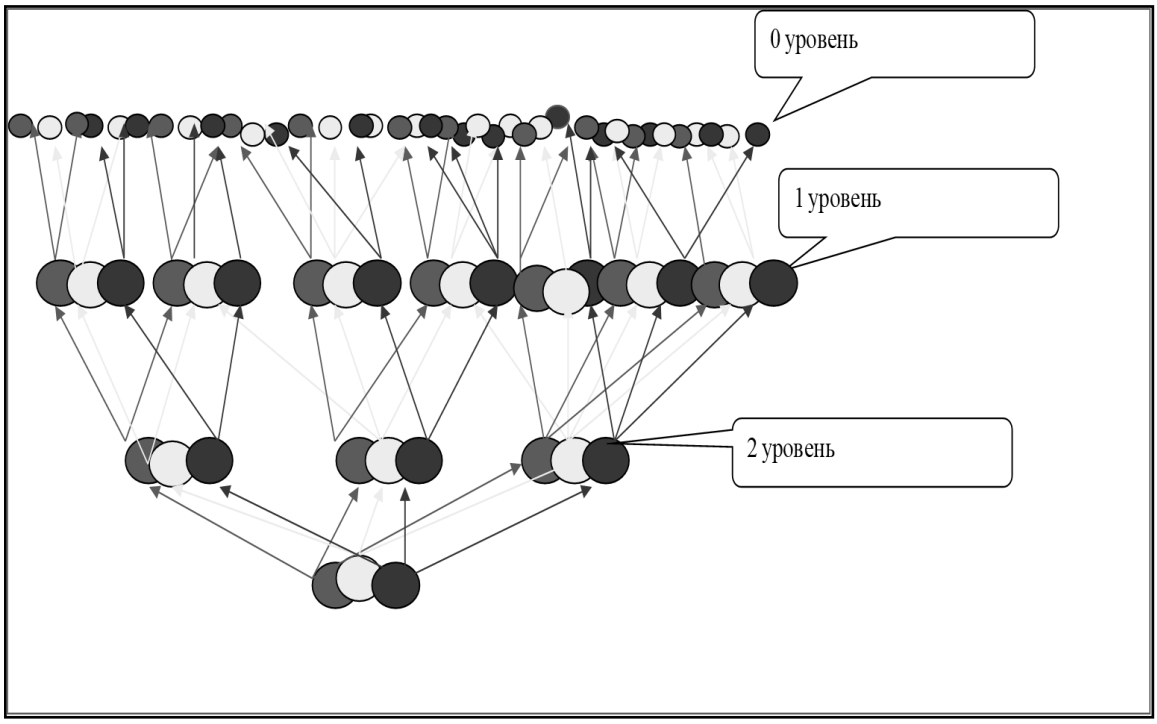

Рис. 2. Динамічна зміна ланцюжка перетину розшарування бази знань

Фільтрація бази знань – це кінцева система локальних баз знань B_i

$$B_0<=B_1<=\ldots<=B_k,$$

частково-упорядкованих (<=) наступним чином: консеквент кожного правила з B_i міститься в антецеденті правила з B_j ($i < j$).

Щоб побудувати фільтрацію бази знань досить вказати ланцюжок правил ФС для досягнення основної цілі. Потім за допомогою рекурсивного алгоритму будуються інші правила для локальних баз знань ФС шляхом генерування правил і під час консультації з експертом. Експерт аналізує правила надане машиною висновку ФС і може помістити створені правила в локальну базу знань на відповідному рівні ієрархії ФС або заборонити їх використання або дозволити їх використання на певний час. Таким чином, за допомогою алгоритму фільтрації бази знань здійснюється як поповнення локальних баз знань, так і їх адаптація до предметної області. Кількість локальних баз знань відповідає рівням ієрархічної ФС предметної області.

Складність ФС характеризує основну властивість ієрархічної системи. Наприклад, якщо в режимі налагодження БЗ на якомусь рівні ієрархії не знайшлося відповідного правила для прийняття рішення, то ситуація автоматично машиною висновку заноситься у фрейм і далі буде перехід до

наступного рівня, іншими словами консультація не переривається. Крім того, машина висновку формує правило продукції на цю ситуацію ФС і запам'ятовує в базі знань. Після консультації експерт може проаналізувати ситуацію, яка привела до появи нового правила прийняття рішення.

Якщо описана ситуація виникає під час режиму консультації, то машина висновку формує наступний стан ієрархічної функціональної системи на основі глобальної цілі з бази розшарування БЗ і продовжує консультацію.

Зауважимо, що розмірність ФС (множина станів або множина правил прийняття рішень) не означає велику складність її і навпаки.

Таким чином, ієрархічна функціональна система предметної області являє собою набір відфільтрованих локальних баз знань, які дозволяють машині висновку системи "КАРКАС" досягти локальної цілі на кожному рівні ієрархії і відповідно – глобальної цілі.

Отже, модель предметної області розглядається як ієрархічна ФС, в якій результат робить організуючий вплив на всі етапи формування онтології. Класи і зв'язки між ним можна розглядати як логічну конструкцію ФС.

Система "КАРКАС" – це інструментальний засіб для побудови онтології ПрО. Структура предметної області може бути різноманітною, наприклад, вибір рішення серед певного набору варіантів, використання ненадійних знань. Зовнішній вигляд системи представлений на рис. 3.

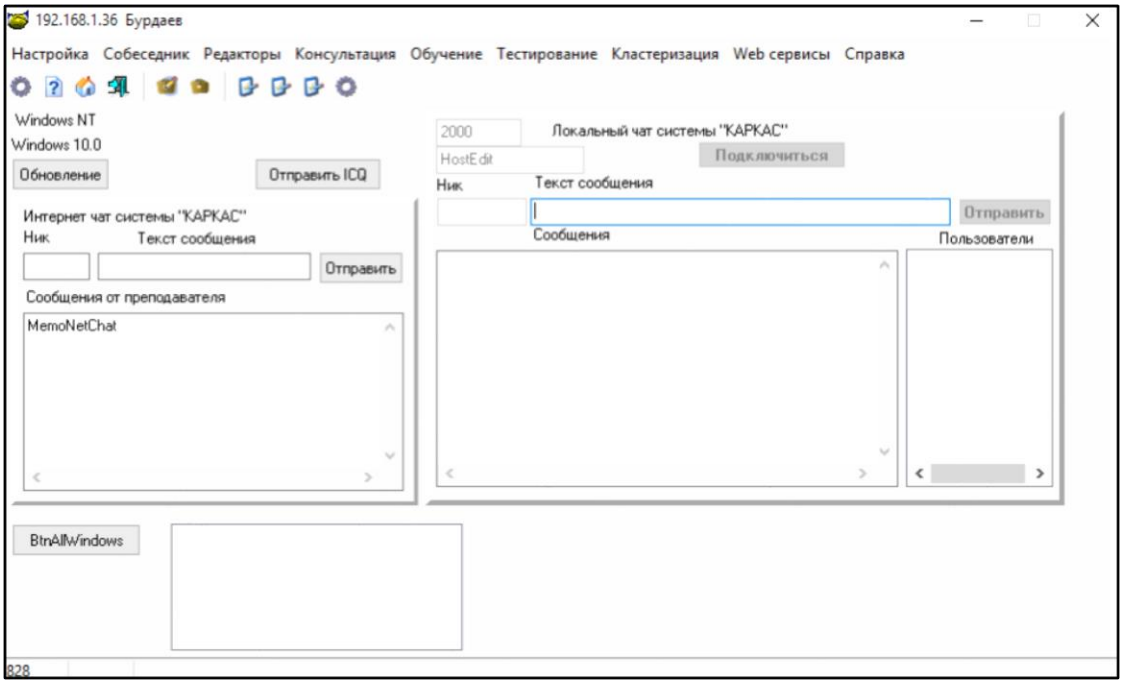

Рис. 3. Зовнішній вигляд системи «КАРКАС"

Система "КАРКАС" побудована за модульним принципом і з цієї причини має можливість приєднання інших додаткових модулів. В архітектурі системи можна виділити наступні основні модулі: візуальний редактор БЗ, співрозмовник (під'єднання месенджерів ICQ, TELEGRAM для віддаленої консультації з користувачем), консультація, навчання і тестування. Для тестування знань використовуються генератор тестів і монітор викладача. Модулі системи представлені у вигляді програмних пасивних агентів. Середовище їх взаємодії – це система "КАРКАС".

Модуль співрозмовник призначений для інтегрування месенджерів з системою «КАРКАС". Месенджери можна розглядати як певний вид браузера, а їх чат-боти як Web-додатки (з елементами штучного інтелекту).

Існує безліч програм для спілкування – Skype, Viber, WhatsApp, ICQ та інші. Але в бізнес-середовищі корпоративним стандартом комунікацій все частіше стає саме безкоштовний менеджер TELEGRAM. Це обумовлено наступними причинами: високим ступенем шифрування даних в ньому, стабільністю роботи, можливістю передачі великих обсягів інформації, відкритістю протоколу, кроссплатформеностью. З іншого боку, TELEGRAM надає бібліотеку на основі API для роботи з чат-ботами.

Майбутнє чат-ботів може бути тільки в ролі природно-мовної оболонки для експертних систем, на основі доступних сервісів для створення розмовних інтерфейсів (Api.ai, Dialogflow, Wit.ai) і платформи Cortana Intelligence від Microsoft.

Модуль он-лайнові консультації (співрозмовник) дозволяє за допомогою месенджера TELEGRAM або ICQ обмінюватися повідомленнями з базами знань системи "КАРКАС" через Інтернет, іншими словами здійснювати консультацію в режимі реального часу. Зауважимо, що формування бази знань і її налагодження здійснюються на локальному комп'ютері.

Алгоритм роботи модуля он-лайнові консультації складається з наступних кроків:

1. Підключення до сервера месенджера (використовується login і номер системи "КАРКАС" для ICQ або токен і адреса чат-бота для TELEGRAM).

2. Користувачу на його номер месенджера відсилається повідомлення про те, що система "КАРКАС" знаходиться в режимі онлайн і чекає повідомлення з кодовим словом, наприклад, "рібс" (модель бази знань по визначенню ризику ішемічної хвороби серця).

3. Після отримання відповіді від користувача завантажується машина висновку для конкретної моделі бази знань і здійснюється діалог між системою і користувачем.

4. В кінці діалогу користувачеві надсилається результат консультації.

5. У разі затримки відповіді від користувача 1 – 2 хвилини сеанс консультації припиняється.

У традиційному підході в реалізації інтерфейсу в експертних системах використовується обмежена природна мова або різноманітні графічні елементи управління. Поява чат-ботів дозволяє використовувати мовний інтерфейс. Зауважимо, що для пояснення користувачеві про те, як і чому сформований факт під час консультації можна використовувати мовний інтерфейс і анімовані персонажі в WINDOWS XP (рис. 5).

Парадигма інтегрування чат-ботів для роботи з експертними системами (ЕС) та експертно-навчальними системами (ЕНС) стає все більш актуальною. Використання TELEGRAM як співрозмовника при роботі з "КАРКАС" дає більше можливостей оперативно консультуватися з експертною системою через смартфон, що, наприклад, важливо для прийняття ефективних рішень в різних предметних областях таких як: медицина, екологія, бізнес. Іншими словами, тепер можна відправити текстове повідомлення запрограмованому чат-боту "РІБС" (бот для визначення ризику ішемічної хвороби серця) і отримати моментально необхідну інформацію, тобто здійснювати консультацію в режимі реального часу. Зауважимо, що формування бази знань і її налагодження відбуваються на локальному комп'ютері.

Таким чином, чат-бот займає таке ж місце, як і будь-який інший інтерфейс в експертній системі. Він дозволяє користувачеві як надати необхідні знання з предметної області (навчання і тестування), так і провести он-лайнові консультацію шляхом зовнішнього виклику модулів системи "КАРКАС" (доступ до пробної версії системи і чат боту на сайті https://www.it-karkas.com.ua).

В системі виділені два режими використання: режим когнітолога – фахівець моделює онтологію ПрО і конструює БЗ, і режим проблемного фахівця, іншими словами кваліфікація користувача недостатньо висока, і тому він потребує допомоги і підтримки своєї діяльності з боку системи.

Система реалізована за допомогою середовища програмування Embarcadero DX Delphi 10.2 Tokyo. Функції, реалізовані в системі, представлені в таблиці 1.

Таблиця 1

Функції системи "КАРКАС"

№	Опис функції
1	Створення БЗ на основі шаблонів для продукцій і фреймів
2	Візуальне редагування БЗ
3	Побудова онтології ПрО: класів, їх екземплярів (об'єктів), атрибутів
4	Кластеризація класів і їхня візуалізація у вигляді дерева об'єктів
5	Створення метаправил за допомогою "дошки оголошення"
6	Формування протоколу консультації
7	Динамічне формування тестів для навчання і контролю
8	Налаштування параметрів тестування
9	Візуалізація оцінок тестування під час консультації
10	Аналіз статистики тестування

В системі "КАРКАС" знання представляються за допомогою ієрархічної функціональної системи, яка використовує продукції, так і фрейми.

Антецеденти правил формуються з пар: атрибут = значення, які пов'язані між собою логічною умовою на їх застосування. Будь-яке правило складається з однієї або декількох таких пар. А консеквенти складаються з об'єктів класів онтології (рис. 4).

Рис. 4. Вид правила у візуальному редакторі системи "КАРКАС"

Фрейм має ім'я, що служить для ідентифікації описуваного ним поняття, і містить ряд описів – слотів, за допомогою яких визначаються основні структурні елементи цього поняття. Слот може містити не тільки конкретне значення, але також ім'я процедури, що дозволяє обчислити це значення за заданим алгоритмом. Фрейм представляє собою сукупність слотів, число яких може бути довільним.

В системі «КАРКАС" структура фрейму містить наступні атрибути:

- ім'я фрейму;
- ім'я слота;
- покажчики спадкування. Вони показують, яку інформацію про атрибути слотів з фрейма верхнього рівня успадковують слоти з аналогічними назвами в даному фреймі;
- демони. Демоном називається процедура, що автоматично запускається при виконанні деякої умови. Демони автоматично запускаються при зверненні до відповідного слоту.

В системі «КАРКАС" символ "н" означає, що значення слота успадковується.

Фрейм можна розглядати як клас, тоді екземпляр фрейма – це об'єкт, приєднана процедура – метод.

З кожним слотом може бути пов'язана така інформація: умова на заповнення (тип, "за замовчуванням", зв'язок з іншими слотами), асоційовані процедури (дії, що виконуються, наприклад, при заповненні цього слоту), рис. 5.

Рис. 5. Демон фрейма: заміщення

Машина висновка – це модуль, що моделює алгоритм міркувань з метою отримання нових фактів з БЗ для вирішення завдання. Машина висновку використовує такі методи висновку: зворотний висновок (від гіпотез до даних) встановлено за замовчуванням; прямий висновок (від даних до гіпотез); байєсовський висновок (застосування формули Байєса); нейлоровський висновок (застосування алгоритму ціни свідоцтв).

Модуль пояснення пояснює хід міркування і прийнятих рішень машиною висновку. Модуль дозволяє користувачеві (когнитологу) отримати відповіді на питання: "Як було отримано те чи інше рішення?". Алгоритм заснований на трасуванні машини висновка.

Трасування машини висновка надає користувачеві можливість стежити за її діями: в режимі перегляду бази фактів виводяться послідовні кроки консультації, імена правил і фреймів, які були активізовані. Є можливість відкоригувати факт, переглянути варіанти відповідей на поставлене запитання, простежити причинно-наслідковий зв'язок правил і фреймів.

Режим налагодження консультації надає можливість переглянути стан ієрархічної функціональної системи, додаткові параметри обробки правил і фреймів, що забезпечує систему переривань консультації для аналізу роботи машини висновку. Надається можливість вибору цілі (об'єкта онтології) консультації для локального тестування частини онтології предметної області.

Режим автоматичного тестування БЗ дозволяє виявити помилки, несумісність рішення задачі. Для користувача ці можливості корисні в тих випадках, коли БЗ перевіряється на повноту і несуперечність, коли БЗ модифікується (поповнюється або переглядається). Іншими словами, система здійснює контроль над зміною БЗ з позиції пошуку будь-яких помилок.

Режим фільтрації БЗ дозволяє встановити ієрархію кластерів БЗ. Причому, ієрархія кластерів важлива для зворотного висновка. Порядок розташування правил і фреймів в кластерах БЗ значення не має.

Якщо машина висновка під час консультації не може знайти інформацію (атрибут, об'єкт, правило, фрейм), то вона запитує її у користувача, при цьому вказується причина.

Алгоритм побудови онтології в системі "КАРКАС" складається з наступних кроків:

1 – визначення доменів атрибутів ПрО;

2 – визначення класів, об'єктів за допомогою продукцій;

3 – формування продукцій і фреймів для опису класів, підкласів, примірників;

4 – фільтрація БЗ;

5 – створення класів, формування ієрархічної функціональної системи.

6 – кластеризація об'єктів і організація їх ієрархії;

Наведемо приклад онтології ПрО для визначення експозиції при оцінці екологічного ризику. Експозиція (вплив) – це контакт організму (рецептора) з хімічним, фізичним або біологічним агентом. Детальний опис моделі онтології наведено в [3]. Ієрархічна функціональна система для визначення виду експозиції при оцінці ризику представлена на рис. 6, де виділена ієрархія класів і їх примірників.

Рис. 6. Ієрархічна функціональна система для визначення виду експозиції

Візуальний редактор БЗ "КАРКАС" – це модуль, що надає когнитологу можливість створювати БЗ в інтерактивному режимі. Редактор включає в себе шаблони мови подання знань (продукції, фрейми), підказки та інші сервісні засоби, що полегшують роботу з базою.

Редактор БЗ містить синтаксичний контроль логічної умови для продукції. Алгоритм синтаксичного контролю розглянуто в [2]. Для синтаксичного контролю введення атрибутів ПрО і їх значень в антецедент продукції в допомогу користувачу запропонований механізм вибору їх зі списку. Редактор БЗ дозволяє клонувати правила. Спочатку створюється правило для клонування. Наприклад, антецедент продукції містить чотири атрибута, число варіантів для першого атрибута дорівнює 3, відповідно для другого – 5, для третього – 4, для четвертого – 2 і консеквентна має один об'єкт. Тоді в результаті клонування цього правила система створить 120 варіантів різних комбінацій правил. Когнитологу залишиться тільки видалити непотрібні правила.

Для подання атрибутів (властивостей) використовується наступна модель: <N D K_1 K_2 A>. Тут N – лексема атрибута. Елемент D – домен атрибута. Елемент K_1 – це значення функції $\mu : Q \rightarrow [0,\ldots,1]$, що дозволяє інтерпретувати висловлювання Q з точки зору його істинності. Елемент K_2 характеризує значимість атрибута (функція $\theta : Q \rightarrow [0,\ldots,1]$). Елемент A характеризує аргументованість атрибута (додаткова інформація експерта про атрибут).

Для подання правил використовується наступна модель: < N L $S_1 \rightarrow S_2$ K_1 K_2 A>. Тут N – лексема правила. Елемент L – логічна умова правила (логічний вираз, що приймає значення істинно або хибно). Елемент $S_1 \rightarrow S_2$ – ядро правила (S_1 – антецедент, S_2 – консеквент). Елемент K_1 характеризує ступінь упевненості експерта в істинності ситуації S_2. Елемент K_2 характеризує значимість правила. Елемент A характеризує аргументованість правила.

Для подання фреймів використовується наступна модель: < N S G U K_1 K_2 A>. Тут N – лексема фрейму. Елемент S – слот (атрибут). Слот може містити не тільки конкретне значення атрибута, але також ім'я процедури, що дозволяє обчислити це значення за заданим алгоритмом. Процедури автоматично запускаються при зверненні до відповідного слоту. Слоти – об'єкти або інші фрейми. Елемент G – цільової слот (викликає процедуру для формування об'єкта). U – покажчик спадкування. Він показує, яку інформацію про атрибути активної бази знань успадковують слоти з аналогічними атрибутами в даному фреймі. В системі «КАРКАС" символ "н" означає, що значення слота успадковується. Дії покажчика спадкування, полягають в наступному, якщо в

результаті роботи машини висновку утворився деякий факт, то всі фрейми містять слоти з ім'ям атрибута у факті і з процедурою ЗАМІЩЕННЯ отримують значення атрибута з факту. Машина висновку використовує і ряд інших процедур: СУМА, ВВЕДЕННЯ, СУМА ЗНАЧЕНЬ, ВИЗНАЧИТИ. Елемент K_1 характеризує ступінь упевненості експерта в істинності значення цільового слота. Елемент K_2 характеризує значимість фрейма. Елемент А характеризує аргументованість фрейму.

На базі інструментарію системи "КАРКАС" створена БЗ для прототипу ЕС "МОБІЛЕ", яка призначена для пошуку мобільного телефону за вказаними параметрами шляхом вибору конкретних параметрів. Особливість складання правил для БЗ полягає в тому, що в консеквенті перераховуються моделі телефонів, а їх кілька сотень. Використання таких продукцій обтяжливе. Тому для подання знань використовується таблиця критеріїв телефонів з їх характеристиками і в консеквент заноситься рядок, в якому перераховуються моделі телефонів з цієї таблиці, і вказується відповідний рівень значущості цього параметра телефону. Оскільки для висновку застосовується прямой ланцюжок міркувань, то після опитування всіх параметрів відбувається сортування моделей телефонів з усіх консеквентів відпрацьованих правил і на перший рівень виносяться ті моделі телефонів, які отримали найвищий відсоток переваги [3].

Результат консультації представлений у вигляді списку переваг за вибором мобільного телефону. Для користувача виводяться характеристики мобільного телефону, за якими здійснювався відбір телефону, зображення телефону і повна характеристика його параметрів.

Система "КАРКАС" може бути використана для адаптованого тестування і навчання студентів по локальній мережі. В цьому режимі основне завдання системи – це надання можливості придбання студентом знань, умінь, навичок з розробки БЗ і створення прототипів ЕС самостійно.

Програмна реалізація системи "КАРКАС" в режимі навчання і тестування студентів заснована на використанні клієнт-серверної технології на основі програмування сокетів. При цьому система реалізує клієнт-серверну взаємодію: "товстий клієнт" – "тонкий сервер", тобто серверна частина реалізує тільки доступ до ресурсів системи (ідентифікація студента, встановлення зв'язку з ним і отримання оцінок тесту від нього), а основна частина програми (машини висновку, пояснення, навчання, БЗ) знаходиться на клієнті.

Іншими словами, система працює і як сервер і як клієнт. Залежно від постановки задачі має три режими функціонування:

1. У разі розробки прототипу БЗ система працює, як інструментарій для створення правил і фреймів.

2. При навчанні студентів (в режимі локальної мережі), один з комп'ютерів мережі оголошується робочим місцем викладача, на якому відстежується процес навчання. На інших комп'ютерах система працює в режимі навчання. Процес навчання відстежується на комп'ютері викладача.

3. У разі адаптованого тестування система використовує сокети для передачі статистик (оцінки, помилки тестованих, образи робочого столу комп'ютера) на комп'ютер викладача.

Висновок. Система "КАРКАС" являє собою інструментарій для розробки прототипів баз знань для ЕС і ЕНС. Подання знань ґрунтується на ієрархічній функціональній системі, яка генеруються системою "КАРКАС" на базі правил продукцій і фреймів. Машина висновку використовує ієрархічну функціональну систему під час проведення консультації з користувачем. Користувач може вибрати різні режими машини висновку: зворотний висновок (від гіпотез до даних) встановлено за замовчуванням; прямий висновок (від даних до гіпотез); байєсовський висновок (застосування формули Байєса); нейлоровський висновок (застосування алгоритму ціни свідоцтв), таблиця критеріїв. За допомогою системи "КАРКАС" розроблений ряд прототипів ЕС в наступних предметних областях: медицина, економіка, мобільний зв'язок і кластерний аналіз багатовимірних даних.

Список літератури

1. В. П. Бурдаев, ″Использование протокола мессенджера ICQ для онлайн консультации с экспертной системой″, на Міжнарод. наук.-практ. конф. Проблеми і перспективи розвитку ІТ-індустрії, Харків, 2017, с. 15.

2. В. П. Бурдаєв, Системи навчання з елементами штучного інтелекту. Харків, Україна: ХНЕУ, 2009.

3. В. П. Бурдаєв, Моделі баз знань. Харків, Україна: ХНЕУ, 2010.

4. В. П. Бурдаев, Сложность динамических систем: монография. Saarbrucken, LAP LAMBERT Academic Publishing, 2015.

5. Burdaev V. P. ″About one concept of constructing a temporal knowledge base″, in Proc. of the first International Congress Fundamental and Applied Studies in the Pacific and Atlantic Oceans Countries, Tokyo University Press, 2014, pp. 272–276.