

СЛУЖБА
ИДЕКЦИЈА

4
2002

ОСНОВНІ ТЕХНОЛОГІЧНІ ХАРАКТЕРИСТИКИ ПРОЦЕСУ СТВОРЕННЯ ПРОТОТИПУ ЕКСПЕРТНОЇ СИСТЕМИ ДІАГНОСТИКИ І ЛІКУВАННЯ РІЗНИХ ВАРІАНТІВ ВІРУСНИХ ГЕПАТИТІВ

ШКУРБА А.В., БУЛАХ І.Є.

*Національний медичний університет ім. О.О. Богомольця,
кафедра інфекційних хвороб,
зав. кафедрою — академік АМН України, проф. Ж.І. Возіанова,
кафедра медичної інформатики та комп'ютерних технологій навчання,
зав. кафедрою д. п. н., проф. І.Є. Булах*

ключові слова:

*експертні системи,
комп'ютерні технології, вірусні гепатити*

Програмні засоби, що базуються на технології і методах штучного інтелекту, одержали значне поширення у світі. Важливість їх, а, особливо, експертних систем і нейронних мереж, полягає в тому, що дані технології істотно розширюють коло практично значимих завдань, які можна вирішувати за допомогою комп'ютера, і їх розв'язання приносить значний економічний ефект [1]. У той же час технологія експертних систем є важливим засобом у рішенні глобальних проблем традиційного програмування — тривалість і, отже, висока вартість розробки додатків, супроводу складних систем, повторне використання програм і т. д. Крім того, об'єднання технологій експертних систем та нейронних мереж з технологією традиційного програмування додає нові якості до комерційних продуктів за рахунок забезпечення динамічної модифікації додатків користувачем, а не програмістом, більшої "прозорості" додатка (наприклад, знання зберігаються на природній, властивій цьому регіону мові, що не вимагає обтяжливих, складних коментарів до них, спрощує навчання і супровід), кращих графічних засобів, користувальницького інтерфейсу і взаємодії [2].

На думку фахівців, вже зараз експертні системи відіграють провідну роль у всіх фазах проектування, розробки, виробництва, розподілу, продажу, підтримки і надання послуг [6]. Перспектива їх розвитку у майбутньому має ще більше зрости. Їхня технологія, одержавши комерційне поширення, забезпечить значний прорив в інтеграції комп'ютерних додатків з готових інтелектуально-взаємодіючих модулів. Не оминув цей процес і медицину, й інсектологію, де з успіхом використовується ціла низка різноманітних експертних систем — *Mycin*, *Нераexpert* та ін. [9].

Комерційний ринок продуктів штучного інтелекту у світі в 1998 році оцінювався приблизно в 1,8 млрд. доларів; з них близько 1 мільярда припадає на частку США. Якщо в 1988 році прибуток від впровадження тільки експертних систем становив лише 3 млн. доларів, то в 1993 році вже досягнув 55 млн. доларів. Передбачують, що прибуток у 2005 році буде сягати щонайменше 500 млн. доларів [1]. Виділяють кілька основних напрямків цього ринку:

- 1) експертні системи, як їх часто позначають ще одним терміном — "системи, засновані на знаннях";
- 2) нейронні мережі і "розмиті" логіки;
- 3) природно-мовні системи.

Хоча використання експертних систем і нейронних мереж приносить значний економічний ефект, але комерційні успіхи прийшли не відразу. Протягом ряду років, починаючи з 1960-х, успіхи спостерігалися в основному в наукових дослідженнях, де розробки демонстрували придатність систем штучного інтелекту для практичного використання. Починаючи з 1988-1990 років, насамперед експертні системи, а в останні роки і нейронні мережі стали активно використовуватися в реальних комп'ютерних додатках.

Причини, що привели системи штучного інтелекту до комерційного успіху, наступні:

- перехід від розробки інструментальних засобів загального призначення до проблемно або предметноспеціалізованих засобів, що забезпечує скорочення термінів розробки додатків, підвищує ефективність використання інструментарію, спрощує і прискорює роботу експерта, дозволяє повторно використовувати інформаційне і програмне забезпечення;
- перехід від систем, заснованих на спеціалізованих мовах штучного інтелекту, до мов традиційного програмування спростив "інтегрованість" і знизив вимоги додатків до швидкодії та ємності пам'яті;
- розроблено інструментальні засоби штучного інтелекту, що легко інтегруються з іншими інформаційними технологіями і засобами;
- розробки ведуться з дотриманням стандартів, які забезпечують дані характеристики;
- створення розподіленої інформаційної системи у даній архітектурі дозволяє знизити вартість устаткування, використовованого в додатку, децентралізувати його, підвищити надійність і загальну продуктивність, оскільки скорочується обсяг інформації, що пересилається між електронно-обчислювальними машинами, і кожен модуль додатка будується на адекватному устаткуванні.

Перераховані причини можуть розглядатися як загальні вимоги до інструментальних засобів створення систем штучного інтелекту [3].

Серед спеціалізованих систем, заснованих на знаннях, найбільш значимі експертні системи реального часу, або, як їх ще називають, "динамічні експертні системи" [5]. Класи задач, які розв'язуються експертними системами реального часу, такі:

- моніторинг у реальному масштабі часу;
- системи керування верхнього рівня;
- системи виявлення несправностей;
- діагностика;
- складання розкладів;
- планування;
- оптимізація;
- системи-порадники оператора;
- системи проектування.

Статичні експертні системи не здатні вирішувати подібні задачі [4], тому що вони не виконують вимог, запропонованих до систем, що працюють у реальному масштабі часу:

1. Представляти дані, що змінюються в часі, які надходять від зовнішніх джерел, забезпечувати збереження й аналіз даних, які змінюються.
2. Виконувати механізм створення судження при кількох різних асинхронних процесах одночасно, тобто планувати відповідно до пріоритетів обробки процесів, що надійшли в систему.
3. Забезпечувати вищезгаданий механізм при обмежених ресурсах (час, пам'ять). Реалізація цього механізму ставить вимоги до високої швидкості роботи системи, здатності одночасно вирішувати кілька задач (тобто в операційних середовищах UNIX, VMS, SUN, Windows NT, 95, 2000, XP, але не в MS-DOS).

4. Забезпечувати "передбачуваність" поведження системи, тобто гарантію того, що кожна задача буде запущена і довершена в суворій відповідності з тимчасовими обмеженнями.

5. Моделювати "навколишній світ", розглянутий у даному додатку, забезпечувати створення різних його станів.

6. Протоколювати свої дії і дії персоналу, забезпечувати повне відновлення після збою.

7. Забезпечувати наповнення бази знань для додатків реального ступеня складності з мінімальними витратами часу і праці (необхідне використання об'єктно-орієнтованої технології, загальних правил, модульності і т.п.).

8. Забезпечувати настроювання системи на розв'язуванні конкретного завдання (проблемна-предметна орієнтованість).

9. Забезпечувати створення і підтримку користувальницьких інтерфейсів для різних категорій користувачів.

10. Забезпечувати рівень захисту інформації (по категоріях користувачів) і запобігати несанкціонованому доступу.

Інструментарій для створення експертних систем реального часу вперше випустила фірма Lisp Machine Inc у 1985 році. Цей продукт призначався для символічних EOM Symbolics і звався Pison. Його успіх привів до того, що група провідних його розробників утворила фірму Gensym, що, значно розвивши ідеї, закладені в Pison, випустила в 1988 році інструментальний засіб (оболонку) під назвою G2 [8]. В даний момент працює його четверта версія, в якій реалізовані додаткові проблемно-орієнтовані середовища та графічні мови. З відставанням від Gensym на два-три роки ряд інших фірм почав створювати свої інструментальні засоби — RTworks (Talarian, США), COMDALE/C або /X (Comdale Techn., Канада) та ін. Але найдоступнішим і найпростішим у використанні залишився продукт фірми Gensym, що й обумовило його застосування нами у створенні експертної програми діагностики і лікування різних варіантів вірусних гепатитів. Окрім цього старі версії оболонки G2 доступні для ознайомлення і використання через сайти мережі Internet, тобто не підпадають під дію обмежувальних законів.

Створення експертних систем зазвичай є все-таки копіткою роботою, що потребує залучання багатьох спеціалістів. Тому цей процес є тривалим та виснажливим, через що, незважаючи на нашу багаторічну працю над створенням експертної програми, поки що всі її компоненти не є доведеними до комерційного використання. Система знаходиться у стані закінчення розробки прототипу та "шліфувально-го" доведення її компонентів до потреб та особливостей роботи системи охорони здоров'я у нашій країні.

Нами на основі оболонки G2 закінчується розробка прототипу додатка (тобто експертної системи), визначено функції, що ним мають виконуватись. При розробці прототипу не використовується традиційне програмування. Прототип, як і в наступному додатку, створюється на структурованій природній мові, з використанням об'єктної графіки, ієрархізації класів об'єктів, правил, динамічних моделей зовнішнього світу. Багатослівність мови зведена до мінімуму шляхом введення операції клонування, що дозволяє розмножити будь-яку сутність бази знань.

Розширення прототипу до додатка проводиться через розширювання і модифікування бази знань. Під час цієї роботи прототип розвивається до такого стану, що починає задовольняти уявленням кінцевого користувача. У великих додатках команда розроблювачів може розбити додаток на окремі модулі, що інтегруються в єдину базу знань.

Проводиться тестування додатка на наявність помилок. У G2 помилки в синтаксисі показуються безпосередньо при введенні конструкцій (структур даних і тверджень, що виконуються) у базу даних. Можуть бути введені тільки конструкції, що не містять синтаксичних помилок. Таким чином, відпадає ціла фаза налагодження додатка (властива традиційному програмуванню), що прискорює розробку додатків. Розроблювач звільнений від необхідності знати детальний синтаксис мови

G2, тому що при введенні в базу знань деякої конструкції йому у вигляді підказки повідомляється перелік усіх можливих синтаксично правильних продовжень[8].

Для виявлення помилок і невизначеностей реалізована можливість "Inspect", що дозволяє переглядати різні аспекти бази знань, наприклад, "показати усі твердження з посиланнями на невизначені сутності (об'єкти, зв'язки, атрибути)", "показати графічно ієрархію заданого класу об'єктів" і таке інше.

Блок динамічного моделювання дозволяє при тестуванні відтворити різні ситуації, адекватні зовнішньому світу. Таким чином, логіка додатка перевіряється в тих умовах, для яких вона створювалася. Користувачі (лікарі, що будуть безпосередньо працювати з системою) беруть участь у тестуванні завдяки керуванню кольором (тобто коли відбувається зміна кольору при настанні заданого чи стану виконання умови) і анімації (тобто переміщення або обертання сутності при настанні стану чи умови). Завдяки цьому вони зможуть зрозуміти й оцінити логіку роботи додатка, не аналізуючи правила і процедури, а розглядаючи тільки графічне зображення керованого процесу і т.п. [7].

Для перевірки виконання обмежень використано можливість "Meters", яка обчислює статистику продуктивності і використання пам'яті.

Отриманий додаток через універсальність засобів G2 цілком переноситься на різні платформи в операційні середовища UNIX, Sun OS, DEC, HP-UX, VMS, Windows NT, -95, -2000 й XP. База знань зберігається в звичайному ASCII-файлі, що однозначно цілком інтерпретується на кожній з підтримуваних платформ. Перенос додатка не вимагає його перекомпіляції і полягає в простому переміщенні файлів. Функціональні можливості і зовнішній вигляд додатка не перетерплюють при цьому ніяких змін. Додаток може працювати як у "повному режимі" (тобто призначений для розробки) середовища, так і під обмежений "runtime", що не дозволяє модифікувати базу знань — тобто не викликати в ній небажаних змін через помилковість роботи недосвідчених користувачів. Останній режим роботи зазвичай характерний для комерційного використання експертних систем.

Не тільки сам розроблювач даного додатка, але і будь-який користувач може легко його зрозуміти і супроводжувати, тому що всі об'єкти або класи, правила, процедури, функції, формули, моделі зберігаються в базі знань у вигляді структурованої природної мови й у вигляді графічних об'єктів. Для її перегляду використовується можливість "Inspect". Супровід спрощується й за рахунок того, що різним групам користувачів може видаватися не вся інформація, а тільки її частина, яка відповідає їхнім потребам.

Наш прототип експертної системи реального часу з питань діагностики та лікування різних варіантів вірусних гепатитів складається з бази знань, машини висновку, підсистеми моделювання і планувальника системи. Також додатково можуть бути задіяні у процесі роботи інформаційний блок та прикладні програми, які здатні вирішувати обмежені проблеми, що можуть виникнути в процесі роботи системи (оцінювання КОС, загрози виникнення фульмінантної печінкової недостатності та ін.).

Усі знання в нашому прототипі зберігаються в двох типах файлів: бази знань і бібліотеки знань. У файлах першого типу зберігаються знання про додатки: визначення всіх об'єктів, об'єкти, правила, процедури і т.п. У файлах бібліотек зберігаються загальні знання, що можуть бути використані більш ніж в одному додатку, наприклад, визначення стандартних об'єктів. Файли баз знань можуть перетворитися в бібліотеки знань і навпаки.

З метою забезпечення повторного використання додатків реалізовано засіб, що дозволяє поєднувати з поточним додатком раніше створені бази і бібліотеки знань. При цьому конфлікти в поєднаних знаннях виявляються і відображаються на дисплеї монітору.

Знання структуруються — передбачена відповідна ієрархія класів, модулів, робочих просторів. Кожну з них також можна показати на дисплеї. Клас, базове поняття об'єктно-орієнтованої технології — основа подання знань в оболонці G2. Даний підхід становить основну тенденцію в програмуванні взагалі, оскільки він

зменшує надмірність і спрощує визначення класів (визначається не весь клас, а тільки його відмінності від суперкласу), дозволяє використовувати загальні правила, процедури, формули, зменшує їхнє число та й є природним для людини способом опису сутностей. При такому підході структури даних представляються у вигляді класів об'єктів (чи визначень об'єктів), що мають визначені атрибути. Класи успадковують атрибути від суперкласів і передають свої атрибути підкласам.

Усе, що зберігається в базі знань і з чим оперує система, є екземпляром того чи іншого класу. Більш того, усі синтаксичні конструкції прототипу є класами. Для збереження спільності навіть базові типи даних — символічні, числові та інші значення нечіткої логіки — представлені відповідними класами. Опис класу включає посилання на суперклас і містить перелік атрибутів, специфічних для класу.

Для структуризації прототипу використовуються "модулі" і "робочі простори". Незважаючи на те, що функції цих конструкцій схожі, між ними є істотні розходження. Додаток може бути організований у вигляді однієї чи кількох баз знань, названих модулями. В останньому випадку говорять, що додаток представлений структурою (ієрархією) модулів. Робочі простори є контейнерним класом, у якому розміщуються інші класи і їхні екземпляри, наприклад, об'єкти, зв'язки, правила, процедури і т.д. Кожен модуль (база знань) може містити будь-яке число робочих просторів. Розходження між "модулями" і "робочими просторами" полягає в наступному. Модулі розділяють додаток на окремі бази знань, спільно використовувані в різних додатках. Вони корисні в процесі розробки додатка, а не його виконання. Робочі простори, навпаки, виконують свою роль при виконанні додатка. Вони містять у собі різні сутності і забезпечують розбивку додатка на невеликі частини, що легше зрозуміти й обробляти.

Сутності в базі знань з погляду їхнього використання можна розділити на структури даних і виконувани твердження. Прикладами перших є об'єкти і їхні класи, зв'язки, відносини, перемінні, параметри, списки, масиви, робочі простори. Прикладами других — правила, процедури, формули, функції.

Основу виконуваних тверджень баз знань становлять правила і процедури. Правила в прототипі мають традиційний вид: ліва частина (антецедент) і права частина (консеквент). Крім них, використовується ще чотири типи правил: *initially*, *unconditionally*, *when* і *whenever*. Кожне з типів правил може бути як загальним, тобто таким, що стосується всього класу, так і специфічним, що стосується до конкретних екземплярів класу.

Можливість представляти знання у вигляді загальних правил, а не тільки спеціалізованих, дозволяє мінімізувати надмірність бази знань, спрощує її наповнення і супровід, скорочує число помилок, сприяє повторному використанню знань (загальні правила запам'ятовуються в бібліотеці і можуть використовуватися в подібних додатках).

Мова програмування, використовувана у прототипі на основі G2 для представлення процедурних знань, є досить близьким родичем Паскаля. Крім стандартних керуючих конструкцій, мова розширена елементами, що враховують роботу процедури в реальному часі: чекання настання подій, дозвіл іншим задачам переривати її виконання, директиви, що задають послідовне чи рівнобіжне виконання операторів.

Для машини висновку G2 властивий багатий набір способів порушення правил. Передбачені, наприклад, такі випадки: "Визначений об'єкт на екрані переміщений користувачем чи іншим правилом", "Визначене відношення між об'єктами встановлене чи знищено", "Перемінна не одержала значення в результаті звертання до свого джерела даних" і т. д.

У зв'язку з тим, що експертна система керує значною кількістю задач, що одночасно виконуються, необхідний планувальник системи. Хоча користувач ніколи не взаємодіє з ним, "машинний" планувальник контролює всю активність, видиму користувачем. Планувальник визначає порядок обробки задач, взаємодіє з джерелами даних і користувачами, запускає процеси і здійснює комунікацію з іншими процесами в мережі.

Таким чином, за допомогою оболонки G2 створено прототип експертної системи діагностики та лікування різних варіантів вірусних гепатитів значно швидше, ніж з використанням традиційних методів програмування, що дозволить відчутно знизити працевитрати на супровід і перенос її на інші можливі платформи та сервіси комп'ютерного забезпечення, які використовуються в нашій країні. Загалом створений прототип може бути застосований при комерційному використанні або через державну систему охорони здоров'я в Україні, що дозволить уніфікувати процес діагностики та лікування вірусних гепатитів і допоможе знизити кількість помилок, зменшити відсоток ускладнень, удосконалити терапію хворих на цю патологію.

ЛІТЕРАТУРА

1. Попов Э.В. с соавт. Статические и динамические экспертные системы // Финансы и статистика — 1996.— 318 с.
2. Boegl K. Design and Implementation of a Web-Based Knowledge Acquisition. Toolkit for Medical Expert Consultation Systems. Dissertation.— 1997.— Technische Universitet Wien, Osterreich.
3. Colomb R.M. Representation of propositional expert systems as partial functions. Artificial Intelligence (to appear). — 1999.— From <http://www.csee.uq.edu.au/colomb/>
4. Manickam S., Abidi S. S. R. Experienced Based Medical Diagnostics System over The World Wide Web (WWW), Proceedings of The First National conference on Artificial Intelligence Application In Industry. — 1991.— Kuala Lumpur.— Pp. 47 - 56.
5. McNamee L. A. et alt. Scaling an Expert System Data Mart: More Facilities in Real-Time.— Section of Medical Informatics, Washington University, St. Louis, Missouri / Information Systems, BJC Health System, St. Louis, Missouri. — 1999.— 12 p.
6. Menzies T. Critical success metrics: Evaluation at the business-level. — International Journal of Human-Computer Studies, special issue on evaluation of KE techniques.— 2000.— P. 48-66.
7. Menzies T., Cukic B. Adequacy of Limited Testing for Knowledge Based Systems. — NASA/WVU Software Research Lab, 100 University Drive, Fairmont, WV USA/Dept. Computer Science and Electrical Engineering, West Virginia Uni., Morgantown, WV USA.—April 10, 2000.—P.18
8. Moore B. et al. Questions and Answers about G2.— 1993.— Gensym Corporation.— Pp.26-28.
9. Shortliffe E. H., Fagan L. M., Yu V. L. The Infectious Diseases Physician and the Internet. In Mandell, G.L., Bennett, J.E. and Dolin, R. (Eds.), Mandell, Douglas, and Bennett's Principles and Practice of Infectious Diseases. — 2000.— Churchill Livingstone, Inc., Pennsylvania.— Pp. 3258-3263.

УДК: 616.36-002-022-036.1-07-08:579.89
Шкурба А.В., Булах И.Е.

Основные технологические характеристики процесса создания прототипа экспертной системы диагностики и лечения различных вариантов вирусных гепатитов

В работе охарактеризованы основные особенности программных средств, базирующихся на технологиях и методах искусственного интеллекта, преимущественно экспертных систем. Очерчены те свойства их, которые привели к коммерческому успеху в различных отраслях, в том числе и в медицине, инфектологии. Описан ход создания прототипа экспертной системы диагностики и лечения различных вариантов клинического течения вирусных гепатитов. Охарактеризованы ключевые моменты и особенности этого процесса.

UDC: 616.36-002-022-036.1-07-08:579.89
Shkurba A.V., Bulach I.E.

The basic technical characteristics of creation's process of the expert system's prototype of diagnostics and treatment of various variants viral hepatitis

In work the basic features of software basing technologies and methods of artificial intelligence, mainly of expert systems are characterized. Those properties them are outlined which have resulted in commercial success in various branches, including in medicine, infectology. The course of creation of the expert system's prototype of diagnostics and treatment of various variants of clinical current viral hepatitis is described. The key moments and features of this process are characterized.