Îmbunatatirea clasificarii bayesiene prin algoritmi genetici, cu aplicatii în text mining

Flavian VASILE, masterand
Managementul Informatizat al Proiectelor, A.S.E. Bucuresti

This paper studies the possibility of increasing the accuracy of classification and obtaining knowledge about the quantitative correlations between the representative features for the classification. The algorithm starts from a naive Bayesian classifier and incrementally increase complexity with a state-of-the-art genetic algorithm: N.E.A.T.
The paper its based on the results obtained using the application "Mentat"[2], developed and presented in the graduation thesis "Evolutive Text Mining" by the same author [1]. It also contains a survey about the types of software developed for text mining and future development.
Keywords: *text mining, Bayesian classifier, genetic algorithm, correlation, representative features, optimising, networks.*

Introducere

Data mining-ul a aparut ca o continuare a metodelor traditionale de analiza a depozitelor de date (pe volume mari). Datorita costurilor de implementare fidicate a aparut necesitatea identificarii unor solutii de extragere a esentialului dintr-un volum foarte mare de date, la costuri mult mai reduse, solutii cunoscute astazi sub denumirea de data mining. Odata cu expansiunea Internet-ului si a informatiei de tip text în format electronic, a aparut necesitatea extragerii automate de cunostinte si din text si astfel data mining-ul a cunoscut o noua specializare: **text mining-ul**.
Spre deosebire de data mining, text mining-ul presupune un software care se adreseaza publicului larg consumator de servicii în retea, motivele pentru aceasta fiind universalitatea cererii de achizitionare de informatie în timp real si costurile mici (pretul conexiunii) de achizitionare a informatiei, comparativ cu data mininig-ul.
Text mining-ul are drept obiectiv principal extragerea automata de cunostinte, ce trebuie sa îndeplineasca la rândul lor cerintele de: *noutate, validitate, operationalitate.*

Aplicatii de text mining

Aplicatii de text mining se pot clasifica în aplicatii online si aplicatii offline. În prezent, cele mai cunoscute aplicatii ale text mining sunt:
a) Aplicatii online pentru:
- *cautarea inteligenta pe Internet*, care presupune *analiza de continut* (cu ajutorul tehnicilor de text mining documentele regasite de un motor de cautare sunt filtrate fiind pastrate doar rezultatele cu sensul cautat), *dezvoltarea unui profil al utilizatorului* (documentele sunt aduse automat fara o solicitare expresa din partea utilizatorului, plecând de la un profil al preferintelor conform carora programul cauta independent documente relevante).
- *regasirea stirilor interesante*: selectarea de stiri de pe Usenet este o provocare care apare în viata de zi cu zi a multor utilizatori de Internet.
- *regasirea de raspunsuri la întrebari*: exista întrebari frecvente la care altcineva poate raspunde sau a raspuns deja. Acestea se numesc *frequently asked questions* (F.A.Q.) si se gasesc împreuna cu raspunsurile lor în baze de date.
- *filtrarea postei*: partitionarea multimii scrisorilor electronice în grade de prioritate în functie de reactiile anterioare ale utilizatorului si emiterea de recomandari de stergere a unor mesaje pasibile de a fi comerciale sau neinteresante.
b) Aplicatii offline pentru:

- *clasificarea documentelor preluate de pe Internet.*
- *rezumarea documentelor* (obtinând astfel documente condensate si rezumate, abstracturi ale lucrarilor depozitate)
- *concatenarea documentelor* (de exemplu crearea de relatari ale unor stiri pe baza informatiilor preluate din mai multe surse)

Aplicatiile mentionate nu respecta întru totul obiectivul de extragere a cunostintelor, aceste sisteme realizând cel mult extragere de informatii. Extragerea cunostintelor reclama mai mult de la metodele de text mining si anume întelegerea, macar aproximativa, a documentelor si crearea unui sistem de generare într-o forma inteligibila de cunostinte noi. În prezent, interpretarea rezultatelor este realizata în continuare de utilizator pe un numar însa mai mic de informatii mai exacte.

O directie de dezvoltare a text miningului.

Una dintre directiile de dezvoltare cu un impact potential deosebit de amplu este extragerea automata interdependentelor dintre termenii relevanti ai domeniului supus analizei.

Deoarece în text mining se pleaca de la ideea ca un sistem artificial de tratare a limbajului are obiective limitate si nu trebuie sa obtina o interpretare corecta a întelesului textelor tratate, regulile sintactice folosite în limbaj nu trebuie neaparat implementate. Totusi, tratarea cuvintelor ca entitati independente si clasificarea informatiilor doar dupa indici derivati din frecvente si raportari la nivel de cuvânt are unele limite deoarece *reprezentarea datelor nu este o problema colaterala, ea exprima în fapt ipotezele care se fac asupra proprietatilor sau mai bine zis metaproprietatilor limbajului si a formei sale concrete, textul.*

În prezent, cea mai mare parte din metodele de text mining se bazeaza pe metode inductive (de construire a unor retele neuronale sau a unor arbori de decizie), care de obicei fac o cautare locala, de tip greedy, în spatiul legilor.

Algoritmii genetici au caracteristici interesante pentru aplicatiile de data si text mining, cum ar fi:

- capacitatea de a explora mai în profunzime spatiul de solutii,
- capacitatea de a înlatura problema interactiunii atributelor,
- capacitatea de a opera o cautare cu final deschis pentru descoperirea de pattern-uri,
- capacitatea de a descoperi cunostinte de nivel calitativ înalt (reguli if-then, etc.).

Ca dezavantaj, algoritmii genetici au timpi mari de prelucrare în cazul volumelor mari de date.

Implementarea algoritmilor genetici pe retele neuronale. Algoritmul N.E.A.T.

Deoarece algoritmii genetici au nevoie de o solutie de start, aceasta solutie poate fi una din structurile traditionale de tratare a documentelor.

În acest scop am introdus în cadrul algoritmului un clasificator Bayesian naiv ca genom de start. Optimizarea structurii retelei ar fi dus la surprinderea interdependentelor pornind de la presupunerea ca o structura ce reprezinta din ce în ce mai bine domeniul documentelor considerate drept interesante de catre utilizator se apropie asimptotic de structura reala a conceptelor din cadrul acelui domeniu. Astfel o clasificare corecta a documentelor înseamna în cele din urma o "învatare" a regulilor de clasificare folosite de catre utilizator, iar interdependentele ce se stabilesc între termeni în cadrul structurii de implementare a regulii sunt valide si operationale.

Metoda de optimizare a structurii retelei clasificatorului naiv propusa în aceasta lucrare se bazeaza pe *algoritmul NEAT* [3] (NeuroEvolution of Augmenting Topologies) prezentat în lucrarea "Evolving Neural Networks through Augmenting Topologies", autori: Kenneth O. Stanley si Risto Mukklainen, Department of Computer Sciences, The University of Texas at Austin, un raport tehnic datat 28 Iunie

2001, ceea ce face din acest algoritm unul din cei mai noi algoritmi genetici.
Algoritmul genetic dezvoltat de cei doi cercetatori americani evolueaza în paralel atât structura cat si ponderile retelei, lucru rar întâlnit pâna acum în literatura de specialitate si de multe ori soldat cu esecuri. Inovatiile pe care algoritmul le aduce cresc valoarea perspectivei genetice din care domeniul se hraneste cu idei. Astfel, *cele trei puncte cheie* pe care se bazeaza evolutia sunt:
1. codificarea genetica cu markere genetice ce identifica în mod unic genele si folosirea în cadrul reproducerii a informatiilor despre gene;
2. protejarea inovatiei genetice prin speciere;
3. cresterea incrementala de la o structura minimala.
1. *Codificarea genetica.* Una din cele mai cunoscute probleme din cadrul algoritmilor genetici este problema permutatiilor sau a competitiei conventiilor. Aceasta înseamna ca exista mai multe modalitati de a reprezenta o solutie la o problema de optimizare a ponderilor într-o retea. Când genomurile ce reprezinta aceeasi solutie nu au aceeasi forma, reproducerea nu este împiedicata iar urmasul este foarte probabil nefezabil (ca si în cazul reproducerii umane cosangvine!).
Algoritmul rezolva aceasta problema introducând o notatie unica a genelor, fiecare dintre acestea având un identificator dat de ordinea aparitiei, numit "historic marking" ce are omolog în natura: genele se interschimba dupa ce genoamele participante sunt supuse unui test de asemanare si genele cu acelasi stramos sunt aliniate.
2. *Specierea.* Inovatia apare în cadrul evolutiei retelelor neuronale atunci când o noua substructura este adaugata prin mutatie. Aceasta adaugare de noi noduri si/sau conexiuni prin intermediul mutatiilor nu poate sa duca din prima generatie la un fitness crecut ci la unul mai scazut. Pâna când acestor noi structuri vor dobândi o functie si vor fi conectate la întreaga retea, scorul genomului scade vertiginos si poate fi eliminat din competitia reproducerii înainte de a putea sa-si dovedeasca capacitatile. Daca în schimb toate structurile noi ar fi pastrate artificial pentru un timp oarecare ar aparea probleme de calcul si de stabilire a acestei perioade. Autorii au folosit înca odata o solutie folosit înca si de natura: specierea. Astfel, fiecare individ este nevoit sa concureze în interiorul speciei sale iar pe ansamblu speciile concureaza între ele prin intermediul randamentului lor mediu.
Specierea are nevoie însa de o functie de asemanare pentru a putea grupa indivizii în specii. Aceasta face uz tocmai de identificatorii genelor pentru a putea recunoaste indivizii asemanatori. Populatia se segmenteaza astfel pe specii, fiecare într-o anumita nisa variabila. Functia de asemanare este de forma:

$$d = \frac{c_1 E}{N} + \frac{c_2 E}{N} + c_3 \overline{W}$$

unde:
E = numarul de gene în exces (al caror identificator este mai mare de cel mai mare identificator comun, apartin individului mai "tânar" genetic)
D = numarul de gene diferite (al caror identificator este mai mic de cel mai mare identificator comun)
N = numarul de gene din genomul mai lung
W*med* = diferenta medie a ponderilor genelor comune
c1,c2,c3 = ponderea fiecarui factor
Fitnessul unei specii este media fitnessurilor indivizilor (aritmetica, geometrica sau armonica).
3. *Cresterea incrementala de la o structura minimala.* În majoritatea cazurilor algoritmii genetici pornesc de la o colectie de genomuri generate aleator, ceea ce genereaza o multitudine de probleme: marirea artificiala a spatiului de cautare, numar crescut de calcule, probleme de codificare si generare aleatoare, obtinerea unei solutii prea complexe. Solutia este de obicei introducerea unui factor de penalizare a cresterii complexitatii în cadrul functiei de evaluare, dar algoritmul tot pleaca de la un

spatiu prea complex si ireal prin raport cu realitatea.
În realitate, un moment de start este determinat de evolutia incrementala de pâna atunci si situatia respectiva este modelata tot de aceleasi legi si nu este haotica. De aceea, algoritmul NEAT pleaca de la o populatie de indivizi identici cu o structura minimala (retea monostrat complet conectata– fara strat ascuns).
Evolutia va fi protejata prin speciere, lucru imposibil pâna acum. În acest mod, orice genom produs si perpetuat are o justificare de a exista, este o inovatie valida si care a depasit solutia de start.

Elemente novative aduse de modificarea algoritmului pentru problemele de text mining.
Algoritmul si elementele novative au fost testate prin dezvoltarea aplicatiei Mentat. În cazul concret al aplicatiei „Mentat", a fost folosita ca solutie de start un clasificator clasic pentru text mining, o retea Bayesiana monostrat antrenata, ce opera deja clasificari performante. Folosirea retelei antrenate este posibila tocmai datorita ipotezei naive a independentei atributelor ce duce la forma monostrat a retelei. Evolutia ulterioara a retelei avea sa produca doua efecte în cazul unui succes:
- sa îmbunatateasca rezultatele clasificatorul naiv, considerat cel mai potrivit pentru aplicatiile de text mining;
- sa ofere solutiei o justificare teoretica prin eliminarea ipotezei simplificatoare si introducerea de dependente prin evoluarea treptata a unui strat ascuns.

Acest strat ascuns reprezinta evolutia genetica a structurii si are doua surse de aparitie:
1. mutatia de adaugare a unui nod pe un arc ce leaga direct stratul de intrare si cel de iesire
2. mutatia de adaugare a unui arc între doua noduri neconectate pâna acum (deoarece toate nodurile de intrare si iesire erau complet conectate, aceste arce nu apar decât între nodurile din stratul ascuns si celelalte noduri)

Noile retele sunt separate pe specii conform algoritmului iar conectarea noilor noduri este "premiata" pentru fiecare conexiune nou introdusa în sistem. Aceasta premiere se pastreaza doar în cazul în care ipoteza introdusa prin noua conexiune este verificata la noua evaluare a generatiei: daca mai multe noduri din stratul de intrare se conecteaza la acelasi nod din stratul ascuns înseamna ca sunt interdependente si determina împreuna rezultatul din stratul de iesire cu o mai mare putere decât fiecare din ele datorita efectului de sinergie. Acest principiu este modelat matematic prin introducerea premierii.
Aceasta este pentru moment nimic mai mult decât o ipoteza: nu a fost verificata pe o instanta reala. Daca instanta probeaza ipoteza, premierea ramâne. Daca nu, pentru fiecare conexiune neverificata reteaua primeste o penalizare mai mare decât premierea initiala. Astfel sunt promovate structurile cu adevarat inovative.
De ce este nevoie de speciere? Deoarece testarea se face la fiecare generatie din motive de calcul doar pentru un singur document pe clasa, ceea ce înseamna ca probarea unei ipoteze în întregimea sa se face de-a lungul mai multor generatii.

Rezultate
Conform rezultatelor obtinute, perspectivele algoritmului sunt promitatoare. Clasificarea, în cazul unei antrenari medii (10 documente/domeniu), se face foarte aproape de o clasificare umana, iar valoarea extragerii cunostintelor exprimata prin frazele-cheie extrase la nivelul claselor desi are o interpretare subiectiva, a fost probata prin cautari multiple ale caror rezultate au fost de fiecare data adaugarea la aceeasi clasa cu un scor crescut.

Concluzii
Versiunea actuala a aplicatiei a urmarit în primul rând probarea unor ipoteze din cadrul algoritmului si adâncirea cunoasterii informatiei structurata textual. Deoarece rezultatele au fost satisfacatoare, o versi-

une ulterioara, ar putea cuprinde ca îmbunatatiri:

- *integrarea pe motor de cautare/agent inteligent de cautare*: cautarile dupa frazele-cheie ale unui domeniu s-ar putea face automat iar rezultatele s-ar constitui ca noi cunostinte ale domeniului, astfel întreaga etapade achizitionare de cunostinte ar putea deveni automata;
- *depistarea unor noi domenii de interes* nu numai din bazinul documentelor neclasificate ci si din adresele cautate;
- *integrarea altor surse de informatii* în format electronic: ziare electronice, newsgroup-uri, discussion groups, e-mail;
- *aducerea optionala si a imaginilor* ce insotesc documentatia, pentru a obtine o baza multimedia de cunostinte;
- *explorarea si mai detaliata a fiecarui domeniu* cu ajutorul arborilor de decizie implementati pe grupurile de cuvinte interdependente identificate.

Bibliografie

1. Flavian Vasile – "Text Mining Evolutiv" – Lucrare de licenta, ASE, Bucuresti, 2002;
2. Flavian Vasile – "Mentat" – aplicatie text mining pentru Internet 2002 www.ideaboys.ro;
3. Constanta-Nicoleta Bodea – "Inteligenta artificiala – Calcul neuronal", Editura ASE 2002;
4. Kenneth O. Stanley, Risto Mukklainen – "Evolving Neural Networks through Augmenting Topologies", Technical Report TR-Ai-01-290, June 28, 2001;
5. Anders Holst – "The Use of a Bayesian Neural Network Model for Classification Tasks", Dissertation - September 1997, STOCKOLMS UNIVERSITET;
6. Dunja Mladenic – „Machine Learning on non-homogeneous, distributed text data", Ljubljana, 1998
7. Robert Armstrong, Dayne Freitag, Thorsten Joachims, Tom Mitchel – „Web Watcher – A Learning Apprentice for the World Wide Web";
8. Dunja Mladenic, Marko Grobelnik – „Predicting content from hyperlinks"
9. Peter F Brown, Vincent J Della Pietra, Peter V de Jonza, Jenifer C Lai – „Class-based n-gram models of natural language";
10. David D Lewis, William A Gale – „A Sequential Algorithm for Training Text Classifiers" SIGIR 1994;
11. Andrew McCallum, Kamal Nigam, Jason Rennie, Kristie Seymore – „A Machine Learning Approach to Building Domain-Specific Search Engines" www.cora.justresearch.com;
12. Jose M. Bernardo – „Bayesian Statistics" Departamento de Estadistico, Facultad de Matematicas, 46100 – Burjassot, Valencia, Spain;
13. Kamal Paul Nigam – „Using Unlabeled Data to Improve Text Classification", May 2001;
14. David Heckerman (heckerma@microsoft.com) - „A tutorial on Learning with Bayesian Networks" November 1996;
15. Jason D.M.Rennie - „Improving Multi-class Text Classification with Naive Bayes";
16. McCallum, K. Nigam – „Employing EM and Pool-Based Active Learning for Text Classification".