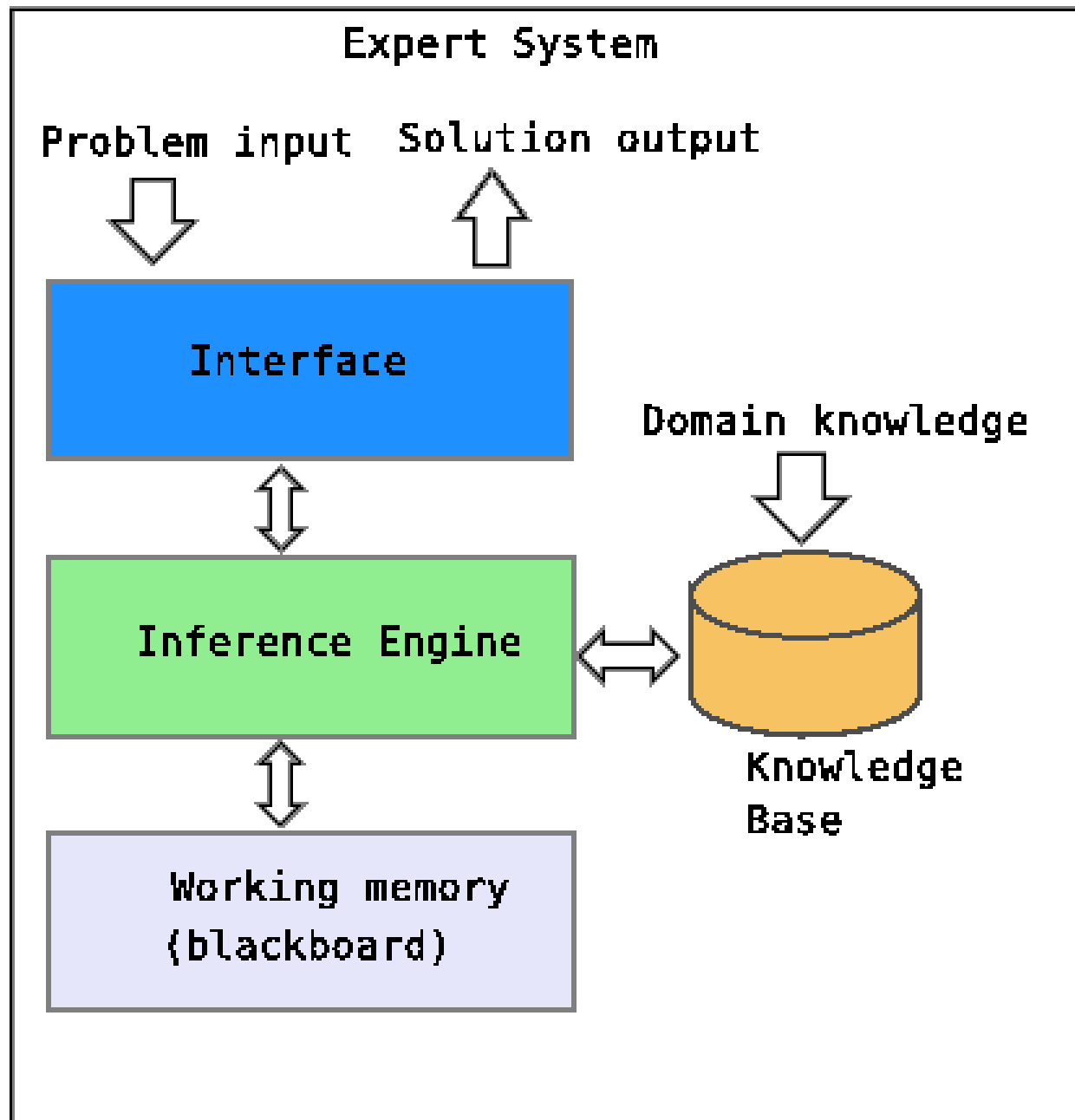


Ekspertni sistemi



Osnove ekspertnih sistema

- “ Ekspertni sistem (ES) je sistem VI koji se bazira na znanju
- “ Ovakvi sistemi se koriste u uskim oblastima za rešavanje problema koje je teško formalizovati
- “ Pod formalizovanjem se podrazumeva formulisanje rešenja problema nezavisno od konkretnih uslova u datom slučaju, tj. opšteg rešenja koje će sa sigurnošću važiti u svakom konkretnom slučaju
- “ Takvo rešenje može se koncizno izraziti preko odgovarajućeg algoritma
- “ Za algoritamski izraženo rešenje može se relativno jednostavno napisati program u nekom od klasičnih proceduralnih programskih jezika.

Osnove ekspertnih sistema

- “ U slučajevima kada algoritam za rešavanje nije poznat, ne postoji, ili nije pogodan za direktnu primenu na računaru (problem NP tipa koji premašuje računarske resurse koji su na raspolaganju), potreban je drugačiji prilaz koji daju ES
- “ Dodatne otežavajuće okolnosti mogu još da budu nepotpuni podaci o problemu koji se rešava, koji uz to mogu biti i nedovoljno pouzdani ili subjektivni
- “ Klasičan način koji se primenjuje za rešavanje takve vrste problema je naravno konsultacija ili angažovanje jednog ili tima stručnjaka iz te oblasti, eksperta, koji svojim znanjem, iskustvom, domišljatošću, pa čak i intuicijom rešava problem

Osnove ekspertnih sistema

- “ Termini domišljatost i intuicija, pre svega označavaju da se između ostalog koriste i nestandardne metode i postupci svojstveni datom subjektu, tj. ekspertu.
- “ Ekspertni sistem je računarski program koji treba da simulira aktivnost eksperta i da u fizičkom odsustvu osobe eksperta, reši problem na način sličan načinu na koji ekspert postupa

Osnovne karakteristike ekspertnih sistema

- “ ES predstavljaju alternativan način za automatizaciju postupka rešavanja problema primenom računara, u odnosu na klasičan pristup programiranja korišćenjem proceduralnih programskih jezika
- “ Neke od osnovnih karakteristika ES istovremeno predstavljaju i suštinske razlike u odnosu na klasična rešenja. Neke od ovih odlika ES navedene su u prethodnom poglavlju u delu o ES
- “ Ovde će te karakteristike biti nešto detaljnije razmotrene

Baziranost na znanju (knowledge based)

- “ Najvažnija karakteristika ES je da se baziraju na znanju
- “ Ekspertsko znanje se može odnositi na konkretan problemski domen kao i na upravljačko znanje koje se odnosi na samo funkcionisanje ES - tzv. Meta znanje (Meta Knowledge) ili “znanje o znanju”
- “ Ove dve vrste znanja su odvojene, ali se mogu predstaviti u istoj formi, tako da se razmatraju na isti način
- “ Postoji više različitih metoda za predstavljanje znanja u bazi znanja ES - produkciona pravila, okviri, semantičke mreže, o čemu će više reči biti kasnije

Poseduju mehanizam za zaključivanje (inference engine)

- “ Osnovne strukturne komponente ES su Baza Znanja (BZ) (Knowledge Base), Mehanizam Zaključivanja, (MZ), Radna Memorija (RM) i Korisnički Interface (KI) (User Interface)
- “ Mehanizam zaključivanja u toku rada ES pretražuje bazu znanja i na osnovu sadržaja baze znanja i stanja podataka i činjenica u radnoj memoriji vrši zaključivanje
- “ Rezultat zaključivanja mogu biti nove činjenice u radnoj memoriji koje MZ nadalje može ravnopravno da koristi za donošenje novih zaključaka
- “ Rad MZ traje do postizanja odgovarajućeg cilja ili do stanja u RM kada više nije moguće izvesti nijedan zaključak

Preovlađuje obrada simboličkih podataka (symbolic processing)

- “ Podaci sa kojima ES rade mogu u principu biti proizvoljnog tipa
- “ Međutim specifičnosti u samom pristupu problemu koje ES donose se između ostaloga ogledaju i u tome da postoje značajne potrebe pa odatle i mogućnosti za korišćenje i obradu podataka koji nisu numeričke prirode, tzv. simboličkih podataka
- “ To se pre svega odnosi na samu BZ i RM u kojima su skoro nezavisno od oblasti primene i konkretnog problema u velikoj meri zastupljeni podaci nenumeričke prirode

Imitiranje misaonog procesa eksperta

- “ BZ sadrži znanje eksperta
- “ Znanje se organizuje na takav način da ES pri rešavanju problema postupi slično ili isto kao i ekspert u toj situaciji
- “ Ovde treba napomenuti da se radi o funkcionalnoj sličnosti ili istovetnosti, što ne mora da ima nikakve veze sa samim mehanizmom misaonih procesa u ljudskom mozgu, koji je u velikoj meri nepoznat, i nije tema kojom ćemo se baviti
- “ Adekvatno organizovanje znanja u BZ iziskuje aktivnu saradnju samog eksperta kao i izbor najpogodnijeg metoda predstavljanja znanja

Mogućnost rada sa nepouzdanim podacima (low confidence data)

- “ Jedan od zahteva koji se postavljaju pred ES je mogućnost rada sa nepouzdanim podacima (low confidence data), kao i u slučaju kada svi potrebni podaci nisu poznati ili dostupni (unknown data)
- “ U takvim slučajevima veoma je važno da ES ne prekine rad, već da oponašajući eksperta nastavi rad i izvrši moguću delimičnu analizu koja zavisi od raspoloživih podataka
- “ Gradacija rešenja koje daje ES treba da zavisi od raspoloživih podataka i njihove pouzdanosti

Mogućnost obrazlaganja dobijenih zaključaka i rezultata

- “ Osim znanja o problemu koji se rešava, ekspertni sistemi poseduju i znanje o sopstvenoj strukturi i operacijama, pomenuto meta znanje
- “ To omogućava da korisnik može u toku rada da od ES traži da obrazloži logički redosled svojih zaključaka, tj. da rekonstruiše put zaključivanja kojim je došao do odgovarajućeg zaključka
- “ Obrazloženje bi trebalo da bude što potpunije i po mogućstvu u formi što bližoj prirodnom jeziku
- “ Značaj obrazloženja je u tome da olakšava praćenje rada ES i traženje i ispravku eventualnih greški

Ekspertni sistemi se od drugih oblasti VI razlikuju po tome što:

- “ Rešavaju teške zadatke na nivou eksperta
- “ Naglašavaju strategije rešavanja problema karakteristične za tu oblast
- “ Imaju sposobnost samosaznanja, tj, mogu da opravdaju i obrazlože svoje zaključke kao i da daju objašnjenja
- “ Rešavaju zadatke koji zahtevaju interpretaciju, dijagnozu, predviđanje, oblikovanje, planiranje

Neka ograničenja i nedostaci ekspertnih sistema su:

- “ Ne mogu da prepoznaju i rešavaju probleme za koje je njihovo znanje nedovoljno ili neprimenljivo. Osnovna pretpostavka je naravno da se ES adekvatno koristi – za rešavanje problema
- “ Nemaju nezavisna sredstva za proveravanje svojih zaključaka
- “ Nemaju dovoljno znanja o svojim ograničenjima i mogućnostima
- “ Obrazloženja o zaključcima su dosta uprošćena
- “ Jezik za izražavanje činjenica i odnosa je dosta uprošćen

Klasifikacija ekspertnih sistema

- “ ES danas nalaze primenu praktično u svim delatnostima i ima ih raznih vrsta
- “ Jedna od mogućih podela ES je u odnosu na osnovni problem ili zadatak koji preovlađuje u datom konkretnom slučaju
- “ Pošto je ES maksimalno prilagođen tipu problema koji se rešava, zahtevi koje dati problem postavlja u velikoj meri određuju karakteristike ES

Tipovi ekspertnih zadataka

- “ Neki tipični primeri zadataka koji se u praksi susreću su: interpretacija, dijagnostika, monitoring, prognoza, planiranje, projektovanje
- “ Svaki od ovih navedenih zadataka ima svoje specifičnosti, po kojima se razlikuje od ostalih
- “ Moguće je takođe neke složenije probleme svesti na jedan ili više ovih zadataka, tako da ih možemo smatrati i osnovnim tipovima zadataka koje ES rešavaju

Interpretacija

- “ Zadatak interpretacije je izvođenje odgovarajućih zaključaka, tumačenja ili objašnjenja na osnovu nekog skupa podataka
- “ Pri tome je osnovni zahtev da tumačenje bude korektno i saglasno sa podacima
- “ Ako je moguće više različitih interpretacija podataka, onda je poželjno da razmatranje pojedinih mogućih interpretacija bude sistematsko
- “ Odbacivanje pojedinih mogućnosti se pri tome vrši samo kada se sa sigurnošću utvrdi da dati slučaj interpretacije ne zadovoljava za dati skup podataka

Interpretacija

- “ Jedan od osnovnih problema pri interpretaciji podataka je neizbežno prisustvo šuma i greški
- “ Šum nastaje usled raznih smetnji i neželjenih uticaja na podatke koji se interpretiraju
- “ Usled toga se među korisnim podacima nalaze i lažni podaci koji u odsustvu šuma ne bi ni bili prisutni, i koje na neki način treba detektovati i razdvojiti od pravih podataka
- “ Uticaj greški je drugačiji, jer nema novih lažnih podataka, već su vrednosti pravih podataka izmenjene u izvesnoj meri

Interpretacija

- “ Mogući su i kombinovani efekti, i to naročito u slučaju kada su lažne vrednosti usled šuma i vrednosti pravih podataka istog reda veličine
- “ Konkretni postupci za umanjivanje ili otklanjanje uticaja šuma i greški značajno zavise od vrste konkretnog problema i fizičke prirode podataka koji se interpretiraju

Dijagnostika

- “ Pod dijagnostikom se podrazumeva proces čiji je cilj da se ustanovi mesto (lokalizacija), vrsta i uzrok neispravnosti u nekom sistemu
- “ Kao što je dobro poznato sistem može biti živo biće, kada dijagnozom treba da se ustanovi mesto i vrsta oboljenja ili povrede tkiva ili organa
- “ Dijagnostika u medicini konkretno predstavlja veoma važnu oblast primene ES koji su se tu pokazali kao veoma uspešni, o čemu svedoče uspešne realizacije ES - MYCIN, ONCOCYN
- “ Zadatak dijagnostike se delimično zasniva i na već pomenutom zadatku interpretacije podataka koji mogu predstavljati na pr. simptome oboljenja ili nekog kvara

Dijagnostika

- “ Od velikog značaja za dijagnostiku je i poznavanje datog problemskog domena i veza koje postoje između podsistema i ispitivanog sistema
- “ Najvažniji problemi koji se javljaju u dijagnostici su:
- “ Ako istovremeno u sistemu postoje dva ili više defekata (kvarova, neispravnosti, oboljenja, povreda) može doći do pojave da simptomi jednog defekta maskiraju simptome drugog defekta
- “ Moguće je da uređaji za ispitivanje i merenje na sistemu daju netačne podatke, tako da treba unapred računati sa takvim mogućnostima i izvršiti redundantna merenja koja će dati nesaglasne rezultate u slučaju neispravnosti

Dijagnostika

- “ Simptomi nekada mogu biti privremeni, što zahteva praćenje ponašanja ispitivanog sistema u dužem vremenskom periodu – monitoring
- “ Neophodno potrebni podaci za dijagnozu mogu biti teško dostupni ili skopčani sa raznim opasnostima

Monitoring

- “ Kod zadatka monitoringa potrebno je kontinualno pratiti jednu ili više veličina koje su od značaja – indikatori za ispravno funkcionisanje ispitivanog sistema
- “ Često je za uspešno praćenje procesa neophodno i adekvatno tumačenje izmerenih vrednosti, tako da je u tom slučaju kao podzadatak uključena i interpretacija
- “ Za zadatak monitoringa karakteristično je to da je vremenski kritičan, tj. vrši se u realnom vremenu i ON LINE

Monitoring

- “ Ovi pojmovi su detaljno objašnjeni u predmetu akvizicija podataka i to u odeljku o digitalnim merenjima
- “ Monitoring podrazumeva prepoznavanje i signaliziranje u slučaju kritičnih situacija kada mereni podaci mogu da poprimaju nedozvoljene vrednosti, da se menjaju nedozvoljenom brzinom, da budu u nekoj nedozvoljenoj kombinaciji itd
- “ Da li će se neka konkretna situacija prepoznati kao kritična, može da zavisi i od stanja sistema u celini

Prognoza

- “ Prognoza je predviđanje toka događaja u budućnosti na osnovu poznavanja modela sistema i ponašanja u prošlosti i sadašnjosti
- “ Prognoza kao zadatak VI uvek podrazumeva nemogućnost egzaktnog predviđanja ponašanja nekog sistema bilo zato što ne postoji odgovarajući model, ili podaci kojima se raspolaže nisu potpuni
- “ Slučaj kada je moguće egzaktno odrediti ponašanje nekog sistema ne predstavlja problem i nije od interesa za VI
- “ Rezultat prognoze su razne moguće varijante i teorije budućih događaja, kao i procena njihove osetljivosti u odnosu na zavisnost od promene ulaznih podataka
- “ Po mogućstvu, dobro je koristiti podatke iz različitih izvora.

Planiranje

- “ Planiranje je sastavljanje redosleda akcija koje treba izvršiti u određenom redosledu, da bi se ostvario neki cilj
- “ Teži se optimalnom planu, kojim će se uz što manji utrošak sredstava (vremena, energije, finansija) istovremeno zadovoljiti svi postavljeni zahtevi
- “ Ako dođe do konflikta između postavljenih ciljeva, treba odrediti prioritet ciljeva
- “ Problem planiranja može da bude vrlo složen, tako da je teško odrediti sve posledice koje mogu da nastupe u raznim slučajevima
- “ U slučaju velikog mnoštva detalja, treba razmatrati prvo one najvažnije
- “ Ako je područje planiranja nedovoljno poznato, javlja se i neodređenost kao problem koji treba rešiti

Projektovanje

- “ Zadatak projektovanja je izrada detaljne dokumentacije o izgradnji nekog sistema koja zadovoljava razne zahteve
- “ Postoji značajna sličnost između projektovanja i planiranja
- “ Raznorazna ograničenja koja postoje u vezi projekta po pravilu ne omogućavaju primenu jedinstvene teorije koja će omogućiti usklađivanje ograničenja i usvojenih rešenja
- “ Obimni i složeni projekti zahtevaju podelu na veći broj manjih potprojekata, što opet stvara dodatne probleme njihovog usklađivanja, pošto su vrlo retko sasvim nezavisni
- “ U toku izrade obimnih projekata, potrebno je voditi evidenciju o usvojenim rešenjima i odgovarajućim razlozima
- “ To može mnogo da pomogne pri modifikaciji projekta, kada je potrebno ponovo razmotriti razne varijante

Tipovi ekspertnih sistema

- “ U prethodnom odeljku navedeni tipski zadaci ES predstavljaju i osnov za klasifikaciju ES.
- “ Neki tipovi ES namenjeni su upravo rešavanju pojedinih odgovarajućih problema, i to su na pr. ES za interpretaciju, monitoring, dijagnostiku, prognozu, planiranje i projektovanje
- “ Neki drugi složeniji ES mogu da obuhvataju i više osnovnih zadataka kao na primer obučavanje i upravljanje
- “ ES za interpretaciju namenjeni su nalaženju opisa situacije ili stanja, davanju objašnjenja na osnovu podataka sa kojima raspolažu. Tipične konkretne primene su za prepoznavanje oblika, razumevanje govora, analizu slike, određivanje hemijske strukture jedinjenja, analizu signala itd.

Tipovi ekspertnih sistema

- “ ES za interpretaciju namenjeni su nalaženju opisa situacije ili stanja, davanju objašnjenja na osnovu podataka sa kojima raspolažu. Tipične konkretne primene su za prepoznavanje oblika, razumevanje govora, analizu slike, određivanje hemijske strukture jedinjenja, analizu signala itd.
- “ ES za prognozu izvode moguće logične posledice na osnovu date situacije u sistemu koji se ispituje. Tu spadaju vremenska prognoza, demografska prognoza, prognoza stanja na putevima, prognoza letine, prognoza vojnopolitičke situacije, kretanja u ekonomiji itd.

Tipovi ekspertnih sistema

” ES za dijagnostiku zaključuju o poremećaju u radu ispitivanog sistema na osnovu rezultata posmatranja. ES se primenjuju za dijagnostiku u medicini, elektronskim i mehaničkim uređajima i postrojenjima, za računarske sisteme, itd. ES obično rešavaju zadatke ovog tipa tako što povezuju simptome neispravnog funkcionisanja sistema sa odgovarajućim mogućim uzrocima. Jedan od načina za uspostavljanje ovih veza je korišćenjem odgovarajućih tabela, ili korišćenjem znanja o strukturi sistema, o slabim mestima u sistemu.

Tipovi ekspertnih sistema

- “ ES sistemi za projektovanje izgrađuju konfiguraciju sistema koja treba da zadovoljava ograničenja data projektnim zadatkom. U te zadatke spadaju sinteza i sastavljanje elektronskih uređaja, projektovanje građevinskih objekata, predviđanje budžeta itd. Jedan od dodatnih zahteva koji se postavljaju pred zadatak projektovanja može biti minimizacija u odnosu na troškove na primer.
- “ ES za planiranje projektuje plan rada. Ovde spadaju automatsko programiranje, planiranje kretanja robota, planiranje u projektovanju, sastavljanje maršruta, planiranje komunikacija, planiranje eksperimenata, planiranje vojnih operacija itd.

Tipovi ekspertnih sistema

- “ ES za monitoring vrše neprekidno praćenje veličina od značaja za funkcionisanje sistema, i proveravaju prikupljene rezultate u cilju eventualnog blagovremenog otkrivanja alarmantnih stanja i procesa u sistemu. Postoji mnoštvo ES za monitoring koji se koriste u atomskim elektranama, za praćenje kretanja vazdušnih masa, praćenje stanja bolesnika, finansija, itd. Ovi sistemi uvek rade u realnom vremenu.
- “ ES za obučavanje planira komunikaciju sa studentom u toku koje student uči, otkriva i koriguje slabosti i nedostatke u znanju. Rad takvih sistema se zasniva na opisu znanja studenta, što omogućava interpretiranje njegovog ponašanja i nalaženje načina za korekciju.

Tipovi ekspertnih sistema

- “ ES za upravljanje omogućavaju adaptivno upravljanje nekim sistemom koji se reguliše. ES sistemi se koriste za upravljanje u slučajevima kada tradicionalni sistemi automatskog upravljanja i regulisanja (kontinualni i digitalni), usled složenosti i velikog broja veličina ne zadovoljavaju. Upravljanje ES obuhvata stalni monitoring, interpretaciju dobijenih podataka, prognoziranje budućeg ponašanja sistema, dijagnostiku eventualnih problema i poremećaja u sistemu, planiranje postupaka za otklanjanje poremećaja i problema, kontrolu sprovođenja postupaka. ES za upravljanje nalaze primenu za regulisanje vazdušnog saobraćaja, upravljanje poslovanjem, upravljanje ratnim operacijama, upravljanje računarskim mrežama itd.

Istorijski pregled ES

- “ Prvi ES se javljaju sredinom 60 - tih godina. Oblast primene i problematika ovih ES je vezana za medicinu, hemiju, matematiku, rudarstvo, izgradnju računarskih sistema, dijagnostiku kvarova na uređajima i postrojenjima, analizu i razumevanje glasa, itd.
- “ Ovde će kao ilustracija i primer ukratko biti pomenuta dva izuzetno značajna ES, DENDRAL i MYCIN.
- “ Veliki značaj ovih ES je osim činjenice da su bili izuzetno uspešni u svom domenu, i taj da su predstavljali svojevrsne uzore i time određivali dalji smer razvoja ES.

Struktura ekspertnog sistema

- “ Različiti ekspertni sistemi koji obavljaju različite zadatke u raznim oblastima primene osim neizbežnih specifičnosti uglavnom imaju istu ili sličnu strukturu.
- “ Osnovna karakteristika strukture ekspertnih sistema je razdvojenost dva osnovna elementa, baze znanja od mehanizma zaključivanja.
- “ Mehanizam zaključivanja analizira bazu znanja pri procesu rešavanja problema u kome se preko niza zaključaka dolazi do rešenja.
- “ Ovakva struktura je pre svega uslovljena uviđanjem činjenice da se u postupku izvođenja zaključaka primenjuju uglavnom isti mehanizmi zaključivanja na bazama znanja koje mogu biti bitno različite sadržine.

Struktura ekspertnog sistema

- “ Izdvojena logika, tj mehanizam zaključivanja, ne mora da bude uslovljena konkretnim sadržajem baze znanja, već se može na isti način primenjivati na različite sadržaje.
- “ To omogućava da se ekspertni sistem izgrađuje i prilagođava konkretnoj problematici pre svega formiranjem potrebne baze znanja za taj problem.
- “ U slučaju kada treba izvršiti izmene ekspertnog sistema u cilju prilagođavanja dovoljno je napraviti izmene samo u bazi znanja.
- “ Kod klasičnih programa pravila zaključivanja su ugrađena u algoritam, a samim tim i u proceduralni deo programa.

Struktura ekspertnog sistema

- “ Kod ekspertnih sistema u bazu znanja se ugrađuju pravila zaključivanja.
- “ U slučaju izmena kod klasičnih programa neophodno je promeniti proceduralni deo kojim se realizuje algoritam obrade podataka.
- “ Kod ekspertnih sistema kao što je već rečeno menja se samo baza znanja, dok mehanizam zaključivanja ostaje isti, što ih čini fleksibilnim.
- “ Činjenice, podaci i zaključci neposredno vezani za tekući problem koji ES razmatra su dinamičke prirode i čuvaju se u radnoj memoriji.

Struktura ekspertnog sistema

- “ Činjenice može da zadaje korisnik u toku rada, a takođe i na osnovu tih činjenica izvedeni zaključci predstavljaju činjenice koje se dalje mogu koristiti u procesu zaključivanja.
- “ Mehanizam zaključivanja rešava problem tako što koristi podatke iz radne memorije i pravila iz baze znanja.
- “ Konkretna način rada mehanizma za zaključivanje zavisi od izabranog načina predstavljanja znanja, kao i od usvojene strategije pretraživanja baze znanja, o čemu će biti reči kada se budu razmatrali konkretni načini predstavljanja znanja.

Struktura ekspertnog sistema

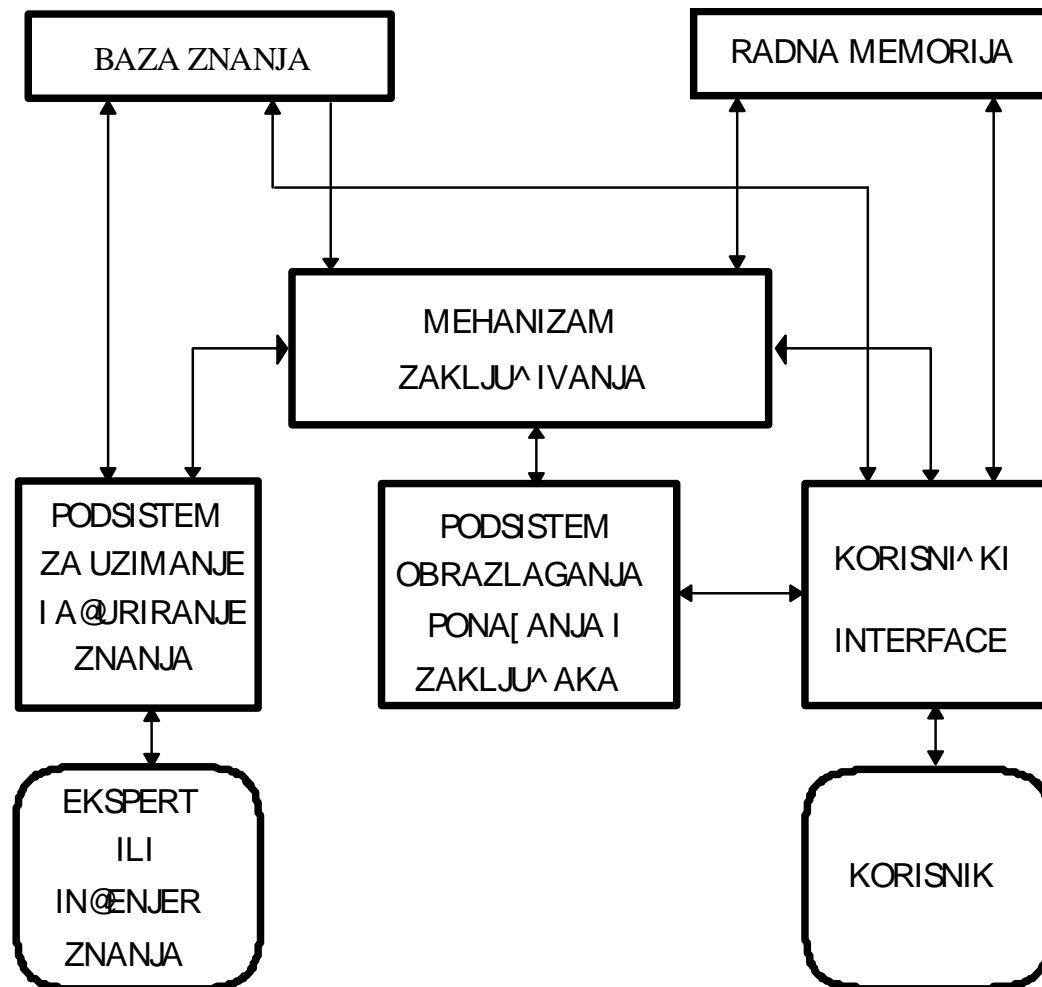
- “ Da bi se što više olakšala komunikacija ekspertnog sistema sa korisnikom koristi se poseban programski modul koji se naziva korisnički interface.
- “ Preko ovog interface-a razmenjuju se informacije između korisnika i ekspertnog sistema u toku procesa rešavanja problema.
- “ Korisnik može da odgovara na pitanja sistema, najčešće kada je potrebno uneti potrebne činjenice za problem koji se rešava, kao i da postavlja pitanja koja mogu biti u vezi sa zahtevom da se obrazlože dobijeni zaključci do kojih je ES došao.

Struktura ekspertnog sistema

- “ Podsistem za obrazlaganje zaključaka najčešće funkcioniše tako što omogućava identifikovanje koraka u procesu zaključivanja, i izlistavanje aktiviranih elemenata baze znanja u toku izvršavanja programa.
- “ Izmene u bazi znanja se kao što je već pomenuto mogu vršiti manuelno, dakle od strane inženjera znanja ili eksperta u datom domenu, ili automatski bez čovekove intervencije.
- “ Automatske izmene u bazi znanja podrazumevaju mašinsko učenje čime se generališe iskustvo u formi novog znanja.

Struktura ekspertnog sistema

” Na osnovu do sada pomenutih komponentata ekspertnog sistema, može se globalna struktura prikazati sledećom slikom.



Baza znanja

- “ Neophodna pretpostavka za funkcionisanje ekspertnog sistema je da raspolaže znanjem u vezi problema koji rešava.
- “ Postoje različita mišljenja o tome šta je znanje i kako ga bliže odrediti i definisati.
- “ Jedno od prihvatljivih određenja znanja je da je to relativno istinito uverenje prihvaćeno na osnovu stroge i nepristrasne procene vrednosti.
- “ Tokom razvoja AI postepeno se prelazilo od sistema zasnovanih na pretraživanju, ka sistemima zasnovanim na znanju.

Baza znanja

- “ Znanje koje je određenom tehnikom reprezentovano u memoriji računara tako da mu se može po potrebi na adekvatan i efikasan način pristupiti, naziva se baza znanja.
- “ Postoje različiti načini za reprezentovanje znanja u ES od kojih su najpoznatiji produkciona pravila, semantičke mreže, trojke OAV, frame-ovi, objekti itd.
- “ Kod klasičnih proceduralnih programa (jezici kao Basic, Pascal, Fortran, C...) znanje je algoritamskog tipa - proceduralno i izraženo je samom programskom procedurom, što znači da je pogodno za simboličko predstavljanje putem dijagrama toka (flow chart).

Baza znanja

- “ Suprotno tome je deklarativno znanje koje je opisnog karaktera, i u programima se nalazi u bazi podataka.
- “ Praksa pokazuje da je za ES najvažnije znanje koje ne pripada isključivo ni jednoj ni drugoj kategoriji.
- “ Znanje od najvećeg značaja za ES je tzv. “conditional knowledge” ili u prevodu uslovno znanje.
- “ Znanje kod ES može imati veoma složenu strukturu.
- “ U literaturi se dosta često sreću razni predlozi za podelu znanja po tipu, sastavu, funkciji, odnosu, nivou.

Baza znanja

- “ Jedna od mogućih podela po tipu, je na “know what” i “know how”
- “Know what” bi u tom slučaju odgovaralo deklarativnom znanju, dok bi “know how” odgovaralo proceduralnom znanju.
- Podela znanja analogna sa razdvajanjem baze znanja i mehanizma zaključivanja kod ES, je na upravljačko i domensko znanje.
- Domensko znanje određuje šta je poznato o domenu - oblasti primene, dok upravljačko znanje određuje način primene domenskog znanja.

Baza znanja

- ” Često se smatra da je domensko znanje deklarativno, dok je upravljačko znanje proceduralno.
- ” Van Harmelen međutim pokazuje da to nije ispravno stanovište, jer da li će znanje biti deklarativnog ili proceduralnog karaktera zavisi od usvojenog načina reprezentacije znanja.
- ” Na taj način, i domensko i upravljačko znanje može biti deklarativno, ako se predstavi na pr. logički, ili proceduralno ako se predstavi produkcionim pravilima.

Baza znanja

- “ Još neke podele znanja su:
- “ Prema odnosu (relationship) - Deklarativno i proceduralno znanje
- “ Prema funkciji (function) - Suštinsko, bitno (essential) i heurističko znanje
- “ Prema nivou (level) - Plitko (shallow) i duboko (deep) znanje

Produkcionalna pravila

- “ Produkcionalna pravila su dosta čest način predstavljanja znanja kod ES, i omogućavaju da se u bazi znanja predstavi i memoriše ono znanje koje je po svom karakteru zaključivanje, a koje se može izraziti u obliku AKO - TADA (IF - THEN).
- “ Ovako predstavljeno znanje se može opisati i operacijom matematičke logike - \Rightarrow implikacijom.
- “ U AKO (IF) delu, koji se još naziva i levi deo pravila (left hand side rule) nalazi se uslov - supporting condition (još se koriste i termini IF klauzula - IF clause, antecedent - antecedent, premisa - premise).

Produkcionalna pravila

- “ Ovaj uslov može biti zadovoljen ili ne.
- “ Uslov može biti i neki složeniji logički izraz dobijen kombinovanjem pojedinih uslova logičkim operatorima konjukcije I (AND), disjunkcije ILI (OR) ili negacije (NOT).
- “ Ako se može utvrditi da je uslov istinit, zaključak u ONDA (THEN) delu (koriste se i termini THEN klauzula - THEN clause, konsekvens - consequence) prihvata se kao istinit.
- “ Ovakav način zaključivanja kod produkcionih pravila odgovara načinu zaključivanja koji se u matematičkoj logici naziva modus ponens.

Produkcionalna pravila

- ” Ako su p i q iskazi, onda je matematički zapis pravila zaključivanja modus ponens sledeći :

$$p, p \Rightarrow q \mid = q$$

- ” Ako je istinita vrednost iskaza p i implikacije $p \Rightarrow q$ tačna, onda je i iskaz q takođe tačan.
- ” Za iskaz q se u tom slučaju kaže da je semantička posledica formula p i $p \Rightarrow q$. p u tom slučaju odgovara uslovu kod produkcionih pravila, dok q odgovara zaključku.
- ” Značajan uticaj na primenu produkcionih pravila ima teza M. Rychener-a u kojoj je ukazano na osnovne pogodnosti ovog metoda.

Produkcionalna pravila

- “ Pravila predstavljaju nezavisne elemente znanja (chunks).
- “ Programer, ili inženjer znanja (knowledge engineer) ne mora da vodi računa o redosledu primene pravila u bazi znanja, niti je važan fizički redosled kojim su pravila uneta ili zapisana u bazi znanja.
- “ To omogućava postupno (incremental) razvijanje baze znanja bez potrebe da se menjaju postojeća pravila (naravno, samo ako su tačna i adekvatna problemu koji se rešava).
- “ Sintaksa koja se koristi kod različitih programskih sistema (OPS5, I2+, CLIPS, Eclipse...) koji se baziraju na produkcionim pravilima može biti dosta različita, tako da se u ovom razmatranju neće ulaziti u sintaksne detalje.

Produkciona pravila

- “ Sa druge strane, sintaksa će biti data u delu gde se razmatraju karakteristike sistema I2+, i kod novijeg i savremenijeg sistema CLIPS 6.22.
- “ Sve informacije o CLIPS-u kao i besplatni download programa i dokumentacije na web adresi:
www.ghg.net/clips/CLIPS.html - stara adresa!
- “ Nova adresa:
- “ <http://clipsrules.sourceforge.net/>

Nalaženje uzoraka pravila

- “ Ako je levi deo pravila (Left Hand Side - LHS) koji se može sastojati od jednog ili više uslova (conditional element) zadovoljen, dolazi do tzv. aktiviranja pravila (rule activation).
- “ Uslov pravila najčešće predstavlja nešto što se naziva mustra ili šablon (pattern).
- “ Zadati uslov pravila se poredi sa činjenicama i podacima u radnoj memoriji (pattern matching), u cilju nalaženja podataka i činjenica koji se uklapaju ili zadovoljavaju uslove pravila.
- “ Zavisno od oblika konkretnog uslova, podaci iz radne memorije mogu se na razne načine uklapati, tako da uslov na levoj strani pravila bude zadovoljen.

Nalaženje uzoraka pravila

- “ Ako u uslovu figurišu konkretne vrednosti podataka, potrebno je da i u radnoj memoriji postoji takav identičan podatak, dok ako se radi o promenljivoj ili džokeru (wild card), veći broj podataka iz radne memorije može zadovoljavati isti uslov.
- “ U slučaju kada u uslovu pravila figurišu promenljive mogu da postoje još i neka dodatna ograničenja (constraints).
- “ Vrednost iz radne memorije koja zadovoljava uslov i odgovara datoj promenljivoj, dodeljuje se toj promenljivoj (variable binding).

Nalaženje uzoraka pravila

- ” Ako se u istom ili u nekom drugom uslovu istog pravila pojavljuje ista promenljiva na više različitih mesta, podaci iz radne memorije koji odgovaraju toj promenljivoj moraju imati istu vrednost kao podatak koji odgovara prvom pojavljivanju te promenljive. Isto označene lokalne promenljive u uslovima različitih pravila su nezavisne.
- ” Pravilo čiji su uslovi zadovoljeni elementima radne memorije, i koje je samim tim i aktivirano, naziva se konkretan uzorak pravila (rule instantiation).

Nalaženje uzoraka pravila

- “ Neka činjenica može biti prisutna u radnoj memoriji iz dva osnovna razloga:
- “ unošenje od strane korisnika, učitavanje iz datoteke, rezultat neke spoljašnje procedure, itd.
- “ kao posledica izvršavanja desne strane nekog drugog prethodno aktiviranog i izvršenog pravila.
- “ U zavisnosti od stanja radne memorije, istovremeno može biti aktivirano nijedno, jedno ili više pravila.
- “ Jedno isto pravilo istovremeno može biti aktivirano različitim elementima radne memorije, tj. mogu postojati različiti konkretni uzorci jednog istog pravila.

Nalaženje uzoraka pravila

- “ Na desnoj strani pravila (Right Hand Side - RHS) može biti jedan ili više zaključaka ili akcija.
- “ Akcije su procedure koje izvršavaju neki algoritamski zadatak.
- “ Prema tome, rezultat izvršavanja aktiviranog pravila (rule firing) može biti izvođenje jednog ili više zaključaka sa desne strane pravila, kao i izvršavanje odgovarajućih akcija.
- “ Izvođenje zaključka znači da se odgovarajuća činjenica upisuje u radnu memoriju ravnopravno sa ostalima.
- “ Ove činjenice menjaju stanje radne memorije, tako da je moguće da rezultat izvršavanja pravila bude aktiviranje novih pravila iz baze znanja.

Nalaženje uzoraka pravila

- “ Efekat izvršavanja pravila može takođe da bude i brisanje već po+stojećih činjenica iz radne memorije, što može dovesti do deaktiviranja već aktiviranih pravila.
- “ U slučaju da baza znanja sadrži veliki broj pravila, proces poređenja levih strana pravila sa činjenicama u radnoj memoriji može vremenski dugo da traje, čak i na brzim računarima.
- “ Zato se koriste razni postupci u cilju povećanja efikasnosti poređenja.
- “ Do sada najefikasniji postupak za to je izražen tzv. RETE, algoritmom.

Nalaženje uzoraka pravila

- “ Ovaj algoritam je jedinstven zato jer je najmanje nekoliko puta brži od bilo kog drugog algoritma, a sa porastom broja pravila u bazi znanja, relativna brzina u odnosu na ostale algoritme raste.
- “ Tek primenom ovog algoritma produkcionim sistemima postaju dovoljno moćni i za tako ozbiljne primene kao što je na pr. izuzetno uspešni R1, ES za konfiguraciju računara VAX firme DEC (Digital Equipment Corporation), kasnije preimenovan u XCON.

Rezolucija konflikta

- “ Ranije pomenuta mogućnost istovremenog aktiviranja dva ili više pravila iz baze znanja, donosi problem izbora samo jednog pravila koje će se izvršiti.
- “ Rešavanje tog problema naziva se rezolucija konflikta (conflict resolution).
- “ Postoje razni načini i kriterijumi koji se mogu koristiti, tako da se kod pojedinih sistema može birati između više raznih strategija, na pr. 7 kod sistema CLIPS 6.0
- “ Kod nekih produkcionih sistema broj konfliktnih pravila se smanjuje brojčanim izražavanjem prioriteta pravila (salience kod CLIPS-a), tako da su konfliktna samo ona aktivirana pravila koja imaju isti najveći prioritet.

Rezolucija konflikta

- “ Neki od principa na kojima se može bazirati rezolucija konflikta su: refrakcija, starost činjenica u radnoj memoriji, specifičnost.
- “ Refrakcija je pojava u biologiji da jednom nadraženi neuron koji zato proizvede impuls, ne može odmah ponovo da proizvede novi impuls iako je nadražaj prisutan, već je potrebno da protekne neko vreme posle koga neuron ponovo može da reaguje na nadražaj.
- “ Analogno tome, uklanjanje se aktivacija neposredno izvršenog pravila, iako uslovi za aktiviranje tog pravila mogu i dalje biti zadovoljeni podacima u radnoj memoriji.

Rezolucija konflikta

- “ Na osnovu starosti, mogu se birati pravila koja su zadovoljena sa najstarijim ili najsvežijim činjenicama iz radne memorije.
- “ Takođe, pravila se mogu uređivati prema specificitetu, tj. prema broju uslova pravila, itd.

Ulančavanje unapred

- “ Ulančavanje unapred je jedna od dve osnovne strategije pretraživanja baze znanja koju može da koristi mehanizam za zaključivanje kod produkcionih sistema pri zaključivanju.
- “ Aktiviranje nekog pravila iz baze znanja određeno je trenutnim sadržajem činjenica u radnoj memoriji (context).
- “ Datim činjenicama u radnoj memoriji odgovara određeni skup aktiviranih pravila ili konkretnih uzoraka pravila, pri čemu uslovima na levoj strani svakog aktiviranog pravila odgovara neki podskup činjenica u radnoj memoriji.

Ulančavanje unapred

- “ Pravila koja su aktivirana nazivaju se i konfliktna pravila, pošto se samo jedno od aktiviranih pravila može izvršiti.
- “ Postupkom rezolucije konflikta, bira se samo jedno iz skupa konfliktnih pravila.
- “ Izabrano pravilo se izvršava, što znači da se zaključci na desnoj strani smatraju istinitim i upisuju se u radnu memoriju kao nove poznate činjenice.
- “ Eventualno postojeće procedure na desnoj strani se izvršavaju, što takođe može da dovede do promena u radnoj memoriji.

Ulančavanje unapred

- “ Kao rezultat izvršavanja pravila radna memorija može da se promeni, i novo stanje može da dovede do aktiviranja nekih novih pravila koja ranije nisu bila aktivirana.
- “ Takođe, može se desiti da neka prethodno aktivirana pravila više nisu zadovoljena, pa se zato uklanjaju sa liste aktiviranih pravila (agenda).
- “ Upravo izvršena pravila se uvek uklanjaju sa liste aktiviranih pravila, da ne bi došlo do eventualnog ponovnog aktiviranja, ukoliko su uslovi na levoj strani pravila i dalje zadovoljeni.
- “ Uklanjanje se vrši u saglasnosti sa principom refrakcije koji je pomenut kod rezolucije konflikta.

Ulančavanje unapred

- “ Proces zaključivanja traje sve dok postoji bar jedno aktivirano pravilo na listi.
- “ Ako se prethodno ne dođe do rešenja, stanje u radnoj memoriji koje ne daje ni jedno aktivirano pravilo znači da se do rešenja ne može doći sa datim skupom pravila u bazi znanja, zatim korišćenih ulaznih podataka, kao i izabrane strategije rezolucije konflikta.
- “ Nešto od toga onda treba promeniti da bi se došlo do rešenja.
- “ Odavde se vidi da je postupak ulančavanja pravila unapred vođen ili upravlján podacima.

Ulančavanje unapred

- “ Uvek se ide od datog skupa podataka - činjenica u radnoj memoriji, prema novim zaključcima koji dakle zavise od podataka, ali i od izabrane strategije rezolucije konflikta.
- “ Promenom strategije rezolucije konflikta moguće je promeniti i način rada mehanizma za zaključivanje.
- “ Kod sistema koji koriste strategiju ulančavanja unapred, vremenski najduže traje faza poređenja uslova na levoj strani pravila sa činjenicama u radnoj memoriji (pattern matching).
- “ Jedan od najefikasnijih algoritama za poređenje je već pomenuti RETE algoritam, kome je posvećen jedan od sledećih odeljaka tekućeg poglavlja.

Ulančavanje unazad

- “ Druga osnovna strategija koju može da koristi mehanizam zaključivanja kod produkcionih sistema je ulančavanje pravila unazad (backward chaining).
- “ Za razliku od prethodne strategije ulančavanja unapred, polazi se ne od podataka, već od cilja (goal) koji treba dokazati.
- “ Jedan ili više ciljeva se unapred postavljaju, a zadatak mehanizma za zaključivanje je da na osnovu baze znanja i konteksta odredi koji se od postavljenih ciljeva mogu ostvariti.
- “ U slučaju kada ima više ciljeva, mogu se hijerarhijski organizovati od strane programera, analogno naslovima i podnaslovima u sadržaju knjige.

Ulančavanje unazad

- “ Kod nekih sistema, na pr. kod korišćenog Insight 2+ (I2+)., postoji mogućnost da korisnik učestvuje u izboru cilja koji će se dokazivati.
- “ Jednom od specijalnih naredbi kojima se može u izvesnoj meri kontrolisati proces zaključivanja, GOALSELECT, zadaje se da li će korisnik moći da učestvuje u izboru cilja koji će se dokazivati, čime se vrši veoma važno usmeravanje koje može značajno da skрати vreme pretraživanja baze znanja pri zaključivanju.
- “ U tom slučaju korisnik ima mogućnost da bira jedan od ciljeva istog hijerhijskog nivoa.

Ulančavanje unazad

- “ Proces zaključivanja počinje od liste ciljeva najvišeg hijerarhijskog nivoa.
- “ Ukoliko korisnik izabere neki od ciljeva datog nivoa, prikazuju se dalje eventualno postojeći ciljevi nižeg nivoa za izabrani cilj.
- “ Na taj način ako je korisnik u stanju da odredi cilj, to mu se i omogućava, sve do cilja najnižeg nivoa.
- “ Ako korisnik ne može da se opredeli za neki od prikazanih ciljeva datog nivoa, postoji mogućnost da odgovori sa “nepoznato” (unknown), kada mehanizam za zaključivanje automatski preuzima dalju pretragu hijerarhije ciljeva.

Ulančavanje unazad

- “ Najčešće se radi o sistematskom pretraživanju ciljeva.
- “ Da bi se mogao dokazati cilj nekog nivoa, baza znanja treba da ima neko znanje o tom cilju.
- “ To znači da treba da postoji bar jedno ili više pravila sa zaključkom na desnoj strani pravila koji je doslovce (literally) jednak cilju koji se dokazuje.
- “ Mehanizam za zaključivanje pronalazi takva pravila u bazi znanja.
- “ Sva takva pravila ako ih ima više, sačinjavaju tzv. ILI klasu (OR class) pravila koja odgovaraju cilju.

Ulančavanje unazad

- “ Izbor jednog pravila iz ILI klase pravila može da podseća na rezoluciju konflikta kod ulančavanja unapred.
- “ Međutim suštinska razlika je u tome što sva ta pravila imaju isti zaključak - jednak cilju, dok kod ulančavanja unapred situacija može biti znatno složenija, jer konfliktna pravila mogu imati sasvim različite zaključke, tako da se izmenom pravila koje će se izvršiti (promenom strategije rezolucije konflikta), menja i kontekst, odnosno samo zaključivanje.
- “ Izbor jednog pravila iz ILI klase vrši se najčešće na osnovu faktora pouzdanosti (certainty factor - CF) koji se može dodeliti pravilima u toku razvoja baze znanja.

Ulančavanje unazad

- “ Ako za jedan ili više uslova na levoj strani pravila ne postoje elementi u radnoj memoriji koji odgovaraju ovim uslovima, onda se traži da li u bazi znanja postoje pravila sa zaključkom koji je jednak uslovu na levoj strani pravila.
- “ Ako takva pravila / pravilo postoji, onda se razmatraju uslovi na levoj strani tog pravila.
- “ Na taj način se vrši povezivanje - ulančavanje (chaining) pravila.
- “ Kada se stigne do uslova na levoj strani pravila koji se ne može odrediti na osnovu drugih pravila jer takva pravila ne postoje u bazi znanja, a u radnoj memoriji takođe nema elemenata koji odgovaraju tom uslovu, onda se od korisnika očekuje da unese odgovarajuću vrednost podatka.

Ulančavanje unazad

- “ Rezultat ovakvog ulančavanja pravila u bazi znanja je da se formira svojevrsno I / ILI (AND / OR) stablo.
- “ Koren ovakvog stabla je cilj koji se dokazuje, čvorovi koji se granaju su podciljevi - zaključci pravila, a grane odgovaraju uslovima na levoj strani pravila, koji mogu biti povezani sa I i ILI operacijama.
- “ Listovi drveta odgovaraju uslovima na levoj strani pravila za koje ne postoje pravila u bazi znanja sa istim zaključkom.
- “ Kada se u procesu pretraživanja baze znanja stigne do listova, od korisnika se zahteva da unese potrebne podatke.

Ulančavanje unazad

- “ U slučaju da korisnik ne zna odgovarajuće podatke, ili su podaci takvi da ne potvrđuju cilj koji se dokazuje, dalji rad mehanizma za zaključivanje - pretraživanje baze znanja se obustavlja.
- “ Iako cilj u tom slučaju nije dokazan, dobijeni međuzaključci mogu da predstavljaju pokazatelj dokle se došlo u rešavanju problema.
- “ Ovde treba napomenuti da mehanizam zaključivanja uvek prvo proverava da li odgovarajući podaci koji odgovaraju uslovima pravila možda već postoje u radnoj memoriji.

Ulančavanje unazad

- “ Ako su odgovarajući podaci prisutni u radnoj memoriji, ulančavanje se ne vrši, čak iako postoji jedno ili više pravila sa zaključkom jednakim ispitivanom uslovu.
- “ Razlog za to je da prisustvo potrebnih podataka u radnoj memoriji znači da je uslov već zadovoljen i da samim tim nikakvo daljnje pretraživanje baze znanja nije potrebno.
- “ Još jedna dodatna mogućnost za kontrolisanje procesa zaključivanja, tj. pretraživanje baze znanja je redosled uslova sa leve strane pravila.

Ulančavanje unazad

- “ Redosled provere uslova sa leve strane pravila jednak je redosledu kojim su ti uslovi navedeni u listingu baze znanja sa leve strane pravila koje se ispituje.
- “ Redosled uslova može da utiče na brzinu pretraživanja baze znanja.
- “ Na početku treba da se nalaze elementi koji se brže mogu proveriti, a elementi koji zahtevaju dužu proveru, tj. iza kojih se nalazi duži lanac treba da budu na kraju.
- “ Kada su elementi uslova povezani sa logičkim operatorom AND,. provera se prekida kada se dođe do prvog uslova pravila koji nije zadovoljen.

Ulančavanje unazad

- “ Redosled uslova može biti i logički nametnut, ako postoji međuzavisnost između pojedinih elemenata uslova.
- “ Na primer, proces ispitivanja (lanac pravila) nekog od uslova pravila može da postavi određene podatke - na pr. vrednosti promenljivih u kontekstu, koje se koriste pri ispitivanju drugih uslova koji su na taj način zavisni od pravila u lancu koja postavljaju te podatke.
- “ U tom slučaju uslovi koji zavise od drugih uslova u okviru istog pravila treba da budu navedeni posle ovih uslova na levoj strani pravila u listingu baze znanja.

Ulančavanje unazad

- ” Kod sistema za ulančavanje unazad, može da se koristi još i mogućnost izvršavanja alternativne grane na desnoj strani pravila iza ELSE, kada jedan ili više uslova na levoj strani pravila nisu zadovoljeni.
- ” Do izvršavanja desne strane pravila iza ELSE, dolazi samo u slučaju kada se do tog pravila dođe pri ulančavanju unazad.
- ” Kada bi takva mogućnost postojala kod sistema sa ulančavanjem unapred, izvršavanje takve alternativne grane ne bi ni izbliza bilo u toj meri determinisano, već bi se teoretski moglo izvršiti bilo kada, kada dati uslovi na levoj strani pravila nisu zadovoljeni.

Ulančavanje unazad

- “ U slučaju kada postoji više uslova na levoj strani koji su povezani sa AND, desna strana iza ELSE se u opštem slučaju izvršava kada bilo koji od uslova na levoj strani nije zadovoljen.
- “ Međutim, ako je potrebno da se deo iza ELSE izvrši samo kada određeni uslovi na levoj strani nisu ispunjeni, onda je potrebno obezbediti da ostali uslovi na levoj strani budu ispunjeni.
- “ To se može postići na taj način, što se u lancima koji odgovaraju uslovima koji treba da budu zadovoljeni ugrade mehanizmi koji obezbeđuju da je uslov ili zadovoljen ili ako to nije moguće, onda ES dolazi do nekog drugog cilja, i korisniku se signalizira da se dati cilj ne može postići.

Ulančavanje unazad

- “ Jedan od načina da se postigne ispunjenje uslova je da se kada je to moguće, stalno ispituju novi podaci sve dok se ne nađu oni sa kojima je uslov zadovoljen (primenom petlje sa FORGET i LOOP).
- “ U tom slučaju podaci koji se ispituju ne nalaze se u radnoj memoriji, već u nekoj bazi podataka.
- “ Dakle u oba slučaja i kada je uslov zadovoljen i kada nije zadovoljen, ne izvršava se alternativni deo pravila iza ELSE što je i bio cilj.
- “ To znači da možemo biti sigurni da će se deo iza ELSE izvršiti samo u slučaju neispunjavanja tačno određenih uslova na levoj strani pravila, a ne bilo kojih.

Ulančavanje unazad

- “ Naravno da se konstrukcija IF ... THEN ... ELSE može realizovati i bez korišćenja ELSE, ali uz korišćenje dva ili više pravila umesto jednog.
- “ Jedno pravilo bi odgovaralo ispunjenju svih uslova na levoj strani, kada se izvršava deo iza THEN a druga bi imala negaciju (NOT) samo onih uslova kada se hoće izvršavanje dela iza ELSE (sada bi to bilo iza THEN, ali ovog drugog pravila).
- “ Na taj način se eksplicitno navodi koji uslovi treba da budu zadovoljeni a koji ne, i ne moraju se uzimati u obzir nikakva dodatna razmatranja.

Ulančavanje unazad

- ” Osnovna razlika između metoda ulančavanja unapred i unazad, je u tome da kada su uslovi na levoj strani pravila zadovoljeni, kod “sistema unapred” dato pravilo se istog trenutka aktivira i stavlja na mesto u listi (agenda) određeno prethodno usvojenom strategijom rešavanja konflikta.
- ” Kod “sistema unazad”, činjenica da je leva strana pravila zadovoljena sa trenutnim kontekstom stanja, uopšte ne mora da znači da će se to pravilo ikada i razmatrati, ukoliko se do njega ne dođe pri ulančavanju unazad.

Ulančavanje unazad

” Obrnuto, može se reći da će se dato pravilo razmatrati nezavisno da li je uslov na levoj strani već zadovoljen ili ne, ako se do tog pravila dođe pri ulančavanju unazad, tj. ako je zaključak tog pravila jednak uslovu na levoj strani nekog drugog pravila koje se razmatra.

Poređenje metoda za ulančavanje unapred i unazad

- “ Interesantno je napraviti poređenje ulančavanja unapred i unazad, u pogledu mogućnosti zamene jednog sa drugim.
- “ Iako se ovi metodi dosta razlikuju, pre svega u smeru ulančavanja, tj. smeru povezivanja pravila pri pretraživanju baze znanja i zaključivanju, moguće je na primer sistemom za ulančavanje unapred simulirati ulančavanje unazad.
- “ Pri tome treba praviti razliku između različitih nivoa znanja i to. takozvanog višeg nivoa, Nivoa Znanja (Knowledge Level) i nižeg nivoa, simboličkog nivoa znanja na nižem nivou, na kome se posmatraju sam mehanizam zaključivanja, pravila i smer ulančavanja, naravno nije moguće ništa promeniti.

Poređenje metoda za ulančavanje unapred i unazad

- “ Međutim, pogodnom organizacijom baze znanja moguće je postići da na višem nivou znanja, po smislu stvarni smer zaključivanja bude sasvim drugačiji.
- “ Kod sistema I2+ koji se zasniva na ulančavanju unazad, moguće je korišćenjem naredbe CYCLE za vrući start baze znanja i naredbe FORGET za selektivno zaboravljanje podataka, postići da na višem nivou znanja stvarni smer zaključivanja bude suprotan, tj. od podataka prema cilju.

Poređenje metoda za ulančavanje unapred i unazad

- “ Postavlja se pitanje kada i u kom slučaju koristiti metodu predstavljanja znanja pomoću produkcionih pravila.
- “ Osim već pomenutih razloga koji su u vezi predstavljanja znanja koje se teško ili ne može formalizovati (izraziti algoritamski), i uvek onda kada se znanje može na pogodan način izraziti preko velikog broja uslova (tipa IF...THEN kod proceduralnih programa).
- “ U tom slučaju, produkciona pravila su prirodniji način izražavanja takvog znanja od odgovarajućeg proceduralnog programa.

Semantičke mreže

- “ Semantičke mreže predstavljaju suštinski drugačiji način predstavljanja znanja kod ES u odnosu na produkcione sisteme.
- “ Dok je kod produkcionih sistema osnovna karakteristika predstavljanje iskustvenog, specifičnog, heurističkog znanja, semantičke mreže se koriste za predstavljanje znanja o pojmovima, odnosa između pojmova, za određivanje hijerarhijske pripadnosti pojmova, i tome slično.
- “ Za simboličko predstavljanje semantičkih mreža koriste se grafovi.

Semantičke mreže

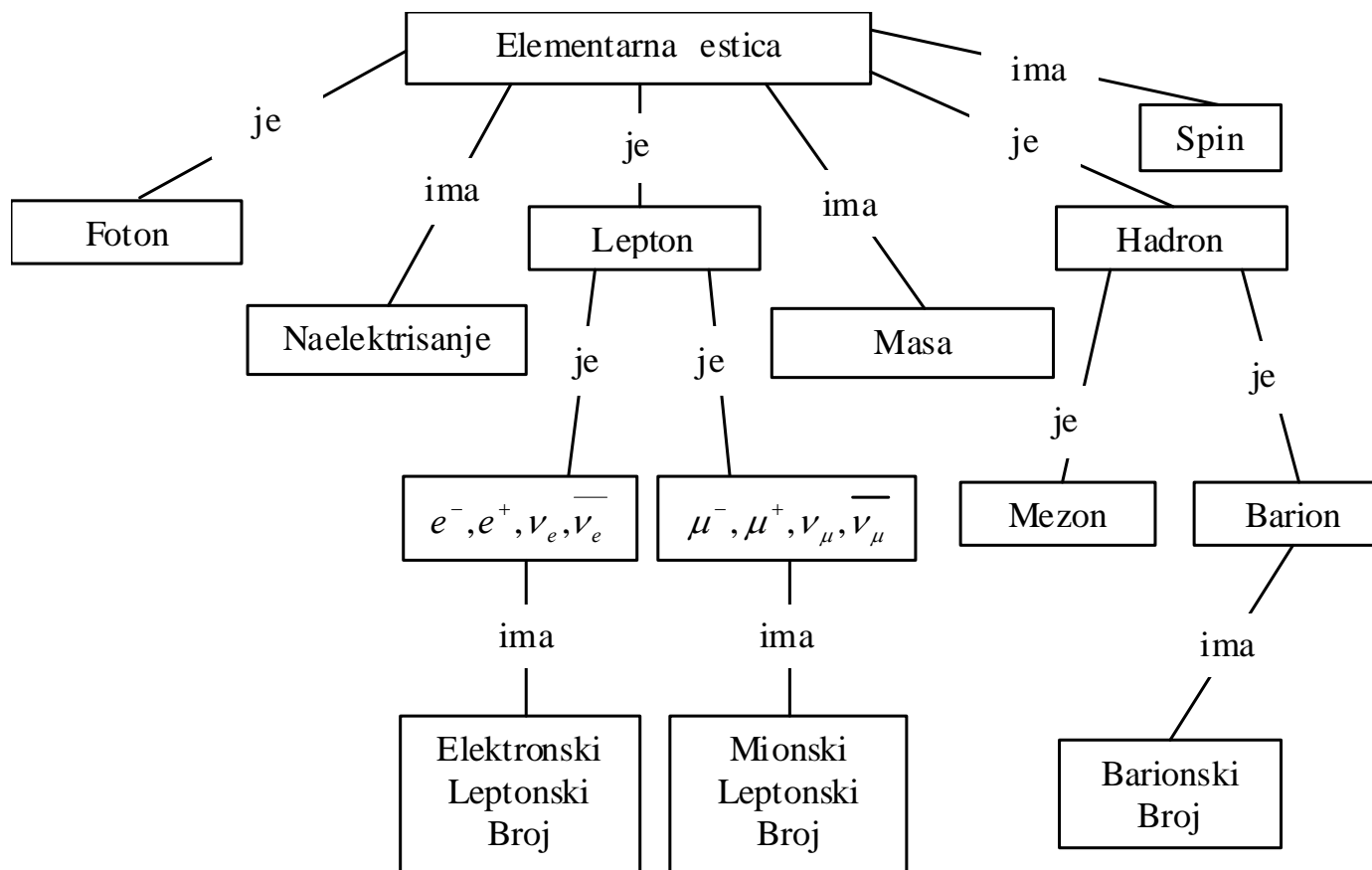
- “ Primena mreža za predstavljanje znanja počinje sa idejom da se ljudsko znanje predstavi pomoću mrežne strukture koja se sastoji od čvorova i usmerenih grana.
- “ Čvorovi su označeni imenima i predstavljaju pojmove.
- “ Usmerene grane koje povezuju čvorove, takođe su označene imenima i određuju vrste veza između čvorova, tj. pojmova koje predstavljaju.
- “ Jedna od veoma značajnih osobina semantičkih mreža, koja se koristi i kod drugih metoda predstavljanja, je mogućnost nasleđivanja (inheritance) osobina čvorova.

Semantičke mreže

- ” Čvorovi na nižim hijerarhijskim nivoima mogu da nasleđuju osobine čvorova na višim hijerarhijskim nivoima ako postoje odgovarajuće grane, tj. veze.
- ” Nasleđivanje donosi uštede u memorijskom prostoru, ali na račun smanjenja brzine rada programa.

Semantičke mreže

- ” Pojam na najvišem hijerarhijskom nivou na slici 13 je Elementarna čestica.
- ” Čvorovi Naelektrisanje, Masa i Spin predstavljaju karakteristike čvora Elementarna čestica.



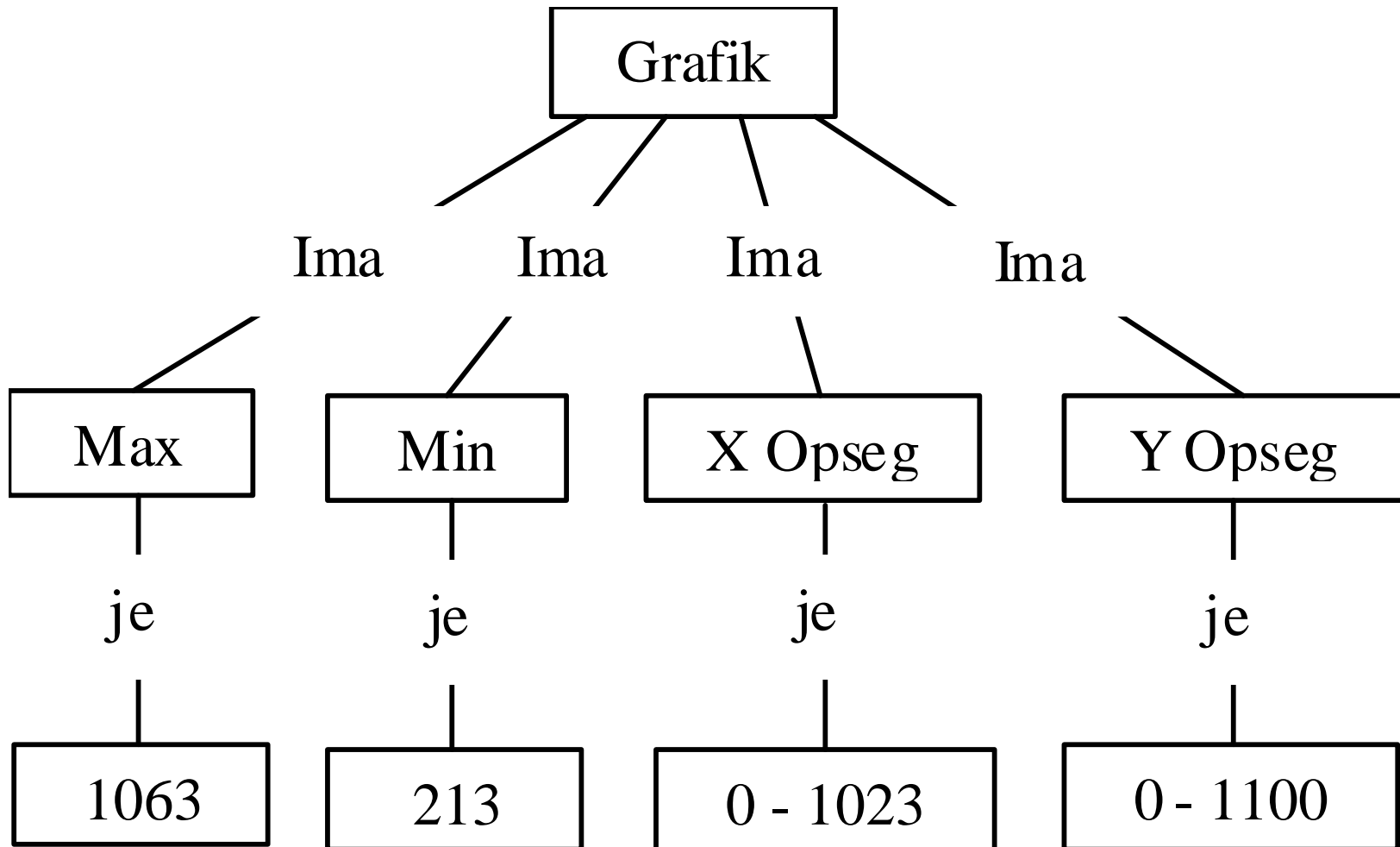
Semantičke mreže

- ” Pojmovi na sledećem nižem hijerarhijskom nivou su Foton, Lepton i Hadron.
- ” Pojmovi na nižim hijerarhijskim nivoima nasleđuju osobine pojmova sa viših nivoa.
- ” Tako Fotoni, Leptoni, Hadroni (Mezoni i Barioni) takođe imaju svojstva Masa, Naelektrisanje i Spin nasleđena od čvora Elementarna čestica.
- ” Čvorovi sa elektronskim i mionskim leptonskim česticama imaju i odgovarajuće leptonske brojeve, dok barioni imaju barionski broj.
- ” Ali leptoni nemaju barionski broj, niti barioni leptonski što i sledi iz putanja nasleđivanja na slici

Trojke OAV

- “ Specijalan slučaj semantičkih mreža kada postoje samo tri vrste čvorova, su trojke OAV.
- “ U tom slučaju čvorovi mogu da predstavljaju Objekte, Attribute objekata i Vrednosti atributa - OAV.
- “ Čvorovi mogu biti povezani sa samo dve vrste veza čiji su nazivi “jeste” (IS A) i “ima” (HAS A).
- “ Usmerene grane Objekat -> Atribut su tipa “ima”, dok su grane Vrednost -> Atribut tipa “jeste”.
- “ U slučaju ako je u sistemu predstavljen samo jedan objekat, trojke OAV se redukuju na parove AV, kada nema ni nasleđivanja.

Trojke OAV



Okviri

- “ Okviri (Frames) su način predstavljanja znanja koji je slično semantičkim mrežama prilagođen predstavljanju strukture pojmova, ali koji za razliku od mreža još objedinjava i procedure.
- “ Okvir odgovara nekom pojmu koji se predstavlja, i sastoji se od niza slotova (slots).
- “ Slotovi predstavljaju attribute kojima se pojam opisuje, i sadrže odgovarajuće vrednosti koje atributi mogu da imaju.
- “ Osim odgovarajućih vrednosti, slotovima mogu da budu pridružene i procedure koje se pozivaju ili aktiviraju kada se na neki način pristupa slotovima, na pr. kada se unose vrednosti, ili kada se već unesene vrednosti menjaju.

Okviri

- “ Ovakve procedure se nazivaju i demoni (demons) i mogu da na određeni način izračunavaju vrednost slota ili da rade nešto drugo.
- “ Slično čvorovima kod semantičkih mreža, i okviri su međusobno povezani vezama.
- “ Ove veze mogu da odgovaraju odnosu klasa - superklasa ili klasa - primerak klase (objekat)
- “ Kod okvira takođe postoji nasleđivanje atributa i vrednosti atributa od hijerarhijski viših okvira.
- “ Pri tome specifične vrednosti atributa hijerarhijski nižih okvira imaju prednost nad nasleđenim vrednostima od hijerarhijski viših okvira.

Okviri

- “ U slučaju ako nije poznata konkretna vrednost atributa nekog okvira, onda se uzima nasleđena vrednost od odgovarajućeg atributa hijerarhijski višeg okvira.
- “ Jedan okvir može da nasledi attribute i vrednosti atributa ne samo od jednog okvira, nego i od dva ili više okvira.
- “ U tom slučaju za okvir se kaže da ima više od jednog pretka (ancestor).
- “ Odgovarajući graf koji predstavlja hijerarhijsku strukturu okvira onda nije drvo, jer u slučaju drveta, svaki čvor može da ima samo jednog pretka, dok predak može da ima više potomaka (descendent).

Objekti

- ” Objektno Orijentisano Programiranje OOP (Object Oriented Programming) je danas jedan od veoma važnih i široko zastupljenih načina programiranja pored klasičnog proceduralnog, zatim programiranja baziranog na pravilima, konekcionističkog načina programiranja koje se bazira na neuralnim mrežama i dr.
- ” Objektno programiranje ima sličnosti sa predstavljanjem znanja pomoću frame-ova, jer je i tu zastupljeno objedinjavanje predstavljanja osobina objekata i procedura koje se koriste, kao i hijerarhijsko nasleđivanje

Objekti

- “ Osnovne razlike zbog kojih je OOP napredniji način programiranja je mogućnost nasleđivanja ne samo osobina (slots) već i samih procedura od hijerarhijski viših klasa.
- “ Neke od osnovnih osobina OOP su apstrakcija, nasleđivanje, enkapsulacija, polimorfizam.
- “ Bitne karakteristike nekog realnog sistema predstavljaju se slotovima klase objekata koja reprezentuje model sistema.
- “ Objekti su konkretni primeri klase objekata (instances), i imaju konkretne vrednosti slotova.
- “ Slotovi objekata su dobijeni od klase kojoj objekti pripadaju.

Objekti

- “ Klasa može da nasledi slotove kako od direktnih nadklasa (direct superclass), tako i od svih klasa duž staze nasleđivanja (inheritance path).
- “ Komunikacija između objekata vrši se preko poruka (messages).
- “ Da bi objekat mogao da odgovori na poruku određenog imena (message name) koja mu je upućena, mora postojati odgovarajući proceduralni programski kod (message handler).
- “ Ovaj kod se pridružuje određenoj klasi, i izvršava se kada objekat te klase primi poruku odgovarajućeg imena.

Objekti

- “ Isto kao i slotovi, i kod za odgovarajuću poruku može se naslediti od viših klasa duž putanje nasleđivanja.
- “ Enkapsulacija predstavlja svojstvo da se objektu može pristupati samo na strogo određeni način, tj. isključivo preko poruka.
- “ Na primer, slotovima nekog objekta može se pristupiti bilo u cilju dobijanja njihovih vrednosti ili promene vrednosti samo preko sistema poruka, tj. slanjem odgovarajućih poruka.
- “ Jedna od interesantnih mogućnosti je da iste poruke upućene objektima koji potiču od različitih klasa, mogu da daju sasvim različite efekte.

Objekti

- “ Efekat odgovarajuće poruke zavisi ne samo od same poruke, već i od klase objekta kome se upućuje.
- “ Različite klase mogu imati sasvim različite programske kodove koji se aktiviraju porukom istog imena.
- “ Ova osobina naziva se polimorfizam (polymorphism).
- “ Ovaj koncept se može proširiti i na funkcije, tako da se definicija neke funkcije proširi analognim mogućnostima i za neke sasvim druge vrste podataka.
- “ Jednostavan primer je sabiranje brojeva, koji mogu biti tipa INTEGER ili FLOAT.
- “ U slučaju stringova, analogna operacija sabiranja predstavlja operaciju konkatencije stringova, tj. nadovezivanje stringova.
- “ Izbor operacije koja će se izvršiti pozivom date funkcije, zavisi od tipa argumenata te funkcije, pri čemu programer ne mora da vodi računa o tome, već se izbor vrši automatski.

Radna memorija

- “ Radna memorija kao komponenta ES fizički predstavlja deo RAM memorije računara u kome se u toku rada ES beleže i čuvaju sve poznate činjenice i podaci sa kojima ES u tom trenutku raspolaže.
- “ Sadržaj radne memorije ES u nekom trenutku naziva se kontekst (context) situacije ili problema.
- “ Sadržaj radne memorije je promenljiv, jer se radom ES, tj. zaključivanjem dolazi do novih činjenica, od korisnika se može zahtevati unos potrebnih podataka itd.
- “ Kod ES koji koriste mehanizam zaključivanja unapred, radna memorija sadrži i listu trenutno aktiviranih pravila (agenda).

Radna memorija

- “ Aktivacija pravila kao što je rečeno, zavisi od konteksta.
- “ Redosled aktiviranih pravila zavisi od izabrane strategije rešavanja konflikta.
- “ Izvršavanjem pravila na vrhu liste, može da dođe do promene i konteksta i samim tim i liste aktiviranih pravila.
- “ Pravilo koje se izvršava u THEN delu (desna strana) može da potvrdi (assert) neke nove činjenice, kao i da povuče (retract) neke od postojećih činjenica.
- “ Time se menja kontekst, a samim tim i lista pravila koja odgovara novom kontekstu.

Radna memorija

- “ U radnoj memoriji se čuvaju i vrednosti lokalnih i globalnih promenljivih.
- “ Kod sistema koji koriste okvire i objekte, vrednosti slotova čuvaju se u radnoj memoriji.
- “ Mogućnosti ES da daju objašnjenja dobijenih zaključaka u toku rada, najčešće se bazira na memorisanju redosleda izvršavanja pravila, što se takođe čuva u radnoj memoriji.
- “ Kod sistema sa ulančavanjem unazad, za svaki uslov ispitivanog pravila kome ne odgovara nijedan od podataka u radnoj memoriji traži se da li postoji pravilo u bazi znanja koje ima isti zaključak kao i uslov pravila koje se ispituje.

Radna memorija

- “ Ako takvo pravilo postoji, onda se vrši ulančavanje, dok se u suprotnom traži od korisnika da unese traženi podatak koji se upisuje u radnu memoriju.
- “ Podatak jednom upisan u radnu memoriju, ES može da koristi do kraja procesa zaključivanja, ukoliko se pre toga nekom naredbom ova vrednost ne obriše ili promeni.

Mehanizam za zaključivanje

- “ Mehanizam za zaključivanje (MZ) (Inference Engine) je komponenta ES koja se za razliku od baze znanja i radne memorije u daleko manjoj meri prilagođava konkretnom problemu koji se rešava.
- “ Kod softverskih alata za razvoj ES, kao što su ES Building Tools (ESBT) ili ES Shells, MZ je zamišljen da služi univerzalno za bilo koji konkretan slučaj problematike koja se rešava primenom ES.
- “ Naime, uočeno je u praksi da se ista ili slična pravila za zaključivanje mogu koristiti u velikom mnoštvu konkretnih slučajeva.

Mehanizam za zaključivanje

- “ Kod ES sa pravilima, razmotreni su neki osnovni opšti principi rada mehanizma za zaključivanje pri ulančavanju unapred i unazad.
- “ Zavisno od konkretnog ESBT, mogu postojati razne mogućnosti prilagođavanja mehanizma za zaključivanje konkretnom problemu koji se razmatra.
- “ Pre svega, neki ESBT omogućavaju izbor metoda zaključivanja unapred/unazad.
- “ Kontrola rada mehanizma za zaključivanje kod sistema sa ulančavanjem unapred je pre svega u tome da se može birati jedna od više mogućih strategija razrešavanja konflikta

Mehanizam za zaključivanje

- “ Moguće je zatim bazu znanja podeliti u više nezavisnih segmenata ili modula, od kojih svaki ima svoju nezavisnu listu konfliktnih pravila.
- “ Na taj način, mehanizam za zaključivanje nezavisno radi u svakom modulu čime se postiže strukturiranost baze znanja, posebno kod velikih sistema koji mogu imati više stotina ili hiljada pravila.
- “ Kod sistema sa ulančavanjem unazad, u slučaju kada postoji hijerarhijska struktura ciljeva može se kontrolisati izbor ciljeva tako da se izbor prepusti korisniku, ili da mehanizam za zaključivanje sam sistematski pokušava da dokaže jedan ili više ciljeva.

Mehanizam za zaključivanje

- “ Moguća su i neka druga podešavanja, kao na primer izbor kada će i kakve poruke MZ da ispisuje, proizvoljno postavljanje praga za vrednost faktora pouzdanosti, sistematsko ispitivanje ILI (OR) klase pravila korišćenjem naredbi MULTI i EXHAUSTIVE, itd.
- “ Kod sistema zasnovanih na okvirima i objektima, proces zaključivanja je drugačiji nego kod sistema zasnovanih na pravilima.
- “ Zaključivanje kod ovih sistema je manje očigledno, i odvija se kroz proces nasleđivanja, izvršavanja procedura pridruženih slotovima i klasama objekata koje se iniciraju nekim događajima - demoni, ili sistemom predaje poruka objektima.

RETE algoritam

- “ Kod sistema sa ulančavanjem unapred gde je proces zaključivanja vođen podacima, potrebno je vršiti proveru tj. poređenje leve strane pravila u bazi znanja sa podacima, kada god dođe do bilo kakve promene činjenica u radnoj memoriji.
- “ Pod promenom se podrazumeva dodavanje nove, izmena ili brisanje već postojeće činjenice u radnoj memoriji.
- “ U slučaju velikog broja činjenica u radnoj memoriji, kao i baze znanja sa velikim brojem pravila, ovaj proces može da bude dugotrajan i da znantno usporava ukupni rad ES.

RETE algoritam

- “ Naivan prilaz problemu provere radne memorije i utvrđivanju pravila čija je leva strana zadovoljena činjenicama iz radne memorije, je da se svako pravilo poredi sa svim podacima iz radne memorije, što je naravno sasvim neprihvatljivo.
- “ Pomenuti RETE algoritam se pri pretraživanju radne memorije ograničava samo na promenjene podatke, i na ona pravila iz baze znanja čiji uslovi mogu biti zadovoljeni izmenjenim podacima.
- “ Najpribližniji po performansama RETE algoritmu za prepoznavanje uzoraka je TREAT

RETE algoritam

” Prema tvrđenjima firme Haley Enterprise, RETE je nekoliko puta brži od TREAT algoritma za manji broj pravila, a sa povećanjem broja pravila taj odnos se uvećava u korist RETE algoritma. U odnosu na ostale postojeće algoritme, RETE je brži za redove veličina.

Veza sa korisnikom

- “ Sistem za vezu sa korisnikom (User Interface) omogućava komunikaciju sa ES u raznim fazama razvoja, kao i za razne potrebe programera - inženjera baze znanja ili krajnjeg korisnika.
- “ Mogućnosti koje pruža korisnički interface mogu biti raznovrsne i u velikoj meri zavise od konkretno izabranog alata za razvoj (ESBT).
- “ Da bi ES mogao uspešno da komunicira sa korisnikom, jedan od osnovnih zahteva je da ES osim domenskog znanja o problemu koji se rešava takođe poseduje i neophodno znanje o komunikaciji sa korisnikom.

Veza sa korisnikom

- “ Ovaj zahtev je poznat i pod nazivom “Treći zakon ekspertnih sistema”
- “ Mogućnosti korišćenog alata za razvoj ES I2+, su mnogobrojne, i uslovno se mogu grupisati na sledeći način:
 - “ Rad sa bazom znanja
 - “ Rad sa proceduralnim Pascal-skim programima
 - “ Rad sa bazom podataka
 - “ Programska komunikacija sa korisnikom
- “ Prve tri stavke vezane su za razvojnu fazu komponenata ES (baza znanja, Pascal-ski programi, baza podataka), dok se zadnja četvrta stavka odnosi na komunikaciju u toku rada samog ES.

Veza sa korisnikom

- “ Rad sa bazom znanja obuhvata editovanje samog teksta - izvornog koda baze znanja u file-u tipa PRL (skraćeno od PRoduction Rules), kao i kompilaciju u kod u file-u tipa KNB (skraćeno od KNowledge Base) koji neposredno obrađuje mehanizam za zaključivanje.
- “ U toku rada ES, tj. pretraživanja baze znanja, ES može da ispisuje poruke korisniku, kao i da zahteva unos onih podataka do kojih se na drugi način ne može doći.
- “ Poruke ES mogu se podeliti na poruke o greškama, i na sve ostale poruke koje se odnose na domen baze znanja kada sistem ispravno funkcioniše.

Veza sa korisnikom

- “ Kada se proces zaključivanja ES prekine radi ispisa neke poruke ili se čeka na unos podataka, korisnik može da postavi zahtev za obrazloženje postupka zaključivanja, kao i za obrazloženje zbog čega se zahteva unos podatka od korisnika.
- “ Ovaj vid komunikacije sa korisnikom je obrazložen u sledećem odeljku,

Sistem za obrazlaganje ponašanja i zaključaka

- “ Tokom rada ES najčešće dolazi do niza međuzaključaka koji prethode konačnom rešenju problema.
- “ Korisnik tokom procesa rada ES daje potrebne podatke koje ES od njega traži i koristi u daljem procesu rešavanja problema.
- “ Komunikacija ES sa korisnikom je od značaja kako za rad samog ES, tako i za razumevanje rezultata rada ES od strane korisnika.
- “ U toku rada ES korisnik može da postavi zahtev ili upit (query) ES u cilju dobijanja dodatnog obrazloženja za podatke koje ES traži od korisnika, kao i obrazloženje za zaključke do kojih je ES došao.

Sistem za obrazlaganje ponašanja i zaključaka

- “ U već pomenutom ES MYCIN korisnik može da zahteva od ES odgovore na klasična pitanja “kako” i “zašto”.
- “ Pitanja tipa “kako” odnose se na objašnjenje na koji je način ES došao do odgovarajućeg zaključka, dok su pitanja tipa “zašto” uzvratna pitanja ES, i odnose se na objašnjenje zbog čega ES traži od korisnika neki podatak ili zbog čega je neka druga informacija potrebna.
- “ Odgovori na ova pitanja mogu se dobiti na osnovu razmatranja traga procesa zaključivanja (reasoning trace) i konteksta - trenutnog stanja radne memorije.

Sistem za obrazlaganje ponašanja i zaključaka

- “ Kao odgovor na pitanje tipa “kako”, ES na pr. u slučaju korišćenog sistema I2+ sa ulančavanjem unazad, može da da hronološki prikaz niza dobijenih zaključaka sve do tekućeg pravila koje se razmatra, ili hronološki niz podataka koje je uneo korisnik i na osnovu toga dobijenih zaključaka.
- “ Takođe, u toku rada je moguć prikaz kompletne baze znanja sa ulančanim pravilima (AND / OR tree).
- “ Na pitanje “zašto” odgovor se može dobiti prikazom tekućeg pravila koje ES razmatra.

Sistem za obrazlaganje ponašanja i zaključaka

- “ Moguć je i kompletan pregled tekućeg konteksta baze znanja, gde se može videti pouzdanost (certainty factor, confidence) i vrednost svake činjenice (fact) i promenljive.
- “ U tom pregledu, podaci su grupisani po tipovima - proste činjenice (simple facts), numerički (numeric), string i složene objekat - atribut.
- “ Osim toga, za svaki podatak je dato i poreklo označeno odgovarajućim slovom

Sistem za obrazlaganje ponašanja i zaključaka

- “ A (Answered by User) - uneseno od strane korisnika
- “ D (Denied by User) - negirano od strane korisnika
- “ E (External Program) - Podaci iz spoljašnjeg programa
- “ I (Initialized) - inicijalizovani podaci
- “ C (Calculated) - podaci izračunati na osnovu drugih podataka
- “ U (Unknown to User) - nepoznato korisniku
- “ R (Rule concluded) - zaključak pravila

Sistem za obrazlaganje ponašanja i zaključaka

- “ Istraživanja u ovoj oblasti usmerena su kako na unapređivanje ovih osnovnih mogućnosti, tako i na razvijanje složenih sistema koji će moći da daju objašnjenja usaglašena sa kontekstom
- “ Prema dosadašnjim istraživanjima, kontekst je određen sa najmanje pet sledećih komponenata:
 - “ Problemska situacija
 - “ Subjekti učesnici
 - “ Način komunikacije sa sistemom
 - “ Tekući dijalog korisnika i sistema
 - “ Spoljašnje okruženje

Prikupljanje znanja

- “ Prikupljanje znanja (Knowledge Acquisition - KA) je suštinski važna i nezaobilazna faza u razvoju svakog ES.
- “ Usled raznih teškoća i problema koji mogu da prate ovu fazu, često se smatra da ova faza predstavlja usko grlo u razvoju ES.
- “ Pri tome se podrazumeva da u razvoju ES učestvuju najmanje dve osobe ili grupe ljudi, i to ekspert - stručnjak za datu oblast iz koje je i problem koji se rešava, i programer - inženjer baze znanja koji iskustvo i znanje eksperta ugrađuje u bazu znanja ES, koristeći pritom jednu ili više metoda za predstavljanje znanja.

Prikupljanje znanja

- “ Osnovni uzroci teškoća kod prikupljanja znanja su sledeći:
- “ Ljudsko znanje je vrlo složeno i može biti nedovoljno sistematizovano ili loše formulisano
- “ Ljudi često imaju problema da na jasan način izraze znanje koje imaju, kao i na koji način ga koriste za rešavanje problema
- “ Ako neko sve više postaje stručnjak u datoj oblasti, sve manje je svestan složenih misaonih procesa koji vode do rešenja, pa samim tim ima i teškoća u egzaktnom izražavanju načina na koji je došao do rešenja, a što je neophodno radi ugrađivanja u bazu znanja

Prikupljanje znanja

- “ Znanje koje se od eksperta dobija najčešće ne samo da nije formulisano u formi pogodnoj za bazu znanja, već ga je potrebno i eksplicitno izraziti
- “ Naravno, moguće je da ekspert i programer budu jedna ista osoba, u kom slučaju se može reći da su navedeni problemi manje izraženi, ali u suštini pomenuti problemi ostaju i dalje.
- “ Ovde treba dodati i to da zahtevi razvoja većih ekspertnih sistema najčešće prevazilaze mogućnosti samo jednog čoveka.

Prikupljanje znanja

- “ Za sam proces dobijanja, izvlačenja, crpljenja znanja od stručnjaka (Knowledge Elicitation), može se reći da predstavlja akviziciju znanja u užem smislu. Akvizicija znanja u širem smislu pored toga obuhvata i tačno definisanje uloge, zadataka i strukture ES, kao i neizbežno testiranje, otklanjanje grešaka i usavršavanje postojećeg znanja.
- “ Pokušaji za rešavanje problema akvizicije znanja kao uskog grla usmereni su na usavršavanje klasičnih tehnika i razvijanje sistema za mašinsko - automatsko učenje.

Prikupljanje znanja

- “ Klasične tehnike akvizicije znanja obuhvataju razgovor sa stručnjakom, posmatranje stručnjaka na delu kada rešava nove slučajeve kao i već rešene, audio - vizuelno snimanje u cilju kasnije detaljne analize i sl.
- “ Sistemi za mašinsko učenje se uglavnom zasnivaju na analizi većeg broja ulaznih podataka za već rešene probleme.
- “ Zadatak sistema je da pronađe pravila kojima se eksplicitno izražava znanje koje kao implicitno postoji u podacima test primera.

Prikupljanje znanja

- “ Ovakvi sistemi vrše induktivno zaključivanje pri nalaženju novih pravila.
- “ Razvijeno je više ovakvih sistema u raznim oblastima primene.
- “ Jedan od vrlo interesantnih primera za automatsko učenje je sistem RULES koji primenom jednostavnog algoritma za induktivno učenje analizira dati skup test primera, i nalazi odgovarajući skup pravila.
- “ Svako nađeno pravilo se ispituje da li sadrži irelevantne uslove, tj. uslove koji su redundantni za dati skup test primera.

Prikupljanje znanja

- “ Sistem nije zavisn od problemske oblasti, i testiran je na primerima iz različitih oblasti.
- “ Nađena pravila omogućavaju rešavanje problema tipa klasifikacije.
- “ RULEARN[®] je sistem za automatsko učenje koji je razvijen za potrebe procene brzine korozije kod metala
- “ Ovaj sistem nalazi pravila koja izražavaju zavisnost brzine korozije od uslova u kojima se metali nalaze - temperatura, koncentracija kiseline, procentualni sadržaj silicijuma u metalu.
- “ Pravila nađena sistemom za induktivno učenje se koriste u ES za rešavanje novih problema.

Predstavljanje nepouzdanosti

- “ Pored svog neprikosnovenog informativnog sadržaja, karakteristika ljudskog znanja je i nepouzdanost ili nesigurnost. Izvori nepouzdanosti u znanju, samim tim i kod ES mogu se podeliti na nepouzdanosti kod podataka, nepouzdanosti u samom pojednostavljenom problemskom modelu predstavljenom u bazi znanja, kao i na nepouzdanosti kod uverenja.
- “ Osnovni izvori nepouzdanosti kod podataka ili činjenica su greške merenja, šum, nepotpunost, ili poznavanje sa određenom verovatnoćom.

Predstavljanje nepouzdanosti

- “ Domensko znanje sadržano u problemskom modelu može biti nepotpuno, aproksimativno, neadekvatno i ograničeno samim modelom.
- “ Uverenja ugrađena u bazu znanja mogu biti nedovoljno konzistentna ili nedovoljno opravdana.
- “ Pošto se nabrojane nepouzdanosti ne mogu izbeći, potrebno je da se pre svega uoče tamo gde postoje i da se adekvatno predstave u okviru ES.
- “ S obzirom na raznovrsnost mogućih nepouzdanosti, postoje i razni pristupi ovom problemu.
- “ Sve pristupe bi mogli podeliti na one kod kojih se sama nepouzdanost eksplicitno ne predstavlja, kao i na one sa eksplicitnim predstavljanjem.

Predstavljanje nepouzdanosti

- “ U prvu grupu između ostalih spada i takav pristup kod koga se sama strategija rešavanja problema prilagođava tako da se što je u mogućoj većoj meri smanji uticaj nepouzdanosti.
- “ U prvu grupu između ostalih spada i takav pristup kod koga se sama strategija rešavanja problema prilagođava tako da se što je u mogućoj većoj meri smanji uticaj nepouzdanosti.
- “ Neke od metoda za eksplicitno predstavljanje nepouzdanosti kod ES su račun verovatnoće, faktori izvesnosti (certainty factors, confidence), fuzzy logika, koji će u sledećim odeljcima biti ukratko razmotreni.

Verovatnoća

- ” Klasičan način za predstavljanje nepouzdanosti je račun verovatnoće. U slučaju na pr. produkcionih pravila koja povezuju zaključak pravila sa n uslova, za izračunavanje verovatnoće zaključka z , kada su poznate pojedinačne verovatnoće uslova u_i , može da se primeni izraz za uslovnu verovatnoću:

$$P(z | u_1, u_2, u_3, \dots, u_n) = \frac{P(z, u_1, u_2, u_3, \dots, u_n)}{P(u_1, u_2, u_3, \dots, u_n)}$$

- ” Neposredna primena gornjeg izraza zahteva poznavanje združenih verovatnoća u razlomku, što najčešće nije slučaj, pogotovo ako su uslovi u_i zavisni.

Verovatnoća

” Može se napisati i obrnuto:

$$P(u_1, u_2, u_3, \dots, u_n | z) = \frac{P(z, u_1, u_2, u_3, \dots, u_n)}{P(z)}$$

” Smenom u prethodni izraz, dobija se Bayes-ova formula koja je pogodnija za primenu.

$$P(z | u_1, u_2, u_3, \dots, u_n) = \frac{P(u_1, u_2, u_3, \dots, u_n | z)P(z)}{P(u_1, u_2, u_3, \dots, u_n)}$$

” Uslovna verovatnoća za premise u prisustvu zaključka u brojiocu zadnjeg izraza, lakša je za procenu, i može se najlakše proceniti na osnovu prakse.

” Dalje uprozavanje je moguće, ukoliko se može pretpostaviti da su ovi uslovi nezavisni.

Faktori pouzdanosti

- “ Faktori pouzdanosti predstavljaju nešto drugačiji i jednostavniji način za tretman nepouzdanosti, nego što je to slučaj sa korišćenjem računa verovatnoće.
- “ Osnovni problem kod korišćenja verovatnoće je izračunavanje združene verovatnoće međusobno povezanih uslova, zbog čega se i koristi ponekad neadekvatna aproksimacija da su uslovi nezavisni.
- “ Takođe se postavlja pitanje i koliko su procene verovatnoća subjektivne, pa samim tim i opravdanost primene računa verovatnoće.
- “ Faktorima pouzdanosti se kvantitativno izražava mera sigurnosti uslova - premisa ili zaključka i to u okviru nekog intervala brojeva, najčešće od 0 do 100, mada može biti i neki drugi interval.

Faktori pouzdanosti

- “ Minimalna vrednost intervala, na pr. 0 znači da je uslov ili zaključak pravila sigurno neistinit, dok maksimalna vrednost (100) znači potpunu istinitost.
- “ Ostale vrednosti između ekstremnih označavaju proporcionalnu meru pouzdanosti, što je vrednost veća, veća je i pouzdanost.
- “ Dodeljeni faktor pouzdanosti uslova ima zadovoljavajuću vrednost ukoliko je veći od vrednosti praga pouzdanosti (Threshold).
- “ U slučaju da je vrednost dodeljenog faktora pouzdanosti manja od praga, uslov se smatra neistinitim.

Faktori pouzdanosti

- “ Ako pravilo ima više uslova u_i sa različitim faktorima pouzdanosti CF_i , rezultujuća pouzdanost se određuje u zavisnosti od logičkih operacija koje povezuju uslove.
- “ Rezultujuća pouzdanost više uslova povezanih operacijom AND jednaka je faktoru pouzdanosti sa minimalnom vrednošću, dok je u slučaju kada su uslovi povezani operacijama OR, rezultujuća pouzdanost jednaka faktoru pouzdanosti sa maksimalnom vrednošću.

Faktori pouzdanosti

- “ Negacija uslova u_i sa NOT menja pripadajuću dodeljenu pouzdanost CF_i u CF_i' kao

$$CF_i' = 100 - CF_i$$

- “ Pouzdanost zaklju ka pravila CF_p izra unava se kao proizvod rezultuju eg faktora pouzdanosti uslova CF_{rez} i dodeljenog faktora pouzdanosti zaklju ka CF_z .

$$CF_p = CF_{rez} CF_z / 100$$

- “ vredost se deli sa sto, da bi se rezultat doveo u interval 0 - 100.
- “ U slu aju da vrednost nekog podatka korisniku nije poznata, podatak se evidentira kao nepoznat (Unknown).

Faktori pouzdanosti

- “ Uslov čija se istinitost ne može odrediti zbog nepoznatog podatka takođe se tretira kao nepoznat.
- “ Ukoliko pored nepoznatog uslova postoji još neki uslov pravila sa faktorom pouzdanosti većim od praga, zaključak pravila se može izvesti.
- “ Faktori pouzdanosti dolaze do izražaja i kada u bazi znanja postoji više pravila sa istim zaljučkom.
- “ Mehanizam zaključivanja u tom slučaju daje prioritet pravilu čiji zaključak ima najveći faktor pouzdanosti.