مقاله پژوهشی

طبقه‌بندی خودآموز نیمه‌نظارتی مبتنی بر ساخت همسایگی

منا عمادی، جعفر تنها، محمدابراهیم شیری و مهدی حسین‌زاده اقدم

چکیده: به‌کارگیری داده‌های بدون برچسب در خودآموزی نیمه‌نظارتی می‌تواند به طور قابل توجهی دقت طبقه‌بند نظارت‌شده را بهبود بخشد، اما در برخی موارد ممکن است دقت طبقه‌بندی را به مقدار چشم‌گیری کاهش دهد. یکی از دلایل چنین تنزلی، برچسب‌گذاری اشتباه به داده‌های بدون برچسب می‌باشد. در این مقاله، روشی را برای برچسب‌گذاری با قابلیت اطمینان بالا به داده‌های بدون برچسب پیشنهاد می‌کنیم. طبقه‌بند پایه در الگوریتم پیشنهادی، ماشین بردار پشتیبان است. در این روش، برچسب‌گذاری فقط به مجموعه‌ای از داده‌های بدون برچسب که از مقدار مشخصی به مرز تصمیم نزدیک‌تر هستند انجام می‌شود. به این داده‌ها، داده‌های دارای اطلاعات می‌گویند. اضافه‌شدن داده‌های دارای اطلاعات به مجموعه آموزشی در صورتی که برچسب آنها به درستی پیش‌بینی شود در دستیابی به مرز تصمیم بهینه تأثیر به‌سزایی دارد. برای کشف ساختار برچسب‌زنی در فضای داده از الگوریتم اپسیلون- همسایگی (DBSCAN) استفاده شده است. آزمایش‌های مقایسه‌ای روی مجموعه داده‌های UCI نشان می‌دهند که روش پیشنهادی برای دستیابی به دقت بیشتر طبقه‌بند نیمه‌نظارتی خودآموز به نسبت برخی از کارهای قبلی عملکرد بهتری دارد.

کلیدواژه: **الگوریتم اپسیلون- همسایگی (DBSCAN)، الگوریتم خودآموزی، طبقه‌بندی نیمه‌نظارتی، ماشین بردار پشتیبان.**

۱- مقدمه

الگوریتم‌های یادگیری ماشین به سه دسته الگوریتم‌های نظارتی، بدون نظارت و نیمه‌نظارتی تقسیم می‌شوند. در یادگیری نظارتی، مجموعه‌ای از جفت‌های ورودی و خروجی به ماشین داده می‌شود و ماشین تلاش می‌کند تا تابعی از ورودی به خروجی را یاد بگیرد. یادگیری نظارت‌شده یک مسئله تحقیقاتی فعال در زمینه داده‌کاوی و یادگیری ماشین است و تا کنون به طور گسترده‌ای در حفاظت از سیستم‌های قدرت، داروهای بیولوژیکی، تشخیص چهره، پردازش تصویر و تشخیص شیء مورد استفاده قرار گرفته است [۱] تا [۶]. یادگیری نظارتی برای آموزش طبقه‌بندی کارآمد به نمونه‌های با برچسب متکی است اما محدودیت آن در دستیابی به داده‌های برچسب‌دار است، زیرا در دنیای واقعی اکثر داده‌ها بدون برچسب هستند و تنها درصد کمی از آنها برچسب دارند. از طرفی در عمل به دست آوردن داده‌های برچسب‌دار، فرایندی بسیار پرهزینه، زمان‌بر و دشوار است.

یادگیری بدون نظارت، سعی بر یافتن الگوها و ساختار خاص در داده‌های بدون برچسب دارد و آموزش واقعی در آن رخ نمی‌دهد. از آنجایی که داده‌های بدون برچسب به وفور در دنیای واقعی یافت می‌شوند، دسته‌بندی‌ای به نام طبقه‌بندی نیمه‌نظارتی مطرح می‌شود که نمونه‌ای از یادگیری به همراه یافتن راهی جهت بهبودبخشیدن به یادگیری نظارتی با استفاده از داده‌های بدون برچسب است. در یادگیری نیمه‌نظارتی، حجم عظیمی از داده‌های بدون برچسب وجود دارد و فقط مقدار اندکی از داده‌ها برچسب‌دار هستند، از این رو برچسب‌دارکردن داده‌های بدون برچسب یکی از مسائل حائز اهمیت در یادگیری ماشین به شمار می‌رود [۷]. یادگیری نیمه‌نظارتی از روش‌های یادگیری ماشین است و در آن می‌توان از داده‌های بدون برچسب و برچسب‌دار به صورت هم‌زمان برای بهبود دقت یادگیری استفاده کرد. از این رو در این نوع یادگیری لزومی به برچسب‌داربودن تمام داده‌های جمع‌آوری‌شده برای آموزش طبقه‌بند نیست و بسیار منطبق با نیازهای دنیای کنونی است.

رویکردهای متنوعی از یادگیری نیمه‌نظارتی در سراسر دنیا در حال مطالعه و بررسی است که اغلب آنها طبقه‌بندی را بر اساس فرضیات متفاوتی که وابسته به ارتباط میان توزیع داده‌های برچسب‌دار و بدون برچسب می‌باشند، انجام می‌دهند. برای کنارگذاشتن این فرضیات، تکنیک‌های خودبرچسب‌زن[1] مطرح شدند که شامل دو روش باهم‌آموز[2] و خودآموز[3] می‌باشند [۸].

باهم‌آموز تکنیکی است که فضای ویژگی را به صورت دو دیدگاه مستقل شرطی در نظر می‌گیرد که هر دیدگاه قادر به آموزش یک طبقه‌بند است. همچنین دو طبقه‌بند قادر به آموزش یکدیگر هستند تا بتوانند کلاس‌ها را به طور کامل پیش‌بینی کنند [۴]. روش خودآموز یک متد مبتنی بر غلاف[4] است که از پیش‌بینی‌های یادگیر پایه برای بهبود کارایی خودش استفاده می‌کند. در هر مرحله، مجموعه داده‌های برچسب‌دار با نمونه‌های بدون برچسبی که انتخاب شده‌اند تا بر اساس پیش‌بینی طبقه‌بند پایه به آنها برچسب اختصاص داده شود به روز رسانی می‌شود [۹]. با این حال عملکرد این الگوریتم تا حد زیادی بستگی به نمونه‌های بدون برچسب انتخاب‌شده دارد. پیش‌بینی درست برچسب برای داده‌های بدون برچسب، نقش قابل توجهی در بهبود عملکرد مدل ایفا می‌کند. اگر نمونه‌های بدون برچسبی که به اشتباه طبقه‌بندی شده‌اند به مجموعه آموزشی اضافه گردند، منجر به انتشار خطا در روند آموزش می‌شوند. به طور کلی برای طبقه‌بندی نیمه‌نظارتی با استفاده از چارچوب خودآموز، دو چالش مهم وجود دارد: ۱) انتخاب مجموعه کاندیدای مناسب از داده‌های

این مقاله در تاریخ ۱۷ مهر ماه ۱۴۰۰ دریافت و در تاریخ ۱۷ بهمن ماه ۱۴۰۰ بازنگری شد.

منا عمادی، گروه مهندسی کامپیوتر، واحد بروجرد، دانشگاه آزاد اسلامی، بروجرد، ایران، (email: emadi.mona@pnu.ac.ir).

جعفر تنها (نویسنده مسئول)، گروه مهندسی برق و الکترونیک، دانشگاه تبریز، تبریز، ایران، (email: tanha@tabrizu.ac.ir).

محمدابراهیم شیری، گروه علوم کامپیوتر، دانشگاه امیرکبیر، تهران، ایران، (email: shiri@aut.ac.ir).

مهدی حسین‌زاده اقدم، گروه مهندسی کامپیوتر، دانشگاه بناب، بناب، ایران، (email: mhaghdam@ubonab.ac.ir).

1. Self-Labeled
2. Co-Training
3. Self-Training
4. Wrapper

بدون برچسب در هر دور از فرایند آموزش می‌باشد. این مجموعه نقاط را نقاط داده‌ای دارای اطلاعات[1] می‌نامیم. ۲) پیش‌بینی برچسب‌های درست برای زیرمجموعه‌های انتخاب‌شده است. مطالعات اخیر در این زمینه تمایل به انتخاب نمونه‌هایی را دارند که در هر دور از نظر تخمین برچسب توسط طبقه‌بند پایه، بالاترین قابلیت اطمینان را دارند [۱]، [۹] و [۱۰]. اینها داده‌هایی هستند که معمولاً از مرز تصمیم دور می‌باشند. از این رو، این الگوریتم‌ها نمی‌توانند به طور قابل توجهی اطلاعات را از داده‌های بدون برچسب استخراج نمایند و مرز تصمیمی که نهایتاً به دست خواهد آمد بسیار نزدیک به مرز تصمیم طبقه‌بند اولیه خواهد بود [۱۱].

مسئله دیگر، در ارتباط با تخمین برچسب طبقه‌بندهای پایه است که در اکثر مواقع ممکن است تخمینی با احتمال مطمئن به پیش‌بینی‌های آنها اختصاص داده نشود [۹]. در [۱۰]، زیرمجموعه‌ای از داده‌های بدون برچسب بر مبنای فاصله آنها از مرز تصمیم انتخاب شده است. اگرچه نقاط داده‌ای انتخاب‌شده با توجه به معیار برآورد احتمال طبقه‌بند پایه استفاده‌شده قابلیت اطمینان بالایی دارند، اما آنها نقاط داده‌ای دارای اطلاعات یا به بیان دیگر نقاط آموزنده‌ای نیستند که بتوانند قادر به بهبود مرز تصمیم باشند؛ زیرا تأثیر کمی بر روی موقعیت ابرصفحه[2] می‌گذارند. علاوه بر این، اضافه‌کردن همه نقاط بدون برچسب به مجموعه برچسب‌دار زمان‌بر است و ممکن است مرز تصمیم را تغییر ندهد.

برای غلبه بر مشکلات ذکرشده در بالا، این مقاله یک الگوریتم خودآموز نیمه‌نظارتی جدید را بر مبنای الگوریتم ساخت همسایگی مطرح می‌کند. روش پیشنهادی، زیرمجموعه‌ای از نقاط بدون برچسب نزدیک به مرز تصمیم را به عنوان نقاط دارای اطلاعات در طی فرایند آموزش انتخاب می‌کند. این انتخاب اگرچه برای رسیدن به مرز تصمیمی بهینه، واقع‌بینانه است اما احتمال این را دارد که نقاط داده‌ای به اشتباه برچسب زده شوند. برای حل این مشکل و برچسب‌گذاری صحیح نمونه‌های بدون برچسب، یک الگوریتم مبتنی بر ساخت همسایگی ارائه شده است. این الگوریتم با استفاده از DBSCAN همسایه نمونه‌ها را پیدا می‌کند. الگوریتم DBSCAN در دسته روش‌های خوشه‌بندی مبتنی بر چگالی قرار دارد. این روش خوشه‌بندی بر این فرض استوار است که خوشه‌ها شامل ناحیه‌ای از فضای داده با تراکم بالا می‌باشند که از نواحی با چگالی کمتر جدا شده‌اند. مزیت روش پیشنهادی این است که به شکل داده‌ها حساس نیست و بنابراین قادر به شناسایی خوشه‌هایی با اشکال مختلف غیر کروی می‌باشد. تعداد خوشه‌ها به طور هم‌زمان و خودکار تعیین می‌شود و همچنین در شناسایی نویز کارآمد است. بخش‌های اصلی این مطالعه شامل موارد زیر است:

- این مقاله تأثیر نقاط نزدیک به مرز تصمیم را برای بهبود کارایی طبقه‌بندی بررسی می‌کند.
- معیاری جدید برای اندازه‌گیری شباهت بین نقاط داده برچسب‌دار و بدون برچسب معرفی کرده است.
- روش پیشنهادی یک روش مبتنی بر توافق بین پیش‌بینی‌های طبقه‌بند پایه و الگوریتم مبتنی بر ساخت همسایگی به منظور اختصاص‌دادن برچسب به داده‌های بدون برچسب می‌باشد.
- راهکاری کارآمد جهت اطمینان از درستی برچسب پیش‌بینی شده ارائه داده است.

نتایج آزمایشگاهی بر روی بعضی از مجموعه‌های داده‌ها UCI [۱۲] نشان می‌دهند که الگوریتم پیشنهادی بهتر از دیگر الگوریتم‌های مطرح‌شده در این حوزه است.

باقی مقاله به شرح زیر سازمان‌دهی می‌شود: بخش ۲ کارهای مرتبط با یادگیری نیمه‌نظارتی را مورد مطالعه قرار می‌دهد. بخش ۳ هدف تحقیق و الگوریتم پیشنهادی را بیان می‌کند. بخش ۴ آزمایش‌ها و نتایج آنها را بر روی برخی از مجموعه داده‌ها ارائه می‌دهد و آنها را با بعضی از الگوریتم‌های مطرح در این حوزه مقایسه می‌کند. در نهایت بخش ۵ به نتیجه‌گیری می‌پردازد.

۲- تحقیق‌های پیشین انجام‌شده

یک روش برای مقابله با مشکلات یادگیری نیمه‌نظارتی، الگوریتم‌های خودبرچسب‌گذار هستند که از طبقه‌بند نظارت‌شده برای نشان‌دادن نمونه‌های برچسب‌دار با کلاس ناشناخته و بدون هیچ فرضیه خاصی درباره داده ورودی، استفاده می‌کنند [۱۳]. تکنیک‌های خودبرچسب‌گذاری شامل دو روش شناخته‌شده به نام‌های باهم‌آموز و خودآموز می‌باشند. آموزش باهم‌آموز، فضای ویژه را در دو دیدگاه مستقل از نظر شرطی بررسی می‌کند [۱۴]. هر دیدگاه قادر است یک طبقه‌بند را آموزش دهد و سپس به دیگران آموزش دهد که کلاس‌ها را کاملاً پیش‌بینی کنند [۱۵] و [۱۶]. علاوه بر این، رویکردهای پیشرفته‌ای برای آموزش هم‌زمان چندین یادگیری وجود دارد که نیازی به تقسیم ویژگی‌ها به صورت آشکار و یا روش آموزش تکرار متقابل ندارند [۱۳] و [۱۷]. همان طور که از نام خودآموز مشخص است، خودآموزی تلاش می‌کند تا به صورت مکرر داده‌های برچسب‌دار مجموعه آموزشی را بیشتر و بیشتر کند [۱۸]. برای شروع، یک طبقه‌بند با داده‌های دارای برچسب اولیه آموزش داده می‌شود و سپس داده‌های بدون برچسب با بیشترین اطلاعات و اطمینان انتخاب می‌گردند. آن گاه به تدریج داده‌های کاندیدا به همراه برچسب‌های پیش‌بینی‌شده آنها به مجموعه آموزشی اضافه می‌شوند و این روش تا رسیدن به همگرایی تکرار می‌گردد. خودآموز با موفقیت در بسیاری از داده‌های واقعی اعمال شده است [۱۹]. محدودیت این روش در تعداد داده‌های برچسب‌دار و توزیع آنهاست، زیرا اگر این داده‌ها نتوانند ساختار فضای داده را به خوبی نشان دهند، فرایند آموزش به فضای واقعی داده‌ها نزدیک نشده و طبقه‌بند به درستی آموزش نمی‌بیند. روش‌هایی برای بهبود روش خودآموز ارائه شده‌اند که از جمله آنها، روش کمک یادگیری[3] است که سعی بر آموزش یک طبقه‌بند متمایزکننده[4] استفاده از مدل تولیدی دارد [۶]. ولی این روش نتوانست به طور ریشه‌ای محدودیت روش خودآموز را برطرف کند، زیرا مدل تولیدی تنها با داده‌های برچسب‌دار آموزش می‌بیند. سپس الگوریتم نیمه‌نظارتی فازی c-means ارائه شد، از هر دو نوع داده برچسب‌دار و بدون برچسب برای نشان دادن ساختار واقعی داده استفاده می‌کرد، اما محدودیت‌های این الگوریتم باعث ناکارآمدی آن در توزیع‌های غیر کروی بود[5] [۱۴].

گروهی از الگوریتم‌های یادگیری نیمه‌نظارتی، مبتنی بر حاشیه‌ها می‌باشند که به دلیل داشتن پایه ریاضی از اهمیت زیادی برخوردار هستند (مانند بوستینگ، ماشین بردار پشتیبان و TSVM[6]). الگوریتم‌های ماشین

1. Informative Data Points
2. Hyperplane
3. Help-Training
4. Discriminative
5. Non-Spherical
6. Transductive Support Vector Machine

جدول ۱: مزایا و معایب سه روش ساخت همسایگی.

روش‌های ساخت همسایگی	مزایا	معایب
k-NN	– سادگی روش – ناپارامتریک است. – رویکردی شهودی دارد. – در تشخیص نقاط پرت قوی است. – در برخورد با ویژگی‌های عددی و پیوسته خوب است. – خوشه‌هایی از اشکال دلخواه را کشف می‌کند.	– تعیین همسایگی فقط براساس فاصله است. – حساسیت به ویژگی‌های نامربوط دارد. – در مدیریت داده‌ها با انواع مختلط بسیار مشکل دارد. – فقدان مبانی نظری – بر روی مقادیر عددی/ دودویی نامشخص است. – نمی‌تواند خوشه‌هایی با چگالی متفاوت را به طور مؤثر پیدا کند.
DBSCAN	– پیچیدگی متوسط دارد. – پردازش آن سریع است. – توانایی تشخیص نقاط پرت – نیازی به تعریف تعداد خوشه‌ها ندارد.	– نمی‌تواند با چندین نقطه پرت کار کند. – برای مجموعه داده‌های بزرگ، هزینه محاسباتی بالایی دارد. – عدم قطعیت در تنظیم پارامترهای ورودی – برای داده‌های با ابعاد بالا مناسب نیست. – هزینه محاسباتی بالا
DPC	– عدم وجود فاز تکرار – مراکز از ویژگی‌های اساسی نقاط داده استخراج می‌شوند.	– فقط برای مجموعه داده‌های کوچک مناسب است. – نیاز به تعریف پارامتر دارد. – تعداد خوشه‌های مناسب به دست آمده از گراف تصمیم ممکن است برابر نباشند و با تعداد خوشه‌های ایده‌آل قابل مقایسه نباشند.

بردار پشتیبان در بسیاری از برنامه‌های کاربردی واقعی موفق به کسب موفقیت‌هایی شده‌اند [۲۰]. در روش پیشنهادی، طبقه‌بند پایه مورد استفاده ماشین بردار پشتیبان می‌باشد.

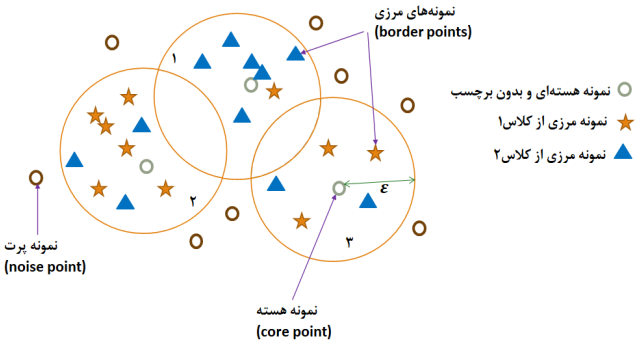
الگوریتم‌های خوشه‌بندی مبتنی بر چگالی، یکی از روش‌های اصلی در داده‌کاوی هستند که نسبت به روش‌های قبلی، پایه ریاضی قوی‌تری دارند [۲۱]. الگوریتم DBSCAN پایه روش‌های خوشه‌بندی مبتنی بر چگالی است. در این الگوریتم برای تعریف اپسیلون همسایگی، شعاع اپسیلونی (ε) با دایره تعیین می‌شود [۲۲]. در دایره‌ای به شعاع اپسیلون، اگر نقاط درون دایره در حداقل فاصله نقطه مورد نظر قرار گرفته باشند، آن گاه به آن نقاط، نقاط همسایگی نمونه بدون برچسب گفته می‌شود. اما اگر نقاط، خارج از آستانه مورد نظر باشد به آن نقاط، نقاط پرت یا حاشیه‌ای گفته می‌شود [۲۳].

یکی دیگر از الگوریتم‌های مبتنی بر چگالی، الگوریتم DPC می‌باشد که در مجله معتبر علوم آمریکا به چاپ رسیده است. در این الگوریتم برای تشخیص خوشه‌های غیر کروی و یافتن تعداد خوشه‌ها به طور خودکار برای تمام نقاط داده، دو کمیت چگالی محلی (ρ_i) و دلتا (δ_i) به کمک ماتریس فاصله محاسبه می‌شوند [۱۵]. پس از محاسبه این دو کمیت، نقاط مطمئن با استفاده از گراف تصمیم به دست می‌آیند. نقاط مطمئن، نقاطی در گراف تصمیم هستند که چگالی بالا و فاصله بیشتری از نقاط پرتراکم دیگر دارند. این الگوریتم وابسته به پارامتری به نام فاصله قطع[1] است که برای تعیین درصد همسایگی در محاسبه چگالی محلی نقاط به کار می‌رود و باعث می‌شود که دقت الگوریتم همواره به یک پارامتر اولیه وابسته باشد. مرجع [۳] از این الگوریتم برای مشخص‌کردن ساختار فضای داده استفاده کرده است، به طوری که پس از محاسبه دو کمیت چگالی محلی و دلتا برای تمامی نقاط، ساختار واقعی داده با اشاره‌کردن هر نقطه به نزدیک‌ترین نقطه‌ای که چگالی آن بیشتر از خودش می‌باشد به دست آمده است. سپس از روش نیمه‌نظارتی خودآموز برای برچسب‌زنی به داده‌های بدون برچسب استفاده کرده است، به طوری که ابتدا طبقه‌بند با مجموعه داده‌های برچسب‌دار آموزش دیده و سپس برای نقاط بدون برچسبی که مابعد داده‌های پرچگال برچسب‌دار هستند، برچسبی پیش‌بینی کرده و آنها را به مجموعه آموزشی برچسب‌دار اضافه و از مجموعه داده‌های بدون برچسب کم کرده و بار دیگر با مجموعه داده‌های برچسب‌دار جدید آموزش دیده است. سپس همین کار را برای نقاط بدون برچسب که ماقبل داده‌های پرچگالش هستند، انجام داده و بدین ترتیب تمام نقاط بدون برچسب را برچسب زده است. در [۲۴] الگوریتمی برای بهبود عملکرد DPC ارائه گردیده که از هسته فازی برای انتخاب خوشه مناسب استفاده شده است. این الگوریتم مبتنی بر گراف KNN می‌باشد و راهکاری را برای برچسب‌گذاری داده‌های مرزی و همچنین داده‌هایی با اشکال و چگالی‌های متفاوت ارائه می‌دهد. در [۲۵] یک روش خوشه‌بندی با استفاده از DPC ارائه شده است. در این روش با ایده K نزدیک‌ترین همسایه، پارامتر قطع و چگالی محلی هر نمونه محاسبه شده است و برچسب‌گذاری با استفاده از انتشار برچسب انجام می‌شود.

یکی دیگر از الگوریتم‌هایی که برای خوشه‌بندی و تعریف همسایگی داده‌ها استفاده می‌شود، k نزدیک‌ترین همسایه (k-NN) است که از فاصله اقلیدسی استفاده می‌کند. این الگوریتم با مجموعه داده‌هایی که مشابه یکدیگر هستند، شروع می‌شود و سپس خوشه‌ها را مشخص می‌کند. k نزدیک‌ترین همسایه، الگوریتمی بسیار ساده است و در تعیین همسایگی مفید می‌باشد. تنها معیار استفاده‌شده برای تعیین همسایگی در این الگوریتم، فاصله و مکان نقاط است [۲۶]. یکی از الگوریتم‌های مورد مقایسه در این مطالعه SSApollo می‌باشد [۲۷] که برای حل مسائل طبقه‌بندی نیمه‌نظارتی از DPC و دایره آپولونیوس استفاده کرده است.

اکثر الگوریتم‌های طبقه‌بندی خودآموز جهت برچسب‌زدن به نمونه‌های بدون برچسب از پیش‌بینی طبقه‌بند پایه استفاده نموده و همچنین اقدام به برچسب‌زدن به تمامی مجموعه نقاط بدون برچسب کرده‌اند که این عمل نه‌تنها زمان‌بر است، بلکه در بعضی مواقع منجر به کاهش دقت نیز می‌شود. در الگوریتم پیشنهادی، علاوه بر انتخاب مجموعه داده بدون برچسب به عنوان کاندیدایی برای برچسب‌زدن، از الگوریتم ساخت همسایگی استفاده شده تا برچسب مطمئن را به نمونه بدون برچسب اختصاص دهد. در الگوریتم پیشنهادی برای پیداکردن نقاط همسایه نمونه بدون برچسب در فرایند خودآموز از الگوریتم DBSCAN استفاده شده است. در جدول ۱ مقایسه تحلیلی سه الگوریتم ساخت همسایگی آمده است [۲۸].

1. Cut of Distance



شکل ۱: نحوه تخصیص برچسب به داده‌های بدون برچسب با استفاده از الگوریتم همسایگی DBSCAN.

۳- یادگیری نیمه‌نظارتی

در این بخش، ابتدا مسائل طبقه‌بندی نیمه‌نظارتی و اهداف تحقیق را بیان و سپس الگوریتم پیشنهادی را مطرح می‌کنیم.

۳-۱ اهداف تحقیق و تعریف مسئله

در الگوریتم نیمه‌نظارتی هم داده‌های برچسب‌دار $x_l = (x_1, x_2, \ldots, x_l)$ با برچسب‌های $\{1, 2, \ldots, k\}$ و هم داده‌های بدون برچسب $x_u = (x_{l+1}, x_{l+2}, \ldots, x_{l+u})$ با برچسب‌های ناشناخته وجود دارند، به طوری که تعداد داده‌های برچسب‌دار خیلی کمتر از داده‌های بدون برچسب می‌باشد. هر دو گروه از داده‌ها (برچسب‌دار و بدون برچسب) در توزیع داده‌هایی یکسان به طور مستقل هستند. همان طور که قبلاً بیان شد، در چارچوب خودآموز دو چالش اصلی وجود دارد: ۱) انتخاب مجموعه مناسب از نمونه‌های بدون برچسب در هر تکرار فرایند خودآموز و ۲) پیش‌بینی درست برچسب نمونه‌های موجود در زیرمجموعه‌های انتخاب‌شده. برای غلبه بر این مسائل، بیشتر الگوریتم‌های یادگیری نیمه‌نظارتی، تخمین‌های احتمالی طبقه‌بندهای پایه را به منظور اختصاص برچسب به داده‌های بدون برچسب به کار می‌گیرند. این الگوریتم‌ها از همین ابزار به عنوان معیار انتخاب استفاده می‌کنند [۱]، [۹] و [۲۹] تا [۳۱]. استفاده از چنین معیاری ممکن است منجر به انتخاب نمونه‌های مطمئنی شود که به اشتباه طبقه‌بندی شده‌اند و نتیجه آن انتشار خطا و همچنین عدم بهبود مرز تصمیم می‌باشد [۲۷] و [۳۲]. هدف ما در این مطالعه، تمرکز بر روی نقاطی است که نزدیک به مرز تصمیم هستند. این نمونه‌های بدون برچسب به طور کلی از نظر پیش‌بینی احتمال طبقه‌بند پایه، اطمینان بالایی ندارند. بعد از انتخاب این نمونه‌ها باید به دنبال ابزاری بود تا برچسب درستی به این نمونه‌های دارای اطلاعات اختصاص داد و سپس آنها را به مجموعه داده‌های برچسب‌دار اصلی در هر تکرار فرایند آموزش اضافه کرد. بر اساس این ایده در الگوریتم پیشنهادی از الگوریتم‌های نیمه‌نظارتی مبتنی بر حاشیه استفاده می‌کنیم.

۳-۲ الگوریتم پیشنهادی

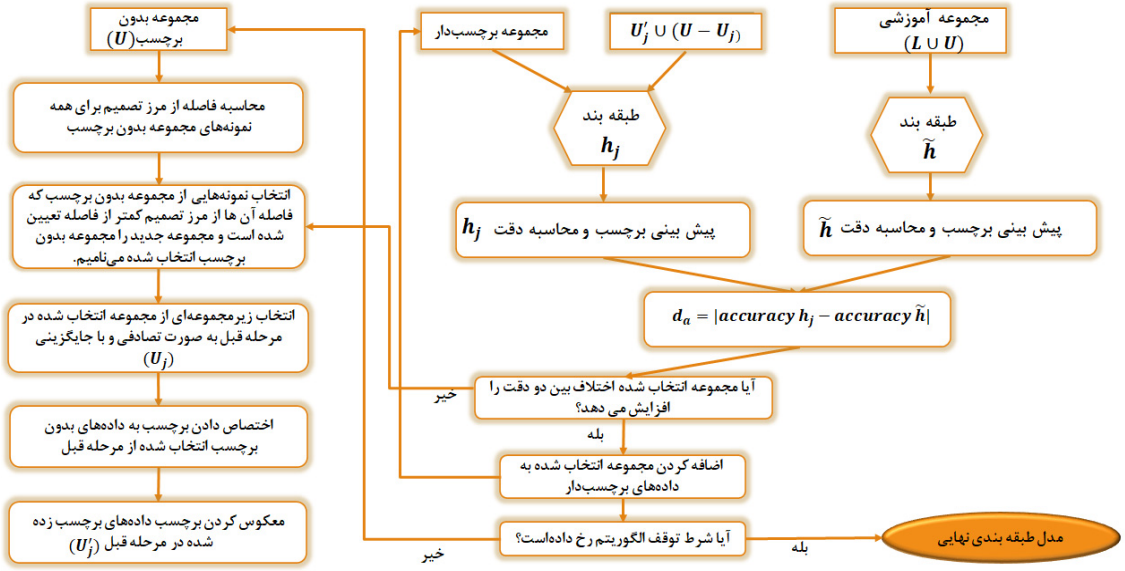
روش ساده اپسیلون همسایگی (DBSCAN)، همسایگی را بر اساس شعاعی کوچک و بر اساس پارامتر فاصله اپسیلون تعیین می‌نماید و در این ناحیه از شعاع همسایگی، داده‌ها برچسب‌گذاری می‌شوند [۲۳] و [۳۳]. در دایره‌ای با شعاع اپسیلون، اگر نقاط داخل دایره با فاصله حداقل MINPTS از نقطه مورد نظر تعریف شوند، به عنوان نقاط همسایه اصلی برای آن نقطه انتخاب می‌گردند. اما اگر نقاط، خارج از نقاط تعریف‌شده قرار داشته باشند، به عنوان نقاط مرزی یا نویز نشان داده می‌شوند. منطقه تصمیم‌گیری و تراکم در بین این نقاط در فضای همسایگی، آنها را در برچسب‌گذاری مرتبط می‌کند [۳۴].

در الگوریتم پیشنهادی قرار است که برچسب‌گذاری داده‌های بدون برچسب به روشی انجام گردد که باعث افزایش دقت طبقه‌بند پایه شود. اگر تمامی داده‌های بدون برچسب را اضافه کنیم، علاوه بر این که سایز فضای فرضیه خیلی بزرگ می‌شود، باعث ایجاد نویز نیز می‌گردد. بنابراین در این الگوریتم، داده‌های بدون برچسبی انتخاب شده‌اند که حاوی اطلاعات مهمی باشند و این داده‌ها از حد مشخصی به حاشیه نزدیک‌تر هستند. برای تشخیص درستی یا نادرستی برچسب پیش‌بینی شده و همچنین تأثیرگذاری مجموعه بدون برچسب انتخاب‌شده به روش زیر عمل می‌شود:

ابتدا با استفاده از الگوریتم ماشین بردار پشتیبان استاندارد و فقط با مجموعه آموزشی اولیه، برچسب‌گذاری نمونه‌های بدون برچسب انجام می‌گردد. سپس نمونه‌های بدون برچسبی که در مرحله قبل برچسب خورده‌اند به مجموعه آموزشی اضافه می‌شوند $(L \cup U)$ و مجدداً با الگوریتم ماشین بردار پشتیبان عمل آموزش تکرار می‌گردد و دقت، محاسبه می‌شود. اضافه‌کردن همه داده‌های بدون برچسب کار درستی نیست. در الگوریتم پیشنهادی، نمونه‌های بدون برچسبی اضافه می‌شوند که از فاصله مشخصی به حاشیه نزدیک‌تر هستند و بنابراین بعد از محاسبه فاصله داده‌ها از مرز تصمیم، آنهایی را که فاصله‌شان از مقدار معینی کمتر باشد در مجموعه‌ای جداگانه ریخته و در مرحله بعدی، قبل از اضافه‌کردن آنها به مجموعه آموزشی، برچسب آنها را پیش‌بینی می‌نماید. برچسب‌گذاری نمونه‌های بدون برچسب بر اساس همسایگان آنها با الگوریتم اپسیلون همسایگی انجام می‌شود. نحوه اختصاص برچسب به نمونه‌های بدون برچسب با الگوریتم DBSCAN در شکل ۱ نشان داده شده است.

سه گروه همسایگی را در شکل ۱ می‌توان مشاهده کرد. در الگوریتم پیشنهادی با استفاده از الگوریتم همسایگی DBSCAN، نمونه‌های همسایه داده بدون برچسب مورد نظر را که از بین مجموعه برچسب‌دار انتخاب شده‌اند پیدا کرده و سپس برچسب کلاس اکثریت را به نمونه بدون برچسب اختصاص می‌دهد. به عنوان مثال در شکل ۱ به نمونه بدون برچسبی که در گروه همسایگی ۱ قرار دارد، برچسب کلاس ۲ اختصاص می‌یابد و به نمونه‌های بدون برچسب گروه‌های همسایگی ۲ و ۳، برچسب کلاس ۱ اختصاص می‌یابد.

الگوریتم پیشنهادی در فرایند خودآموزی، تمام داده‌های بدون برچسب را برچسب‌گذاری نمی‌کند و فقط نمونه‌هایی که از مقدار تعیین‌شده‌ای به مرز تصمیم نزدیک‌تر هستند، انتخاب می‌گردند. این داده‌ها به دلیل نزدیکی به مرز تصمیم، تأثیرگذاری فراوانی در دستیابی ابرصفحه بهینه دارند و واقعاً آموزنده هستند. تأثیر مثبت اضافه‌شدن این نمونه‌ها در صورتی است که برچسب آنها به درستی پیش‌بینی شود، زیرا این داده‌ها به دلیل نزدیکی به مرز تصمیم، عدم قطعیت دارند و احتمال این که برچسب تخمین زده شده برای آنها اشتباه باشد، زیاد است. در الگوریتم پیشنهادی در هر بار تکرار از میان داده‌های بدون برچسب نزدیک به مرز، N زیرمجموعه به صورت تصادفی انتخاب می‌شوند. ؟؟؟ مشکل عدم قطعیت برچسب تخمین زده شده توسط طبقه‌بند پایه را با استفاده از الگوریتم ساخت همسایگی DBSCAN و همچنین با روش ردکردن معکوس همان زیرمجموعه حل کرده است. طبقه‌بند استفاده‌شده در این الگوریتم ماشین بردار پشتیبان می‌باشد.



شکل ۲: نمایی از الگوریتم پیشنهادی.

روش کار چنین است که ابتدا طبقه‌بند $\tilde{h}$ را بر روی مجموعه $L \cup U$ آموزش می‌دهد و سپس برای هر یک از N زیرمجموعه از داده‌های بدون برچسبی که انتخاب می‌شوند $U_j (J = 1, 2, \ldots, N)$، زیرمجموعه معکوس آن (U'_j) ایجاد می‌گردد. U'_j از معکوس‌کردن برچسب نقاط داده‌ای در U_j به دست می‌آید. برچسب داده‌های U_j با استفاده از همسایگانی که عضو مجموعه آموزشی برچسب‌دار می‌باشند، بر اساس الگوریتم ساخت همسایگی DBSCAN پیش‌بینی می‌شود. نحوه تشخیص برچسب داده‌ها توسط این الگوریتم در شکل ۱ نشان داده شده است. بعد از معکوس‌کردن برچسب داده‌های پیش‌بینی شده، طبقه‌بند h_j را بر روی مجموعه $L \cup u'_j \cup (u - u_j)$ آموزش می‌دهد. در نهایت زیرمجموعه‌ای انتخاب می‌شود که باعث ایجاد ماکسیمم اختلاف دقت برای دو طبقه‌بند h_j و $\tilde{h}$ گردد. در واقع زیرمجموعه‌ای که با معکوس‌شدن برچسب‌های آن، دقت طبقه‌بند به شدت کاهش یابد انتخاب می‌شود و به مجموعه برچسب‌دار اضافه می‌گردد. با این روش نقاطی برچسب‌گذاری شده‌اند که بالاترین اطمینان را در پیش‌بینی برچسب داشته‌اند. شکل ۲ نمایی از الگوریتم پیشنهادی را نشان می‌دهد.

در این الگوریتم در ابتدای کار به تمام داده‌های بدون برچسب، وزن یکسانی اختصاص داده می‌شود. سپس در هر دور از الگوریتم خودآموز، داده‌های بدون برچسبی که برای اضافه‌شدن به مجموعه برچسب‌دار انتخاب شده‌اند، وزنشان کاهش می‌یابد. با کاهش وزن، شانس انتخاب مجدد آنها در دورهای بعدی اجرای الگوریتم کاهش می‌یابد؛ زیرا فرایند انتخاب داده‌های بدون برچسب به صورت تصادفی و با جایگشت است. در پایان هر دور، شرط $\overline{w} > c \times \alpha$ بررسی می‌شود و اگر شرط برقرار نباشد الگوریتم پایان می‌یابد. c یک ثابت مثبت و دستگیره‌ای است که زمان خاتمه الگوریتم را مشخص می‌کند. هرچه مقدار c بزرگ‌تر باشد، الگوریتم زودتر خاتمه می‌یابد. α نرخ کاهش وزن نمونه‌های بدون برچسب را نشان می‌دهد. یکی دیگر از ثابت‌هایی که در الگوریتم استفاده شده است، δ می‌باشد که مقدار آن با میانگین فاصله داده‌های بدون برچسب تا مرز تصمیم تنظیم شده است. الگوریتم ۱ چهارچوب را با جزئیات بیشتری نشان داده است.

الگوریتم ۱: الگوریتم پیشنهادی خودآموز مبتنی بر ساخت همسایگی

داده‌های ورودی: مجموعه برچسب‌دار (L)، مجموعه بدون برچسب (U)، نرخ کاهش وزن (α) و مقدار مثبت ثابت (C) که این مقدار کنترل می‌کند الگوریتم چه زمانی خاتمه یابد.

داده‌های خروجی: دقت مدل طبقه‌بندی

۱) فاصله داده‌های بدون برچسب را از مرز تصمیم پیدا کن (d).

۲) میانگین فاصله را پیدا کن و حد فاصله تعیین‌شده برای انتخاب داده‌های بدون برچسب انتخابی را برابر آن قرار بده $(\delta = \bar{d})$.

۳) وزن اولیه تمامی نمونه‌های بدون برچسب را برابر یک قرار بده $(\forall x_i \in U : w[x_i] = 1)$.

۴) میانگین وزن همه نمونه‌های بدون برچسب را با $\overline{w}$ نشان بده.

۵) طبقه‌بند $\tilde{h}$ را بر روی مجموعه‌ای که از اجتماع داده‌های برچسب‌دار و بدون برچسب ایجاد شده است آموزش داده و دقت را محاسبه کن.

۶) تا زمانی که مقدار $\overline{w} > C \times \alpha$ برقرار است، مراحل زیر را تکرار کن:

۷) برای تمامی نمونه‌های بدون برچسب مراحل زیر را انجام بده:

۸) داده‌های بدون برچسبی را که فاصله آنها از مرز تصمیم، کمتر یا مساوی δ است در مجموعه‌ای به نام U_δ قرار بده. این مجموعه در ابتدا تهی است.

۹) به طور تصادفی N زیرمجموعه از U_δ انتخاب کن و مراحل ۱۰ تا ۱۴ را انجام بده.

۱۰) وزن نمونه انتخاب‌شده را کاهش بده تا شانس انتخاب آن در دورهای بعدی فرایند خودآموز کاهش یابد.

۱۱) با استفاده از الگوریتم DBSCAN نزدیک‌ترین همسایه‌های نمونه بدون برچسب را انتخاب کن. نمونه‌های همسایه انتخاب‌شده عضو مجموعه برچسب‌دار می‌باشند.

۱۲) برچسب کلاس اکثریت همسایگان را به نمونه بدون برچسب انتخاب‌شده اختصاص بده.

۱۳) طبقه‌بند h_j را روی $L \cup u'_j \cup (u - u_j)$ آموزش داده و دقت را محاسبه کن.

۱۴) مجموعه‌ای که بیشترین اختلاف را برای دو طبقه‌بند h_j و $\tilde{h}$ ایجاد می‌کنند، انتخاب کرده و به مجموعه داده‌های برچسب‌دار اضافه کن.

۱۵) مراحل ۶ تا ۱۴ را تکرار کن تا الگوریتم خودآموز پایان پذیرد.

۱۶) دقت مدل طبقه‌بند نهایی را محاسبه کن.

جدول ۲: مشخصات مجموعه داده‌ها.

نام مجموعه داده	تعداد نمونه	تعداد صفات	تعداد کلاس
Heart	۲۷۰	۱۳	۲
Blood	۷۴۸	۴	۲
Hepatitis	۱۵۵	۱۸	۲
Vote	۴۳۵	۱۶	۲
Mushroom	۸۱۲۴	۲۲	۲
Liver	۳۴۵	۶	۲
Breast	۶۹۹	۱۰	۲
Sonar	۲۰۸	۶۰	۲
German	۱۰۰۰	۲۴	۲
Colic	۳۰۰	۲۷	۲
Iris	۱۵۰	۴	۳
wine	۱۷۸	۱۳	۳

۳-۳ پیچیدگی محاسباتی الگوریتم پیشنهادی

پیچیدگی محاسباتی الگوریتم ساخت همسایگی $O(n^2)$ است که در آن، n برابر با سایز مجموعه داده می‌باشد. الگوریتم به ازای نمونه‌های عضو زیرمجموعه بدون برچسب، اعمال برچسب‌زدن و تخصیص وزن را انجام می‌دهد. از آنجایی که $U \subseteq n$ می‌باشد، پیچیدگی کلی الگوریتم $O(n^2)$ است.

٤- نتایج آزمایش‌ها

در این بخش، آزمایش‌هایی تنظیم شده که متد پیشنهادی را با چندین الگوریتم دیگر بررسی می‌کند. الگوریتم پیشنهادی می‌تواند برای هر نوعی از یادگیری مبتنی بر حاشیه استفاده گردد. در این مطالعه از یادگیر پایه ماشین بردار پشتیبان استفاده شده است. در آزمایش اول، کارایی ماشین بردار پشتیبان، فقط با استفاده از داده‌های برچسب‌دار گزارش شده است. در آزمایش دوم با ماشین بردار پشتیبان استاندارد که در غلاف الگوریتم خودآموز قرار گرفته است و همچنین روش خودآموز مبتنی بر فاصله که ST-DB[1] [۳۵] نام دارد، عمل مقایسه انجام شده است. همچنین الگوریتم پیشنهادی به ترتیب با الگوریتم‌های این حوزه مانند TSVM[2] [۳۶] و خودآموز استاندارد (ST) [۳۷] و SSApollo [۲۷] مقایسه شده است. در آزمایش‌ها، از مجموعه داده واقعی برای آزمایش عملکرد الگوریتم استفاده شده و جزئیات این مجموعه داده‌ها در جدول ۱ آمده است.

٤-۱ مجموعه داده‌ها

چندین مجموعه داده UCI در این آزمایش‌ها استفاده گردیده است. در جدول ۲ به طور خلاصه مشخصات ۱۲ مجموعه داده از مخزن داده UCI مورد استفاده در این آزمایش‌ها نشان داده شده است. ما مجموعه داده‌های واقعی را انتخاب کرده‌ایم تا در صورتی که در واقعیت، مجموعه داده‌ها از نظر پراکندگی و توزیع شبیه اینها بودند، الگوریتم بتواند برچسب‌گذاری نمونه‌ها را انجام دهد (مانند مجموعه داده‌های پزشکی و صنعتی). الگوریتم پیشنهادی برای مجموعه داده‌های باینری و چندکلاسه کارایی دارد.

1. Distance-Based Self-Training
2. Transductive SVM

٤-۲ تنظیمات آزمایشی

برای هر مجموعه داده، ۳۰٪ از داده‌ها به طور تصادفی برای مجموعه تست کنار گذاشته می‌شوند. در ابتدا داده‌های مجموعه آموزشی به ۹۰٪ داده‌های بدون برچسب و ۱۰٪ داده‌های برچسب‌دار تقسیم می‌شوند. کلاس‌ها برای همه مجموعه‌ها به نسبت برابر با مجموعه داده اصلی انتخاب می‌شوند و هر آزمایش با زیرمجموعه‌های متفاوتی از تست و آموزش، ۱۰ بار تکرار شده است. برای بررسی کارایی روش پیشنهادی، نرخ میانگین دقت (MAR) برای هر ۱۰ آزمایش تکرار شده که برای محاسبه آن از یک مجموعه تست متشکل از تعداد نمونه‌های x_i با ω شناخته‌شده برای مرحله آزمون استفاده می‌گردد. دو مقدار از نرخ دقت (AR) و میانگین نرخ دقت (MAR)[3] به ترتیب زیر محاسبه می‌شوند

$$AR = \frac{1}{t}\sum_{i=1}^{t}\psi(\omega, f(x_i)),\psi(\omega, f(x_i)) = \begin{cases} 1 & , \text{if } \omega = f(x_i) \\ 0 & , \text{else} \end{cases} \quad (1)$$

$$MAR = \frac{1}{n}\sum_{k=1}^{n} AR_k \quad (2)$$

$f(x_i)$ برچسب محاسبه‌شده برای x_i و n تعداد دفعات تکرار برای محاسبه AR است. MAR بیان‌کننده توانایی الگوریتم پیشنهادی است.

برای اجرای شبیه‌سازی‌ها از نرم‌افزار Matlab R۲۰۱۸b بر روی سیستمی با پردازنده Intel(R) Core(TM) i۷ با حافظه ۱۶ گیگابایت و سیستم عامل ۶۴bit windows۱۰ استفاده شده است.

٤-۳ بحث پیرامون نتایج

جداول ۳ و ۴ کارایی طبقه‌بندی الگوریتم پیشنهادی و سایر الگوریتم‌ها را در شرایطی که فقط ۱۰٪ و ۲۰٪ از داده‌های آموزشی برچسب‌دار هستند، نشان می‌دهد. در این جدول‌ها، ستون اول کارایی طبقه‌بند SVM نظارت‌شده را نشان می‌دهد. ستون دوم، سوم، چهارم و پنجم به ترتیب بیان‌کننده کارایی طبقه‌بندهای خودآموز استاندارد (ST)، خودآموز مبتنی بر فاصله (ST-DB)، TSVM و SSApollo می‌باشند. نهایتاً ستون آخر کارایی طبقه‌بند الگوریتم پیشنهادی است. برای همه متدها SVM به کار گرفته شده است. بهترین کارایی طبقه‌بندی برای هر مجموعه داده در جدول‌ها با فونت ضخیم نشان داده شده است.

جدول ۳، نتایج آزمایش‌ها را برای زمانی که فقط ۱۰٪ داده‌ها برچسب‌دار هستند، نشان می‌دهد. همان طور که مشاهده می‌شود الگوریتم پیشنهادی به طور چشم‌گیری دقت طبقه‌بندی ماشین بردار پشتیبان نظارت‌شده را برای اکثر مجموعه داده‌ها بهبود می‌دهد.

جدول ۴ نتایج آزمایش‌ها را برای زمانی که فقط ۲۰٪ داده‌ها برچسب‌دار هستند، نشان می‌دهد. همان طور که مشاهده می‌شود الگوریتم پیشنهادی به طور چشم‌گیری دقت طبقه‌بندی ماشین بردار پشتیبان نظارت‌شده را برای اکثر مجموعه داده‌ها بهبود می‌دهد.

با محاسبه نرخ میانگین دقت مشاهده می‌شود که در جدول ۳ الگوریتم پیشنهادی، کارایی الگوریتم طبقه‌بندی نظارت‌شده ماشین بردار پشتیبان را برای ۶ مجموعه داده از بین ۱۲ مجموعه داده به طور قابل توجهی بهبود می‌دهد. همچنین در جدول ۴ نیز مشاهده می‌شود که کارایی الگوریتم پیشنهادی در بین ۶ تا از ۱۲ مجموعه داده بهتر از سایر الگوریتم‌ها است.

3. Mean Accuracy Rate

جدول ۳: دقت طبقه‌بندی با ۱۰٪ داده‌های برچسب‌دار.

مجموعه داده	Supervised SVM	ST	ST-DB	TSVM	SSApollo	الگوریتم پیشنهادی
Heart	۶۲٫۹۶	۶۴٫۸۷	۶۳٫۷۰	۶۳٫۹۸	۷۰٫۳۱	**۷۰٫۳۷**
Blood	۴۹٫۳۲	۵۰٫۴۰	۵۰٫۷۸	۵۵٫۳۲	۶۲	**۶۲٫۰۵**
Hepatits	۴۴٫۶۸	۴۴٫۱۱	۴۴٫۷۸	۴۶٫۰۲	۴۶٫۶۷	**۴۶٫۸۱**
Vote	۸۲٫۲۶	۸۴٫۷۶	۸۲٫۱۹	۸۵٫۰۵	۸۶٫۸۸	**۸۷٫۷۹**
Mushroom	۵۶٫۹۶	۵۹٫۴۵	۵۹٫۲۳	۶۱٫۴۶	**۷۲**	۶۵٫۸۶
Liver	**۷۲٫۱۲**	۷۱٫۴۰	۷۲٫۱۰	۷۱٫۸۸	۶۱٫۹۰	۷۱٫۱۵
Breast	**۹۱٫۴۳**	۸۸٫۰۴	۸۹٫۳۳	۹۰	۸۸٫۵۷	۸۸٫۵۷
Sonar	۵۶٫۴۵	۵۵٫۲۳	۵۵٫۰۲	۵۷٫۰۱	**۶۶٫۷۰**	۵۳٫۲۳
German	۶۷٫۶۷	۶۷٫۸۹	۶۸	۶۷٫۵۰	۷۰٫۸۹	**۷۱٫۶۷**
Colic	۵۳٫۳۳	۵۵٫۴۵	۵۳٫۲۰	۵۴٫۴۳	**۶۰٫۸۹**	۶۰٫۰۰
iris	۹۱٫۴۰	۸۶	۹۲٫۸	۸۹٫۱۲	۹۳٫۷۶	**۹۳٫۸۰**
wine	۸۵٫۳۰	۸۹٫۸۱	۸۴٫۵۰	۸۹٫۹۱	**۹۰٫۴۰**	۹۰٫۰۹

جدول ۴: دقت طبقه‌بندی با ۲۰٪ داده‌های برچسب‌دار.

مجموعه داده	Supervised SVM	ST	ST-DB	TSVM	SSApollo	الگوریتم پیشنهادی
Heart	۸۰٫۲۵	۸۱٫۶۰	۸۰٫۷۰	۸۰٫۹۱	۸۲٫۹۷	**۸۲٫۲۷**
Blood	۵۴٫۵۲	۵۵٫۸۷	۵۶٫۴۰	۵۶٫۸۷	۶۱٫۴۵	**۶۱٫۶۱**
Hepatits	۵۵٫۳۲	۵۶٫۴۵	۵۶٫۲۳	۵۷٫۰۳	۵۹٫۴۳	**۵۹٫۵۷**
Vote	۸۳٫۲۱	۸۳٫۱۱	۸۴٫۵۷	۸۵٫۰۲	۸۴٫۹۸	**۸۵٫۵۰**
Mushroom	۵۶٫۷۹	۵۵٫۸۹	۵۶٫۵۶	۶۱٫۲۳	**۶۵٫۰۰**	۶۴٫۸۷
Liver	۶۸٫۲۷	۶۷٫۰۸	۶۸٫۲۰	**۶۸٫۹۰**	۶۸٫۰۰	۶۷٫۳۱
Breast	**۹۳٫۳۳**	۹۲٫۴۳	۹۲٫۵۵	۹۳٫۰۰	۹۰٫۵۹	۹۰٫۰۰
Sonar	۶۶٫۴۰	۶۶٫۰۵	۶۶٫۲۰	۶۶٫۵۰	**۶۶٫۳۲**	۶۰٫۰۰
German	۶۸٫۶۷	۶۸٫۷۰	۶۸٫۲۳	۶۹٫۰۳	۷۰٫۴۲	**۷۰٫۶۷**
Colic	۷۰٫۰۰	۷۱٫۰۲	۷۱٫۴۵	۷۰٫۹۸	**۷۴٫۳۴**	۷۳٫۳۳
Iris	۹۲٫۶۰	۸۸٫۰۰	۹۳٫۹۰	۹۱٫۰۰	۹۴٫۷۶	**۹۵٫۰۰**
wine	۸۸٫۴۸	۹۰٫۸۱	۸۹٫۵۰	۹۰٫۹۱	**۹۱٫۴۰**	۹۰٫۴۳

۴-۴ نسبت‌های مختلف اعداد بدون برچسب

به منظور ارزیابی تأثیر تعداد داده‌های برچسب‌دار بر دقت طبقه‌بندی، مجموعه‌ای از آزمایش‌ها با نرخ‌های مختلف داده‌های برچسب‌دار (از ۱۰٪ تا ۵۰٪) انجام شده و سپس کارایی الگوریتم پیشنهادی با الگوریتم ماشین بردار پشتیبان نظارت‌شده مقایسه گردیده است. مجموعه تست به صورت جداگانه برای ارزیابی کارایی به کار گرفته شده است. شکل ۳ کارایی الگوریتم‌ها را با نرخ‌های مختلف روی چهار مجموعه داده نشان می‌دهد. همان طور که در شکل آمده است، وقتی مقدار داده‌های برچسب‌دار در دسترس افزایش می‌یابد، اختلاف بین الگوریتم SVM و الگوریتم پیشنهادی کاهش می‌یابد. حتی در بعضی از مجموعه داده‌ها الگوریتم ماشین بردار پشتیبان نظارت‌شده بهتر از الگوریتم پیشنهادی عمل می‌کند. همان طور که نشان داده شده است، الگوریتم پیشنهادی به طور قابل توجهی کارایی SVM را در حالتی که داده‌های برچسب‌دار فقط ۱۰ یا ۲۰٪ است بهبود می‌دهد.

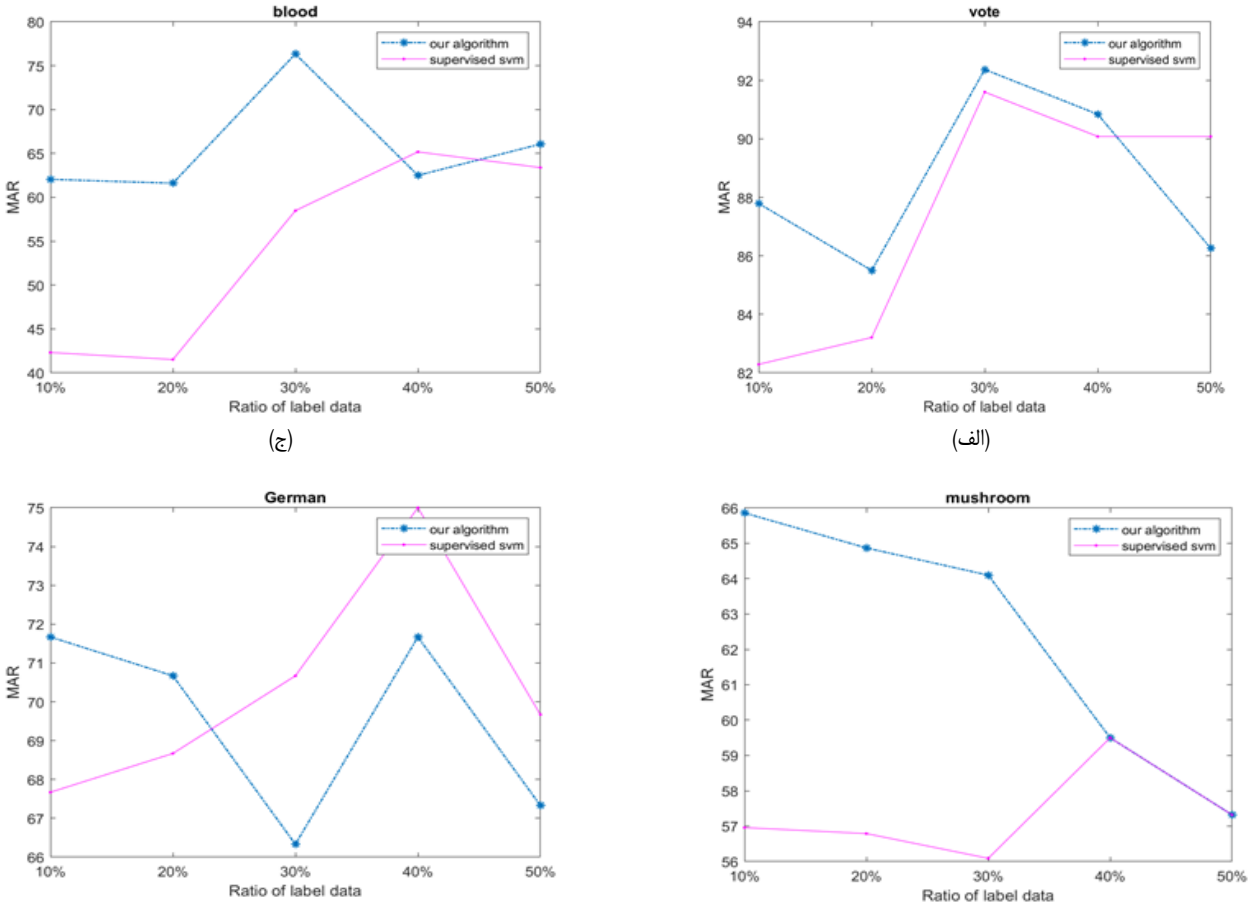
۴-۵ تأثیر انتخاب داده‌های نزدیک به مرز تصمیم

در بیشتر مجموعه داده‌ها، برچسب‌زدن تمامی داده‌های بدون برچسب نه‌تنها به بهبود کارایی طبقه‌بند پایه منجر نمی‌شود، بلکه می‌تواند دقت طبقه‌بندی را کاهش دهد که این کار همچنین زمان اجرای الگوریتم را افزایش می‌دهد. اگرچه نمونه‌های بدون برچسب دورتر از مرز تصمیم قابل اعتمادتر هستند، اما دارای اطلاعات نمی‌باشند. آنها نقش مهمی در بهبود مرز تصمیم ایفا نمی‌کنند و در نتیجه به جای اضافه‌کردن تمامی داده‌های بدون برچسب به مجموعه آموزشی، فقط مجموعه‌ای از داده‌های بدون برچسب را اضافه می‌کنیم که از مقدار معینی به مرز تصمیم نزدیک‌تر باشند. در آزمایش‌ها، میانگین فاصله داده‌های بدون برچسب تا مرز تصمیم به عنوان مقدار آستانه در نظر گرفته شده است. به عبارت دیگر، داده‌های بدون برچسبی که فاصله آنها تا مرز تصمیم کمتر یا مساوی میانگین فاصله باشند، به مجموعه آموزشی اضافه می‌گردند. چندین آزمایش برای نشان‌دادن این مسئله انجام شده است.

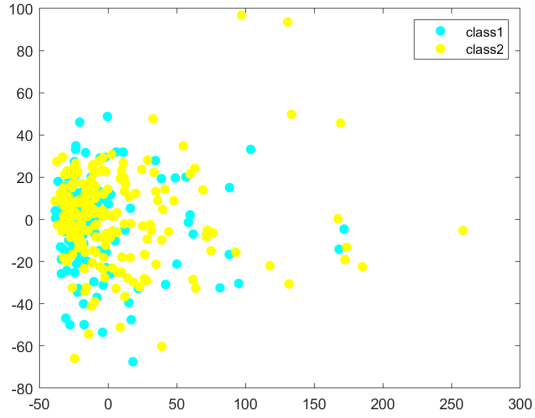
جدول ۵ عملکرد طبقه‌بندی الگوریتم پیشنهادی را بر روی چندین مجموعه داده نشان می‌دهد. ستون ۲ نشان‌دهنده دقت طبقه‌بند پیشنهادی در حالتی است که تمام داده‌های بدون برچسب اضافه شده‌اند. ستون ۳، دقت طبقه‌بند الگوریتم پیشنهادی را در حالتی که فقط نمونه‌های بدون برچسب نزدیک به مرز تصمیم اضافه شده‌اند نشان می‌دهد. همان طور که مشاهده می‌شود، الگوریتم پیشنهادی با انتخاب نمونه‌های نزدیک به مرز تصمیم در ۶ تا از ۱۰ مجموعه داده عملکرد بهتری را نشان می‌دهد. در این آزمایش‌ها نرخ داده‌های برچسب‌دار ۱۰٪ در نظر گرفته شده است.

۴-۶ بحث و بررسی

همان طور که در جداول ۳ و ۴ مشاهده می‌شود، الگوریتم پیشنهادی با استفاده از نمونه‌های بدون برچسب، عملکرد طبقه‌بندی الگوریتم ماشین



شکل ۳: دقت الگوریتم پیشنهادی و SVM نظارت‌شده با نرخ داده‌های برچسب‌دار از ۱۰٪ تا ۵۰٪ بر روی مجموعه داده‌های مختلف.



شکل ۴: مجموعه داده liver.

جدول ۵: میانگین نرخ دقت الگوریتم پیشنهادی با انتخاب همه داده‌های بدون برچسب و انتخاب نمونه‌های نزدیک به مرز تصمیم.

مجموعه داده	همه داده‌های بدون برچسب	داده‌های بدون برچسب انتخابی
Heart	۷۰٫۳۷	۷۰٫۳۷
Blood	۶۲	۶۲٫۰۵
Hepatits	۴۶٫۷۹	۴۶٫۸۱
Vote	۸۷٫۸۰	۸۷٫۷۹
Mushroom	۶۶	۶۵٫۸۶
Liver	۷۰٫۰۷	۷۱٫۱۵
Breast	۸۳٫۰۳	۸۸٫۵۷
Sonar	۵۴	۵۳٫۲۳
German	۶۹٫۴۵	۷۱٫۶۷
Colic	۵۹٫۷۶	۶۰٫۰۰
iris	۹۵٫۴۰	۹۵٫۹۶
wine	۸۹٫۲۲	۸۹٫۴۰

بردار پشتیبان نظارت‌شده را برای اکثر مجموعه داده‌ها بهبود داده است. با این حال، با توجه به نتایج گزارش‌شده برای مجموعه داده‌های mushroom، liver و breast نشان داده می‌شود که الگوریتم پیشنهادی برای تعدادی از مجموعه داده‌ها به خوبی کار نمی‌کند. به طور نمونه مجموعه داده liver رسم شده و همان طور که در شکل ۴ می‌توان مشاهده کرد، توزیع داده‌های هر دو کلاس هم‌پوشانی زیادی دارند. از این رو پیداکردن نمونه‌های نزدیک به مرز تصمیم کار آسانی نیست. در نتیجه الگوریتم پیشنهادی در مقایسه با سایر الگوریتم‌ها به خوبی کار نمی‌کند. در مقابل، برای مجموعه داده‌هایی که دارای توزیعی کاملاً مجزا هستند، الگوریتم پیشنهادی دارای عملکرد بهتری نسبت به سایر الگوریتم‌ها می‌باشد. همچنین الگوریتم پیشنهادی در مجموعه داده‌های با ابعاد بالا به علت استفاده از DBSCAN معمولاً کارایی خوبی ندارد.

۵- جمع‌بندی

در این مقاله، یک الگوریتم نیمه‌نظارتی خودآموز مطرح شد. در فرایند خودآموز به مجموعه‌ای از نقاط بدون برچسب، برچسب اختصاص داده می‌شود. طبقه‌بند پایه مورد استفاده SVM است که الگوریتمی مبتنی بر حاشیه می‌باشد. مجموعه‌ای از آزمایش‌ها بر روی تعدادی مجموعه داده انجام شد و کارایی الگوریتم پیشنهادی مورد ارزیابی قرار گرفت. بر طبق نتایج آزمایش‌ها نتیجه گرفته شد که الگوریتم پیشنهادی بهتر از دیگر الگوریتم‌ها عمل می‌کند. نقاط بدون برچسبی که نسبتاً به مرز تصمیم

نزدیک‌تر هستند در فرایند خودآموز برچسب‌دار شدند. اضافه‌شدن این نقاط به مجموعه برچسب‌دار تأثیر مثبتی بر روی پیداکردن مرز تصمیم بهینه داشت؛ زیرا این نقاط اگرچه نقاط مطمئنی نیستند و به علت نزدیکی به مرز تصمیم خطر تعلق‌داشتن به کلاس دیگر را دارند، اما نقاطی با اطلاعات بیشتر هستند. در نتیجه، این نقاط نسبت به نقاط دورتر، در رسیدن به بهبود کارایی طبقه‌بند مؤثرتر می‌باشند و اضافه‌کردن این نقاط به مجموعه داده‌های برچسب‌دار در فرایند خودآموز در به دست آمدن مرز تصمیم بهینه کمک زیادی می‌کنند. جهت اطمینان از تشخیص درست برچسب علاوه بر استفاده از الگوریتم همسایگی DBSCAN، از مکانیزم انتخاب نمونه‌ها از طریق ردکردن برچسب اشتباه استفاده شد. از معایب الگوریتم پیشنهادی، وابستگی آن به انتخاب مناسب چندین پارامتر است. همچنین به دلیل ساختار هندسی دایره‌ای DBSCAN، الگوریتم برای مجموعه داده‌هایی که نمونه‌های کلاس‌ها هم‌پوشانی زیادی دارند به خوبی کار نمی‌کند. کارهای آینده می‌توانند در زمینه مجموعه طبقه‌بندی نیمه‌نظارتی داده‌های نامتوازن و کلان‌داده‌ها باشند.

مراجع

[1] D. Wu, *et al.*, "Self-training semi-supervised classification based on density peaks of data," *Neurocomputing*, vol. 275, pp. 180-191, Jan. 2018.

[2] N. Zeng, Z. Wang, H. Zhang, W. Liu, and F. E. Alsaadi, "Deep belief networks for quantitative analysis of a gold immunochromatographic strip," *Cognitive Computation*, vol. 8, no. 4, pp. 684-692, 2016.

[3] N. Zeng, Z. Wang, and H. Zhang, "Inferring nonlinear lateral flow immunoassay state-space models via an unscented Kalman filter," *Science China Information Sciences*, vol. 59, no. 11, Article ID: 112204, 10 pp., 2016.

[4] N. Zeng, H. Zhang, W. Liu, J. Liang, and F. E. Alsaadi, "A switching delayed PSO optimized extreme learning machine for short-term load forecasting," *Neurocomputing*, vol. 240, pp. 175-182, May 2017.

[5] Y. Cao, H. He, and H. H. Huang, "LIFT: a new framework of learning from testing data for face recognition," *Neurocomputing*, vol. 74, no. 6, pp. 916-929, May 2011.

[6] F. Pan, J. Wang, and X. Lin, "Local margin based semi-supervised discriminant embedding for visual recognition," *Neurocomputing*, vol. 74, no. 5, pp. 812-819, Feb. 2011.

[7] D. Mallis, E. Sanchez, M. Bell, and G. Tzimiropoulos, "Unsupervised learning of object landmarks via self-training correspondence," *Advances in Neural Information Processing Systems*, vol. 33, pp. 4709-4720, 2020.

[8] G. Zhang, J. Wang, G. Shi, J. Zhang, and W. Dou, "A semi-supervised classification method for hyperspectral images by triple classifiers with data editing and deep learning," in *Proc. EIA Int. Conf. Cloud Computing, Smart Grid and Innovative Frontiers in Telecommunications,* pp. 171-183, Beiging, China, 4-5 Dec. 2019.

[9] J. Tanha, M. Van Someren, and H. Afsarmanesh, "Boosting for multiclass semi-supervised learning," *Pattern Recognition Letters*, vol. 37, pp. 63-77, Feb. 2014.

[10] D. Zhang, L. Jiao, X. Bai, S. Wang, and B. Hou, "A robust semi-supervised SVM via ensemble learning," *Applied Soft Computing*, vol. 65, pp. 632-643, Apr. 2018.

[11] J. Tanha, "MSSBoost: a new multiclass boosting to semi-supervised learning," *Neurocomputing*, vol. 314, pp. 251-266, Nov. 2018.

[12] C. Blake and C. Merz, *UCI Repository of Machine Learning Databases*, http://www.ics.uci.edu/?mlearn/MLRepository.html, University of California. Department of Information and Computer Science, Irvine, CA, p. 55, 1998.

[13] Z. H. Zhou and M. Li, "Semi-supervised learning by disagreement," *Knowledge and Information Systems*, vol. 24, no. 3, pp. 415-439, 2010.

[14] A. Blum and T. Mitchell, "Combining labeled and unlabeled data with co-training," in *Proc. of the 11th Annual Conf. on Computational Learning Theory*, pp. 92-100, Madison, WN, USA, 24 – 26 Jul. 1998.

[15] C. X. Ling, J. Du, and Z. H. Zhou, "When does co-training work in real data?" in *Proc. Pacific-Asia Conf. on Knowledge Discovery and Data Mining,* pp. 596-603, Bangkok, Thailand, 27-30 Apr. 2009.

[16] Z. Jiang, S. Zhang, and J. Zeng, "A hybrid generative/discriminative method for semi-supervised classification," *Knowledge-Based Systems*, vol. 37, pp. 137-145, Jan. 2013.

[17] S. Sun, "A survey of multi-view machine learning," *Neural Computing and Applications*, vol. 23, no. 7-8, pp. 2031-2038, Feb. 2013.

[18] M. Li and Z. H. Zhou, "SETRED: self-training with editing," in *Proc. Pacific-Asia Conf. on Knowledge Discovery and Data Mining,* pp. 611-621, Hanoi, Vietnam, 18-20 May 2005.

[19] R. Chen, *et al.*, "Semi-supervised anatomical landmark detection via shape-regulated self-training," *Neurocomputing*, vol. 471, pp. 335-345, Jan. 2022.

[20] Z. Yang and Y. Xu, "Laplacian twin parametric-margin support vector machine for semi-supervised classification," *Neurocomputing*, vol. 171, pp. 325-334, Jan. 2016.

[۲۱] ش. پوربهرامی، ا. خالدی و ل. خانعلی، "الگوریتم جدید خوشه‌بندی ارسال داده در شبکه‌های حسگر بی‌سیم با استفاده از دایره آپولونیوس،" *نشریه مهندسی برق و مهندسی کامپیوتر ایران، ب- مهندسی کامپیوتر، سال ۱۷، شماره ۳، صص. ۲۲۶-۲۱۹، پاییز ۱۳۹۸.*

[۲۲] ع. زاده بابایی، ع. باقری و خ. افشار، "ارائه یک الگوریتم خوشه‌بندی مبتنی بر چگالی با قابلیت کشف خوشه‌های با چگالی متفاوت در پایگاه داده‌های مکانی،" *نشریه مهندسی برق و مهندسی کامپیوتر ایران، ب- مهندسی کامپیوتر، سال ۱۵، شماره ۳، صص. ۱۸۶-۱۷۱، پاییز ۱۳۹۶.*

[23] M. Ester, H. P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proc. 2nd Int. Conf. on Knowledge Discovery and Data Mining*, *KDD'96,* pp. 226-231, Portland, ON, USA, 2-4 Aug. 1996.

[24] A. Lotfi, P. Moradi, and H. Beigy, "Density peaks clustering based on density backbone and fuzzy neighborhood," *Pattern Recognition*, vol. 107, Article ID: 107449, Nov. 2020.

[25] S. A. Seyedi, A. Lotfi, P. Moradi, and N. N. Qader, "Dynamic graph-based label propagation for density peaks clustering," *Expert Systems with Applications*, vol. 115, pp. 314-328, Jan. 2019.

[26] Y. Qin, Z. L. Yu, C. D. Wang, Z. Gu, and Y. Li, "A novel clustering method based on hybrid K-nearest-neighbor graph," *Pattern Recognition*, vol. 74, pp. 1-14, Feb. 2018.

[27] M. Emadi, J. Tanha, M. E. Shiri, and M. H. Aghdam, "A selection metric for semi-supervised learning based on neighborhood construction," *Information Processing & Management*, vol. 58, no. 2, Article ID: 102444, Mar. 2021.

[28] S. Pourbahrami and L. M. Khanli, *A Survey of Neighbourhood Construction Models for Categorizing Data Points*, arXiv preprint arXiv:1810.03083, 2018.

[29] S. Khezri, J. Tanha, A. Ahmadi, and A. Sharifi, "STDS: self-training data streams for mining limited labeled data in non-stationary environment," *Applied Intelligence*, vol. 50, no. 5, pp. 1-20, 2020.

[30] X. Gu, "A self-training hierarchical prototype-based approach for semi-supervised classification," *Information Sciences*, vol. 535, pp. 204-224, Oct. 2020.

[31] M. M. Adankon and M. Cheriet, "Help-training for semi-supervised support vector machines," *Pattern Recognition*, vol. 44, no. 9, pp. 2220-2230, Sept. 2011.

[32] M. Emadi and J. Tanha, "Margin-based semi-supervised learning using apollonius circle," in *Proc. Int Conf. on Topics in Theoretical Computer Science,* pp. 48-60, Tehran, Iran, 26-28 Aug. 2020.

[33] S. Pourbahrami, L. M. Khanli, and S. Azimpour, "A novel and efficient data point neighborhood construction algorithm based on Apollonius circle," *Expert Systems with Applications*, vol. 115, pp. 57-67, Jan. 2019.

[34] S. Pourbahrami, M. A. Balafar, L. M. Khanli, and Z. A. Kakarash, "A survey of neighborhood construction algorithms for clustering and classifying data points," *Computer Science Review*, vol. 38, Article ID: 100315, Nov. 2020.

[35] J. Tanha, M. van Someren, and H. Afsarmanesh, "Semi-supervised self-training for decision tree classifiers," *International J. of Machine Learning and Cybernetics*, vol. 8, no. 1, pp. 355-370, 2017.

[36] T. Joachims, "Transductive inference for text classification using support vector machines," in *Proc. of The 26th Int. Conf. on Machine Learning, ICML'99,* pp. 200-209, 1999.

[37] M. Belkin, P. Niyogi, and V. Sindhwani, "Manifold regularization: a geometric framework for learning from labeled and unlabeled examples," *J. of Machine Learning Research*, vol. 7, no. 85, pp. 2399-2434, 2006.

منا عمادی تحصیلات خود را در مقاطع کارشناسی و کارشناسی ارشد مهندسی کامپیوتر در سال‌های ۱۳۸۸ و ۱۳۹۰ از دانشگاه آزاد اسلامی واحد اراک و در مقطع دکتری مهندسی کامپیوتر- سیستم‌های نرم افزاری در سال ۱۴۰۰ از دانشگاه آزاد اسلامی واحد بروجرد به پایان رسانده است و هم‌اکنون استادیار دانشکده مهندسی کامپیوتر دانشگاه پیام نور می‌باشد. زمینه تحقیقاتی مورد علاقه ایشان در زمینه یادگیری ماشین و داده کاوی می‌باشد.

جعفر تنها تحصیلات خود را در مقاطع کارشناسی و کارشناسی ارشد علوم کامپیوتر از دانشگاه امیرکبیر و در مقطع دکتری هوش مصنوعی دانشگاه آمستردام هلند به پایان رسانده است. و هم‌اکنون دانشیار دانشگاه تبریز می‌باشد. زمینه‌های تحقیقاتی مورد علاقه ایشان داده کاوی یادگیری ماشین می‌باشد.

محمد ابراهیم شیری دکترای علوم کامپیوتر. استادیار دانشگاه صنعتی امیرکبیر زمینه-های تحقیقاتی ایشان داده کاوی و هوش مصنوعی می‌باشد.

مهدی حسین‌زاده اقدم دکترای تخصصی مهندسی کامپیوتر- هوش مصنوعی، دانشگاه علم و صنعت ایران را در سال ۱۳۹۵ اخذ نموده انست. در حال حاضر استادیار دانشگاه بناب می‌باشد. زمینه‌های تحقیقاتی مورد علاقه ایشان داده کاوی ، یادگیری ماشین، بازیابی اطلاعات و سیستم‌های توصیه‌گر می باشد.