2nd INTERNATIONAL CONFERENCE ON
RESEARCH IN SCIENCE AND TECHNOLOGY
Istanbul-Turkey 14 March 2016
RSTCONF
دومین کنفرانس بین المللی
پژوهش در علوم و تکنولوژی
ترکیه - استانبول ۲۴ اسفند ۱۳۹۴

سیستم مبتنی بر دانش برای کلاسبندی متن با استفاده از الگوریتم ID6NB

آیدین ناصری فرد
دانشجوی کارشناسی ارشد مدیریت فناوری اطلاعات دانشگاه علوم و تحقیقات تهران
Naserifard.naserifard@gmail.com
سارا سید اسماعیل صراف
دانشجوی کارشناسی ارشد مدیریت فناوری اطلاعات دانشگاه علوم و تحقیقات تهران
Sara_s1366@yahoo.com
دکترمحمدرضا بابایی
استادیار گروه مدیریت صنعتی دانشگاه آزاد اسلامی واحد شهر ری
Babaei.mohammdreza@gmail.com

چکیده

این مقاله یک الگوریتم جدید به نام ID6NB را برای توسعه درخت تصمیم[1]، معرفی می‌کند که شامل الگوریتم ID3 غیر افزایشی[2] Quinlan است. این الگوریتم راه‌حل‌هایی برای دو مشکل ذیل ارائه می‌کند: ۱- وضعیتی که در آن رای گیری اکثریت[3] تصمیم نادرست می‌دهد (یعنی ساخت دو نوع قانون متفاوت برای داده یکسان).

۲- کاهش ابعاد[4] در الگوریتم غیر افزایشی درخت تصمیم‌گیری، تخمین صفت مناسب برای یک گره جایی که دو یا چند صفت بهره اطلاعاتی[5] یکسانی دارند. مشکل اکثریت به کمک الگوریتم Naive Bayes حل می‌شود. برای کاهش ابعاد نیز یک راه حل ارائه شده است. در نهایت، دقت طبقه بندی[6] به شدت بهبود یافته است. آزمایش گسترده و گسترش یافته در تعدادی از مجموعه داده‌های واقعی و مصنوعی نشان می‌دهد که ID6NBیک الگوریتم دسته‌بندی state-of-the-art است که نسبت به سایر روش‌های یادگیری درخت تصمیم‌گیری، خروجی بهتری دارد.

واژه های کلیدی: داده کاوی، کاهش ابعاد، طبقه بندی، درخت تصمیم، رای اکثریت، Naive Bayes

[1] Decision tree
[2] Incremental
[3] Majority Voting
[4] Dimensionality Reduction
[5] Information Gain.
[6] Classification

۱ مقدمه

فرایند کشف دانش در مجموعه داده KDD[۷] است که توسط Fayyad و همکارانش تعریف شده است. KDD به عنوان فرایندی جهت شناسایی ساختار داده معتبر و مفید از ساختار داده بی اهمیت است. داده کاوی گام اصلی در فرایند KDD است که با شمارش الگوهای نمایش داده شده در یک مجموعه داده است. کلاسبندی وظیفه اولیه داده کاوی است که هدف اش، یادگیری یک تابع ای است که رکوردهای مجموعه داده را به یک از چندین کلاس از قبل تعیین شده بر مبنای ویژگی های ان رکوردها، دسته بندی می کند. روش های رایج دسته بندی مانند: Back Propagation, Naive Bayes, SVM, ID3, C4.5 همگی به منظور بهبود عملکرد، طراحی شده اند. جنبه های دیگر کشف دانش مانند دو قانون متفاوت درباره ی داده یکسان، نشان کننده‌ی وجود ویژگی های مرتبط[۸] است که مشخص کننده ی شرط دومی برای الگوریتم های موجود می باشند. بنابراین مدل های کلاس بندی ناشی از داده های واقعی برای داده های متناقض و ناچیز موثر نیستند. در این مقاله ID6NB، برای حل این مشکل، راه حلی ارائه نموده است.

۱,۱تئوری اطلاعات و کلاسبندی

طبقه بندی داده ها فرایند ای است به منظور کاهش میزان عدم قطعیت[۹] یا بهره اطلاعاتی در مورد ویژگی کلاس بند[۱۰] است. در نظریه اطلاعات Shannon، اطلاعات به منظور کاهش یا حذف عدم قطعیت تعریف شده است.

در عملیات طبقه بندی، اطلاعات بیشتر منجر به این می شود که با دقت بیشتری نمونه های جدید را در کلاس های واقعی بگنجانیم. یک مدل که مقدار اطلاعات را افزایش نمی دهد غیر مفید است و دقت پیش بینی آن بهتر از یک روش تصادفی نیست.

متوجه شدیم برای اینکه یک نتیجه چند مقداری[۱۱] نسبت به پیش بینی نتیجه دو مقداری[۱۲]، به دقت پیش بینی شود، اطلاعات بیشتری مورد نیاز است. اطلاعات نظریه روش ما یک مدل کلی از وابستگی شرطی بین متغیر های تصادفی را نشان می دهد. اگر هر چیزی منجر به X شود، درجه غیر قطعیت آن با آنتروپی غیر شرطی[۱۳] اندازه گیری می شود: $\log p(x)$ $H(x)= \sum p(x)$. آنتروپی با واریانس که عاری از ماهیت است، فرق دارد و فقط به احتمال توزیع شده مقادیر تصادفی نسبت به مقادیر به هم پیوسته، وابسته است. بنابراین در انجام وظایف طبقه بندی جایی که بر چسب کلاس مهم نیست، کاهش آنتروپی ویژگی هدف، می تواند شرطی برای انتخاب فرضیه بهتر باشد. به عنوان نمونه در الگوریتم های ID3, C4.5 برای پیدا کردن بهترین ویژگی که به تصمیم یک گره از درخت تصمیم گیری نیاز است، از آنتروروپی استفاده می شود.

در این مقاله برای اولین بار با آوردن یک مثال با جزئیات کامل از الگوریتم ID6NB، یک راه جدید برای جدا سازی قوانین ارائه می کنیم و در نهایت یک مقایسه ی جامع این روش نسبت به الگوریتم های دیگر آورده‌ایم.

۲,۱کاهش ابعاد و انتخاب ویژگی

به حداقل رساندن تعدادی از ویژگی های مرتبط یا ویژگی ها در یک مدل کلاسبند، نسبت به افزایش سرعت یادگیری الگوریتم کلاسبندی، به دلایل متعددی مهم است. John و همکارانش بین دو مدل انتخاب مجموعه خوب از ویژگی های تابع هدف تفاوت قائل شده اند. از ویژگی های مدل فیلتر، فرض انتخاب ویژگی ها قبل از اعمال یک الگوریتم استقرایی[۱۴] در حالی

7 Knowledge Discovery in Databases
8 relevant features
9 Amount of uncertainty
10 Classifier
11 multivalue
12 binary
13 unconditional entropy
14 Induction

که در مدل های Wrapper ، دقت پیش بینی الگوریتم های اسقرایی را برای تعیین ویژگی ها به کار می برند. یک مورد بررسی وجود فیلتر و مدل هایWrapper برای انتخاب ویژگی در [8] وجود دارد. از طرفی دیگر معمولا مدل های Wrapperبا محاسبات قابل توجهی همراه هستند زیرا چندین بار نیاز به اجرای مجدد الگوریتم می باشد. پس روش فیلتر اجازه نمی دهد که یک الگوریتم کلاسبندی از پتانسیل خود به خوبی بهره برداری نماید. برخلاف روش فیلتر و مدل هایWrapper، الگوریتم ID6NB که در این مقاله آورده شده است، انتخاب خودکار ویژگی ها را به عنوان بخش جدایی ناپذیر از فرایند یادگیری، پیاده سازی می نماید. بنابراین حداقل مجموعه ای از ویژگی ها دریک بار اجرای الگوریتم استفاده می‌شود.

۳,۱قالب بندی مقاله

مطالب مرتبط در بخش ۲ آمده است. مشکلات در بخش ۳ آمده است. بخش ۴ الگوریتم پیشنهادی را شرح می دهد. در بخش ۵ الگوریتم ID6NB را با الگوریتم های خیلی رایج درخت تصمیم گیری، مقایسه می‌کنیم و عملکرد الگوریتم های مختلف بر روی مجموعه داده های استاندارد با معیارهای[15] متفاوت را ارزیابی می‌نماییم. در بخش۶ نتیجه گیری مقاله همراه با یکی از کارهایی که برای تحقیقات پیشنهاد می شود، آمده است.

۲ کارهای مرتبط

الگوریتم ID3 یک الگوریتم یادگیری مفید است زیرا می تواند به صورت موثرتری درخت تصمیم عمومی‌ای بسازد. این الگوریتم، برای کارهای یادگیری غیر افزایشی در کلاس بندی قوانین، انتخاب مناسب تری نسبت به روش های افزایشی، می باشد و نیاز به ساخت درخت تصمیم هر بار، نمی باشد. تکنیک های بسیاری برای ساخت مدل های مبتنی بر درخت تصمیم افزایشی وجود دارد. در خیلی از تلاش های پیشین شامل ID4[12], ID5[18], ID5R[18], ITI[19] هستند که همه این سیستم ها بهره اطلاعاتی را به عنوان اندازه گیری کننده برای انتخاب ویژگی ها به کار می برند. آن ها به صورتی طراحی شده اندکه به صورت افزایشی درخت تصمیم را می سازند و یک نمونه از درخت را در هر زمان با نگه داشتن های اطلاعات آماری لازم می سازد.

الگوریتم ID4 [۱۳] درخت تصمیم می سازد. الگوریتم ID5 [۱۴] با ساخت یک درخت تصمیم کار می‌کند و به روز رسانی آن به عنوان یک الگوریتم جدید در دسترس است. الگوریتم ID3 برای یادگیری افزایشی، با افزودن هر نمونه جدید به مجموعه یادگیری می تواند به کار رود. این کار از نا کارمد است.

ID5 [۱۶] و ID5R [۱۸] هر دو درخت تصمیم افزایشی برای بهبود کاستی های ID4 هستند. تفاوت اصلی وقتی محسوس است که ساخته دوباره درخت مورد نظر است. به دلیل اینکه صفات هر گره در هر زیر درخت که دارای کوچکترین مقدار آنتروپی را دارند، دور انداخته نمی شوند بلکه ویژگی گره های هر زیر درخت هر گره به آن گره مرتبط است. درID5 [۱۶]، بر خلاف ID5R [۱۸]، زیر درختان به صورت بازگشتی، به‌روزرسانی نمی‌شوند. بازساخت درخت کارامد نیست و زیر درخت ساخته شده مانند زیر درخت ساخته شده در ID3 نیست. نتیجه حاصل شده از الگوریتم ID5 بر روی نمونه های یکسان با الگوریتم ID3 یکسان نیست [۱۰]. البته الگوریتم ID5R نیز گارانتی نمی کند که نتایج یکسانی ارئه دهد [۱۷].

ITI[16] [۱۹] یک برنامه ای است که صورت اتوماتیک از نمونه های برچسب زده شده، درخت تصمیم، می‌سازد. یکی از جنبه های جنبه های الگوریتم ITI [۲۰] این است که مکانیزمی برای استقرای درخت تصمیم ارائه می‌نماید. اگر در حال اجرا، برای هر گره ای درخت ای را بسازد، نمونه های برچسب خورده جدید آن را نیز می‌سازد. پس الگوریتم اصلاح درخت را نیز داراست در نتیجه این روش جایگزین ای برای ساختن یک درخت از ابتدا، براساس مجموعه ی نمونه های برچسب شده، که

15 Benchmark
16 Incremental Tree Includer

معمولا خیلی گران است، خواهد بود. ITI دسته متغیرهای عددی و نمادین و از دست رفته را مورد پردازش می کند و شامل روش حرص مجازی نیز می باشد.

۳ وضعیت مسئله

۱ الگوریتم درخت تصمیم تا زمانی که شرط نهایی بر چسب کلاس را مشخص نماید، به صورت متناوب کار می‌کند. اما الگوریتم، هنگامی که اکثریت با شکست مواجه می شود، تمایل به انتخاب خود سرانه ی بر چسب کلاس دارد.
۲ وقتی که الگوریتم درخت تصمیم خودش، مجموعه صفات را به دست می اورد، روش Wrapper نامیده می‌شود. در این روش اگر دو مقدار بهره اطلاعاتی بزرگتر یا مساوی داشته باشند، الگوریتم این مشکل را به صورت موثر نمی تواند رفع نماید.
هدف این مقاله ارائه راه حل های ممکن برای مشکلات ذکر شده است.

۴ کارهای پیشنهادی

۱,۴ رخداد استثنا در کاهش ابعاد طول

الگوریتم درخت تصمیم کاهش ابعاد را در دسته بندی رعایت می نماید. در عملیات کاهش ابعاد، صفات با بالاترین بهره اطلاعاتی، انتخاب می شود و در بقیه موقعیت های ممکن، یعنی زمانی که دو صفت یا بیشتر بهره ی اطلاعاتی یکسان دارند، یک استثنا رخ می دهد. راه حل این است که به صفت خوب و بد در کاهش ابعاد، از هم جدا شوند. امتیاز این روش این است که در مقایسه با الگوریتم های دیگر تحت همان شرایط، ویژگی‌ها مناسب تری را انتخاب، می نماید.

۲,۴حل و فصل رخداد استثنا در کاهش ابعاد

عمق درخت تصمیم را 'd' در نظر بگیرید همراه با دو صفات A_i A_j که بهره اطلاعاتی یکسانی دارند، راه حل هایی که برای این حالت ممکن است مفید باشد به شرح ذیل می باشد:

۱. اگر این موقعیت در عمق 0 رخ بدهد، ریشه انتخاب می شود سپس موقتا دو درخت، با داشتن هر یک به عنوان ریشه، می سازیم و بر روی داده متنی داده شده، اعمال می نماییم سپس صفت با بالاترین دقت، انتخاب می شود.
۲. اگر این موقعیت در عمق درخت از ۱ تا d-1 رخ دهد:
   a. سپس از شاخه های دیگر گره پدر[17] می‌گذریم و صفاتی که هیچ کدوم از زیر شاخه هایش اصلا اتفاق نیافتاده را، حذف می‌کنیم.
   b. در غیر این صورت، از شاخه های دیگر گره پدر عبور می‌کنیم و صفاتی که در آن عمق مشاهده می‌شود را نگاه می داریم.
   c. یا در غیر این صورت، A_i و A_j اتفاق می افتد در حالی که از شاخه های دیگر عبور می‌کند. صفتی که در گره پدر، بهره اطلاعاتی بیشتری دارد، انتخاب می‌شود.

به روز رسانی موثر تر الگوریتم ID6NB، برای کاهش ابعاد در شکل ۱ نشان داده شده است.

[17] Parent

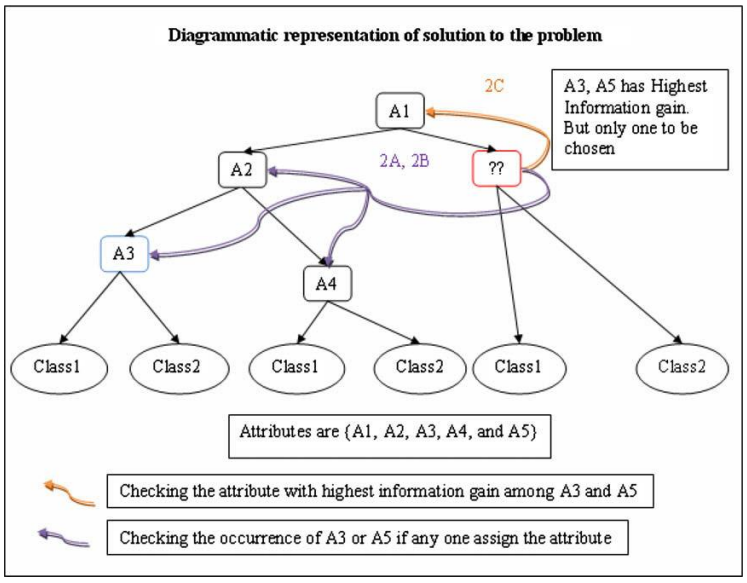
Algorithm:
The ID6NB algorithm update procedure for effective Dimensionality Reduction
Input: Attribute Ai, Aj and depth parameter {0, 1… (Leaf-1)}.
Output: A decision tree with optimal dimension
Method:

1. If depth D=0 then
   Form separate decision tree for both a_i and a_j, Compare the accuracy of decision tree and assign appropriate attribute.

2. Else // i.e. the depth between D=1 to D=leaf-1
   Traverse to the parent of this node and check for occurrence
   - If (A_i occurred)
     Then A_i
   - Else if (A_j occurred)
     Then Aj
   - Else
     Both A_i and A_j has occurred or not occurred
     Choose the attribute with highest information gain among A_i, A_j from the parent of this node.

شکل ۱- به روز رسانی الگوریتم ID6NB، برای کاهش ابعاد موثر تر

همان طور که درشکل۱ مشاهده می شود، شرط ۱عمق را ب ررسی می کند که گره در ریشه است یا نه. عبارت else، پیشامد صفات را بررسی می نماید. مراحل کلیدی 2a, 2b, 2c به صورت واضح در شکل ۲، آمده است.



شکل۲- الگوریتم ID6NB ای که روش جدیدی برای کاهش ابعاد ارائه می کند

۳,۴رخداد استثنا به علت شکست رای گیری اکثریت

در الگوریتم درخت تصمیم یکی از شرایط پایان دهنده، اکثریت اراء می باشد. بر اساس رای گیری اکثریت، وقتی جدولی دو صفت دارد، صفت با بیشترین رکورد انتخاب می شود. الگوریتم های قدیمی از حل این مشکل خاص ناتوانند. در روش ذکر شده این مقاله، برای حل مشکل ذکر شده، از الگوریتم Naive Bayes برای انتخاب برچسب کلاس، استفاده می‌شود. حل این مشکل برای حذقف داده های نویزی از مجموعه داده، مفید است و به افزایش دقت شناسایی داده، کمک می‌کند.

۴،۴حل مسئله رد رای گیری برچسب[18] کلاس جایی که رای گیری اکثریت با شکست مواجه می شود

به منظور اجتناب از انتخاب بر چسب کلاس به علت شکست در انتخاب اکثریت، انتخاب کلاس بـا الگـوریتم Naive bayes پیشنهاد می شود. الگوریتم Naive bayes در حقیقت شامل الگوریتم درخت تصمیم گیری گسترش یافته است که بهتـرین راه حل را ارائه می دهد(شکل ۳).

1) if attribute-list is empty then
2) if training data rules is null
3) return N as a leaf node labeled with the most common class in samples // Majority voting
   - This rule traversal from the root to the leaf is alpha rule.
   - If (record satisfy alpha rule)
     Correctly classified as (as normal DECISION TREE INDUCTION).
   - Else
     Incorrectly classified (as normal DECISION TREE INDUCTION).
4) else
5) **return N as a leaf node labeled with the attribute corresponding to class label value from probability based algorithm (Naive Bayesian algorithm)**
   - This rule traversal from the root to the leaf is beta rule.// An unique rule
   - If (record satisfy beta rule)
     Correctly classified.(This type of records cannot be handled by DECISION TREE INDUCTION algorithm)
   - Else
     Incorrectly classified.

شکل ۳- الگوریتم ID6NB- الگوریتم استقرایی درخت تصمیم به تنهایی با به روز رسانی شرط پایانی

در الگوریتم درخت تصمیم گیری قدیمی، هر مجموعه داده ای می تواند شامل هر دو این دو دسته هـا باشـد: ۱- شـامل داده ای باشد.۲- شامل همان داده نباشد. اما در الگوریتم ID6NB هر مجموعه داده می تواند در یکی از این۴ دسته ها باشد: ۱- شامل داده هایی که با قوانین Alpha مشخص می شوند، باشد.۲- شامل داده هایی که با قوانین Alphaمشخص می شوند، نباشد. ۱- شامل داده هایی که با قوانین Beta مشخص می‌شوند، باشد.۲- شامل داده هایی که با قوانین Beta مشـخص مـی شوند، نباشد(شکل ۴).

[18] Calss lable

Algorithm:
ID6NB. Generate a decision tree from the given training data.
Input: The training samples, samples, represented by discrete-valued attributes; the set of candidate attributes, attribute-list.
Output: A decision tree and set of rules.

Method:

1) Create a node N;
2) if samples are all of the same class, C then
3) return N as a leaf node labeled with the class C;
4) if attribute-list is empty then
5) if training data rules is null
6) return N as a leaf node labeled with the most common class in samples // Majority voting
7) else
8) **return N as a leaf node labeled with the attribute corresponding to class label value from probability based algorithm (Naive Bayesian algorithm)**
9) select test-attribute, the attribute among attribute-list with the highest information gain;
10) label node N with test-attribute;
11) for each known value a_i of test-attribute //partition the samples
12) grow a branch from node N for the condition test-attribute= a_i ;
13) let s_i be the set of samples in samples for which test-attribute= a_i;// a partition
14) if s_i is empty then
15) attach a leaf labeled with the most common class in samples;
16) else attach the node returned by Generate_decision_tree (s_i ,attribute-list-test-attribute);

شکل ۴- الگوریتم ID6NB

۱,۴,۴قوانین Beta و Alpha

امکان داده در مجموعه داده آموزشی به علت خطای انسانی، داده های نویزی، وجود داشته باشد که الگوریتم باید این خطا ها را شناسایی و حل نماید. وقتی قوانین شکل می گیرد با رکود هایی که قابل قبول قوانین Alpha هستند ولی قوانینی ای که با هم متناقضد قوانین Beta نام دارند. جدول ۱ مجموعه داده ای ازمجموعه داده All Electronics customer databse orginalرا نمایش می دهد (داده ها از[Qui86] به سمت می ایند). مقدار برچسب کلاس دو مقدار Yes, No است. پس دو کلاس متفاوت داریم. فرض کنیم C1 مربوط به کلاس Yes است و C2 مربوط به کلاس No است. ده نمونه از کلاس Yes و شش نمونه از کلاس No وجود دارد. برای محاسبه بهره اطلاعاتی برای هر صفت، اطلاعات مورد نیاز را محاسبه می کنیم. حال آنتروپی هر صفت را محاسبه می کنیم. با صفت Age محاسبات را اغاز می نماییم. باید Yesیا No بودن کلاس هر نمونه را در نظر بگیریم. به صورت زیر محاسبات را انجام می دهیم:

For age = ‘‘<=30”: s11 = 3, s12 = 4 then I (s11, s21) = 0.
For age = ‘‘31. . .40”: s11 = 4, s12 = 0 then I (s12, s22) = 0.985.
For age = ‘‘>40”: s11 = 3, s12 = 2 then I (s13, s23) = 0.97
I(S1,S2) = I(10,6)=-10/16log(10/16)-/16log(6/16)=0.954

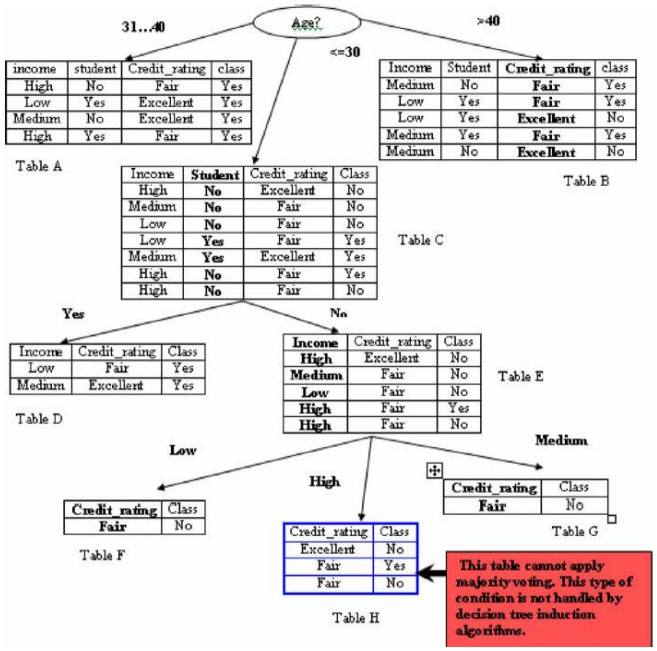
شکل ۱- مجموعه داده ای از All Electronics Customer Databse Orginal

RID	Age	Income	Student	Credit_rating	Class: Buys_computer
1	>40	Medium	No	Fair	Yes
2	>40	Low	Yes	Fair	Yes
3	>40	Low	Yes	Excellent	No
4	>40	Medium	Yes	Fair	Yes
5	>40	Medium	No	Excellent	No
6	31…40	High	No	Fair	Yes
7	31…40	Low	Yes	Excellent	Yes
8	31…40	Medium	No	excellent	Yes
9	31…40	High	Yes	Fair	Yes
10	<=30	High	No	Excellent	No
11	<=30	Medium	No	Fair	No
12	<=30	Low	No	Fair	No
13	<=30	Low	Yes	Fair	Yes
14	<=30	Medium	Yes	Excellent	Yes
15	<=30	High	No	Fair	Yes
16	<=30	High	No	Fair	No

حال بهره اطلاعاتی را محاسبه می کنیم:

$Gain(age)=I(_{s1,s2})-E(age) =0.220.$

بـــرای صـــفات دیگـــر نیـــز مشـــابه همـــین روال محاســـباتی انجـــام مـــی دهـــیم. نتـــایج $Gain(income)=$ 0.003,$Gain(student)=0.138$, $Gain(Credit\_rating)=$ 0.029 بـه دسـت مـی آیـد. همـان طـور کـه مشـاهده می‌شود، Age بیشترین بهره اطلاعاتی را دارد، پس انتخاب می شود. یک گره به نام Age ساخته می شود و زیر شـاخه هـای این گره، صفات Age می شود. در شکل ۵، نمایی از درخت ساخته شده، نمایش داده شده است. پس سه جدول A, B, C با سه شرط Age=”31…40”, Age=”<=30”, Age=”>40” ساخته می‌شود. در جـدول A، همـه عضـو کـلاس Yes هستند. پس یک برگ در آنجا اضافه شده و بر چسب Yes به خود می‌گیرد.



شکل۵- صفت age بیشترین بهره اطلاعاتی را دارد و به عنوان ریشه درخت انتخاب می شود و زیر شاخه های آن، صفات age می شوند. تقسیم بندی قوانین برای هر شاخه در قالب جدول آمده است.

در جدول C، نمی توان کلاسی را تخمین زد پس دوباره عملیات محاسبه بهره اطلاعاتی را انجام می دهـیم، دو جـدول E, D به دست می آید. در جدول D همه نمونه ها، عضو کلاس Yes هستند. پس یک برگ در آنجا اضـافه مـی شـود و بـر چسـب

Yes می خورد اما جدول E نیاز به محاسبه بهره اطلاعاتی دارد. سه جدول F, G, H به دست می آید که جدول F به برگ No و جدول G به برگ No می رسد. اما در جدول H، در مورد کلاس نمی تواند به نتیجه برسد. داده های ۱۵و۱۶ به عنوان داده ی آزمایشی انتخاب می شوند و بقیه‌ی داده های آموزشی محسوب می شوند. بر حسب داده های آموزشی احتمالات محاسبه می شوند و حال سوال این است که اگر شرایط ذیل برقرار باشند، برچسب کلاس چیست:

Age =‘‘<=30” and student = ‘‘no” and income = ‘‘high” and creditrating= ‘‘fair”

کلاسبندی Bayesian یک کلاس بند آماری است که می تواند تابع عضویت احتمالی میزان تعلق به یک کلاس را محاسبه نماید. طبق تئوری Bayes، فرض کنید X همان نمونه داده ای است که کلاس اش مشخص نیست و H فرصیه ای است که مشخص می کند داده ی X متعلق به کلاسC است. ما می خواهیم P(H|X) را مشخص کنیم یعنی احتمال اینکه Hای بیاییم با فرض مشاهدهX ، احتمالات P(X|H),P(H),P(X) از روی داده ها محاسبه می شود. تئوری Bayes برای طبقه بندی احتمال پسین مفید است. تئوری Bayes به شکل ذیل است:

$$P(H|X)=\frac{P(X|H)P(H)}{P(X)}$$

ما نیاز داریم که $P(C_i)$ $P(X|C_i)$ به ازای iهای برابر ۱یا ۲، بیشترین مقدار را داشته باشد. احتمال پسین از روی داده های داده شده محاسبه می شود. برای محاسبه P(X|C) این شرط ها را محاسبه می کنیم. بنابراین تئوری Bayes، مقدار No را می دهد. از آنجایی که مقدار کلاس بر مبنای مجموعه داده یادگیری جایی که رای اکثریت ممکن نیست، مشخص می شود، قوانینی که در این روش به کار می رود، قوانین Beta نام دارند. زیرا این قوانین ویژگی های منحصر به فردی برای برطرف کردن مشکل رای اکثریت دارند.

۵ نتایج تجربی و ارزیابی عملکرد

۱،۵مرور کلی

عملکرد الگوریتم ID6NB بر روی مجموعه عمومی و در دسترس ارزیابی شد. تمام این مجموعه داده ها در مخزن[19] سایت UCI موجود هستند[۱] و به طور گسترده ای توسط جامعه داده کاوی برای ارزیابی الگوریتم یادگیری استفاده می شوند [۱]. مجموعه دادهای که توسط ما انتخاب می شوند شامل داده های متنوع ای در طبیعت هستند. رکورد های این مجموعه داده ها دارای شکل بهره اطلاعاتی یکسان یا بزرگ تر هستند و مشکل دو قانون با ویژگی های یکسان اما بهره اطلاعاتی متفاوت را نیز دارند. این مجموعه داده ها به صورت بهینه کلاس بندی می شوند و کاهش ویژگی ها به حد مطلوب انجام می شود.

۲،۵ارزیابی عملکرد بررسی مجموعه داده های Monk

آزمایش اندازه گیری عملکرد الگوریتم پیشنهادی انجام شد. بر مبنای نتایج به دست آمده از عملکرد الگوریتم پیشنهادی مقاله، در مقایسه با الگوریتم های مختلف موجود بر روی متون، دقت طبقه بندی افزایش یافته است[۱۵]. جدول ۲ و ۳ پیش بینی دقت برای الگوریتم ID6NB در مقایسه با الگوریتم های دیگر دقت تصمیم گیری بروی مجموعه داده ها ی متناوب را نشان می دهد. همان طورکه ملاحظه می نمایید دقت پیش بینی ID6NB نسبت به بقیه الگوریتم ها بهتر است.

جدول ۲- پیش بینی دقت - مقایسه با روش های دیگر

Dataset	ID5R (%)	IDL (%)	ID5R-hat (%)	TDIDT (%)	ID6NB (%)
Monk-1	81.7	97.2	90.3	75.7	97.2
Monk-2	61.8	66.2	65.7	66.7	74.53

[19] Repository

جدول ۳- پیش بینی دقت - مقایسه با روش های دیگر

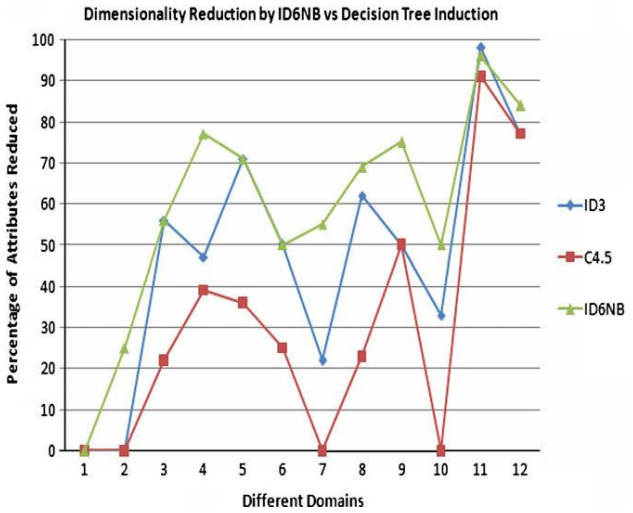
Dataset	ID3 (%)	ID3, no windowing (%)	ID5R (%)	ID6NB (%)
Monk-1	98.6	83.2	79.7	98.6
Monk-2	67.9	69.1	69.2	73.33
Monk-3	94.4	95.6	95.2	98.6

۳,۵کاهش ابعاد

کاهش در ابعاد، هدف مهم فرایند کشف دانش است. دردنیای واقعی، مجموعه داده شامل برخی از بخش های نـامربوط اسـت. بر خلاف طبقه بند ساده ی Bayes که تمام صفات را به کار می برد، الگوریتم‌هـای درخـت تصـمیم، صـفات غیـر مربـوط را حذف می کنند[۱۱]. در الگوریتم ای که در این مقاله ارائه شد، کاهش ابعاد در کلاس بندی از اهداف اصلی است. همان که در جدول ۴ مشاهده می شود، هیچ کدام از الگوریتم های ID3, ID6NB, C4.5 همه ی صفات را استفاده نکرده اند و صـفات کمتری ID6NB در استفاده شده است. آزمایشات درکامپیوتر پنتیوم ۴ انجام شده است. جدول ۴ نشان می دهـد کـه مـدلی توسط این الگوریتم به دست می آید، به میزان قابل توجهی کوچکتر از ID3, C4.5 است. همان طور که ملاحظه مـی شـود، متوسط صفات به کار رفته در ID6NB، باز کمتر از میانگین صفات یعنی ۷,۲۵ است (شکل ۶ را ملاحظه نمایید).

جدول۴- جدول کلی

Dataset	Available input attributes	Selected input attributes			Dim. reduction (%)			Run time (s)		
		ID3	C4.5	ID6NB	ID3	C4.5	ID6NB	ID3	C4.5	ID6NB
All Electronics original	4	4	4	4	0	0	0	0.01	0.05	0.1
All Electronics extended	4	4	4	3	0	0	25	0.01	0.05	0.1
Breast	9	4	7	4	56	22	56	0.06	0.05	0.12
Chess	36	19	22	8	47	39	77	0.55	0.33	0.8
Credit	14	4	9	4	71	36	71	0.33	0.11	0.3
Diabetes	8	4	6	4	50	25	50	0.61	0.11	0.35
Glass	9	7	9	4	22	0	55	0.27	0.11	0.3
Heart	13	5	10	4	62	23	69	0.11	0.05	0.16
Iris	4	2	2	1	50	50	75	0.05	0.00	0.1
Liver	6	4	6	3	33	0	50	0.06	0.06	0.12
Lung cancer	57	1	5	2	98	91	96	0.05	0.00	0.1
Wine	13	3	3	2	77	77	84	0.16	0.06	0.18
Mean	14.75	5.08	7.25	3.58	47.17	30.25	59	0.19	0.08	0.23



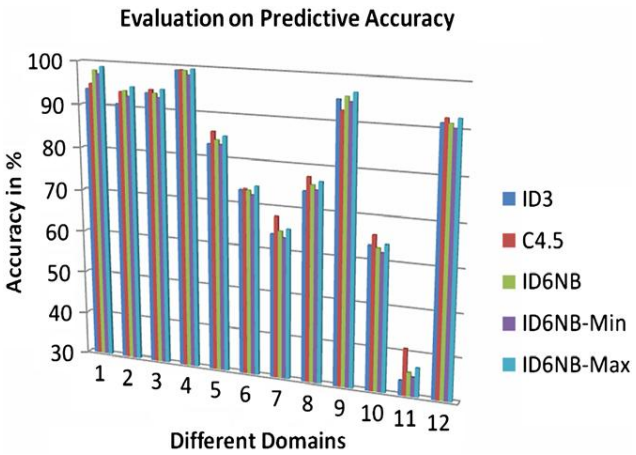
شکل ۶- کاهش ابعاد

۴،۵دقت پیش بینی

به طور معمول چهار روش برای ارزیابی دقت وجود دارد از جمله با استفاده از مجموعه داده آموزشی، مجموعه داده آزمایشی، Cross Validation, Percentage Spliting. جدول ۵ برای هر مجموعه داده دقت پیش بینی را در مقایسه با روش های دیگر درخت تقسیم گیری تخمین می زند. همان طور که در جدول ۴ مشاهده می شود دقت پیش بینی ID6NB کمی از دقت پیش بینی C4.5 کم تر است اما در All Electronic Orginal , All Electronic Extend, iris، الگوریتم ID6NB حتی بهتر از C4.5 عمل کرده است. در زمینه کاهش ابعاد و حل مسئله اکثریت، در تمام مجموعه داده ها، مقدار کمی دقت در حد یک در صد از بین رفته است که با توجه به تعداد ورودی ها قابل توجه است. پس ID3 با توجه به کمترین میانگین دقت و بیشترین ورودی صفات امتیاز کمتری نسبت به بقیه روش ها، دارا است. البته با توجه به کار مدل را باید انتخاب کرد. در بسیاری از که دقت را می توان فدای کاهش ابعاد نمود، الگوریتم ID6Nb را می توان انتخاب نمود.

جدول ۵- پیش بینی دقت - مقایسه با روش های دیگر

Dataset	ID3	C4.5	ID6NB	ID6NB-Min	ID6NB-Max
All Electronics original	93.75	95	98	97.2	98.9
All Electronics extended	90.625	93.5	93.75	92.5	94.75
Breast	93.6	94.4	93.6	92.6	94.6
Chess	99.1	99.2	99.1	98.1	99.5
Credit	83.1	85.9	84.1	83.1	85.1
Diabetes	73.3	73.5	73.3	72.3	74.3
Glass	63.8	67.9	64.6	63.1	65.2
Heart	74.3	77.5	75.7	74.7	76.6
Iris	94.9	92.6	95.6	94.5	96.6
Liver	63.5	65.9	63.1	62.1	64
Lung cancer	33.4	40.9	35.5	34.5	36.8
Wine	91.3	92.4	91.3	90.3	92.5
Mean	79.55	81.55	80.63	79.58	81.57



شکل۷- ارزیابی پیش بینی دقت

۶ نتایج پیاده‌سازی

الگوریتم پیشنهادی در این مقاله با استفاده از نرم افزارMatlab R2015b پیاده سازی گردیده و روی داده‌های مربوطه مورد ارزیابی قرار گرفت. داده‌های استفاده شده در این مقاله از آدرس اینترنتی http://archive.ics.uci.edu/ml/ دانلود شده و استفاده گردید.

نتایج حاصل از پیاده سازی، برای دقت ارزیابی در جدول های ذیل ارائه شده است.

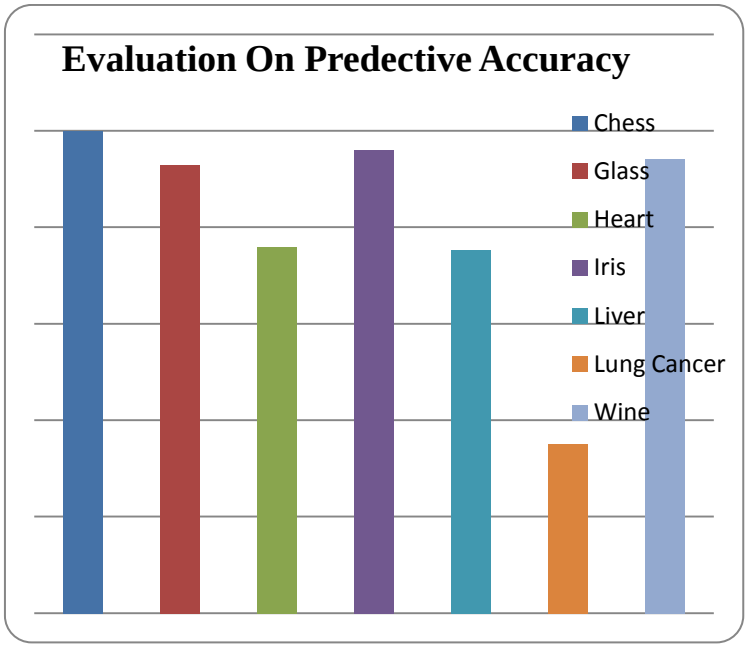
جدول ۶. نتایج بدست آمده از پیاده‌سازی الگوریتم ID6NB برای پیش بینی دقت بر روی مجموعه داده Monk

Dataset	ID6NB(%)
Monk-1	98.39

Monk-2	97.22
Monk-3	74.58

جدول ۷. نتایج بدست آمده از پیاده‌سازی الگوریتم ID6NB برای پیش بینی دقت بر روی کل مجموعه داده ها

Dataset	ID6NB(%)	ID6NB-Min	ID6NB-Max
Chess	100	100	100
Glass	92.99	91.99	93.99
Heart	75.97	74.95	76.99
Iris	96.11	95.11	97.11
Liver	75.17	74.21	76.13
Lung Cancer	35.12	34.12	36.12
Wine	94.18	93.21	95.15



شکل ۸- ارزیابی پیش بینی دقت

از مقایسه جداول ۶ و ۷ با جداول ۸ و ۹، می‌بینیم که نتایج حاصل از پیاده‌سازی تا حد زیادی مشابه نتایج ارائه شده در مقاله می‌باشد؛ با این حال، تفاوت‌های حاصل را می‌توان ناشی در اختیار نداشتن جزئیات کامل پیاده‌سازی دانست.

۷ نتیجه گیری

در این مقاله یک الگوریتم جدید برای ساخت یک مدل طبقه بندی ساده و منطقی معرفی شد که ID6NB نامیده می شود. در نسخه قبلی ID3، درخت بر اساس بهره اطلاعاتی صفات و دیگر شرایط پایانی ساخته می‌شد و در نسخه های بعدی ID3 مانند ID4, ID5, ID5R برروی بهینه کردن درخت، تایید کرده اند. در این مقاله کارهایی بر روی خطاهای ID3 و بهبود کارایی آن با از بین بردن داده های اضافی و کاهش ابعاد انجام شد. در روش ارائه شده، خطای رای گیری اکثریت به کمک

احتمال Naive Bayesian رفع شد که این راه حلی برای داده های نویزی است. نقطه قوت این الگوریتم، کاهش ابعاد و کلاس بندی است (شکل ۷ را ملاحظه نمایید).

۸ منابع

[1] C.L. Blake, C.J. Merz, UCI Repository of Machine Learning Databases, http://archive.ics.uci.edu/ml/

[2] T.M. Cover, J.A. Thomas, Elements of Information Theory, Wiley, 1991.

[3] U. Fayyad, G. Piatetsky-Shapiro, P. Smyth, From data mining to knowledgediscovery: an overview, in: U. Fayyad, G. Piatetsky-Shapiro, P. Smyth, R.Uthurusamy (Eds.), Advances in Knowledge Discovery and Data Mining, AAAI/MIT Press, 1996, pp. 1–36.

[4] Ian H. Witten, Eibe Frank, Data Mining: Practical Machine Learning Toolsand Techniques with Java Implementations, Morgan Kaufman Publishers,2000.

[5] James Joyce, Bayes theorem, Standford encyclopedia of philosophy, 2003.

[6] Jiawei Han and Micheline Kamber, Data Mining Concepts and Techniques2005.

[7] G.H. John, R. Kohavi, K. Pfleger, Irrelevant features and the subset selectionproblem, in: roceeding of the 11th Int'l Conf.Machine Learning, 1994, 121–129.

[8] H. Liu, H. Motoda, Feature Selection for Knowledge Discovery and Data mining,Kluwer Academic, Boston, 1998.

[9] J.R. Quinlan, Induction of decision trees, Machine Learning 1 (1) (1986) 81–106.

[10] J.R. Quinlan, Simplifying decision trees, International Journal of Man-MachineStudies 27 (1987) 221–234.

[11] J.R. Quinlan, C4.5: Programs for Machine Learning, Morgan Kaufman,1993.

[12] J.C. Schlimmer, R. Granger Jr., Beyond incremental processing: trackingconcept drift, Proceedings of AAAI 1 (1986) 502–507.

[13] J.C. Schlimmer and D. Fisher, A case study of incremental concept induction,in: Proceedings of the Fifth National Conference on Artificial Intelligence,Philadelpha, PA, Morgan Kaufmann, 1986, 496–501.

[14] J.C. Schlimmer, R.H. Granger Jr., Incremental learning from noisy data, MachineLearning 1 1986) 317–334.

[15] S. Thrun, J. Kreuziger, R. Hamann, W. Wenzel, et.al., The MONK's Problems:A Performance Comparison of Different Learning Algorithms, Tech. ReportCMU-CS-91-197, Computer Science Department, Carnegie MellonUniversity,1991.

[16] P.E. Utgoff, ID5: An incremental ID3, in: Proceedings of the Fifth InternationalConference on Machine Learning,Morgan Kaufmann Publishers, San Mateo,California, 1988, 107–120.

[17] P.E. Utgoff, Improved training via incremental learning, in: Proceedings of theSixth International workshop on Machine Learning. Ithaca, Ithaca, New York,United States, 1989.

[18] P.E. Utgoff, Incremental induction of decision trees, Machine Learning 4 (1989)161–186.

[19] P.E. Utgoff, An improved algorithm for incremental induction of decisiontrees, in: Proceeding of the 11th Int'l Conf. Machine Learning, 1994, 318–325.

[20] P.E. Utgoff, Decision tree induction based on efficient tree restructuring, in:Journal of Machine Learning, Springer, 2004, pp. 5–44.

[22] WEKA-Open Source Collection of Machine Learning Algorithm.Fig. 7. Evaluation on predictive accuracy.