

UNIVERSITATEA „BABEȘ-BOLYAI” DIN CLUJ NAPOCA
Facultatea de Științe Economice și Gestiunea Afacerilor

**CONTRIBUȚII LA UTILIZAREA
SISTEMELOR DE ASISTARE A DECIZIILOR
BAZATE PE TEHNOLOGII „MACHINE LEARNING”
ÎN PROBLEME DE AFACERI**

- rezumatul tezei de doctorat -

Conducător științific:

Prof. dr. Ioan Ștefan NIȚCHI

Doctorand:

Cristina Ofelia SOFRAN (căs. STANCIU)

Cluj Napoca
2010

CUPRINS

| | |
|---|-----------|
| LISTA DE FIGURI ȘI LISTA DE TABELE..... | 7 |
| INTRODUCERE | 10 |
| | |
| CAPITOLUL I. PROCESUL DECIZIONAL | 15 |
| 1.1. SISTEMUL INFORMAȚIONAL – DECIZIONAL, COMPONENTĂ MANAGERIALĂ..... | 15 |
| 1.2. DECIZIA – DEFINIRE, CLASIFICARE, CARACTERISTICI..... | 20 |
| 1.3. STABILIREA ȘI DESCOMPUNEREA PROBLEMELOR DECIZIONALE | 24 |
| 1.4. ADOPTAREA DECIZIILOR..... | 26 |
| 1.5. MODELAREA PROCESELOR DECIZIONALE..... | 28 |
| 1.6. EVALUAREA PROCESELOR DECIZIONALE | 31 |
| 1.7. LIMITE ȘI PERSEPECTIVE ALE SISTEMULUI INFORMAȚIONAL - DECIZIONAL..... | 32 |
| 1.8. CONCLUZII ȘI CONTRIBUȚII PROPRII | 34 |
| | |
| CAPITOLUL II. INTELIGENȚA ARTIFICIALĂ ȘI SISTEMELE INTELIGENTE. IMPLICAȚIILE TEHNOLOGIILOR MACHINE LEARNING ÎN ASISTAREA DECIZIILOR..... | 36 |
| 2.1. CONCEPTUL DE SISTEM INTELIGENT | 36 |
| 2.2. TIPURI DE SISTEME INTELIGENTE..... | 39 |
| 2.3. SISTEME EXPERT. PERSPECTIVELE SISTEMELOR EXPERT SI ALE INTELIGENȚEI ARTIFICIALE..... | 42 |
| 2.4. CONSIDERAȚII ASUPRA CONCEPTULUI BUSINESS INTELLIGENCE..... | 47 |
| 2.4.1. Factori care influențează BI | 49 |
| 2.4.2. Tipuri de aplicații de BI | 50 |
| 2.5. CONCEPTUL MACHINE LEARNING..... | 53 |
| 2.6. ÎNVĂȚAREA REALIZATĂ DE SUBIECTUL UMAN ȘI ÎNVĂȚAREA REALIZATĂ DE AGENTUL DE ÎNVĂȚARE | 56 |
| 2.7. TEHNOLOGII MACHINE LEARNING | 58 |
| 2.7.1. Arbori de decizie | 58 |
| 2.7.2. Tehnologia Reinforcement Learning..... | 65 |
| 2.7.3. Învățarea bazată pe exemple..... | 70 |
| 2.7.4. Învățarea deductivă și învățarea inductivă | 72 |
| 2.7.5. Învățarea supervizată și învățarea nesupervizată..... | 72 |
| 2.8. APLICAȚII ALE MACHINE LEARNING | 73 |
| 2.9. CONCLUZII ȘI CONTRIBUȚII PROPRII | 77 |
| | |
| CAPITOLUL III. ELEMENTE DE PROIECTARE A SISTEMELOR DE ASISTARE A DECIZIILOR..... | 78 |
| 3.1. SISTEME INFORMATICE DE ASISTARE A DECIZIILOR | 78 |
| 3.1.1. Considerații teoretice cu privire la sistemele informatice de asistare a deciziilor.. | 78 |
| 3.1.2. Clasificări ale sistemelor informatice de asistare a deciziilor | 82 |
| 3.1.3. Componentele sistemului informatic de asistare a deciziilor | 86 |
| 3.1.4. Dezvoltarea sistemelor informatice de asistare a deciziilor | 90 |
| 3.2. SOLUȚII DE DEPOZITARE ȘI EXPLOATARE A DATELOR ÎN CADRUL SISTEMELOR DE ASISTARE A DECIZIILOR | 91 |

| | |
|--|-----|
| 3.2.1. Depozite de date și tehnologii OLAP..... | 92 |
| 3.2.2. Data Mining – tehnologii avansate de procesare a datelor..... | 98 |
| 3.3. ETAPE ÎN DEZVOLTAREA SISTEMELOR SOFTWARE | 104 |
| 3.4. PROIECTAREA ORIENTATĂ PE OBIECTE..... | 107 |
| 3.5. PRINCIPII DE PROIECTAREA INTERFEȚEI CU UTILIZATORUL | 109 |
| 3.6. ARHITECTURA SISTEMELOR DE ASISTARE A DECIZIILOR..... | 113 |
| 3.6.1. Arhitectura în rețea..... | 113 |
| 3.6.2. Arhitectura centralizată | 114 |
| 3.6.3. Arhitectura ierarhizată..... | 114 |
| 3.7. ARHITECTURI DE SISTEME DISTRIBUITE | 116 |
| 3.7.1. Arhitectura multiprocesor..... | 117 |
| 3.7.2. Arhitectura client-server..... | 117 |
| 3.7.3. Arhitectura de obiecte distribuite | 120 |
| 3.8. SISTEME DE ASISTARE A DECIZIILOR DE GRUP..... | 121 |
| 3.9. CONCLUZII ȘI CONTRIBUȚII PROPRII | 124 |

CAPITOLUL IV. STRATEGIA DE DEZVOLTARE ȘI IMPLEMENTARE A UNUI SISTEM DE ASISTARE A DECIZIILOR126

| | |
|---|-----|
| 4.1. ABORDAREA OBIECTUALĂ ÎN DEZVOLTAREA SISTEMELOR DE ASISTARE A DECIZIILOR | 126 |
| 4.2. LIMBAJE DE PROGRAMARE PENTRU DEZVOLTAREA SISTEMELOR DE ASISTARE A DECIZIILOR..... | 128 |
| 4.3. SELECTAREA UNEI STRATEGII DE IMPLEMENTARE | 131 |
| 4.4. JUSTIFICAREA SOLUȚIEI DE REPREZENTARE A DATELOR PRIN INTERMEDIUL FORMATULUI XML..... | 134 |
| 4.4.1. Considerații asupra setului de date utilizat..... | 134 |
| 4.4.2. Formatul XML pentru reprezentarea datelor..... | 136 |
| 4.4.3. Analogia dintre formatul XML propus și fișierele arff din sistemul WEKA | 139 |
| 4.5. STRUCTURA FIȘIERELOR XML ÎN CADRUL SISTEMULUI PROPUȘ..... | 142 |
| 4.6. CONCLUZII ȘI CONTRIBUȚII PROPRII | 144 |

CAPITOLUL V. SOLUȚII DE DEZVOLTARE, IMPLEMENTARE ȘI INTEGRARE A ALGORITMILOR DE DECIZIE ÎN CADRUL SISTEMELOR DE ASISTARE A DECIZIILOR.....146

| | |
|---|-----|
| 5.1. METODE DE MANEVRARE A FIȘIERELOR XML..... | 146 |
| 5.1.1. Încărcarea și salvarea fișierelor XML | 146 |
| 5.1.2. Operații de intrare/ieșire cu fișierul XML | 147 |
| 5.2. STRUCTURAREA BIBLIOTECILOR DE CLASE DIN CADRUL SISTEMULUI | 151 |
| 5.2.1. Reprezentarea entităților XML în cadrul bibliotecii de clase..... | 151 |
| 5.2.2. Modelul top-down în biblioteca de clase | 152 |
| 5.2.3. Descrierea claselor implementate..... | 155 |
| 5.2.4. Crearea definițiilor de atribute. Șablonul Factory | 159 |
| 5.3. IMPLEMENTAREA ALGORITMILOR DE ÎNVĂȚARE..... | 163 |
| 5.3.1. Definirea claselor abstracte pentru algoritmi | 164 |
| 5.3.2. Algoritmi pentru arbori de decizie | 167 |
| 5.3.3. Algoritmi pentru învățarea bazată pe exemple. K- Nearest Neighbour | 174 |
| 5.4. INTERFAȚA GRAFICĂ A SISTEMULUI | 180 |
| 5.5. CONCLUZII ȘI CONTRIBUȚII PROPRII | 183 |

| | |
|--|------------|
| CAPITOLUL VI. VERIFICAREA, VALIDAREA ȘI TESTAREA SISTEMULUI DE ASISTARE A DECIZIILOR..... | 185 |
| 6.1. VERIFICAREA ȘI VALIDAREA SISTEMELOR SOFTWARE..... | 185 |
| 6.2. NECESITATEA TESTĂRII SISTEMULUI DE ASISTARE A DECIZIILOR..... | 187 |
| 6.3. TESTAREA APLICAȚIEI .NET..... | 189 |
| 6.3.1. Practici de testare | 189 |
| 6.3.2. Testarea bibliotecii de clase | 191 |
| 6.3.3. Testarea efectivă..... | 194 |
| 6.3.4. Testarea claselor de date decizionale | 198 |
| 6.4. CONCLUZII ȘI CONTRIBUȚII PROPRII | 201 |
| | |
| CONCLUZII ȘI PROPUNERI | 202 |
| REFERINȚE BIBLIOGRAFICE | 208 |
| ANEXE | 216 |
| ANEXA 1. CLASE DIN <i>ML.LIB.DATA</i> | 216 |
| ANEXA 2. CLASE DIN <i>ML.LIB.ALGORITHM</i> | 227 |

INTRODUCERE

Succesul oricărei organizații depinde în mare măsură de deciziile celor aflați la diferitele nivele de conducere, iar în condițiile economico-sociale actuale luarea deciziilor potrivite a devenit o sarcină dificilă. Dacă inițial procesul decizional era considerat în teoria managementului o activitate exclusiv umană, în urma creșterii cantității de date și a dificultății gestionării și procesării acestora, procesul decizional se desfășoară cu sprijinul tehnologiilor informatice și de comunicație. Acest fapt a condus la abordări interdisciplinare în studiul deciziilor, la informatizarea deciziei și la apariția sistemelor informatice de asistare a deciziilor.

Sistemele de asistare a deciziilor au rol de sprijin al managerului în abordarea de către acesta a mediului complex și a situației cu privire la care urmează să fie adoptată o decizie. Un pas important în acceptarea acestor sisteme, care a dus implicit la dezvoltarea acestora în timp, a fost înțelegerea și acceptarea ideii că aceste instrumente nu urmăresc înlocuirea decidentului, ci sprijinul și asistarea acestuia.

În condițiile dezvoltării societății cunoașterii și a organizațiilor bazate pe cunoaștere, a spirit și complexitatea problemelor manageriale, iar sistemele informatice de asistare a deciziilor au început să fie realizate utilizând soluții ce au la bază tehnologii ale inteligenței artificiale. Scopul abordării tehnologiilor inteligente în realizarea sistemelor de asistare a deciziilor a fost utilizarea raționamentului uman atât pentru a conduce la soluții pentru probleme decizionale complexe, cât și pentru a determina un plus de cunoaștere în cadrul organizației, sporind calitatea procesului managerial.

Lucrarea de față urmărește să realizeze o investigație asupra mediului decizional și a sistemelor informatice de asistare a deciziilor, a sistemelor inteligente și tehnologiilor Machine Learning, și de asemenea asupra modalităților de utilizare ale acestor tehnologii pentru a îmbunătăți procesul decizional. Studiul teoretic a determinat realizarea unui software decizional având la bază o bibliotecă de clase reprezintă o fundație solidă pentru construirea unor algoritmi de decizie complecși. Astfel a fost clădit un mediu expandabil și reutilizabil, cu caracter practic, ce permite utilizarea algoritmilor cu scop în asistarea decidenților în rezolvarea unor probleme decizionale de mare complexitate.

Lucrarea este structurată pe șase capitole (figura i), în prima parte insistând pe prezentarea teoretică a noțiunilor de bază și a conceptelor utilizate, apoi continuând cu descrierea modalității de realizare a sistemului propus.

În cadrul societății contemporane, denumită societatea cunoașterii, organizațiile trebuie să-și adapteze managementul unei gândiri proactive, să accepte ideea de raționalizare a componentei informaționale și decizionale, cu scopul de a deveni din ce în ce mai competitive și performante. O infrastructură informatică bine pusă la punct, împreună cu sistemele informatice de asistare a deciziilor, constituie un sprijin incontestabil al decidenților care se confruntă cu o multitudine de probleme în cadrul unei organizații.

Lucrarea se dorește a fi o abordare nouă în ceea ce privește dezvoltarea sistemelor de asistare a deciziilor, evidențiind noi modalități de implementare a algoritmilor de decizie, prin intermediul unui sistem expandabil care a fost implementat respectând principii avansate de ingineria programării. Soluția abordată este un răspuns la trendurile impuse de nivelul actual al tehnologiilor software și permite implementarea unor sisteme de asistare a deciziilor utilizând cele mai noi tehnologii Machine Learning.

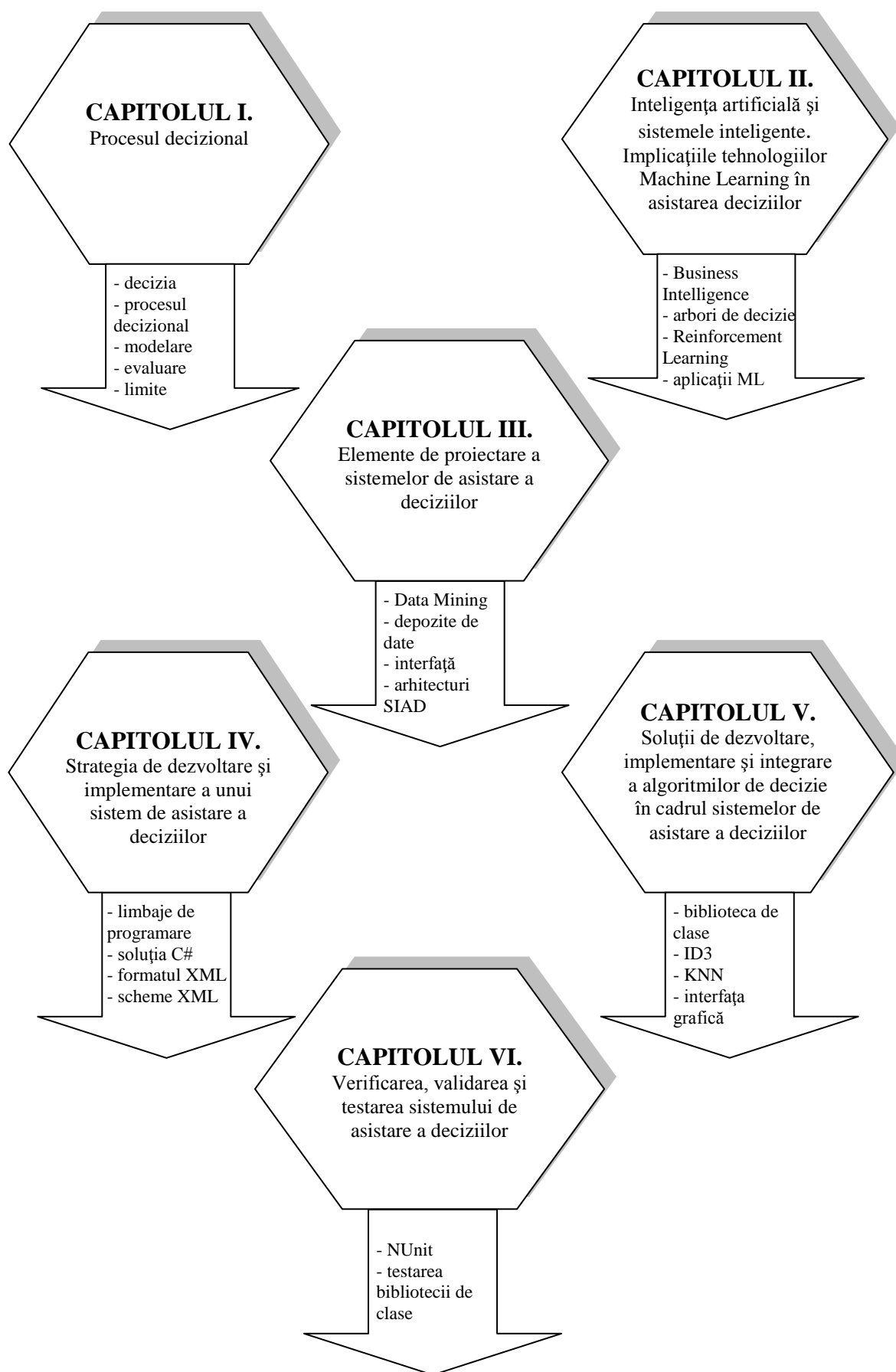


Figura i. Structura tezei

CAPITOLUL I. PROCESUL DECIZIONAL

În cadrul Capitolului I. sunt descrise elementele procesului decizional, arătând relația dintre sistemul informațional și cel decizional, care în urma modificărilor complexe pe care societatea le-a suferit datorită faptului că informația a devenit resursa esențială a organizației, au evoluat și ele în mod firesc. În scopul descrierii complete a sistemului decizional, se începe cu detalierea noțiunii de decizie, arătând caracteristicile și clasificarea deciziilor după o serie de criterii, definirea decidentului și a situației decizionale. Este stabilită structura procesului decizional, aceasta fiind formată, în viziunea noastră, din patru etape: pregătirea deciziei, alegerea soluției optime și luarea deciziei, implementarea deciziei și controlul aplicării, evaluarea rezultatelor deciziei. Alegerea soluției optime și adoptarea deciziei fiind etapa cea mai importantă a procesului decizional, arătăm principalele tehnici și metode de căutare a deciziei optime, și descriem procedeul de modelare a proceselor decizionale. Pentru ca procesul decizional să își dovedească finalitatea, rezultatele acestuia se supun evaluării, stabilindu-se astfel calitatea întregului proces.

Sfârșitul secolului XX a stat sub semnul tranziției, de la societatea industrială, în cadrul căreia principala resursă a unei întreprinderi se identifica prin capital, la societatea post industrială, cunoscută și sub denumirea de societate informațională, în cadrul căreia resursa cea mai de preț a devenit informația, iar producția de bază este cea intelectuală [Nițchi03]. Această societate informațională în care viețuim este adânc influențată de dezvoltarea tehnologiilor informaționale și de comunicație, care determină o modificare substanțială a procesului de informare, proces axat în jurul noțiunii de informație, și care determină definirea structurală și funcțională a sistemului informațional.

Asemenea oricărei alte activități întreprinse de un subiect uman, managementul implică o serie de decizii, care se vor clasifica în funcție de diverse criterii. Decizia poate fi definită ca o consecință a unor activități conștiente de alegere a unei direcții de acțiune și a angajării în aceasta, ceea ce implică și alocarea de resurse pentru atingerea scopului vizat. Luarea unei decizii este justificată prin următoarele elemente: decidentul, situația și problema de decizie, procesul decizional.

Procesul decizional ia naștere dintr-o serie de activități decizionale. Acest proces are la bază conștientizarea unei situații decizionale și culegerea datelor, apoi continuă cu

căutarea, proiectarea și modelarea alternativelor, alegerea soluției propice, iar în final se va adopta decizia și se vor evalua rezultatele respectivei direcții de acțiune.

Procesul decizional se desfășoară pe parcursul a mai multor etape și faze, iar structura procesului decizional rezultă din numărul și ordinea de succesiune a etapelor. Procesul decizional se desfășoară pe parcursul a patru etape principale, fiecare etapă cuprinzând mai multe faze (Figura 1.).

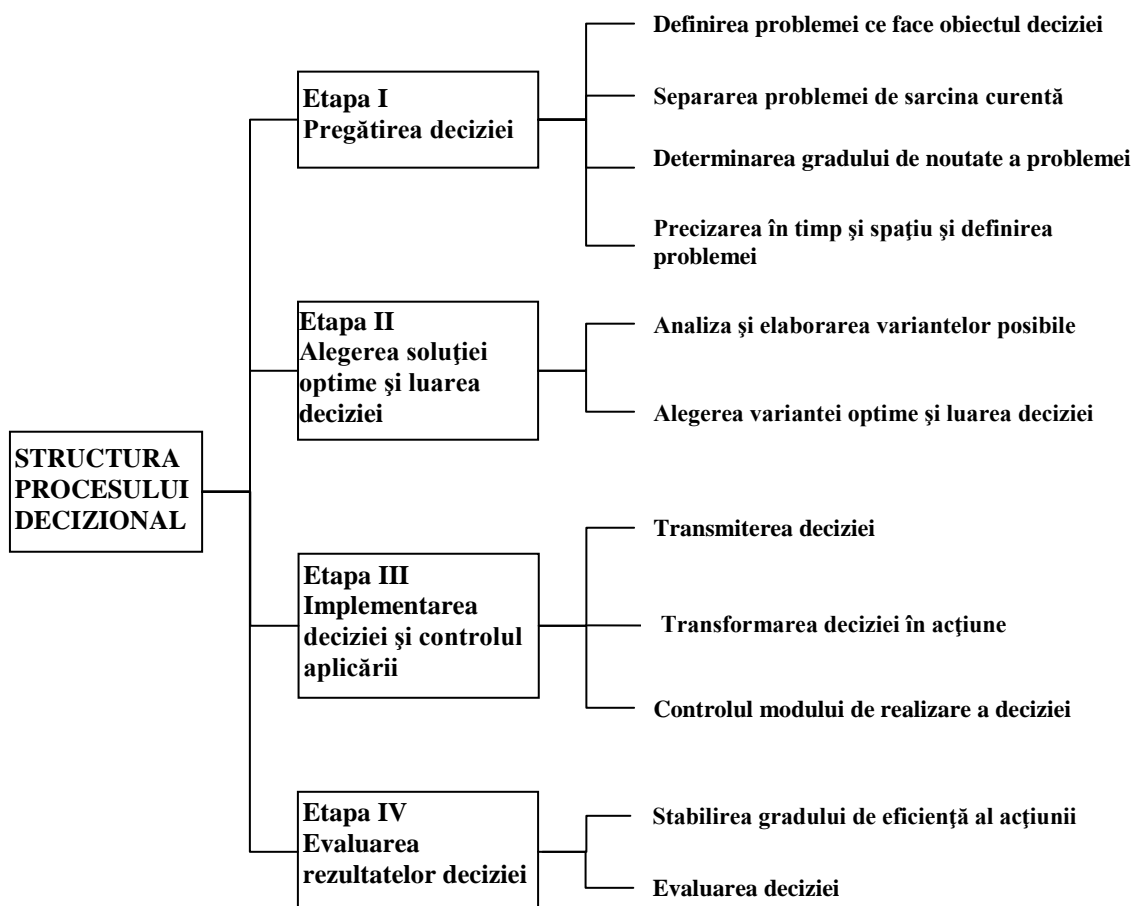


Figura 1. Structura procesului decizional

Activitatea de modelare a procesului decizional se va realiza ținând cont de fiecare etapă din structura procesului decizional (Figura 1.). Astfel, pentru fiecare activitate decizională se vor construi unul sau mai multe modele, luând în considerare modalitățile de desfășurare ale acestor activități. Trebuie menționat că procesul de modelare tinde să fie limitat, imperfect, deoarece în ciuda elaborării cu maximă atenție a modelului, acesta nu va reflecta întocmai obiectul modelat, prin urmare poate afecta și calitatea perspectivei asupra procesului decizional.

CAPITOLUL II. INTELIGENȚA ARTIFICIALĂ ȘI SISTEMELE INTELIGENTE. IMPLICAȚIILE TEHNOLOGIILOR MACHINE LEARNING ÎN ASISTAREA DECIZIILOR

Capitolul II prezintă la început concepte legate de inteligența artificială, insistând pe conceptul de sistem inteligent, implicațiile și utilizarea acestor sisteme inteligente în domeniul economic. Sunt prezentate tipurile de sisteme inteligente – sistemele neuronale artificiale, sisteme inteligente bazate pe algoritmi genetici, sisteme fuzzy, sisteme expert, sisteme hibride – insistând apoi pe o prezentare detaliată a sistemelor expert. În contextul discuției asupra sistemelor inteligente și a utilizării acestora la scară tot mai largă în domeniul economic se ajunge la conceptul Business Intelligence, ale cărui aplicații sunt prezentate. Introducerea conceptului Machine Learning arată că există un efort continuu pentru realizarea sistemelor informatice care își îmbunătățesc performanțele prin experiența dobândită, așa-numita activitate de „învățare” a acestora realizată de către un agent de învățare fiind similară că învățarea realizată de subiectul uman. În prezentarea tehnologiilor Machine Learning se insistă pe arborii de decizie, tehnologia Reinforcement Learning, învățarea bazată pe exemple, învățarea deductivă și învățarea inductivă, învățarea supervizată și învățarea nesupervizată, acestea permițând înțelegerea corectă a acestui concept.

Unele sisteme bazate pe tehnologii Machine Learning tind să elimine necesitatea intuiției umane în cadrul analizei datelor, în timp ce altele adoptă o abordare colaborativă între om și mașină. Intuiția umană nu poate fi complet eliminată din moment ce proiectantul sistemului trebuie să specifice modul de reprezentare al datelor și ce mecanism va fi utilizat pentru căutarea lor. Machine Learning se aseamănă cu o tentativă de automatizare a părților unei metode științifice.

Machine Learning face referire la modificările din cadrul sistemelor ce execută diferite sarcini care au legătură cu domeniul inteligenței artificiale, sarcini ce implică recunoaștere, diagnostic, planificare, controlul roboților, previziune, care nu pot fi definite complet decât prin intermediul exemplelor, specificând datele de intrare și rezultatele așteptate. Este de dorit ca rezultatele să poată fi deduse în ipoteza în care există o serie de date de intrare, dar fără a avea o funcție intrare-ieșire bine definită, ci doar prin aproximarea relațiilor implicite. De multe ori însă, corelațiile și legăturile se

află „ascunse” în cadrul cantităților uriașe de date, dar prin intermediul tehnologiilor Machine Learning acestea pot fi extrase.

Deseori sunt proiectate sisteme care nu funcționează eficient în mediul în care sunt utilizate, deoarece anumite caracteristici ale modului de lucru nu au putut fi clar definite la momentul construirii lor, dar metodele Machine Learning vin în sprijinul acestora. Informațiile se diversifică și generează noi fluxuri de cunoștințe, ceea ce ar necesita reimplementarea sistemelor de inteligență artificială, dar având în vedere că aceasta nu este o soluție practică, se preconizează faptul că tehnologiile Machine Learning ar face față cu succes acestor situații.

Domeniul Machine Learning caută să răspundă la întrebarea „cum se pot realiza sisteme informatice care își îmbunătățesc performanțele prin experiență, și care sunt legile fundamentale care guvernează toate procesele de învățare?”. Pentru a deschide calea către răspunsul la această întrebare, în cele ce urmează vom prezenta o analogie între învățarea realizată umană și cea a unui agent de învățare, și vom descrie câteva tehnologii uzuale din acest domeniu.

Una dintre tehnologiile Machine Learning de mare importanță o reprezintă arborii de decizie. Arborii de decizie sunt o tehnică aplicabilă atât pentru clasificare cât și pentru predicție, rezultatul luând forma unei arborescențe care prezintă o ierarhie de reguli logice stabilite automat prin explorarea unei baze de exemple. Exemplele au forma unor înregistrări compuse din mai multe atribute. Regulile se obțin ca efect al subdivizării din ce în ce mai detaliate a ansamblului exemplilor, în funcție de conținutul atributelor.

Arborii de decizie se dovedesc a fi instrumente excelente pentru realizarea de decizii financiare sau legate de numere, în cazurile în care o cantitate mare de informație complexă trebuie să fie luată în calcul. Aceștia oferă o structură eficientă prin care deciziile alternative și implicațiile alegerii acestora pot fi evaluate, și ajută la formarea unei viziuni corecte, echilibrate a riscurilor și recompenselor care pot rezulta în urma unei anumite alegeri.

Majoritatea algoritmilor dezvoltați pentru arbori de decizie utilizați cu succes în probleme de învățare sunt variațiuni ale aceluiași algoritm care implică o căutare descendentă în spațiul de arbori de decizie posibili. Un exemplu în acest sens este algoritmul ID3, realizat în 1986 de către profesorul australian J.R. Quinlan, și succesorul acestui algoritm, C4.5.

Reinforcement Learning se referă la un tip de algoritm Machine Learning. Ideea algoritmului este foarte simplă: un agent explorează un mediu iar în final primește o recompensă sau o penalizare. Astfel, agentul află dacă a procedat corect sau nu, fără a i se explica motivele.

În Reinforcement Learning, numită și *învățarea cu un critic* sau *învățarea recompensată*, nu se oferă nici un fel de indicii despre așteptări, singurul feedback fiind că rezultatul va fi categorisit ca fiind corect sau greșit. Situația este similară cu cea a unui critic ce afirmă doar că un anumit lucru e corect sau greșit, dar nu precizează în ce sens este greșit. Deseori însă recompensa este amânată. Aceste tehnici de învățare a căror recompensă este întârziată este momentan testată în anumite jocuri, cum ar fi șah sau table.

Reinforcement Learning se referă la o clasă de probleme care prezintă un agent ce explorează un mediu, și în acest mediu agentul înregistrează diverse acțiuni. Mediul este cel care răspunde la acțiuni prin recompense sau penalizări.

Algoritmii Reinforcement Learning caută să găsească un mod de acțiune în așa fel încât să se maximizeze recompensa. Subiectului care trebuie să învețe nu i se spune în ce direcție trebuie să acționeze, ca în cazul majorității tehnicilor Machine Learning, ci acesta trebuie să descopere singur care acțiune îi va aduce cea mai eficientă recompensă prin încercarea acțiunilor.

În cazurile cele mai complexe, acțiunile efectuate vor afecta nu doar recompensele imediate, ci și pe cele viitoare. Prin urmare tehnica Reinforcement Learning se distinge prin două caracteristici: cea de încercare – eșec și cea de recompensă amânată. Reinforcement Learning nu se va defini prin descrierea caracteristicilor metodelor de învățare, ci prin caracterizarea problemei de învățare. Ideea de bază este faptul că trebuie să fie înțelese aspectele problemei reale cu ajutorul unui agent de învățare, care e capabil de a înțelege domeniul și capabil de a lua măsurile corespunzătoare.

CAPITOLUL III. ELEMENTE DE PROIECTARE A SISTEMELOR DE ASISTARE A DECIZIILOR

Pe parcursul Capitolului III sunt discutate implicațiile sistemelor de asistare a deciziilor în domeniul managerial. În acest sens, inițial se prezintă în detaliu sistemele informatice de asistare a deciziilor, definiții ale acestora, clasificări, componente și direcții de dezvoltare, apoi se discută soluții de depozitare a datelor, insistând pe problema depozitelor de date și a tehnologiilor OLAP, respectiv pe conceptul Data Mining. În urma abordării acestor noțiuni, se va contura o structură a unui sistem informatic de asistare a deciziilor în cadrul unei organizații puternic informatizată. În cadrul aceluiași capitol sunt prezentate etape de proiectare a sistemelor software în conformitate cu standardele ingineriei software, arhitecturile sistemelor de asistare a deciziilor, arhitecturile sistemelor distribuite și sunt aduse în discuție sistemele de asistare a deciziilor de grup.

Activitatea managerială din cadrul organizațiilor a suferit modificări semnificative odată cu dezvoltarea societății informaționale, iar noile tehnologii informaționale au influențat pozitiv cel mai important domeniu al acestei activități, și anume adoptarea deciziilor. Sarcinile decizionale devin din ce în ce mai greu de îndeplinit fără ca decidentul uman să fie asistat de instrumente informatice, cunoscute sub denumirea de sisteme informatice de asistare a deciziilor (SIAD) sau sisteme suport de decizie (SSD).

În mod similar cu dificultatea găsirii unei definiții pentru sistemele de asistare a deciziilor, nici asupra componentelor unui sistem de asistare a decizie nu s-a căzut de acord în rândul autorilor cu preocupări în acest domeniu. Astfel, conform lui Sprague și Carlson, componentele sistemului informatic de asistare a deciziilor sunt:

- componenta de gestiune a datelor;
- componenta de gestiune a modelelor;
- componenta de gestiune a dialogului cu utilizatorului;
- arhitectura sistemului de asistare a deciziilor.

În accepțiunea noastră, componentele unui sistem informatic de asistare a deciziilor sunt asemănătoare celor identificate de Sprague în 1982 (Figura 2.), anume:

- interfața cu utilizatorul;
- subsistemele bazate pe cunoaștere;

- modulul de gestiune a datelor;
- modulul de gestiune a modelelor.

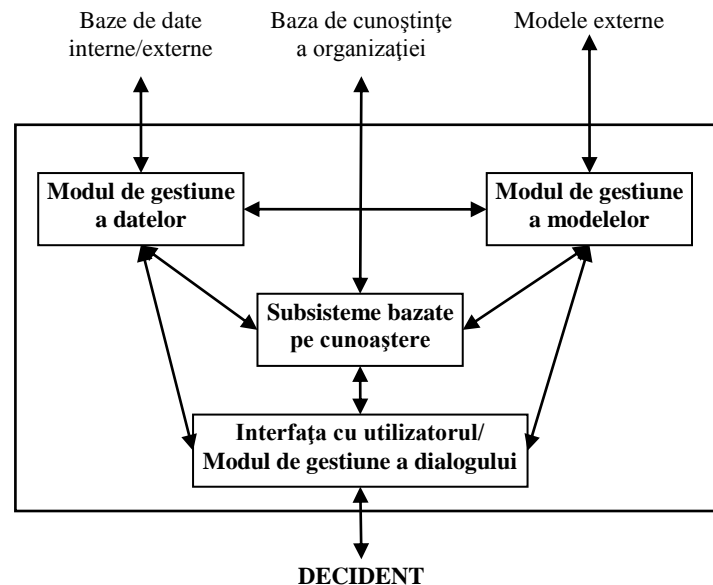


Figura 2. Componentele unui sistem de asistare a deciziilor
(adaptare după: Lungu, 2003)

La nivelul unei organizații avem de-a face cu cantități foarte mari de date, care provin atât din surse interne, cât și din surse externe. Sursele interne sunt reprezentate în mare parte de sistemul de producție, în timp ce sursele externe de date sunt reprezentate de parteneri, clienți, mediu, piață etc. Volumul datelor provenite din surse interne este superior volumului datelor ce provin din surse externe, dar acesta din urmă este în creștere datorită dezvoltării unor tehnici avansate de colectare a datelor.

Volumul important de date din cadrul întreprinderii trebuie să fie înmagazinat și păstrat în siguranță, pentru a putea fi exploatat, principalele medii de stocare al acestora fiind reprezentate de depozitul de date și magazia de date.

Tehnologiile disponibile pentru gestionarea datelor și informațiilor trebuie să contribuie la o mai bună înțelegere a trecutului și la previzionarea viitorului prin intermediul eficientizării deciziilor luate, aici intervenind tehnologiile Data Mining.

DW pot fi deosebit de utile diferitelor categorii de decidenți, iar principalele moduri în care se beneficiază de datele din cadrul DW sunt soluțiile de procesare analitică on-line (OLAP) și tehnicile Data Mining. Tehnologia OLAP se referă la posibilitatea de agregare a datelor din cadrul unui DW, având capacitatea de a obține din volumul mare de date informații utile procesului decizional din cadrul unei

organizații. Conform specialiștilor, un termen alternativ care ar fi mai reprezentativ pentru descrierea conceptului OLAP ar fi FASMI (Fast Analysis of Shared Multidimensional Information – Analiza rapidă a informațiilor partajate multidimensionale). Esența oricărui sistem OLAP este cubul OLAP, cunoscut și sub denumirea de cub multidimensional, format din fapte numerice numite măsuri, categorisite după dimensiuni. Aceste măsuri rezultă din articolele tabelor din cadrul bazelor de date relaționale. Rezultatele cerințelor utilizatorilor pot fi obținute prin parcurgerea dinamică a dimensiunilor cubului de date, la diferite niveluri de sinteză sau detaliere.

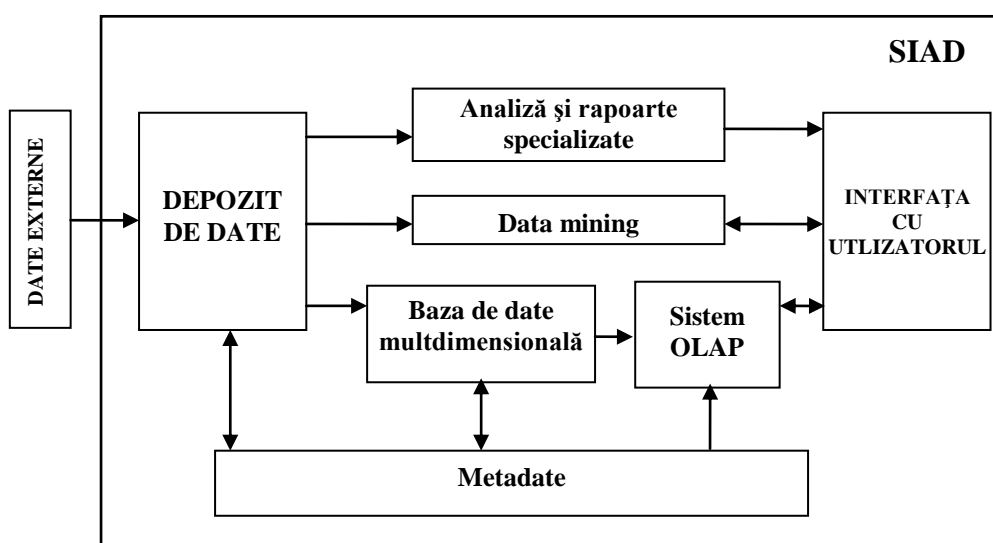


Figura 3. Componentele unui sistem de asistare a deciziilor într-o organizație puternic informatizată

Tehnologiile Data Mining integrate în sistemele de asistare a deciziilor determină existența unui instrument de asistare a deciziilor bazat încă pe interacțiunea om-mașină (om-sistem de calcul), iar aceste două entități luate împreună reprezintă un spectru de tehnologii informatice analitice care realizează o platformă pentru o combinație optimă între o analiză dictată de date, dar condusă de om [Ganguly05].

DW este o componentă importantă a unui sistem de asistare a deciziilor bazat pe date, iar legăturile acestuia cu alte componente sunt prezentate în Figura 3.

CAPITOLUL IV. STRATEGIA DE DEZVOLTARE ȘI IMPLEMENTARE A UNUI SISTEM DE ASISTARE A DECIZIILOR

Capitolul IV demarează cu abordări legate de dezvoltarea efectivă a sistemelor de asistare a deciziilor și legate de limbajele de programare pentru dezvoltarea acestora. Se subliniază avantajele abordării obiectuale în dezvoltarea sistemelor de asistare a deciziilor, justificând mai apoi selectarea limbajului C# și a mediului integrat Microsoft Visual Studio 2005 pentru implementarea efectivă a sistemului ce se află la baza prezentei lucrări. Alegerea soluției de reprezentare a datelor prin intermediul formatului XML se motivează prin compararea acestuia cu alte formate utilizate în cadrul altor sisteme similare, dar sunt evidențiate avantajele majore care arată că această soluție este una în mod clar superioară. Structura stabilită pentru fișierele XML este construită pe baza unei scheme XML, esențială pentru validarea sistemului și implicit pentru verificarea datelor din cadrul fișierului, rezultând astfel o modalitate de reprezentare simplă și cu un grad de siguranță sporit.

Sistemul propus și descris în continuare se vrea de fapt un framework al unui sistem de asistare a deciziilor ce va permite implementarea unor algoritmi de actualitate în scopul asistării decidenților în rezolvarea unor probleme decizionale complexe. În acest sens, sistemul este realizat îndeplinind în măsură cât mai mare principiile Design Patterns [Gamma 95], prin urmare codul poate fi considerat reutilizabil.

Sistemul a fost dezvoltat cu ajutorul mediului integrat de dezvoltare (IDE - Integrated Development Environment) Microsoft Visual Studio 2005, utilizând .NET framework 2.0, codul fiind scris în limbajul de programare C#.

Soluția Visual Studio este împărțită în două proiecte, unul reprezentând programul executabil, ce conține interfața grafică cu utilizatorul (GUI – Graphical User Interface), iar celălalt o bibliotecă de clase, ce conține logica de gestionare a relațiilor, atributelor și instanțelor precum și un număr de algoritmi de decizie implementați spre exemplificare. Această structură a soluției a fost realizată cu scopul de a separa partea de logică (nucleu) de partea de interfață, structură care de altfel este o recomandare generală în procesul de proiectare a aplicațiilor informatice.

Proiectul aferent bibliotecii de clase utilizate în sistemul menționat conține mai multe ansambluri de clase, și implicit mai mulți identificatori (Figura 4.). Ansamblul de clase de bază se va numi *ML.Core*; clasele legate de relații și subcomponentele acestora se vor afla în ansamblul de clase *ML.Core.Relations*, iar algoritmi se vor afla în ansamblul de clase *ML.Core.Algorithms*.

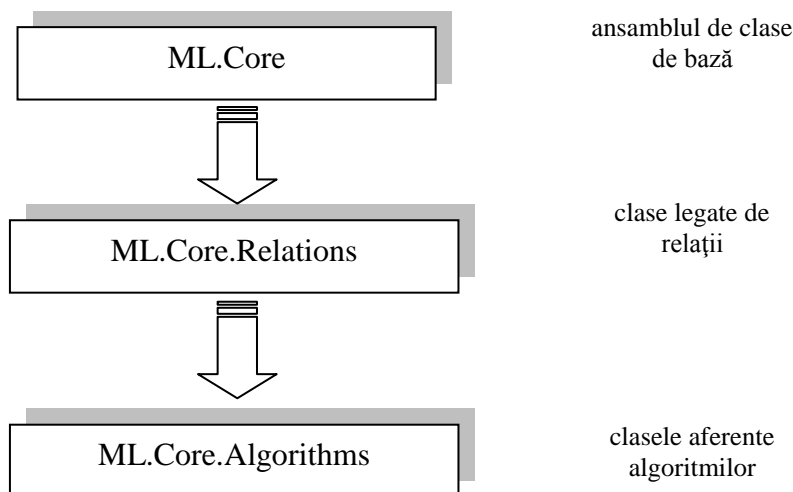


Figura 4. Ansamblurile de clase din cadrul bibliotecii de clase

Ansamblul alcătuit de modelele de date reprezentate în format XML împreună cu schemele XSD aferente și cu aplicații dezvoltate în limbaje de programare care oferă biblioteci de funcții pentru prelucrarea acestor modele de date (.NET, Java packages, Qt), reprezintă o soluție puternică și eficientă, dar mai ales elegantă din punct de vedere al programării orientate pe obiecte.

Ideea de a lucra cu fișiere XML a venit din cauza numeroaselor avantaje ale acestui format, dintre care menționăm:

- ușurarea procesului de structurare a datelor;
- „citirea” fișierelor nu întâmpină dificultăți, fiind din acest punct de vedere un format asemănător formatului text;
- independența de platformă fără a necesita licență.

Stocarea datelor necesare unui SIAD în format XML poate îmbrăca mai multe forme, iar varianta propusă în lucrarea prezentă a fost inspirată de proiectul WEKA, dezvoltat în scopuri de cercetare de către o echipă de la Waikato University din Noua Zeelandă. Reprezentarea datelor în WEKA se realizează cu ajutorul formatului ARFF

(Attribute-Relation File Format), care este un fișier text ce descrie o listă de instanțe ale unui set de atribute.

Similar cu maniera în care WEKA stochează aceste date în fișiere text, în fișierul XML se vor defini mai întâi atributele utilizate, identificate prin nume și tip, apoi datele propriu-zise, așa-numitele instanțe, fiecare reprezentând o enumerare de valori ale atributelor anterior definite. Lipsa unor atribute din definiția unei instanțe denotă o valoare necunoscută care va fi tratată diferit de algoritmi de decizie.

Dacă este să evidențiem îmbunătățirile față de alte formate utilizate, cum ar fi formatul simplu de tip text (plain-text) implementat spre exemplu de WEKA, acestea se concretizează în faptul că fișierul XML poate fi validat la încărcare, folosind o schemă de validare XML și funcțiile oferite de biblioteca de clase care include și metoda de parcurgere a documentului XML. De asemenea, se poate realiza și o validare la scrierea fișierului XML pe disc, pentru a verifica dacă fișierul va fi scris corect și va trece de o validare la încărcare ulterioară.

Documentul XML gestionat de sistem reprezintă o așa-numită relație, elementul *Relation* aflându-se la rădăcina arborelui XML (Figura 5.), acest element înglobând două elemente majore: lista de definiții ale atributelor (*Attributes*) și lista de instanțe (*Instances*) aferente acestor atribute. Fiecărei relații i se va atribui un nume, pentru a facilita identificarea, nume care va fi afișat în bara de titlu a interfeței grafice a aplicației.

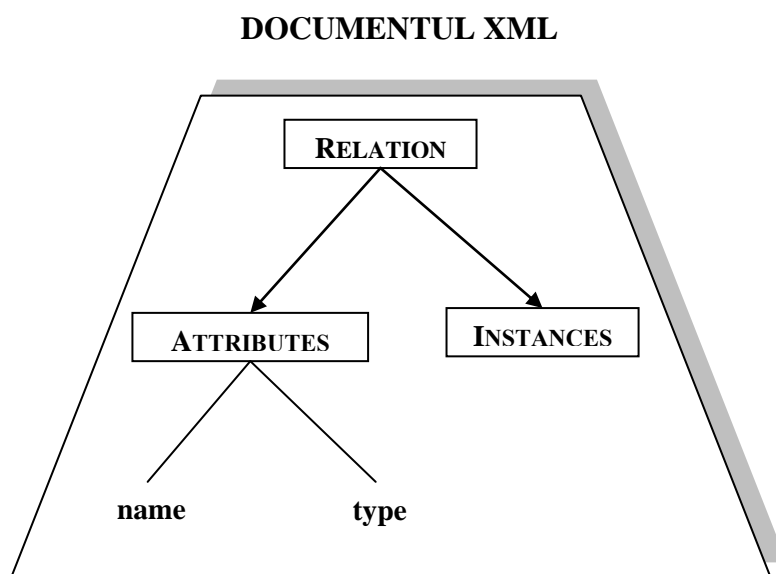


Figura 5. Structura documentului XML

CAPITOLUL V. SOLUȚII DE DEZVOLTARE, IMPLEMENTARE ȘI INTEGRARE A ALGORITMILOR DE DECIZIE ÎN CADRUL SISTEMELOR DE ASISTARE A DECIZIILOR

În cadrul Capitolului V sunt descrise inițial metodele de manevrare a fișierelor XML, asigurându-se optimizarea operațiilor de încărcare și salvare a fișierelor, și a celor de intrare/ieșire cu fișierele XML. Soluția de implementare a sistemului este prezentată la nivel de detaliu, cu reprezentarea entităților XML și a modelului top-down în cadrul bibliotecii de clase, biblioteca de clase fiind astfel realizată încât să ofere sistemului caracterul expandabil și reutilizabil. Este realizată prezentarea claselor abstracte pentru algoritmi, iar pentru a demonstra corectitudinea funcționalității acestora sunt implementați algoritmul pentru arbori de decizie (ID3) și algoritmul pentru învățarea bazată pe exemple (K-Nearest Neighbour). La finalul acestui capitol am propus, cu titlu exemplificativ, o variantă de interfață grafică, cu scopul de a demonstra funcționarea sistemului.

Urmând doctrina dezvoltării unui software orientat pe obiecte și cea a șabloanelor de dezvoltare, am considerat ca fiind potrivită crearea a câte o clasă pentru aproape toate tipurile de elemente XML aflate în cadrul unui document de relații. Alternativa ar fi fost ca întreaga procesare a elementelor XML să se execute doar în cadrul clasei *RelationDocument* și probabil folosirea pe toată întinderea bibliotecii de clase doar a claselor .NET corespunzătoare entităților XML (*XmlElement*, *XmlAttribute*, *XmlNode* etc.). Această abordare ar fi fost în primul rând greoi de utilizat din exteriorul bibliotecii de clase, caracterizând întreaga bibliotecă drept slab reutilizabilă, și în al doilea rând metodele de încărcare (*Load*) și salvare (*Save*) ar fi devenit aglomerate și neinteligibile.

După modul în care au fost structurate documentele XML reținem că elementul rădăcină a oricărui document XML reprezentând o relație este *Relation*. Astfel, în cadrul bibliotecii de clase va exista o clasă omonimă, care va implementa interfața *IXmlEntity*.

În Figura 6. este reprezentată în detaliu diagrama de clase aferentă claselor principale din cadrul bibliotecii de clase.

Implementarea algoritmilor în cadrul bibliotecii de clase constituie în mod firesc un pas major în procesul de dezvoltare software. Biblioteca de algoritmi își propune să ofere baza unui suport tehnic solid de dezvoltare pentru aproape orice algoritm de

decizie, și totodată să prezinte utilizatorilor acestei biblioteci posibilitatea de a folosi clasele de bază definite aici drept punct de plecare pentru implementarea de noi algoritmi, care se folosesc de un set de date ce poate fi reprezentat în formatul XML descris în capitolele anterioare (Figura 7.).

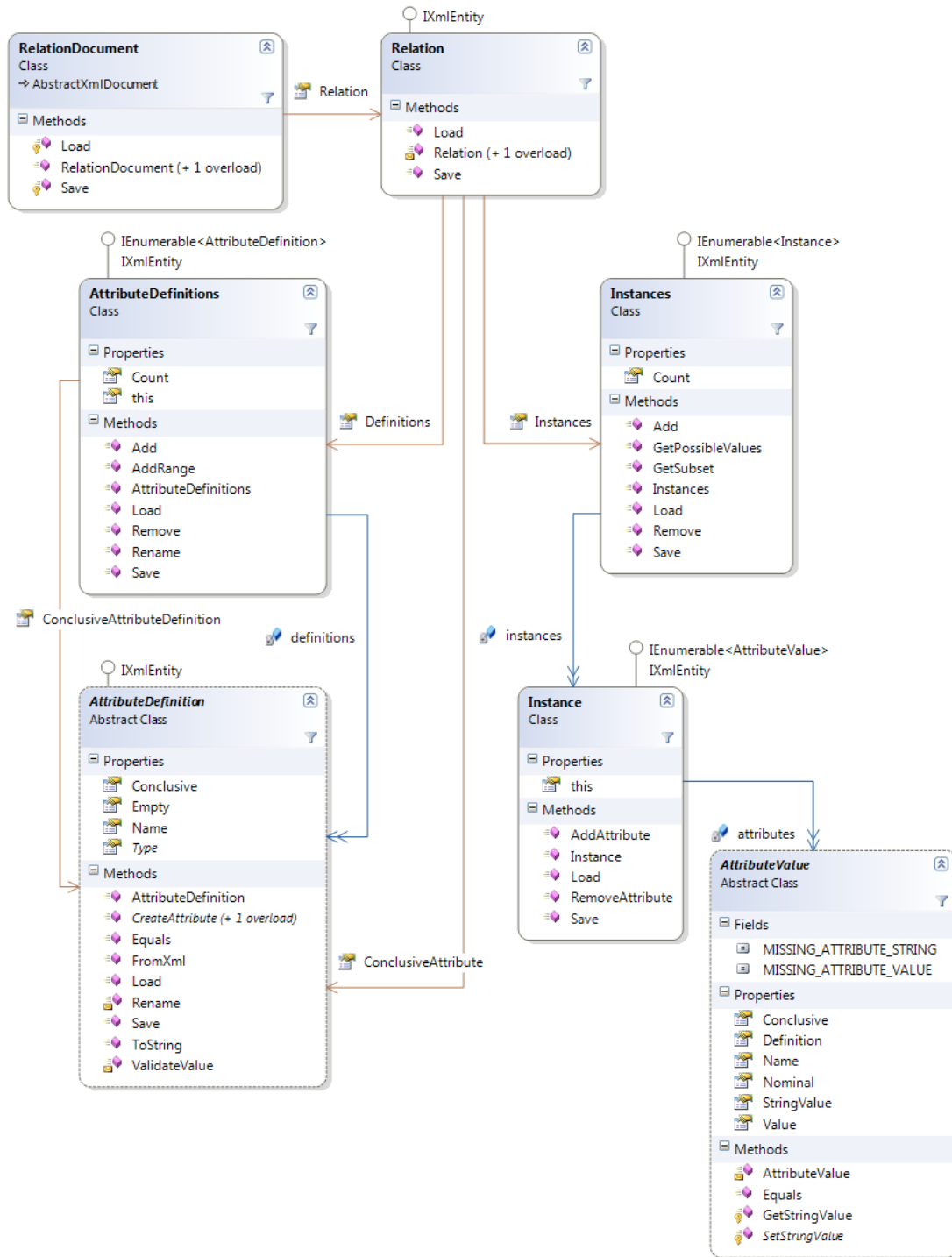


Figura 6. Diagrama de clase aferentă claselor principale din cadrul bibliotecii de clase

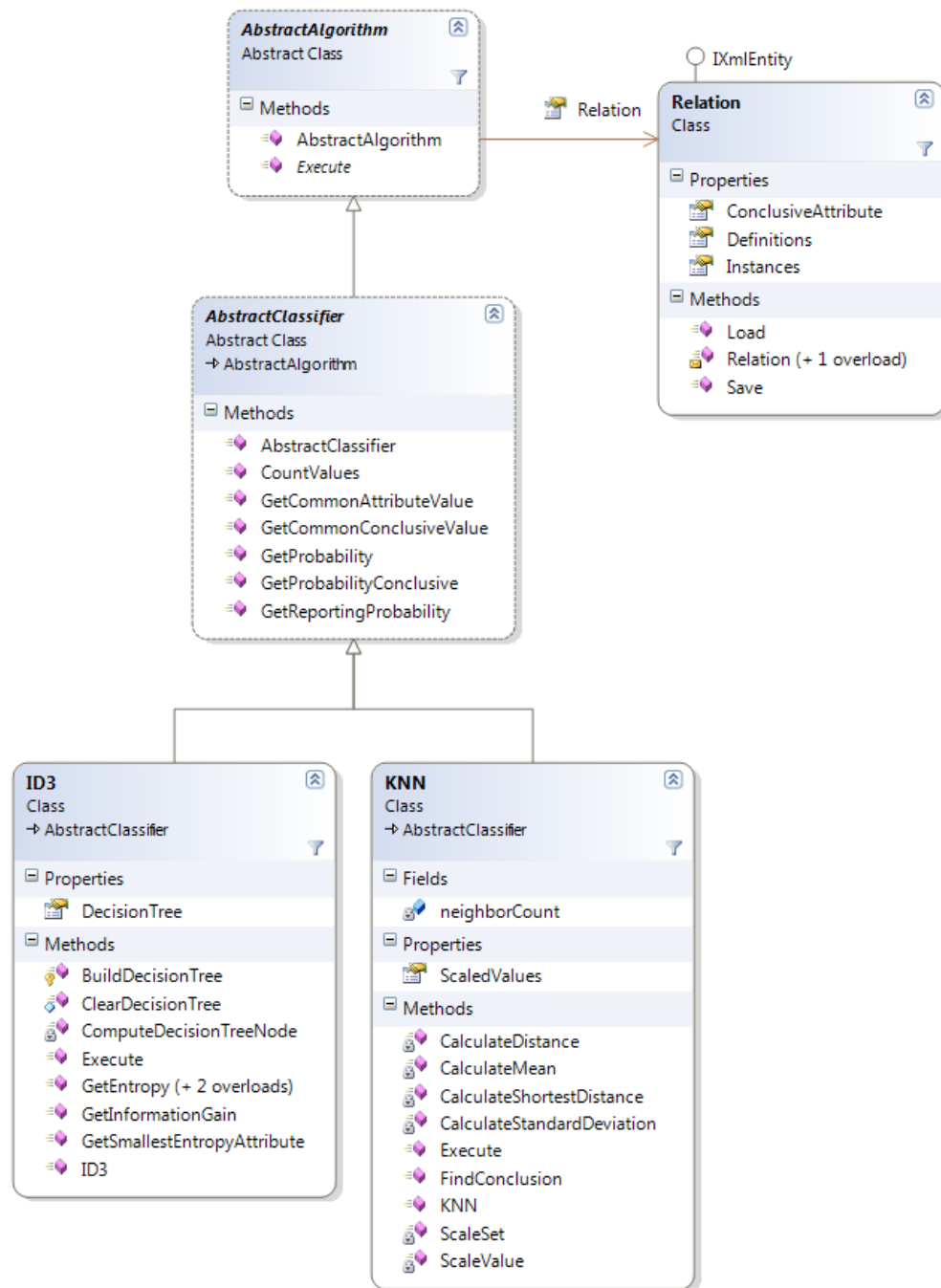


Figura 7. Ierarhia de clase de algoritmi

În scopul exemplificării algoritmilor de învățare, este prezentată implementarea algoritmului ID3 și a algoritmului KNN. Biblioteca de algoritmi a fost astfel construită încât adăugarea noilor algoritmi nu va implica probleme, dat fiind că am reușit realizarea unui framework intuitiv, în cadrul căruia supraîncărcarea metodelor este prezentă peste tot pentru a simplifica procesul.

CAPITOLUL VI. VERIFICAREA, VALIDAREA ȘI TESTAREA SISTEMULUI DE ASISTARE A DECIZIILOR

Având în vedere că în realizarea oricărui sistem software verificarea, validarea și testarea acestuia este un pas foarte important și deloc de ignorat, în ultimul capitol, Capitolul VI, demonstrăm cu ajutorul aplicației NUnit că sistemul funcționează conform așteptărilor, prin urmare am detaliat modalitatea de testare a aplicației. Se va realiza o bibliotecă de clase de testare, generică și expandabilă, având proprietatea de polimorfism, la fel ca biblioteca de clase a sistemului asupra căruia este efectuată testarea.

Verificarea și validarea au rolul de a asigura faptul că sistemul este potrivit pentru îndeplinirea scopului propus. În nici un caz, realizarea acestor activități nu vor determina ca sistemul să fie perfect, complet lipsit de erori, ci presupun ca sistemul rezultat să fie suficient de bun pentru sarcinile propuse.

Indiferent de metoda de testare utilizată, aceasta se va dovedi eficientă doar în condițiile în care au fost evidențiate cât mai multe erori și posibilele propagări ale acestora, care pot dăuna semnificativ calității sistemului. Cu ajutorul metricilor testării se pot urmări progresele software-ului pe măsură ce testarea și corectarea evoluează.

Testarea unei aplicații sau bibliotecii de clase .NET se realizează cu ajutorul aplicației NUnit, care la rândul ei oferă o bibliotecă întreagă de clase și funcții specifice testării și validării. Testarea se face automat, intern, printr-un mecanism de aruncare și prindere a excepțiilor în cazurile invalide.

Cea mai potrivită și recomandată practică este cea a definirii unei clase de test pentru fiecare clasă-țintă testată, și respectiv, a definirii unei metode de testare pentru fiecare proprietate publică din clasa-țintă. În cazul de față, de exemplu, vom avea clasa *TestID3* care va testa clasa *ID3*, și în cadrul clasei *TestID3* vom avea printre altele metoda *TestCountValues* care va testa metoda *CountValues* din clasa *ID3*.

Biblioteca de funcții oferită de NUnit funcționează pe baza asocierii unor attribute .NET cu clasele și metodele de testare, pentru a identifica operațiunile pe care acestea le vor desfășura în momentul rulării suitei de teste.

CONCLUZII ȘI PROPUNERI

Lucrarea de față evidențiază complexitatea procesului decizional în societatea contemporană și dovedește utilitatea sistemelor informatice de asistare a deciziilor, în condițiile în care modalitatea decizională evoluează. Mai mult, am arătat că nici sistemele informatice clasice nu mai răspund cu succes la cerințele decidenților, prin urmare am propus o soluție bazată pe tehnologii de mare actualitate, mai precis, tehnologiile Machine Learning.

Informația reprezintă baza oricărei decizii manageriale de calitate. Posibilitatea de a desfășura un proces decizional de calitate este direct proporțională cu cantitatea informațiilor de care se dispune și de calitatea acestora. Un impact semnificativ în acest sens îl au mijloacele informatice, deoarece acestea permit stocarea și gestionarea informațiilor, respectiv facilitează transmiterea și punerea în practică a deciziilor în timp util.

Procesul decizional trebuie perceput ca un proces complex, care în condiții ideale se desfășoară pe parcursul mai multor etape și faze. În acest sens se demarează cu pregătirea deciziei, după care se va alege soluția optimă și se va lua decizia efectivă, apoi se implementează decizia și se va controla modul de realizare al acesteia. Ultima etapă, deseori trecută cu vederea, presupune evaluarea rezultatelor deciziei, cu scopul de a se încerca o îmbunătățire a eficacității deciziilor viitoare. Etapele procesului decizional sunt de puține ori însă respectate în această formă în situațiile practice, în realitate ele nefiind clar definite și delimitate. Modelarea procesului decizional se realizează în concordanță cu fiecare etapă din structura procesului decizional, adică pentru fiecare activitate decizională se poate construi un model, sau mai multe, dacă este cazul.

Ideea că sistemele inteligente pot îndeplini sarcini considerate până recent exclusiv umane a atras interesul cercetătorilor din toate domeniile, aplicațiile din domeniul inteligenței artificiale devenind numeroase și materializându-se prin tehnologii noi. Una dintre aceste tehnologii, generată de dezvoltarea domeniului inteligenței artificiale și de interesul spre sisteme capabile să „învețe” în mod similar cu individul uman, este reprezentată de tehnologia Machine Learning.

Sistemele bazate pe tehnologii Machine Learning nu elimină neapărat intuiția umană, ci de obicei este preferată o abordare colaborativă între om și mașină, motiv

pentru care acestea se pretează a fi utilizate și la realizarea sistemelor de asistare a deciziilor.

Principalele tehnologii Machine Learning asupra cărora am insistat în studiul teoretic sunt arborii de decizie și Reinforcement Learning, dar am trecut în revistă și altele, cum ar fi învățarea bazată pe exemple, învățarea deductivă și învățarea inductivă, învățarea supervizată și învățarea nesupervizată.

Sistemele informatice de asistare a deciziilor sunt acele sisteme care vin în sprijinul decidentului uman în soluționarea problemelor decizionale complexe, semistructurate și nestructurate. Un aspect important de subliniat este faptul că în continuare omul deține controlul asupra procesului decizional, iar aceste sisteme doar oferă alternative de decizie, fundamentate cu ajutorul instrumentelor de modelare și analiză a datelor.

Având în vedere că în cadrul organizațiilor avem, de obicei, de-a face cu un volum foarte mare de date, care trebuie înmagazinate și păstrate, apoi gestionate și prelucrate, s-au căutat soluții eficiente pentru optimizarea acestor procese. O soluție în acest sens este depozitul de date, sistem complex care conține datele operaționale și istorice ale organizației, provenite atât din surse interne, cât și externe. În ceea ce privește prelucrarea volumului uriaș de date, tehnologia Data Mining răspunde cu succes acestei provocări. Tehnologiile Data Mining sunt potrivite pentru stabilirea de șabloane în cadrul seturilor de date, chiar date brute, neprocesate, oferind diferite rezultate ce pot fi utilizate în cadrul sistemelor informatice de asistare a deciziilor.

Principalul obiectiv al lucrării a fost acela de a pune bazele unei biblioteci de clase care permite implementarea unor algoritmi de decizie complecși. Astfel a fost realizat un mediu expandabil și reutilizabil, ce permite crearea algoritmilor de actualitate ce au ca scop asistarea decidenților în rezolvarea unor probleme decizionale de mare complexitate. De asemenea, în prealabil s-a urmărit realizarea unui studiu bine fundamentat cu privire la mediul decizional, la sistemele informatice de asistare a deciziilor și la tehnologiile Machine Learning, respectiv la modalitățile în care aceste tehnologii pot interveni în eficientizarea și modernizarea procesului decizional.

În ansamblul ei, soluția prezentată este una puternică și eficientă, „elegantă” din punct de vedere al programării orientate pe obiecte, fiind formată din modele de date reprezentate în format XML împreună cu schemele XSD aferente și cu aplicații dezvoltate în limbajul de programare C#, care oferă biblioteci de funcții pentru prelucrarea acestor modele de date.

Aplicația a fost inspirată de proiectul WEKA, sistem utilizat pentru aplicații Machine Learning, ce oferă o interfață uniformă cu utilizatorul, dar și un număr de algoritmi diferiți, inclusiv cei de inducție a regulilor, învățarea bazată pe exemple algoritmi de regresie, algoritmi pentru reguli relaționale. WEKA este un sistem ce se poate aplica foarte bine pentru rezolvarea problemelor de Data Mining. Formatul fișierului, deși soluția propusă folosește formatul XML, conține aceleași informații care se regăsesc în formatul .arff definit de WEKA.

Ideea de a utiliza fișiere XML a fost determinată de numeroasele avantaje ale acestui format. Astfel, formatul XML permite ca procesul de structurare a datelor să se realizeze cu mare ușurință, iar citirea acestor fișiere nu impune dificultăți, fiind oarecum similare cu formatul text. De asemenea, fișierele XML se pretează pentru reprezentarea structurilor arborescente și, foarte important, validarea datelor din cadrul lor se realizează prin intermediul unor scheme XML. Prin urmare o îmbunătățire importantă față alte formate utilizate.

Particularizând pentru cazul realizării unui framework pentru un sistem integrat de asistare a deciziilor, considerăm că formatul XML este potrivit, dat fiind că poate reprezenta și trata cu ușurință structuri arborescente, este simplu de utilizat și nu impune multe probleme în cazul alterării structurilor (este flexibil) – dar prezintă un dezavantaj, și anume faptul că nu excelează în cazul cantităților foarte mari de date, nefiind indexabil.

Adăugarea implementării claselor reprezentând algoritmi în cadrul bibliotecii de clase a venit în mod natural ca o parte importantă a procesului software. Această secțiune a bibliotecii de clase a fost concepută având în minte caracteristicile principale ale algoritmilor și ale tipurilor de algoritmi, dar și intenționând să oferim posibilitatea extinderii acestor clase pentru orice abstractizare a unei implementări a unui algoritm.

În mod evident, direcțiile de studiu din cadrul acestei lucrări nu au fost epuizate. Avem în vizor continuarea cercetărilor în domeniul sistemelor informatice de asistare a deciziilor bazate pe tehnologii Machine Learning. De la bun început am avut în vedere posibilitatea de dezvoltare viitoare a aplicației, cu scopul de a permite implementarea mai multor algoritmi de decizie și posibilitatea de a utiliza noi tehnologii.

O direcție de dezvoltare viitoare ar fi posibilitatea de comunicare a aplicației cu alte baze de date. În acest sens, avem în vedere posibilitatea de a utiliza baze de date MySQL în primă fază, iar ulterior orice alte conexiuni, pentru a permite lucrul cu cantități foarte mari de date. Portarea poate fi realizată ușor, dat fiind că există deja

implementarea claselor, trebuind doar să fie adaptate și conectate la entități din baze de date. De asemenea, ca o altă direcție de studiu viitoare, de mare actualitate, ar fi Web Services, ce presupune crearea de metode web (ASP .NET) care să permită inițializarea setului de date și executarea de algoritmi asupra acestora.

Concluzia principală ce se desprinde este că lucrarea prezintă o abordare nouă în ceea ce privește dezvoltarea sistemelor de asistare a deciziilor, evidențiind noi modalități de implementare a algoritmilor de decizie, prin intermediul unui sistem expandabil care a fost implementat respectând principii avansate de ingineria programării.

BIBLIOGRAFIE SELECTIVĂ

1. Airinei, D., *Depozite de date*, Editura Polirom, Iași, 2002
2. Airinei, D., *Sisteme de asistare a deciziilor -note de curs*, Universitatea A. I. Cuza Iași, Facultatea de Economie și Administrarea Afacerilor, 2006
3. Alter, S., *Decision Support Systems*, Addison Wesley, 1980
4. Andone I., Mockler R., Dologite D., Țugui Al., *Dezvoltarea sistemelor inteligente în economie*, Editura Economică, București, 2001
5. Awad, E. M., *Building Expert Systems: Principles, Procedures, and Applications*, West Publishing, Minneapolis/St. Paul, 1996
6. Bishop, C., *Pattern Recognition and Machine Learning*, Springer Publishing, 2006
7. Boldur-Lățescu, *Logica decizională și conducerea sistemelor*, Ed. Academiei Române, București, 1992
8. Cojocariu, A., A. Munteanu, Sofran (Stanciu), Cristina-Ofelia, *Machine Learning techniques for enhancing decision support systems within organizations*, 29th International Convention MIPRO, Conference on Business Intelligence Systems (BIS), Opatija, Croatia, May 22 -26, 2006
9. Cojocariu, A., A. Munteanu, Sofran (Stanciu), Cristina-Ofelia, *On Machine Learning Technologies for Knowledge Discovery*, The 8th International Conference on Informatics in Economy „Information & Knowledge Age, Ed.Economică, ISBN 973-8360-04-8, 2007
10. Dietterich, T. G., *Machine Learning In Nature Encyclopedia of Cognitive Science*, London. Macmillan, 2003
11. Filip F.Gh., *Decizie asistată de calculator: decizii, decidenți – metode de bază și instrumente informatice asociate*, Editura Tehnică, București, 2005
12. Gamma, E., Helm, R., Johnson, R., Vlissides, J., *Design Patterns – Elements of Reusable Object – Oriented Software*, Addison – Wesley, 1995
13. Ganguly, A. R., Gupta A., *Data Mining Technologies and decision Support Systems for Business and Scientific Applications*, Encyclopedia of Data Warehousing and Mining, Blackwell Publishing, 2005
14. Graz, P., Watson, H., *Decision Support in the Data Warehouse*, Prentice Hall, Upper Saddle River Publishing, 1998
15. Hand, D., Mannila, H., Smyth, P., *Principles of Data Mining*, MIT Press, 2001
16. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I., *The WEKA Data Mining Software: An Update*, SIGKDD Explorations, Volume 11, Issue 1, 2009
17. Holsapple, C., Whinston, A., *Decision Support Systems: A Knowledge Based Approach*, St. Paul, West Publishing, 1996
18. Inmon, W.H., *Building the Data Warehouse, 3rd Edition*, Wiley Computer Publishing, USA, 2002

19. Jurca, I., *Ingineria programării* – note de curs – Universitatea Politehnica Timișoara, Facultatea de Automatică și Calculatoare, 2003
20. Langley, P., *Elements of Machine Learning*, Morgan Kaufmann, San Francisco, 1996
21. Lucey, T., *Management Information Systems – 7th edition*, DP Publications, London, 1995
22. Lungu, I, colectiv, *Sisteme informatice – Analiză, proiectare și implementare*, Editura Economică, București, 2003
23. Militaru, V., *Studiu comparat asupra tehnicilor de data mining utilizate în rezolvarea problemelor de regresie și clasificare*, Revista *Informatica Economica*, nr. 3(27), 2003
24. Mitchell, T., *Machine Learning*, McGraw – Hill Publishing, 1997
25. Myers, G. J., *The Art of Software Testing*, Wiley & Sons Publishing, New Jersey, 2004
26. Nicolescu, O. și alții, *Sistemul informațional managerial al informației*, Ed. Economică, București, 2001
27. Nilsson, N. J., *Introduction to Machine Learning: An Early Draft of a Proposed Textbook*, Stanford University, 1996
28. Nițchi, Șt., Avram-Nițchi, Rodica, *Data mining, o nouă eră în informatică*, Revista Byte, România, <http://www.byte.ro/byte97-02/18tend.html>, Februarie 1997
29. Nițchi, Șt., Racovițan, D., colectiv, *Inițiere în informatica economică și de afaceri*, Editura Risoprint, Cluj-Napoca, 2003
30. Oancea, Mirela, *Sisteme informatice pentru asistarea deciziei financiare*, Editura ASE, București, 2005
31. Oprea, D., *Analiza și proiectarea sistemelor informaționale economice*, Editura Polirom, Iași, 1999
32. Power, D., *Decision Support Systems Hyperbook.*, Cedar Falls, IA: DSSResources.COM, <http://dssresources.com/dssbook/>, 2000
33. Power, D., *Decision Support Systems: Concepts and Resources for Managers*, Westport, 2002
34. Quinlan, J. R., *Induction of Decision Trees*, Mach. Learn. 1, 1 (Mar. 1986), 81-106, 1986
35. Radu, I, Vlădeanu, D., *Fundamentarea deciziilor complexe prin tehnici de simulare*, Editura Economică, București, 2002
36. Radu, I., colab, *Informatică și management – o cale spre performanță*, Editura Universitară, București, 2005
37. Russell, S. J., Norvig, P., *Artificial Intelligence: A Modern Approach* (2nd ed.), Upper Saddle River, New Jersey: Prentice Hall, 2003
38. Schneiderman, B., *Design the User Interface*, Addison Wesley Publisher, 1997

39. Sofran (Stanciu), Cristina – Ofelia, *Importanța sistemelor de asistare a deciziei în cadrul societății informaționale*, ANALE Seria Științe Economice (Universitatea „Tibiscus” din Timișoara), Vol. XII, ISSN 1582 – 6333, 2006
40. Sofran (Stanciu), Cristina – Ofelia, *Decision Support Systems – Present and Perspective*, Universitatea din Oradea – Facultatea de Științe Economice Integrarea Europeană – noi provocări pentru economia României Anale Seria Științe Economice (Universitatea din Oradea) TOM XIV ISSN 1582 – 5450, 2007
41. Sofran (Stanciu), Cristina – Ofelia, *Solutions for the development of Decision Dupport Systems*, ANALE Seria Științe Economice (Universitatea „Tibiscus” din Timișoara), Vol. XV, ISSN. 1582 – 6333, 2008
42. Sofran (Stanciu), Cristina – Ofelia, *Business intelligence for the decision system improvement*, Proceedings of the International Conference Research people and actual tasks on multidisciplinary sciences, Lozenec, Bulgaria, 2007
43. Sofran (Stanciu), Cristina – Ofelia, A. Cojocariu, A. Munteanu, *On Decision Trees Algorithms for Economic Processes Enhancement*, microCAD International Scientific Conference, Miskolc, Hungary, 2008
44. Sofran (Stanciu), Cristina – Ofelia, A. Cojocariu, *XML Technologies for Improving Data Management for Decision Algorithms*, 20th DAAAM International Conference, Vienna, 2009
45. Sommerville, I., *Software Engineering (6th Edition)*, Addison – Wesley Publishing, 2000
46. Sprague, R. H., Carlson, E. D., *Building effective decision support systems*, Prentice-Hall, 1982
47. Turban E., Aronson J., *Decision Support Systems and Intelligent Systems*, Prentice Hall, SUA, 2001
48. Turban, E., McLean, E., Wetherbe, J., *Information Technology For Management: Improving Quality And Productivity*, John Wiley Publishing, 1995
49. De Ville, B., *Decision Trees for Business Intelligence and Data Mining*, SAS Press Series, 2006
50. Vlahovic, N., *Discovering Tacit Knowledge in Business Decision Making*, WSEAS TRANSACTIONS on BUSINESS and ECONOMICS, Issue 3, Volume 5, Pages 72-81, March 2008
51. van der Vlist, E., *The W3C's Object-Oriented Descriptions for XML*, O'Reillys Publishing, 2002
52. Witten, I. H., Frank, E., *Data Mining – Practical Machine Learning Tools and Techniques, Second Edition*, Norgan Kaufmann Publishing, 2005
53. Witten, I. H, Frank, E., Trigg, L., Hall, M., Holmes, G., Cunningham, S. J., *Weka: Practical Machine Learning Tools and Techniques with Java Implementations*, Proceedings of the ICONIP/ANZIIS/ANNES'99 Workshop on Emerging Knowledge Engineering and Connectionist-Based Information Systems 192-196, 1999
54. Zaharie D., Albulescu F., Bojan I., Ivacenco V., Vasilescu C., *Sisteme informatice pentru asistarea deciziei*, Editura Dual Tech, 2001