

SISTEME EXPERT IN MFC

1. OBIECTUL DE STUDIU AL INTELIGENȚEI ARTIFICIALE

1.1. DEFINITII

Domeniul inteligenței artificiale

Este greu de dat o definiție precisă a domeniului inteligenței artificiale. În general, se consideră că obiectul său de studiu se referă la modalitățile prin care poate fi imitată inteligența umană cu ajutorul calculatoarelor electronice și a unor programe performante. Totuși, nici inteligența umană nu este o noțiune clarificată pe deplin. Există șapte tipuri de inteligență identificate de psihologul Howard Gardner

- *Inteligența lingvistică* – se referă la capacitatea și plăcerea de a citi, scrie, povesti sau rezolva cuvinte încrucișate
- *Inteligența logico-matematică* – presupune descoperirea modelelor, categoriilor și relațiilor și se manifestă de exemplu în rezolvarea problemelor aritmetice sau în jocurile de strategie
- *Inteligența spațială* – se referă la posibilitatea de a gândi în imagini și la ușurința rezolvării unor probleme geometrice spațiale; de exemplu găsirea drumului într-un labirint, talentul de a desena sau de a construi figuri din cuburi Lego
- *Inteligența corporal-kinestezică* – implică o mare sensibilitate în identificarea și prelucrarea senzațiilor fizice, de exemplu „a simți” ritmului unui dans
- *Inteligența muzicală* – presupune existența „urechii muzicale”, adică a posibilității de a percepe și distinge sunete care par la fel altor persoane
- *Inteligența interpersonală* – este dovedită de spiritul de conducător, de ușurința comunicării și de existența empatiei, adică a capacității de a înțelege sentimentele altora
- *Inteligența intrapersonală* – reflectă o bună cunoaștere a propriilor sentimente și posibilități

Testele de inteligență clasice cuprind de obicei întrebări referitoare la primele trei tipuri; pentru o evaluare cât mai completă și exactă a inteligenței unei persoane sunt necesare testări mult mai laborioase.

Ce este inteligența umană?

Rezumând, putem spune că, în general, inteligența umană este capacitatea de a învăța ușor și bine, de a sesiza ceea ce este esențial, de a rezolva situații și probleme noi pe baza experienței acumulate anterior

Ce este inteligența artificială?

IA este domeniul de studiu care își propune să explice și să modeleze comportamentul inteligent în termenii proceselor de calcul

IA nu este o știință pură sau o nouă disciplină inginerescă, ci are o clară natură de interdisciplinaritate, care implică știința calculatoarelor, matematică, psihologia proceselor cognitive etc.

Din punct de vedere ingineresc, IA se ocupă cu generarea reprezentărilor procedurilor care în mod automat și autonom permit rezolvarea problemelor rezolvate până acum numai de oameni. Obiectivul IA este abordarea inteligenței ca pe un calcul posibil de efectuat, fezabil.

1.2. TESTUL TURING

O definiție operațională a inteligenței este testul Turing.

Testul constă într-o conversație la distanță (de exemplu, pe chat) între un om și un calculator; la sfârșitul testului, calculatorul se consideră inteligent când omul nu poate spune dacă a dialogat cu un alt om sau cu o mașină.

Se înlocuiește astfel întrebarea „pot mașinile să gândească?” cu problema dacă „se pot comporta similar cu oamenii într-un context bine definit”.

Testul Turing

Din test se desprind câteva calități implicite ale unui sistem inteligent:

- prelucrarea limbajului natural;
- posibilitatea reprezentării cunoașterii, a informațiilor apriori și a celor căpătate în timpul dialogului;
- raționamentul automat, pentru utilizarea informațiilor în răspunsuri, concluzii sau pentru construirea de întrebări proprii;
- învățarea, pentru adaptarea la noile circumstanțe.

Definiții inteligența artificială

Patrick Winston

IA este studiul ideilor care permit calculatoarelor să efectueze lucrurile care fac oamenii să pară inteligenți. IA este studiul proceselor computaționale care fac posibile percepția, raționamentul și acțiunea.

Marvin Minsky

A inițiat ipoteza conform căreia inteligența nu este un singur lucru, ci o colecție de strategii, iar IA este știința identificării concrete a acestor strategii și a modului în care se încadrează într-un tot coerent

Elaine Rich

IA este domeniul care studiază cum se execută cu calculatoare electronice sarcini pentru care omul este astăzi cel mai bun

Definiție

În consecință, inteligența artificială poate fi descrisă drept acel domeniu al informaticii care se ocupă cu proiectarea și construirea sistemelor capabile să realizeze funcții ale intelectului uman, cum ar fi învățarea din experiență, înțelegerea limbajului natural sau utilizarea unui raționament pentru rezolvarea problemelor

Mai scurt, domeniul poate fi considerat partea computațională a capacității de a atinge scopuri.

1.3. INTELIGENȚA NATURALĂ ȘI ARTIFICIALĂ

Este evident faptul că este mai ușor de exprimat ce trebuie să facă mașinile inteligente decât descrierea a ceea ce trebuie să fie ele. Structura arhitecturală a calculatoarelor electronice rămâne încă foarte diferită de structura sistemelor biologice, și această diferență se regăsește și în tipurile de performanțe ale acestora.

Recunoașterea limbajului, de exemplu, nu este asociată neapărat cu inteligența unui om, de vreme ce marea majoritate a oamenilor sunt capabili să realizeze acest lucru, totuși această problemă este încă greu de realizat computațional. Reciproc, calculele aritmetice foarte complexe sunt rezolvate ușor de calculatoare, însă cu dificultate de către oameni.

Comportamentul inteligent

- *flexibilitatea* – disponibilitatea de adaptare la condiții noi
- *feed-back (reacția)* – posibilitatea de a compara rezultatele acțiunilor cu așteptările și apoi modificarea corespunzătoare a acțiunilor
- *memoria* – pentru înmagazinarea informațiilor în vederea utilizării ulterioare

2. ISTORICUL IA

Una din preocupările constante ale ființei umane a fost imaginarea unor modalități care să simplifice diverse aspecte ale vieții cotidiene. Invențiile ingineresti, de natură mecanică, și-au propus mereu maximizarea performanțelor și minimizarea efortului uman, însă prezența factorului uman a fost o condiție sine qua non a rezolvării oricărei probleme. Abia în secolul XX s-a pus cu adevărat problema automatizării unor procese și transformarea poziției omului din efectuant în supervisor.

Antichitatea

Urme ale preocupărilor pentru inteligență artificială se pot întâlni încă din antichitate. Primul sistem expert se consideră a fi un papirus egiptean, datând din mileniul III î.Hr., care consta în 48 de observații asupra diagnosticării și tratării rănilor la cap. Cunoștințele erau organizate sub formă de reguli: dacă există un anumit simptom, atunci se recomandă un anumit tratament. Trebuie menționat că această realizare se bazează pe o altă invenție extraordinară a omului, scrisul, prin intermediul căruia obiectele fizice ale lumii reale pot fi reprezentate și manipulate în mod simbolic.

În secolul IV î.Hr., Aristotel descoperă logica silogistică, primul sistem formal de raționament deductiv. Aceasta a avut o influență uriașă asupra dezvoltării științifice ulterioare a societății umane, deoarece noi cunoștințe puteau fi adăugate la fondul de cunoaștere existent prin demonstrarea de teoreme:

Sunt consacrate cele trei „Principii ale gândirii”:

- al identității (A este A)
- al noncontradicției (A nu poate fi simultan B și non-B)
- al terțului exclus (A este sau B sau non-B)

Trebuie menționat că acest model nu reflectă exact procesul raționamentului uman, fapt amendat de apariția noilor tipuri de logică din secolul XX, mai apropiate de deducția naturală, sau de „raționamentul bunului simț” (engl. “common sense reasoning”).

Renașterea

În secolul XVII, Blaise Pascal realizează un calculator mecanic, capabil să realizeze adunări și scăderi. Mașina a fost perfecționată apoi de Gottfried Wilhelm Leibnitz, astfel încât să realizeze înmulțiri și împărțiri. Calculatorul nu s-a bucurat de mare succes; Leibnitz a rămas cunoscut în special ca inventatorul analizei matematice, deși la vremea respectivă nu era la fel de celebru, fiind eclipsat de Newton.

Robotul

La începutul secolului XX, este menționată pentru prima dată o mașină capabilă să îndeplinească sarcini în locul omului. În 1920, scriitorul ceh Karel Čapek publică piesa de teatru R.U.R. (Rosumovy Univerzální Roboti – Roboții universali ai lui Rosum), în care apare termenul *robot*, adoptat pe plan internațional după traducerea în limba engleză.

În cehă, cuvântul înseamnă „muncitor, servitor” și a fost propus de fapt de fratele scriitorului, Josef.

John von Neumann

În 1928, John von Neumann (născut în Budapesta, cu numele János) propune *teorema minimax*, fundamentală în teoria jocurilor, care specifică faptul că orice joc de sumă zero cu doi jucători poate fi jucat rațional, cu o strategie optimă. Cea mai cunoscută realizare a sa este *arhitectura secvențială* pentru calculatoarele electronice.

Von Neumann a colaborat la dezvoltarea primelor calculatoare în Statele Unite – în 1945, la Universitatea Pennsylvania, a fost construit *ENIAC*.

Primele calculatoare

Primul calculator funcțional programabil, numit *Z3*, a fost produs în Germania, în 1941, de către Konrad Zuse.

Calculatorul a fost distrus în 1944 în timpul celui de-al doilea război mondial și a fost reconstruit mai târziu, ca piesă de muzeu.

În 1948, la Universitatea Manchester din Marea Britanie s-a realizat *Mark I*, primul calculator valorificat din punct de vedere comercial; acesta a fost și primul calculator care a scris scrisori de dragoste. Calculatoarele menționate se bazau pe tehnologia *tubului vidat* (a lămpii electronice).

Rețelele neuronale

În 1943, Warren McCulloch și Walter Pitts propun o arhitectură inteligentă bazată pe *rețele neuronale artificiale*, care încearcă să simuleze funcționarea creierului organismelor vii.

Ca element distinctiv de logica simbolică explicită, în această abordare cunoștințele sunt înmagazinate de conexiunile dintre neuroni. Rețelele neuronale sunt sisteme paralele, distribuite, cu o capacitate de a învăța

După șase ani, în 1949, Donald Hebb formulează o lege de bază pentru *învățarea neuronală*: conexiunile dintre neuronii care se activează simultan se întăresc, iar cele dintre neuronii care se activează în contratimp slăbesc

În 1958, Frank Rosenblatt propune un model practic al acestei paradigme, și anume *Perceptronul*.

2.1. NAȘTEREA „INTELIGENȚEI ARTIFICIALE”

În 1956 are loc conferința de la Dartmouth College, organizată de John McCarthy, pentru a dezbate modalitățile în care calculatoarele pot fi făcute să se comporte inteligent. Unii cercetători erau interesați de îmbunătățirea capacității de rezolvare a problemelor (de demonstrarea teoremelor, de jocuri, adică de activitățile care necesită un raționament).

Alții erau interesați de recunoașterea modelelor sau de prelucrarea limbajului natural. Aici s-a lansat termenul de *Inteligență Artificială*. Deși la vremea ei conferința nu a fost considerată un succes, în anii care au urmat au apărut tot mai multe laboratoare de cercetare a domeniului IA

Raționamentul simbolic

Tot în 1956, Alan Newell și Herbert Simon propun un program care simulează raționamentul uman, *General Problem Solver*, bazat pe logica predicativă. McCarthy inventează în 1959, la MIT AI Lab, *limbajul LISP*, destinat programării declarative. Acest limbaj, cu unele modificări, se folosește și azi, după aproape 50 de ani.

Istoria calculatoarelor

Din punct de vedere tehnologic în anii 1960, apare *tranzistorul*, un evident progres având în vedere că un tranzistor înlocuia cam 40 de lămpi electronice. În 1962 apar primii roboți industriali.

În 1965, se evidențiază proiectul *Dendral*, primul sistem expert destinat analizei chimice spectroscopice a structurii moleculare. Sistemul a fost coordonat de Edward Feigenbaum și laureatul premiului Nobel, geneticianul Joshua Lederberg.

Tot în 1965, Lotfi Zahed (născut în Azerbaidjan) pune bazele *logicii fuzzy*, care contrazice principiul terțului exclus; teoria a fost întâmpinată cu scepticism și chiar ostilitate, însă succesul aplicațiilor sale practice ulterioare a fost de necontestat. În 1966, Joseph Weizenbaum și Kenneth Colby crează programul *Eliza*, psihologul computerizat, care se va dovedi unul din cele mai celebre programe de IA create vreodată. El conversează în limbaj natural, analizează frazele interlocutorului pentru a extrage cuvinte-cheie și apoi afișează o replică, selectată dintr-o mulțime de variante predefinite.

În multe situații, conversația este fluentă, iar unii „pacienți” chiar s-au atașat emoțional de program, spre disperarea cercetătorilor.

Tot în perioada anilor '60 are loc un nou salt tehnologic, dând startul generației a treia de calculatoare, bazată pe *circuitele integrate* (sau chip-urile semiconductoare). De la inventarea lor, numărul de tranzistoare de pe un chip s-a dublat la fiecare doi ani.

În 1967 apare primul program de șah care învinge un jucător profesionist: *MacHack*, inventat de Richard Greenblatt. După un an, Ross Quillian propune reprezentarea cunoașterii prin *rețele semantice*. În 1969, lucrarea *Perceptronii* a lui Marvin Minsky determină întreruperea finanțărilor pentru studiul rețelelor neuronale. În studiu se arată limitele rețelelor neuronale.

În 1970, a luat ființă limbajul *Prolog*, ca rezultat al cercetărilor lui Alain Colmerauer, de la Universitatea din Marsilia, legate de formalizarea logică, ca bază a unui limbaj pentru prelucrarea limbajului natural.

Prolog este cel mai utilizat limbaj de programare logică și declarativă și este intens utilizat pentru dezvoltarea aplicațiilor de IA, fiind un instrument eficient de construire a sistemelor expert. În același an, Edward Shortliffe propune la Universitatea Stanford sistemul *Mycin*, unul din cele mai renumite sisteme expert

Acesta era construit pentru a sugera diagnostice medicale, însă mulți medici nu erau siguri de diagnosticul pus de calculator. De aceea, s-a adăugat modulul *Teiresias*, care, răspunzând la comanda „why” (de ce), putea justifica răspunsul prin afișarea regulilor urmate și a concluziilor

În 1973, John Holland publică o lucrare în care introduce noțiunea de *algorithm genetic*. Doi ani mai târziu, Marvin Minsky propune reprezentarea cunoașterii prin *cadre*.

Începutul anilor '70 aduce și o nouă etapă tehnologică, cea a *microprocesoarelor*, care determină apariția generației a patra de calculatoare, care continuă până în prezent. Așa-numitele circuite integrate monolitice conțin milioane de tranzistoare pe un singur chip.

În 1985, apare un nou limbaj de programare declarativ, *Clips*, propus de NASA. În 1986, Minsky, în lucrarea *Societatea minții*, descrie teoretic mintea umană ca o colecție de agenți cooperanți, care interacționează.

La începutul anilor '90, se dezvoltă paradigma rezolvării inteligente distribuite a problemelor, prin intermediul agenților. Rodney Brooks imaginează o arhitectură reactivă pentru agenți, care se constituie într-o alternativă la abordarea logică clasică. Tot Brooks coordonează la MIT Cog Project, care aduce un mare progres în tentativa de a construi un robot umanoid.

În 1991, Rao și Georgeff propun arhitectura BDI, în care starea agentului este definită de trei caracteristici: convingeri (engl. „beliefs”), dorințe (engl. „desires”) și intenții (engl. „intentions”):

- *Convingerile* reprezintă cunoștințele pe care le are agentul despre mediu;
- *Dorințele* corespund scopurilor agentului sau, altfel spus, task-urilor pe care trebuie să le îndeplinească acesta;
- *Intențiile* pot fi văzute ca o mulțime de planuri care, executate de obicei într-o ordine bine stabilită, conduc la îndeplinirea unui scop.

Modelul încearcă o adaptare în domeniul agenților a filosofiei raționamentului practic, care presupune mai întâi stabilirea scopului, printr-un proces deliberativ, și apoi evidențierea mijloacelor disponibile pentru atingerea aceluia scop.

Prezentul

În 1997, *Deep Blue* îl învinge pe Garry Kasparov, campionul mondial de șah la acea vreme. Tot în același an, are loc primul campionat de fotbal cu roboți. La sfârșitul anilor '90, tehnici de IA sunt folosite de programele de căutare pe Internet pentru extragerea de cunoștințe. Alte proiecte interesante, dezvoltate în această perioadă, sunt așa-numitele *Camera inteligentă* sau *Agenții emoționali*.

În 2000, se construiesc deja animale de casă robotizate (*robot-pets*) interactive, cu mare succes comercial, campionii în acest domeniu fiind firmele japoneze. Cynthia Breazeal

coordonează proiectul *Kismet*, care încearcă construirea unui robot cu o față care exprimă sentimente. Se înregistrează un progres și în sectorul mașinilor inteligente autonome. Robotul *Nomad* explorează regiuni îndepărtate din Antarctica în căutare de fragmente de meteoriți

Se dezvoltă roboți care să poată transmite imagini de pe suprafața planetei Marte și se cercetează posibilitățile de creștere a autonomiei vehiculelor spațiale.

Viitorul

Se preconizează că în jurul anilor 2030, calculatoarele vor egala din punctul de vedere al complexității și puterii de procesare capacitățile omului

Vis și temeri

- Case inteligente
- Roboți de familie conectați în intranet
- Automobile autonome
- Explorarea spațiului
- Educație personalizată
- Interfețe neuronale
- Omul – pe cale de dispariție?
- Nemurirea artificială
- Degradarea relațiilor sociale umane

3. DOMENII DE CERCETARE ȘI APLICAȚII

Raționamentul logic – se bazează pe teoria logico-matematică clasică și analizează modul în care noi cunoștințe pot fi derivate din altele existente în mod deductiv.

Reprezentarea cunoașterii – este un domeniu fundamental, deoarece studiază modalitățile în care cunoștințele din lumea reală pot fi exprimate și simbolizate în vederea manipulării computaționale.

Rețelele neuronale – se bazează pe analogiile cu modul de organizare a sistemului nervos uman pentru înmagazinarea implicită a cunoștințelor și pentru realizarea unor procese de învățare și generalizare în condiții de informații incomplete sau afectate de zgomote.

Teoria jocurilor – studiază strategiile pe care trebuie să le urmeze într-o confruntare cu reguli bine definite participanți ale căror interese nu coincid.

Percepția – cercetează reprezentarea într-o formă accesibilă computațional a stimulilor care provoacă reacții la nivelul organelor senzoriale umane; cele mai studiate sunt percepția vizuală și cea auditivă.

Algoritmii genetici – studiază modul în care soluția unei probleme poate fi găsită prin mecanisme inspirate din procesele de evoluție biologică.

Învățarea automată – se ocupă de studiul procedurilor care determină învățarea prin interacțiunea continuă cu mediul de execuție.

Agenții – cercetează comportamentul unor entități autonome (programe, roboți etc.) distribuite logic sau spațial și în interacțiune.

Ramuri de aplicație

- *Rezolvarea generală a problemelor*, care caută soluții bazate în special pe raționament logic.
- *Sisteme expert*, care înmagazinează cunoștințe specializate, provenite de la experți umani.
- *Prelucrarea limbajului natural*
- *Recunoașterea textului (OCR) sau vocii*
- *Robotică*
- *Educație*, învățarea asistată de calculator
- *Ingineria programării*, programe de tip „wizard”, generatoare de cod, pe baza opțiunilor programatorilor

4. REPREZENTAREA CUNOAȘTERII

4.1. PROBLEMATICA REPREZENTĂRII SIMBOLICE

Filosofia analitică presupune că există o lume reală „obiectivă”, independentă de orice observator iar simbolurile sunt entități care pot ține locul obiectelor din lumea reală, într-un sens abstract și absolut.

Diferențele dintre observatori sunt de cele mai multe ori ignorate, deoarece se presupune că aceștia împărtășesc un set comun de semnificații ale simbolurilor. Este cazul, în special, al matematicii și limbajelor formale. În domeniul științelor sociale, diferențele culturale de reprezentare simbolică și interpretare nu mai pot fi ignorate

Acțiuni și simboluri diferite pot avea aceeași semnificație pentru observatori din culturi diferite, pe când aceleași acțiuni și simboluri pot avea semnificații diferite. De exemplu, în China lui Mao Tse-tung, culoarea roșie a semaforului însemna „trecere permisă”. Informatica și, mai ales, inteligența artificială subscriu ideii că descrierea simbolică a realității este posibilă, întrucât aceasta este o modalitate convenabilă de prelucrare a fenomenelor reale complexe

Definiții

Reprezentarea este o schemă sau modalitate folosită pentru a captura elementele esențiale ale domeniului unei probleme.

O *reprezentare manipulabilă* este o reprezentare care facilitează prelucrarea computațională.

Reprezentarea cunoștințelor într-un calculator constă în găsirea și realizarea unei corespondențe între lumea reală și un sistem simbolic ce permite efectuarea raționamentelor.

Fapte

Reprezentarea cunoașterii presupune pe de o parte existența unor *fapte*, adică adevăruri într-o anumită situație sau univers de discurs relevant și pe de altă parte *reprezentarea faptelor* într-un formalism ales. Mai exact, faptele sunt lucrurile pe care vrem să le reprezentăm, iar reprezentările lor sunt lucrurile pe care le vom putea manipula efectiv.

Modelarea lumii reale

Din observațiile făcute asupra lumii reale se identifică acele caracteristici cărora li se pot asocia semnificații determinante în cadrul modelului. Descrierea caracteristicilor are ca scop deosebirea imaginii obiectului dat de imaginile celorlalte obiecte ale lumii înconjurătoare, la fel cum obiectul fizic se deosebește de restul obiectelor lumii reale. Descrierea reflectă cunoașterea despre obiectul respectiv.

Cunoașterea trebuie raportată întotdeauna la un „subiect cunoscător”, un agent care are capacitatea de interpretare a unei informații, pe care o poate avea la dispoziție direct sau prin extragerea din memorie. *Abstractizarea*: reprezentarea numai a informațiilor necesare unui anumit scop, element esențial pentru gestionarea complexității. *Raportul dintre expresivitate și eficiență*: unele reprezentări expresive sunt ineficiente, de aceea, pentru a crește eficiența trebuie sacrificată expresivitatea, fără a limita capacitatea reprezentării de a surprinde esențialul.

Expresivitate și eficiență

De exemplu, o reprezentare expresivă a numerelor neîntregi este cea sub formă de fracții sau radicali. Din punct de vedere computațional, aceasta nu este însă eficientă, așa că se preferă reprezentarea în virgulă mobilă (memorându-se cifrele semnificative și poziția lor față de virgulă). În acest caz, nu se poate reprezenta exact orice număr, însă se poate aproxima cu o precizie dorită, ceea ce este suficient în majoritatea cazurilor.

Piese de cunoaștere

În vederea utilizării prin program, cunoașterea este memorată sub forma unor *piese de cunoaștere*, ce descriu:

- fapte, fenomene, procese, evenimente
- obiecte cu însușiri dintr-o parte a lumii reale ce constituie domeniul de competență al programului

Pieșele de cunoaștere alcătuiesc un model al lumii, la care programul are acces prin intermediul unor proceduri specifice:

- organizare
- clasificare
- căutare
- recunoaștere

Componentele modelului unui sistem

Pentru a facilita raționamentul bazat pe model, modelul unui sistem trebuie să includă:

- *Concepte*: subdiviziuni logice ale sistemului în subsisteme.
- *Entități*: componente individuale (sau obiecte), care pot fi la rândul lor descompuse în *primitive* (elemente atomice), care formează baza elementară a modelului.
- *Proprietăți, attribute, valori* ale entităților sau conceptelor (de exemplu: culoarea unui obiecte, funcția unui subsistem); unele sunt de tip cantitativ („vârstă”, „înălțime”), altele de tip calitativ („inteligent”, „frumos”).
- *Constrângeri și asociații*: includ relațiile și interacțiunile cauzale dintre entități sau concepte (de exemplu: „conținut în”, „determină” etc.).

Sistem cognitiv

Totalitatea pieselor de cunoaștere, a modului de stocare și a procedurilor de acces la acestea definesc un *sistem cognitiv*.

Activitatea de reprezentare a cunoașterii poate fi gândită ca fiind bazată pe trei componente majore:

- stabilirea sistemului de meta-reprezentare
- stabilirea sistemului de clasificare
- stabilirea sistemului de organizare

Sistemul de meta-reprezentare

Are ca obiectiv asigurarea unor modalități adecvate de reprezentare a cunoașterii despre modelul lumii reale, cât și pentru componentele proprii ale sistemului inteligent. Prin meta-reprezentare se înțelege descrierea cunoștințelor despre cunoașterea însăși. Astfel, domeniul de competență al meta-reprezentării este cunoașterea abstractă.

Principalele piese de meta-cunoaștere se referă la:

- reprezentarea obiectelor
- reprezentarea funcțiilor
- reprezentarea relațiilor
- reprezentarea regulilor de inferență
- reprezentarea strategiilor de raționament
- auto-reprezentare („imaginea” sistemului inteligent despre el însuși)

Tipuri de reprezentări

În funcție de modul în care se dispune de informația necesară pentru reprezentarea esențialului unei piese de cunoaștere, distingem mai multe tipuri de reprezentări:

- *Reprezentarea descriptivă;*
- *Reprezentarea comparativă;*
- *Reprezentarea constructivă (generativă);*
- *Reprezentarea enumerativă.*

Reprezentarea descriptivă

Se realizează prin caracterizarea directă a componentelor, structurii și proprietăților piesei de cunoaștere.

Reprezentarea comparativă

Pentru a se reprezenta o piesă de cunoaștere *A*, expresia *B* conține o referință la o piesă de cunoaștere auxiliară, denumită *prototip*, și o informație ce marchează diferențele dintre *A* și piesa prototip:.

Reprezentarea constructivă (generativă) și enumerativă

Reprezentarea constructivă (generativă) presupune o relație generatoare a elementelor componente, structurii și proprietăților piesei de cunoaștere. *Reprezentarea enumerativă* presupune referiri la elementele unei mulțimi, care este definită ca piesă de cunoaștere.

Combinarea modalităților de reprezentare

O reprezentare completă se poate obține doar printr-o combinație finită a metodelor de reprezentare, deoarece cunoașterea despre un obiect este dată parțial în termeni descriptivi, parțial în termeni comparativi și parțial în termeni generativi sau enumerativi

Sistemul de clasificare

Urmărește gruparea în clase, precum și ordonarea acestora după criterii ce rezultă din analiza relevanței proprietăților folosite pentru caracterizarea obiectelor ce formează piesele de cunoaștere

Se dorește stabilirea unui număr finit de clasificări ale pieselor de cunoaștere, pentru a realiza ordinea conceptuală

Clasificarea presupune descoperirea unor grupări numite *clase* (sau *categorii*), pe baza unor asemănări între elementele aceleiași clase

Prin criteriu de clasificare se înțelege stabilirea categoriilor logice, a proprietăților ce determină pe mulțimea pieselor de cunoaștere o împărțire în clase

4.2. ARHITECTURA BAZEI DE CUNOȘTINȚE

Reprezentarea cunoașterii urmărește descrierea sub formă simbolică a universului în care sistemul efectuează raționamente. Interesează în special cunoștințele despre entități (reprezentări ale obiectelor) și despre relațiile dintre ele. O arhitectură de descriere a bazei de cunoștințe poate fi construită pe trei niveluri, în funcție de locul în care se prelucrează cunoștințele

Nivelul intern

Nivelul intern descrie structura de stocare fizică a cunoștințelor în baza de cunoștințe. La acest nivel se descriu detaliile complete ale stocării, precum și modul de acces la cunoștințe.

Nivelul conceptual

Nivelul conceptual descrie structura întregii baze de cunoștințe pentru o comunitate de utilizatori.

La acest nivel se face o descriere completă a bazei de cunoștințe, ascunzându-se detaliile legate de stocarea fizică, concentrându-se asupra descrierii entităților, tipurilor de date, a relațiilor dintre ele și a restricțiilor asociate.

Nivelul extern

Nivelul extern descrie baza de cunoștințe din perspectiva diferiților utilizatori, care au fiecare propriile interese. La acest nivel există tendința să li se ascundă utilizatorilor detaliile de care nu sunt interesați.

Observații

În multe sisteme nu există o distincție netă între cele trei niveluri. Totuși, în majoritatea sistemelor cognitive se remarcă un nivel conceptual puternic, care suplinește de multe ori celelalte niveluri. De asemenea, în aplicațiile practice apare o contopire între nivelurile conceptual și extern.

Reprezentări procedurale și declarative

Din punctul de vedere al programatorului, reprezentarea cunoașterii poate fi *declarativă* (știm *ce*) sau *procedurală* (știm *cum*). Reprezentările declarative sunt mai costisitoare, ocupă mai mult spațiu, pot avea elemente redundante, însă ștergerea sau adăugarea de elemente se face foarte simplu (aceste operații sunt de fapt actualizări ale unor *liste de fapte*). Reprezentările procedurale sunt mai compacte, dar mai puțin flexibile, întrucât se bazează pe o *mulțime de acțiuni* (de obicei de forma „dacă-atunci”).

Ierarhia cunoașterii

În funcție de nivelul de complexitate semantică, cunoașterea poate fi structurată pe mai multe niveluri, sub forma unei piramide (Ackoff, 1988).

Semnale

La baza piramidei stau *semnalele* sau *zgomotele*. Acestea nu au semnificație simbolică, adică nu pot fi înțelese de către un subiect cunoscător. De exemplu, o scriere hieroglifică necunoscută poate fi confundată cu o serie de desene abstracte, incoerente.

Date

Pe nivelul inferior al ierarhiei sunt *datele*, fapte simple ale lumii reale, la cel mai de jos nivel de înțelegere. Datele pot fi definite ca secvențe de simboluri, care încep să aibă sens. Acestea au o valoare pur sintactică, adică pot fi descrise complet prin reprezentări structurate, formale; de aceea, datele sunt ușor de stocat și prelucrat cu ajutorul calculatorului.

Informații

Datele, acumulate într-un context cu semnificație, devin *informații*. Ele sunt abstracții informale, care nu pot fi formalizate printr-o teorie logico-matematică. Dacă spunem „Iașul e un oraș frumos”, această informație are o natură subiectivă, depinde ce înțelege fiecare om prin „frumos”.

Enunțul anterior este reprezentabil într-un calculator, însă sub formă de date; pentru a fi prelucrat computațional, „frumos” ar trebui cuantificat de exemplu pe o scară numerică, de la 1 la 10. Diferența fundamentală dintre date și informații este faptul că datele au un caracter pur sintactic, pe când informațiile dispun de caracteristici semantice.

Informații analitice

Uneori este utilă separarea și reguparea informațiilor după anumite criterii. Această analiză ajută la determinarea relațiilor, modelelor, tendințelor și excepțiilor, sub forma *informațiilor analitice*.

Cunoștințe

Cunoștințele reprezintă următorul nivel de înțelegere. Ele derivă din informații cărora li se aplică reguli. La acest nivel acționează tehnicile de *data mining*, care pot identifica tendințe ascunse sau modele nebanale. „Inducția de reguli” este o tehnică de data mining care furnizează rezultate de forma: „oamenii care cumpără lapte cumpără și cumpara si pâine în 80% din cazuri”.

Inteligenta

Cunoștințele, la care se adaugă judecăți și valori, precum și experiență și învățare, se transformă în inteligență. Pe acest nivel se realizează utilizarea cunoașterii acumulate. În exemplul anterior, prin analiza tendințelor de piață, se pot organiza strategii de vânzare cu anumite scopuri (de exemplu, creșterea vânzărilor pentru un produs într-o anumită perioadă).

Înțelepciune

Noțiunea de înțelepciune a fost mult studiată și din punct de vedere filosofic. De-a lungul timpului, filosofii au încercat să înțeleagă natura înțelepciunii și modul în care aceasta poate fi atinsă. Înșuși cuvântul „filosofie” înseamnă etimologic „dragoste de înțelepciune”.

Reprezentarea cunoașterii presupune studiul a 3 obiective:

- domeniul cunoașterii
- limbajul de reprezentare a cunoașterii
- mecanismele de inferență

Domeniul cunoașterii are în vedere ceea ce trebuie reprezentat din domeniul problemei de rezolvat. De exemplu, pentru domeniul economic se vor include informații despre:

- obiecte reale (agenți economici, indicatori, bunuri patrimoniale, evenimente, procese/fenomene,...) împreună cu proprietățile lor;
- Regulamentele, politici sau strategii ale întreprinderii;
- Planuri și programe focalizate spre anumite obiective;
- Relații implicate de către toate obiectele de mai sus
- Tranzacții între agenții economici sau dintre agenți economici și instituții
- Relații temporale de genul “ în timpul”, “inainte”,...

Limbajul de reprezentare a cunoașterii

- este un limbaj care ne spune cum se reprezintă cunoașterea și cu ce metode

Mecanismele de inferență

- se referă la prelucrările cele mai adecvate ale cunoașterii din domeniu în scopul obținerii unor piese de cunoaștere

Metode de reprezentare a cunoașterii

Clasificare

- Metode declarative
 - Se utilizează la reprezentarea faptelor și/sau aserțiunilor
 - Includ rețelele semantice, cadrele/obiectele structurate și logica predicatelor
- Metode procedurale
 - Se utilizează la reprezentarea acțiunilor sau procedurilor
 - Includ regulile de producție.

În dezvoltarea sistemelor expert de gestiune mai există și alte metode cum ar fi: scenarii, liste, tabele de decizie, rețele neuronale.

Metoda regulilor de producție

Metoda regulilor de producție este cea mai folosită metodă de reprezentare a cunoașterii în sistemele expert. Conform acestei metode, o regulă de producție are două părți: partea de IF și partea de THEN, care stabilesc o relație ce reprezintă scopul regulii.

Descrierea relației IF-THEN (dacă-atunci) presupune următoarele perechi de termeni:

<antecedent><consecință>

<condiție><acțiune>

<premisă><concluzie>

În locul relației if-then se mai poate utiliza și semnul →

Sintaxa unei reguli de producție este:

Dacă (If)

Condiție_1

Condiție_2

...

Atunci (Then)

Concluzie_1

Concluzie_2

Altfel (Else)

Concluzie_1'

Concluzie_2'

Unde Condiție_1, Condiție_2 sunt antecedentele sau premisa iar Concluzie_1, Concluzie_2 sunt acțiunea sau consecința. Concluzie_1' sau concluzie_2' sunt acțiuni sau concluzii în situația în care condițiile din partea de If nu sunt adevărate. Conectorii logice AND, OR, NOT se pot utiliza pentru toate condițiile și concluziile.

Există mai multe categorii de reguli:

- Reguli de cauzalitate;
- Reguli de recomandare;
- Reguli directivă;
- Reguli strategice;
- Reguli euristice;

Utilizarea factorilor de certitudine

Dacă un expert stabilește o asociere inexactă între premisele și concluziile unor raționamente intervine necesitatea utilizării regulilor cu factori de certitudine.

Exemplu:

```
If inflatia este mare then
    Aproape sigur ratele dobânzilor sunt mari
```

Dacă se utilizează o scală [0,1] atunci:

```
If inflatia este mare then
    Aproape sigur ratele dobânzilor sunt mari FC=0,9
```

FC poartă numele de factor de certitudine.

- 0 este imposibil
- 1 este sigur

4.3. METODE DE ACHIZIȚIE A CUNOAȘTERII

Cunoașterea poate fi extrasă/colectată de o echipă de lucru, utilizând metode specifice. În urma achiziției, aceasta este prelucrată și reprezentată în baza de cunoștințe. Extragerea cunoașterii se realizează prin metode manuale, metode semiautomate și metode automate.

Metode manuale

Multe din metodele manuale sunt în general structurate în jurul interviului. Ca o caracteristică generală, metodele manuale sunt lente, scumpe și uneori lipsite de acuratețe.

Interviul

Interviul este cea mai veche și adesea cea mai utilizată metodă de extragere/colectare a informației. Interviul se folosește pentru:

- Explorarea și identificarea relațiilor dintre concepte sau pentru a verifica o piesă de cunoaștere;
- Captarea unor piese de cunoaștere care există deja.

O problemă importantă în colectarea datelor o reprezintă validitatea acestora. Această validitate este investigată în legătură cu două aspecte:

- Corectitudine sub aspect logic;
- Înțelegerea identică de către specialiști diferiți.

Avantaje ale interviului

- Interviul oferă cele mai bune oportunități pentru evaluarea validității cunoașterii achiziționate;
- Metodă eficientă pentru extragerea cunoașterii referitoare la subiecte complexe;
- Experții, în general, agreează interviurile ca metodă de colectare a cunoașterii.

Interviul nestructurat

Interviul nestructurat se folosește atunci când cognoticianul (persoană care colectează cunoașterea de la experți) vrea să exploreze o anumită problemă. Întrebările la care răspunde expertul nu urmăresc un anumit scenariu ci sunt axate pe spontaneitatea cognoticianului.

Interviul structurat

Interviul structurat se utilizează pentru un anumit scop. Întrebările sunt standardizate și sunt de tipul:

- Întrebări cu opțiuni multiple;
- Întrebări cu două opțiuni;
- Întrebări cu ordonare.

Interviul semistrukturat

Interviul semistrukturat presupune elaborarea unor întrebări cu un anumit timp înainte și lăsarea la latitudinea expertului a exprimării răspunsurilor după libera sa voință.

Problemele organizării și desfășurării interviului

Pentru asigurarea succesului unui interviu se vor respecta următoarele recomandări:

- Organizarea unui mediu adecvat și a bunelor raporturi cu expertul;
- Adresarea întrebărilor;
- Ascultarea atentă a răspunsurilor;
- Evaluarea rezultatelor fiecărei sesiuni.

Metoda observației-în-timpul-lucrului

În cazul acestei metode, cognoscianul va observa zilnic comportamentul expertului în domeniu, la locul său de muncă. Eventual, se pot adresa și întrebări lămuritoare pentru anumite sarcini sau operații, mai subtile, ale expertului.

Metoda brainstorming-ului

În cazul acestei metode, atenția cognoscianului este îndreptată către mai mulți experți astfel încât cunoașterea poate fi colectată prin obținerea mai multor idei. Toate soluțiile propuse sunt considerate egale ca importanță. Fazele brainstorming-ului sunt:

- Inițierea sesiunii de brainstorming;
 - Cognoscianul explică obiectivele urmărite, rolul fiecărui participant;
- Prezentarea problemei care trebuie abordată;
 - Problema trebuie să fie din domeniul expertizei celor invitați la ședință;
- Generarea ideilor
 - Cognoscianul invită experții să-și anunțe ideile. Ideile sunt notate pe o tablă sau afișate cu un mijloc de proiecție;

Metoda analizei protocoalelor

Un protocol este o înregistrare sau o documentație pas cu pas despre comportamentul unui expert în timpul soluționării problemei. Expertul este întrebat despre realizarea acțiunilor sale și este rugat să povestească pas cu pas ce face și cum acționează.

În contrast cu metodele interactive de interviu, analiza protocolului implică o relație cu sens unic de la expert la cognoscian.

Metode semiatomate

Aceste metode sunt menite să crească productivitatea cognoscianilor și sau experților fiind grupate în două categorii:

- Cele care sprijină experții în construirea bazei de cunoștințe fără aportul cognoscienților;
- Cele care sprijină cognoscienții în executarea fazelor specifice achiziției cunoșterii într-o modalitate rapidă și eficientă.

Metode automate

În acest caz, rolul cognoscianului sau expertului este uneori chiar eliminat, achiziția cunoșterii făcându-se chiar de analist cu ajutorul unor instrumente automate.

Instrumente de achiziție a cunoșterii

Instrumentele inteligente menite să sprijine cognoscianul în munca sa sunt:

- Editoare și interfețe;
- Controloare;
- Editoare grafice;
- Ajutoare în achiziția cunoșterii.

Editoare și interfețe

Editoare și interfețele se referă la editoarele de texte și editoarele de baze de cunoștințe specializate în introducerea cunoșterii în baza de cunoștințe.

Controloarele de consistență

Sunt componente software care permit revizuirea bazei de cunoștințe.

Editoare grafice

Componente software utilizate pentru achiziția cunoșterii prin imagini.

Ajutoarele în achiziția cunoșterii

Componente software care asistă cognoscienții la crearea și modificarea regulilor din baza de cunoștințe.

4.4. RAȚIONAMENTUL

Metode de raționament

Căutarea

Căutarea este utilizată în sistemele expert pentru localizarea cunoașterii. Aceasta implică trecerea de la o piesă de cunoaștere la alta în toate direcțiile posibile. Totodată se efectuează raționamente pe baza pieselor de cunoaștere găsite.

O problemă se consideră rezolvată dacă din starea inițială se ajunge în starea finală sau starea scop care de fapt este soluția finală.

Fiecare stare intermediară a soluției reprezintă o soluție parțială iar trecerea de la o soluție la alta se face utilizând operatori. Strategia de control definește operatorii (procedurile) care determină cum se face trecerea de la o stare la alta.

Căutarea poate fi sistematică și euristică.

Căutarea sistematică

Căutarea sistematică este o metodă prin care se investighează spațiul stării. Această operațiune se continuă până când este localizată starea scop. Aceste proceduri de căutare exhaustivă sunt mari consumatoare de timp însă sunt mai eficiente decât cele de căutare aleatoare.

Căutarea euristică

Căutarea euristică se referă la informații având ca sursă experiența umană. Dacă într-un arbore de căutare există mai multe căi de a ajunge la soluția scop atunci euristica izvorâtă din experiență poate alege calea optimă pentru a ajunge la soluția scop.

Strategii de raționament

În conceperea sistemelor expert există trei strategii de raționament: deducția, inducția și abducția.

Deducția

Deducția este procesul de raționament prin care aplicând principiile generale la cazurile particulare se poate obține o concluzie particulară.

Inducția

Inducția este procesul de raționament care pleacă de la premise de ordin particular și ajunge la concluzii de ordin general.

Abducția

Abducția este cel de-al treilea tip de strategie inferențială. Este asemănătoare deducției dar spre deosebire de deducție aceasta nu este o formă naturală a inferenței.

Spre exemplu în regula:

***Dacă** nu are baterie*

***Atunci** mașina nu pornește*

Concluzia că dacă mașina nu pornește atunci în mod sigur nu are baterie este evident eronată. Utilizând abducția este posibil să ajungem la rezultate eronate. Abducția poate însă furniza rezultate posibile care pot fi luate în calcul.

Incertitudinea

Deseori sistemele expert sunt puse în situația să raționeze cu cunoaștere incompletă, incertă sau imprecisă. Incertitudinea este utilizată în sistemele expert utilizând factorii de certitudine. Aceștia măsoară gradul de încredere al cunoașterii implicate în sistemele expert. De exemplu afirmația: “La anul inflația va crește” cu factorul de certitudine 50% are semnificația că specialistul are încrederea de 50% în faptul că inflația va fi mai mare la anul.

Factorii de certitudine se pot schimba în timpul proceselor inferențiale astfel încât o nouă cunoaștere este adăugată. Factorii de certitudine se pot asocia atât premiselor (IF) cât și concluziilor (THEN) din reguli.

IF A (FC=0.7)

THEN B

IF C

THEN D (FC=0.3)

Logica fuzzy

Tehnica fuzzy reprezintă o cale de a lucra cu valori neambigue în contextul cunoașterii imprecise. Logica fuzzy își propune să definească termenii imprecizi ca “înalt”, “mare”, “mic”. Utilizând logica fuzzy se poate defini un asemenea termen prin asocierea unor limite cu ajutorul unor probabilități. În acest context, termenii de mai sus pot fi cuantificabili fără a mai fi ambigui.

5. SISTEME EXPERT

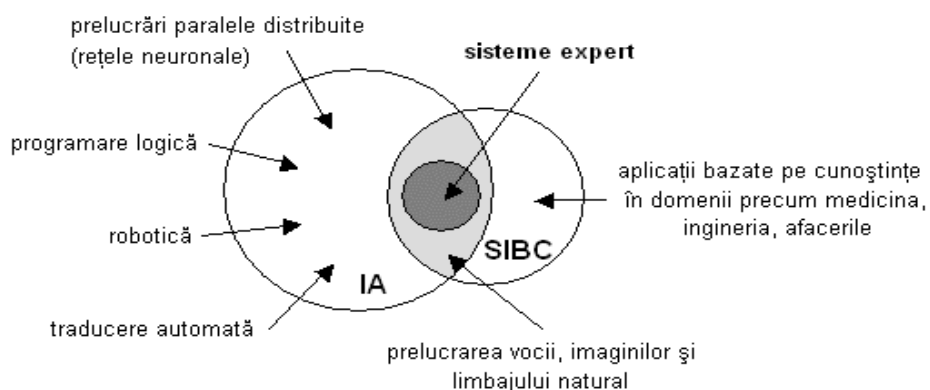
5.1. SISTEME INFORMATICE BAZATE PE CUNOSTINTE

Un *sistem informatic bazat pe cunoștințe* este un sistem care surprinde capacitatea unui expert uman de a rezolva probleme într-un domeniu specific prin reprezentarea cunoștințelor

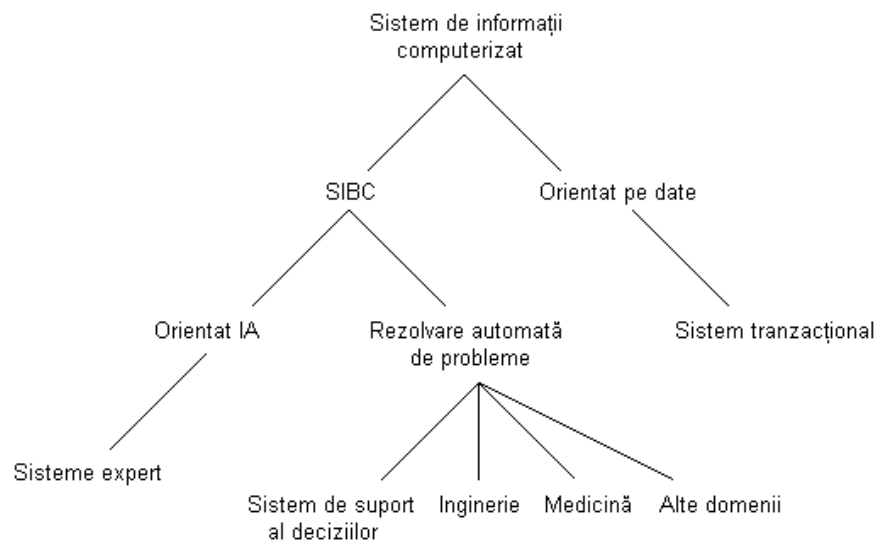
necesare într-un mediu computațional (într-un calculator). Știința care se ocupă cu proiectarea și dezvoltarea acestor sisteme se numește *ingineria cunoașterii*.

Întrucât pentru identificarea cunoștințelor necesare este nevoie de un expert uman, care nu este neapărat în contact direct cu nevoile și cerințele utilizatorilor, este necesară existența unui *inginer de cunoștințe*, care interviează experții.

Baza de cunoștințe rezultă astfel din interacțiunea dintre inginerul de cunoștințe și experții. Deși mulți cercetători nu fac diferența între sistemele informatice bazate pe cunoștințe și sistemele expert, primele pot fi considerate mai generale, deoarece nu se referă numai la domeniul inteligenței artificiale. Aplicații ale SIBC există și în medicină, inginerie, afaceri etc. De asemenea, unele ramuri ale inteligenței artificiale nu intră în categoria SIBC.



Clasificarea generală a unui sistem informatic bazat pe cunoștințe



5.2. SISTEME EXPERT

Un *sistem expert* este un program care utilizează cunoaștere și proceduri de inferență pentru a rezolva probleme suficient de dificile pentru a necesita în mod normal intervenția unui expert uman în vederea gășirea soluției

Pe scurt, sistemele expert sunt programe care înmagazinează cunoștințe specializate, introduse de experți.

Caracteristici ale SE

Aceste sisteme se folosesc deseori în situații în care nu există o soluție algoritmică clară. Principala caracteristică a acestora este prezența unei baze de cunoștințe împreună cu un algoritm de căutare adecvat tipului de raționament. De cele mai multe ori, baza de cunoștințe este foarte mare, de aceea este foarte importantă modalitatea de reprezentare a cunoașterii.

Baza de cunoștințe a sistemului trebuie separată de program, care la rândul său trebuie să fie cât mai stabil. Cel mai utilizat mod de reprezentare a cunoașterii este o mulțime de *reguli de producție*.

Operațiunile acestor sisteme sunt apoi controlate de o procedură simplă, a cărei natură depinde de natura cunoștințelor. Ca și în alte programe de inteligență artificială, când alte tehnici nu sunt disponibile, se apelează la căutare. Sistemele expert construite până în prezent diferă din acest punct de vedere. Se pune problema: pot fi scrise reguli atât de precise încât în orice situație să existe o singură soluție aplicabilă? Și, de asemenea, este necesară gășirea tuturor soluțiilor sau este suficientă doar una?

DENDRAL

Primul sistem expert a fost dezvoltat începând din 1965, la Stanford, de către Edward Feigenbaum și laureatul premiului Nobel, geneticianul Joshua Lederberg. *DENDRAL*, chimistul computerizat, este primul program bazat pe cunoștințe destinat raționamentului științific. El reușea să determine structura unor compuși chimici organici pe baza analizei spectroscopice a moleculelor. Pe măsură ce setul său de reguli a crescut, sistemul a devenit însă dificil de menținut și dezvoltat în continuare.

MYCIN

Un alt sistem expert celebru este *MYCIN*, creat în 1972 de Edward H. Shortliffe, tot la universitatea Stanford. În dizertația sa de doctorat, el a demonstrat puterea sistemelor bazate pe reguli pentru reprezentarea cunoașterii și inferențe în domeniul diagnosticului și tratamentului medical. Sistemul diagnostica infecții bacteriene ale sângelui și propunea tratamente corespunzătoare, cu dozajul antibioticelor calculat în funcție de greutatea pacientului.

MYCIN

MYCIN dobânda informații suplimentare prin întrebări adresate utilizatorului, precum:

- „A suferit recent arsuri pacientul?” sau
- „Are pacientul alergii cunoscute la medicamentul X?”

Se remarcă de asemenea tratarea cunoștințelor în condiții de incertitudine, spre deosebire de sistemul *DENDRAL*, care nu putea realiza acest lucru

Programul folosea reguli de tipul:

- „Dacă microorganismul este Gram pozitiv și morfologia microorganismului este Cocci și modul de organizare a microorganismului este lanț, atunci este destul de evident (cu probabilitatea 0,7) că microorganismul este un streptococ”

MYCIN + Teiresias

Când a fost construit *MYCIN*, mulți doctori nu aveau încredere în diagnosticele sale, deoarece nu puteau verifica dacă raționamentul pe care se baza concluzia era corect. De aceea, sistemului expert i-a fost adăugat un modul numit *Teiresias*, care putea răspunde la comanda „why” (de ce), listând regulile pe care s-a bazat procesul de decizie.

EMYCIN

Ca rezultat al experienței dobândite cu Mycin, Bill Van Melle a demonstrat posibilitatea de generalizare a reprezentării cunoașterii în programul său *EMYCIN* („Empty MYCIN”, 1979), care a devenit un model pentru multe „shell”-uri de sisteme expert comerciale (adică o structură de bază pe care se pot impune reguli noi).

Prospector

O legendă a sistemelor expert este *Prospector*, un program construit pentru identificarea formațiunilor geologice. Corelând date obținute din foraj asupra straturilor geologice parcurse, acestea trebuia să depisteze diverse depozite de minerale. El a prezis existența unui depozit de molibden în valoare de 150 de milioane de dolari, însă s-a dovedit mai apoi că depozitul fusese descoperit anterior iar *Prospector* a fost construit ca prin același raționament să ajungă la aceeași concluzie.

XCON

În 1980, *XCON* („eXpert CONfigurer”) a devenit primul sistem expert utilizat pe scară largă din punct de vedere comercial. Scopul său era de a asista utilizatorii de calculatoare VAX în configurarea acestora, adică încerca să determine ce schimbări erau necesare față de configurația standard de 50-100 de componente, astfel încât sistemul să fie complet și adecvat nevoilor utilizatorilor. Proiectul început de John McDermott la universitatea Carnegie Mellon s-a dezvoltat continuu, numărul de reguli crescând de la 750 (în 1979) la 5500 (în 1995).

Structura generală a unui SE

Un sistem expert trebuie să conțină obligatoriu următoarele module principale, care formează așa-numitul *sistem esențial*:

- Baza de cunoștințe
- Motorul de inferență

Baza de cunoștințe

Baza de cunoștințe este formată din ansamblul cunoștințelor specializate introduse de expertul uman. Cunoștințele stocate aici sunt în principal descrierile obiectelor și ale relațiilor dintre acestea. Baza de cunoștințe face parte din sistemul cognitiv, cunoașterea fiind memorată într-un spațiu special organizat.

Forma de stocare trebuie să asigure căutarea pieselor de cunoaștere specificate direct prin simboluri identificatoare sau indirect, prin proprietăți asociate sau inferențe care pornesc de la alte piese de cunoaștere.

Motorul de inferență

Motorul (sau mecanismul) de inferență reprezintă noutatea sistemelor expert. El preia din baza de cunoștințe faptele utilizate pentru construirea raționamentului. Mecanismul de inferență urmărește o serie de obiective majore, cum ar fi:

- alegerea strategiei de control în funcție de problema curentă;
- elaborarea planului de rezolvare a problemei după necesități;
- comutarea de la o strategie de control la alta;
- executarea acțiunilor prevăzute în planul de rezolvare.

Deși mecanismul de inferență este constituit dintr-un ansamblu de proceduri în sensul obișnuit al termenului, modul în care utilizează cunoștințele nu este prevăzut prin program, ci depinde de cunoștințele pe care le are la dispoziție.

Pe lângă acestea, un sistem expert mai conține o serie de module ce asigură comunicarea cu operatorul și expertul uman.

Modulul de comunicație

Modulul de comunicație asigură interfețele specifice pentru utilizatori cât și pentru achiziția de cunoștințe. Interfața cu utilizatorul este cea care permite dialogul între utilizator și sistem în limbaj cvasinatural, comunică mecanismului de inferență cererile utilizatorului și utilizatorului rezultatele prelucrărilor. Ea facilitează în egală măsură achiziția enunțului problemei inițiale și comunicarea rezultatului.

Modulul de achiziție a cunoștințelor

Modulul de achiziție a cunoștințelor preia cunoștințele specializate furnizate de expertul uman prin intermediul inginerului de cunoștințe într-o formă nespecifică reprezentării interne. O serie de cunoștințe pot apărea sub forma unor fișiere specifice bazelor de date sau altor programe externe. Acest modul recepționează cunoștințele, verifică validitatea lor și generează în final o bază de cunoștințe coerentă.

Modulul de explicare

Modulul de explicare permite trasarea drumului urmat în procesul de raționament de către sistemul rezolutiv și emiterea justificărilor pentru soluțiile obținute, evidențiindu-se astfel cauzele eventualelor greșeli sau motivul unui eșec. El ajută expertul să verifice consistența bazei de cunoștințe.

Explicare și actualizare

În funcție de aplicația pentru care este construit, structura efectivă a unui sistem expert poate fi diferită față de structura standard.

De exemplu, faptele inițiale pot fi achiziționate atât de la utilizator cât și de la un echipament de conducere automatizată. Este însă important ca sistemele expert să aibă două caracteristici:

- Să explice raționamentul. Dacă acest lucru nu este posibil, utilizatorii umani s-ar putea să nu-l accepte. În acest scop, trebuie să existe suficiente meta-cunoștințe pentru explicații iar programul trebuie să ruleze în pași inteligibili.
- Să poată dobândi noi cunoștințe și să-și poată modifica cunoștințele vechi. De obicei, singurul mod de a introduce cunoștințe într-un sistem expert e prin interacțiune cu un expert uman.

5.3. DEZVOLTAREA SISTEMELOR EXPERT

Ciclul de viață a sistemelor expert

Ciclul de viață al unui produs este perioada dintre momentul inițial de începere a activității și se termină cu retragerea din exploatare a produsului. Pentru programele convenționale s-au adoptat mai multe modele cum ar fi: Modelul în cascadă, modelul în spirală, modelul incremental și modelul liniar. Aceste modele funcționează și pentru sistemele expert.

Modelul clasic în cascadă este cel mai familiar programatorilor. În cadrul acestui model, fiecare fază se termină cu o verificare și o validare. Modelul incremental este o variantă a modelului în cascadă. Ideea de bază constă în dezvoltarea unui produs program prin creșterea incrementală a capacității sale funcționale.

Modelul în spirală constituie o altă cale de vizualizare a modelului incremental. Fiecare nou circuit adaugă o nouă capacitate funcțională sistemului expert.

Există autori care utilizează alte modele pentru ciclul de viață al unui sistem expert. Ciclul de viață al unui sistem expert se desfășoară pe 3 faze:

- Dezvoltare
- Integrare și utilizare
- Actualizare și întreținere

În faza de dezvoltare au loc următoarele activități:

- Analiza problemei (se determină structura de bază a sistemului);
- Achiziția cunoașterii (se colectează și se interpretează cunoașterea obținută de la expert);
- Reprezentarea cunoașterii (se determină metoda de reprezentare a cunoașterii după care se alege instrumentul de dezvoltare);
- Dezvoltarea prototipului;
- Dezvoltarea sistemului expert de livrat;
- Evaluarea și documentarea.

Procesul de dezvoltare a sistemelor expert nu este liniar deoarece unele faze pot fi executate simultan sau pot avea numeroase reveniri la fazele anterioare.

Inițializarea proiectului

Activitățile specifice în faza de inițializare sunt:

- Definirea problemelor și evaluarea necesităților

- Evaluarea soluțiilor alternative
- Verificarea adecvării problemei
- Considerarea cerințelor managementului

Definirea problemelor și evaluarea necesităților

Definirea și delimitarea clară a problemei simplifică mult lucrările din fiecare fază și ajută elaborarea unor sisteme expert funcționale și performante.

Evaluarea soluțiilor alternative

Înainte de a începe un proiect de dezvoltare a unui sistem expert trebuie luate în considerare toate celelalte soluții alternative de soluționare a problemei. Se poate constata de exemplu că nu este nevoie neapărat de un sistem expert. Problema poate fi rezolvată cu un SGBD sau cu programe de calcul tabelar.

Verificarea adecvării problemei

Simpla constatare că alte soluții alternative nu sunt potrivite pentru rezolvarea unei probleme nu înseamnă că un sistem expert este neapărat necesar.

Considerarea cerințelor managementului

Cerințele managementului au în vedere:

- Disponibilitatea resurselor financiare și alocarea lor de la bugetul întreprinderii;
- Disponibilitatea altor resurse;
- Restricții legale și de altă natură;
- Convingerea managementului de valoarea proiectului.

Analiza și proiectarea

Proiectarea conceptuală

Proiectarea conceptuală este o activitate prin care se obțin cât mai multe idei generale despre cum va arăta sistemul și cum va soluționa problemele. Din această activitate rezultă trăsăturile generale ale sistemului expert, interfețele cu alte sisteme, domeniul riscului, cerințele de resurse, structura echipei de proiectare. Echipa de proiectare este inițiată apoi în problematica proiectului de sistem și fiecare dintre participanți își începe activitatea.

Resursele de calcul

Resursele de calcul se referă la echipamentele și platformele în configurația hardware și software corespunzătoare cerințelor sistemului expert.

Studiu de fezabilitate

Studiu de fezabilitate pentru un sistem expert este similar cu cel elaborat pentru un sistem informatic oarecare.

Elementele studiului de fezabilitate sunt:

Fezabilitatea economico-financiară care se fundamentează pe:

- Costul dezvoltării sistemului
- Costul întreținerii
- Venituri anticipate
- Analiza cash-flow
- Analiza riscului

Fezabilitatea tehnică care se fundamentează pe:

- Cerințele de rețea;
- Cerințele de interfețe;
- Disponibilitatea cunoașterii și a datelor;
- Securitatea cunoașterii confidențiale;
- Metoda de reprezentare a cunoașterii;
- Disponibilitatea și compatibilitatea hardware-ului și software-ului;

Fezabilitatea operațională și impactul, care se fundamentează pe:

- Disponibilitatea resurselor umane și a celorlalte resurse;
- Prioritatea proiectului în raport cu celelalte;
- Evaluarea necesităților;
- Rezultatele organizării și implementării;
- Sprijinul managementului și utilizatorilor;
- Mediul utilizator.

Analiza cost-beneficiu

Realizarea sistemelor expert presupune investiții, investiții care se vor lua în seamă la calculul costurilor și vor fi raportate la beneficiile așteptate. Factorii de care trebuie ținut seama într-o asemenea analiză sunt:

- Factorul cost, pentru toate instrumentele utilizate la dezvoltarea sistemului expert (limbaje, generatoare, echipament suplimentar, etc...);
- Costul expertului;
- Costul cognoscianului și a programatorilor;
- Timpul necesar testării sistemului și instruirii personalului utilizator;
- Costul consultaților din exterior.

Prototipizarea rapidă

Un prototip este un sistem expert de dimensiune mică care poate fi dezvoltat într-un timp scurt. Prototipul are drept scop sprijinirea proiectanților în luarea deciziei cu privire la structura bazei de cunoștințe. Avantajele dezvoltării unui prototip sunt:

Permite formularea unei idei despre ceea ce este realizabil în aplicația completă;

- Oferă un mijloc de studiu pentru eficacitatea reprezentării cunoașterii și implementării bazei de cunoștințe;
- Înlătură erorile din sistemul final;
- Oferă o idee despre finalizarea proiectului;
- Accelerează procesul de achiziție a cunoașterii;

...

Dezvoltarea

Faza de dezvoltare începe odată cu terminarea prototipului demonstrativ și aprobarea sa de către management. În funcție de natura sistemului expert strategia poate îmbrăca următoarele forme:

- Continuarea prin prototipizare;
- Abordarea cu un ciclu de viață structurat;
- Ambele abordări.

Procesul de dezvoltare poate avea o durată mai mare sau mai mică în funcție de complexitatea sistemului.

Construirea bazei de cunoștințe

Construirea bazei de cunoștințe este o sarcină extrem de importantă a echipei de proiectare. Echipa de proiectare trebuie să cunoască intrările în sistem, cu alte cuvinte faptele inițiale, pentru a putea începe procesele inferențiale. De asemenea, echipa trebuie să elaboreze o schiță pentru aceste intrări și ieșiri un plan lucru foarte necesar în cazul sistemelor complexe.

După scrierea regulilor, acestea pot fi introduse imediat cu un generator de reguli, după care se pot face toate modificările necesare.

Testarea, evaluarea și îmbunătățirea

Prototipul și versiunile îmbunătățite se supun testării și evaluării performanței atât pentru cazuri ipotetice cât și pentru cazuri din domeniul real. Principiul general acceptat este de a testa sistemul pe cazuri istorice oferite de utilizator.

Metoda cea mai comună în evaluarea unui sistem expert constă în compararea performanței sale cu un criteriu general acceptat, de exemplu cu decizia unui manager expert sau cu decizia unui auditor expert.

Validarea se referă la determinarea faptului dacă sistemul a fost construit corect și oferă un nivel de acuratețe acceptabil. Verificarea confirmă dacă sistemul expert este construit în conformitate cu specificațiile de proiectare.

Implementarea

Implementarea este o etapă de durată care poate cuprinde următoarele faze:

Acceptarea de către utilizator

Din cauza psihologiei utilizatorilor pot apare probleme la implementare mai ales dacă în fazele precedente aceștia nu au fost consultați.

Instalarea, demonstrarea și utilizarea

Instalarea, demonstrarea și utilizarea presupune:

- Instalare după ce sistemul a fost testat în domeniu, iar soluțiile au fost acceptate;
- Instalare și funcționare în paralel cu expertul uman.

Credibilitatea sistemului expert crește odată cu demonstrațiile reușite, acceptate de către experți, pe cazuri propuse de experți și validate de către aceștia.

Instruirea și securitatea

Instruirea în utilizarea unui produs de tip sistem expert este o fază obligatorie în etapa de implementare și se va face conform unei planificări realizate de echipa de proiectare.

Sistemul expert este un produs software deci distribuirea acestor produse trebuie făcută în conformitate cu legislația de protecție a acestor tipuri de produse.

Documentarea, integrarea și testarea în domeniu

În faza de documentare echipa de proiectare pune la punct o documentație a produsului care constă din manuale imprimate, manuale on-line și alte documentații necesare utilizatorilor finali.

Testarea sistemului este de asemenea o fază deosebit de importantă care se aplică și se coordonează cu atenție. Această fază poate aduce noi elemente în baza de cunoștințe sau poate determina schimbări în structura bazei de cunoștințe.

Postimplementarea

Postimplementarea este o fază care urmează distribuirii sistemului expert. Activitățile cele mai importante sunt: exploatarea, operarea, întreținerea, actualizarea, extinderea și evaluarea sistemului.