

Список використаних джерел:

1. Реута О.В. Порівняння матричної і воксельної моделей тривимірного тіла для задачі його реконструкції // Прикладна геометрія та інженерна графіка. – К.: КНУБА, 2009. – Вип. 82. – С. 203–207.
2. Реута О.В. Використання напівпрозорих вокселів в дискретних моделях просторових об'єктів // Праці Таврійського державного агротехнічного університету. Прикладна геометрія та інженерна графіка. – Мелітополь: ТДАТУ, 2012. – Вип. 4, т. 52. – С. 107–111.

Вакарь Л.Г.

студент,

Харківський національний університет радіоелектроніки

КОНЦЕПЦІЯ ПОБУДОВИ ПРОДУКЦІЙНОЇ СИСТЕМИ ДЛЯ ДВИГУНА ВИСНОВКІВ ЕКСПЕРТНИХ СИСТЕМ

В сучасному технічно складному світі, експерти відіграють дуже важливу роль. Вони є незамінним елементом в структурі багатьох організацій. Проте людські експерти мають ряд недоліків: суб'єктивне судження, обмеження швидкості прийняття рішень, невеликий час роботи на протязі доби тощо. Тому останнім часом, в галузях де прийняття рішень можливе на основі набору відомих фактів без особистої участі експерта, активно впроваджуються експертні системи. Експертна система – це комп'ютерна програма, здатна імітувати поведінку людського експерта у вузькій галузі знань [1, с. 52]. Наприклад, експертна система CaDet здатна розпізнавати рак на ранній стадії.

Експертна система складається з кількох компонентів: бази знань, двигуна висновків, модулю придбання знань, інтерфейсу пояснення [2]. База знань це сукупність правил збережених у форматі “якщо-то”. Двигун висновків (рис. 1) витягує правила із бази знань і, якщо відомі факти відповідають предикатам правила, застосовує його для виведення нових фактів.

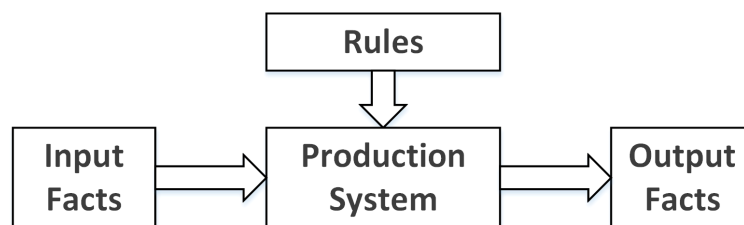


Рис. 1. Схема роботи двигуна висновків

Для виконання правил та отримання нових фактів двигуни висновків використовують алгоритми продукційних систем (ПС). Сучасні експертні системи використовують двигуни висновків створені на основі імплементацій rete-алгоритму. Rete-алгоритм становить собою дуже швидкий засіб зіставлення фактів з предикатами правил. Висока швидкодія досягається шляхом зберігання

в оперативній пам'яті інформації про правила представлені у вигляді мережі [3, с. 83]. Правила для rete-алгоритму повинні бути написані з використанням спеціально розробленого синтаксису. Rete-алгоритм дозволяє виконувати двигуну висновків тільки ті правила які можуть бути виконані виходячи з наданих вхідних фактів.

Попри свою ефективність, моделі rete-алгоритму базуються на правилах, синтаксис яких не може охопити складну бізнес-логіку. На даний момент це питання вирішується додаванням до двигуна висновків додаткового шару бізнес-логіки [4]. Цього можна уникнути якщо представити правила у вигляді класів об'єктно-орієнтованої мови програмування (ООМП). Такі правила можуть мати безпосередній доступ до API додатку в якому використовується двигун висновків. На жаль rete-алгоритм не може бути застосований до правил написаних на ООМП через складність та непередбачуваності синтаксису. Двигуни висновків які здатні обробляти такі правила, наприклад EasyRules, використовують алгоритм Маркова як основу ПС. Алгоритм Маркова – це упорядкована група правил, які застосовуються в порядку пріоритету [3, с. 81]. На відміну від rete-алгоритму, алгоритм Маркова вимушений тестувати більше правил за необхідну, для прийняття експертного рішення, кількість. Це впливає на швидкість роботи двигуна висновків [3, с. 82].

У зв'язку з цим є актуальною розробка концепції ПС яка б поєднувала в собі можливість використовувати в правилах складну бізнес-логіку та ефективно обирати тільки необхідні для експертного рішення правила. Концепція ПС повинна поєднати дві базові ідеї. Перша це представлення правил у вигляді класів ООМП. Друга – використання ідей реалізованих в rete-алгоритмі для зменшення кількості правил які можуть бути використані під час процесу отримання експертного рішення. Ефективність роботи rete-алгоритму забезпечує представлення наборів предикатів правил у вигляді спрямованого ациклічного графа (САГ) на кінці якого знаходяться елементи виводу нових фактів [5]. Така концепція алгоритму дозволяє швидко знаходити правила, які можна застосувати до набору вхідних фактів проходячи до них крізь мережу графу.

Хоча синтаксис ООМП і не дозволяє побудувати мережу з предикатів правил, можна побудувати мережу із самих правил. Якщо мережа правил буде представлена у формі САГ виникне можливість використання властивостей графу для знаходження необхідних для виконання правил. Наприклад, для цього може бути використаний алгоритм пошуку в глибину [6, с. 26]. Для реалізації такого підходу концепція ПС повинна обробляти особливу структуру правила написаного на ООМП. Правило повинно складатися з масиву вхідних фактів, набору предикатів та функції. Функція виводить новий факт із масиву вхідних фактів, якщо ті відповідають набору предикатів. Факт який виводить одне правило може бути використаний іншим правилом. Це дозволяє побудувати мережу правил пов'язаних між собою зв'язками вхідних та вихідних фактів. Факт який продукує правило не може бути використаний самим правилом. Така заборона дозволяє, для заданого набору правил,

побудувати САГ. При цьому кожне правило може бути використане тільки один раз.

Концепція роботи ПС виглядає наступним чином. При старті ПС завантажує масив правил з якого створюється САГ. Після цього ПС готова приймати запити представлені у вигляді масиву вхідних фактів. Обробка вхідних фактів та отримання експертного висновку здійснюється у два етапи. На першому етапі ПС використовує масив вхідних фактів для знаходження тільки тих правил, які можуть бути використані для їх обробки. Знайдені, для поточного набору класів вхідних фактів, правила зберігаються для повторного використання. Якщо до ПС вдруге надійде набір фактів представлений тим самим набором класів ООМП, вона використає збережений набір правил. На другому етапі ПС виконує обрані правила на наборі вхідних фактів та зберігає кінцеві факти в масив результатів. На схемі роботи ПС (рис. 2) показано процес виконання обраних правил. При цьому ПС оминає правила які не можуть бути виконані на вхідному наборі фактів.

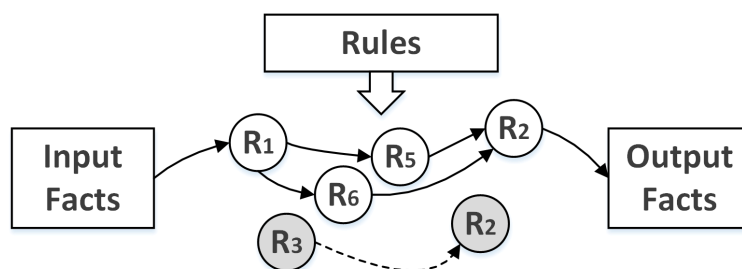


Рис. 2. Схема роботи продукційної системи

Таким чином в даному дослідженні запропонована концепція ПС, яка здатна використовувати правила зі складною бізнес-логікою та водночас обирати для виконання тільки ті правила які можуть бути застосовані виходячи із вхідного набору фактів. Експертні системи створені на основі запропонованої концепції зможуть швидше обробляти запити користувачів в складних, бізнес орієнтованих додатках.

Список використаних джерел:

1. Negnevitsky M. Artificial Intelligence: A Guide to Intelligent Systems. Second ed. // Harlow, 2005. – 415 p.
2. Forsyth R. The Architecture of Expert Systems. // Expert Systems: Principles & Case Studies. Ed. Forsyth R. / London, 1984. P. 9–17.
3. Джарратано Д., Райли Г. Экспертные системы: принципы разработки и программирование, 4-е издание. – Москва, 2007. – 1152 с.
4. Poblete R.E., Fuente, D.D., Alonso M. Using Cloud Computing with RETE Algorithms in a Platform as a Service (PaaS) for Business Systems Development. // Proceedings of the 2011 International Conference on Artificial Intelligence / Las Vegas, Vol. 2, 2011. P. 654–658.
5. How the Rete Algorithm Works? URL: <https://www.sparklinglogic.com/rete-algorithm-demystified-part-2/> (дата звернення: 10.05.2019).
6. Bang-Jensen J., Gutin G. Digraphs: Theory, Algorithms and Applications, Second ed. // London, 2008. – 795 p.