ساخت درخت تصمیم پیشرفته بر پایه انتخاب صفات و نمونه داده به منظور تشخیص نقص در ماشین آلات دوار

چکیده

در این مقاله رویکرد جدیدی که از مشکلات بیش برازش و پیچیدگی که در ساخت درخت تصمیم نادیده گرفته میشوند ارائه میشود. درخت های تصمیم, ابزارهایی کارآمد برای ساخت نمونه های طبقه بندی به خصوص در مهندسی صنعتی هستد. در مرحله ی ساخت این درخت ها دو مشکل عمده وجود دارد که عبارتند از: انتخاب صفات مناسب و اجزای پایگاه داده . در پژوهش پیش رو، ازانتخاب صفات و نمونه داده به منظور غلبه بر مشکلات ذکر شده استفاده شده است. جهت اثبات رویکرد پیشنهادی، چندین آزمایش روی 10 مجموعه داده ی معیار انجام و نتایج آنها با رویکردهای کلاسیک مقایسه شده است. در پایان، کارکرد مؤثر رویکرد پیشنهادی را در ساخت قواعد غیر پیچیده ی تصمیم به منظور تشخیص نقص در ماشین آلات دوار را ارائه خواهیم داد.

واژگان کلیدی: ساخت درخت تصمیم، هرس کردن، نمودار پژوهش، انتخاب صفات، نمونه داده

1- مقدمه

در زمینه ی صنعتی، با پیچیدگی تجهیزات نصب شده خطرات شکست و اختلال در حال افزایش هستند. این پدیده بر کیفیت محصول اثر میگذارد، باعث میشود ماشین فوراً خاموش شود و به عملکرد صحیح کل سیستم تولید لطمه میزند. ماشین های دواررده ای عمده از تجهیزات مکانیکی هستند و به بیش ترین دقت و کنترل بی وقفه نیاز دارند تا از عملکرد بهینه آنها اطمینان حاصل شود. بطور سنتی، تجزیه و تحلیل های ارتعاش و بسیاری از روش های پردازش سیگنال به منظور بدست آوردن اطلاعات سودمند جهت کنترل شرایط عملکرد مورد استفاده قرار گرفته اند. خلف و دیگران (2013) دامنه بسامد را به منظور کسب اطلاعات و تشخیص نقص مورد تجزیه و تحلیل قرار دادند. تجزیه و تحلیل سپسترال (روش غیر خطی پردازش سیگنال) برای ساخت نشانگر نقص سیستم پر توان به کار رفته است (بداوی و دیگران، 2004) و تبدیل فوریه کوتاه مدت (روش پردازش

سیگنال برای تجزیه و تحلیل امواج متغیر) ارائه شد (موشر و دیگران، 2003). روش های دیگر مثل توزیع ویگنر–ویله (بایدر و بال، 2001)، تجزیه و تحلیل امواج کوچک پیوسته (کانکار و دیگران، 2011) و تجزیه و تحلیل امواج کوچک گسسته (دی جی ابالا و دیگران، 2008) نیز مورد استفاده قرار گرفته اند.

از الگوریتم های طبقه بندی نیز میتوان در ساخت سیستم های تشخیصی کنترل شرایط استفاده کرد. به عنوان مثال, شبکه های عصبی (چن و چن، 2011) ماشین بردار پشتیبان (دنگ و دیگران، 2011) و طبقه بندی بیزین (یانگ و دیگران، 2005) همه به کار گرفته شده اند. در هر حال, تکنیک های درخت تصمیم همچنان برای کاربردهای مهندسی به دیگر تکنیک ها ترجیح داده میشوند, زیرا به کاربران این امکان را میدهند که به آسانی به رفتار مدل های ساخته شده در مقابل طبقه بندی های ذکر شده در قسمت بالا پی ببرند. استفاده از درخت های تصمیم برای چنین کاربردهایی در مقالات پژوهشی بیشماری از جمله سوگوماران و راماچاندران (2007)، ژاوو و ژانگ (2008)، ساکتیول و دیگران (2010) و سوگوماران و دیگران (2007) گزارش شده است.

ساخت درخت تصمیم (Decision Tree,DT) دو مرحله رشد و هرس را شامل میشود. در مرحله رشد, طبق قوانین مشخص جدا کننده تا زمانی که همه ی نمونه های هر یک از زیرمجموعه ها، در زیرهمان طبقه مخصوص به خود (خالص) قرار گیرند یا به معیارهای توقف برسند؛ داده های آموزشی (نمونه ها) بطور پی در پی به دو یا چند زیرمجموعه که رو به پایین ترسیم میشوند تفکیک میشوند. به طور کلی, این مرحله ی رشد، درخت تصمیم بزرگی را تولید میکند که نمونه های یادگیری را در برمیگیرد و ابهامات داده ها (به طور ویژه سروصدا و تغییرهای باقی مانده) را مورد توجه قرار میدهد. رویکردهای هرس کردن برپایه مدل های هیوریستیک (مدل سیستماتیک پردازش اطلاعات) با از بین بردن تمامی بخش های درخت تصمیم که ممکن است بر پایه داده های پر سروصدا و یا غلط باشند از مشکلات بیش برازش جلوگیری میکند. این امر, از پیچیدگی و اندازه درخت تصمیم می کاهد. مرحله هرس میتواند یک درخت تصمیم رشد یافته را یا از قسمت بالای آن یا از قسمت پایینش هرس کند. علاوه بر این, بسیاری از مدل های هیوریستیک موجود بسیار چالش برانگیز هستند (بریمن و دیگران، 1984؛ نیبلت و براتکو، 1987؛ کوپینلن، 1987)، اما متأسفانه هیچ یک از روش ها به تنهایی بهتر از دیگر روش ها عمل نمیکند(مینگرز، 1989؛ اسپوسیتو و دیگران، 1997).

در مورد مشکلات مرحله رشد, دو راه حل وجود دارد: راه حل اول با کاهش میزان داده های یادگیری و ساده سازی قواعد تصمیم، پیچیدگی درخت تصمیم را کاهش میدهد (پیراموتو، 2008). راه حل دوم برای غلبه بر مشکلات بیش برازش, انتخاب صفات را به کار میگیرد (ییلدیز و آلپیدین، 2005؛ کوهاوی و جان، 1997). برای غلبه بر اندازه درخت تصمیم و خطرات بیش برازش, ترکیب انتخاب صفات و کاهش داده ها را به منظور ساخت درخت تصمیم پیشرفته و هرس نشده پیشنهاد میکنیم. بدین ترتیب, مشکل ساخت درخت تصمیم بهینه به اکتشاف مشکل فضای پژوهش نمودار ترکیبی تبدیل خواهد شد. ویژگی کلیدی این موضوع, کد گذاری هر یک از زیرمجموعه های صفت A_i, و زیرمجموعه نمونه های X_j، به مجموعه (A_i, X_j) میباشد. تمامی مجموعه های احتمالی (A_i, X_j) نمودار فضای پژوهش را تشکیل میدهند. نتایج نشان میدهند که نمودار پیشنهادی تا حد زیادی عملکرد این درخت تصمیم را در مقایسه با درخت های تصمیم استاندارد و هرس شده و همچنین درخت های تصمیمی که تنها بر پایه انتخاب صفات یا کاهش داده ها ترسیم میشوند بهبود میبخشد. بقیه این مقاله را مباحث زیر تشکیل میدهند:

در بخش دوم به طور خلاصه به بحث راجع به برخی از مطالعات پیشین در زمینه ساخت درخت تصمیم پرداخته میشود. در بخش سوم مفاهیم به کار گرفته شده در این پژوهش معرفی میشوند. در بخش چهارم به شرح رویکرد خود بر پایه انتخاب صفات و نمونه پایگاه داده به منظور بهینه سازی معمول ساخت درخت تصمیم میپردازیم. در بخش پنجم نتایج آزمایشاتی که در آنها از 10 مجموعه دهده معیاراستفاده شده است گزارش میشوند. در بخش ششم، درخت تصمیم پیشرفته و هرس نشده برای مشکل تشخیص نقص در ماشین آلات دوارمورد استفاده قرار گرفته است. در پایان, بخش هفتم, از این مطالعه نتیجه گیری میکند.

2- کارهای مربوطه

در این بخش به رویکردهای پس از هرس که در راستای ارتقای DTC پیشنهاد شده اند خواهیم پرداخت. هدف مشترک آن ها کاهش:

1) پیچیدگی درخت و
2) سرعت خطای مجموعه داده های مستقل آزمون بود.

روش های هرس دارای تفاوت های متعددی هستند که می توان آن ها را به شرح زیر خلاصه کرد:

1. ضرورت مجموعه داده های آزمون
2. تولید یک سری درخت های فرعی هرس شده یا پردازش یک درخت واحد
3. هرس معیارهای تعیین

بریمن و دیگران (1984) هرس پیچیدگی خطا را ارائه دادند که از خطای پیچیدگی هزینه بهره می برند. در اقدام هرس از جریمهٔ سرعت خطا براساس اندازهٔ درخت فرعی استفاده می گردد. خطاها و اندازه برگ های درخت (پیچیدگی) هر دو در این روش هرس لحاظ می شوند. اندازه گیری خطر هزینه – پیچیدگی تمام درخت های فرعی احتمالی در یک DT T_0 اولیه به عنوان خطای $R(t)$آموزشی که به محصول فاکتور α و تعداد برگ ها $|\tilde{t}|$ در درخت فرعی t اضافه می شود محاسبه می گردد. یعنی: (1) مجموعه ایی از درختان تصمیم فرعی با کمترین مقدار α برای هرس انتخاب می شوند. در پایان، درخت فرعی که به درستی هرس شده t از توالی α درخت های فرعی با استفاده از یک مجموعهٔ داده آزمون مستقل انتخاب می شود. انتخاب نهایی مبنی بر سرعت خطا یا خطای معیار (با فرض توزیع دو جمله ایی) می باشد.

هرس خطای کاهش که توسط کینلان (1987) مطرح شد مجموعه ایی از DT های هرس شده با استفاده از مجموعه داده های آزمون را تولید می کند. یک DT T_0 کامل ابتدا با استفاده از مجموعه داده های آموزشی رشد می کنند. سپس مجموعه داده های آزمون مورد اسشتفاده قرار می گیرد و بر ای هر گره یا نود در T_0، تعداد خطاهای دسته بندی در دستگاه هرس زمان ایجاد می گردد که درخت فرعی t در مقایسه با تعداد خطاهای دسته بندی زمانی که به t به یک برگ تبدیل می شود نگه داشته شود. سپس، تفاوت مثبت بین دو خطا به گرهٔ ریشهٔ درخت فرعی تخصیص داده می شود. گرهٔ دارای بزرگ ترین اختلاف هرس می گرلدد. این پروسه تا زمانی تکرار می شود که عمل هرس منجر به افزایش سرعت دسته بندی نادرست گردد. در پایان، کوچک ترین نسخهٔ صحیح ترین درخت با توجه به مجموعهٔ داده های آزمون تولید می شود.

در مقایسه با هرس خطای کاهش، ضرورت مجموعه می توان از داده های آزمون مجزا با استفاده از هرس خطای بدبینانه اجتناب کرد (PEP). PEP از سرعت اصلاح پیوستهٔ دو جمله ایی برای دستیابی به یک برآورد واقعی تر

از سرعت دسته بندی نادرست استفاده مخی کند. اصلاح دسته بندی نادرست به تعداد برگ ها و دسته بندی های نادرست بستگی دارد.

هرس مبنی بر خطا (EBP) به عنوان نسخهٔ پیشرفتهٔ PEP است که درخت را براساس استراتژی Post - order از بالا به پایین قطع می کند. هیچ مجموعهٔ دادهٔ هرسی نیاز نیست و سرعت اصلاح پیوستهٔ دو جمله ایی PEP استفاده می شود. از این رو، تفاوت این جاست که در هر تکرار، EPP احتمال پیوند یک شاخهٔ t_y به جای والد y را در نظر می گیرد. خطاهای برآورد t_x، t_y در تعیین این به کار می روند که آیا برای هرس گرهٔ x مناسب هستند (درخت ریشه دار شده با y به جای برگ) یا برای جایگزینی آن با t_y (بزرگ ترین درخت فرعی) یا حفظ t_x اصلی.

اخیراً لو و دیگران (2013) یک روش هرسی جدید براساس خطر ساختاری گره های برگ ایجاد کرده اند. این روش تحت این فرضیه ارائه شد که برگ های با میانگین صحت بالا که درخت می تواند داده های آموزشی را بسیار خوب دسته بندی کند و حجم بالای این گونه برگ ها عموماً عملکرد خوبی را نشان می دهند. همانند روش های رایج هرس، مجموعه ایی از درخت های فرعی تولید می شود. پروسه از هر گرهٔ x را روی DT T_0 بازدید می کند (t_x یک درخت فرعی است که x ریشهٔ آن می باشد). برای هر درخت فرعی t_x، گره های هرس پذیر یافت می شوند (برگ ها دو بچهٔ آن ها هستند) و خطرات ساختاری اندازه گیری می شوند. در پایان، درخت فرعی که خطر ساختاری را به حداکثر می رساند برای هرس انتخاب می شود.

روش های دیگری برای پس هرس پیشنهاد شده اند مثل هرس ارزش بحرانی، هرس خطای کمینه و هرس DI (که عمق و ناخالصی گره ها را متعادل می کند). انتخاب DT نیز با روایی مواجه بوده است و الگوریتم های دسته ایی برای بیرون کشیدن بهترین درخت از مجموعه مدل های مختلف مورد استفاده قرار گرفت. برای انتخاب قوی ترین DT، تمام روش ها ایجاد می گردند و عملکردها براساس مجموعه های روایی و آموزشی مجزا اندازه گیری می شوند. در این کار، هدف اصلی ساخت DT ها بدون زیر هرس یا تناسب داده های آموزشی و بدون انتخاب یکی از روش های مختلف هرس می باشد. دو کار قبل نشان داده اند که DT های هرس نشده

درصورتی که اصلاح laplace برای محاسبهٔ احتمالات دسته استفاده شود نتایج شبیه به درختان هرس شده می دهند.

شناسایی مجموعه های کوچک تر با ویژگی های بسیار پیشگویانه در بسیاری از شیوه های آموزشی و یادگیری لحاظ شده اند. انتخاب ویژگی دارای هدف مشابه با روش های هرس می باشد یعنی حذف ویژگی های نامربوط، اضافه و پر سروصدا در فاز ساخت جهت نیل به عملکرد خود DT. مطالعات بسیاری در زمینهٔ مدل های دسته بندی انجام شده است. در این مطالعات، تکنیک های wrapper برای انتخاب شاخصه مورد استفاده قرار گرفته اند. از یک الگوریتم آموزشی هدف برای برآورد ارزش زیرمجموعه های ویژگی استفاده می شود. فرایند تحت رابطه دو جمله ایی "$\underline{C}$" بین زیرمجموعه های ویژگی استفاده می شود. پروسهٔ پژوهش و جستجوی را می توان براساس اولین عرض و طول یا ترکیبی از هر دو انجام داد (مثلاً *A). wrapper ها معمولاً در مقایسه با فیلترها بهتر عمل می کنند اما عملکرد بهتر ناشی از یک هزینهٔ محاسباتی است. در مورد بدتر، زیرمجموعه های 2^m ویژگی ها باید تست شوند (m تعداد ویژگی ها می باشد).

DT ها را همانند انتخاب ویژگی می توان با کاهش پیچیدگی داده ها و همچنین کاهش تأثیرات شاخصه های ناخواستهٔ داده ها بهبود بخشید. کاهش داده ها اساساً شامل کاهش dimentionality و یا کاهش نمونه می باشد.

در نتیجه، تکنیک های مختلف هرس مورد مطالعه قرار گرفتند اما هیچ یک برای طیف وسیعی از مشکلات کفایت نمی کند. کارهایی بر روی انتخاب ویژگی و نمونه برداری داده ها جهت ارتقای DTC انجام گرفته است. برای تحقق یک DT بهتر برای یک کاربرد خاص، الگوریتم IUDT را پیشنهاد می دهیم که ترکیبی از شیوهٔ جدید نمونه برداری رندوم پایگاه داده ها با پروسهٔ wrapper انتخاب ویژگی می باشد. هدف اصلی ما در این مطالعه کاهش تعداد مؤثر نمونه ها و ویژگی های داده های آموزشی و در نتیجه به حداقل رساندن اندازهٔ DT می باشد.

3. مقدماتی

ما قبل از این که به تشریح رویکرد بپردازیم نتایج اساسی ترتیب مجموعه ها، مشکلات تناسب دسته بندی کننده (classifier)، ارتباط ویژگی و مشکلات حشو و نهایتاً DTC را ارائه می دهیم. تعاریف ارائه شده در این بخش همان بار معنایی را دارند که تعاریف ارائه شده در Davey (2002) دارند.

3.1 ترتیب نسبی

مجموعهٔ دارای ترتیب در واقع مجموعه ایی از عناصر با نسبت ترتیبی می باشد.

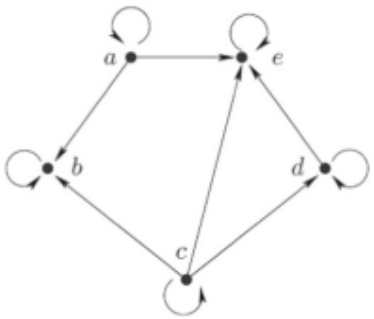
تعریف 1: رابطهٔ دوتایی R در مجموعهٔ E در صورتی که انعکاسی، گذرا و نامتقارن باشد یک ترتیب نسبی است.

- $\forall x \in E,\ xRx$ (reflexivity),
- $\forall (x,y) \in E \times E,\ (x\ R\ y\ et\ y\ R\ x) \Longrightarrow (x = y)$ (anti-symmetry),
- $\forall (x,\ y,\ z) \in E \times E \times E,\ (x\ R\ y\ et\ y\ R\ z) \Longrightarrow (x\ R\ z)$ (transitivity).

مثال 1: در صورتی که $X = \{a,b,c,d,e\}$ و $P = (x,\leq)$ یک مجموعه مرتب باشد $\leq$ ترتیب زیر را تعریف می کند:

$$X:\ \leqslant = \{(a,b),\ (a,e),\ (c,b),\quad (c,d),\ (c,e),\ (d,e),\ (a,a),\ (b,b),\ (c,c),\ (d,d),\ (e,e)\}.$$

ما می توانیم مجموعهٔ ترتیبی P را به عنوان نمودار جهت دار که گره های آن مطابق با عناصر x می باشد و کناره های آن رابطه $\leq$ با حلقه هایی که دو جفت (x,X) را نشان می دهند ارائه دهیم. شکل 1 این مثال را نشان می دهد.



شکل 1

تعریف 2 (نمودار): نمودار $G = (V,E)$ شامل مجموعه ایی از موارد عمودی V و مجموعه کناره های $E \subseteq V*V$ می باشد. هر کنارهٔ $e \in E$ با یک جفت نامرتب موارد عمودی مرتبط است.

در مورد نمودار جهت دار، هر کناره $e \in E$، جفت مرتب موارد عمودی مرتبط است. در ادامهٔ این بخش، نمودار تحقیقاتی L و یک عنصر در L (یعنی رأس $V \in G$) را به عنوان الگو نشان می دهیم.

3.1.1 الگوهای اختصاصی سازی و تعمیم دهی

اختصاصی سازی (تعمیم دهی) یک رابطهٔ دوگانه است که ترتیب نسبی $\leq$ در الگوها "L" تعریف می کند. الگوی ϕ در مقایسه با دیگر الگوی θ درصورتی که $\varphi \leq \theta$ کلی تر می باشد. مشابهاً، θ نسبت به ϕ اختصاصی تر است یعنی برای رابطهٔ $a \leq e$ در مثال 1، a نسبت به e کلی تر است و e نسیت به a اختصاصی تر است.

3.1.2 اپراتورهای اختصاصی سازی و تعمیم دهی

L مجموعهٔ نسبتاً مرتبی از الگوها می باشد. اپراتور اختصاصی سازی P_s هر الگوی $\varphi \in L$ را به یک مجموعه الگو که خاص تر است مرتبط می سازد: (4) مشابهاً، ما می توانیم یک اپراتور تعمیم دهی P_g تعریف کنیم به گونه ایی که (5). گفته می شود که اپراتور P_s در صورت مستقیم (بدون واسطه) است که فقط ϕ را به کلی ترین الگوهای مجموعه الگوهایی که خاص تر هستند $P_s(\varphi)$ مرتبط سازد. $P_s(\varphi) = \min(P_s(\varphi))$ در نتیجه گفته می شود اپراتور P_g در صورتی مستقیم است که فقط ϕ را با خاص ترین الگوهای مجموعه الگوهایی که کلی تر هستند مرتبط سازد $P_g(\varphi), P_g(\varphi) = \max(P_g(\varphi))$.

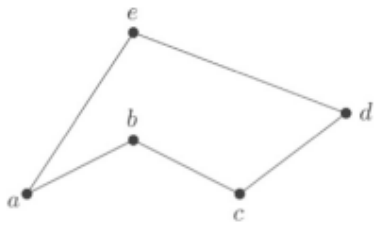
3.1.3 الگوهای بیشینه و کمینه

درصورتی که رابطهٔ اختصاصی سازی در L باشد و $\emptyset \subseteq L$ یک مجموعه الگو باشد کمینه (φ) به عنوان کلی ترین مجموعهٔ الگوهای $\emptyset$ و بیشینه $(\emptyset)$ به عنوان اختصاصی ترین مجموعهٔ الگوهای $\emptyset$ مطرح می گردد.

$$\min(\Phi) = \{\varphi \in \Phi | \nexists \theta \in \Phi \text{ s.t. } \theta \prec \varphi\}$$

$$\max(\Phi) = \{\varphi \in \Phi | \nexists \theta \in \Phi \text{ s.t. } \varphi \prec \theta\}$$

با استفاده از یک رابطه ترتیبی مستقیم می توانیم یک مجموعهٔ نسبتاً مرتب با یک نمودار جهت دار و نمودار غیر مرور Hasse ارائه دهیم. شکل 2 مثال 1 را به عنوان نمودار Hasse نشان می دهد.



شکل 2

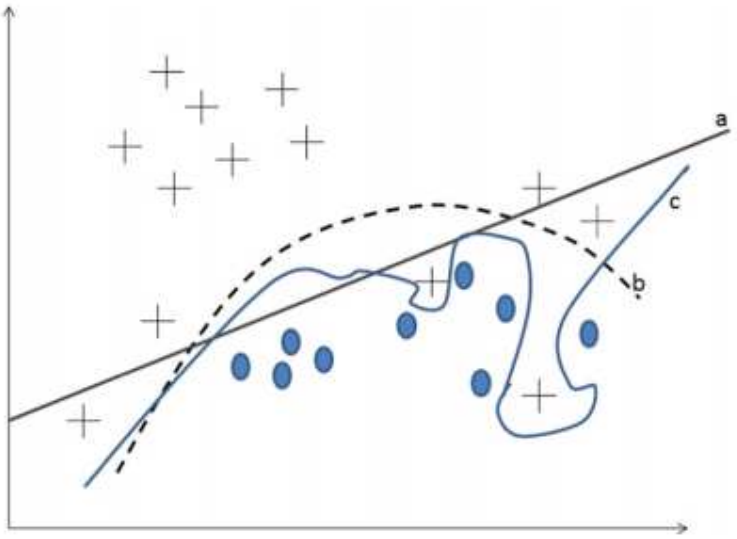
3.1.4 استراتژی قطع نمودار جستجو

اختصاصی سازی نمودار با رابطهٔ ترتیبی اختصاصی سازی که در L تعریف شد تولید می شود که به شیوه های مختلف قطع می گردد.

- جستجوس اولین پهنا: عناصر فضای جستجو (نمودار اختصاصی سازی) به شیوهٔ از بالا به پایین قطع می کند و الگوهای سطح به سطح تولید می کند. در صورتی که تمام موتیف ها دارای سطح یکسان باشند قبل از موتیف های خاص تر بررسی می شوند که به این apriori - like یا شبه استقرایی گفته می شود.
- جستجونی اولین پهنا (عرض): با جستجوی مورد بلافاصله بعدی هر الگوی تولید شده به سریع ترین راه حل ممکن دست میابد و تا جایی که ممکن باشد سطح الگو را قبل از بررسی اشکال همان سطح اختصاصی سازی می کند.

3.2 تناسب

داده های نمونه ارائه شده در شکل 3 را ملاحظه کنید. فرض کنید که $(|\bar{t}|)$ a + R(t) = t, Le, RC, (t). دایره ها و علامت های به اضافه در این شکل مطابق با مشاهدات دو دسته باشند (c,b,a) و مدل مختلف دسته بندی کننده را شکل دهند.



شکل 3

دسته بندی کننده (a) داده ها را براساس خط مستقیم دسته بندی می کند. این کار منجر به دسته بندی ضعیف می شود و به عنوان یک دسته بندی خطرناک تلقی می گردد. درعوض، دسته بندی کننده (c) داده های آموزشی را تناسب می کند و بیش از پیش به نمونه ها وابسته می گردد. درنتیجه، ظرفیت پیش بینی دیگر دسته های داده ها کاهش میابد. درنهایت، دسته بندی کننده (b) بهترین تعمیم دهی درخصوص فرایند آموزشی را ارائه می دهد و کمترین احتمال دسته بندی نادرست داده های جدید را به همراه دارد.

تعریف 3 (تناسب): $h \in H$ مجموعهٔ آموزشی S را درصورتی تناسب می کند که $h' \in H$ ایجاد گردد و دارای بیشترین خطای مجموعهٔ آموزشی و کمترین خطای آزمون در داده های آزمون باشد.

3.3 خصوصیات ویژگی

شناسایی ویژگی های مربوطه (اضافه) هدف اصلی الگوریتم انتخاب ویژگی است.

3.3.1 مرتبط بودن

در آموزش ماشینی، مرتبط بودن (ارتباط) شامل سه دسته می باشد: ارتباط قوی، ارتباط ضعیف و عدم ارتباط که به ترتیب اهمیت بیان شد. ویژگی های شدیداً مرتبط باید توسط الگوریتم انتخاب ویژگی حمایت کردند. با این

حال رویبز و دیگران (2006) بیان می کنند که هیچ تضمینی وجود ندارد که یک شاخصه ضرورتاً برای یک الگوریتم فقط به خاطر ارتباطش مفید باشد (یا برعکس). ویژگی های با ارتباط ضعیف را می توان تحت حمایت قرارداد یا نداد که این بستگی به اندازهٔ ارزیابی مثلاً درستی، سادگی) و سایر ویژگی های انتخابی دارد. ویژگی های نامربوط را باید حذف کرد.

3.3.2 ارتباط افزایشی

کارونا و فریتاژ (1994) ارتباط افزایشی را با در نظر گرفتن یکنواختی درستی و ترتیب مجموعهٔ زیرمجموعه ها تعریف می کند $P(R,\underline{C})$.

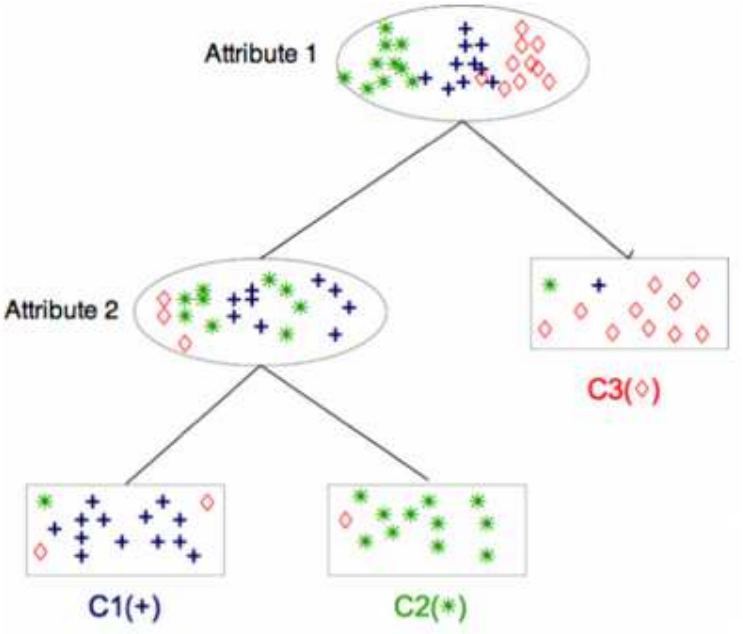
تعریف 4 (سودمندی افزایشی): با توجه به داده های D، الگوریتم آموزشی D و زیرمجموعه ایی از ویژگی ها x، ویژگی e از لحاظ افزایشی با توجه به x برای T مفید است. این درصورتی است که درستی این فرضیه که T با استفاده از گروه ویژگی ها $x \cup \{e\}$ تولید می کند در مقایسه با درستی حاصل از استفاده از فقط زیر مجموعه ویژگی های x بهتر باشد.

برای دستیابی به یک مجموعهٔ شاخصهٔ پیش گویانه از ویژگی ها تعریف فوق در کار حاضر مفید است.

3.3.3 حشو یا افزونگی

DT ها به صورت بازگشتی به گونه ایی که در شکل 4 نشان د اده شده پس از رویکرد بالا به پایین ساخته می شود. DT ها متشکل از یک ریشه، چند گره، شاخه و برگ می باشد. DT ها براساس استفاده از یک توالی ویژگی برای تقسیم مثال های آموزشی به n دسته رشد می کنند. ساخت درخت را می توان به صورت زیر تشریح کرد. اولاً، شاخصی که بهترین تقسیم مثال های آموزشی را تضمین می کند انتخاب می شود و زیرمجموعه های جمعیت به گره های جدید توزیع می شوند. همین عملیات برای هر گره (جمعیت زیرمجموعه) تکرار می شود تا این که عملیات تقسیم دیگری امکان پذیر نباشد. گره های انتهای متشکل از جمعیت های همان دسته (نسبت افزایش یافته در همان نوع DT) می باشند. عملیات دسته بندی یک فرد را به یک گرهٔ انتهایی (برگ) تخصیص می دهد. بدین ترتیب به مجموعه قوانین جهت یافته به سمت این برگ را تأمین می

کند. مجموعه قوانین DT را شکل می دهند. واضح است که اندازه و عدم قطعیت مثال های آموزشی موضوع اساسی در DTC می باشند و در نتیجه هدف ما این است که در حین حفظ عملکرد بالا، پیچیدگی را کمتر کنیم. عملکرد DT عمدتاً مبتنی بر تعیین اندازهٔ آن می باشد. ثابت شده است که اندازه درخت با تعداد مشاهدات داده های آموزشی رشد می کند.



شکل 4

4. رویکرد IUDT

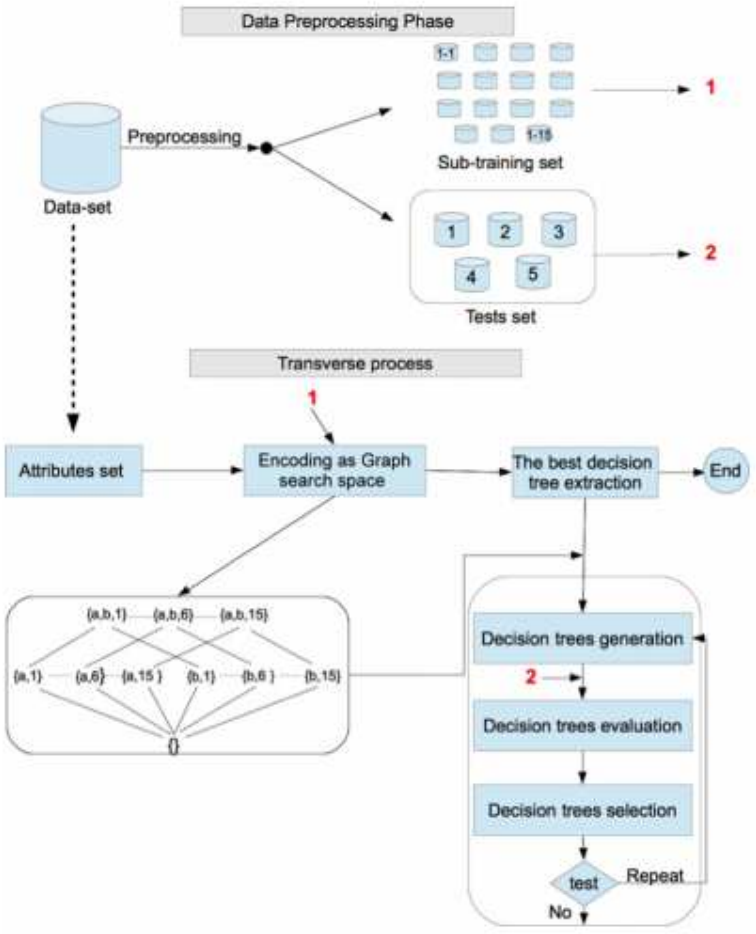
در این قسمت به تشریح ایدهٔ جایگزینی فاز پس هرس توسط انتخاب ومیژگی و کاهش مجموعهٔ داده ها خواهیم پرداخت. هدف اصلی این است که نشان دهیم که عملکرد یک DT هرس نشده را می توان با کمک این دو مرحله ارتقا داد. انتخاب ویژگی wrapper می تواند منجر به حذف ویژگی های اضافه و نامربوط گردد و کاهش داده ها مسئله پیچیدگی اندازه را کاهش می دهد. از این رو باید بهترین زیرمجموعهٔ ویژگی و زیرمجموعهٔ مثال های آموزشی مورد استفاده جهت ساخت بهترین درخت تصمیم هرس نشده t^* را تعیین کنیم. شاخصهٔ کلیدی روش پیشنهادی به شرح زیر می باشد:

1) پیش پردازش داده ها

2) تعریف ترکیبات ویژگی به عنوان فضای تحقیقاتی level wise (نمودار اختصاصی سازی) و

3) کاربرد یک جستجوی اولین پهنا محور در فضای جستجوی جهت یافتن بهترین درخت تصمیم هرس نشده *t

شکل 5 شمایی از روش پیشنهادی را نشان می دهد.



شکل 5: توصیف رویکرد IUDT.

4.1 پیش پردازش داده ها

تحلیل های تجربی درخصوص روش های هرس DT تقسیم داده های آموزشی و آزمون زیر را ارائه می دهد: 25 نمونه از 70٪ داده های آموزشی و 30٪ داده های آزمون، 9 نمونه از 60 درصد داده های آموزشی و 40٪ داده های آزمون و 10 نمونه از 25٪ داده های آموزشی و 75٪ داده های آزمون. از نمونه های متعدد در رد این

فرضیه استفاده شد که تقسیم رندومی واحد داده ها ممکن است منجر به نتایج غیرمعرف گردد. برای دستیابی به نتایج معرف در مرحلهٔ آزمایش و کاهش اندازهٔ DT بدون هرس قوی هدف به مجموعه داده ها 5 بار به صورت رندوم به یک مجموعهٔ آموزشی50٪ و یک مجموعه آزمون 50٪ تقسیم شدند. هر مجموعه آموزش بیشتر به صورت رندوم به سه زیرمجموعه آموزشی حاوی 50٪ کل مجموعهٔ آموزشی تقسیم شدند.

شکل 6

4.2 کد گذاری

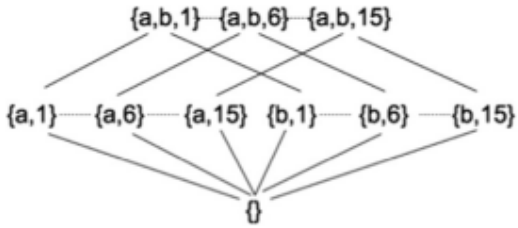
ما از الگوریتم wrapper برای انتخاب بهترین جفت (X,i) استفاده کردیم. X زیرمجموعه ویژگی E می باشد وi یک زیرمجموعه آموزشی از مجموعهٔ آموزشی I می باشد، (X,i) بهترین t* DT را می سازد و مسئله باید به صورت نمودار تحقیقاتی ارائه گردد. نمودار تحقیقاتی یک مجموعه مرتب P از مجموعه جفت های R براساس رابطه دوتایی $\underline{C}$ می باشد. کاربرد مرحلهٔ پیش پردازش داده ها باعث افزایش تعداد زیرمجموعه های قطع شده توسط الگوریتم wrapper می گردد یعنی $15*2^m$. مجموعه جفت های R به صورت زیر تعریف می شود:

$$R = \{ (X,i \mid X \underline{C} E,I \in 1)\}$$

مثال 2: برای یک مجموعهٔ داده که حاوی دو ویژگی {a,b} است، مجموعهٔ جفت ها به صورت زیر بیان می شود:

$$R = (a,1) , \dots , (a,15) , (b,1) , \dots (b,15) , (a,b,1) , \dots (a,b,15)\}$$

شکل 7 نمودار تحقیقاتی $P < R, \underline{C} >$ را نشان می دهد.



شکل 7

وظیفهٔ یافتن t* (بهترین DT هرس نشده را می توان به صورت تشریح کرد). یک پایگاه داده ایی D و زبان LT را برای عناصر R جفت های (X,i) که برای ساخت DT ها استفاده می شوند در نظر بگیرید. مجموعهٔ I از یک 15 زیرمجموعه آموزشی و مجموعهٔ S از 5 مجموعه آزمون 7 به ترتیب و همچنین تابع عینی F می باشد. مسئله اصلی استخراج t* است به گونه ایی که $t^* = \{t(X,i) \in LT \mid Max\ F(t)\}$.

هدف عینی هر درخت ساخته شده با i را با استفاده از دیگر زیرمجموعه های آموزشی $\forall a \in Si$ و مجموعه آزمون b ارزیابی می کند. هدف عینی با تابع $W: 1 \rightarrow |1 - 5|$ محاسبه می شود. $b = w(i)$ با توجه به اندازه X، تابع $W: I \rightarrow [1 .. 5]$ به صورت زیر داده می شود:

(2)

$$W(c) = \begin{cases} 1 & \text{درصورتی که } 1 \le c \le 3 \\ 2 & \text{درصورتی که } 4 \le c \le 6 \\ 3 & \text{درصورتی که } 7 \le c \le 9 \\ 4 & \text{درصورتی که } 10 \le c \le 12 \\ 5 & \text{درصورتی که } 13 \le c \le 15 \end{cases}$$

4.3 کاوش فضای تحقیق

جستجوی اولین پهنا جهت قطع نمودار اقتباس می شود. روش جستجویی پیشنهادی دارای شاخصه های شبیه به الگوریتم های کاووشی مجموعه آیتم های تکرار شوندهٔ استقرایی می باشد. جستجوی بهترین درخت هرس شده t^* با زیرمجموعه تهی $\emptyset$ آغاز می گردد. استثنائاً اندازه همزمان با پیش روی از $\emptyset$ به زوج های دارای یک ویژگی و یک شاخص آموزشی فرعی i افزایش میابد. پروسهٔ جستجو به شیوهٔ از پایین به بالا پیش می رود. در هر تکرار اندازهٔ زیرمجموعه های تازه کشف شده $(X,i) \in R$ با توجه به خاصیت ارتباط افزایش (IRP) که یک مستند $IRP: L_{k+1} \rightarrow \{0,2\}$ می باشد تا یک افزایش میابد. داوطلب های واسط جدید L_{k+1} با اتصال دو زیرمجموعه مشابه اما نسبتاً متفاوت (توسط یک ویژگی) که قبلاً کشف شده اند C_k ایجاد می شوند. داوطلب های جدید C_{k+1} در واقع L_{k+1} هستند که خاصیت ارتباط نسبی (PRP) که مستند $PRP: L_{k+1} \rightarrow (0,1)$ می باشد را تأمین می کند. این فرایند مرتباً تکرار می شود و بین فازهای ارزیابی و تولید داوطلب قرار می گیرد تا ایبن که هیچ داوطلب جدیدی در C_{k+1} $(C_{k+1} = 0)$ وجود نداشته باشد. برای هر زوج بررسی شده $(X,i) \in R$ یک DT، با استفاده از فقط یک زیرمجموعه از مثال های i ساخته می شود. همان طور که قبلاً نشان داده شد، توالی ویژگی ها براساس معیارهای تقسیم (مثلاً شاخص گینی، شاخص های مبتنی بر خلوص و شاخص twoing) مرتب می شود. در این مرحله، رویکرد پیشنهادی دارای دو فرصت است: تعریف مدل DT شخصی و مدل پیش تعریف شده (مثل BFTree، J48، REPTee، Simplecart) که با محدودیت اجتناب از فاز هرس مواجه می شود. هرDT ساخته شده براساس فرمول (3) ارزیابی می گر دد. الگوریتم 1 که کاذب روش ادیب پیشنهادی را ارائه می دهد.

الگوریتم 1 . بهترین تابع DT جستجوی هرس نشده

ورودی: پایگاه داده D ، زبان LT ، مسندهای IRP و PRP

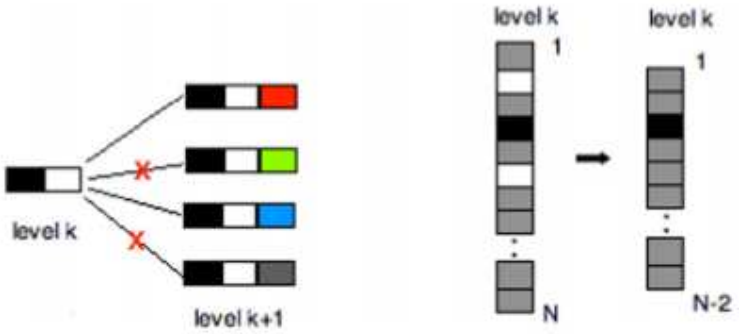
خروجی: $^{DT}\, t^*(X,i).$ بهینه

1: $L_1 = \{(e,i) | e \in E, i \in I\}$
2: $C_1 = \{(e,i) | (e,i) \in L_1, IRP(t(e,i)) \text{ and } PRP(t(e,i))\}$
3: $t^* = Best-of(C_1)$
4: $i=1$
5: **while** C_i is not empty **do**
6: $L_{i+1} = \{(Y,i) | \forall (X,i) \in C_i, \forall e \in E, Y = X \cup e\}$
7: $C_{i+1} = \{(X,i) | (X,i) \in L_{i+1}, IRP(t(X,i)) \text{ and } PRP(t(X,i))\}$
8: $t' = Best-of(C_{i+1})$
9: **if** $Accuracy(t^*) < Accuracy(t')$ **then**
10: $t^* = t'$
11: **end if**
12: $i=i+1$
13: **end while**
14: **return** t^*

امروزه، هر الگوریتم پنهان سازی که بیش از 30 مشخصه را در نظر می گیرد هزینه ی محاسباتی بسیار زیادی دارد. روش های زیادی برای سرعت بخشی به فرایند پیمایش وجود دارد. مفاهیم اولیه مبتنی بر به حداقل رساندن زیر مجموعه های ارزیابی یا استفاده از یک جستجوی تصادفی شده هستند. برای فائق آمدن بر این مشکل، وابستگی افزایشیو نسب 5 درصدی از ویژگی های دقت زیرمجموعه اتخاذ شده اند. دقت DT ساخته شده با استفاده از زوج (X,i) به شرح زیر محاسبه شده است:

$$Accuracy(t(X,i)) = average\left(\sum_{a \neq i}^{I} Accuracy(t(X,i),a) + Accuracy(t(X,i),w(i))\right). \quad (3)$$

شکل 8 نمونه ای از IRP و PRP را نشان می دهد. هدف IRP حذف L_{K+1} کاندیدتولید شده با اضافه کردن مشخصه های خاکستری و سبز است، و هدف PRP حفظ زیر مجموعه های مشخصه ای است که دارای دقتی هستند که بیشتر از دقت بهترین کاندیدهای متوسط (L_{K+1}) منهای 5 ٪ می باشد (یعنی در سمت راست شکل 8، کاندید سیاه بهترین است، کاندیدهای سفید حذف شده اند، و تنها کاندیدهای خاکستری نگه داشته شده اند).



شکل 8

5- روایی IUDT

روش پیشنهادی روی 10 پایگاه داده یادگیری ماشین استاندارد پیاده سازی و تست شده است که این پایگاه های داده از مجموعه UCI استخراج شده اند. کد آن در جاوا با استفاده از چارچوب WEKA و کتابخانه GUAVA Google پیاده سازی شده است. آزمایشاتی انجام شدند و با نتایج حاصل شده از DTهای هرس شده WEKA اصلی مقایسه شدند، یعنی DTP (تکنیک ساخت درخت تصمیم گیری با استفاده از فاز هرس کردن)، یعنی پیاده سازی الگوریتم انتخاب مشخصه که شبیه الگوریتم پنهان سازی (wrapper) عمل می کند، یعنی IUDTAS (ساخت درخت تصمیم هرس نشده بهبود یافته تنها با استفاده از انتخاب مشخصه) و یک الگوریتم ثانویه که تنها مرحله نمونه گیری را در نظر می گیرد، یعنی IUDTSE (ساخت درخت تصمیم هرس نشده بهبود یافته با استفاده از تنها داده های نمونه گیری شده).

Dataset	*No. classes*	*No. attributes*	*No. instances*	*%Base error*
Zoo	7	17	101	59.40
Glass	7	9	214	35.52
Sonar	2	60	208	46.64
Ecoli	8	7	336	57.44
Diabetes	2	8	768	34.90
Hepatitis	2	19	155	20.65
Tic-tac-toe	2	9	958	34.65
Breast-cancer	2	9	286	29.72
Primary-tumor	21	17	339	75.22
Waveform-5000	3	40	5000	66.16

جدول 1

پایگاه های داده ای در جدول 1 شرح داده شده اند. ستون %Base Error (درصد خطای پایه) به درصد خطای حاصل شده اشاره می کند که در صورتی حاصل می شود که تکراری ترین رده همیشه پیش بینی شده باشد. به علاوه، پایگاه داده های انتخاب شده نشان دهنده ی کاربردهای مختلفی هستند. سه DT به نام J48، SimpleCart و REPTree برای اعمال روش های مختلف هرس کردن با پیاده سازی WEKA استاندارد خود انتخاب شدند. مشخصه های DT اصلی در نظر گرفته شده در آزمایشات در جدول 2 ارائه شده اند.

DTs	Split criteria	Pruning method	Principal standard options
J48 (C 4.5)	Gain ratio	Error-based	The confidence pruning is 0.25 The minimum number of instances at leaves is 2 One fold is used for pruning Consider the sub-tree raising operation when pruning
REPTree	Gain ratio	Reduced-error	The minimum number of instances at leaves is 2 One fold is used for pruning
SimpleCart	Gini index	Error-complexity	Binary split for nominal attributes The minimum number of instances at leaves is 2 One fold is used for pruning Five fold internal cross-validation

جدول 2

Data	# Attributes	Size	Accuracy
Zoo	3	11	90.50
Glass	5	29	70.56
Sonar	7	17	91.10
Ecoli	5	15	85.26
Diabetes	5	25	80.92
Hepatitis	6	15	84.09
Tic-tac-toe	8	73	84.70
Breast-cancer	5	51	79.02
Primary-tumor	10	42	53.40
Waveform-5000	7	179	83.80

جدول 3

Data	# Attributes	Size	Accuracy
Zoo	5	13	95.00
Glass	5	27	76.40
Sonar	6	17	91.58
Ecoli	5	15	83.33
Diabetes	6	57	77.60
Hepatitis	3	9	79.22
Tic-tac-toe	9	76	81.83
Breast-cancer	5	68	78.49
Primary-tumor	10	40	56.30
Waveform-5000	6	295	83.08

جدول 4

Dataset	# Attributes	Size	Accuracy
Zoo	5	13	95.00
Glass	4	19	72.66
Sonar	6	17	94.95
Ecoli	3	19	85.56
Diabetes	5	61	78.71
Hepatitis	6	17	89.93
Tic-tac-toe	9	49	93.05
Breast-cancer	6	31	77.92
Primary-tumor	10	43	52.51
Waveform-5000	6	231	82.19

جدول 5

آزمایشات مبتنی بر DTهای گزارش شده در جدول 2 هستند و در زمینه ی استاندارد خودشان پیاده سازی شده اند، اما بون فاز هرس کردن (DTهای هرس نشده).

جدول 3، 4 و 5 دقت رده بندی را به ترتیب برای J48 ، REPTree و SimpleCart، در زمانی که الگوریتم IUDT در ده پایگاه داده آزمایشی اعمال شده است لیست می نمایند. جداول نشان می دهند که تعداد مشخصه های استفاده شده برای ساختن DT کمتر از تعداد ویژگی های داده ای است و تنها حدود یک ششم از آنها در مورد مجموعه ویژگی های بزرگ استفاده شده اند. نتایج دقت نشان می دهند که DT تنها زمانی که تکراری ترین رده بطور مداوم پیش بینی شده از دقت مبتنی بر درصد پیشی می گیرد.

Dataset	J48		REPT		SCart	
	Size	Accuracy	Size	Accuracy	Size	Accuracy
Zoo	13	95.20	1	43.60	1	43.60
Glass	35	80.56	9	72.33	9	69.90
Sonar	19	89.42	3	77.88	9	80.76
Ecoli	25	82.26	13	82.26	15	82.02
Diabetes	31	81.40	39	80.57	5	77.55
Hepatitis	9	87.01	7	85.71	17	90.12
Tic-tac-toe	97	84.88	64	79.16	45	92.94
Breast-cancer	20	71.04	1	72.16	1	72.16
Primary-tumor	46	52.30	20	44.85	21	48.04
Waveform-5000	341	85.36	87	80.28	49	79.36

جدول 6

Dataset	J48		REPT		SCart	
	Size	Accuracy	Size	Accuracy	Size	Accuracy
Zoo	11	94.00	1	45.00	7	55.50
Glass	17	74.29	25	74.06	17	71.72
Sonar	11	85.09	9	84.61	9	83.89
Ecoli	17	84.11	9	84.22	29	84.52
Diabetes	27	81.70	45	81.90	47	82.29
Hepatitis	11	89.61	9	87.66	13	88.61
Tic-tac-toe	73	84.70	91	83.97	49	92.95
Breast-cancer	27	77.62	61	73.07	7	77.27
Primary-tumor	49	50.88	50	50.14	45	54.28
Waveform-5000	197	83.98	191	84.47	155	82.48

جدول 7

جدول 6 ، DTهای WEKAی ساخته شده با استفاده از فاز هرس کردن را نشان می دهد. طرح این درخت ها اغلب مبتنی بر فاز ساخت (رشد و هرس کردن) می باشد که در آن 50 درصد نمونه ها به عنوان مجموعه ی آزمایشی مورد استفاده قرار گرفته اند.

همان طور که در روش پیشنهادی گفتیم، هر پایگاه داده به طور انتخابی طی 5 مرحله به مجموعه های آزمایشی و مجموعه های تست تقسیم شد. نتایج ارائه شده در جدول 6 نشان دهنده ی دقت پیش بینی میانگین در 5 پایگاه داده ی i-test است که مقایسه ای از دقت و اندازه بین روش پیشنهادی و پیاده سازی استاندارد DTP ارائه می کند، یعنی با استفاده از فاز هرس کردن.

Dataset	# Attributes	Size	Accuracy
Zoo	5	11	93.60
Glass	2	27	79.81
Sonar	3	13	87.30
Ecoli	3	17	80.71
Diabetes	3	11	78.95
Hepatitis	3	9	85.45
Tic-tac-toe	5	124	84.63
Breast-cancer	2	12	75.94
Primary-tumor	8	38	53.72
Waveform-5000	6	189	81.21

جدول 8

Dataset	# Attributes	Size	Accuracy
Zoo	4	11	91.20
Glass	2	45	80.93
Sonar	3	27	87.11
Ecoli	3	53	84.52
Diabetes	3	89	84.79
Hepatitis	3	27	89.09
Tic-tac-toe	4	94	78.91
Breast-cancer	2	21	75.94
Primary-tumor	8	54	53.60
Waveform-5000	8	505	86.01

جدول 9

جدول 6 نشان می دهد که نتایج یک مدل متفاوت از نتایج مدل دیگر است. معمولاً درخت های J48 بزرگتر از DTهای REPTree و SimpleCArt هستند. این رخداد به علت تکنیک هرس کردن اعمال شده می باشد. بر خلاف تکنیک های خطاهای کاهش یافته (REPTree) و هرس کردن پیچیدگی خطا (SimpleCart) تکنیک EBP توسط J48 به کار گرفته شد (و در بخش 2 در مورد آن بحث شده است) و با یکی از زیردرخت های یک زیر درخت X که برای هرس شدن انتخاب شده بود، پیوند یافت. علاوه بر این، دقت DTهای J48 بهترین مورد برای شش مجموعه ی داده ای است. واضح ایت که این نتایج تصویری از هرس زنی بیش از حد و هرس کردن کم در نظر گرفته شده در برخی پایگاه های داده ای را ارائه می نماید. هرس زنی بیش از حد می تواند در مورد پایگاه داده های Zoo و Breast Cancer با استفاده از REPTree و SimpleCart مشاهده گردد. یک مورد کم هرس کردن در پایگاه داده ی Ecoli شرح داده شده که در آن دقت مدل های J48 و REPTree وقتی اندازه ی J48 خیلی بزرگتر می شود، یکسان است. متعاقباً می توان استنباط کرد که J48 یک DT کم هرس شده است.

برای نشان دادن کارایی روش پیشنهادی در مقابل الگوریتمی که تنها یک فرایند نمونه گیری را بدون انتخاب ویژگی اعمال می نماید، ما DT ها را با استفاده از 15 زیر مجموعه یادگیری از نمونه های آزمایشی i-j تولید کرده ایم. نتایج ارائه شده در جدول 8 دقت پیش بینی میانگین در 5 پایگاه داده ی i-test را نشان می دهد. می توان دید که اگر چه اندازه های درخت ها بیشتر از DT های هرس شده است، دقت آن ها بسیار بهتر است.

دقت الگوریتم هایی که تنها گزینش مشخصه را بدون نمونه گیری داده ها اعمال می نمایند، در جداول 10-8 گزارش شده است. این نتایج با پیاده سازی یک الگوریتم پنهان سازی بدست آمده اند که از فضای گراف جستجو می گذرند، همانند روش پیشنهادی، و از زیرمجموعه ی 50٪ تصادفی (نمونه ها) به عنوان داده های یادگیری استفاده می کنند. بهترین DT های اعتبار سنجی شده بواسطه ی خروجی پیش پردازش شده ی یکسان مورد استفاده در روش پیشنهادی انتخاب شده اند.

Dataset	# Attributes	Size	Accuracy
Zoo	5	11	94.40
Glass	2	47	82.80
Sonar	4	27	91.73
Ecoli	3	45	81.66
Diabetes	2	145	82.60
Hepatitis	3	27	89.09
Tic-tac-toe	6	105	90.43
Breast-cancer	2	17	75.94
Primary-tumor	7	61	55.14
Waveform-5000	8	233	86.41

جدول 10

Dataset	J48		REPT		SCart	
	Size	Accuracy	Size	Accuracy	Size	Accuracy
Zoo	+	−	+	+	+	+
Glass	+	−	−	+	−	+
Sonar	+	+	−	+	−	+
Ecoli	+	+	−	+	−	+
Diabetes	+	=	−	−	−	+
Hepatitis	−	−	−	−	=	=
Tic-tac-toe	+	=	−	+	−	=
Breast-cancer	−	+	+	+	−	+
Primary-tumor	+	+	−	+	−	+
Waveform-5000	+	−	−	+	−	+

جدول 11

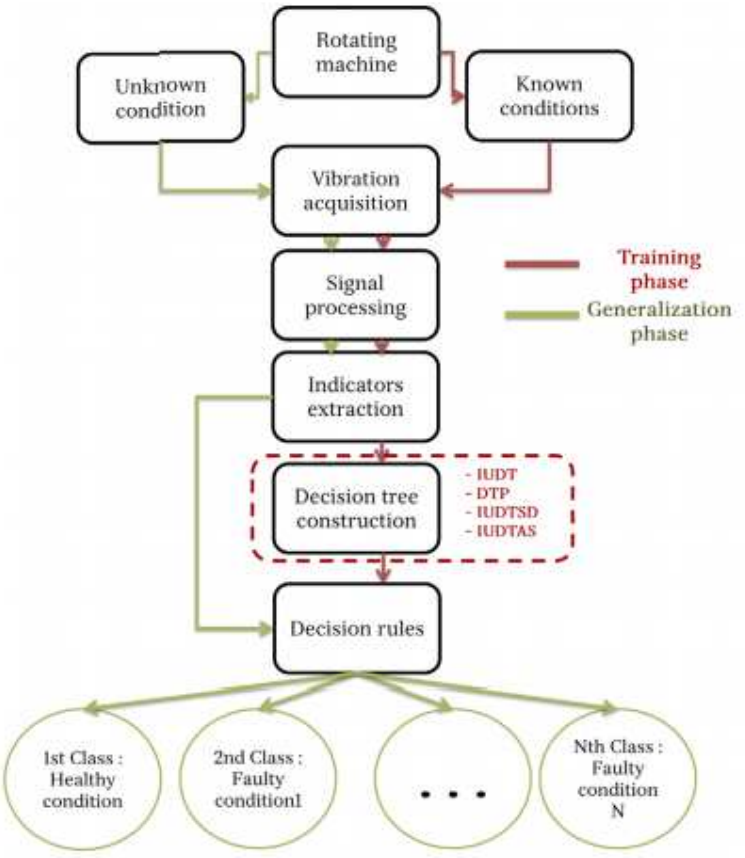
Dataset	J48		REPT		SCart	
	Size	Accuracy	Size	Accuracy	Size	Accuracy
Zoo	=	-	+	+	-	+
Glass	-	-	-	+	-	=
Sonar	-	+	-	+	-	+
Ecoli	+	+	-	=	+	+
Diabetes	+	=	-	-	-	-
Hepatitis	-	-	=	-	-	+
Tic-tac-toe	=	=	+	-	=	=
Breast-cancer	-	+	-	+	-	+
Primary-tumor	+	+	+	+	+	-
Waveform-5000	+	=	-	+	-	=

جدول 12

Dataset	J48		REPT		SCart	
	Size	Accuracy	Size	Accuracy	Size	Accuracy
Zoo	-	=	-	+	-	+
Glass	-	-	+	-	+	-
Sonar	-	+	+	+	+	+
Ecoli	+	+	+	-	+	+
Diabetes	-	+	+	-	+	-
Hepatitis	-	-	+	-	+	=
Tic-tac-toe	+	=	+	+	+	+
Breast-cancer	-	+	-	+	-	+
Primary-tumor	-	-	+	+	+	-
Waveform-5000	+	+	+	-	+	-

جدول 13

جداول 13-11 نتایج حاصل شده از UDT را بر مبنای تلفیقی از نمونه گیری و انتخاب ویژگی با WEKAی اصلی هرس کردن DT (جدول 11) مقایسه می کند، یعنی روشی که تنها نمونه گیری داده ها را اعمال می نماید، یعنی IUDTSD (جدول 12)، و روشی که تنها انتخاب مشخصه را بکار می برد ، یعنی IUDTAS (جدول 13). نماد + اشاره بر این دارد که روش پیشنهادی نتایج بهتری نسبت به روش مقایسه ای ایجاد می کند. در غیر اینصورت ما از نماد – استفاده می کنیم. شایان ذکر است که DT های بیش از حد هرس شده با اندازه 1 همیشه به عنوان بدترین موارد در نظر گرفته شده اند و دقت در صورتی که تفاوت میان IUDT و روش مقایسه ای در بازه ی [1- , +1] باشد، یکسان در نظر گرفته می شوند. در غیر این صورت درخت با بیشترین دقت بهترین مورد در نظر گرفته می شود.


Rotating machine
Unknown condition
Known conditions
Vibration acquisition
Signal processing
Indicators extraction
Decision tree construction
- IUDT
- DTP
- IUDTSD
- IUDTAS
Decision rules
Training phase
Generalization phase
1st Class : Healthy condition
2nd Class : Faulty condition1
. . .
Nth Class : Faulty condition N

شکل 9

شکل 10

مقایسه در جدول 11 به وضوح این مورد را آشکار می کند که روش پیشنهادی عموماً دقیق تر از DTP است (در مورد REPTree و SimpleCart). در مورد J48 هر روش روی 4 پایگاه داده بهتر عمل می کند. DTهای استاندارد هرس شده در REPTree و SimpleCart کوچکترند، اما J48 کوچکترین DT را برای IUDT ارائه می نماید.

جدول 12 نتایج حاصل شده از IUDT را با آنهایی که از کاربرد نمونه گیری داده ها، یعنی IUDTSD حاصل شده اند را مقایسه می کند. بوضوح مشخص است که دقت IUDT بسیار بهتر از IUDTSD است. اما در مقابل نتایج دقت، نتایج اندازه نشان می دهند که روش نمونه گیری مزایایی در خود دارد. جدول نشان می دهد که نتایج دقت IUDT وقتی از مدل های REPTree و SimpleCart استفاده می شود بهتر هستند، اما DTهای ساخته شده خیلی بزرگتر از نتایج روش نمونه گیری هستند. در مورد J48 ما توازنی میان دقت و نتایج اندازه داریم. این نشان می دهد که استفاده از نمونه گیری داده ها قطعاً DT با پیچیدگی کمتری ارائه می کند، هر چند هزینه آن قدرت بیشتر است.

جدول 13 نتایج IUDT را با IUDTAS مقایسه می کند که در آن تنها انتخاب مشخصه به کار برده شده است. واضح است که نتایج دقت و اندازه از روش پیشنهادی بهتر هستند. بطور معمول جدول نشان می دهد که IUDT نتایج دقت و اندازه ی بهتری با مدل های REPTree و SimpleCart ارائه می دهد، به جز در مورد اندازه های DT ی J48 که بزرگتر از آنهایی هستند که روش انتخاب مشخصه را به کار می برند. این نتایج نشان می دهند که استفاده از تلفیقی از انتخاب ویژگی و نمونه گیری داده ها DT قدرتمندتر و با پیچدگی کمتری ارائه می نماید.

6- کاربرد تشخیص نقص در یک ماشین دوار

حال ما کاربرد روش پیشنهادی را برای تشخیص نقصان در ماشین های دوار در نظر می گیریم. برخی از نقصان های اصلی که روی عملکرد صحیح چنین ماشین هایی تأثیر می گذارند، به طور تجربی و با آزمایش روی بستر تست تولید شده اند. وظیفه ی نمایش شرایط می تواند به وظیفه رده بندی تبدیل شود که در آن هر شرط (خوب و عیب دار) به عنوان یک رده در نظر گرفته شده است. هدف استخراج اطلاعات از حسگرهای ارتعاش

برای نشان دادن شرایط فعلی ماشین است. روش های بررسی شده در این مقاله پس از آن برای جستجو به دنبال ساخت غیر پیچیده ی قوانین تصمیم موثر استفاده شده اند و به دنبال آن شماتیک در شکل 9 نشان داده شده است.

6-1- مطالعه تجربی

بستر تست نمایش یافته در شکل 10 از سه شافت، دو دنده، یکی با شش دندانه و دیگری با 48 دندانه، 6 مکان مرتبط، یک زوج و یک تسمه دندانه دار تشکیل شده است. سیستم با یک موتور الکتریکی با سرعت متغیر DC همراه با طیف سرعت دوار از 0 تا 1500 دور در دقیقه مشتق شده است.

اثرات ارتعاش برای نمایش دادن شرایط بستر تست مورد استفاده قرار گرفته اند. سیگنال های ارتعاض با استفاده از شتاب سنج ثابت شده روی مکان مربوطه که به یک سیستم اکتساب داده ای متصل است که با نرم افزار OROS تجهیز شده است، بدست آمدند. اثرات ارتعاش تحت سه سرعت دوار مختلف (300، 900 و 1500 دور در دقیقه) تحت یک شرایط عملکردی نرمال و با سه نقصان مختلف ثبت و ضبط گردیدند: عدم توازن توده، نقص چرخ دنده و تسمه عیب دارد.

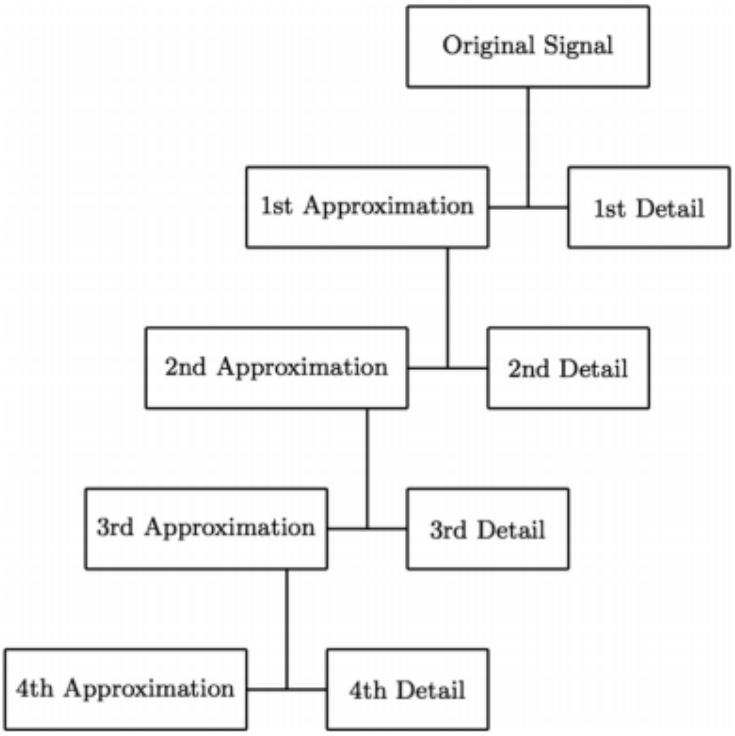
6-2- پردازش سیگنال

تبدیل موج کوچک بطور گسترده ای در دو دهه اخیر مورد مطالعه قرار گرفته و استفاده آن رشد قابل توجه و جذابیتی در تحلیل ارتعاش داشته است.. فرمولاسیون تنوع گسسته آن (DWT) که مستلزم زمان محاسباتی کمتری نسبت به شکل پیوسته ی آن است در معادله زیر نشان داده شده است:

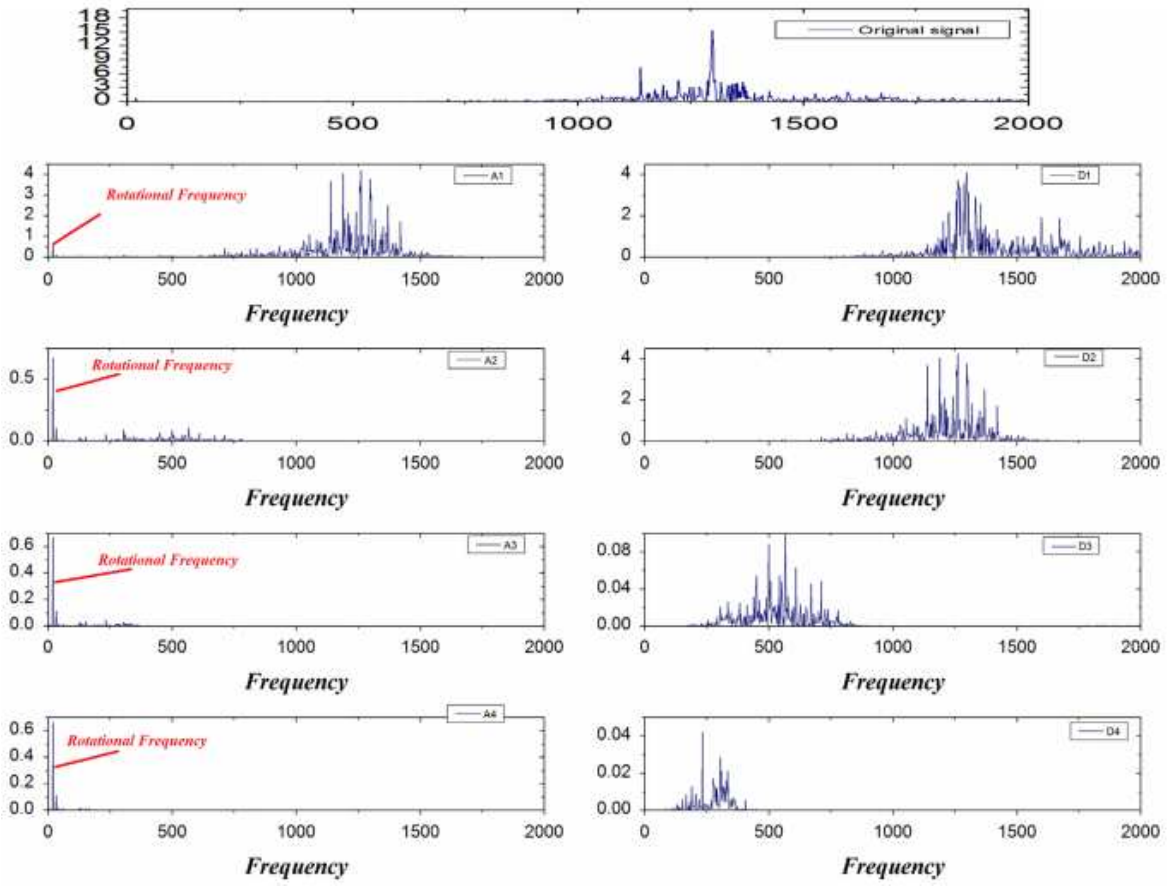
$$DWT(j,k)=\frac{1}{\sqrt{2^j}}\int_{-\infty}^{+\infty}s(t)\psi^*\left(\frac{t-2^jk}{2^j}\right)dt \qquad (4)$$

Mallat (1989) استفاده موثر از تبدیل موج کوچک گسسته با اعمال فیلترهای پیشین روی چندین سطح مختلف را معرفی کرده است. سیگنال های منتج شده ضرایب تقریب و ضرایب جزئیات نامیده شده اند. برای غلبه بر نمونه گیری کم به شمار آورده شده بواسطه ی تجزیه، ضرایب به بازسازی فیلترها برای ایجاد سیگنال های

جدید که تقریب ها (A) و جزئیات (D) نامیده شده اند مرتبط گشته است. شکل 11 اصل تجزیه DWT را شرح می دهد. در مطالعه ی حاضر موج کوچک Daubechies با دو سطح تجزیه برای استخراج تقریب ها و جزئیات سیگنال های اصلی مورد استفاده قرار گرفتند. فضای فرکانس با اعمال یک تبدیل فوریه سریع (FFD) به هر سیگنال اصلی و همچنین به هر یک از تقریب ها، جزئیات و شرایط همان گونه که در شکل 12 برای نمونه عدم توازن توده تحت سرعت چرخشی 900 دور در دقیقه نشان داده شده است، تبدیل یافت.



شکل 11



شکل 12

3-6 استخراج شاخص ها

سی و پنج سیگنال اصلی تحت چهار شرایط مختلف عملکردی (رده ها) و سه سرعت چرخش مختلف ثبت و ضبط شدند که جمعاً 420 سیگنال اصلی حاصل می نماید. عامل ارشد، مربع میانگین ریشه، مورب بودن، و واریانس از شکل های موقتی این سیگنال ها استخراج شده بودند. از طیف فرکانس ما ماکسیمم دامنه، فرکانس آن، فرکانس دومین بالاترین دامنه، وقفه میان دو فرکانس بالاترین دامنه، وقفه میانگین میان چهار بالاترین فرکانس دامنه، و مربع میانگین ریشه را بدست آوردیم. این 11 شاخص نیز از هر یک از 4 سیگنال حاصل شده از کاربرد ترکیب DWT استخراج شده اند که جمعاً 55 شاخص از سیگنال اصلی ارائه کردند.

DTs	IUDT		DTP		IUDTSD		IUDTAS	
	Accuracy	Size	Accuracy	Size	Accuracy	Size	Accuracy	Size
J48	89.52	21	91.66	33	92.38	19	96.66	29
REPTree	98.41	21	85.31	27	92.14	19	90.47	29
SimpleCart	95.37	23	92.30	35	90.47	19	96.66	25

جدول 14

AOT	Indicator	Originating signal
1	Skewness	Original signal
2	Root mean square of the spectra	Original signal
3	Signal root mean square	First approximation signal
4	Crest factor	First detail signal
5	Mean interval between the frequencies of the four maximum amplitudes	Original signal

جدول 15

4-6- نتایج و بحث

نتایج آزمایشی با استفاده از سه الگوریتم DT در جدول 14 داده شده اند. این نتایج اولویت عمومی بر حسب دقت رده بندی را نشان می دهد. درخت ها با اندکی پیچیدگی کمتر با روش IUDTSD ساخته شدند اما این درخت ها کارایی بسیار کمتری از خود به نمایش گذاشتند. IUDT نیز می تواند برای ارائه ی تعمیم بیش از حد، همانطور که در بخش 2-3 کشف شد، در نظر گرفته شود.

بهترین نتایج با استفاده از REPTree بدست آمدند و قوانین تصمیم تولید شده بواسطه ی الگوریتم های آزمایشی در ضمیمه داده شده اند. بر حسب خروجی IUDT ، REPTree بهترین دقت رده بندی را ارائه داده است. جدول 15 شاخص های انتخابی مورد استفاده برای ساختن آن در ترتیب حضور در درخت (AOT) را لیست می کند و نشان دهنده ی سیگنالی است که هر شاخص از آن نشئت گرفته است.

از جدول 15 می توان مشاهده کرد که تنها شاخص های مستخرجه از سیگنال اصلی و تجزیه سطح اول برای ساختن بهترین درخت استفاده شدند که در آن 4 سطح تجزیه انجام شده است. با استخراج انحصاری شاخص

های حفظ شده در DTC می توان هم در حافظه ذخیره سازی و هم در زمان محاسبه صرفه جویی کرد، خصوصاً در وظایف تشخیص و نگهداری که مستلزم بایگانی داده ها در طولانی مدت هستند.

7- نتیجه گیری

محبوبیت DT قویاً مرتبط با سادگی آنها، راحتی فهم آن ها و تشابه نزدیک به منطق انسانی است. اما هر مدل DT مزایا و محدودیت های خاص خودش را دارد و این است که انتخاب یک DT را سخت می کند. انتخاب مدل قویاً وابسته به عملکرد روش هرس کردن است که کاربران را مجبور می کند مدلی بر طبق نیازهای خود انتخاب کنند. این مسأله اشاره بر این دارد که کاربران UDT باید تمام تکنیک های هرس کردن و رشد را مطالعه نمایند تا بتوانند مناسب ترین مدل را انتخاب کنند که البته این کار کار مشکلی است. هنگام ساختن DT ها فاز هرس کردن برای کاهش پیچیدگی و سایز مدل مفید است. اما اغلب مشکلی در دقت بوجود می آورد. خصوصاً برای پایگاه داده های کوچک. ما در این مقاله الگوریتم بهبود یافته ای بدون فاز هرس کردن ارائه کرده ایم که انتخاب مشخصه و فرایندهای نمونه گیری داده ها را با هم تلفیق می کند. نتایج آزمایشی روی 10 پایگاه داده ی محک نشان داد که روش پیشنهادی دقت DT ها را افزایش می دهد و بطور موثری اندازه ی آن ها را کاهش می دهد و این از مشکلات کم هرس کردن یا بیش از حد هرس کردن جلوگیری می نماید. در نهایت روش IUDT در کاربرد عملی تشخیص نقصان در تجهیزات دوار اعمال شد. این مورد کارایی آن را بر حسب دقت رده بندی و پیچیدگی درخت نشان داده است. علاوه بر این استخراج تنها شاخص های منتخب بواسطه ی الگوریتم پیشنهادی امکان صرفه جویی قابل توجهی در حافظه ی ذخیره سازی و زمان محاسبه را فراهم می کند. ما نتیجه می گیریم که روش پیشنهادی عملی است، خصوصاً در مورد پایگاه داده های کوچک و وقتی که دانش ادراکی در مورد مشخصه های داده ها در دست نیست، روش موثری است.