مقاله پژوهشی

الگوریتم نیمه‌نظارتی جمعی با استفاده از معیار انتخاب مبتنی بر آستانه امتیاز اطمینان در جریان‌داده‌های غیر ایستا

شیرین خضری، جعفر تنها، علی احمدی و آرش شریفی

چکیده: در این مقاله، یک الگوریتم طبقه‌بندی نیمه‌نظارتی جمعی با استفاده از معیار انتخاب مبتنی بر آستانه امتیاز اطمینان تحت عنوان SSE-CBS در محیط‌های غیر ایستا ارائه می‌شود. رویکرد پیشنهادی از داده‌های دارای برچسب و فاقد برچسب با هدف مقابله با انواع تغییر مفهوم در جریان داده‌ها استفاده می‌کند. SSE-CBS مکانیزم مشهور وزن‌دهی بر اساس دقت الگوریتم‌های جمعی مبتنی بر بلوک را با ماهیت افزایشی الگوریتم درخت هافدینگ تلفیق می‌کند. الگوریتم پیشنهادی به طور تجربی با ۸ رویکرد منطبق بر جدیدترین دستاوردها، از جمله مدل‌های طبقه‌بندی نظارتی، نیمه‌نظارتی، منفرد و الگوریتم‌های جمعی مبتنی بر بلوک روی مجموعه داده‌های متنوع مقایسه شده است. بر اساس نتایج تجربی، SSE-CBS بهترین میانگین دقت طبقه‌بندی را نسبت به سایر رویکردهای نیمه‌نظارتی داراست و قادر است در محیط‌های دارای تغییر مفهوم با محدودیت داده برچسب‌دار عملکرد مناسبی داشته باشد.

کلیدواژه: **الگوریتم‌های طبقه‌بندی نیمه‌نظارتی، معیار انتخاب، مدل‌های طبقه‌بندی جمعی، تغییر مفهوم، جریان‌کاوی داده.**

۱- مقدمه

امروزه پیشرفت در فناوری، در بسیاری از حوزه‌ها مانند شبکه‌های حسگر، نظارت بر شبکه و شبکه‌های اجتماعی به تولید حجم عظیمی از داده منجر شده است. این داده‌ها همواره در حال تغییر و تحول بوده و حجمشان به قدری زیاد است که نمی‌توان آنها را در یک منبع ذخیره‌سازی نگهداری کرد و تنها بخش کوچکی از داده‌ها در هر لحظه در دسترس است [۱]. داده‌های دارای این ویژگی‌ها را جریان داده گویند. جریان‌کاوی داده‌ها امروزه از اهمیت بسزایی برخوردار است و هدف اصلی آن بهره‌برداری از دانش یا الگوهای پنهان داده‌ها با استفاده از روش‌های یادگیری ماشین و داده‌کاوی است [۲]. در [۳] یک بررسی جامع از الگوریتم‌های استخراج جریان داده‌ها با در نظر گرفتن مشکلات الگوریتم‌های خوشه‌بندی و طبقه‌بندی در این زمینه ارائه شده است، به این ترتیب که یک مدل کلی برای جریان‌کاوی را ارائه کرده و تفاوت بین داده‌کاوی سنتی و جریان‌کاوی را نشان می‌دهد. در [۲]، [۴]، [۵] و [۶] آخرین دستاوردهای جریان‌کاوی در حوزه‌های مختلف بحث و بررسی شده‌اند.

یکی از مهم‌ترین چالش‌های موجود در بحث جریان‌کاوی داده‌ها وجود تغییر مفهوم[1] در این داده‌ها می‌باشد که باعث می‌گردد به طور کلی نتوان روش‌های قدیمی را برای طبقه‌بندی جریان داده‌ها به کار برد. فرایند تغییر مفهوم در جریان داده‌ها در واقع از گرایش طبیعی رخدادهای دنیای واقعی به تغییر در طول زمان ناشی می‌گردند. بنابراین مدل داده‌ای که بر اساس داده‌های مشخصی در گذشته ساخته می‌شود، به مرور زمان دقت خود را از دست می‌دهد و بر روی داده‌های جدید قابل استفاده نمی‌باشند زیرا با تغییرات جریان داده‌ها هماهنگ نشده است. چالش بعدی مواجه‌شدن با داده‌های فاقد برچسب است، چرا که به علت سرعت بالای دریافت داده‌ها برچسب‌گذاری همه داده‌ها فرایندی بسیار پرهزینه و زمان‌بر است و نیازمند نظارت شخص خبره و تجهیزات خاص است [۷] و [۸].

در سال‌های اخیر، روش‌های زیادی برای مواجهه با داده‌های دارای تغییر مفهوم در طبقه‌بندی ارائه شده‌اند که به طور کلی عبارتند از روش‌های مبتنی بر پنجره[2] [۹]، الگوریتم‌های برخط [۱۰]، روش‌های مجهز به آشکارساز[3] [۱۱] و رویکردهای جمعی[4] [۱۲]. رویکردهای جمعی یکی از روش‌های مشهور برای بهبود دقت در مسایل یادگیری ایستا هستند و برای این که در محیط‌های پویا قابل استفاده باشند باید توسعه داده شوند، به عنوان مثال با بهبود ساختار الگوریتم (از طریق جایگزین کردن جدیدترین مولفه آموزش دیده با ضعیف‌ترین مولفه موجود)، استفاده از روش‌های بروزرسانی (از طریق بروزرسانی وزن مولفه طبقه‌بندها هنگام رای‌گیری) و یا بهره‌گیری از آموزش برخط با ورود هر نمونه داده (مانند الگوریتم Online Bagging [۱۳]).

در این مقاله طبقه‌بندهای مبتنی بر رویکردهای جمعی ارائه می‌شوند که داده‌های آموزشی را به صورت متوالی از بلوک‌هایی با اندازه ثابت تحت عنوان چانک[5] دریافت می‌کنند. در این روش‌ها، زمانی که بلوک داده فرامی‌رسد، یک مؤلفه طبقه‌بند جدید آموزش می‌بینند، مؤلفه‌های موجود ارزیابی می‌شوند و وزن‌هایشان به روز می‌شود. بر اساس نتایج ارزیابی، مؤلفه طبقه‌بند جدید آموزش‌دیده با بلوک داده اخیر، جایگزین ضعیف‌ترین مؤلفه می‌شود [۱۴].

با این حال، در صورت وقوع تغییر مفهوم در یک چانک با طول ثابت

این مقاله در تاریخ ۲۹ دی ماه ۱۳۹۸ دریافت و در تاریخ ۱۷ تیر ماه ۱۳۹۹ بازنگری شد.

شیرین خضری، دانشکده مهندسی مکانیک، برق و کامپیوتر، دانشگاه آزاد اسلامی، واحد علوم و تحقیقات، تهران، ایران، (email: sh.khezri@srbiau.ac.ir).

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

علی احمدی، دانشکده مهندسی کامپیوتر، دانشگاه خواجه نصیرالدین طوسی، تهران، ایران، (email: ahmadi@kntu.ac.ir).

آرش شریفی، دانشکده مهندسی مکانیک، برق و کامپیوتر، دانشگاه آزاد اسلامی، واحد علوم و تحقیقات، تهران، ایران، (email: a.sharifi@srbiau.ac.ir).

1. Concept Drift
2. Windowing Techniques
3. Detection Techniques
4. Ensemble Approaches
5. Chunk

الگوریتم‌های جمعی مذکور قادر نیستند به صورت مطلوبی به تغییرات واکنش نشان دهند. به ویژه در تغییرات ناگهانی، این الگوریتم‌ها بسیار کند واکنش نشان می‌دهند، زیرا طبقه‌بندهای ایجادشده از بلوک داده‌های منسوخ با وزن‌های نادرست در تصمیم‌گیری مشارکت دارند. بنابراین تنظیم اندازه مناسب برای چانک داده‌ها چالش مهمی در این دسته الگوریتم‌هاست. استفاده از چانک داده با اندازه کوچک، واکنش به تغییرات ناگهانی را تسهیل می‌کند در حالی که در بازه‌های زمانی بدون تغییر مفهوم کارایی الگوریتم کم می‌شود و هزینه محاسباتی نیز افزایش می‌یابد.

از طرف دیگر الگوریتم‌های جمعی برخط مانند Online Bagging و Leveraging Bagging [۱۵] نسبت به تغییرات ناگهانی به خوبی واکنش نشان می‌دهند، اما از مکانیزم‌های وزن‌دهی استفاده نمی‌کنند که بهره‌گیری از آنها می‌تواند راهکاری مناسب در واکنش به تغییرات تدریجی باشد. علاوه بر این، الگوریتم‌های جمعی برخط به این علت که با دیدن هر نمونه به روز رسانی می‌شوند، اغلب هزینه‌های محاسباتی بالاتری نسبت به الگوریتم‌های مبتنی بر بلاک دارند. بنابراین ترکیب ویژگی‌های مشخصی از این دو گروه می‌تواند شیوه مناسبی برای برخورد با هر دو نوع از تغییرات ناگهانی و تدریجی باشد.

از سوی دیگر اکثر رویکردهای مذکور فرض را بر این گذاشته‌اند که داده‌های برچسب‌دار به طور کامل در دسترس هستند و تعداد معدودی از این روش‌ها به کمبود داده‌های برچسب‌دار پرداخته‌اند. در عمل، دریافت داده‌های کاملاً برچسب‌دار در بسیاری از حوزه‌ها عملی نیست و به دلیل رشد بسیار سریع داده‌ها، برچسب‌گذاری آنها غیر ممکن است. وفور داده‌های بدون برچسب و عدم مشارکت آنها در آموزش مدل، ممکن است به از دست رفتن اطلاعات مفید نهفته در آنها منجر گردد. لذا استفاده از نمونه‌های بدون برچسب در آموزش و به روز رسانی مدل‌های یادگیری مفید خواهد بود. در داده‌کاوی به چنین رویکردهایی الگوریتم‌های یادگیری نیمه‌نظارتی[1] گویند. هدف اصلی رویکردهای نیمه‌نظارتی استخراج ساختار پنهان داده‌های بدون برچسب و ترکیب آن با اطلاعات صریح داده‌های برچسب‌دار برای بهبود دقت طبقه‌بندی است [۱۶]. یادگیری نیمه‌نظارتی می‌تواند به عنوان یک مسئله طبقه‌بندی یا خوشه‌بندی تعریف شود. روش‌های مختلفی برای طبقه‌بندی نیمه‌نظارتی داده‌ها وجود دارد، از جمله روش خودآموز[2] [۱۷]، آموزش همکارانه[3] [۱۸] و رویکردهای مبتنی بر حاشیه[4] مانند MSSBoost [۱۹]. این رویکردها عمدتاً برای داده‌کاوی سنتی توسعه یافته‌اند و مستقیماً برای جریان داده‌ها قابل استفاده نیستند. در این مقاله ما بر الگوریتم طبقه‌بندی نیمه‌نظارتی در جریان‌داده‌های غیر ایستا تمرکز می‌کنیم.

از جمله رویکردهای نیمه‌نظارتی که جهت برخورد با مسئله تغییر مفهوم در جریان داده‌ها ارائه شده‌اند می‌توان به الگوریتم‌های مبتنی بر EM[5]، الگوریتم‌های مبتنی بر خوشه[6] و رویکردهای جمعی اشاره کرد. به طور کلی، این روش‌ها با توجه به فرض اصلی آنها طبقه‌بندی می‌شوند. فرض اول، از مجموعه‌ای محدود از داده‌های برچسب‌دار در هر چانک استفاده می‌کند (به [۶] و [۱۶] مراجعه کنید). فرض دوم، داده‌های دارای برچسب فقط در چانک اول موجود هستند، نمونه‌های آن عبارتند از AMANDA [۲۰] و COMPOSE [۲۱]. با این حال بسیاری از این رویکردها، یادگیری نیمه‌نظارتی را یک کار خوشه‌بندی می‌دانند و به بهبود سطح شناخت خوشه‌ها با استفاده از داده‌های دارای برچسب به عنوان دانش قبلی تمایل دارند که همیشه یک روش بهینه در حوزه‌های مختلف نیست. در واقع در این حالت، از پتانسیل داده‌های برچسب‌دار به عنوان یک منبع مناسب برای آموزش یک مدل پیش‌بینی مناسب چشم‌پوشی شده است.

هدف این مقاله ارائه یک روش نیمه‌نظارتی با تکیه بر رویکردهای جمعی برای ساخت مدل‌های طبقه‌بندی در جریان‌داده‌هاست. به این صورت که از تعداد کمی داده برچسب‌دار و تعداد زیادی داده فاقد برچسب استفاده می‌شود و اطلاعات ضمنی موجود در آنها با اطلاعات صریح داده‌های برچسب‌دار ترکیب می‌گردد تا مدل طبقه‌بند کارا و قدرتمندی ایجاد شود. انتظار می‌رود الگوریتم نیمه‌نظارتی پیشنهادی SSE-CBS[7] با تغییرات مفهوم گوناگون به خوبی واکنش نشان دهد. هدف این است که در محیط‌های دارای محدودیت داده دارای برچسب، در کنار حفظ شمای ساده یادگیری مؤلفه طبقه‌بندها در الگوریتم‌های مبتنی بر چانک و وزن‌دهی پیش‌بینی‌ها از ویژگی‌های برجسته روش‌های برخط نیز به خوبی استفاده شود.

نوآوری اصلی این کار به روز رسانی افزایشی مؤلفه‌های مجموعه در محیط‌های با داده برچسب‌دار محدود با هدف بهبود واکنش در برابر انواع تغییرات مفهوم و همچنین کاهش تأثیر اندازه چانک است. به روز رسانی برخط به همه مؤلفه‌های مجموعه امکان می‌دهد که به طور هم‌زمان با جدیدترین مفاهیم سازگار شوند. برای نشان‌دادن کارایی الگوریتم پیشنهادی، در آزمایش‌ها از مجموعه کاملی از داده‌های مصنوعی و داده‌های واقعی استفاده می‌شود. نتایج تجربی نشان می‌دهند هنگامی که تعداد بسیار کمی داده برچسب‌دار وجود داشته باشد، روش پیشنهادی بهتر از سایر الگوریتم‌های یادگیری نیمه‌نظارتی عمل می‌کند. بخش‌بندی مقاله به این صورت است که در بخش دوم مروری بر کارهای مرتبط، در بخش سوم الگوریتم پیشنهادی، در بخش چهارم نتایج ارزیابی روش پیشنهادی در مقایسه با دیگر روش‌ها و در نهایت نتیجه‌گیری در بخش پنجم ارائه شده است.

۲- کارهای مرتبط

وقوع تغییرات در مفهوم و الگوی داده‌ها دقت نتایج طبقه‌بندی را کاهش می‌دهد. برای مدیریت تغییر مفهوم در محیط‌های پویا، روش‌های یادگیری تطبیقی[8] نیاز است به این معنی که هنگام وقوع تغییر، مدل فعلی بایستی به روز رسانی شود تا دقت آن حفظ گردد. در سال‌های اخیر، روش‌های متعددی برای کنترل تغییر مفهوم در جریان داده ارائه شده‌اند [۲۲] و [۲۳]. این روش‌ها بر اساس دو جنبه کلی طبقه‌بندی می‌شوند: تعداد طبقه‌بندها و استراتژی به روز رسانی.

دسته اول، شامل دو گروه مدل‌های طبقه‌بندی منفرد[9] و مدل‌های طبقه‌بندی جمعی است. تعدادی از رویکردهای طبقه‌بندی منفرد تغییر مفهوم را با استفاده از اطلاعات آماری مربوط به توزیع داده‌ها تشخیص می‌دهند و در صورت کشف تغییر مفهوم، مدل را به روز می‌کند یا مجدداً

1. Semi-Supervised Learning
2. Self-Training
3. Co-Training
4. Margin-Based
5. Expectation-Maximization
6. Cluster-Based
7. Semi-Supervised Ensemble Using Confidence-Based Selection Metric
8. Adaptive
9. Single-Mode Classifier

آن را آموزش می‌دهد [۲۴]. الگوریتم VFDT یک الگوریتم طبقه‌بندی منفرد افزایشی است که برای جریان‌کاوی محیط‌های ایستا [۲۵] ارائه شده و در [۲۶] نسخه توسعه‌یافته آن با امکان تشخیص تغییر مفهوم تحت عنوان CVFDT[1] آمده است. در این رویکرد از یک پنجره با اندازه ثابت روی نمونه‌های اخیر برای به روز رسانی مدل طبقه‌بند استفاده می‌شود.

رویکردهای مبتنی بر الگوریتم‌های جمعی، متشکل از مجموعه‌ای از مؤلفه طبقه‌بندهای وزن‌دار هستند که با به روز رسانی مؤلفه‌ها و وزن‌هایشان قادر به کنترل تغییرات خواهند بود. در [۱۲] طبقه‌بند AWE[2] مبتنی بر چانک داده برای کاوش جریان داده‌های دارای تغییر مفهوم ارائه شده است. ایده اصلی الگوریتم این است که با ورود هر چانک یک طبقه‌بند جدید آموزش می‌بینند. بعد از آموزش طبقه‌بند جدید، همه مؤلفه‌های طبقه‌بندی جمعی[3] بر اساس جدیدترین چانک داده ارزیابی می‌شوند. این ارزیابی‌ها بر اساس نسخه خاصی از میانگین خطای مربعات (MSE)[4] انجام می‌شوند که بر اساس آن الگوریتم، k بهترین مؤلفه را برای ایجاد مجموعه انتخاب کرده و وزن هر طبقه‌بند بر اساس دقتش تنظیم می‌شود. معمولاً جدیدترین مؤلفه با ضعیف‌ترین مؤلفه موجود جایگزین می‌شود. ترکیب پیش‌بینی مؤلفه‌ها بر اساس رأی‌گیری اکثریت وزن‌دهی شده[5] انجام می‌شود.

در [۱۳] الگوریتم Online Bagging یک روش جمعی ارائه شده است که مؤلفه‌هایش از نوع یادگیرنده‌های افزایشی هستند که تصمیماتشان با استفاده از رأی‌گیری اکثریت ترکیب می‌شود. نمونه‌برداری[6] برای بگینگ دسته‌ای به صورت افزایشی با ارائه هر نمونه به یک مؤلفه، k بار انجام می‌شود که k با توزیع پواسن تعریف می‌شود. نسخه اصلاح‌شده آن تحت عنوان Leveraging Bagging در [۱۵] ارائه شده است که قصد دارد خاصیت تصادفی بیشتری به ورودی و خروجی طبقه‌بندهای پایه اضافه کند. در [۲۷] الگوریتم جمعی افزایشی دیگری تحت عنوان DWM[7] ارائه شده است. در این الگوریتم، مجموعه‌ای از طبقه‌بندهای افزایشی بر اساس دقتشان بعد از دریافت هر نمونه، وزن‌دهی می‌شوند و با بروز خطا در هر مؤلفه‌ای وزن آن مؤلفه بر اساس یک معیار از پیش تعیین شده توسط کاربر کاهش می‌یابد. در [۲۸] الگوریتم جمعی افزایشی دیگری با رویکردی متفاوت تحت عنوان HOT[8] ارائه شده است. تعمیم درخت هافدینگ (HT)[9] شامل گره‌های ترجیح[10] است که صرفاً به جای انتخاب بهترین صفت شکاف[11]، همه صفات امیدبخش[12] نگهداری می‌شوند. سپس برای هر یک از این صفات یک زیردرخت تصمیم ساخته می‌شود. تصمیم‌گیری بر اساس ترکیب وزنی پیش‌بینی همه زیردرخت‌های قابل استفاده است. الگوریتم‌های SEA[13] [۱۴] و Learn++.NSE در [۲۹] نیز جزو رویکردهای مبتنی بر چانک داده هستند. الگوریتم SEA ساختار خود را بر اساس تغییر مفهوم تغییر می‌دهد. استراتژی جایگزینی ضعیف‌ترین طبقه‌بند پایه بر اساس دقت[14] مؤلفه‌هاست و تصمیم‌گیری نیز مبتنی بر رأی‌گیری اکثریت انجام می‌شود. در این الگوریتم تعیین اندازه مناسب چانک داده دشوار است. مسئله قابل توجه این است که شباهت توزیع داده‌ها در چانک‌ها به شدت به اندازه چانک بستگی دارد. چانک داده بزرگ‌تر طبقه‌بندهای دقیق‌تری خواهد ساخت اما ممکن است حاوی بیش از یک تغییر باشد. از طرف دیگر چانک داده‌های کوچک‌تر در تشخیص تغییرات بهتر عمل می‌کنند اما معمولاً به تولید طبقه‌بندهای ضعیف‌تری منجر می‌شوند. الگوریتم‌های جمعی که روی چانک‌های بزرگ ساخته شده‌اند نسبت به تغییرات ناگهانی عکس‌العمل بسیار کندی نشان می‌دهند. برای غلبه بر واکنش کند الگوریتم AWE[15] [۱۲] به تغییرات، در [۳۰] رویکرد ACE[16] ارائه شده که متشکل از یک طبقه‌بند برخط، تعداد زیادی طبقه‌بند دسته‌ای[17] و یک آشکارساز خطا[18] است. با ورود هر داده، طبقه‌بند برخط به صورت افزایشی آموزش داده می‌شود و بلوک داده به تدریج گسترش می‌یابد. به علاوه آشکارساز، تغییر دقت میانگین هر طبقه‌بند دسته‌ای را روی بلوک جاری چک می‌کند. در صورت کشف تغییر مفهوم، یک طبقه‌بند دسته‌ای جدید ایجاد شده و طبقه‌بند برخط مجدداً تنظیم[19] می‌شود. خروجی نهایی از تجمیع پیش‌بینی‌های طبقه‌بند برخط و طبقه‌بندهای دسته‌ای با رأی اکثریت وزن‌دهی شده به دست می‌آید. ویژگی بارز الگوریتم ACE این است که تعداد مؤلفه‌های پایه مجموعه محدود نیست. همین ویژگی باعث می‌شود که به خوبی با تغییرات بازگشتی[20] مقابله کند و نیز به علت وجود طبقه‌بند برخط و آشکارساز تغییر نسبت به تغییرات ناگهانی عکس‌العمل سریع‌تری داشته باشد. نقطه ضعف آن این است که برای کشف تغییر مفهوم به جای استفاده از کل مجموعه تنها بر اساس کارایی بهترین طبقه‌بند دسته‌ای عمل می‌کند. با هدفی مشابه برای مقابله با تغییر مفهوم تدریجی و ناگهانی، در [۳۱] الگوریتم AUE۲[21] ارائه شده که نسخه بهبودیافته الگوریتم AUE [۳۲] است. به این صورت که مکانیزم وزن‌دهی مبتنی بر دقت روش‌های شورایی مبتنی بر چانک را با طبیعت افزایشی درخت هافدینگ ترکیب می‌کند. این الگوریتم مجموعه‌ای وزن‌دهی شده از طبقه‌بندها را نگهداری می‌کند و کلاس نمونه‌های ورودی را با تجمیع پیش‌بینی مؤلفه‌ها از طریق قاعده رأی‌گیری وزن‌دهی شده پیش‌بینی می‌کند. بعد از هر چانک داده، یک طبقه‌بند جدید ایجاد می‌شود که جایگزین ضعیف‌ترین مؤلفه شورا می‌شود. کارایی هر مؤلفه طبقه‌بند با تخمین خطای پیش‌بینی روی جدیدترین چانک داده ارزیابی می‌شود و وزن مؤلفه‌ها بر اساس دقتشان تنظیم می‌شود. در [۳۳] نیز یک رویکرد جمعی مبتنی بر جنگل (ARF)[22] ارائه شده که از شیوه مؤثری برای نمونه‌گیری بهره می‌برد و با تجهیز به رویکردهای آشکارساز تغییر مفهوم، قادر به مقابله با انواع تغییرات مفهوم است.

دسته دوم، شامل دو روش به روز رسانی کور[23] و آگاهانه[1] است. در

1. Concept-Adapting Very Fast Decision Tree
2. Accuracy Weighted Ensemble
3. Ensemble
4. Mean Square Error
5. Weighted Majority Vote
6. Sampling
7. Dynamic Weighted Majority
8. Hoeffding Option Tree
9. Hoeffding Tree
10. Option Node
11. Best Split Attribute
12. Promising Attribute
13. Streaming Ensemble Accuracy
14. Accuracy
15. Accuracy Weighted Ensemble
16. Adaptive Classifer Ensemble
17. Batch Classifier
18. Drift Detector
19. Reset
20. Recurring Drift
21. Accuracy Updated Ensemble
22. Adaptive Random Forest
23. Blind

روش کور، الگوریتم طبقه‌بندی در بازه‌های منظم و بدون توجه به این که آیا تغییری رخ داده است یا خیر، به روز رسانی می‌شود. در صورتی که در روش آگاه، مدل‌ها معمولاً مجهز به یک روش تشخیص تغییر[۲] هستند و فقط هنگام وقوع تغییر مفهوم به روز رسانی می‌شوند.

در روش کور می‌توان به روش‌های نمونه وزن‌دهی شده[۳] [۳۴] و پنجره با اندازه ثابت[۴] [۳۵] اشاره کرد. در روش نمونه وزن‌دهی شده، به هر نمونه بر اساس سن[۵] آن وزنی داده می‌شود، به این صورت که به داده‌های قدیمی‌تر وزن کمتری منسوب می‌شود تا روی داده‌های جدیدتر که حاوی مفاهیم جدیدتری هستند بیشتر تمرکز شود. در روش پنجره با اندازه ثابت، تعداد ثابتی از نمونه‌ها با گذر زمان مورد توجه قرار می‌گیرند. متأسفانه هنگام استفاده از پنجره با اندازه ثابت، کاربر گرفتار مصالحه[۶] می‌شود. چرا که اگر طبقه‌بند روی یک پنجره کوچک از نمونه‌ها ساخته شود، به سرعت به تغییرات واکنش نشان می‌دهد ولی در دوره‌هایی که مفهوم پایدار است کارایی سیستم کاهش می‌یابد. از طرف دیگر اگر طبقه‌بند روی یک پنجره بزرگ از مثال‌ها ساخته شود در دوره‌هایی که مفهوم پایدار است کارایی خوبی دارد، اما در مواجهه با تغییر مفهوم کند عمل می‌کند. پنجره زمانی با اندازه انعطاف‌پذیر نمونه‌ای از روش‌های آگاه است. یک قاعده کلی در تنظیم اندازه پنجره این است که اگر تغییر مفهوم کشف شد اندازه پنجره کاهش یابد تا از ورود داده‌های منسوخ جلوگیری شود، در غیر این صورت اندازه پنجره افزایش یابد تا داده‌های جدید بیشتری را شامل شود [۲۰]. از آنجا که روش‌های آگاه روند کاراتری برای تشخیص تغییر مفهوم و اجتناب از به روز رسانی‌های خارج از کنترل دارند بیشتر مورد توجه واقع شده‌اند. بحث اصلی چگونگی کشف تغییر مفهوم است که با تکیه بر رویکردهای راه‌انداز[۷]، به تغییرات مفهوم واکنش نشان می‌دهند و زمانی که طبقه‌بند بایستی بازسازی[۸] یا به روز رسانی شود هشدار[۹] می‌دهند. بیشتر تحقیقات موجود حداقل یک شاخص کارایی را با گذر زمان مد نظر قرار داده‌اند [۱۱]، [۳۰] و [۳۶]. دقت طبقه‌بند، معمول‌ترین شاخص است، به این صورت که اگر افت مداومی[۱۰] در دقت رخ دهد تغییر مفهوم آشکار شده است. رویکرد دیگر، نظارت بر توزیع داده در دو پنجره مختلف است که اختلاف توزیع بیانگر تغییر مفهوم است. در [۳۷] روش تشخیص تغییر مفهوم EDDM[۱۱] طراحی شده که بر نرخ خطای طبقه‌بندی نظارت دارد. اگر میزان خطا به سطح اخطار رسید، بیانگر این است که توزیع داده تغییر کرده است. الگوریتم‌های دیگری نیز ارائه شده‌اند که تغییر مفهوم را با استفاده از روش‌های پنجره از طریق مقایسه توزیع داده پنجره‌ها کشف می‌کنند. به عنوان مثال ADWIN[۱۲] [۹] انواع مختلفی از تغییرات را با استفاده از پنجره‌های شناور با جدیدترین نمونه‌ها کشف می‌کند به این صورت که اگر میانگین بین دو زیرپنجره از آستانه‌ای بزرگ‌تر باشد تغییر رخ داده است.

1. Inform
2. Change Detection
3. Weighted Examples
4. Fixed Size Time Windows
5. Age
6. Trade off
7. Trigger Approaches
8. Retrain
9. Alarm
10. Consistent Drop
11. Early Drift Detection Method
12. Adaptive Windowing

همان گونه که بحث شد روش‌های زیادی برای غلبه بر تغییر مفهوم در طبقه‌بندی ارائه شده‌اند ولی اکثر آنها فرض را بر این گذاشته‌اند که داده‌های برچسب‌دار به طور کامل در دسترس هستند و تعداد معدودی از آنها به کمبود داده‌های برچسب‌دار اشاره کرده‌اند. با این حال، در عمل معمولاً تعداد کمی داده برچسب‌دار و تعداد بسیار زیادی داده فاقد برچسب[۱۳] وجود دارد. الگوریتم‌های نیمه‌نظارتی رویکرد مناسبی برای حل مشکل کمبود داده برچسب‌دار در طبقه‌بندی هستند.

در [۳۸] یک رویکرد نیمه‌نظارتی برای مسئله طبقه‌بندی در جریان‌داده‌ها ارائه شده است. روال الگوریتم به این صورت است که جریان داده را به چانک‌های با اندازه برابر تقسیم می‌کند، سپس روی داده‌های هر چانک یک الگوریتم خوشه‌بندی اعمال می‌شود و اطلاعات آماری داده‌های متعلق به هر خوشه ذخیره می‌گردد و به آن "میکروخوشه" گفته می‌شود. میکروخوشه‌ها به عنوان یک مدل طبقه‌بندی به کار می‌روند. برای طبقه‌بندی یک نمونه بدون برچسب از الگوریتم k-NN برای یافتن Q نزدیک‌ترین میکروخوشه از نمونه استفاده می‌شود و کلاسی انتخاب می‌شود که بیشترین تکرار داده برچسب‌دار در این Q خوشه را داشته باشد. این الگوریتم به منظور مقابله با مسئله تغییر مفهوم، مجموعه‌ای از L مدل را نگهداری می‌کند. هر زمان که یک مدل جدید با استفاده از چانک داده جدید ساخته شد، L مدل از $L+1$ مدل جدید بر اساس دقتشان روی چانک جدید انتخاب می‌شوند. الگوریتمی مشابه در [۳۹] تحت عنوان SPASC ارائه شده که از طبقه‌بندی مبتنی بر خوشه و الگوریتم EM برای بهره‌برداری از اطلاعات از داده‌های بدون برچسب استفاده می‌کند. آنها ادعا می‌کنند این رویکرد خوشه‌های خالصی بر روی داده‌های برچسب‌دار ایجاد می‌کند.

اخیراً الگوریتم نیمه‌نظارتی SCo-Forest[۱۴] [۴۰] برای حل مشکلات طبقه‌بندی در جریان داده‌ها با داده‌های برچسب‌دار محدود[۱۵] پیشنهاد شده است. SCo-Forest رویکرد جنگل تصادفی همکارانه[۱۶] [۴۱] را برای جریان داده‌ها توسعه داده و در رویکرد پیشنهادی برای مقابله با تغییر مفهوم از آشکارساز ADWIN استفاده شده است.

در [۲۴] یک الگوریتم نیمه‌نظارتی منفرد بر اساس الگوریتم خودآموز[۱۷]، تحت عنوان STDS[۱۸] ارائه شده است که از روش واگرایی KL[۱۹] برای اندازه‌گیری اختلاف توزیع بین چانک‌های متوالی به منظور تشخیص تغییر مفهوم استفاده می‌کند.

در تعدادی از رویکردهای پیشنهادی فرض بر این است که نمونه‌های برچسب‌دار فقط در حین آموزش اولیه طبقه‌بندها در دسترس هستند. پس از این مرحله، تمام داده‌های دریافتی فاقد برچسب هستند. این فرض تأخیر تأیید شدید (EVL)[۲۰] نام‌گذاری شده است. AMANDA [۲۰] یکی از الگوریتم‌هایی است که تحت فرض EVL کار می‌کند و یک مدل طبقه‌بندی نیمه‌نظارتی منفرد برای مقابله با تغییر مفهوم می‌باشد. در مرحله اول با داده‌های برچسب‌دار یک مدل طبقه‌بند ایجاد می‌شود، سپس

13. Unlabeled Data
14. Streaming Co-Forest
15. Limited Labeled Data
16. Co-Random Forest
17. Self-Training
18. Self-Training Data Streams
19. Kullback-Leibler
20. Extreme Verification Latency

جدول ۱: ویژگی‌های کلیدی الگوریتم‌های برتر.

الگوریتم	تعداد طبقه‌بند	شیوه مواجهه با تغییر مفهوم	شیوه یادگیری
CVFDT	منفرد	کور	نظارتی
SEA	جمعی	کور	نظارتی
HOT	جمعی	کور	نظارتی
DWM	جمعی	کور	نظارتی
AWE	جمعی	کور	نظارتی
Online Bagging	جمعی	آگاه	نظارتی
AUE۲	جمعی	کور	نظارتی
ARF	جمعی	آگاه	نظارتی
AMANDA	منفرد	کور	نیمه‌نظارتی
SPASC	جمعی	کور	نیمه‌نظارتی
STDS	منفرد	آگاه	نیمه‌نظارتی
SCo-F	جمعی	آگاه	نیمه‌نظارتی

با استفاده از آن، نمونه‌های بدون برچسب را طبقه‌بندی[1] کرده و با انتساب وزن به آنها بر اساس یک الگوریتم مبتنی بر چگالی اهمیت نمونه‌های فاقد برچسب را مشخص می‌کند. نهایتاً نمونه‌ها فیلتر شده و چگال‌ترین نمونه‌ها و به عبارتی بارزترین آنها به عنوان هسته اصلی (CSE)[2] انتخاب می‌شوند که از آنها به عنوان داده‌های آموزشی برچسب‌دار برای مدل طبقه‌بند در چانک‌های بعدی که تماماً بدون برچسب هستند استفاده می‌شوند. ویژگی‌های کلیدی تعدادی از الگوریتم‌های برتر در جدول ۱ ذکر شده‌اند.

همان طور که بحث شد بسیاری از الگوریتم‌های نیمه‌نظارتی در جریان‌داده‌ها عمدتاً از روش‌های مبتنی بر خوشه‌بندی استفاده کرده‌اند اما روش‌های خوشه‌بندی معمولاً از عدم وجود یک معیار شباهت[3] مناسب رنج می‌برند. لذا در این مقاله یک رویکرد جمعی نیمه‌نظارتی جدید ارائه شده که از روش خودآموز[4] استفاده می‌کند و قادر است تغییر مفهوم را در جریان‌داده‌های با حداقل تعداد داده برچسب‌دار کنترل کند.

۳- چارچوب الگوریتم نیمه‌نظارتی جمعی

اکثر الگوریتم‌های طبقه‌بندی جریان‌داده معمولاً قادر به مقابله با یک تغییر مفهوم خاص در محیط‌های با داده تماماً برچسب‌دار هستند. هدف این مقاله ارائه الگوریتمی است که قادر به مواجهه با انواع تغییر مفهوم در محیط‌های با داده برچسب‌دار محدود باشد. بدین منظور یک الگوریتم نیمه‌نظارتی جمعی پیشنهاد داده‌ایم که مکانیزم وزن‌دهی بر اساس دقت الگوریتم‌های جمعی مبتنی بر بلوک را با ماهیت افزایشی الگوریتم درخت هافدینگ ترکیب می‌کند و این الگوریتم را SSE-CBS نامیده‌ایم. در این بخش ابتدا تنظیمات جریان داده‌های نیمه‌نظارتی ارائه می‌شود، در زیربخش بعد توضیحاتی در مورد طبقه‌بند درخت هافدینگ، سپس جزئیات رویکرد پیشنهادی SSE-CB و نهایتاً پیچیدگی زمانی رویکرد پیشنهادی ارائه خواهد شد.

۳-۱ تنظیمات جریان داده نیمه‌نظارتی

فرض کنید S جریان داده‌ای است که داده‌ها به صورت چانک‌های با اندازه یکسان $\{D_1,\dots,D_n\}$، دریافت شده و پردازش می‌شوند و هر چانک حاوی d نمونه است که تعداد کمی از آنها برچسب‌دار (l) و بقیه بدون برچسب (u) هستند به طوری که تعداد داده‌های برچسب‌دار خیلی کمتر از داده‌های بدون برچسب هستند، یعنی $l \ll u$.

۳-۲ طبقه‌بند درخت هافدینگ

یادگیرنده‌های درخت‌های تصمیم‌گیری کلاسیک مانند ID۳ و C۴.۵ و CART فرض می‌کنند که تمام نمونه‌های آموزشی می‌توانند به صورت هم‌زمان در حافظه اصلی ذخیره شوند و بنابراین به شدت به تعداد نمونه‌های مجموعه آموزشی وابسته هستند. یادگیرنده‌های درخت‌های تصمیم‌گیری مبتنی بر دیسک مانند SLIQ و SPRINT فرض می‌کنند که نمونه‌ها در دیسک ذخیره شده‌اند و با خواندن‌های تکراری آنها به صورت ترتیبی (به صورت کارامد یک بار برای هر سطح درخت) عمل یادگیری را انجام می‌دهند. این روش‌ها وقتی که اندازه مجموعه آموزشی مورد استفاده بسیار بزرگ شود و هنگام یادگیری از درختان پیچیده (درختان با سطوح فراوان) هزینه بسیار زیاد دارند و وقتی که مجموعه داده بسیار بزرگ باشد و در فضای دیسک در دسترس قرار نگیرد این روش‌ها شکست می‌خورند.

بنابراین یک یادگیرنده درخت تصمیم‌گیری به نام درخت هافدینگ [۲۵] برای مجموعه داده بسیار بزرگ طراحی شده است. در این یادگیرنده هر نمونه را حداکثر یک بار می‌خواند و زمانی ثابت برای پردازش آن دارد. این امر امکان داده‌کاوی مستقیم به صورت برخط منبع داده را حتی بدون ذخیره نمونه‌ها ممکن می‌سازد و از سوی دیگر ساختن درخت‌های بسیار پیچیده با هزینه محاسبات قابل قبول ممکن می‌شود. عمل تقسیم با استفاده از بهترین خصوصیت موجود صورت می‌پذیرد. البته هنگام تقسیم این نکته نیز مد نظر قرار می‌گیرد که تعداد داده‌های مورد استفاده شرایط و حدود هافدینگ را برآورده سازد. این روش این خصوصیت را دارد که خروجی آن شبیه به خروجی یادگیرنده‌های قدیمی می‌باشد. موضوعاتی که این روش به آنها می‌پردازد شامل موارد زیر است:

- **یکسانی خصوصیات مورد استفاده برای تقسیم (شکافتن) و ایجاد درخت تصمیم:** فرض کنید در مرحله‌ای هستید که می‌خواهید داده‌های موجود را بر اساس یکی از خصوصیات تقسیم کنید و به عبارتی شاخه‌های زیرین جدیدی ایجاد نمایید. حال بحث اصلی این است که کدام خصوصیت باید برای تقسیم داده‌ها استفاده گردد. برای مشخص‌کردن خصوصیتی که باید برای تقسیم مورد استفاده قرار گیرد، معمولاً از معیارهایی نظیر بهره اطلاعاتی استفاده می‌شود. حال اگر مقادیر این معیارها برای دو یا چند خصوصیت که بیشترین مقدار این معیار را در بین بقیه خصوصیات دارند، یکسان بود، کدام خصوصیت باید برای تقسیم داده‌ها مورد استفاده قرار گیرد؟ هنگامی که به چنین مشکلی برمی‌خوریم، این مطلب که تصمیم‌گیری را تا زمان نامحدود و نامشخصی به تعویق بیندازیم تا داده‌های کافی برای تصمیم‌گیری صحیح دریافت گردند، غیر قابل قبول است، بلکه باید در مورد انتخاب یکی از خصوصیات بر اساس مجموعه رکوردها و داده‌هایی که تا کنون دریافت شده‌اند، تصمیم‌گیری نماییم و نتایج حاصل از اشتباه در انتخاب را نیز قبول کنیم.
- **حافظه با حدود مشخص:** درخت ارائه‌شده تا زمانی که حافظه

1. Classify
2. Core Support Extraction
3. Similarity Metric
4. Self-Training

موجود باشد، قابلیت رشد و بزرگ‌شدن را دارد. اما با توجه به محدودیت در وجود حافظه لازم است تا روش‌هایی در ارتباط با نگهداری و بسط یک درخت کارا به وجود آید.

- **کارایی و دقت:** این موضوع یکی از ویژگی‌های اصلی است که هر الگوریتم جریان‌کاوی باید حتماً به آن دقت کند.

برای یافتن بهترین صفت جهت آزمون در یک گره داده‌شده، کافی است زیرمجموعه‌ای کوچک از نمونه‌های آموزشی که از آن گره می‌گذرند مورد توجه قرار گیرند. بنابراین با داشتن جریانی از نمونه‌ها، اولین نمونه‌ها برای انتخاب آزمون در ریشه استفاده می‌شوند. وقتی که صفت ریشه انتخاب شد نمونه‌های موفق به سمت برگ‌های مربوط فرستاده می‌گردند و برای انتخاب صفت مناسب در آنجا استفاده می‌شوند و این کار به صورت بازگشتی انجام می‌شود.

۳-۳ رویکرد پیشنهادی SSE-CBS

الگوریتم پیشنهادی مجموعه‌ای از مؤلفه طبقه‌بندهای وزن‌دار را نگهداری می‌کند و کلاس داده‌های ورودی را با تجمیع[1] پیش‌بینی‌های مؤلفه‌ها با استفاده از قانون رأی‌گیری وزن‌دهی شده منسوب می‌کند. با ورود هر چانک داده، یک طبقه‌بند جدید با استفاده از داده‌های برچسب‌دار آن ساخته می‌شود. طبقه‌بند جدید جایگزین ضعیف‌ترین طبقه‌بند موجود (طبقه‌بند با حداقل وزن) در مجموعه می‌شود. همچنین وزن (کارایی) هر مؤلفه با تخمین خطای پیش‌بینی آن روی داده‌های برچسب‌دار جدیدترین چانک ارزیابی می‌شود. بعد از جایگزینی ضعیف‌ترین مؤلفه، داده‌های بدون برچسب با استفاده از مؤلفه‌های موجود در مجموعه برچسب‌گذاری شده و امتیاز اطمینان هر یک از نمونه‌ها محاسبه می‌شود. سپس یک زیرمجموعه از آنها با استفاده از یک معیار جدید انتخاب شده و به داده‌های برچسب‌دار چانک جاری اضافه می‌شوند و مؤلفه‌های موجود در مجموعه با داده‌های برچسب‌دار به روز رسانی می‌شوند. در اینجا از درخت هافدینگ به عنوان طبقه‌بند پایه[2] مجموعه استفاده شده است. از آنجایی که می‌توان الگوریتم پیشنهادی را به عنوان یک چارچوب کلی در نظر گرفت، بنابراین هر الگوریتم افزایشی دیگری را می‌توان به عنوان طبقه‌بند پایه استفاده کرد.

برای هر چانک ورودی D_i، وزن هر مؤلفه طبقه‌بند h_j برابر است با $\alpha_{ij} : j = ۱,\ldots,k$ که بر اساس تخمین نرخ خطا روی داده‌های برچسب‌دار چانک D_i بر اساس (۳) محاسبه می‌شود

$$MSE_{ij} = \frac{۱}{|L_i|} \sum_{\{x,y\} \in L_i} (۱ - f_y^j(x))^۲ \qquad (۱)$$

$$MSE_r = \sum_y p(y)(۱ - p(y))^۲ \qquad (۲)$$

$$\alpha_{ij} = \frac{۱}{MSE_{ij} + MSE_r + \varepsilon} \qquad (۳)$$

تابع $f_y^j(x)$ احتمال انتساب کلاس y به نمونه x توسط طبقه‌بند h_j را نشان می‌دهد. با الهام از الگوریتم AUE۲، مقدار MSE_{ij} خطای پیش‌بینی طبقه‌بند h_j را روی داده‌های برچسب‌دار چانک D_i تخمین می‌زند و MSE_r خطای میانگین مربعات یک طبقه‌بند با پیش‌بینی تصادفی است و به عنوان مرجع برای توزیع کلاس فعلی (با تقریب روی داده برچسب‌دار چانک جاری L_i) استفاده می‌شود. به علاوه مقدار مثبت خیلی کوچک ε به معادله افزوده می‌شود تا اطمینان دهد که حتی اگر MSE_r و MSE_{ij} صفر شوند، وزن α_{ij} قابل محاسبه است. هدف فرمول وزن‌دهی ارائه‌شده در (۳) ترکیب اطلاعات دقت طبقه‌بندها و توزیع کلاس داده‌های چانک جاری است. علاوه بر انتساب وزن به هر مؤلفه موجود در رویکرد جمعی، یک طبقه‌بند جدید h_{new} با داده‌های برچسب‌دار چانک جاری ایجاد می‌شود. از آنجایی که h_{new} با استفاده از جدیدترین داده‌ها آموزش دیده است همانند یک طبقه‌بند بی‌عیب و نقص با آن رفتار می‌شود و وزنی بر اساس (۴) به آن منسوب می‌شود. در مقایسه با تابع استفاده‌شده در وزن‌دهی مؤلفه‌های موجود در مجموعه، هنگام محاسبه وزن طبقه‌بند کاندید[3]، α_{new}، خطای پیش‌بینی h_{new} روی D_i در نظر گرفته نمی‌شود

$$\alpha_{new} = \frac{۱}{MSE_r + \varepsilon} \qquad (۴)$$

این رویکرد بر این فرض استوار است که جدیدترین بلاک، بهترین نماینده توزیع داده فعلی در آینده نزدیک است. از آنجایی که h_{new} با جدیدترین چانک داده آموزش دیده است با آن به عنوان بهترین طبقه‌بند ممکن رفتار می‌شود. اگر تعداد مؤلفه‌های موجود در مجموعه از k کمتر باشد (k ظرفیت مجموعه) طبقه‌بند کاندید h_{new} به مجموعه اضافه می‌شود و در غیر این صورت جایگزین ضعیف‌ترین مؤلفه موجود می‌شود.

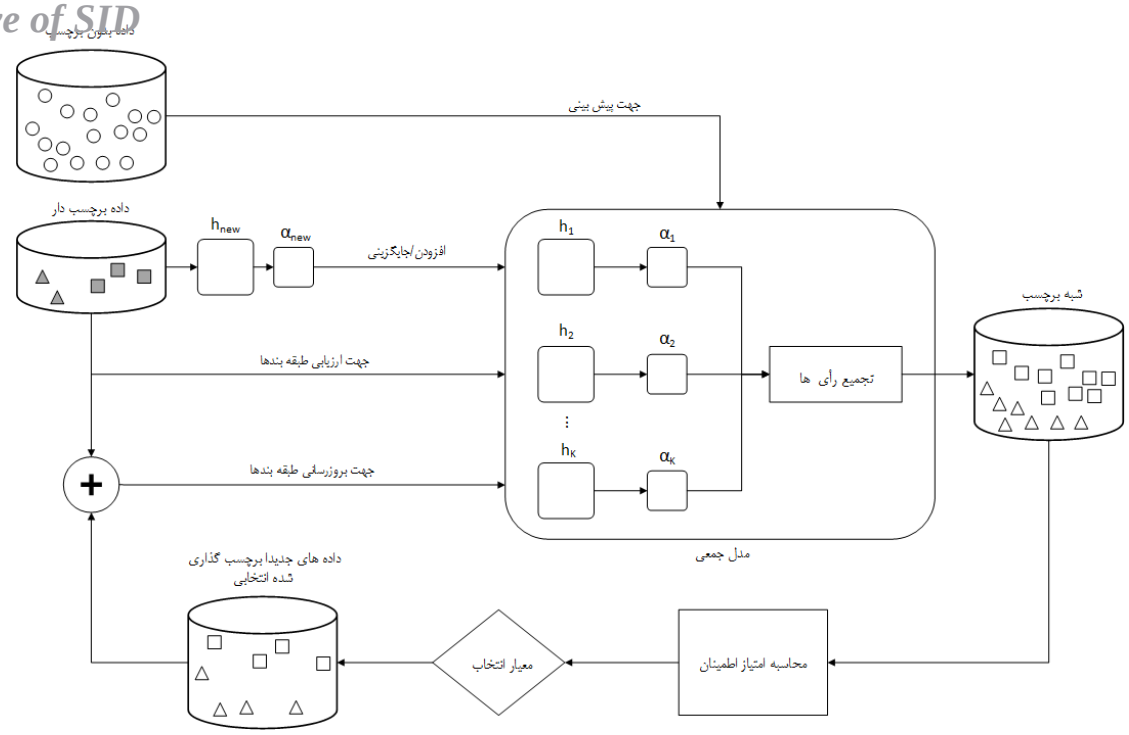
در مرحله بعد بر اساس رأی مؤلفه‌های موجود در مجموعه، به هر داده بدون برچسب یک شبه‌برچسب[4] منسوب (رابطه (۵)) و امتیاز اطمینان هر یک از آنها محاسبه می‌شود (رابطه (۶)). در (۵) با فرض آن که تعداد کلاس‌ها M است، $p_j(y_m|x)$ احتمال انتساب کلاس y_m به نمونه x توسط طبقه‌بند h_j را نشان می‌دهد و m_{MAP} بیانگر اندیس کلاس منسوب‌شده به نمونه بدون برچسب x توسط رأی اکثریت وزن‌دهی در مجموعه است. همچنین در (۶)، $Confidence(x)$ نحوه محاسبه امتیاز اطمینان کلاس منسوب‌شده به نمونه x را نشان می‌دهد که از نسبت احتمال کلاس اکثریت به همه کلاس‌ها توسط مجموعه مؤلفه‌های وزن‌دهی شده به دست می‌آید

$$m_{MAP} = \arg\max_{m \in M} \sum_{j=۱}^{K} \alpha_j p_j(y_m|x) \qquad (۵)$$

$$Confidence(x) = \frac{\sum_{j=۱}^{K} \alpha_j p_j(y_{m_{MAP}}|x)}{\sum_{m=۱}^{M} (\sum_{j=۱}^{K} \alpha_j p_j(y_m|x))} \qquad (۶)$$

سپس یک زیرمجموعه از داده‌های اخیراً برچسب‌گذاری شده با امتیاز اطمینان بزرگ‌تر از یک آستانه[5] انتخاب و به داده‌های برچسب‌دار چانک جاری اضافه می‌شوند. سپس با استفاده از این داده‌ها، مؤلفه‌های موجود در مجموعه به صورت افزایشی به روز رسانی می‌شوند. مدل کلی الگوریتم پیشنهادی در شکل ۱ ارائه شده است. کلیت الگوریتم پیشنهادی منطبق بر شکل ۱ در سه بخش خلاصه می‌شود: ارزیابی مدل‌های موجود، برچسب‌گذاری داده‌های بدون برچسب، انتخاب داده‌های جدیداً برچسب‌گذاری شده و به روز رسانی مدل‌های موجود. به این صورت که با

1. Aggregation
2. Base Classifiers (Learner)
3. Candidate Classifier
4. Pseudo-Label
5. Threshold



شکل ۱: مدل کلی الگوریتم پیشنهادی SSE-CBS.

Input: S : Data stream of examples partitioned into chunks such D, each containing L and U ;
L : Labeled data; U : Unlabeled data; E : Ensemble of k weighted incremental classifiers;

1. $E \leftarrow \varnothing$;
2. **For** all data chunks $D_i \in S$
3. $h_{new} \leftarrow$ new component classifier built on L_i ;
4. $\alpha_{new} = \frac{1}{MSE_r + \varepsilon}$ (4)
5. **For** all classifiers $h_j \in E$ do
6. - Apply h_j on L_i to derive MSE_{ij} ;
7. - Compute weight α_{ij} based on (3);
8. **End for**
9. **If** $|E| < k$ then
10. $E \leftarrow E \cup h_{new}$;
11. **Else** Replace h_{new} with the weakest classifier available in E ;
12. **End if**
13. **For** all $x \in U$ do
14. - Assign *pseudo-label* to x based on Ensemble prediction using (5);
15. - Measure *confidence* for assigned pseudo-label to x using (6);
16. **End For**
17. - Sample $P\%$ most confident pseudo-label examples that their Confidence greater than threshold $T : U' \leftarrow Select(U_i, P\%)$;
18. - Combine them with labeled samples in current chunk $L_i \leftarrow L_i \cup U'$;
19. **For** all classifiers $h_j \in E - \{h_{new}\}$
20. Incrementally update classifier h_j with L_i ;
21. **End for**
22. **End For**

Output $= Sign(\sum_{j=1}^{K} \alpha_j \times h_j)$

شکل ۲: شبه‌کد الگوریتم پیشنهادی SSE-CBS.

داده‌های برچسب‌دار چانک ورودی، یک مؤلفه طبقه‌بند جدید ایجاد شده و به مجموعه افزوده می‌شود، سپس با استفاده از همین داده‌های برچسب‌دار، مؤلفه‌های موجود ارزیابی و بر اساس خطای طبقه‌بندی به هر یک از آنها وزنی منسوب می‌شود. سپس بر اساس رأی اکثریت وزن‌دار مجموعه، به داده‌های بدون برچسب شبه‌برچسبی منسوب شده و امتیاز اطمینان هر یک محاسبه می‌شود. سپس درصدی از داده‌های جدیداً برچسب‌گذاری شده که حاوی اطلاعات مفیدی هستند انتخاب شده و به همراه داده‌های برچسب‌دار چانک جاری برای به روز رسانی مؤلفه طبقه‌بندهای موجود استفاده می‌شوند. این رویه با ورود هر چانک داده تکرار می‌شود.

الگوریتم SSE-CBS را می‌توان یک رویکرد نیمه‌نظارتی ترکیبی در نظر گرفت که در برابر تغییرات ناگهانی واکنش نشان می‌دهد و با مفاهیم در حال تغییر تدریجی نیز می‌تواند به تدریج هماهنگ شود. چون با ورود هر چانک داده طبقه‌بندها به روز می‌شوند، بنابراین قادر به مقابله با تغییر ناگهانی هستند. از طرف دیگر، مواجهه با تغییرات تدریجی به وسیله وزن‌دهی طبقه‌بندها بر اساس خطای پیش‌بینی و انتساب بیشترین وزن ممکن به جدیدترین طبقه‌بند حاصل می‌شود. شبه‌کد الگوریتم پیشنهادی در شکل ۲ ارائه شده است.

۳-٤ تحلیل پیچیدگی زمانی رویکرد پیشنهادی

در این بخش پیچیدگی زمانی الگوریتم پیشنهادی SSE-CBS تحلیل می‌گردد. در الگوریتم پیشنهادی با ورود هر چانک داده، یک طبقه‌بند ایجاد می‌شود، طبقه‌بندهای موجود ارزیابی می‌گردند و جدیدترین طبقه‌بند جایگزین ضعیف‌ترین مدل مجموعه می‌شود. از آنجا که حداکثر مدل‌های موجود در مجموعه برابر k است بنابراین هزینه زمانی این مرحله برابر $k+1$ خواهد بود. حال برچسب داده‌های بدون برچسب با رأی مدل‌های مجموعه پیش‌بینی می‌شوند که زمان اجرای آن بستگی به تعداد داده بدون برچسب موجود در چانک دارد و برابر است با $|U|$. سپس درصدی از آنها به همراه داده‌های برچسب‌دار چانک جاری جهت به روز رسانی

مدل‌های مجموعه استفاده می‌شوند که هزینه به روز رسانی آنها برابر k خواهد بود. این فرایند برای هر چانک تکرار می‌شود. بنابراین اگر مجموعه داده ورودی برابر n چانک باشد پیچیدگی زمانی الگوریتم حداکثر برابر است با

$$T(n) \in \mathrm{O}(n\times(k+1)+|U|+k) = \mathrm{O}(n\times(k+|U|)) \qquad (۷)$$

٤- نتایج

نتایج تجربی ارائه‌شده در این بخش به منظور ارزیابی عملکرد الگوریتم پیشنهادی است. ابتدا تنظیمات مربوط به آزمایش‌ها، سپس تعریف مجموعه داده‌ها و نهایتاً بصری‌سازی نتایج روی مجموعه داده‌ها ارائه می‌شود.

٤-١ تنظیمات

کارایی طبقه‌بندی الگوریتم پیشنهادی با ۸ الگوریتم HT، AUE-LL، AMANDA [۲۰]، COMPOSE [۲۱]، STDS [۲۴]، AUE۲ [۳۱]، ARF [۳۳] و SCo-F [۴۰] مقایسه شده است. الگوریتم AUE۲ و ARF الگوریتم‌های جمعی نظارتی روی چانک داده‌های تماماً برچسب‌دار هستند. الگوریتم AUE-LL یک الگوریتم جمعی نظارتی مشابه AUE۲ است با این تفاوت که فقط از درصد محدودی از داده‌های برچسب‌دار موجود در هر چانک استفاده می‌کند. الگوریتم SCo-F یک الگوریتم نیمه‌نظارتی جمعی است که الگوریتم Co-Forest سنتی را به منظور استفاده در جریان داده‌ها توسعه داده و از آشکارساز ADWIN برای برخورد با تغییر مفهوم استفاده می‌کند. در ساختار این الگوریتم نقایصی موجود است که با اعمال تغییرات و بهبود آن مورد استفاده قرار گرفته است. STDS یک الگوریتم نیمه‌نظارتی منفرد است که به یک آشکارساز تغییر مفهوم مجهز است. الگوریتم‌های COMPOSE و AMANDA الگوریتم‌های نیمه‌نظارتی منفرد تحت فرض تأخیر تأیید بی‌نهایت هستند. علاوه بر این، الگوریتم HT به عنوان مرجعی در مقایسات آمده که هیچ مکانیزم آشکارساز تغییر مفهومی در آن تعبیه نشده است.

جهت مقایسه معنادار و اصولی، در همه الگوریتم‌ها مقادیر برابری برای پارامترهای یکسان در نظر گرفته شده است. از جمله این که برای تمامی الگوریتم‌های جمعی تعداد مؤلفه‌ها (k) برابر ۱۰ در نظر گرفته شده است. برای یافتن تعداد مؤلفه مناسب در آزمایش‌های مقدماتی، مقدار این پارامتر بین ۵ تا ۳۰ مؤلفه مقداردهی و مقایسه شد که برای تعداد ۱۰ مؤلفه کارایی خوبی داشتند و با افزایش تعداد مدل‌ها زمان اجرا به شدت افزایش می‌یافت بدون آن که بهبود چشم‌گیری در دقت مدل جمعی ایجاد کند. در کلیه مجموعه داده‌ها اندازه چانک (D) برابر ۵۰۰ است، زیرا ۵۰۰ حداقل اندازه مناسب برای رویکردهای مبتنی بر بلوک است و چانک با اندازه کوچک‌تر به شدت دقت آنها را کاهش می‌دهد. در رویکرد پیشنهادی به علت به روز رسانی افزایشی، الگوریتم حداقل وابستگی را به اندازه چانک دارد و می‌تواند چانک‌های کوچک‌تری را پردازش کند بدون آن که کارایی‌اش کاهش یابد. در هر مجموعه داده، ۲۰ درصد داده‌ها به عنوان داده تست و ۸۰ درصد به عنوان داده آموزشی در نظر گرفته می‌شوند. در هر چانک تعداد ثابتی از داده‌ها بین ۲٪ تا ۱۰٪ برچسب‌دار هستند.

در الگوریتم‌های STDS، SCo-F و SSE-CBS در هر مرحله ۲۰٪ از داده‌های بدون برچسبی که با اطمینان بالاتر از آستانه ۹۰٪ برچسب‌گذاری شده‌اند به همراه شبه‌برچسبشان به داده‌های برچسب‌دار چانک جاری جهت به روز رسانی طبقه‌بندهای موجود در مجموعه اضافه می‌شوند. در الگوریتم پیشنهادی از درخت هافدینگ با تنظیمات پیش‌فرض MOA[1] به عنوان یادگیرنده پایه استفاده شده است. در MOA می‌توان انتخاب کرد که در برگ درخت هافدینگ از چه روش طبقه‌بندی استفاده شود، کلاس اکثریت[2] یا نایو بیز که در این مقاله از طبقه‌بند نایو بیز برای پیش‌بینی در برگ‌ها استفاده شده است. همچنین مدل‌های طبقه‌بندی با ۱۲ مجموعه داده ارزیابی شده‌اند.

٤-٢ مجموعه داده‌ها

در این مقاله با ابزار MOA [۴۲]، ۹ مجموعه داده مصنوعی SEA_S، SEA_F، Hyp_F، Hyp_S، RBF_B، RBF_GR، RBF_ND، LED_M و LED_ND را تولید کرده‌ایم و همچنین از سه مجموعه داده واقعی Elec، Cov و Poker موجود در پایگاه UCI[3] استفاده می‌کنیم. در جدول ۲ توضیح مختصری در مورد ویژگی‌های هر مجموعه داده ذکر شده است.

٤-٣ نتایج تجربی

در این بخش آزمایش‌های متعددی برای مقایسه عملکرد SSE-CBS با تعدادی از روش‌های نظارتی و نیمه‌نظارتی با استفاده از مجموعه داده‌های مختلفی انجام شده است. هدف از انجام این آزمایش‌ها، نشان‌دادن تأثیر استفاده از داده‌های بدون برچسب در بهبود عملکرد مدل طبقه‌بندی است.

برای بررسی حساسیت الگوریتم‌ها به تعداد داده‌های برچسب‌دار، تعدادی آزمایش با در نظر گرفتن نسبت‌های مختلف داده برچسب‌دار در هر چانک که بین ۲٪ تا ۱۰٪ می‌باشد انجام شده است، به جز در مورد الگوریتم‌های AUE۲ و ARF که چانک داده برای آنها تماماً برچسب‌دار است. انتظار می‌رود با افزایش تعداد داده‌های دارای برچسب، تفاوت بین الگوریتم‌های نظارتی و نیمه‌نظارتی کاهش یابد. نتایج ارزیابی‌ها در جداول ۳ تا ۵ نشان داده شده است. جداول از دو بخش تشکیل شده‌اند: الگوریتم‌های نظارتی و نیمه‌نظارتی. در بخش نیمه‌نظارتی دو ستون اول نتایج ارزیابی مدل‌های طبقه‌بندی HT و AUE-LL با استفاده از LL[4] (تعدادی محدود داده‌های برچسب‌دار) و دو ستون دوم نتایج ارزیابی مدل‌های AUE۲ و ARF را با استفاده از FL[5] (داده‌های تماماً برچسب‌دار) به ترتیب نشان می‌دهند. پنج ستون بخش نیمه‌نظارتی، نتایج الگوریتم‌های STDS، COMPOSE، AMANDA، SCO-F و SSE-CBS را با استفاده از داده‌های LU[6] (ترکیب داده‌های برچسب‌دار و درصدی از داده‌های بدون برچسب) نشان می‌دهد. در تمام آزمایش‌ها، برای مدل‌های موجود در بخش FL کل داده‌های چانک برچسب‌دار هستند و برای مدل‌های موجود در بخش‌های LL و LU، نسبت داده‌های برچسب‌دار در هر چانک بین ۲٪ تا ۱۰٪ متغیر است که به ترتیب در جداول ۳ تا ۵ نشان داده شده‌اند.

با توجه به جداول ۳ تا ۵ مشاهده می‌شود که در ترکیب درصدی از داده‌های بدون برچسب (به ترتیب ۱۰٪، ۵٪ و ۲٪) با داده‌های برچسب‌دار موجود، SSE-CBS عملکرد بهتری نسبت به سایر مدل‌های طبقه‌بند

1. Massive Online Analysis
2. Majority Class
3. https://archive.ics.uci.edu/ml/index.php
4. Limited Labeled Data
5. Fully Labeled
6. Labeled and Unlabeled

جدول ۲: ویژگی‌های مجموعه داده‌های استفاده‌شده.

مجموعه داده	تعداد نمونه	تعداد صفات	تعداد کلاس	درصد نویز	تعداد تغییر مفهوم	نوع تغییر
Hyp_s	۱ میلیون	۱۰	۲	٪۵	۱	افزایشی[1]
Hyp_F	۱ میلیون	۱۰	۲	٪۵	۱	افزایشی
RBF_B	۱ میلیون	۲۰	۴	٪۰	۲	افزایشی
RBF_GR	۱ میلیون	۲۰	۴	٪۰	۴	تدریجی[2]
RBF_ND	۱ میلیون	۲۰	۲	٪۰	۰	ندارد
SEA_S	۱ میلیون	۳	۲	٪۱۰	۳	ناگهانی[3]
SEA_F	۲ میلیون	۳	۲	٪۱۰	۹	ناگهانی
LED_M	۱ میلیون	۲۴	۱۰	٪۱۰	۲	ترکیبی (ناگهانی + تدریجی)
LED_ND	۱۰ میلیون	۲۴	۱۰	٪۲۰	۰	ندارد
Elec	۴۵۴۳۱	۸	۲	–	–	نامشخص
Cover Type	۵۸۱۱۳۹	۵۴	۷	–	–	نامشخص
Poker	۱ میلیون	۱۰	۱۰	–	–	نامشخص

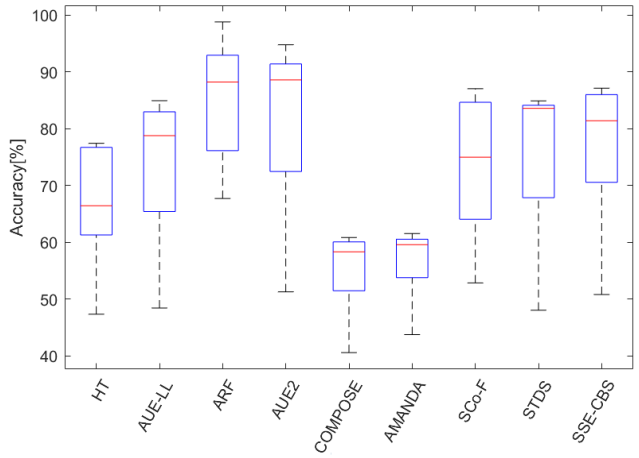
1. Incremental
2. Gradual
3. Sudden (Abrupt)

جدول ۳: میانگین دقت طبقه‌بندی برای ۱۰٪ داده برچسب‌دار در هر چانک.

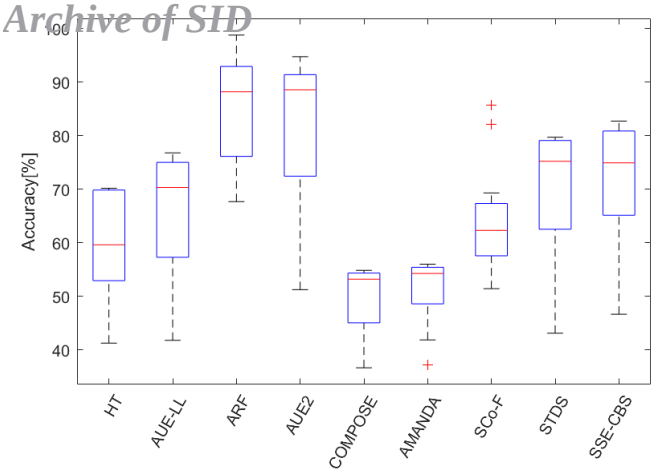
مجموعه داده	الگوریتم‌های نظارتی				الگوریتم‌های نیمه‌نظارتی				
	LL		FL		LU				
	HT	AUE-LL	ARF	AUE۲	COMPOSE	AMANDA	SCO-F	STDS	SSE-CBS
SEA_S	۷۷٫۴۲	۸۲٫۸۱	**۹۰٫۰۷**	۸۹٫۱۹	۵۹٫۷۳	۶۰٫۴۴	**۸۶٫۹۶**	۸۴٫۳۵	۸۶٫۰۹
SEA_F	۷۶٫۹۸	۸۳٫۳۹	**۸۹٫۶۶**	۸۸٫۷۲	۵۹٫۹۸	۶۰٫۰۳	**۸۷٫۰۴**	۸۳٫۶۵	۸۵٫۸۸
HYP_S	۷۷٫۳۱	۸۴٫۹۳	۸۵٫۴۷	**۸۸٫۴۳**	۵۸٫۲۸	۶۰٫۵۶	۷۸٫۰۲	۸۴٫۷۴	**۸۷٫۱۳**
HYP_F	۷۶٫۴۲	۸۳٫۱۲	۸۶٫۷۶	**۸۹٫۴۶**	۵۸٫۱۹	۵۹٫۱۱	۷۹٫۹۷	۸۴٫۸۹	**۸۶٫۵۳**
RBF_B	۶۸٫۳۲	۸۰٫۳۱	**۹۶٫۲۷**	۹۴٫۷۷	۶۰٫۸۲	۶۰٫۵۱	۸۵٫۲۲	۸۳٫۸۹	**۸۴٫۵۷**
RBF_GR	۶۴٫۵۴	۷۹٫۱۴	**۹۸٫۸**	۹۴٫۴۳	۶۰٫۵۶	۶۰٫۷۵	۸۴٫۰۸	۸۳٫۶۳	**۸۵٫۲۸**
RBF_ND	۶۹٫۸۲	۷۸٫۳۹	**۹۴٫۹۲**	۹۳٫۳۳	۵۸٫۳۴	۵۹٫۰۸	۷۱٫۹۴	۸۳٫۵۲	**۷۷٫۴۳**
LED_M	۶۱٫۲۵	۶۲٫۹	**۷۰٫۱۱**	۶۷٫۵۸	۴۶٫۶۲	۴۷٫۹۸	۵۲٫۸۳	۶۲٫۸۵	**۶۴٫۹۱**
LED_ND	۴۷٫۳۲	۴۸٫۴۱	**۶۷٫۷**	۵۱٫۲۶	۴۰٫۵۵	۴۳٫۷۲	**۵۵٫۶۸**	۴۸٫۰۲	۵۰٫۷۸
Elec	۶۱٫۳۱	۷۰٫۰۹	**۸۲٫۱۶**	۷۷٫۳۲	۶۰٫۱۵	۶۱٫۵۳	۶۲٫۳۶	۷۲٫۸۱	**۷۵٫۲۴**
COV	۶۱٫۶۲	۶۷٫۹۳	**۹۰٫۹۶**	۸۵٫۲۰	۵۱٫۸۲	۵۳٫۱۶	۶۶٫۰۵	۷۴٫۴۴	**۷۸٫۲۱**
Poker	۵۷٫۲۶	۶۰٫۷۷	**۶۸٫۸۲**	۶۶٫۱۰	۵۱٫۰۹	۵۴٫۳۶	۶۵٫۷۴	۶۱٫۳۲	**۶۵٫۸۵**

جدول ۴: میانگین دقت طبقه‌بندی برای ۵٪ داده برچسب‌دار در هر چانک.

???	الگوریتم‌های نظارتی				الگوریتم‌های نیمه‌نظارتی				
	LL		FL		LU				
	HT	AUE-LL	ARF	AUE۲	COMPOSE	AMANDA	SCO-F	STDS	SSE-CBS
SEA_S	۷۱٫۴۲	۷۸٫۴۳	**۹۰٫۰۷**	۸۹٫۱۹	۵۵٫۸۲	۵۶٫۹۲	**۸۶٫۲۹**	۸۰٫۵۴	۸۴٫۶۵
SEA_F	۷۱٫۲۱	۷۸٫۴۴	**۸۹٫۶۶**	۸۸٫۷۲	۵۵٫۹۱	۵۷٫۲۲	**۸۶٫۷۳**	۷۹٫۲۱	۸۴٫۲۳
HYP_S	۷۲٫۶۲	۷۹٫۴۴	۸۵٫۴۷	**۸۸٫۴۳**	۵۴٫۲۷	۵۶٫۴۱	۷۲٫۶۹	۸۱٫۶۴	**۸۵٫۵۲**
HYP_F	۷۲٫۴۱	۷۹٫۰۸	۸۶٫۷۶	**۸۹٫۴۶**	۵۵٫۵۳	۵۵٫۱۵	۷۴٫۸۴	۸۱٫۱۹	**۸۴٫۱۳**
RBF_B	۶۴٫۲۸	۷۶٫۸۶	**۹۶٫۲۷**	۹۴٫۷۷	۵۶٫۳۸	۵۷٫۳۲	**۸۴٫۲۴**	۷۹٫۱۱	۸۲٫۰۹
RBF_GR	۶۱٫۲۳	۷۴٫۶۶	**۹۸٫۸**	۹۴٫۴۳	۵۶٫۶۳	۵۶٫۸۷	**۸۳٫۳۵**	۷۹٫۸۹	۸۰٫۹۴
RBF_ND	۶۵٫۱۵	۷۳٫۰۲	**۹۴٫۹۲**	۹۳٫۳۳	۵۴٫۷۸	۵۶٫۴۲	۷۰٫۸۹	۸۰٫۸۲	**۷۲٫۲۱**
LED_M	۵۴٫۴۵	۵۶٫۵۵	**۷۰٫۱۱**	۶۷٫۵۸	۴۲٫۶۶	۴۴٫۱۶	۵۲٫۳	۶۰٫۶۲	**۶۱٫۷۸**
LED_ND	۴۲٫۲۳	۴۲٫۲۳	**۶۷٫۷**	۵۱٫۲۶	۳۷٫۰۳	۳۹٫۰۴	۵۵٫۵۹	۴۴٫۱۳	**۴۸٫۸۸**
Elec	۵۸٫۰۸	۶۳٫۳۴	**۸۲٫۱۶**	۷۷٫۳۲	۵۶٫۶۸	۵۷٫۷۷	۶۲٫۰۷	۶۸٫۸۷	**۷۲٫۵۵**
COV	۵۷٫۷۸	۶۱٫۶۶	**۹۰٫۹۶**	۸۵٫۲۰	۴۵٫۴۷	۵۱٫۳۵	۶۵٫۷۹	۶۹٫۲۳	**۷۵٫۱۲**
Poker	۵۴٫۸۷	۵۷٫۳۹	**۶۸٫۸۲**	۶۶٫۱۰	۴۸٫۲۱	۴۹٫۶۷	**۶۵٫۳۹**	۵۹٫۱۲	۶۳٫۹۳



شکل ۳: نمودار جعبه‌ای دقت الگوریتم‌ها روی تمام مجموعه داده‌ها با ۱۰٪ داده برچسب‌دار.



شکل ۴: نمودار جعبه‌ای دقت الگوریتم‌ها روی تمام مجموعه داده‌ها با ۲٪ داده برچسب‌دار.

جدول ۵: میانگین دقت طبقه‌بندی برای ۲٪ داده برچسب‌دار در هر چانک.

مجموعه داده	الگوریتم‌های نظارتی				الگوریتم‌های نیمه‌نظارتی				
	LL		FL		LU				
	HT	AUE-LL	ARF	AUE۲	COMPOSE	AMANDA	SCO-F	STDS	SSE-CBS
SEA_S	۶۹٫۹۲	۷۶٫۱۱	**۹۰٫۰۷**	۸۹٫۱۹	۵۴٫۱۱	۵۴٫۳۳	۸۲٫۱۷	۷۹٫۵۸	**۸۲٫۴۶**
SEA_F	۷۰٫۱۸	۷۴٫۶۱	**۸۹٫۶۶**	۸۸٫۷۲	۵۴٫۰۵	۵۴٫۷۸	**۸۵٫۷۱**	۷۸٫۱۱	۸۲٫۷۲
HYP_S	۶۹٫۷۸	۷۶٫۷۹	۸۵٫۴۷	**۸۸٫۴۳**	۵۲٫۷۷	۵۵٫۶۱	۵۸٫۱۶	۷۹٫۷۳	**۸۱٫۱۴**
HYP_F	۷۰٫۱۹	۷۵٫۴۵	۸۶٫۷۶	**۸۹٫۴۶**	۵۳٫۶۵	۵۳٫۷۶	۵۸٫۶۱	۷۹٫۶۵	**۸۰٫۳۶**
RBF_B	۶۱٫۵۵	۷۱٫۸۶	**۹۶٫۲۷**	۹۴٫۷۷	۵۴٫۸۶	۵۶٫۰۱	۶۹٫۳۲	۷۸٫۶۳	**۸۰٫۶۳**
RBF_GR	۵۷٫۷۲	۷۰٫۹۶	**۹۸٫۸**	۹۴٫۴۳	۵۴٫۶۱	۵۵٫۲۸	۶۵٫۰۳	۷۶٫۲۸	**۸۰٫۰۱**
RBF_ND	۶۲٫۳۲	۶۹٫۷۳	**۹۴٫۹۲**	۹۳٫۳۳	۵۲٫۰۲	۵۴٫۲۱	۵۷٫۰۱	۷۴٫۱۹	**۶۹٫۴۶**
LED_M	۵۳٫۱۶	۵۵٫۴۶	**۷۰٫۱۱**	۶۷٫۵۸	۴۰٫۴۷	۴۱٫۸۸	۵۱٫۴۴	۵۷٫۴۲	**۵۸٫۹۷**
LED_ND	۴۱٫۲۷	۴۱٫۷۹	**۶۷٫۷**	۵۱٫۲۶	۳۶٫۶۶	۳۷٫۱۵	**۵۲٫۸۵**	۴۳٫۱۳	۴۶٫۶۷
Elec	۵۳٫۱۴	۶۱٫۶۵	**۸۲٫۱۶**	۷۷٫۳۲	۵۴٫۷۱	۵۵٫۶۷	۶۱٫۰۷	۶۷٫۶۸	**۶۹٫۸۷**
COV	۵۲٫۷۳	۵۹٫۱۶	**۹۰٫۹۶**	۸۵٫۲۰	۴۴٫۱۹	۴۸٫۶۲	۶۵٫۳۸	۶۸٫۸۲	**۶۹٫۳۶**
Poker	۵۲٫۶۱	۵۳٫۸۹	**۶۸٫۸۲**	۶۶٫۱۰	۴۵٫۹۲	۴۸٫۶۱	**۶۳٫۵۸**	۵۷٫۱۷	۶۰٫۹۵

نیمه‌نظارتی داشته و عملکرد آن تقریباً نزدیک به مدل‌های طبقه‌بند آموزش‌دیده با داده‌های تماماً برچسب‌دار است. همچنین با استفاده از نتایج تجربی مشاهده می‌شود که SSE-CBS به طور قابل توجهی عملکرد الگوریتم نظارتی AUE-LL را در همه ۱۲ مجموعه داده بهبود می‌بخشد. بهترین نتایج در هر دو بخش نظارتی و نیمه‌نظارتی پررنگ شده‌اند. به منظور مقایسه سریع‌تر، نتایج موجود در جداول ۳ تا ۵ به صورت نمودار جعبه‌ای[1] به ترتیب در شکل‌های ۳ تا ۵ نشان داده شده‌اند.

علاوه بر این در شکل ۶- الف تا ۶- د، دقت الگوریتم‌ها را روی برخی از مجموعه داده‌ها برای حالتی که تعداد داده‌های برچسب‌دار در هر چانک ۱۰٪ است به صورت گرافیکی نشان داده‌ایم. در هر نمودار عملکرد الگوریتم‌ها در محور Y و تعداد نمونه‌ها در محور X نشان داده شده و به منظور خوانایی بهتر فقط تعدادی از الگوریتم‌ها در نمودار آمده‌اند.

شکل ۶- الف دقت الگوریتم‌ها را روی مجموعه داده SEA_S که حاوی تغییر مفهوم ناگهانی است نشان می‌دهد. تغییرات بعد از هر ۲۵۰۰۰۰ نمونه رخ می‌دهند. در نمودار ارائه‌شده، به وضوح قابل درک است که افت دقت حوالی نمونه‌های ۲۵۰ هزار، ۵۰۰ هزار و ۷۵۰ هزار نشان‌دهنده تغییر مفهوم است.

شکل ۶- ب دقت تعدادی از الگوریتم‌ها را روی مجموعه داده Hyp_F نشان می‌دهد که حاوی تغییر مفهوم افزایشی است. در این مجموعه داده، الگوریتم‌های AUE۲ و SSE_CBS بهترین عملکرد را دارند.

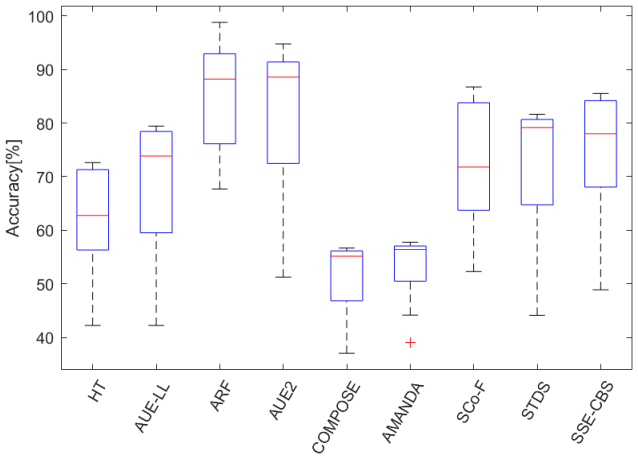
شکل ۶- ج نمودار دقت الگوریتم‌ها را روی مجموعه داده Cov نشان می‌دهد. نمودار دقت نشان می‌دهد که بعد از AUE۲ الگوریتم SSE-CBS بهترین کارایی را دارد. همچنین با توجه به کارایی الگوریتم‌های طبقه‌بندی می‌توان دریافت که مجموعه داده تحلیل‌شده احتمالاً حاوی تغییر مفهوم است.

سرانجام، شکل ۶- د نمودار دقت را روی مجموعه داده LED_M نشان می‌دهد. در این مجموعه داده، تغییر مفهومی پیچیده با ترکیب دو تغییر مفهوم تدریجی تولید شده است. بعد از ۵۰۰ هزار نمونه، مفاهیم به طور ناگهانی تعویض می‌شوند اما مواجهه با تغییرات تدریجی در مفهوم جدید بسیار دشوار است. با این حال الگوریتم‌های SSE-CBS و AUE۲ بهترین عملکرد را نسبت به بقیه الگوریتم‌ها دارند.

۵- نتیجه‌گیری

در این مقاله، یک الگوریتم نیمه‌نظارتی جمعی ترکیبی تحت عنوان SSE_CBS برای مقابله با انواع تغییر مفهوم‌ها در محیط‌های با کمبود داده برچسب‌دار ارائه شده است. الگوریتم پیشنهادی با هشت الگوریتم

1. Box Plot



شکل ۵: نمودار جعبه‌ای دقت الگوریتم‌ها روی تمام مجموعه داده‌ها با ۵٪ داده برچسب‌دار.

از جدیدترین روش‌های موجود درحوزه جریان‌کاوی مقایسه شده است که متشکل از طبقه‌بندهای منفرد و مدل‌های جمعی نظارتی و نیمه‌نظارتی است. ما مجموعه کاملی از نتایج را روی داده‌های مصنوعی و داده‌های واقعی ارائه داده‌ایم. نتایج تجربی نشان می‌دهند که روش ارائه‌شده نسبت به روش‌های نیمه‌نظارتی و بیشتر رویکردهای نظارتی بهتر عمل کرده است، چرا که قادر است از ترکیب مزایای رویکردهای مبتنی بر چانک و برخط در مواجهه با انواع تغییر مفهوم به خوبی استفاده کند. همچنین مشاهده می‌شود که روش پیشنهادی زمانی که فقط ۲٪ داده دارای برچسب در هر چانک وجود دارد، عملکرد خوبی دارد و بیانگر این است که الگوریتم پیشنهادی توانسته از اطلاعات داده‌های بدون برچسب به خوبی بهره برد. به عنوان کار آینده، قصد داریم معیار انتخاب داده‌های بدون برچسب در الگوریتم SSE_CBS را بهبود بخشیم به گونه‌ای که از اطلاعات نهفته در داده‌های فاقد برچسب بهره برده و قادر به انتخاب داده‌هایی باشد که عملکرد طبقه‌بند را بهبود بخشند.

مراجع

[1] S. Krishnaswamy, J. Gama, and M. M. Gaber, "Mobile data stream mining: from algorithms to applications," in *Proc. IEEE 13th Int. Conf. Mob. Data Manag.*, pp. 360-363, Bengaluru, India, 23-26 Jul. 2012.

[2] M. M. Gaber, J. Gama, S. Krishnaswamy, J. B. Gomes, and F. Stahl, "Data stream mining in ubiquitous environments: state-of-the-art and current directions," *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.*, vol. 4, no. 2, pp. 116-138, Feb. 2014.

[3] H. L. Nguyen, Y. K. Woon, and W. K. Ng, "A survey on data stream clustering and classification," *Knowl. Inf. Syst.*, vol. 45, no. 3, pp. 535-569, Dec. 2015.

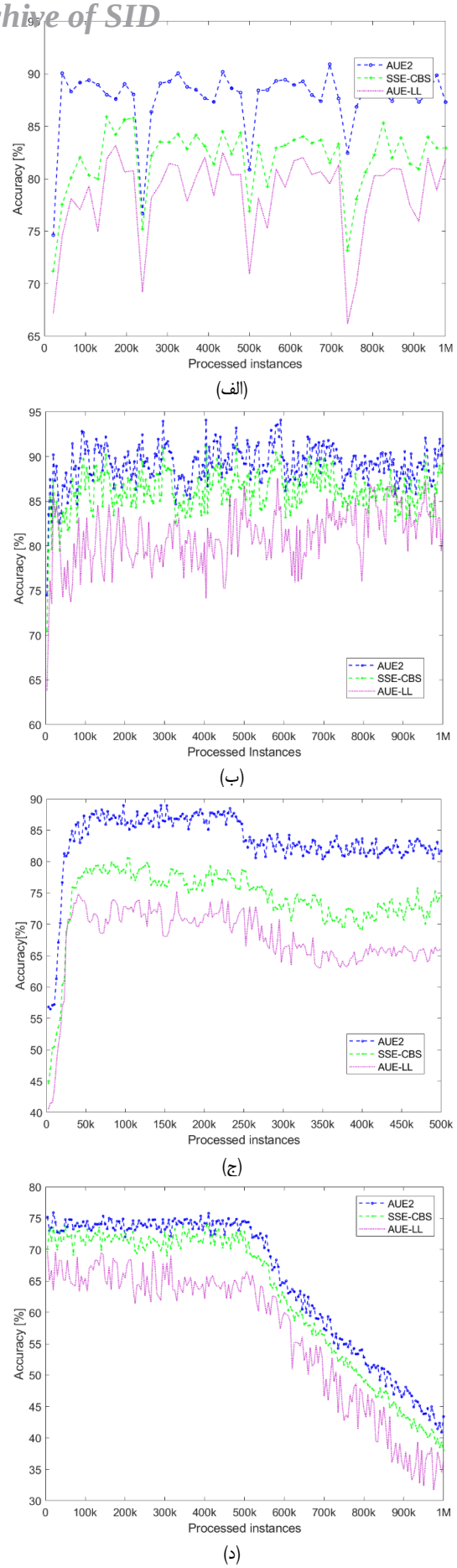
[4] S. Ramirez-Gallego, B. Krawczyk, S. Garcia, M. Wozniak, and F. Herrera, "A survey on data preprocessing for data stream mining: current status and future directions," *Neurocomputing*, vol. 239, pp. 39-57, May 2017.

[5] S. Wares, J. Isaacs, and E. Elyan, "Data stream mining: methods and challenges for handling concept drift," *SN Appl. Sci.*, vol. 1, no. 11, pp. 1-19, Nov. 2019,.

[6] H. M. Gomes, N. Zealand, and A. Bifet, "Machine learning for streaming data : state of the art, challenges, and opportunities," *ACM SIGKDD Explorations Newsletter*, vol. 21, no. 2, pp. 6-22, Nov. 2019.

[7] C. Woolam, M. M. Masud, and L. Khan, "Lacking labels in the stream: classifying evolving stream data with few labels," in *Proc. Int. Symp. on Methodologies for Intelligent Systems,* pp. 552-562, Sept. 2009.

[8] M. M. Masud, *et al.*, "Facing the reality of data stream classification: coping with scarcity of labeled data," *Knowledge and Information Systems*, vol.33, no. 1, pp. 213-244, Oct. 2012.



شکل ۶: ارزیابی دقت طبقه‌بندی با ۱۰٪ داده برچسب‌دار در هر چانک روی مجموعه داده‌های مختلف، (الف) مجموعه داده SEA_S، (ب) مجموعه داده Hyp_F، (ج) مجموعه داده Cov و (د) مجموعه داده LED_M.

Neural Networks and Learning Systems, vol. 25, no. 1, pp. 81-94, Jan. 2014.
[32] D. Brzezinski and J. Stefanowski, "Accuracy updated ensemble for data streams with concept drift," in Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. LNAI 6679, pt. 2, pp. 155-163, 2011.
[33] H. M. Gomes, *et al.*, "Adaptive random forests for evolving data stream classification," *Mach. Learn.*, vol. 106, no. 9-10, pp. 1469-1495, Oct. 2017.
[34] R. Klinkenberg, "Learning drifting concepts: example selection vs. example weighting," *Intell. Data Anal.*, vol. 8, no. 3, pp. 281-300, Aug. 2004.
[35] D. Kifer, S. Ben-David, and J. Gehrke, "Detecting change in data streams," in *Proc. 30th Int. Conf. Very Large Data Bases Conf.*, pp. 180-191, Aug. 2004,.
[36] A. Bifet and R. Gavald, "Learning from Time-Changing Data with Adaptive Windowing a".
[37] A. Bifet, *et al.*, "Early drift detection method," in *Proc. 4th ECML PKDD Int. Work. Knowl. Discov. from Data Streams*, vol. 6, pp. 77-86, 2006.
[38] M. M. Masud, J. Gao, L. Khan, J. Han, and B. Thuraisingham, "A practical approach to classify evolving data streams: training with limited amount of labeled data," in *Proc. IEEE Int. Conf. Data Mining, ICDM'08*, pp. 929-934, Pisa, Italy, 15-19 Dec. 2008.
[39] M. J. Hosseini, A. Gholipour, and H. Beigy, "An ensemble of cluster-based classifiers for semi-supervised classification of non-stationary data streams," *Knowl. Inf. Syst.*, vol. 46, pp. 567-597, 2016.
[40] Y. Wang and T. Li, "Improving semi-supervised co-forest algorithm in evolving data streams," *Applied Intelligence*, vol. 48, pp. 3248-3262, 2018
[41] M. Li and Z. H. Zhou, "Improve computer-aided diagnosis with machine learning techniques using undiagnosed samples," *IEEE Trans. Syst. Man, Cybern. A Syst. Humans*, vol. 37, no. 6, pp. 1088-1098, Nov. 2007.
[42] A. Bifet, *et al.*, "MOA: massive online analysis, a framework for stream classification and clustering," *HaCDAIS*, vol. 11, pp. 44-51, , Aachen, Germany, Sept.2010.

شیرین خضری مدرک کارشناسی خود را در رشته علوم کامپیوتر در سال ۱۳۸۶ از دانشگاه تبریز و مدارک کارشناسی ارشد و دکتری را در رشته هوش مصنوعی به ترتیب در سال‌های ۱۳۸۹ و ۱۳۹۹ از دانشگاه آزاد اسلامی واحد قزوین و دانشگاه آزاد اسلامی واحد علوم و تحقیقات تهران اخذ کرده است. در حال حاضر عضو هیأت علمی دانشگاه پیام نور است و زمینه‌های پژوهشی مورد علاقه ایشان یادگیری ماشین و جریان‌کاوی است.

جعفر تنها مدرک دکتری و پسادکتری خود را به ترتیب در سال‌های ۱۳۹۲ و ۱۳۹۴ از دانشگاه آمستردام هلند در رشته هوش مصنوعی اخذ نموده و هم‌اکنون عضو هیأت علمی دانشگاه تبریز می‌باشد. حوزه‌های پژوهشی مورد علاقه ایشان داده‌کاوی، جریان‌کاوی و یادگیری ماشین است.

علی احمدی مدرک پسادکتری خود را در سال ۱۳۸۶ از دانشگاه هیروشیمای ژاپن، مدارک دکتری و کارشناسی ارشد خود را به ترتیب در سال‌های ۱۳۸۳ و ۱۳۸۰ از دانشگاه اوساکای ژاپن و مدرک کارشناسی خود را در سال ۱۳۶۹ از دانشگاه صنعتی امیرکبیر اخذ کرد و هم‌اکنون عضو هیأت علمی گروه مهندسی کامپیوتر دانشگاه صنعتی خواجه نصیرالدین طوسی است. ایشان در سال‌های اخیر در حوزه‌های پژوهشی مرتبط با یادگیری تعاملی، داده‌کاوی، موتور جستجوی تصویر، واقعیت مجازی و پردازش تصویر فعالیت می‌کنند.

آرش شریفی مدرک کارشناسی خود را در رشته مهندسی کامپیوتر گرایش سخت‌افزار از دانشگاه آزاد اسلامی واحد تهران‌جنوب و مدارک کارشناسی ارشد و دکتری خود را به ترتیب در سال‌های ۱۳۸۶ و ۱۳۹۱ از دانشگاه آزاد اسلامی واحد علوم و تحقیقات تهران اخذ کرده و هم‌اکنون عضو هیأت علمی دانشکده مهندسی کامپیوتر دانشگاه آزاد اسلامی واحد علوم و تحقیقات تهران می‌باشد. زمینه‌های پژوهشی فعلی ایشان پردازش تصویر، یادگیری ماشین و یادگیری عمیق است.

[9] A. Bifet and R. Gavalda, "Learning from time-changing data with adaptive windowing," in *Proc. 7th SIAM Int. Conf. Data Min.*, vol. 5, pp. 443-448, Minneapolis, MN, USA, Apr, 2007.
[10] A. Bifet, G. Holmes, B. Pfahringer, R. Kirkby, and R. Gavalda, "New ensemble methods for evolving data streams," in *Proc. of the ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, pp. 139-147, Paris, France, Jun. 2009.
[11] J. Gama, P. Medas, G. Castillo, and P. Rodrigues, "Learning with drift detection," in *Proc. Brazilian Symp. Artif. Intell.*, pp. 286-295, Sao Luis, Brazil, 29 Sept.-1 Oct. 2004.
[12] H. Wang, W. Fan, P. Yu, and J. Han, "Mining concept-drifting data streams using ensemble classifiers," in *Proc. 9th ACM SIGKDD Int. Conf. Knowl. Discov. Data Min. - KDD'03*, vol. 2, pp. 226-235, Washington, D.C., USA, Aug. 2003.
[13] N. C. Oza and S. Russell, "Experimental comparisons of online and batch versions of bagging and boosting," in *Proc. 7th ACM SIGKDD Int. Conf. Knowl. Discov. Data Min. - KDD'01*, pp. 359-364, San Francisco, CA, Aug. 2001.
[14] W. N. Street and Y. Kim, "A streaming ensemble algorithm (SEA) for large-scale classification," in *Proc. 7th ACM SIGKDD Int. Conf. Knowl. Discov. Data Min. - KDD'01*, pp. 377-382, Aug. 2001.
[15] A. Bifet, G. Holmes, and B. Pfahringer, "Leveraging bagging for evolving data streams," in Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics, vol. LNAI 6321, pt. 1, pp. 135-150, Barcelona, Spain, 19-23 Sept. 2010.
[16] J. Tanha, M. Van Someren, and H. Afsarmanesh, "Boosting for multiclass semi-supervised learning," *Pattern Recognit. Lett.*, vol. 37, no. 1, pp. 63-77, Feb. 2014.
[17] J. Tanha, M. Van Someren, and H. Afsarmanesh, "Semi-supervised self-training for decision tree classifiers," *Int. J. Mach. Learn. Cybern.*, vol. 8, pp. 355-370, Jan. 2017.
[18] A. Blum and T. Mitchell, "Combining labeled and unlabeled data with co-training," in *Proc. Elev. Annu. Conf. Comput. Learn. Theory, COLT'98*, pp. 92-100, Madison, WI, USA, Jul. 1998.
[19] J. Tanha, "MSSBoost: a new multiclass boosting to semi-supervised learning," *Neurocomputing*, vol. 314, pp. 251-266, Nov. 2018.
[20] R. S. Ferreira, G. Zimbrao, and L. G. M. Alvim, "AMANDA: semi-supervised density-based adaptive model for non-stationary data with extreme verification latency," *Inf. Sci.*, vol. 488, pp. 219-237, Jul. 2019.
[21] K. B. Dyer, R. Capo, and R. Polikar, "Compose: a semisupervised learning framework for initially labeled nonstationary streaming data," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 25, no. 1, pp. 12-26, Jan. 2014.
[22] J. Gama, I. Zliobaite, A. Bifet, M. Pechenizkiy, and A. Bouchachia, "A survey on concept drift adaptation," *ACM Comput. Surv.*, vol. 46, no. 4, Article No. 44, 37 pp., Mar. 2014.
[23] M. Althabiti and M. Abdullah, "Streaming data classification with concept drift," *Biosci. Biotech. Res. Comm. Special Issue*, vol. 12, no. 1, pp. 177-184, Jan. 2019.
[24] S. Khezri, J. Tanha, A. Ahmadi, and A. Sharifi, "STDS: self-training data streams for mining limited labeled data in non-stationary environment," *Appl. Intell.*, vol. 50, pp. 1448-1467, Jan. 2020.
[25] P. Domingos and G. Hulten, "Mining high-speed data streams," in *Proc. of the 6th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, *KDD'00*, pp. 71-80, Aug. 2000.
[26] G. Hulten, L. Spencer, and P. Domingos, "Mining time-changing data streams," in *Proc. of the 7th ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, *KDD'01*, pp. 97-106, Aug. 2001.
[27] J. Kolter and M. Maloof, "Dynamic weighted majority: an ensemble method for drifting concepts," *J. Mach. Learn. Res.*, vol. 8, pp. 2755-2790, 2007.
[28] R. Kirkby, *Improving Hoeffding Trees*, PhD Thesis, Department of Computer Science, the University of Waikato, 2007.
[29] R. Elwell and R. Polikar, "Incremental learning of concept drift in nonstationary environments," *IEEE Trans. Neural Netw.*, vol. 22, no. 10, pp. 1517-1531, Oct. 2011.
[30] K. Nishida, K. Yamauchi, and T. Omori, "ACE: adaptive classifiers-ensemble system for concept-drifting environments," in *Proc. of the 6th Int. Conf. on Multiple Classifier Systems*, *MCS'05,* pp. 176-185, Jun. 2005.
[31] D. Brzezinski and J. Stefanowski, "Reacting to different types of concept drift: the accuracy updated ensemble," *IEEE Trans. on*