

# ПРИКЛАДНІ ІНТЕЛЕКТУАЛЬНІ СИСТЕМИ ОБРОБКИ ЕКОНОМІЧНИХ ДАНИХ

## Самостійна робота за Змістовним модулем 2: *Експертні системи, як різновид систем штучного інтелекту*

### 1. Сутність експертного аналізу

Сучасні дослідження у сфері експертного аналізу сконцентровані на розробці та впровадженні програмних систем, в основу яких покладені методи штучного інтелекту. Для того, щоб програмна система мала можливості експерта, вона повинна відповідати наступним умовам:

- програмна система повинна володіти знаннями, тобто мати до них доступ та уміти їх використовувати;
- знання, якими володіє програмна система, повинні бути спрямовані на певну предметну область;
- на основі цих знань програмна система має бути здатна знаходити способи вирішення проблем;
- програмна система повинна мати змогу поповнювати та оновлювати знання.

Як зазначалось у попередньому розділі, не слід плутати знання з інформацією, і, тим більше, з даними (див. інформаційну піраміду). Наприклад, дані про певний обчислювальний комплекс, систематизовані в посібник з експлуатації, являють собою інформацію. Проте її наявність не дає можливість особі, яка не є фахівцем, відразу виправити ситуацію при збої в роботі комплексу. Лише після детального вивчення посібника і визначення відповідно до цієї проблеми логічних взаємозв'язків між окремими частинами одержаної інформації, можна сподіватися на усунення проблеми. У процесі її усунення відбувається набуття знань, і дилетант робить перший крок до перетворення в експерта.

Підсумок наведених міркувань дає змогу сформулювати визначення експертної системи в наступному вигляді. Експертна система -це програмна система, яка оперує знаннями в певній предметній області з метою вироблення рекомендацій для вирішення проблем. Експертна система може повністю взяти на себе функції, виконання яких вимагає залучення досвіду людини-фахівця, або виконувати роль асистента для людини, що приймає рішення. Людина-фахівець предметної області, що співпрацює з експертною системою, може добитися, з її допомогою, результатів найвищого гатунку. При цьому правильний розподіл функцій між інтелектом людини та штучним інтелектом є однією з ключових умов ефективності практичного використання експертних систем.

Перелік типових задач, для вирішення яких призначені експертні системи, включає:

- формування інформації з первинних даних (наприклад, із сигналів, що надходять від гідролокатора);
- діагностика несправностей (як у технічних системах, так і в людському організмі);
- структурний аналіз складних об'єктів (наприклад, хімічних з'єднань);
- вибір конфігурації складних багатокомпонентних систем (наприклад, розподілених комп'ютерних систем);
- планування послідовності виконання операцій, що призводять до заданої мети (у фінансуванні проектів різного рівня ризико-вості).

Найбільш відомі експертні системи та області їх практичного застосування наведені в додатку Б.

Слід зазначити, що для вирішення перерахованих задач можуть застосовуватися програмні системи, які необов'язково належать до класу експертних систем. Це можуть бути як традиційні прикладні системи, так і системи штучного інтелекту. Для того, щоб виділити експертні системи в окремий, чітко визначений клас програмних систем, необхідно окреслити набір ознак, які їм притаманні в тій чи іншій мірі. Ці ознаки визначаються в результаті аналізу відмінних характеристик експертних систем.

## **2. Характеристики експертних систем**

Експертна система відрізняється від програмних систем із чітко визначеним алгоритмічним спрямуванням завдяки наявності наступних ознак:

1. Експертна система моделює не стільки фізичну (або іншу) природу певної предметної області, скільки механізм мислення людини стосовно рішення задач у цій предметній області. Це істотно відрізняє експертні системи від систем математичного або імітаційного моделювання. Не можна стверджувати, що програма повністю відтворює психологічну модель фахівця певної області (експерта). Важливо те, що в ній основна увага приділяється відтворенню засобами комп'ютерної техніки способу вирішення проблем, що застосовується експертом, тобто реалізації міркувань так само, як це робить експерт.

2. Експертна система, крім виконання обчислювальних операцій, формує певні висновки, ґрунтуючись на тих знаннях, якими вона володіє. Знання в системі представлені, як правило, за допомогою спеціальної мови і зберігаються окремо від програмного коду, що формує висновки та міркування. Цей компонент програми прийнято називати базою знань.

3. Під час рішення задач експертною системою, в основному, використовуються евристичні методи, які, на відміну від алгоритмічних, не завжди гарантують успіх. Евристика, по своїй суті, є приблизним правилом, яке в програмному вигляді представляє знання експерта, набуте в міру накопичення практичного досвіду, внаслідок вирішення схожих проблем.

Експертні системи також відрізняються від інших видів систем штучного інтелекту. Ці відмінності полягають у наступному:

1. Експертні системи мають яскраво виражену практичну спрямованість у науковій або господарській діяльності. На відміну від них, інші програми з області штучного інтелекту є суто дослідницькими, і основна увага в них приділяється абстрактним математичним проблемам або спрощеним варіантам реальних проблем (іноді їх називають "іграшковими" проблемами).

2. По-друге, для експертної системи критичною характеристикою є її продуктивність, тобто швидкість отримання результату та рівень його достовірності. Дослідницькі програми штучного інтелекту можуть бути не дуже швидкими, і в них допускаються, в окремих ситуаціях, відмови. Натомість, експертна система повинна за прийнятний час знайти рішення, яке було б не гіршим за те, яке може запропонувати фахівець у відповідній предметній галузі.

3. Експертна система повинна володіти здатністю пояснити, чому запропоноване саме таке рішення, і довести його обґрунтованість. А дослідницькі програми штучного інтелекту надають результат тільки своєму творцю, який і без того (швидше за все) знає, на чому він ґрунтується. Експертна система проектується з розрахунку на взаємодію з різними користувачами, для яких її робота повинна бути, по можливості, прозорою.

Підсумовуючи наведені положення, можна зробити висновок, що експертна система містить знання в певній предметній області, накопичені в результаті практичної діяльності людини, і використовує їх для вирішення проблем, специфічних для цієї області. Цим експертні системи відрізняються від інших систем штучного інтелекту, в яких перевага надається більш загальним і менш пов'язаним з предметними областями теоретичним методам. Процес створення експертної системи часто називають інженерією знань і він розглядається як прикладне застосування методів штучного інтелекту.

### **3. Базові принципи функціонування експертних систем**

В основу експертних систем покладені принципи функціонування систем, заснованих на знаннях. До систем, заснованих на знаннях, належать системи, процес роботи яких базується на застосуванні правил відношень до символічного представлення знань.

Принципи роботи експертної системи, заснованої на знаннях, ілюструє схема, представлена на рис. 1. Згідно з цією схемою користувач передає в експертну систему факти або іншу інформацію і одержує у результаті поради у вигляді експертних знань.

За своєю структурою експертна система розподіляється на два основні компоненти: базу знань і машину логічного висновку. База знань містить знання, на підставі яких машина логічного висновку формує висновки. Ці висновки є відповідями експертної системи на запити користувача, що бажає одержати експертні знання.

Знання в експертній системі можуть бути представлені багатьма способами. Одним з широко вживаних методів представлення знань є правила у формі IF THEN. Його сутність полягає у тому, що в разі виявлення факту,

який узгоджується з шаблоном правила, повинна бути виконана обумовлена правилом дія. Наприклад, стосовно банківської діяльності з кредитування приватних осіб можна створити наступне правило:

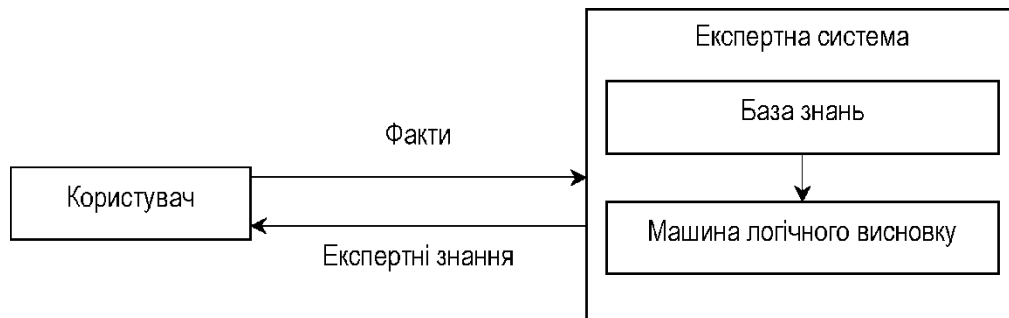


Рис. 1. Принципи роботи системи, заснованої на знаннях

IF людина не має постійного доходу THEN кредит не видавати

У даному прикладі шаблоном є фраза "людина не має постійної роботи". У випадку, коли факт відповідає шаблону - рекомендується виконати дію "кредит не видавати".

Застосовуючи підхід до представлення знань експертів у вигляді правил, було розроблено декілька успішних експертних систем. До них можна віднести систему XCON/R1, створену в компанії Digital Equipment Corporation (DEC), яка здатна проектувати конфігурації комп'ютерних систем, не поступаючись окремим експертам-людям.

Даний підхід виявився також прийнятним для створення значної кількості порівняно невеликих систем з метою вирішення спеціалізованих задач на основі декількох сотень правил. Такі системи не можуть функціонувати на рівні експерта, проте дозволяють застосовувати інтелектуальні технології для вирішення задач, які не вимагають великих обсягів знань.

На відміну від таких систем, класична експертна система втілює в собі недруковані знання, які повинні бути одержані від експерта за допомогою розширених інтерв'ю, що проводяться інженером по знаннях протягом тривалого періоду часу. Такий процес називається інженерією знань. Інженерія знань - це отримання знань від експерта-людини або з інших джерел з метою подальшого їх представлення в експертній системі.

Спочатку інженер по знанням ініціює діалог з експертом-людиною, щоб виявити знання експерта. Цей етап аналогічний етапу роботи, який виконує розробник традиційної програмної системи в ході обговорення вимог до системи з клієнтом, для якого створюється система. Потім інженер по знанням представляє знання в явному вигляді для внесення в базу знань. Після цього експерт проводить оцінку експертної системи і передає критичні зауваження інженеру по знанням. Такий процес повторюється знову і знову до тих пір, доки експерт не оцінить результати роботи системи як прийнятні.

Розглянутий процес створення експертної системи вважається на сьогоднішній день загальноновживаним. Найбільш трудомістким його аспектом

є циклічна задача перенесення знань експерта-людини в базу знань експертної системи. Насправді, ця проблема є настільки важливою, що одержала назву "вузького місця" у процесі набуття знань.

Ще одна проблема, з якою стикаються експертні системи, полягає в тому, що представлені в них експертні знання обмежуються областю знань. Областю знань вважаються знання експерта, що стосуються рішення конкретних задач. Наприклад, банківська експертна система, призначення якої спрямоване на надання рекомендацій щодо видачі кредитів, повинна володіти великим обсягом знань про фізичних та юридичних осіб, охочих отримати кредит. У цьому випадку областю знань є банківська діяльність, а самі знання складаються з відомостей про клієнтів та методів оцінки їх лояльності..

Незважаючи на вказані обмеження, експертні системи показали свою здатність успішно вирішувати практичні задачі, які неможливо було розв'язати за допомогою традиційного програмування, особливо в тих умовах, коли доводиться користуватися невизначеною або неповною інформацією. Це означає, що для ефективного використання будь-якої технології важливо знати, які саме переваги можна отримати в результаті.

#### **4. Переваги експертних систем**

Експертні системи мають наступну низку привабливих особливостей, які дають переваги у разі їхнього практичного використання:

1. Доступність. Для забезпечення доступу до експертних знань можуть застосовуватися тривіальні комп'ютерні апаратні засоби загального користування. У певному значенні цілком виправданим є твердження, що експертна система - це засіб масового виробництва експертних знань.

2. Незначні витрати. Вартість надання експертних знань з розрахунку на одного користувача при застосуванні експертної системи істотно знижується порівняно з експертом-людиною.

3. Зменшена небезпека. Експертні системи можуть використовуватися в такому навколишньому середовищі, яке може виявитися небезпечним для людини.

4. Постійність. Експертні знання об'єктивно нікуди не зникають. На відміну від експертів-людей, які можуть піти на пенсію, звільнитися з роботи або померти, знання експертної системи зберігаються протягом невизначено довгого часу.

5. Можливість отримання експертних знань із багатьох джерел. За допомогою експертних систем можуть бути зібрані знання багатьох експертів, які були залучені до роботи над вирішенням задачі. Рівень експертних знань, скомбінованих шляхом об'єднання досвідів декількох експертів, може перевищувати рівень знань окремо взятого експерта-людини.

6. Надійність. Застосування експертних систем дозволяє підвищити ступінь довіри до того чи іншого висновку шляхом надання додаткової точки зору експерта-посередника в разі непогодження висновків між декількома експертами-людьми. Зрозуміло, такий метод вибору рішень не може

використовуватися, якщо експертна система запрограмована одним з експертів, що бере участь у зіткненні думок.

7. Пояснення. Експертна система здатна детально пояснити свої міркування, які привели до певного висновку. На відміну від неї, експерт-людина може виявитися втомленою, не схильною до пояснень або нездатною робити це постійно. Можливість одержати пояснення сприяє підвищенню довіри до того, що системою було запропоноване правильне рішення.

8. Швидка реакція. У деяких випадках вирішення проблеми може вимагати швидких дій або реагування в реальному часі. Залежно від апаратного та програмного забезпечень, які використовує експертна система, вона реагуватиме швидше і буде більш готовою до роботи, ніж експерт-людина. Такі ситуації можуть потребувати більш швидкої реакції, ніж є у пересічної людини. У такому разі прийнятним варіантом стає застосування експертної системи, що здатна діяти в реальному часі.

9. Об'єктивний результат за будь-яких обставин. Ця властивість може виявитися дуже важливою в екстремальних ситуаціях, коли експерт-людина буде нездатна діяти з максимальною ефективністю через стрес або втому.

10. Можливість застосування як навчальної програми. Експертна система може діяти як інтелектуальна навчальна програма, демонструючи учням як приклади рішення задач та пояснюючи, на чому засновані міркування системи.

11. Можливість інтелектуального аналізу баз даних. Експертні системи можуть використовуватися для доступу до баз даних з метою проведення інтелектуального аналізу їх вмісту. Як приклад можна привести пошук прихованих закономірностей у даних.

Крім наведеного списку переваг, процес розробки експертної системи надає також непряму перевагу. Вона витікає із необхідності перетворення знань експертів-людей в явну форму для введення в комп'ютер. Оскільки знання унаслідок цих дій стають явно відомими, а не присутніми неявно в голові експерта, з'являється можливість перевіряти знання на правильність, несуперечливість та логіку. Після такої перевірки, можливо, виникне необхідність відкоригувати ці знання.

## **5. Методологія формалізації знань**

У другій половині ХХ століття очевидним став факт, що ключем до створення програмних вирішувачів задач, здатних функціонувати на рівні експерта-людини, є знання в певній предметній області. Тобто в ході вирішення задач експертом міркування, насправді, виконують незначну роль. Натомість, експерт покладається на обсяг знань, які сформовані на основі досвіду, набутого протягом значного проміжку часу. У випадку, якщо з'являється задача, яку експерт не може вирішити на основі свого досвіду, у нього виникає необхідність здійснити процес міркування, починаючи від загальних принципів та теоретичних положень.

Отже, здібності експерта з позиції техніки міркувань нічим не кращі в порівнянні зі звичною людиною, якій доводиться діяти в абсолютно незнайомій ситуації. Саме тому перші спроби створення універсальних програм, здатних вирішувати будь-які задачі в будь-якій предметній області, закінчились невдачею, оскільки в їх основу була покладена методологія, орієнтована виключно на процес міркування.

Сучасні експертні системи виробляють рішення, базуючись на формальному представленні знань. У методології формалізації знань експертної системи широко застосовується представлення в базі знань їх поверхневого обсягу. Основна причина такого підходу полягає у складності процесу представлення знань будь-якої предметної області, для якої він здійснюється вперше. У знаннях експерта можуть бути несумісності, двозначності, повтори або інші недоліки. Ці недоліки не стають очевидними до тих пір, доки не будуть зроблені спроби формально представити ці знання в експертній системі.

Поверхневі знання мають евристичний характер і, по суті, є емпіричними правилами, набутими на підставі досвіду, які можуть сприяти досягненню рішення з приблизною точністю. В той же час евристики здатні забезпечувати пошук оптимальних шляхів вирішення багатьох задач. Крім того, методологія формалізації знань, що призвела до успішної розробки експертних систем, дозволила створити інтелектуальні системи, які не обов'язково містять експертні знання.

Експертними знаннями вважаються спеціалізовані знання, відомі лише обмеженій кількості осіб. На відміну від них, звичайні знання це знання, які можна отримати із загальнодоступних інформаційних ресурсів (книг, веб-сторінок, періодичних видань і т.д.). Наприклад, до звичайних знань можна віднести знання про те, як вирішувати квадратні рівняння або здійснювати диференціювання.

У даний час широко застосовуються програмні системи, засновані на звичайних знаннях, які дозволяють виконувати математичні операції як з числовими, так і з символічними даними. Такі системи не можна віднести до класу експертних, оскільки методи, які лежать в їх основі, не належать до евристичних. Прикладом експертної системи може бути програма, що застосовує експертні знання для оцінки ризиків або інвестиційної привабливості проектів із комерціалізації наукових відкриттів.

Незважаючи на це, в наші дні терміни "система, заснована на знаннях" та "експертна система" часто сприймаються як синоніми. Фактично, останнім часом експертні системи стали розглядатися як підхід до програмування, альтернативний по відношенню до традиційного алгоритмічного програмування.

Вивчення проблеми формалізації знань започатковано в багатьох наукових дисциплінах. З практичної точки зору, особливо слід виділити область психології, яка присвячена дослідженню процесів сприйняття інформації людиною - когнітологію або науку про пізнання.

Програмні емулятори експертів призначаються для задач, що не мають задовільного алгоритмічного рішення. Для отримання їх прийняттого рішення використовуються логічні висновки. Ключовою вимогою до експертних систем є реальність висновків та недопущення їх двозначності.

Формування несуперечливих логічних висновків здійснюється на основі методів формальної логіки. Вони базуються на твердженні, що за наявності необхідної кількості істинних фактів висновок завжди повинен бути істинним. На відміну від них, неформальна логіка ґрунтується на методах переконання, що не базуються на фактах, а побудовані на непідтверджених відомостях, які допускають суперечливості.

В основу експертних систем покладені методи формальної логіки. Це визначає ключове значення способу представлення знань в експертних системах, оскільки від нього залежить повнота і точність аналізу фактів. Крім того, від правильного вибору способу такого представлення залежить весь хід розробки, а також ефективність, швидкодія та зручність супроводу системи.

З тематикою представлення знань тісно пов'язана не менш важлива тематика представлення даних, яка розглядається в проектуванні баз даних. Безумовно, бази даних, в основному, розглядаються як ре-позитарії поточних даних, а не знань. Тим не менш, вони можуть бути базовим матеріалом для аналізу прихованих закономірностей з метою формування знань.

Ще один важливий науковий підхід, який повинен бути втілений у технології представлення знань, полягає в семантичних розробках, що дозволяють систематизувати значення символів, що їх описують. Крім того, експертні системи повинні володіти здатністю розпізнавати знаки. Тобто вони як представники систем штучного інтелекту мають розуміти основні значення як простих - статичних знаків (наприклад, знаку Stop), так і знаків, що характеризують типові елементи поведінки (наприклад, людина почервоніла - хвилюється). Проведення досліджень у цьому напрямку є основою наукової області, що має назву семіотика.

## **6. Моделювання процесу рішення задач людиною**

Найбільш адекватна модель рішення задач людиною була запропонована Ньюеллом і Саймоном в 60-х роках минулого століття під час роботи над програмою GPS - однією із спроб створення універсального вирішувача задач. Незважаючи на те, що проект GPS не досяг поставлених перед ним цілей, дана модель до сьогодні вважається базовою при проектуванні експертних систем. Модель Ньюелла-Саймона складається з наступних компонентів: довготривала пам'ять, короткочасна пам'ять, когнітивний процесор. Розглянемо їх детально.

Довготривала пам'ять. Один із найбільш значущих результатів, отриманих Ньюеллом і Саймоном, полягає в доведенні твердження, що значну частину людського пізнання можна представити у вигляді правил форми IF-THEN. Правило являє собою невелику модульну форму знань, що відповідає певному фрагменту знань у предметній області. Фрагменти організуються в



довільній формі і забезпечуються зв'язками. Нижче наведений приклад правила, що представляє фрагмент знань:

IF людина має постійну роботу і володіє нерухомістю THEN можна видавати кредит

Ньюелл і Саймон популяризували використання правил для представлення людських знань і показали, як можуть бути виконані стандартні міркування за допомогою правил. Основуючись на цих досягненнях, психологи, що спеціалізуються у області когнітології, використовували правила як модель для пояснення процесу обробки інформації людиною.

Основна ідея полягає у тому, що мозок людини виділяє стимули із сенсорної вхідної інформації. Стимули активізують у довготривалій пам'яті відповідні правила, за допомогою яких формується відповідна реакція. Довготривала пам'ять виконує роль місця зберігання постійних знань людини. Наприклад, працівники банку засвоїли приблизно такі правила по відношенню до клієнтів:

IF клієнт подав заявку на закриття рахунку THEN банк втратив клієнта

IF клієнт зняв всі гроші з рахунку THEN банк може втратити клієнта

IF клієнт відкрив рахунок в іншому банку THEN банк ймовірно втратить клієнта

З прикладу видно, що останні два правила не виражені з остаточною визначеністю. У даний момент клієнту можуть знадобитися всі гроші, а надалі він знову використовуватиме свій рахунок. Аналогічним чином, відкриття ним в іншому банку рахунку ще не означає, що він не користуватиметься послугами даного банку. Стимули, що виникають, коли співробітник банку дізнається, що клієнт відкрив рахунок в іншому банку, зняв усі гроші з рахунку і подав заявку на закриття рахунку, можуть активізувати також інші правила подібного типу. Наприклад, банк-конкурент знизив тарифи за послуги.

Короткочасна пам'ять. Довготривала пам'ять людини може вміщувати багато правил, що мають просту структуру IF-THEN. На відміну від довготривалої пам'яті, короткочасна пам'ять використовується для тимчасового зберігання знань у період рішення задачі. Незважаючи на те, що довготривала пам'ять може зберігати сотні тисяч (або навіть більше) фрагментів, місткість короткочасної, або робочої пам'яті, значно менша - від чотирьох до семи фрагментів. Як простий приклад, що підтверджує цей факт, можна навести здатність людей одночасно бачити внутрішнім поглядом від чотирьох до семи цифр. Безумовно, люди здатні запам'ятовувати набагато більшу кількість цифр, але ці цифри, поряд із багатьма іншими відомостями, зберігаються в довготривалій пам'яті. Більше того, для забезпечення їх запису на постійне зберігання потрібен час.

Існує теорія, що короткочасна пам'ять здатна вміщувати тільки фрагменти, які можуть бути активними одночасно. Згідно з цією теорією процес розв'язання задачі людиною розглядається як перерозподіл таких активізованих фрагментів у мозку. Зрештою, фрагменти можуть бути активізовані таким чином, що буде вироблена свідомо думка. Наприклад,

працівник банку, що розмовляє з потенційним клієнтом, може відзначити про себе: "Він надав про себе суперечливі відомості - можливо, він шахрай".

Когнітивний процесор. Ще одним елементом, необхідним для вирішення задач людиною, є когнітивний процесор. Когнітивний процесор робить спроби знайти правила, які повинні бути активізовані відповідно до стимулів. У випадку, коли існує багато правил, які можуть бути активізовані одночасно, когнітивний процесор повинен виконати операцію вирішення конфліктів, щоб визначити, яке правило має найвищий пріоритет. Після цього має бути виконане це правило. Наприклад, активізовані два наступні правила:

IF клієнт банку хоче одержати кредит THEN скласти договір IF клієнт банку хоче одержати кредит THEN ознайомити його з умовами

У такому випадку необхідно спочатку виконати дію другого правила, а потім - першого.

У сучасних експертних системах когнітивному процесору відповідає машина логічного висновку. Важливим чинником при проектуванні експертної системи є обсяг знань або ступінь деталізації правил. У випадку, коли ступінь деталізації дуже малий, то розуміння окремого правила без вивчення інших правил стає ускладненим. Навпаки, якщо ступінь деталізації дуже великий, то виникають проблеми при необхідності модифікації експертної системи, оскільки в одному правилі можуть змішуватися декілька фрагментів знань.

## **7. Методологічні засади створення експертних систем**

Виходячи з попереднього матеріалу, стає очевидно, що методологія створення експертних систем відрізняється від методології створення алгоритмічних систем. Це цілком пояснює існування функціональної відмінності між мовами експертних систем та алгоритмічними мовами. Суть відмінності полягає у тому, що покладено в основу представлення.

Алгоритмічні мови програмування націлені на забезпечення підтримки гнучких та надійних методів представлення даних. Наприклад, вони дозволяють легко створювати, а також маніпулювати такими структурами даних, як: масиви, записи, зв'язані списки, стеки, черги, дерева. Сучасні мови, такі як Java і C++, розроблені для надання програмісту істотної допомоги в досягненні абстракції даних. Для цього передбачені структури, що забезпечують інкапсуляцію даних -об'єкти. Тим самим досягається необхідний рівень абстракції, який потім реалізується в програмі. Дані та методи маніпулювання ними тісно переплітаються в об'єктах.

Розробники мов експертних систем зосереджують свою увагу на підтримці гнучких і надійних способів представлення знань. Підхід до розробки експертних систем передбачає застосування двох рівнів абстракції - абстракції даних і абстракції знань. У мовах експертних систем спеціально передбачене відокремлення даних від методів маніпулювання ними. Прикладом подібного відокремлення є застосування фактів (абстракції даних)

і правил (абстракції знань) у мовах експертних систем, орієнтованих на застосування правил.

Цей приклад наглядно ілюструє аналогія з класичним виразом, який оприлюднив Дональд Кнут у своїй книзі з програмування алгоритмічних задач:

Програма = Структура даних + Алгоритми

Стосовно експертних систем цей вираз виглядає таким чином:

Експертна система = Знання + Логічний висновок

Сформульована відмінність у спрямованості між мовами двох типів пояснює різницю в методологіях програмування. З одного боку, в алгоритмічних мовах дані та знання тісно переплітаються, тому програмісти зобов'язані ретельно описувати послідовність виконання операторів програми. З іншого боку, у мовах експертних систем дані явно відокремлені від знань, чим нівелюється необхідність жорсткого контролю над послідовністю виконання коду програми. Як правило, для застосування знань відносно даних використовується повністю відокремлений компонент експертної системи - машина логічного висновку. Таке розділення знань і даних сприяє досягненню високого рівня роз-паралелювання та модульності обчислень.

Таким чином, у вигляді загального визначення мова програмування експертних систем представляє собою сукупність команд, записаних із застосуванням визначеного синтаксису та машини логічного висновку, призначеної для їх виконання. Залежно від реалізації, машина логічного висновку може забезпечувати прямий логічний висновок, зворотний логічний висновок або обидва варіанти розробки висновків.

Прямий логічний висновок є методом формування міркувань у напрямку від фактів до висновків, які виходять з цих фактів. Наприклад, якщо перед виходом із будинку виявиться, що йде дощ (факт), то ви повинні взяти із собою парасольку (висновок).

Зворотний логічний висновок передбачає формування міркувань у зворотному напрямі - від гіпотези (потенційного висновку, який повинен бути доведений) до фактів, які підтверджують гіпотезу. Наприклад, якщо не виглядати у вікно, але побачити, що хтось увійшов до будинку з вологими черевиками і парасолькою, то можна створити гіпотезу, що йде дощ. Щоб підтвердити цю гіпотезу, достатньо запитати цю людину, чи йде дощ. У разі позитивної відповіді буде доведено, що гіпотеза істинна, тому вона стає фактом. При застосуванні зворотного логічного висновку гіпотеза може розглядатися як факт, істинність якого викликає сумнів, і повинна бути встановлена. У такому разі підтвердження гіпотези (або її спростування) може інтерпретуватися як мета, що має бути досягнута.

Слід відмітити, що створювати експертні системи можна також мовами програмування, які за наведеним визначенням не належать до мов експертних систем, наприклад: C++, Паскаль і навіть асемблер.

Для розробки систем штучного інтелекту були створені спеціалізовані мови, орієнтовані на роботу зі знаннями: LISP, PROLOG, FRL, KRL, SMALLTALK, OPS5, PLANNER, QA4, MACSYMA, REDUCE, CLIPS.

Подальша їх еволюція призвела до можливості застосування ідеології об'єктно-орієнтованого підходу до розробки експертних систем.

Поєднання мови програмування та пов'язаних із нею допоміжних програм (утиліт), що дозволяють спростити розробку, налагодження та доставку користувачу програмних продуктів, утворює інструментальний засіб. Сучасні інструментальні засоби розробки експертних систем допускають використання в одній розробці різних підходів, включаючи прямий та зворотний логічні висновки.

Інструментальний засіб експертних систем спеціального призначення, що вимагає від розробника лише представлення бази знань, називається командним інтерпретатором. Командний інтерпретатор надає можливість повторного використання в експертних системах таких програмних компонентів, як машина логічного висновку та користувальницький інтерфейс. Технології командних інтерпретаторів позбавляють від необхідності створювати експертну систему для кожної предметної області, починаючи "з нуля".