

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Запорізький національний технічний університет

МЕТОДИЧНІ ВКАЗІВКИ

до лабораторних робіт з дисципліни

«Сучасні методи та моделі інтелектуальних систем»
для магістрів спеціальності 123 «Комп'ютерна інженерія»
освітньої програми «Спеціалізовані комп'ютерні системи»
усіх форм навчання

Частина 1

2019

Методичні вказівки до лабораторних робіт з дисципліни «Сучасні методи та моделі інтелектуальних систем» для магістрів спеціальності 123 «Комп'ютерна інженерія» освітньої програми «Спеціалізовані комп'ютерні системи» усіх форм навчання. Частина 1 /Укл.: М.Ю. Тягунова, Т.В. Голуб. – Запоріжжя: ЗНТУ, 2019. – 13 с.

Укладачі: М.Ю. Тягунова, к.т.н., доцент
Т.В. Голуб, асистент

Рецензент: М.Б. Ільяшенко, к.т.н., доцент

Відповідальний
за випуск Т.В. Голуб, асистент

Затверджено
на засіданні кафедри КСМ
Протокол № 11 від 20.05.2019

Затверджено
на засіданні НМК КНТ
Протокол № 10 від 31.05.2019

ЗМІСТ

1	Лабораторна робота № 1 Розробка експертних систем	4
1.1	Загальні відомості	4
1.2	Структура експертних систем.....	5
1.3	Етапи створення експертних систем.....	7
1.3.1	Ідентифікація.....	7
1.3.2	Концептуалізація	8
1.3.3	Формалізація	8
1.3.4	Реалізація прототипної версії.....	9
1.3.5	Тестування.....	10
1.4	Завдання до лабораторної роботи	11
1.5	Контрольні питання	11
1.6	Вимоги до оформлення письмових звітів.....	11
	Перелік джерел посилання	12

1 ЛАБОРАТОРНА РОБОТА № 1 РОЗРОБКА ЕКСПЕРТНИХ СИСТЕМ

Мета роботи: навчитися розробляти експертні системи, дотримуючись всіх необхідних вимог.

1.1 Загальні відомості

Експертні системи – це напрям досліджень в галузі штучного інтелекту зі створення обчислювальних систем, що вміють приймати рішення, схожі з рішеннями експертів у заданій предметній області.

Як правило, експертні системи створюються для вирішення практичних завдань у деяких вузькоспеціалізованих областях, де велику роль грають знання «бувалих» фахівців. Експертні системи були першими розробками, які змогли звернути значну увагу до результатів досліджень в галузі штучного інтелекту.

Експертні системи мають одну суттєву відміну від інших систем штучного інтелекту: вони не призначені для вирішення якихось універсальних завдань, як, наприклад, нейронні мережі або генетичні алгоритми. Експертні системи призначені для якісного вирішення задач у певній області, в окремих випадках – областях.

Експертне знання – це поєднання теоретичного розуміння проблеми і практичних навичок її рішення, ефективність яких доведено в результаті практичної діяльності експертів у даній області. Фундаментом експертної системи будь-якого типу є база знань, яка складається на основі експертних знань фахівців. Правильно обраний експерт і вдала формалізація його знань дозволяє наділити експертну систему унікальними та цінними знаннями. Лікар, наприклад, добре діагностує хвороби і ефективно призначає лікування, не тому, що він має якісь вроджені здібності, а тому, що має якісну медичну освіту і великий досвід у лікуванні своїх пацієнтів. Тому цінність всієї експертної системи як закінченого продукту на 90% визначається якістю створеної бази знань.

Експертна система – це не проста програма, яка пишеться одним або кількома програмістами. Експертна система є плодом спільної роботи експертів у даній предметній області, інженерів по знанням і програмістів.

Але варто відзначити, що зустрічаються випадки, коли програми пишуться самими експертами в даній області.

Експерт надає необхідні знання про ретельно відібрані приклади проблем та шляхи їх вирішення. Наприклад, при створенні експертної системи діагностики захворювань лікар розповідає інженерові по знанням про відомі йому захворювання. Далі експерт розкриває список симптомів, які супроводжують кожне захворювання і на закінчення розповідає про відомі йому методи лікування. Інженер по знанням формалізує всю отриману інформацію у вигляді бази знань і допомагає програмісту в написанні експертної системи.

Зараз активний розвиток експертних систем дещо припинився і цьому є ряд причин:

- передача експертним системам «глибоких» знань про предметну область є значною проблемою. Як правило, це є наслідком складності формалізації евристичних знань експертів;

- експертні системи нездатні надати осмислені пояснення своїх міркувань, як це робить людина. Як правило, експертні системи лише описують послідовність кроків, які виконуються в процесі пошуку рішення;

- налагодження і тестування будь-якої комп'ютерної програми є досить трудомісткою справою, але перевіряти експертні системи особливо важко. Це є серйозною проблемою, оскільки експертні системи застосовуються в таких критичних областях, як управління повітряним і залізничним рухом, системами зброї і в ядерній промисловості;

- експертні системи мають ще один великий недолік: вони нездатні до самонавчання. Для того, щоб підтримувати експертні системи в актуальному стані необхідне постійне втручання в базу знань інженерів по знанням. Експертні системи, позбавлені підтримки з боку розробників, швидко втрачають свою актуальність.

1.2 Структура експертних систем

На рисунку 1.1 зображена узагальнена структура експертної системи.

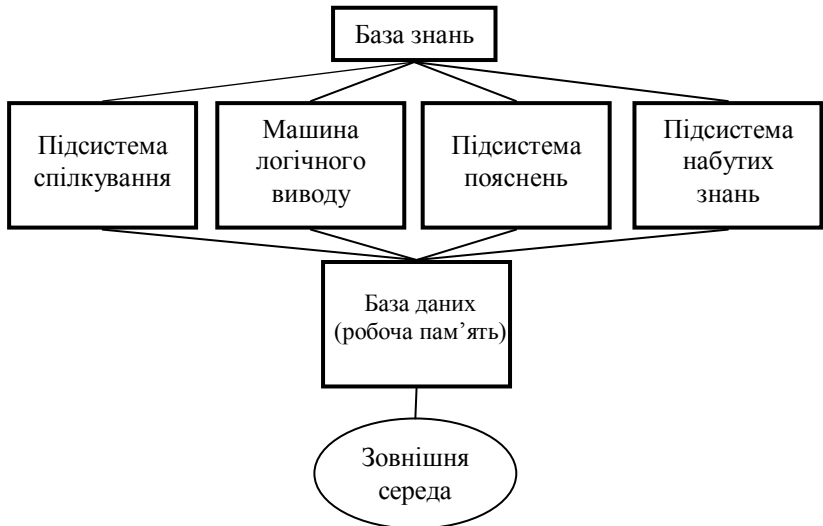


Рисунок 1.1 - Структура експертної системи

База знань призначена для зберігання експертних знань про предметну область, що використовуються при вирішенні завдань експертної системою.

База даних призначена для тимчасового зберігання фактів або гіпотез, що є проміжними рішеннями або результатом спілкування системи із зовнішнім середовищем, в якості якої зазвичай виступає людина, яка веде діалог з експертною системою.

Машина логічного висновку - механізм міркувань, що оперує знаннями і даними з метою отримання нових даних зі знань та інших даних, наявних в робочій пам'яті. Для цього зазвичай використовується програмно реалізований механізм дедуктивного логічного висновку (будь-який його різновид) або механізм пошуку рішення в мережі фреймів або семантичній мережі.

Підсистема спілкування служить для ведення діалогу з користувачем, в ході якого ЕС запитує у користувача необхідні факти для процесу міркування, а також, дає можливість користувачеві в якійсь мірі контролювати і коригувати хід міркувань експертної системи.

Підсистема пояснень необхідна для того, щоб дати можливість користувачу контролювати хід міркувань і, можливо, вчитися в

експертної системи. Якщо немає цієї підсистеми, експертна система виглядає для користувача як "річ в собі", рішенням якої можна або вірити, або ні. Нормальний користувач вибирає останнє, і така ЕС не має перспектив для використання.

Підсистема набутих знань служить для коригування та поповнення бази знань. У найпростішому випадку це - інтелектуальний редактор бази знань, в більш складних експертних системах - шлях здобуття знань з баз даних, неструктурованого тексту, графічної інформації та т.д.

1.3 Етапи створення експертних систем

У проектуванні експертних систем виділяють наступні етапи.

1.3.1 Ідентифікація

Цей етап складається з кількох внутрішніх етапів:

а) визначення учасників та їх ролей в процесі створення і експлуатації експертної системи.

У процесі створення експертної системи можуть брати участь наступні фахівці: інженери по знаннях, експерти, програмісти, керівник проекту, замовники (кінцеві користувачі). При реалізації порівняно простих експертних систем програмістів може не бути. Роль інженера по знаннях - видобування професійних знань з експертів і проектування бази знань експертної системи і її архітектури. Програміст необхідний при розробці спеціалізованого для даної експертної системи програмного забезпечення, коли відповідного стандартного (наприклад, оболонки для створення експертних систем) не існує або його можливостей недостатньо і потрібні додаткові модулі.

В процесі експлуатації можуть брати участь кінцеві користувачі, експерти, адміністратор.

б) ідентифікація проблеми.

На цьому етапі розробники повинні відповісти на ряд питань, що визначають особливості вирішуваних експертами, а отже, майбутньої експертної системою, завдань. Ці особливості визначають і особливості архітектури експертної системи, що формується на наступних етапах. До цих питань відносяться наступні:

- 1) який клас задач повинна вирішувати ЕС;
 - 2) як ці завдання можуть бути охарактеризовані або визначені;
 - 3) які можна виділити підзадачі;
 - 4) які вихідні дані повинні використовуватися для вирішення;
 - 5) які поняття і взаємозв'язки між ними використовуються при вирішенні задачі експертами;
 - 6) який вигляд має рішення і які концепції використовуються в ньому;
 - 7) які аспекти досвіду експерта істотні для вирішення завдання;
 - 8) яка природа і обсяг знань, необхідних для вирішення задачі;
 - 9) які перешкоди зустрічаються при вирішенні задачі;
 - 10) як ці перешкоди можуть впливати на рішення задачі.
- в) визначення необхідних ресурсів - тимчасових, людських, матеріальних
- г) визначення цілей

Цілями, що очікується досягнути при створенні експертних систем, можуть бути: підвищення швидкості прийняття рішення, підвищення якості рішень, тиражування досвіду експертів і т.п.

1.3.2 Концептуалізація

На цьому етапі розробники повинні відповісти на наступні питання:

- які типи даних потрібно використовувати;
- які дані задано та що повинно бути виведено;
- чи мають підзадачі найменування;
- чи мають стратегії найменування;
- чи є ясні часткові гіпотези, які широко використовуються.

1.3.3 Формалізація

Стадія формалізації передбачає переклад основних концепцій, підзадач і характеристик інформаційного потоку, виділених на етапі

концептуалізації, в більш формальне подання, яке ґрунтується на різних методах і схемах інженерії знань.

В процесі формалізації є три важливих факти: простір гіпотез, модель процесу і характеристики даних.

1.3.4 Реалізація прототипної версії

Стадія реалізації включає в себе переклад знань, формалізованих на попередній стадії, в схему уявлення, яка визначається обраною мовою. Як тільки представлені за цією схемою знання стають узгодженими і організованими так, щоб визначити потоки інформації і управління, вони стають робочою програмою.

Конструктор розвиває схему подання знань і використовує її для отримання прототипного варіанту ЕС. Предметно-орієнтовані знання, виявлені і сформульовані на етапі формалізації, визначають зміст структури даних, правил введення і стратегій управління. Схема подання визначає їх форму. Локальна несуперечливість використовуваних для вирішення елементів, яка була досягнута на попередніх стадіях, не гарантує працездатності програми так, як можуть бути глобальні невідповідності між структурами даних і будь-якими правилами або стратегіями управління.

Такі протиріччя повинні бути усунені, щоб забезпечити швидкий розвиток прототипного варіанту ЕС. Створення прототипного варіанту є виключно важливим кроком в процесі побудови ЕС. Окремі фрагменти цієї програми, що, в кінцевому рахунку, викидається, можуть бути збережені і використані в більш пізніх версіях. Однак головною метою діяльності на даному етапі стає перевірка відповідності формальної схеми основним використовуваним ідеям.

Прототипна база знань створюється за допомогою різних програмних засобів, наявних для обраної схеми подання знань (текстові редактори, інтелектуальні редактори, програми для придбання знань). Якщо існуючі засоби виявляються невідповідними, то інженер знань повинен створити новий засіб. Може навіть трапитися так, що потрібно розробити нову систему, призначену для побудови ЕС або нову мову.

1.3.5 Тестування

Стадія тестування передбачає перевірку прототипного варіанту системи і схем представлення знань, використаних для створення цього варіанту. Як тільки прототипний варіант може відпрацювати процес від початку до кінця на 2-3 прикладах, він повинен бути перевірений на багатьох прикладах для того, щоб виявити дефекти в базі знань і механізмі логічного висновку або дедуктивної машини.

Головними характеристиками введення-виведення є отримання даних і представлення результатів. Конкретний метод отримання даних може виявитися помилковим або невідповідним через те, що по ходу роботи задаються невірні питання або збирається недостатньо інформації. Наприклад, запитання можуть бути важкими для розуміння, двозначними або погано сформульованими для користувача системи. Сама система може бути незручною для роботи, в ній може бути відсутнім можливість для користувача перервати хід обчислень, щоб ввести дані відповідно до обраних порядку і форми.

Найбільш очевидне місце для пошуку помилок в міркуваннях - множина правил виведення. Вони рідко бувають незалежними один від одного. Крім усього іншого, правила можуть бути неправильними, суперечливими, неповними або відсутніми взагалі. Якщо посилання в правилі визначені невірно, воно може додаватися в неправильному контексті, порушуючи, таким чином, всю логіку. У таких випадках дуже часто не вдається помічати підвипадки. Висновок правила може бути неправильним часто за своїм змістом або за можливістю розрізнити підвипадки. Але навіть якщо посилання і висновки вірні, вони можуть бути пов'язані неправильними заходами приєднання. Так як правила не є незалежними, то має враховуватися їх вплив на інші правила.

Оцінки, ступені впевненості, ваги зв'язків грають важливу роль, і вони не можуть присвоюватися незалежно від того, як групи правил будуть використовуватися в контексті конкретних випадків.

Часто помилки в прототипній версії виникають через особливості використаних стратегій управління. Якщо система розглядає гіпотезу в порядку, відмінному від відповідного порядку, якого дотримується експерт, то конструктор повинен проаналізувати дедуктивну машину.

1.4 Завдання до лабораторної роботи

Розробити колективно всією групою під керівництвом викладача експертну систему, що дозволяє неспеціалістові в певній області зробити правильний вибір. Наприклад, розробити експертну систему, що дозволяє неспеціалістові автотранспортних засобів придбати бажаний автомобіль.

Результатом роботи є програмний продукт, який реалізує вказані функції. Програмний комплекс ЕС повинен містити наступні обов'язкові елементи:

- базу даних;
- базу знань;
- користувацький інтерфейс;
- інтерфейс експерта;
- механізм логічного висновку.

За необхідності можливе додавання інших елементів.

1.5 Контрольні питання

1. Що собою являють експертні системи?
2. Призначення експертних систем.
3. Що позначає термін «експертне знання»?
4. Переваги використання експертних систем.
5. Недоліки використання експертних систем.

1.6 Вимоги до оформлення письмових звітів

При оформленні кожного звіту необхідно дотримуватися наступних вимог. Звіт повинен містити:

1. Титульний аркуш із зазначенням теми, дисципліни, студента, який виконав звіт, викладача, який приймає звіт;
2. Результати виконання завдань лабораторної роботи;
3. Відповіді на контрольні питання;
4. Висновки.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Demuth H., Beale M. Neural Network Toolbox. For Use with MATLAB. The MathWorks Inc. 1992-2000.
2. Нейронные сети. STATISTICA Neural Networks. Москва. Горячая линия – Телеком. 2000.
3. Principe J.C., Euliano N.R., Lefebvre W.C. Neural and Adaptive Systems. Fundamentals Through Simulations. N. York. J. Wiley & Sons Inc. 2000.
4. Luo F-L., Unbehauen R. Applied Neural Networks for Signal Processing. Cam-bridge University Press. 1998.
5. Официальный сайт поддержки STATISTICA Neural Networks [Электронный ресурс] / Режим доступа: <http://www.statsoft.ru/home/textbook/default.htm>
6. Медведев В.С., Потемкин В.Г. Нейронные сети. Матлаб 6. М.: Диалог МИФИ, 2002. – 496с.
7. Иванников В., Ланнэ А.. Matlab для DSP. Нейронные сети: графический интерфейс пользователя [Электронный ресурс] / Режим доступа: <http://catalog.gaw.ru/index.php?page=document&id=1438>
8. Каллан Р. Основные концепции нейронных сетей. М.: Издательский дом «Вильямс», 2001. – 287с.
9. Уоссермен Ф. Нейрокомпьютерная техника: Теория и практика. – М.: Мир, 1992.
10. Круглов В.В., Борисов В.В. Искусственные нейронные сети. М.: Горячая линия – Телеком, 2001. – 382с.
11. Осовский С. Нейронные сети для обработки информации. М.: Финансы и статистика, 2002. – 344с.
12. Тарков М.С. Нейрокомпьютерные системы. М.: БИНОМ, 2006. –142с.
13. Нейронные сети. STATISTICA Neural Networks. М.: Горячая линия – Телеком, 2001. – 182с.
14. Зайцевский И.В., Свиридов А.П., Слесарев Д.А. Нейронные сети и их приложения. М.: МЭИ, 2002. – 95с.
15. Дьяконов В.П., Круглов В.В. MATLAB 6.5 SP1/7/7 SP1/7 SP2 Simulink 5/6. Инструменты искусственного интеллекта и биоинформатики. М.: СОЛОН-ПРЕСС, 2006. – 456 с.

16. Распределение заданий в компьютерной системе на базе нейронной сети / М.Ю. Юрич, Р.К. Кудерметов // Материалы Седьмой Международной конференции-семинара «Высокопроизводительные параллельные вычисления на кластерных системах», 26–30 нояб. 2007 г., Нижний Новгород. – 2007. – с. 347–353.

17. Yurich M. An Optimal Distribution of Tasks in Parallel Computing System / M. Yurich // Proceedings of the International Conference «Modern Problems of Radio Engineering, Telecommunications and Computer Science», 19–23 Febr. 2008, Lviv–Slavsko, Ukraine. – 2008. – pp.585–587.

18. Юрич М.Ю. Гибридный алгоритм распределения задач в вычислительной системе / Юрич М.Ю. // Сборник научных трудов Донецкого национального технического университета. Серия: Информатика, кибернетика и вычислительная техника. – 2010. – Выпуск 12 (65) – с. 66–72.

19. Скобцов Ю.О. Основы эволюционных вычислений. Учебное пособие. Д.: ДонНТУ, 2008. – 326 с.

20. Ту Дж., Гонсалес Р. Принципы распознавания образов, М. 1978.

21. Панченко Т.В. Генетические алгоритмы: учебно-методическое пособие / под ред. Ю.Ю. Тарасевича. – Астрахань: Издательский дом «Астраханский университет», 2007. – 87 с.

22. Экспертные системы [Электронный ресурс] // Портал искусственного интеллекта / Режим доступа: <http://www.aiportal.ru/articles/expert-systems/expert-systems.html>

23. Нейронные сети. *STATISTICA Neural Networks*: Методология и технологии современного анализа данных / Под редакцией В. П. Боровикова. - 2-е изд., перераб. и доп. - М.: Горячая линия - Телеком, 2008. - 392 е., ил.