RĪGAS TEHNISKĀ UNIVERSITĀTE
Datorzinātnes un informācijas tehnoloģijas fakultāte
Informācijas tehnoloģijas institūts

Sigita MISIŅA-EGLE
Informācijas tehnoloģijas doktora programmas doktorante

INDUKTĪVĀS KLASIFIKĀCIJAS ALGORITMU AR INKREMENTĀLU APMĀCĪBU ANALĪZE UN REALIZĀCIJA

Promocijas darba kopsavilkums

Zinātniskā vadītāja
Dr.sc.ing., asoc. profesore
L. ALEKSEJEVA

Rīga 2009

004.65.021+004.421](043.2)
Mi 826 i

Misiņa-Egle S. Induktīvās klasifikācijas algoritmu ar inkrementālu apmācību analīze un realizācija. Promocijas darba kopsavilkums. - R.: RTU, 2009. – 27 lpp.

Iespiests saskaņā ar 2008. gada 05. septembra ITI Padomes sēdes lēmumu, protokols Nr. 08-04

ISBN

Šis darbs ir izstrādāts ar Eiropas Sociālā fonda atbalstu Nacionālās programmas „Atbalsts doktorantūras programmu īstenošanai un pēcdoktorantūras pētījumiem" projekta „Atbalsts RTU doktorantūras attīstībai" ietvaros.

PROMOCIJAS DARBS
IZVIRZĪTS RĪGAS TEHNISKĀS UNIVERSITĀTĒ
INŽENIERZINĀTŅU DOKTORA GRĀDA IEGŪŠANAI

Promocijas darbs inženierzinātņu doktora grāda iegūšanai tiek publiski aizstāvēts 2009.gada 06. aprīlī Rīgas Tehniskās universitātes Datortehnikas un informācijas tehnoloģijas fakultātē, Meža ielā 1, 3. korpusā, 202. auditorijā.

OFICIĀLIE RECENZENTI

Profesors, Dr.habil.sc.ing. Jānis Grundspeņķis
Rīgas Tehniskā universitāte

Docents, Dr.sc.comp. Jānis Zuters
Latvijas Universitāte

Pētnieks, Ph.D. (sc.tech.) Vitalijs Kolodjažnijs
Bāzeles Universitāte, Šveice

APSTIPRINĀJUMS

Es, Sigita Misiņa-Egle apstiprinu, ka esmu izstrādājusi doto promocijas darbu, kas iesniegts izskatīšanai Rīgas Tehniskajā universitātē inženierzinātņu doktora grāda iegūšanai. Promocijas darbs nav iesniegts nevienā citā universitātē zinātniskā grāda iegūšanai.

Sigita Misiņa-Egle…….(Paraksts)

Datums: 2008 gada 23. decembrī

Promocijas darbs ir uzrakstīts latviešu valodā, satur ievadu, 5 nodaļas, rezultātu analīzi un secinājumus, bibliogrāfisko sarakstu, 3 pielikumus, 23 zīmējumus un ilustrācijas, kopā 127 lappuses. Literatūras sarakstā ir 55 nosaukumi.

VISPĀRĒJAIS DARBA RAKSTUROJUMS

Tēmas aktualitāte

Vairākumā agrāk izstrādāto datu ieguves klasifikācijas algoritmu tiek pieņemts, ka objekta klase ir stabila un nemainīga laikā; šie algoritmi darbojas, klasificējot statiskus datus. Taču daudzās dzīves sfērās pieņēmums par objekta klases nemainīgumu nav spēkā esošs, jo ir novērojama zināšanu novecošanās. Konservatīvie klasifikācijas algoritmi ir neefektīvi tiešsaistes datu klasifikācijas uzdevumos.

Objektu klasifikācijai mainīgu klašu un tiešsaistes datu gadījumā ir piemērota inkrementālā apmācība, kur apmācības piemēri tiek klasificēti laika gaitā. Inkrementālajā apmācībā klases apraksts netiek pieņemts kā konstante; algoritma apmācība notiek inkrementāli, apstrādājot jaunus datu plūsmas piemērus un aizmirstot novecojušus piemērus. Šim apmācības veidam ir savas priekšrocības un trūkumi. Trūkums ir tāds, ka pēc noteikta laika iepriekš klasificētie piemēri, iespējams, vairs netiks korekti klasificēti. Savukārt priekšrocība ir tā, ka zināšanu novecošanās gadījumā inkrementālās apmācības algoritms vairāk uzmanības pievērsīs jaunajiem klases piemēriem un klases apraksta maiņai.

Darba mērķis

Promocijas darba mērķis ir izpētīt un pilnveidot induktīvās klasifikācijas algoritmus ar inkrementālu apmācību tiešsaistes datu apstrādei.

Darbā pētīti, analizēti un uzlaboti induktīvās klasifikācijas algoritmi ar inkrementālu apmācību tiešsaistes datu apstrādei. Lai sasniegtu darba mērķi, nepieciešams realizēt šādus uzdevumus:

1. Izpētīt induktīvās klasifikācijas algoritmu ar inkrementālu apmācību darbības principus.
2. Veikt minēto algoritmu darbības pārbaudi vidēs ar trokšņainiem datiem un klases apraksta izmaiņu gadījumā, konkrēti e-pasta ziņojumu klasifikācijā.
3. Izpētīt e-pasta ziņojumu pazīmju apstrādes tehnoloģijas to izmantošanai induktīvās klasifikācijas uzdevumam.
4. Izstrādāt inkrementālās apmācības algoritma *MLII* hibrīdu, lai uzlabotu oriģinālā algoritma efektivitāti darbam ar dažādiem datu tipiem un paplašinātu tā lietojamību.
5. Izpētīt un analizēt inkrementālās apmācības algoritmu *FLORA2*, tā darbības principus, tajā izmantoto novērošanas loga izmēra pielāgošanas heiristikas metodi.
6. Aprobēt darbā izmantotos inkrementālās apmācības algoritmus izveidojot programmatūru par e-pasta ziņojumu klasificēšanu, veikt iegūto praktisko rezultātu salīdzinošo analīzi.

Pētījuma priekšmets

Pētījuma priekšmets ir datu ieguvē prognozēšanas uzdevumu risināšanai izmantotie induktīvās klasifikācijas algoritmi, kuros pielieto inkrementālu apmācības procesu. Šādi algoritmi efektīvi darbojas vidēs, kur novērojama zināšanu novecošanās, trokšņaini dati un klases apraksta izmaiņas, kā arī šie algoritmi ir piemēroti datu plūsmu apstrādei.

Pētījuma hipotēzes

1. Algoritmi, kuri izmanto inkrementālās apmācības metodi, ir daudz efektīvāki tādos datu ieguves klasifikācijas uzdevumos: a) kas apskata trokšņainus datus; b) kas darbojas ar datu plūsmām; c) kur novērojama strauja zināšanu novecošanās; d) kurus nav iespējams atrisināt, izmantojot klasiskos induktīvās klasifikācijas algoritmus ar statisko apmācību.
2. Daudzkārtainais inkrementālās secināšanas algoritms *MLII*, balstīts uz paplašinātās matricas apmācības algoritmu *HCV*, kurš paredzēts simbolisko mainīgo klasifikācijai un tiek izmantots likumu formulas (konjunktīva formula, kas apraksta ceļu paplašinātajā matricā) iegūšanai, būtu plašāk izmantojams un efektīvāks, ja tiktu balstīts uz algoritmu *CN2,* kurš veido likumus cilvēkam saprotamā formā un apstrādā gan skaitliskus, gan simboliskus datus.

Pētījumu metodes

Promocijas darbā tiek izmantotas datu ieguves, matemātiskās loģikas, informācijas teorijas metodes, kā arī inkrementālās apmācības pieeja un daudzkārtainā inkrementālā secināšana. Darbā izmantoti tādi klasifikācijas uzdevumi, kuri balstīti uz induktīvo secināšanu un paredzēti klasifikatoru ģenerēšanai likumu kopas veidā.

Zinātniskais jaunieguvums

1. Pētīti un analizēti induktīvās klasifikācijas algoritmi ar inkrementālu apmācību tiešsaistes datu klasificēšanai, veikta izmantoto algoritmu praktiska realizācija un eksperimentālo rezultātu salīdzinošā analīze.
2. Izstrādāts daudzkārtainais inkrementālās secināšanas algoritma hibrīds *HMLII*, kurš paplašina oriģinālā algoritma *MLII* lietojamību, ļaujot apstrādāt gan skaitliskus, gan simboliskus datus, uzlabo tā ātrdarbību un uzrāda augstāku klasifikācijas precizitāti.

Darba praktiskais pielietojums

Darba gaitā izpētīti vairāki inkrementālās apmācības algoritmi. Algoritmu darbības principi pārbaudīti, izmantojot praktisku uzdevumu par e-pasta ziņojumu klasificēšanu. Algoritmu apmācības rezultātā tiek ģenerēti klasifikatori likumu formā,

kurus var izmantot ienākošā e-pasta šķirošanai. Vislabākos rezultātus praktiskajā uzdevumā uzrādīja autores izstrādātais hibrīdais algoritms *HMLII*.

Izstrādāta aplikācija programmā *MS Excel* algoritma *HMLII* praktisko eksperimentu realizācijai datu un likumu apstrādei un klasifikatora testēšanai.

Izstrādāta jauna programmatūra *EmailFlora* inkrementālās apmācības algoritma *FLORA2* pētīšanai un realizācijai, e-pasta ziņojumu klasifikatora iegūšanai un testēšanai.

Par darba rezultātiem tika ziņots šādās zinātniskajās konferencēs:

Misina S. Comparative analysis of e-mail classification methods with incremental learning. RTU 49th International Scientific Conference, Riga, Latvia, October 13, 2008.

Misina S. Incremental learning for e-mail classification. "The 9th Fuzzy Days 2006" – International Conference on Computational Intelligence. Dortmund, Germany, September 18 – 20, 2006.

Misina S. Multi-layer incremental learning linked to nonincremental induction."ICAISC" – 8th International Conference on Artificial Intelligence and Soft Computing. Zakopane, Poland, June 25 – 29, 2006.

Misina S. Incremental learning based on non-incremental induction. The 5th International Conference on Operational Research: Simulation and Optimization in Business and Industry. Tallin, Estonia, May 17 - 20, 2006.

Misina S. Incremental learning algorithm FLORA2 for e-mail classification. RTU 47th International Scientific Conference, Riga, Latvia, October 13, 2006.

Misina S., Aleksejeva L. Inductive inference algorithms in e-mail messages filtering. "Mendel 2005" 11th International Conference on Soft Computing, Brno, Czech Republic, June 15 – 17, 2005.

Misina S., Alexeyeva L. Inductive inference algorithm in multi-layer incremental learning. RTU 46th International Scientific Conference, Riga, Latvia, October 13, 2005.

Misina S. Efficiency analysis of real-time classification rule construction methods. "Mendel 2004" 10th International Conference on Soft Computing, Brno, Czech Republic, June 16 – 18, 2004.

Misina S. Inductive inference algorithms in e-mail messages filtering. The 3rd Estonian Summer School in Computer and System Science, Pedase, Estonia, August 08 – 12, 2004.

Misina S. On-line classification rule construction methods. The 9th Estonian Winter School in Computer Science, Lahemaa, Estonia, February 29 – March 5, 2004.

Misina S. Inductive inference algorithms in e-mail messages filtering. RTU 45th International Scientific Conference, Riga, Latvia, October 14 – 16, 2004.

Misina S., Alekseyeva L. Efficiency analysis of real-time classification rule construction methods. RTU 44th International Scientific Conference, Riga, Latvia, October 9 – 11, 2003.

Promocijas darba galvenie rezultāti

Promocijas darbā pētīti un analizēti vairāki klasifikācijas algoritmi ar inkrementālu apmācību, to darbības principi, priekšrocības un trūkumi, praktiskais pielietojums.

Detalizēti galvenie darba rezultāti:

1. Izpētīti induktīvās klasifikācijas algoritmu ar inkrementālu apmācību darbības principi un pielietojums. Secināts, ka inkrementālās apmācības algoritmi spēj pielāgoties izmaiņām klases aprakstā, jo darbojas ar datu plūsmām ierobežotā novērošanas logā, kura izmēri var tikt mainīti atkarībā no tā, vai iespējama klases apraksta nobīde.
2. Ir izpētīta e-pasta klasifikācijas interfeisa aģenta metode *MAGI*, kura balstīta uz statiskās klasifikācijas algoritmu *CN2*. Secināts, ka interfeisa aģenta arhitektūru var efektīvi izmantot e-pasta ziņojumu klasificēšanā, tiek piedāvāts gatavu modeli paplašināt statiskās apmācības algoritma vietā, pielietojot inkrementālās apmācības algoritmu, jo tādi algoritmi ir piemērotāki darbam ar tiešsaistes datiem.
3. Veikta daudzkārtainā inkrementālās secināšanas algoritma *MLII* hibridizācija - izstrādāts algoritms *HMLII,* kurā statiskai apmācībai izmantots algoritms *CN2*, kurš paplašina oriģinālā *MLII* pielietošanu un uzlabo tā efektivitāti.
4. Algoritma *HMLII* praktisko eksperimentu realizācijai datu un likumu apstrādei un klasifikatora testēšanai izveidota aplikācija programmā *MS Excel*, bet likumu ģenerēšanai pielietota programma *Sipina for Windows*.
5. Darbā pētīts inkrementālās apmācības algoritms *FLORA2*, kurš apstrādā datu plūsmas, analizēta adaptīvā novērošanas loga izmēra pielāgošanas heiristikas metodes efektivitāte, kas ļauj algoritmam adaptēties izmaiņām klases aprakstā un izvairīties no neefektīvu likumu iegūšanas, tai pat laikā likumu kopā saglabājot kandidātu likumus, kuri būs noderīgi turpmākajā klasifikācijas procesā.
6. Izstrādāta programma *EmailFlora,* kas realizē inkrementālās apmācības algoritmu *FLORA2* ar adaptīvu datu novērošanas loga izmēra heiristiku un ir paredzēta e-pasta ziņojumu klasifikatora ģenerēšanai un testēšanai.
7. Praktisks e-pasta ziņojumu klasifikācijas uzdevums risināts ar:
    a) interfeisa aģenta metodi;
    b) daudzkārtaino inkrementālās secināšanas algoritmu *MLII*;
    c) daudzkārtaino inkrementālās secināšanas algoritma hibrīdu *HMLII*;
    d) inkrementālās apmācības algoritmu *FLORA2* ar adaptīvu datu loga izmēra heiristiku.

Veikta visos gadījumos iegūto rezultātu salīdzinošā analīze un izdarīti secinājumi par algoritmu efektivitāti un piemērotību konkrētajam uzdevumam.

Publikācijas

Pētījuma rezultāti ir atspoguļoti 11 publikācijās starptautiskos Latvijas Zinātnes padomes atzītos zinātniskajos izdevumos. Publikāciju saraksts ir iekļauts kopējā promocijas darbā izmantojamās literatūras sarakstā. Pētījumi veikti ar Eiropas Sociālā fonda atbalstu Nacionālās programmas "Atbalsts doktorantūras programmu īstenošanai un pēcdoktorantūras pētījumiem" projekta "Atbalsts RTU doktorantūras attīstībai" ietvaros. Darba rezultāti izmantoti IZM - RTU pētniecības projektā R 7085 „Mākslīgais intelekts prognozēšanas un vadības uzdevumos".

Darba struktūra un apjoms

Promocijas darbs satur ievadu, 5 nodaļas, rezultātu analīzi un secinājumus, bibliogrāfisko sarakstu, 3 pielikumus, 23 zīmējumus un ilustrācijas, kopā 127 lappuses. Literatūras sarakstā ir 55 nosaukumi.

Ievadā ir pamatota veikto pētījumu aktualitāte, formulēts pētījuma priekšmets, promocijas darba mērķis un uzdevumi, zinātniskais jaunieguvums, kā arī sniegts īss pētījuma pamatvirzienu raksturojums.

Pirmajā nodaļā sniegts ieskats par induktīvajiem klasifikācijas algoritmiem, to darbības principiem, iedalījumu, to priekšrocībām un trūkumiem. Aprakstīti inkrementālās klasifikācijas algoritmi, to pielietojums darbam ar datu plūsmām un gadījumos, kad novērojama klases apraksta nobīde. Darba pirmajā nodaļā aprakstīti autores izvirzītie uzdevumi inkrementālās apmācības algoritmu uzlabošanai, strādājot ar datu plūsmām un klases apraksta nobīdi.

Otrajā nodaļā pētīta interfeisa aģenta *MAGI* arhitektūra un tā izmantošana e-pasta ziņojumu klasificēšanā. Aprakstīta praktiskā uzdevuma nostādne un dots datu modelis, ar kuru tiek pārbaudīti visu darbā pētīto un izstrādāto metožu darbības principi. Otrajā nodaļā aprakstīti izstrādātie praktiskie eksperimenti interfeisa aģenta *MAGI* apmācībā, izmantojot algoritmu *CN2* inkrementālas apmācības veidā.

Darba trešajā nodaļā dots autores izstrādātā hibrīdā algoritma *HMLII* apraksts, sasaistot induktīvās secināšanas algoritmu *CN2* ar daudzkārtaino inkrementālās secināšanas algoritmu *MLII*. Veikti praktiskie eksperimenti ar algoritmiem *MLII* un *HMLII* e-pasta ziņojumu klasificēšanā.

Ceturtajā promocijas darba nodaļā pētīts un analizēts inkrementālās apmācības algoritms *FLORA2* ar adaptīvo piemēru apakškopas izmēra pielāgošanas heiristiku. Aprakstīti praktiskie eksperimenti ar *FLORA2* e-pasta ziņojumu klasifikācijas uzdevumā.

Piektajā nodaļā sniegti darbā izstrādātās programmatūras apraksti, darbības principi un pielietojums. Analizētas programmu iespējas un ierobežojumi praktisko eksperimentu veikšanā klasifikācijas uzdevumu risināšanā.

Pielikumā pievienota informācija par eksperimentos izmantotajiem datiem un autores izstrādāto programmu pirmteksti.

DARBA ATSEVIŠĶO NODAĻU IZKLĀSTS

Pirmā nodaļa

Pirmajā nodaļā sniegts ieskats par induktīvajiem klasifikācijas algoritmiem, to darbības principiem, iedalījumu un pielietojumiem. Aprakstītas induktīvās klasifikācijas algoritmu priekšrocības, trūkumi un autores izvirzītie uzdevumi to pētīšanā un uzlabošanā.

Induktīvās klasifikācijas algoritmi ir daļa no datu ieguves klasifikācijas algoritmiem. Ar šo algoritmu palīdzību tiek klasificēti gan statiski, gan mainīgi dati. Induktīvās klasifikācijas algoritmi apmācības procesa rezultātā ģenerē lēmumu koku vai likumu kopu, kas vēlāk tiek izmantota kā klasifikators jaunu piemēru klasifikācijai.

Likumu indukcija balstās uz induktīvo secināšanu. Induktīvā secināšana ir pretējs jēdziens dedukcijai. Dedukcijā ir dots pilnībā ticams likums, un tiek izsecināts, kādos specifiskos gadījumos tas pielietojams. Indukcija savukārt no specifiskiem piemēriem izsecina vispārīgu likumu. Likumu indukcijas mērķis ir iegūt tādu likumu kopu, kas atdala vienu klasi no otras. Likumu indukcija no apmācības piemēru kopas, kur katram piemēram definēta klase, ģenerē likumu kopu, kas var tikt izmantota turpmākai klasifikācijai. Induktīvās klasifikācijas algoritms uz iepriekšējo datu pamata veido klasifikācijas modeļus (klasifikatorus). Algoritma darbības princips sastāv no diviem etapiem:

1. Apmācība, pamatojoties uz sākotnējo piemēru kopu, klasifikatora iegūšana;
2. Eksāmena veikšana testēšanas piemēru kopai, izmantojot iegūto klasifikatoru.

Likumus testējot, tos var novērtēt pēc to pārklāšanas (*coverage*) un precizitātes (*accuracy*) vērtībām. Pēc precizitātes vērtības tiek noteikta algoritma efektivitāte – jo lielāka precizitāte, jo efektīvāks ir algoritms konkrētajam uzdevumam. Likuma R pārklāšana [14] ir procentuāla vērtība, ko nosaka ar likumu pārklāto piemēru (likuma nosacījuma daļa atbilst piemēra atribūtu vērtībām, un likuma klase atbilst piemēra klasei) skaitu dalot ar visas datu kopas piemēru skaitu:

$$parkl(R) = \frac{n_{parkl}}{|D|}, \tag{1}$$

kur $|D|$ – datu kopas piemēru skaits;
n_{parkl} – piemēru skaits, ko pārklāj likums R.

Likumu precizitātes noteikšanai [5] tiek izmantota *kopējās klasifikatora precizitātes (KKP)* aprēķina formula (2):

$$KKP = \frac{n_{korekti}}{|D|}, \tag{2}$$

kur $n_{korekti}$ – pareizi klasificētu piemēru skaits ar likumu *R*.

Tomēr gadījumos, kad piemēru sadalījums pa klasēm nav vienlīdzīgs, t.i. kādas klases piemēri ir pārsvarā, *KKP* novērtējums var būt maldinošs. Tādos gadījumos precizitātes mērīšanai izmanto *neskaidrības matricas* (*confusion matrix*) datus [5], kuras izmērs ir *y*y,* kur *y* ir klašu skaits. 1. tabulā parādīta matricas aizpildīšana divu klašu gadījumā, kad mērķa jeb interesējošā klase tiek saukta par *pozitīvo*, bet otra - par *negatīvo*.

1. tabula

Neskaidrības matrica

		Prognozētā klase	
		+	-
Īstā klase	+	f_{++} *(IP)*	f_{+-} *(KN)*
	-	f_{-+} *(KP)*	f_{--} *(IN)*

Neskaidrības matricā tiek ierakstīti klasifikācijas iznākumi, respektīvi, pareizi vai kļūdaini klasificēto piemēru skaits. To aprakstīšanai izmanto sekojošus mainīgos:

- *Īstie pozitīvie (IP)* atbilst pozitīvo ierakstu skaitam un tika klasificēti kā pozitīvie;
- *Kļūdainie negatīvie (KN)* atbilst pozitīvās klases ierakstu skaitam, bet tika klasificēti kā negatīvie;
- *Kļūdainie pozitīvie (KP)* atbilst negatīvās klases ierakstu skaitam, bet tika klasificēti kā pozitīvie;
- *Īstie negatīvie (IN)* atbilst negatīvo ierakstu skaitam un tika klasificēti kā negatīvie.

Klasifikācijas precizitātes noteikšanai no neskaidrības matricas tiek izmantoti divi mēri – pozitīvās klases precizitāte *p* jeb *PKP* un atsaukšana *r* jeb *PKA* (skat. formulas (3) un (4)):

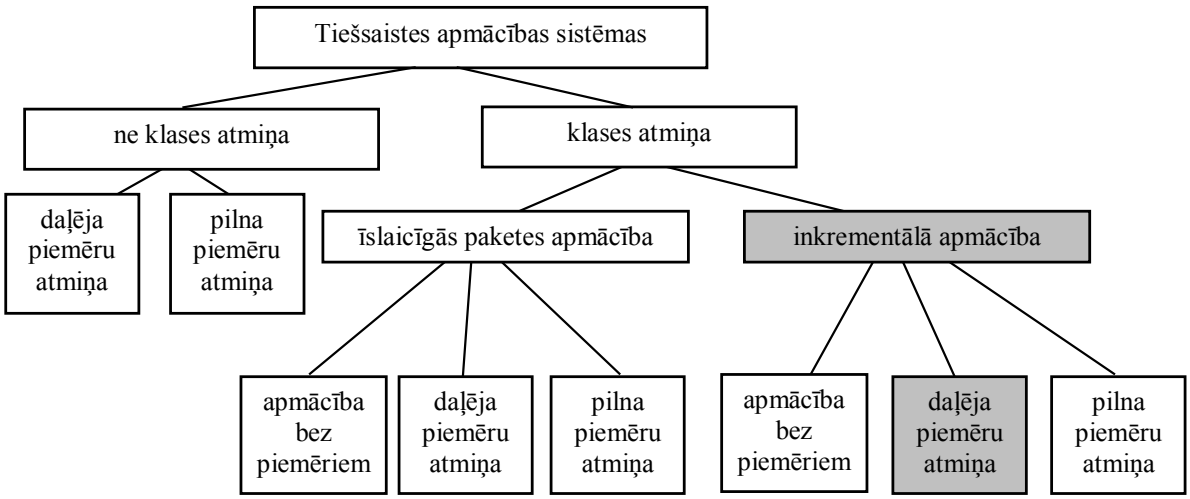
$$p = IP/(IP + KP) \tag{3}$$

$$r = IP/(IP + KN) \tag{4}$$

Tālāk pirmajā nodaļā dots klasifikācijas sistēmu iedalījums un inkrementālās apmācības darbības principi. Klasifikācijas sistēmas vispārīgi var iedalīt divās grupās, vadoties pēc to apmācības veida:

a) statiskās apmācības sistēmas;
b) tiešsaistes (*on-line*) apmācības sistēmas.

Objektu klasifikācijai mainīgu klašu un datu plūsmu gadījumā ir piemērota tiešsaistes apmācība, kur apmācības piemēri tiek klasificēti laika gaitā. Tiešsaistes apmācības sistēmas tiek iedalītas trijās daļās [25]: ne-klases atmiņas apmācība, daļēja klases apmācība, klases atmiņas apmācība (skat. 1. att.).



1. att. Tiešsaistes apmācības sistēmu iedalījums

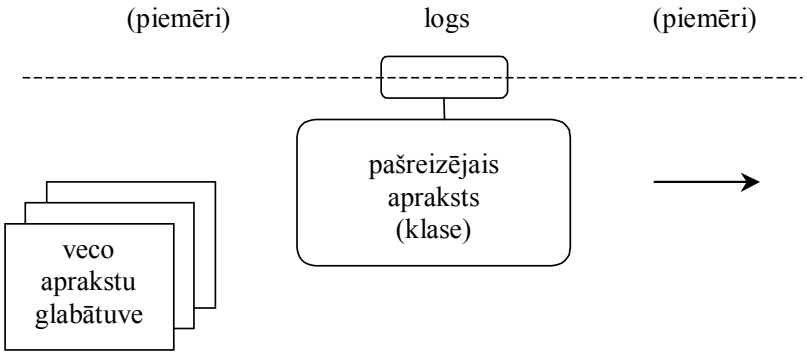
Darba pirmajā nodaļā aprakstīti klases atmiņas, inkrementālās apmācības daļējas piemēru atmiņas algoritmu darbības principi. Inkrementālajā apmācībā klase netiek pieņemta kā konstante, un algoritms mācās inkrementāli, apstrādājot datu plūsmas piemērus. Inkrementālās apmācības algoritms [24] tiek veikts piecos soļos:

1) iemācās likumus no apmācības piemēru kopas;
2) saglabā likumus un atmet apmācības piemērus;
3) izmanto jaunus likumus, lai prognozētu un virzītos tālāk;
4) kad pienāk jauns piemērs, iegūst jaunus likumus, izmantojot vecos likumus un jaunos piemērus;
5) iet uz otro soli.

Daļēja piemēru atmiņa nozīmē to, ka tikai daļa no piemēriem tiek paturēti atmiņā, jo efektīvai apmācībai vidēs ar slēpto kontekstu un klases dreifu nepieciešams tāds algoritms, kurš var atklāt izmaiņas klases aprakstā bez precīzi formulētas informācijas; kurš var atjaunoties pēc izmaiņām kontekstā un izmantot iepriekšējo pieredzi, kad vecais klases apraksts atkārtojas.

Pieaugot datu apjomiem un parādoties datu plūsmām ar trokšņainiem datiem un slēptajiem atribūtiem (piemēram, e-pasta ziņojumi, interneta lapas, žurnāla datnes), klasisko statisko klasifikācijas algoritmu izmantošana vairs nesniedz nepieciešamo

precizitāti. Efektīvāk ar datu plūsmām darbojas inkrementālās apmācības algoritmi, kuros apmācības process notiek nepārtraukti – soli pa solim. Tā kā inkrementālajā apmācībā klases apraksts laika gaitā mainās, apmācības algoritmam ir jāuzticas jaunākajiem piemēriem vairāk. Tas tiek realizēts, izmantojot „loga" metodi [46]. „Logs" ir apmācības kopas piemēru apakškopa (skat. 2. att.), kurus dotajā momentā izmanto algoritms.



2. att. Logs kustībā pa piemēru plūsmu

Vienkāršākajos gadījumos logam ir fiksēts izmērs, un vecākie piemēri tiek izslēgti, tikko pienāk jauns piemērs. Logam ar fiksētu izmēru „labs" loga izmērs nozīmē kompromisu starp ātru pielāgošanos (mazs logs) un labiem un stabiliem apmācības rezultātiem ar vai bez mazām izmaiņām konceptā (liels logs). Visefektīvākais ir logs ar adaptīvu izmēra koriģēšanas heiristiku, jo tad netiek saglabāts nevajadzīgi liels piemēru skaits un notiek pielāgošanās tekošajām klases apraksta izmaiņām.

Viena no lielākajām datu plūsmas apstrādes problēmām ir tā, ka interesējošā koncepcija var būt atkarīga no slēptā konteksta, kas ir atribūts vai atribūtu kopa, kura vērtību izmaiņas var izraisīt arī klases apraksta jeb koncepta nobīdi.

Promocijas darbā izvirzīti šādi uzdevumi inkrementālās apmācības algoritmu uzlabošanai darbam ar datu plūsmām un konceptu nobīdi:

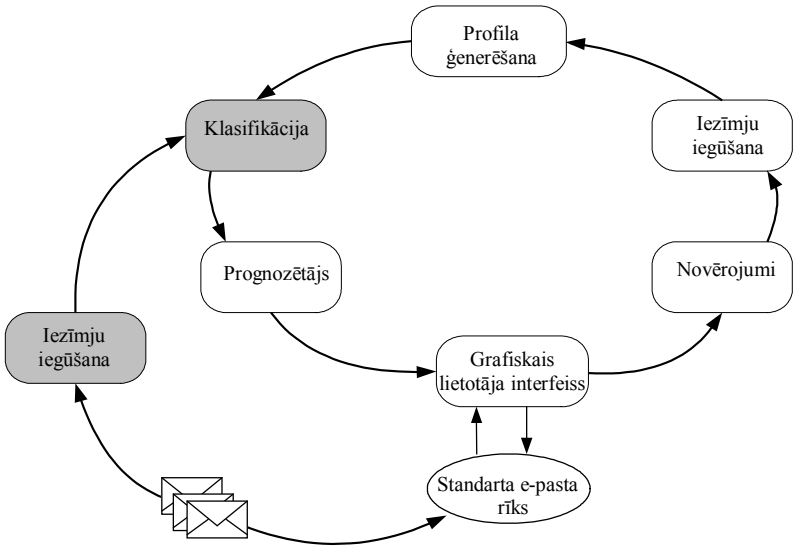
a) veikt inkrementālās apmācības algoritmu darbības pārbaudi vidēs ar trokšņainiem datiem un klases apraksta izmaiņu gadījumā;
b) izpētīt algoritmus, ar kuriem iespējams veikt e-pasta klasifikācijas uzdevumu un, ja nepieciešams, tos uzlabot un modificēt;
c) izpētīt un realizēt adaptīvas novērošanas loga izmēra pielāgošanas heiristikas metodi.

Otrā nodaļa

Otrajā nodaļā ir sniegta izpēte par interfeisa aģenta jēdzienu, arhitektūru, dažādiem aģentu veidiem, to pielietojumu. Aprakstīta konkrēta interfeisa aģenta *MAGI* [38] arhitektūra un tā izmantošana e-pasta ziņojumu klasifikācijā.

Otrajā nodaļā dots detalizēts praktiskā uzdevuma nostādnes apraksts, kā arī datu modelis, ar kura palīdzību tiek pārbaudīti visu darbā pētīto un izstrādāto metožu darbības principi un efektivitāte. Aprakstīti izstrādātie praktiskie eksperimenti *MAGI* apmācībā, izmantojot algoritmu *CN2* inkrementālās apmācības veidā.

Promocijas darbā aprakstīts elektronisko ziņojumu aģents *MAGI* (arhitektūras modeli skat. 2.att.), kurš lietotāja profilu saglabā likumu formā, lai prognozētu lietotāja darbību ienākošo ziņojumu izkārtošanā pa katalogiem. Klasifikācija šajā kontekstā ir darbība, kura aģentam būtu jāizpilda ar ziņojumu. Prognozēšanas etaps novērtē klasifikāciju (atbilstoši tai metodei, kas izmantota algoritmā, ar kuru tiek veikta klasifikācija) katram jaunajam ziņojumam un ģenerē uzticības novērtējumu (*confidence rating*) tam. Uzticības novērtējums katrai darbībai ir likumu skaits, kuri tiek iegūti ar vienu un to pašu darbību. Klasifikācija un uzticības novērtējums kopā veido aģenta lēmumu. Promocijas darbā no *MAGI* arhitektūras izmantoti iezīmju iegūšanas un klasifikācijas etapi (skat. 2.att. iekrāsoto).



3. att. E-pasta interfeisa aģenta arhitektūra

Darbā otrajā nodaļā izpētīts induktīvās secināšanas algoritms *CN2* [2], kurš pielietots e-pasta ziņojumu klasifikācijai [34].

Algoritms *CN2* darbojas iteratīvā veidā, katrā iterācijā meklējot nosacījumu, kas pārklāj lielu skaitu piemēru no vienas klases C_i un dažus no citām klasēm C_j, kur $C_i \neq C_j$. Nosacījums ir konjunkcija, kas veidota no atribūtu vērtībām. Kad piemērots nosacījums ir atrasts, tad algoritms piemērus, kurus nosacījums pārklāj, izslēdz no piemēru kopas un pievieno likumu „IF <nosacījums> THEN <noteikt klasi *C*>" likumu saraksta beigās. Process turpinās, kamēr nav iespējams atrast nevienu apmierinošu nosacījumu. Lai noteiktu katra jaunā nosacījuma kvalitāti un nozīmīgumu [2], tas tiek izskatīts un ierindots, izmantojot novērtējuma funkciju. Darbā izmantota

Laplasa kļūdas novērtējuma funkcija [1], kas aprēķina nosacījuma kvalitāti, veicot klasifikāciju:

$$LA = \frac{(n_c + 1)}{(n_{tot} + k)}, \tag{5}$$

kur n_{tot} – kopējais piemēru skaits, ko pārklāj likums;
n_c – piemēru skaits, ko pārklāj un pareizi klasificē likums;
k – klašu skaits uzdevumā.

Tālāk darba otrajā nodaļā aprakstīta algoritma *CN2* praktiska pielietošana e-pasta ziņojumu uzdevumā, imitējot aģenta uzdevumu. Detalizēti aprakstīts praktiskā uzdevuma datu modelis, kā arī piemēru ģenerēšana no e-pasta ziņojumiem, izmantojot iezīmju iegūšanas metodi, kur ar iezīmi tiek saprasts vārds no e-pastu aprakstošās informācijas: e-pasta adrese, temats, pamatteksts [30].

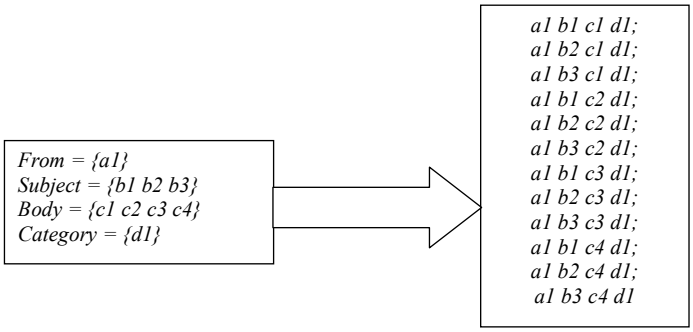
Kā pamatdati praktiskajā uzdevumā izmantota e-pasta korespondence no foruma par programmas *Lotus Notes* specifiskām funkcijām, ierobežojumiem un pielietošanu. Visi ziņojuma teksti ir latviešu valodā, izņemot specifiskus vārdus – datorterminus.

Elektroniskie ziņojumi no foruma aprakstīti ar četriem atribūtiem [34] un klases pazīmi:

- *From* – sūtītāja e-pasta adrese (bez specifiskā simbola "@" un domēna vārda);
- *Subject* – ziņojuma temats, aprakstīts ar 3 visbiežāk sastopamajiem vārdiem;
- *Body* – ziņojuma pamatteksts, aprakstīts ar 4 visbiežāk sastopamajiem vārdiem;
- *Category* – ziņojuma kategorija, viena no šādām: jautājums, atbilde, ieteikums, informācija;
- *Classes* – darbība: "pārsūtīt" vai "dzēst" jauno ienākošo ziņojumu.

Ģenerējot piemērus no e-pasta ziņojumiem, visbiežāk sastopamo vārdu iegūšanai no laukiem *Subject* un *Body* tika pielietots *Levenšteina* attāluma algoritms. *Levenšteina* attālums ir līdzīguma mērs starp divām virknēm, attiecīgi starp avota virkni un mērķa virkni. Jo lielāks *Levenšteina* attālums, jo atšķirīgākas ir virknes. Programmprodukts *levenstain.jsp*, kas balstīts uz *Levenšteina* attāluma algoritmu [12], tika pielāgots un izmantots tieši šī uzdevuma risināšanai.

Praktiskā uzdevuma apmācības un testa kopas piemēri tika veidoti, kombinējot vārdus no katra e-pasta ziņojumu aprakstošā lauka (skat. 4.att.).



4.att. Apmācības piemēru konstruēšanas mehānisms

Pēc 40 e-pasta ziņojumu apstrādes tika iegūti 410 piemēri; 160 no tiem pieder klasei "dzēst", bet 250 - klasei "pārsūtīt".

Induktīvās secināšanas algoritms *CN2* ir pielietots, lai inducētu likumus no apmācības kopas piemēriem. Praktisks uzdevums risināts, izmantojot brīvprogrammu *CN2* versija 6.1 [3].

Apmācības process tika sadalīts soļos, t.i. vairākkārtīgi darbinot klasifikācijas algoritmu *CN2*, lai imitētu aģenta inkrementālo apmācību. Sākotnējos eksperimentos, apmācības procesā izmantojot algoritmu *CN2*, tika iegūts 21 likums [34]. Šie likumi tika pārbaudīti ar 28 testa kopas piemēriem. Likumi pārklāj lielu daļu testa kopas piemēru, tomēr to precizitāte šajos eksperimentos ir no 46% līdz 57%, kas nav augsts rādītājs.

Lai uzlabotu praktisko eksperimentu rezultātus, tika veikts vēl viens eksperiments, kurā apmācības procesā pielietota pieckārtēja šķērsvalidācijas [17] metode. Vidējā likumu precizitāte eksperimentā ar šķērsvalidāciju iegūtajai likumu kopai ir 83.60%.

Sākotnējā motivācija, kādēļ aģenta apmācības procesā izmantots algoritms *CN2*, ir tāda, ka šis algoritms ģenerē cilvēkam viegli saprotamus likumus, izpildot induktīvo secināšanu no apmācošās kopas piemēriem, kas satur specifiskas iezīmes. Tomēr *CN2* ir statiskās apmācības algoritms un, kaut arī tas tika darbināts vairākkārtīgi, lai imitētu inkrementālo apmācību, e-pasta ziņojumu klasificēšanas uzdevumā, kas ir uzdevums ar datu plūsmu, piemērotāks ir inkrementālās apmācības algoritms.

Trešā nodaļa

Viens no uzdevumiem inkrementālās apmācības algoritmu uzlabošanai darbam ar datu plūsmām un konceptu nobīdi ir izstrādāt inkrementālās apmācības algoritma *MLII* hibrīdu, lai uzlabotu algoritma efektivitāti darbam ar dažādiem datu tipiem un paplašinātu tā lietojamību, kā arī, lai pierādītu izstrādātā algoritma lietojamību praktiskajā uzdevumā par e-pasta ziņojumu klasifikāciju.

Trešajā nodaļā aprakstīts daudzkārtainais inkrementālās secināšanas algoritms *MLII* [49], kurš promocijas darba gaitā saistīts ar induktīvās secināšanas algoritmu *CN2* [2]. Oriģinālajā *MLII* versijā autori apmācības procesā izmanto heiristiskās pārklāšanas algoritmu *HCV* [50], kurš balstīts uz paplašinātās matricas (*extension*

matrix) pieeju un rezultātā ģenerē konjunktīvo likumu formulu un ir piemērots tikai simbolisko datu [29] klasificēšanai. Taču algoritms *CN2* veido likumus cilvēkam saprotamākā formā un apstrādā gan skaitliskus, gan simboliskus datus (*HCV* darbojas tikai ar simboliskiem datiem). Tāpēc, lai uzlabotu *MLII* veiktspēju, promocijas darbā tika izstrādāts hibrīds *HMLII* [35], kur apmācības procesā izmantots algoritms *CN2*.

Darba trešajā nodaļā detalizēti aprakstīta daudzkārtainā inkrementālās secināšanas algoritma *MLII* darbība [49], kas sastāv no trīs posmiem:

A. Sākotnējo datu kopu sadalīšana noteikta skaita kārtās (apakškopās) ar aptuveni vienādu izmēru, gadījuma veidā sajaucot datus;
B. Likumu kopas iegūšana apmācības procesā no pirmās piemēru apakškopas, izmantojot vispārināšanas (*generalization*) algoritmu;
C. Pāreja uz iepriekšējās likumu kopas attīrīšanu (*refinement*). Šī pāreja tiek izpildīta ar pāraprakstīšanas operatoru, sauktu par reducēšanu (*reduction*), kas iegūst jaunu raksturīgo piemēru kopu, pārbaudot otrajā posmā iegūto likumu kopu ar nākošās datu apakškopas piemēriem.

Datu sadalīšana (A posms) samazina troksni oriģinālajā apmācības kopā un vienmērīgi sadala dažādu klašu piemērus.

Vispārināšana (B posms) tiek lietota sākotnējās informācijas saspiešanai jeb kompresijai. Vispārināšana ietver apmācības piemēru apakškopu ar kādu atsevišķu konceptu ievērošanu, piemēru identificēšanu un tad koncepta definīcijas formulēšanu, balstoties uz kopējām iezīmēm. Algoritmā *MLII* ir piemērota diskriminantā vispārināšana ar izslēgšanu [49]. Vispārināšanas likums ir apraksta transformācija vispārīgāka veida definīcijā.

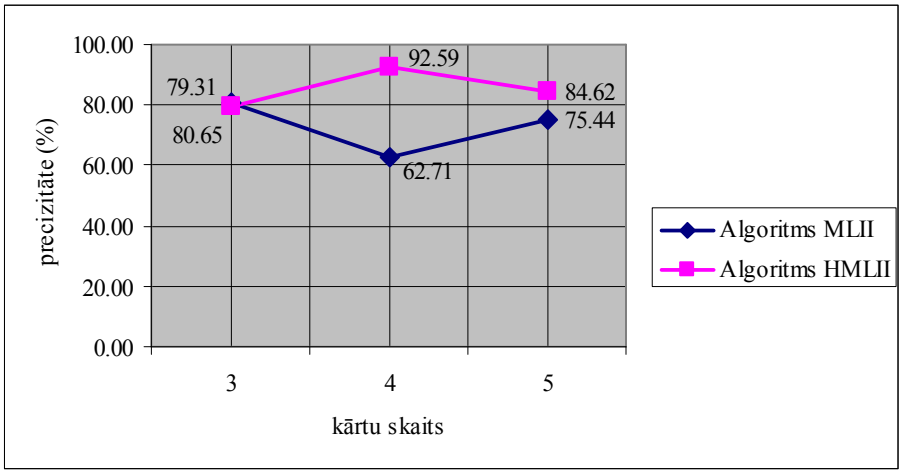
Reducēšana (C posms) iegūst jaunu kopu uzvedības piemēru, pārbaudot iepriekšējā soļa likumu kopas uzvedību ar nākošo datu kopu.

Otrajā posmā *MLII* darbina kādu eksistējošu induktīvās secināšanas algoritmu, lai iegūtu sākotnējās likumu kopas. Darbā tika izvēlēts algoritms *CN2,* jo tas ģenerē klasifikatoru *IF...THEN* likumu formā un nodrošina iespēju apstrādāt skaitliskus un simboliskus datus.

Tālāk darba trešajā nodaļā dots algoritmu *MLII* un *HMLII* praktiskais pielietojums [35] e-pasta ziņojumu klasifikācijai. Lai izpētītu un salīdzinātu šo algoritmu darbību, tika veikti trīs eksperimenti [32] ar dažādu kārtu skaitu katrā.

No pirmās datu apakškopas jeb kārtas apmācībā, pielietojot algoritmu *CN2,* tika iegūti likumi, darbinot apmācību ar programmu *Sipina for Windows - Research version* [44]. Iegūtajiem likumiem pielietotas vispārināšanas un reducēšanas operācijas. Tādā pat veidā apstrādātas pārējās apakškopas visos eksperimentos.

Likumi, kuri iegūti eksperimenta kārtās, tika kombinēti kopā (veidoti metalikumi), un ar testa kopas piemēriem *MS Excel* programmā tika veikta to testēšana. Trešās nodaļas beigās aprakstīti eksperimentu rezultāti (skat. 5. att.) un veikta to analīze.



5. att. Algoritmu *MLII* un *HMLII* testa rezultāti

Salīdzinot ar iepriekšējo interfeisa aģenta e-pasta ziņojumu klasifikācijas uzdevuma risinājumu [34], kur inkrementālā apmācība tika nodrošināta, apstrādājot apakškopas pa soļiem, bez jebkādas papildus likumu apstrādes, *HMLII* uzrāda labākus rezultātus (pozitīvā precizitāte līdz 92.59%) un dod mazāku skaitu nepareizi klasificētu piemēru. Hibrīdā algoritma *HMLII* apmācības rezultātā iegūto klasifikatoru precizitāte ir nozīmīgi lielāka, nekā risinot to pašu e-pastu klasifikācijas uzdevumu, izmantojot oriģinālo algoritmu *MLII*, jo tad pozitīvā precizitāte tika iegūta no 62.71% līdz 80.65%. Bez tam pēc eksperimentu rezultātiem var secināt, ka *HMLII* kārtu skaita palielināšana neuzlabo likumu precizitāti [36].

Ceturtā nodaļa

Viens no promocijas darba uzdevumiem ir izpētīt un realizēt adaptīvas novērošanas loga izmēra pielāgošanas heiristikas metodi, un pārbaudīt tās efektivitāti inkrementālās apmācības algoritmā *FLORA2*.

Ceturtā nodaļa veltīta daļējas inkrementālās atmiņas algoritmam *FLORA2* [46], un tajā pētīts algoritma darbības princips, *FLORA2* izmantotā novērošanas loga izmēra pielāgošanas heiristika, kā arī aprakstīti *FLORA2* praktiskā pielietojuma eksperimenti e-pastu klasifikācijā.

FLORA2 ir viens no *FLORA* algoritmu saimes algoritmiem [27]. *FLORA* algoritmi apmācības procesu veic, izmantojot trīs aprakstu kopas. *ADES* ir visu to aprakstu kopa, kas ir konsekventi (pārklāj tikai pozitīvus piemērus), *PDES* ir kandidātu aprakstu kopa, kas kopumā ņemot ir sakomplektēta, bet nav konsekventa (tā pārklāj pozitīvos piemērus un arī kādu negatīvo), un *NDES* ir konsekvents negatīvo piemēru apraksts (tas nepārklāj nevienu pozitīvu piemēru).

Katrs apraksts var tikt interpretēts kā izteikums *DNF* (disjunktīvās normālformas) formā. Disjunktīvā normālforma ļauj noteikt, vai izteikums ir pretrunīgs. Izteikuma disjunktīvā normālforma ir konjunkciju disjunkcija [19]. Šo konjunkciju locekļi var būt gan patiesi, gan aplami.

Tālāk ceturtajā nodaļā detalizēti pētīta *FLORA2* izmantotā dinamiskā loga izmēra koriģēšanas heiristika. Algoritms *FLORA2* [46] automātiski nosaka un koriģē tā loga izmēru apmācības laikā. Pamatideja ir sašaurināt logu (un aizmirst vecos piemērus), kad gaidāma klases apraksta nobīde, un saglabāt loga izmēru fiksētu, kad klase izskatās stabila. Loga izmēram ir pakāpeniski jāaug, kamēr tiek veidots stabils klases apraksts. Dinamiska loga koriģēšana tiek veikta pēc algoritma [47], kur loga izmērs ir atkarīgs gan no pārklāto pozitīvo piemēru skaita N, gan no aprakstu kopas *ADES* izmēra S (skat. 2. tabulu). Parametru uzstādījumi $lc = 1.2$ un $hc = 4$ un $p = 70\%$ ir algoritma [47] autoru definētas konstantes.

Sākotnēji algoritma *FLORA2* darbības principi tika apgūti un pētīti, risinot vienkāršu loģiskās funkcijas aproksimācijas uzdevumu [28]. Darba ceturtajā nodaļā realizēts praktisks uzdevums par e-pasta ziņojumu klasificēšanu. Eksperimenti tika veikti ar mērķi izpētīt algoritma *FLORA2* darbības principus un loga izmēra pielāgošanas heiristiku [33], kā arī salīdzināt tekošos rezultātus ar tiem rezultātiem, kas iegūti iepriekšējos pētījumos, kad tika izmantots interfeisa aģents ar algoritmu *CN2*, daudzkārtainais inkrementālās secināšanas algoritms un tā hibrīds.

2. tabula

Loga koriģēšanas heiristika

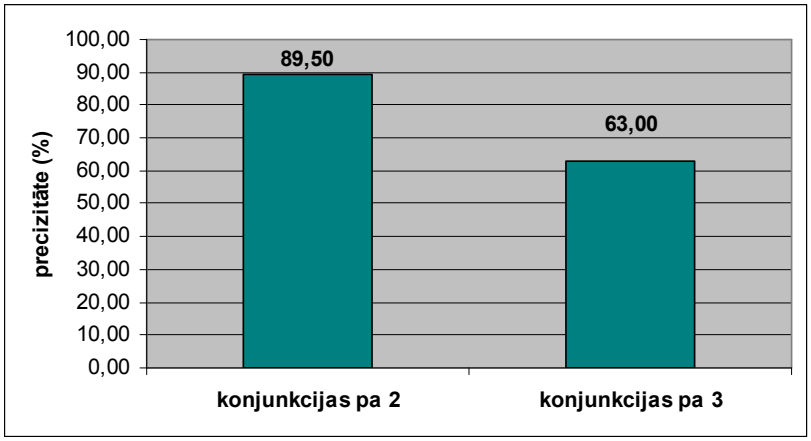
Apzīmējumi:	
N ... pozitīvo piemēru skaits, ko pārklāj *ADES*	
S ... *ADES* kopas izmērs	
Acc ... tekošā precizitāte (kas novērota jaunākajos klasifikācijas mēģinājumos)	
$\|W\|$ ... loga izmērs	
lc ... *ADES* zemākais pārklāšanas slieksnis	
hc ... *ADES* augstākais pārklāšanas slieksnis	
p ... precizitātes pieņemtais slieksnis	
Algoritms:	
Ja ($N/S < lc$) vai (($Acc < p$) un (*Acc samazināšana*))	/* gaidāms klases apraksta dreifs */
tad $L := 0.2 * \|W\|$	/* samazināt logu par 20% */
citādi ja $(N/S > 2 * hc)$ un $(Acc > p)$	/* ļoti stabili */
tad $L := 2$	/* samazināt logu par 1 */
citādi ja $(N/S > hc)$ un $(Acc > p)$	/* pietiekoši stabili */
tad $L := 1$	/* saglabāt logu nemainīgu */
tad $L := 0$	/* palielināt logu par 1 */

Par apmācības datiem tika izmantota e-pasta korespondence. Praktiskā uzdevuma datu modelis aprakstīts promocijas darba otrajā nodaļā.

Sākotnējos eksperimentos tika ģenerēti likumi konjunkciju formā pa diviem atribūtiem [31], lai iegūtu tādus likumus, kuri ir pietiekoši vispārīgi un tādējādi pareizi klasificē vairāk testa kopas piemēru. Eksperimentos likumu klasifikācijas precizitāte tika iegūta līdz 72%.

Tad tika veikti divi eksperimenti, kuros konjunkcijas tika ģenerētas ar diviem un trim atribūtiem, un piemēri gadījuma veidā 10 reizes tika sajaukti, lai mainītu to

secību. Šajos praktiskajos eksperimentos [26] tika izpildīta inkrementālā apmācība un aizmiršana, izmantojot dinamisku loga izmēra kalkulāciju. Rezultātā konjunkcijām pa divi tika iegūts vidējais likumu skaits 181, bet konjunkcijām pa trīs – 291. Likumu klasifikācijas precizitāte tika iegūta robežās no 63% līdz 89.5% (skat. 6. att.).



6. att. Algoritma *FLORA2* testa rezultāti

Ceturtās nodaļas rezultāti rāda, ka *FLORA2* ģenerē lielu daudzumu likumu konjunkciju formā [31], arī „kandidātu" likumus, kuri atbilst abām klasēm un ir noderīgi tālākā apmācības procesā. Salīdzinot ar iepriekšējiem eksperimentiem [35], var secināt, ka e-pasta klasifikācijas uzdevumā algoritms *HMLII* parādīja labāku rezultātu testēšanas procesā, bet algoritms *FLORA2* izveidoja daudz vairāk likumus. Var secināt, ka e-pasta klasifikācijas uzdevumam piemērotāks ir algoritms *HMLII.*

Piektā nodaļa

Vispirms piektajā nodaļā sniegts apraksts par algoritmam *HMLII* izmantoto un izstrādāto programmnodrošinājumu. Praktiskajos e-pasta ziņojumu klasifikācijas eksperimentos ar algoritmu *HMLII* kārtu apstrāde tika veikta ar autores izstrādāto aplikāciju programmā *Microsoft Office Excel 2003*. Apmācības procesa posms ar algoritmu *CN2* tika veikts pielietojot programmu *Sipina for Windows - Research version* [44]. Likumu attīrīšana un testēšana eksperimentos ar *HMLII,* tika realizēta programmā *Microsoft Office Excel 2003* izstrādātajā aplikācijā.

Tālāk piektajā nodaļā aprakstīti promocijas darba autores izstrādātās programmas *EmailFlora* darbības principi, izmantošanas nolūks un pielietojums. Programma *EmailFlora* balstīta uz algoritmu *FLORA2* ar dinamisku loga izmēra koriģēšanas heiristiku un paredzēta e-pasta ziņojumu klasifikatora automatizētai izveidei. Jaunā programma *EmailFlora* izstrādāta *IBManager 3* datubāzē *InterBase* kā datu bāzes iegultās procedūras, bet lietotāja interfeiss ir izstrādāts *Delphi* 7.0 vidē.

Promocijas darba **nobeigumā** tiek aplūkoti darba rezultāti e-pasta ziņojumu klasifikācijas uzdevumā (skat. 3. tabulu). Viszemākā precizitāte ir interfeisa aģentam, balstītam uz algoritmu *CN2*. Tas izskaidrojams ar to, ka algoritms *CN2* ir statiskās apmācības algoritms un tika darbināts vairakkārt, lai imitētu inkrementālo apmācību. Vislabākos rezultātus uzrādīja hibrīdais algoritms *HMLII,* kura apmācības procesā iegūtā klasifikatora testēšanas precizitāte ir no 79.31% līdz 92.59%. Algoritms *HMLII* izmanto daudzkārtaino inkrementālās apmācības metodi, pielietojot datu sadalīšanu, vispārināšanu un attīrīšanu. Ar oriģinālo algoritmu *MLII* pozitīvā precizitāte iegūta no 62.71% līdz 80.65%. Algoritms *FLORA2* e-pasta ziņojumu klasificēšanas uzdevumā ģenerēja vislielāko skaitu likumu, tai skaitā arī „kandidātu" likumus, kas klasificē abu klašu piemērus un var būt noderīgi turpmākā apmācības procesā. Tomēr klasifikācijas precizitāte algoritmam *FLORA2* ir tikai apmierinoša – no 63% līdz 89.50%.

3. tabula

Eksperimentu kopsavilkums

Algoritms	**Vidējais likumu skaits**	**Precizitāte**
CN2	73	83.60%
MLII	5	80.65%
HMLII	10	92.59%
FLORA2	127	89.50%

Pēc iegūtajiem rezultātiem, var secināt, ka jaunā hibrīdā algoritma *HMLII* izstrāde ir lietderīga, jo tā paplašina algoritma *MLII* lietojamību (apstrādā gan skaitliskus, gan simboliskus datus), paātrina ātrdarbību un sniedz augstu precizitāti praktiskajā e-pasta ziņojumu klasifikācijas uzdevumā (ar datu plūsmu, trokšņainiem datiem un klases apraksta nobīdi).

Pielikumā pievienota informācija par eksperimentos izmantotajiem datiem un autores izstrādātās programmas *EmailFlora* pirmteksts. Bez tam pievienots arī autores pielāgotās Levenšteina attāluma aprēķina programmas *levenstain.jsp* pirmteksts programmēšanas valodā *Java*.

PROMOCIJAS DARBA GALVENIE REZULTĀTI

Promocijas darbā pētīti un analizēti vairāki induktīvās klasifikācijas algoritmi ar inkrementālu apmācību – to darbības principi, priekšrocības un trūkumi. Darbā realizēts inkrementālās apmācības algoritmu praktisks pielietojums e-pasta ziņojumu klasifikācijas uzdevumā. Promocijas darba galvenie rezultāti ir šādi:

1. Ir izskatīti vairāki induktīvās klasifikācijas algoritmi, kuri balstīti uz inkrementālo apmācību. Izpētīta un analizēta inkrementālās apmācības algoritmu darbība vidēs ar trokšņainiem datiem un klases apraksta izmaiņu gadījumā – konkrēti, e-pasta ziņojumu apstrādē un klasifikācijā. Secināts, ka šādi induktīvās secināšanas algoritmi ir piemēroti darbam ar datu plūsmām vidēs, kur novērojama zināšanu novecošanās.
2. Veikta e-pasta ziņojumu klasifikācijas interfeisa aģenta metodes *MAGI* izpēte, kura klasifikatora veidošanas etapā balstīta uz statiskās klasifikācijas algoritmu *CN2*. Promocijas darbā izmantoti *MAGI* iezīmju iegūšanas un klasifikācijas etapi. Secināts, ka interfeisa aģenta metode ir piemērota e-pasta ziņojumu klasifikācijai.
3. Realizēta eksistējošā daudzkārtainā inkrementālās secināšanas algoritma *MLII* hibridizācija (izstrādāts algoritms *HMLII*), lai uzlabotu oriģinālo *MLII,* balstītu uz heiristiskās pārklāšanas algoritma *HCV* pielietošanu. Kā statiskās apmācības algoritmu piedāvāts izmantot induktīvās secināšanas algoritmu *CN2*, kurš uzlabo *MLII* algoritma efektivitāti, ģenerējot klasifikatoru vieglāk saprotamā likumu formā, un apstrādā gan skaitliskus, gan simboliskus datus. Algoritma *HMLII* praktiskai realizācijai datu un likumu apstrādei un klasifikatora testēšanai izstrādāta aplikācija programmā *MS Excel*.
4. Izpētīts inkrementālās apmācības algoritms *FLORA2*, kurš darbojas ar datu plūsmām un apmācības procesā pielieto adaptīvu novērošanas loga izmēra pielāgošanas heiristiku.
5. Veikta inkrementālās apmācības algoritma *FLORA2* ar adaptīvu datu novērošanas loga izmēra heiristiku realizācija. Izstrādāta e-pasta ziņojumu klasifikatora iegūšanas programma *EmailFlora*, kurā iespējams uzstādīt apmācības datu sākumkopas izmēru un likumu konjunkciju atribūtu skaitu (no viens līdz trīs). Programma paredzēta e-pasta ziņojumu klasifikatora ģenerēšanai un testēšanai.
6. Pētīts un analizēts pieejamais klasifikācijas algoritmu programmnodrošinājums, tā priekšrocības un trūkumi. Par piemērotām tika izskatītas šādas programmas darbā izmantotā statiskās klasifikācijas algoritma *CN2* darbināšanai: *CN2* versija 6.1, *Sipina for Windows*.

7. Realizēts praktisks e-pasta ziņojumu klasifikācijas uzdevums ar šādām inkrementālās apmācības metodēm: interfeisa aģenta metodi; daudzkārtaino inkrementālās secināšanas algoritmu *MLII*; algoritma *MLII* hibrīdu *HMLII* un inkrementālās apmācības algoritmu *FLORA2* ar adaptīvu datu novērošanas loga izmēra heiristiku. Visos gadījumos veikta iegūto rezultātu salīdzinoša analīze un izdarīti secinājumi par algoritmu efektivitāti un piemērotību konkrētajam uzdevumam. Secināts, ka piemērotākais e-pasta ziņojumu klasifikācijai ir algoritms *HMLII*, kurš praktiskajos eksperimentos uzrādīja vislabāko precizitāti.

BIBLIOGRĀFISKAIS SARAKSTS

1. Clark P., Boswell R. Rule induction with CN2: Some Recent Improvements // Machine Learning – EWSL-91: Proceedings of the 5th European Conference. - Berlin: Springer – Verlag, 1991. – P. 151 - 163.

2. Clark P., Niblett T. The CN2 Induction algorithm // Machine Learning. – 1989. - Vol. 3, No. 4 . - P. 261 - 283.

3. Clark P. Software: CN2 - Rule induction from examples - URL: http://www.cs.utexas.edu/users/pclark/software#cn2 – Last viewed February 2008.

4. Data Streams: Models and Algorithms / Aggarwal Ch.C. – Springer Science+Business Media, LCC, 2007. – 354 p. – ISBN-10: 0-387-28759-0, ISBN-13: 978-0-387-28759-1.

5. Datu ieguve: Pamati / A.Sukovs, L.Aleksejeva, K.Makejeva, A.Borisovs. - Rīga: RTU, SIA "Drukātava", 2007. - 130 lpp.

6. Delany S.J., Cunningham P., Tsymbal A., Coyle L. A case-based technique for tracking concept drift in spam filtering // Journal of Knowledge Based Systems Vol.18, Issue 4, Dublin Institute of Technology, 2005. - P. 187 - 192. - URL: http://www.csi.ucd.ie/UserFiles/publications/1146565915422.pdf – Last viewed May 2008.

7. Esposito F., Ferilli F., Fannizzi N., Basile T.M.A., Di Mauro N. Incremental learning and concept drift in INTHELEX // Intelligent Data Analysis, Vol. 8., Issue 3. – IOS Press Amsterdam, The Netherlands, 2004. – P. 213 - 237. – ISSN 1088-467X.

8. Experience with Learning Agents which Manage Internet-Based Information / P. Edwards, D. Bayer, C.L. Green, T.R. Payne. – AAAI Spring Symposium on Machine Learning for Information Access, 1996. - P. 31 - 40. - URL: http://www2.parc.com/istl/projects/mlia/papers/edwards.ps – Last viewed April 2008.

9. Farhoodi F., Fingar P. Competing for the Future with Intelligent Agents - URL: http://home1.gte.net/pfingar/agents_doc_rev4.htm – Last viewed October 2007.

10. Feature Selection for Knowledge Discovery and Data Mining / Liu H., Motoda H. - Kluwer Academic Publishers, USA, 1998, Second Printing 2000. - 207 p. – ISBN 0-7923-8198-X.

11. Ferrer–Troyano F., Aguilar–Ruiz J. S., Riquelme J. C. Incremental Rule Learning based on Example Nearness from Numerical Data Streams // Proceedings of the 2005 ACM Symposium on Applied computing, USA: Santa Fe, 2005. - P. 568 – 572. – ISBN:1-58113-964-0. - URL: http://www.upo.es/eps/aguilar/papers/ferrer05sac.pdf – Last viewed April 2008.

12. Gilleland M. Levenshtein Distance, in Three Flavors. Merriam Park Software - URL: http://www.merriampark.com/ld.htm – Last viewed August 2008.
13. Goonatilake S., Khebbal S. Intelligent Hybrid Systems, 1st edition. John Wiley & Sons, Inc., New York, USA, 1994. – 635 p. - ISBN:0471942421.
14. Han J., Kamber M. Data Mining: Concepts and Techniques. 2nd Edition. – San Francisco etc.: Morgan Kaufman, 2006. – 800 p.
15. Klinkenberg R. Learning drifting concepts: Example selection vs. example weighting // Intelligent Data Analysis, Vol. 8., Issue 3. – IOS Press Amsterdam, The Netherlands, 2004. – P. 281 - 300. – ISSN 1088-467X.
16. Klinkenberg R., Renz I. Adaptive Information Filtering: Learning in the Presence of Concept Drifts // AAAI-98/ICML-98 Learning for Text Categorization. – 1998. - P. 1 - 4. - URL: http://www-ai.cs.uni-dortmund.de/DOKUMENTE/klinkenberg_renz_98a.pdf – Last viewed September 2006.
17. Kohavi R. A study of cross-validation and bootstrap for accuracy estimation and model selection // Proceedings of the 14th International Joint Conference on Artificial Intelligence (IJCAI-95). – San Mateo, CA: Morgan Kaufman. – 1995. P. 1137 - 1143.
18. Kubat M. Flexible Concept Learning in Real-Time Systems // Journal of Intelligent and Robotic Systems 8. - Netherland: Kluwer Academic Publishers, 1993. - P. 155 - 171.
19. Logic. An Introduction. Second Edition / Churchill R.P. - New York: St.Martin's Press, Inc., 1990. – 635 p.
20. Maes P. Agents that Reduce Work and Information Overload // Communications of the ACM, Volume 37, Issue 7. - July 1994. – P. 30 - 40. - URL: http://www.cs.brandeis.edu/~cs125a/content/agentsmaes.doc – Last viewed April 2008.
21. Maloof M.A. On-line learning with Partial Instance Memory // IJCAI-01 Workshop on Learning from Temporal and Spatial Data, Seattle, WA, 2001.
22. Maloof M.A., Michalski R.S. A Partial Memory Incremental Learning Methodology and its Application to Computer Intrusion Detection // Reports of the Machine Learning and Inference Laboratory, MLI 95-2: Machine Learning and Inference Laboratory, Department of Computer Science, George Mason University, Fairfax, VA, 1995. - P. 6 - 13. - URL: http://www.mli.gmu.edu/papers/91-95/95-02.PDF – Last viewed April 2008.
23. Maloof M., Michalski R. AQ-PM A System for Partial Memory Learning // Intelligent Information Systems VIII, Proceedings of The Workshop held in Ustron, Poland, June 14-18, 1999. - P. 70 - 79.
24. Maloof M., Michalski R. Incremental Learning with Partial Instance Memory // Artificial Intelligence, Vol. 154, Issue 1-2 2004. - P. 95 - 126. – ISSN:0004-3702.

25. Maloof M.A., Michalski R.S. Selecting Examples for Partial Memory Learning // Machine Learning – 2000. – Vol. 41, Issue 1,. Kluwer Academic Publishers Hingham, MA, USA - P. 27 - 52. - URL: http://www.mli.gmu.edu/papers/96-2000/00-11.pdf – Last viewed April 2008.

26. Misina S., Aleksejeva L. A comparative analysis of classification methods with incremental learning in the e-mail filtering task // Scientific Proceedings of Riga Technical University. Series 5. Computer science. Information technology and management science. – Vol. 36. (2008). - P. 116 - 124. - ISSN 1407-7493.

27. Misina S., Alexeyeva L. Efficiency analysis of on-line classification rule construction methods // Scientific Proceedings of Riga Technical University. Series 5. Computer science. Information technology and management science. – Vol. 14. (2003). - P. 122 - 137. - ISSN 1407-7493.

28. Misina S., Aleksejeva L. Efficiency analysis of real-time classification rule construction methods // 10th International Conference on Soft Computing MENDEL 2004. Brno, Czech Republic, 16 - 18 June, 2004. – Brno: Brno University of Technology, 2004. - P. 182 - 187. - ISBN 80-214-2676-4.

29. Misina S., Alexeyeva L. Inductive inference algorithms in decision making task // Scientific Proceedings of Riga Technical University. Series 5. Computer science. Information technology and management science. – Vol. 5. (2001). - P. 136 - 143. - ISSN 1407-7493.

30. Misina S., Alexeyeva L. Inductive inference algorithms in e-mail messages filtering // Scientific Proceedings of Riga Technical University. Series 5. Computer science. Information technology and management science. – Vol. 20. (2004). - P. 10 - 18. - ISSN 1407-7493.

31. Misina S. Example subset size adaptation heuristic in incremental learning // Scientific Proceedings of Riga Technical University. Series 5. Computer science. Information technology and management science. – Vol. 28 (2006). - P. 107 - 114. - ISSN 1407-7493.

32. Misina S. Incremental learning based on non-incremental induction // International Conference on Operational Research: Simulation and Optimization in Business and Industry. Tallinn, Estonia, May 17 - 20, 2006. – Kaunas: Technologija, 2006. - P. 239 - 242. - ISBN 9955-25-061-5.

33. Misina S. Incremental learning for e-mail classification // Computational Intelligence, Theory and Applications. Proceedings of International Conference 9th Fuzzy Days in Dortmund, Germany, Sept. 18 - 20, 2006. Vol. 38. - Berlin Heidelberg: Springer, 2006. - P. 545 - 553. – ISSN 1615-3871.

34. Misina S., Aleksejeva L. Inductive inference algorithms in e-mail messages filtering // 11th International Conference on Soft Computing MENDEL 2005, Brno, Czech Republic, 15 - 17 June, 2005. – Brno: Brno University of Technology, 2005. – P. 63-68. – ISBN 80-214-2961-5.

35. Misina S. Inductive inference algorithm in Multi-layer incremental learning // Scientific Proceedings of Riga Technical University. Series 5. Computer science. Information technology and management science. – Vol. 23. (2005). - P. 41 - 47. – ISSN 1407-7493.

36. Misina S. Multi-layer incremental learning linked to nonincremental induction // International Journal of Information Technology and Intelligent Computing. Vol. 1, No. 3. – Academy of Humanities and Economics, Lodz, Poland in cooperation with IEEE Computational Intelligence Society Poland Chapter, 2006. – P. 477 - 486. - ISSN 1895 – 8648.

37. Payne R. T., Edwards P. Interface Agents that Learn: An Investigation of Learning Issues in a Mail Agent Interface // Applied Artificial Intelligence, Vol. 11, No. 1, January 1997. - P. 1 - 32. - URL: http://users.ecs.soton.ac.uk/trp/Publications.html – Last viewed April 2008.

38. Payne R. T. Learning Email Filtering Rules with Magi - A Mail Agent Interface // master's thesis, Dept. of Computing Science, Univ. of Aberdeen, Scotland. – 1994. - URL: http://users.ecs.soton.ac.uk/trp/Pubs/msc_thesis.pdf – Last viewed September 2004.

39. Quinlan J. R., Induction of decision trees // Machine Learning 1. - 1986. - Vol. 1 – P. 81 - 106.

40. Quinlan J. R., C4.5: Programs for Machine Learning – San Mateo, CA: Morgan Kaufman. – 1993. – 302 p.

41. Rozsypal A., Kubat M. Association mining in time-varying domains // Intell. Data Analysis, Vol. 9, No. 3, 2005. – P. 273 - 288.

42. Russel S. J., Norvig P. Artificial Intelligence. A Modern Approach. 2nd Edition. - Pearson Education, Inc., Upper Saddle River, New Jersey, USA, 2003. - 1081 p. – ISBN 0-13-080302-2.

43. Saganicoff M. Density-adaptive learning and forgetting // University of Pennsylvania for Research of Cognitive Science Technical Report No. IRCS 93-95, USA, 1993. – 10 p. – URL: http://repository.upenn.edu/cgi/viewcontent.cgi?article=1198&context=ircs_reports. – Last viewed October 2008.

44. Sipina for Windows - Research version Laboratoire ERIC. - URL: http://eric.univ-lyon2.fr/~ricco/sipina.html – Last viewed November 2007.

45. Utgoff P.E. Incremental induction of decision trees // Machine Learning. – 1989. – Vol. 4., Kluwer Academic Publishers, Hingham, MA, USA - P. 161 - 186. - URL: http://www.cs.umass.edu/~utgoff/papers/mlj-id5r.pdf. – Last viewed October 2008.

46. Widmer G. Combining robustness and flexibility in learning drifting concepts // Proceedings of the 11th European Conference on Artificial Intelligence, Chichester: John Wiley&Sons, 1994. - P. 468 - 472.

47. Widmer G., Kubat M. Learning Flexible Concepts from Streams of Examples: FLORA2 // Proceeding of 10th European Conference on Artificial Intelligence ECAI 92, Vienna, Austria, 3 – 7 August, 1992. - John Wiley&Sons, 1992. - P. 463 - 467.

48. Widmer G., Kubat M. Learning in the Presence of Concept Drift and Hidden Contexts // Machine Learning. - 1996. – Vol. 23. Issue 1, Kluwer Academic Publishers Hingham, MA, USA - P. 69 - 101. - ISSN:0885-6125. - URL: http://www.miami.edu/eng-electrical/mkubat/Publications/germljfinal.ps – Last viewed September 2006.

49. Wu X., Lo W.H.W. Multi-Layer Incremental Induction // Proceedings of the 5th Pacific Rim International Conference on Artificial Intelligence, Springer – Verlag, London, UK, 1998. - P. 24 - 32.

50. Wu X. Rule Induction with Extension Matrices // Journal of the American Society for Information Science. – Vol. 49, issue 5, 1998. - P. 435 - 454.

51. Yang Y., Wu X. Parameter Tuning for Induction-Algorithm-Oriented Feature Elimination // IEEE Intelligent Systems. Vol. 19, issue 2, 2004. - P. 40 - 49.

52. Yang Y., Wu X., Zhu X., Mining in Anticipation for Concept Change: Proactive-Reactive Prediction in Data Streams // Data Mining and Knowledge Discovery (DMKD), Vol.13., No.3., 2006. - P. 261 - 289. - URL: http://www.cs.uvm.edu/~xwu/Publication/DMKD06-2.pdf – Last viewed May 2008.

53. Zhang S., Wu X. Large Scale Data Mining Based on Data Partitioning // Artificial Intelligence, Vol. 15, 2001. - P. 129 - 139. - URL: http://www-staff.it.uts.edu.au/~zhangsc/scpaper/AAIzwu.pdf – Last viewed September 2007.

54. Zhu X., Wu X., Yang Y. Effective Classification of Noisy Data Streams with Attribute-Oriented Dynamic Classifier Selection // Knowledge and Information Systems (KAIS). Vol. 9, No. 3, Springer-Verlag, London, 2006. - P. 339 - 363. - URL: http://www.csse.monash.edu.au/~yyang/effective_kais.pdf – Last viewed August 2007.

55. Zhu X., Wu X., Chen Q. Eliminating Class Noise in Large Datasets // Proceedings of the 20th International Conference on Machine Learning (ICML - 2003), Washington D.C., USA, 2003. - P. 920 - 927. - URL: http://www.hpl.hp.com/conferences/icml2003/papers/230.pdf – Last viewed September 2007.