فصلنامه علمی - پژوهشی فرماندهی و کنترل، سال ششم، شماره دو، تابستان 1401

یادگیری تجمعی مبتنی بر تابع ضرر کانونی افزایشی (FENIL) برای غلبه بر نامتوازنی دسته‌ای

نسیبه محمودی[1]، حسین شیرازی[2]، محمد فخردانش[3]، کوروش داداش تبار[4]*

تاریخ دریافت: 1401/04/21 **تاریخ پذیرش:** 1401/06/26

چکیده

شبکه‌های عصبی کانولووشنال یکی از موفق‌ترین و پراستفاده‌ترین مدل‌های یادگیری ماشین در دسته‌بندی داده‌ها محسوب می‌شود اما به رغم موفقیت‌های چشمگیری که در دسته‌بندی داده‌ها دارند، در یادگیری نامتوازن، که یکی از چالش‌برانگیزترین مشکلات در یادگیری ماشین است، به نتایج قابل قبولی دست پیدا نمی‌کنند چرا که در این گونه مسائل، معمولا تعداد نمونه‌های یکی از دسته‌ها خیلی بیشتر از نمونه‌های دسته دیگر است و یا هزینه دسته‌بندی اشتباه در دو دسته متفاوت است، این در حالی است که شبکه‌های **CNN** به صورت پیش‌فرض، توزیع دسته‌ها را متوازن و هزینه دسته‌بندی را مساوی در نظر می‌گیرند. یکی از روش‌های موفق در برخورد با مجموعه‌داده‌های نامتوازن، روش‌های تجمعی‌است. آنها با ترکیب تعدادی از تخمین‌گرهای پایه می‌توانند به دقت بالایی دست پیدا کنند و در مقایسه با زمانی که تنها از یک تخمین‌گر استفاده می‌شود، قابلیت اطمینان مدل را افزایش دهند. استفاده از یادگیری تجمعی، مدل‌های یادگیری ماشین را در مواجهه با داده‌های نامتوازن توانمند می‌سازند. در این پژوهش، روشی مبتنی بر یادگیری تجمعی برای شبکه‌های عصبی کانولووشنال معرفی شده‌است که از تجمع تعدادی شبکه **CNN** برای کار با داده‌های نامتوازن استفاده می‌کند. در این مدل از تابع ضرر کانونی برای آموزش **CNN**ها استفاده‌شده‌است، پارامتر گاما در این تابع میزان اهمیت نمونه‌های سخت و آسان را مشخص می‌کند در مدل تجمعی پیشنهادی از پارامتر گاما برای ایجاد تنوع در **CNN**ها استفاده‌شده‌است و این باعث شده‌است هر شبکه کانولووشنال نسبت به شبکه قبلی اهمیت کمتری به داده‌های آسان دهد. همچنین وزن داده‌ها برای آموزش هر شبکه با استفاده از نتیجه دسته‌بندی شبکه **CNN** قبلی مشخص می‌شود. در نهایت برای دسته بندی داده‌های جدید از ترکیب نتیجه همه **CNN**ها استفاده می‌شود. شبکه تجمعی یشنهادی (**FENIL**) برروی چندین مجموعه داده اعمال شده است، براساس نتایج بدست آمده، شبکه **FENIL** نه تنها درمقایسه با روش‌های غیر عمیق مثل آدابوست با درخت تصمیم، دقت و **F1-score** بسیار بالاتری (18/63، 61/ 19بالاتر) دارد، بلکه در مقایسه با روش‌های معمول عمیق دیگر مانند استفاده از یک **CNN** عمیق، رای گیری **CNN**ها و **CNN**های آبشاری و **SMOTE** نیز نتایج بهتری را بدست‌آورده‌است.

واژگان کلیدی: یادگیری نامتوازن، تابع ضرر کانونی، یادگیری تجمعی، یادگیری انتقالی

[1] دانشگاه مالک اشتر، دانشکده برق و کامپیوتر Mahmoodi.nm@mut.ac.ir

[2] دانشگاه مالک اشتر، دانشکده برق و کامپیوتر shirazi@mut.ac.ir (نویسنده مسئول)

[3] دانشگاه مالک اشتر، دانشکده برق و کامپیوتر fakhredanesh@mut.ac.ir

[3] دانشگاه مالک اشتر، دانشکده برق و کامپیوتر dadashtabar@mut.ac.ir

۱. مقدمه

در یادگیری نامتوازن با مسائلی روبرو هستیم که دسته‌ها، احتمال پیشین متفاوت دارند، یعنی تعداد نمونه‌های آموزشی برای دسته‌ها به طور قابل توجهی متفاوت است. به دسته با نمونه‌های آموزشی کم، دسته اقلیت و به دسته با تعداد زیاد نمونه آموزشی دسته اکثریت گفته می‌شود. دسته‌بندی یا پیش‌بینی نامتوازن ممکن است هرجایی اتفاق بیفتد و به طورگسترده در مسائلی که الگوریتم‌های یادگیری ماشین بکار می‌روند، چه در کاربردهای زندگی واقعی و چه در تحلیل داده‌های علمی، وجود دارند، که می‌توان از این میان به تشخیص تقلب [1]، شناسایی خطا [2, 3]، تشخیص ناهنجاری [4] ، تشخیص پزشکی [5, 6]، تشخیص نشت نفت [7] ، تشخیص بدافزار [8]، تحلیل رفتار[9] و غیره اشاره کرد. در الگوریتم‌های استاندارد، توزیع دسته‌ها متوازن و هزینه دسته‌بندی مساوی در نظر گرفته می‌شود، بنابراین مجموعه داده‌های نامتوازن می‌توانند کارایی الگوریتم‌های استاندارد یادگیری ماشین را به شدت تحت تاثیر قرار دهند و حتی با وجود اهمیت بیشتر، تا حد زیادی داده‌های اقلیت نادیده گرفته شوند و آموزش به سمت داده‌های دسته اکثریت تمایل پیدا کند [10]، فرض کنید در مجموعه داده تشخیص کوید19 فقط 10 درصد مبتلا وجود داشته باشد، الگوریتمی که همه نمونه‌ها را سالم تشخیص دهد (دسته اکثریت) می‌تواند به دقت 90 درصد برسد در حالی که هیچ یک از مبتلایان تشخیص داده نشده‌اند، این مثال ضرورت ارائه راهکارها و معیارهای کارایی مناسب برای مواجهه با نامتوازنی دسته‌ای را به خوبی نشان می‌دهد و به همین دلیل یادگیری نامتوازن بسیار مورد توجه محققان و پژوهشگران هوش مصنوعی قرار گرفته است. شبکه‌های عصبی کانولووشنال، به عنوان یکی از شاخه‌های موفق یادگیری ماشین، که درسال‌های اخیر، پیشرفت‌های چشمگیری در زمینه‌های گوناگون داشته اند، مشکلات اجداد خود یعنی شبکه‌های عصبی سنتی در برخورد با داده‌های نامتوازن را به ارث برده‌اند. آناند و همکاران [11] که تأثیرات نامتوازنی دسته‌ای در الگوریتم انتشار رو به عقب[2] در شبکه‌های عصبی را بررسی کردند، نشان دادند که در سناریوهای دسته نامتوازن، طول مولفه گرادیان دسته اقلیت بسیار کمتر از مولفه گرادیان دسته اکثریت است. به عبارت دیگر، دسته اکثریت اساساً بر گرادیانی که وزن مدل را به روزرسانی می‌کند، تسلط دارد. این موضوع در تکرارهای اولیه خیلی سریع خطای گروه اکثریت را کاهش می‌دهد، اما اغلب خطای گروه اقلیت را افزایش می‌دهد و باعث می‌شود شبکه در یک حالت همگرایی کند گیر کند. این مشکل همچنان در شبکه‌های عمیق نیز وجود دارد، و محققان راهکارهای مختلفی برای مقابله با آن ارائه داده‌اند، راه حل‌ها برای برخورد با مساله نامتوازنی دسته‌ای در شبکه‌های عمیق، به سه دسته تقسیم می‌شوند:

1- راه حل‌های" مبتنی بر داده" [12] تلاش دارند با کم یا زیاد کردن نمونه‌های برخی از دسته‌ها (افزایش نمونه‌های دسته اقلیت و یا کم کردن نمونه دسته اکثریت)، بر نامتوازنی دسته‌ای غلبه کنند. از بین روش‌های مبتنی بر داده می‌توان به روش‌هایی مانند افزایش تصادفی داده‌های دسته اقلیت **ROS**[3] [13]، کاهش تصادفی نمونه‌های دسته اکثریت **RUS**[4] [14]، تکنیک افزایش نمونه مصنوعی دسته اقلیت **SMOTE**[5] [15]، روش نمونه‌گیری پویا [16] که نرخ نمونه‌گیری را با توجه به عملکرد هر دسته تنظیم می‌کند و یادگیری دو فاز [17]، [18] اشاره کرد.

2- راه حل‌هایی که با تغییر در ساختار مدل، روند آموزش شبکه، تابع ضرر و یا تغییر دادن هزینه‌ها در یادگیری نامتوازن به دنبال رسیدن به نتایجی به نفع دسته اقلیت هستند به راه حل‌های "مبتنی بر الگوریتم" [12] شهرت دارند. برخلاف روش های داده‌ای، روش‌های الگوریتمی، داده‌های آموزشی را تغییر نمی‌دهند و به مراحل پیش پردازش نیاز ندارند. در مقایسه با **ROS**، که پرطرفدارترین روش مبتنی بر داده است، روش‌های الگوریتمی کمتر بر زمان آموزش تأثیر می‌گذارند و به نظر می رسد که می‌توانند برای مسائل داده‌های عظیم بهتر باشند. به استثنای تعریف هزینه‌های دسته‌بندی اشتباه، روش‌های الگوریتمی به تنظیماتی کمتری نیاز دارند. از میان این روش‌ها

[2] backpropagation

[3] Random over sampling

[4] Random under sampling

[5] Synthetic Minority Over-sampling Technique

می‌توان به میانگین خطای اشتباه (**MFE**) و میانگین مربع خطای اشتباه (**MSFE**) [19]، تابع ضرر کانونی[6] [20] و روش‌های حساس به هزینه [21] اشاره کرد

3- روش‌های ترکیبی که از هر دو روش مبتنی بر داده و مبتنی بر الگوریتم برای مقابله با نامتوازنی داده استفاده می‌کنند [22-24].

دسته‌بندهای تجمعی[7]، که به نام سیستم‌های دسته‌بندهای چندگانه هم شناخته‌شده‌است، عملکرد یک دسته‌بند واحد را با ترکیب چندین دسته‌بند پایه بهبود می‌بخشد. کارایی بالای دسته-بندهای تجمعی، آنها را به یک راه‌حل محبوب برای یادگیری نامتوازن تبدیل کرده‌است[25, 26]. یادگیری نامتوازن تجمعی نوعی از روش‌های ترکیبی هستند که با تغییر در داده‌ها و الگوریتم‌ها سعی در غلبه بر نامتوازنی دسته‌ای دارند. گوا و همکارانش در [25] حدود 527 مقاله را در رابطه با یادگیری نامتقارن بررسی کردند که 218 مقاله درباره ارائه مدل‌های جدید تجمعی با استفاده از مدل‌های موجود برای حل عملی مسائل بوده است. آنها راه‌حل‌های تجمعی را به دو دسته روش‌های تجمعی تکرارشونده (مانند بوستینگ) و روش‌های تجمعی موازی (مانند بگینگ) تقسیم کرده‌اند. در [27] انواع روش‌های بوستینگ و بگینگ و ترکیب آنها در برخورد با نامتوازنی داده بررسی شده اند. از روش‌های تجمعی تکرار شونده که برای حل نامتوازنی دسته‌ای استفاده می‌شود می‌توان به درخت تصمیم براساس گرادیان بوستینگ [8](**GBDT**) [28]، آدابوست[29] و الگوریتم‌های تجمعی مبتنی بر الگوریتم تکاملی[9] (**EA**)[30] اشاره کرد. بوستینگ رایج‌ترین مجموعه از روش‌های یادگیری تجمعی است، که با ترکیب دسته‌بندهای ضعیف یک دسته‌بند قوی می‌سازد[31]. به این صورت که هر دسته تلاش می‌کند خطای دسته‌بند قبلی را کاهش دهد.

شبکه‌های عمیق عصبی دارای معماری‌های متنوع و پارامترهای فوق العاده زیادی هستند که باعث می‌شود یک نامزد خوب برای ایجاد مجموعه متنوع از یادگیرنده‌ها باشند. در بخش 2 به بررسی راهکارهای مبتنی بر یادگیری تجمعی برای نامتوازنی دسته‌ای در شبکه‌های یادگیری عمیق می‌پردازیم.

در این پژوهش برای مسائل یادگیری نامتوازن، شبکه تجمعی از **CNN**ها به نام **FENIL**[10] با تابع‌های ضرر کانونی متنوع از نظر مقادیر مختلف γ پیشنهاد شده است، در [20] گروه تحقیقاتی فیس‌بوک (**FAIR**) تابع ضرر کانونی برای شبکه تشخیص دهنده اشیا خود پیشنهاد دادند که در مقایسه با تابع ضرر آنتروپی متقابل استاندارد، توجه بیشتری به داده‌های سخت می‌کند، که این میزان توجه (جریمه‌ای که برای داده‌های سخت و آسان محاسبه‌می‌کند) بستگی به پارامتر γ دارد و یک چالش تعیین میزان مناسب γ در مسائل مختلف است.

در شبکه پیشنهادی **FENIL** دنباله‌ای از گاماها تولید می‌شود ($\gamma_0, \gamma_1 \dots, \gamma_n$)، برای تابع ضرر شبکه **i**ام، گامای **i**ام را در نظر گرفته می‌شود. **CNN**ها به ترتیب اندیس آموزش می‌یابند و آموزش در هر **CNN**، تمرکز بیشتری روی نتایج اشتباه **CNN** قبلی می‌کند. برای این کار از تکنیک آدابوست به عنوان یک روش تجمعی تکرارشونده، استفاده شده است و هر **CNN** را به عنوان یک تخمین‌گر پایه درنظر گرفته‌شده است. شبکه پیشنهادی **FENIL** روی چند مجموعه‌ی داده‌های نامتوازن مرجع[11] آموزش داده شده است و نتایج بدست آمده، با تکنیک-های مختلف مقابله با نامتوازنی دسته‌ای مقایسه شده است (جزئیات پیاده‌سازی و تحلیل نتایج در بخش 5 آورده‌شده است). نتایج، کارایی سیستم تجمعی پیشنهادی را نشان می‌دهد. شبکه تجمعی **FENIL** در برخورد باداده نامتوازن توانایی‌های جدید زیر را دارد:

1- استفاده از تابع ضرر کانونی افزایشی، این مزیت باعث می‌شود شبکه‌های مختلف با γهای مختلف آموزش ببینند و دسته‌بند نهایی از ترکیب نظر همه دسته‌بندها برای نتیجه نهایی استفاده کند. به این ترتیب هم از مزیت آموزشی داده‌های اکثریت بهره برده می‌شود و هم شبکه‌هایی خواهیم داشت که تمرکز

[6] Focal loss
[7] Ensemble
[8] Gradient Boosting Decision Tree
[9] Evolutionary algorithm

[10] Incremental **F**ocal **En**semble method for multi-class **I**mbalalanced **L**earning
[11] Benchmark

بیشتری را روی داده‌های دسته اقلیت دارند و در نهایت همه این شبکه‌ها در دسته‌بندی داده‌های تست مشارکت می‌کنند.

2- استفاده از تغییر بایاس لایه آخر در اولین **CNN**، ما با تغییر بایاس آخرین لایه در اولین **CNN**، دسته‌بندی‌های اشتباه **CNN** در دسته اقلیت را افزایش می‌دهیم و این موجب می‌شود حجم بیشتری از داده‌های کلاس اقلیت و با وزن بیشتری به دسته‌بند بعدی انتقال پیدا کند.

3- استفاده از یادگیری انتقالی برای افزایش سرعت، از آنجا که هر **CNN** برای آموزش به داده‌های زیادی نیاز دارد، و زمان زیادی نیز برای آموزش صرف می‌کند، برای افزایش کارایی، هر **CNN** در زمان شروع آموزش، از **CNN** از پیش‌آموزش‌دیده[12] قبلی استفاده می‌کند و روی داده‌های وزن‌دار جدید دوباره تنظیم[13] می‌شود.

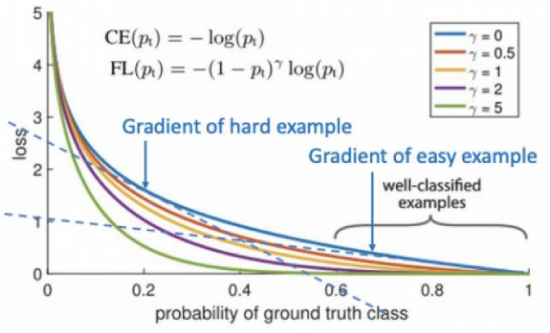
4- در شبکه **FENIL** روش **SAMME.R** که یک تکنیک آدابوست چنددسته‌ای است و در [32] معرفی شده است، برای وزن دهی به نمونه‌های مجموعه داده که در دسته‌بند قبلی به درستی دسته‌بندی نشده‌اند و سپس ترکیب نتایج **CNN** استفاده شده است.

در ادامه در بخش 2 تابع ضرر کانونی شرح داده می‌شود، در بخش 3 به بررسی کارهای انجام شده در زمینه یادگیری عمیق تجمعی می‌پردازیم. در بخش4 معیارهای سنجش کارایی در یادگیری متوازن را بررسی خواهیم‌کرد، در بخش 5 جزئیات راه-کار پیشنهادی ارائه می‌شود و در نهایت در بخش 6 به جزئیات پیاده‌سازی و تحلیل نتایج خواهیم پرداخت و در انتها در بخش 7 جمع‌بندی قرار گرفته‌است.

2. تابع ضرر کانونی

برای غلبه بر عدم توازن شدید بین دسته پیش‌زمینه و پس-زمینه (اشیا) در شناسایی اشیا، لین و همکاران **[20]** تابع ضرر کانونی[14] را معرفی کردند، آنها تابع ضرر آنتروپی متقابل[15] (**CE**) را دوباره بازنویسی کردند به طوری که تأثیر نمونه‌هایی که به راحتی دسته‌بندی شده‌اند، بر روی تابع ضرر را کاهش دهند. این ایده با ضرب تابع ضرر **CE** در یک عامل تعدیل کننده، $\alpha_t (1 - p_t)^{\gamma}$ حاصل می‌شود، هایپرپارامتر $\gamma \geq 0$ میزان کم وزنی مثال‌های آسان را تنظیم می‌کند و $\alpha_t \geq 0$ وزنی دسته‌ای است که برای افزایش اهمیت دسته اقلیت استفاده می‌شود. در شکل 1 تاثیر مقادیر مختلف γ در تابع ضرر کانونی نشان داده شده‌است، با افزایش گاما، در واقع جریمه تولید شده برای داده‌های آسان را کم می‌شود و این باعث کاهش سهم داده‌های آسان در بروز رسانی وزن‌های شبکه **CNN** می‌شود.

$$FL(p_t) = -\alpha_t(1 - p_t)^{\gamma} \log(p_t) \qquad (8)$$



شکل 1- تابع ضرر کانونی برای پیشبینی دسته، با گاماهای مختلف

نسخه چند دسته‌ای و افزایشی این تابع را که در 4-1 درباره آن توضیح داده‌شده‌است، برای آموزش شبکه‌های **CNN** در **FENIL** استفاده شده است.

3. کارهای انجام شده

شبکه‌های عصبی عمیق دارای معماری‌های متنوع و پارامترهای فوق العاده زیادی هستند که باعث می‌شود یک نامزد خوب برای ایجاد مجموعه متنوع از یادگیرنده‌ها (با تنظیم و ترکیب معماری‌های مختلف، تزریق نویز و یا بکاربردن چارچوب آدابوست و ...) در یادگیری تجمعی باشند. برای مثال بهترین نتایج یادگیری عمیق در رقابت **ILSVRC**[16] در سال 2015 روی دیتاست **ImageNet**، براساس یادگیری تجمعی بنا شده بودند [33]. و یا در [34] از **CNN**های مختلفی استفاده

[12] Pre-trained
[13] fine tune
[14] Focal loss
[15] Cross Entropy Loss Function
[16] ImageNet Large Scale Visual Recognition Competition

شده بود که ورودی‌های هریک از آنها به روش‌های مختلفی پیش پردازش شده‌بودند، سپس از میانگین پیش‌بینی همه آنها استفاده شده‌بود. این روش جزو اولین روش‌هایی بود که توانست در مجموعه داده **MNIST** به دقتی نزدیک به دقت انسان دست پیداکند. با توجه به موفقیت‌های مدل‌های عمیق تجمعی و موفقیت‌های پیشین یادگیری تجمعی در مقابله با نامتوازنی دسته‌ای، استفاده از یادگیری عمیق تجمعی برای مقابله با نامتوازنی دسته‌ای مورد توجه محققان در حوزه یادگیری ماشین قرار گرفته‌است [35-37]. در ادامه به بررسی برخی از این روش‌ها می‌پردازیم.

هانگ و همکارانش در [38] از ترکیب چند **CNN** و آموزش همزمان آنها، برای غلبه بر نامتوازنی دسته‌ای استفاده‌کرده‌است. آنها از روش‌های مبتنی بر خوشه‌بندی برای انتخاب نمونه‌های درون یک بسته[17] استفاده کرده‌اند و سپس **CNN**ها را بوسیله این بسته‌های کوچک آموزش داده‌اند. آنها ابتدا دسته‌های اکثریت و اقلیت را به طور جداگانه خوشه‌بندی می‌کنند و سپس پنج‌تایی نمونه‌ها را در بسته‌های کوچک به طوری قرار می‌دهند که تعداد دسته اکثریت و اقلیت در آنها مساوی باشد. ایجینا و همکارانش [39] روش تجمعی برای تشخیص فعالیت انسان در ویدئو ارائه دادند، آنها از ماکسیمم نتایج دسته‌بندها به عنوان نتیجه نهایی استفاده کردند. لیکسبرگ و همکارانش در [40] برای تقسیم بندی حجمی دقیق تومور در تصاویر ام‌آرآی از یادگیری تجمعی استفاده کردند.

تاجبخش و همکارانش [41] سیستم تشخیص پولیپ را پیشنهاد دادند، هریک از **CNN**ها درسیستم تجمعی یکی از ویژگی‌های پولیپ (رنگ، بافت و شکل ظاهری) را تشخیص می‌دهد. زی و همکارانش [42] سه روش تجمعی بر اساس شبکه‌های عصبی عمیق پیشنهاد کردند که دنباله‌هایی از دسته-بندها را باهم ترکیب می‌کند به طوری که ورودی‌هایشان بازنمایی از لایه‌های ژانگ و همکارانش [92] تجمعی از **DNN**ها را ارائه دادند که به طور متفاوت وزن دهی اولیه می‌شود و اختلاف بین خروجی هریک از آنها با میانگین خروجی‌ها، جریمه می‌شود. در [43] یک شبکه عصبی تجمعی ارائه شده است که با استفاده از روش باز نمونه‌برداری[18](بوت استراپت[19]) درکنار گروهی از شبکه‌های عصبی مشکل کمبود رویدادها را بهبود داده‌است. آنها از مجموعه بزرگی از ویدیوهای فوتبالی با هدف شناسایی اتفاق کرنر استفاده کردند.

در این پژوهش برای ساخت شبکه **FENIL** از دنباله‌ای از **CNN**ها استفاده‌شده‌است که در آن میزان اهمیت ورودی‌های هر شبکه و تابع ضرر هر شبکه متفاوت است، میزان اهمیت ورودی‌های هر شبکه براساس الگوریتم آدابوست پیشنهادی **SAMME.R** در [32] محاسبه‌شده‌است. در **SAMME.R** از درخت تصمیم برای تخمین‌گر پایه استفاده می‌شود اما در این پژوهش از **CNN** با تابع ضرر کانونی متفاوت به عنوان تخمین‌گر پایه استفاده شده است که باعث افزایش کارایی زیادی در نتایج داده‌های مصنوعی شده‌است (دقت 18/63 بالاتر و **F-score** /61 19بالاتر).

4. معیارهای ارزیابی برای داده‌های نامتوازن

ماتریس درهم‌ریختگی[20] که در جدول (1) نشان داده شده-است، نتایج یک دسته‌بندی دودویی را نمایش می‌دهد. **TP** تعداد نمونه‌هایی هستند که به درستی در دسته مثبت دسته‌بندی شده اند و **TN** تعداد نمونه‌هایی هستند که به درستی در دسته منفی دسته‌بندی شده‌اند، **FN** و **FP** به ترتیب تعداد نمونه‌هایی است که به طور اشتباه در دسته منفی و مثبت دسته‌بندی شده‌اند.

ماتریس درهم ریختگی

	واقعی مثبت (1)	واقعی منفی (0)
دسته بندی شده مثبت (1)	مثبت درست (TPs)	مثبت نادرست (FPs)
دسته بندی شده منفی (0)	منفی نادرست (FNs)	منفی درست (TNs)

جدول 1- ماتریس درهم‌ریختگی

همه معیارهایی که در ادامه آمده‌است از ماتریس درهم-ریختگی بدست آمده‌است. معیار دقت و خطا (رابطه 1) یک

[17] batch
[18] sampelling method
[19] bootstrapped
[20] confusion matrix

روش ساده برای بیان کارایی دسته‌بند روی مجموعه داده ها را بیان می‌کند اما همانطور که در مثال بعدی بیان می‌شود، در شرایط خاص می‌تواند فریب‌دهنده باشد و نسبت به تغییر در داده‌ها بسیار حساس است. در ساده‌ترین حالت، اگر ترکیب مجموعه داده‌ها، 5 درصد دسته اقلیت و 95 درصد دسته اکثریت باشد، یک راهکار ساده‌ای که همه نمونه‌ها را در دسته اکثریت دسته‌بندی می‌کند، می‌تواند دقت 95 درصد را بدست آورد. در ظاهر دقت 95 درصد روی کل داده‌ها عالی به نظر می‌رسد، با این حال نمی‌تواند منعکس‌کننده این حقیقت باشد که 0 درصد از نمونه‌های دسته اقلیت شناسایی شده‌اند. به این دلیل است که

$$\textbf{precision} = \frac{TP}{TP+FP} \quad (2)$$

$$\textbf{recall} = \frac{TP}{TP+FN} \quad (3)$$

$$\textbf{f} - \textbf{measure} = \frac{(1+\beta^2)\times \textbf{precision}\times \textbf{recall}}{\beta^2\ \textbf{precision}+\textbf{recal}} \quad (4)$$

در رابطه (4) ضریبی است که برای تنظیم اهمیت دقت نسبت به بازخوانی (معمولا یک است) بکار برده می‌شود. صحت، میزان دقیق بودن است (برای مثال نمونه‌هایی که برچسب مثبت خورده اند تا چه حد درست برچسب گذاری شده‌اند) درحالی که بازخوانی معیار کامل بودن است (برای مثال، چه تعداد از نمونه‌های دسته مثبت درست برچسب گذاری شده‌اند). برعکس دقت و نرخ خطا که هر دو به توزیع مجموعه داده حساس بودند، صحت و بازخوانی هردو به توزیع حساس نیستند. یک بررسی ساده روی رابطه‌های 2 و 3 نشان می‌دهد، صحت به توزیع دسته حساس است اما بازخوانی نه. با این حال در صورتی که به درستی استفاده شود، صحت و بازخوانی می‌تواند به صورت کارا، کارایی دسته‌بند را در روند یادگیری نامتوازن ارزیابی کند. به خصوص معیار **F-Measure** (که با **F-score** شناخته می‌شود) با درنظر گرفتن نسبت اهمیت بازخوانی به صحت که توسط کاربر مشخص می‌شود، صحت و بازخوانی را ترکیب می‌کند. **F-Measure** درک عمیق تری از کارایی دسته‌بند نسبت به معیار صحت ارائه می‌کند، در مواردی که با به جای مسئله دو کلاسه با مساله چند کلاسه روبرو هستیم، اگر ماتریس درهم‌ریختگی را **CM** بنامیم برای کلاس **i**ام **recall**$_i$ (بازخوانی دسته **i**ام) به

[21] precision

می‌گوییم، معیار دقت در این حالت اطلاعات مناسبی را درباره کارایی دسته‌بند با توجه به نوع دسته‌بندی ارائه نمی‌دهد.

$$\textbf{Accuracy} = \frac{TP+TN}{TP+FP+TN+FN} \quad (1)$$
$$\textbf{error rate} = 1 - \textbf{Accuracy}$$

با داده‌های نامتوازن، هنگامی که معیارهای ارزیابی به توزیع داده‌ها حساس هستند، تجزیه و تحلیل نسبی مشکل می‌شود. به جای دقت، دیگر معیارهای ارزیابی که معمولا اتخاذ می‌شوند عبارت است از: صحت[21]، بازخوانی[22] و **f-measure (F1-score)** که به صورت زیر تعریف می‌شوند:

صورت تقسیم عنصر $\textbf{CM}_{i,i}$ بر مجموع عناصر ستون **i** ام تعریف می‌شود (رابطه 6) و $\textbf{precision}_i$ (صحت دسته **i**ام) به صورت تقسیم عنصر $\textbf{CM}_{i,i}$ بر مجموع عناصر سطر **i** ام تعریف می‌شود. معیار **F-Measure** برای تک تک دسته‌ها به صورت جداگانه محاسبه می‌شود. معیار دیگر **Balanced accuracy** یا دقت متوازن شده‌است که به طور گسترده از آن در یادگیری نامتوازن برای بررسی کارایی دسته‌بندها استفاده می‌شود، این رابطه در واقع میانگین دقت هر یک از دسته‌ها و از آنجا که مجموع ستون **i**ام ماتریس درهم‌ریختگی برابر با تعداد نمونه‌های دسته **i**ام است می‌توان دقت متوازن شده را به صورت میانگین **recall** دسته‌ها بیان کرد، در این مقاله برای مقایسه کارایی روش پیشنهادی از معیارهای صحت، بازخوانی و **F1-score** (**F1-score**) به این معنی است که $\boldsymbol{\beta}$ یک در نظر گرفته‌شده‌است) و **Balanced accuracy** استفاده شده‌است. در رابطه (5) و (6) **CM** نشان‌دهنده ماتریس درهم‌ریختگی است و در رابطه (7)، **C** برابر تعداد دسته‌ها و $\textbf{acc}_i$ برابر دقت دسته‌بند در هر دسته است.

$$\textbf{precision}_i = \frac{CM_{i,i}}{\sum_j CM_{i,j}} \quad (5)$$

$$\textbf{recall}_i = \frac{CM_{i,i}}{\sum_j CM_{j,i}} \quad (6)$$

$$\textbf{balanced accuracy} = \frac{\sum_i acc_i}{C}, \quad \textbf{acc}_i=\textbf{recall}_i \quad (7)$$

[22] recall

فصلنامهٔ علمی - پژوهشی فرماندهی و کنترل، سال ششم، شمارهٔ دو، تابستان 1401

5. روش پیشنهادی FENIL

در شبکه تجمعی پیشنهادی **FENIL**، دنباله‌ای از شبکه‌های کانولووشنال قرار گرفته‌است که هر شبکه در ساختار تجمعی با اندیس **i** مشخص می‌شود، این شبکه‌ها از نظر معماری یکسان هستند، اما داده ورودی و تابع ضرری که برای آموزش استفاده می‌کنند، متفاوت است. برای وزن دهی به داده‌ها از نسخه چند کلاسه آدابوست که در [32] معرفی شده‌است استفاده کرده‌ایم، روند کار به این صورت است که ضریب داده‌های ورودی به **CNN** بعدی از نتیجه دسته‌بندی **CNN** قبلی مشخص می‌شود. ما برای آموزش **CNN**ها از تابع ضرر کانونی معرفی شده در[20] استفاده کرده‌ایم با این تفاوت که پارامتر γ که نشاندهنده میزان اهمیت داده‌های آموزشی سخت است(داده‌هایی که توسط دسته‌بند به خوبی دسته‌بندی نمی‌شوند)، برای هر **CNN** نسبت به **CNN** قبلی افزایش می‌یابد. در بخش4-1 درباره ساختار پیشنهادی و تابع ضرر پیشنهادی صحبت می‌شود.

5-2. ساختار شبکه پیشنهادی FENIL

ساختار پیشنهادی **FENIL** برای ترکیب **CNN**ها که در شکل 2 نشان داده‌شده‌است، یک ترکیب آبشاری (**Cascade**) از تعدادی شبکه کانولووشنال است، تفاوت این شبکه‌ها در تابع ضرر استفاده شده برای آموزش و وزن داده‌هایی است که برای آموزش استفاده می‌کنند، گرچه داده‌های همه شبکه‌ها یکسان هستند، اما وزن داده‌ها در آموزش برای هر شبکه براساس نتیجه دسته‌بندی **CNN** قبلی مشخص می‌شود. در ادامه تابع ضرر و روند وزن‌دهی داده‌ها را شرح خواهیم داد.

- ✓ **تابع ضرر کانونی افزایشی**

ما برای آموزش هر کدام از دسته‌بندها (**CNN**ها) از نسخه چنددسته‌ای تابع ضرر کانونی که در رابطه (9) تعریف کرده‌ایم استفاده می‌کنیم. (تابع ضرر معرفی شده در [20] برای دو دسته تعریف شده‌است). هر **CNN** تابع ضرر کانونی مختص خود را دارد، که به صورت زیر نشان داده شده است:

$$FL_i = \sum_{c=1}^{C} (1 - p_c^k)^{\gamma_i} t_c^k \log p_c^k \qquad . \gamma_i = \alpha i + \gamma_0$$

t^k بردار هدف برای داده **k** ام است، p^k خروجی سافت مکس آخرین لایه شبکه و درواقع پیشبینی شبکه برای داده **k** ام است، پارامتر γ_i در تابع ضرر کانونی در رابطه(9) برای دسته بند **i** ام، نسبت به دسته بند **i-1** به اندازه α افزایش پیدا می‌کند، و این بدین معنی است که دسته‌بند **i** ام نسبت به دسته‌بند قبلی به نمونه‌های آسان (نمونه‌هایی که با دقت بالایی درست تشخیص داده می‌شوند) اهمیت کمتری می‌دهد، و این موضوع باعث می‌شود تاثیر تعداد زیادی از نمونه‌های کلاس اکثریت که به‌خوبی دسته‌بندی شده‌اند و آموزش شبکه بوسیله آنها قدرت شبکه را بیش از این افزایش نخواهد داد، در بروزرسانی پارامترها کاهش پیدا کنند. α به عنوان یک هایپرپارامتر قابل تنظیم، سرعت تغییر گاما را با پیشروی در شبکه آبشاری **CNN**ها مشخص می‌کند. γ_0 برای اولین **CNN** برابر صفر است، بنابراین اولین شبکه با تابع ضرر آنتروپی متقابل آموزش می‌بیند.

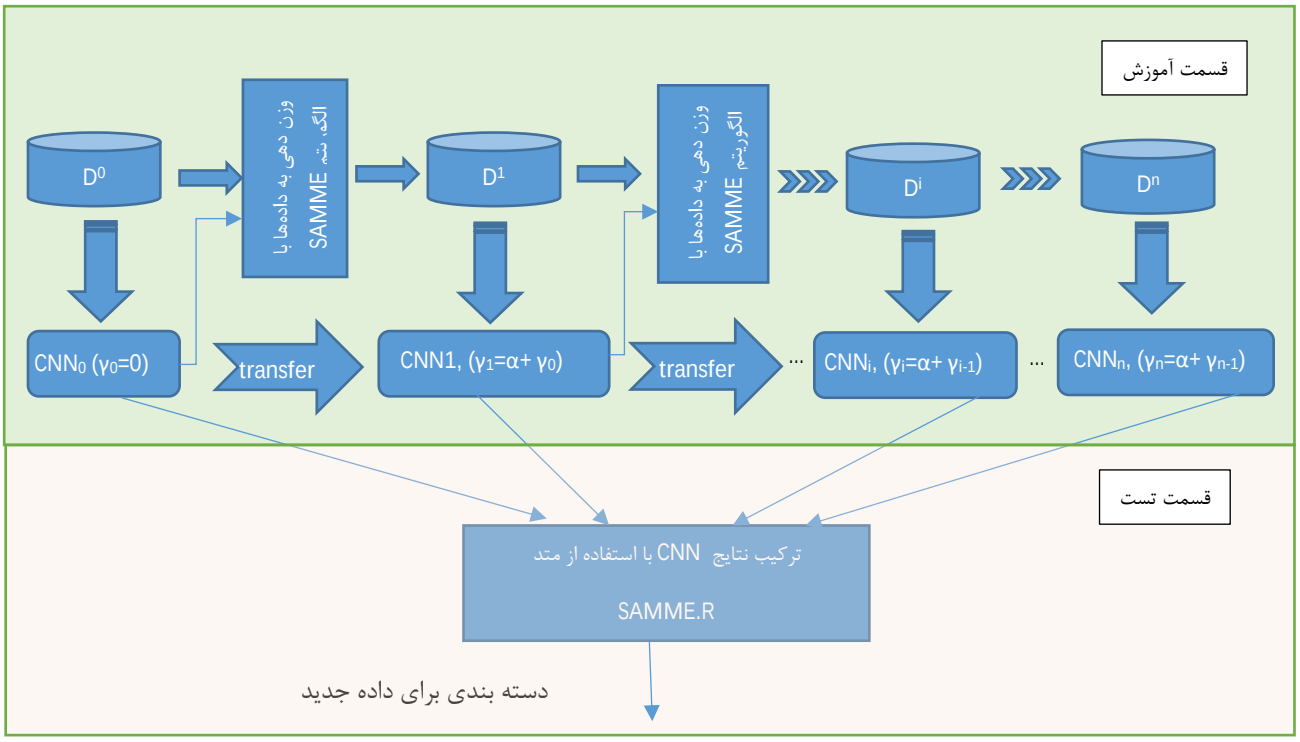
- ✓ **وزن دهی آدابوست برای داده‌های شبکه i ام**

راهکار [32] **SAMME.R**، مشابه ادابوست اولیه، وزن داده-هایی که در دسته‌بند قبلی اشتباه دسته‌بندی شده‌اند را افزایش می‌دهد، بدین صورت که برای وزن دهی داده‌های آموزشی $\mathbf{CNN_i}$ (یعنی $\mathbf{D_i}$) طبق رابطه (10) از نتیجه دسته‌بندی **CNN** قبلی استفاده می‌شود، بنابراین برای بدست آوردن وزن داده **i** ام ($\mathbf{x_i}$) برای $\mathbf{CNN_{k+1}}$ خواهیم داشت:

$$w_i^{k+1} = w_i^k \exp\left(-\frac{C-1}{C} Y_i^T \log P^k(x_i)\right) \quad i = 1 \dots n \qquad (10)$$

در این رابطه، w_i^k وزن داده **i** ام در دسته‌بند **k**ام ، **C** تعداد دسته‌ها و $P^k(x_i)$ پیشبینی دسته‌بند **k**ام برای داده **i**ام ($P^k = [p_c^k]$) و Y_i مقادیر هدف داده‌ی **i**ام است. طبق رابطه (10) ، هرچه لگاریتم پیشبینی $(\log P^k(x_i))$ و مقدار واقعی ($\mathbf{Y_i}$) به هم نزدیک‌تر باشد، به دلیل علامت منفی که در حاصل‌ضرب این دو بردار وجود دارد، وزن کوچکتر می‌شود، و هرچه بیشتر از هم فاصله داشته‌باشند، وزن بیشتر می‌شود. بعد از بدست آوردن وزن همه داده‌ها، وزن‌ها نرمال‌سازی می‌شوند. بدین ترتیب وزن داده‌های آموزشی $\mathbf{CNN_{k+1}}$ با استفاده از خروجی $\mathbf{CNN_k}$ یعنی P^k مشخص می‌شود، بنابراین می‌توانیم بگوییم:

$$D^k = \{w_i^k x_i\} \quad i = 1 \dots n$$



شکل 2– ساختار پیشنهادی برای شبکه تجمعی FENIL

✓ **یادگیری انتقالی**

ما در این ساختار از مزیت یادگیری انتقالی استفاده می‌کنیم، به این صورت که هر **CNN** در زمان شروع آموزش، وزن‌های اولیه خود را از **CNN** قبلی می‌گیرد و سپس روی داده‌های جدید تنظیم دقیق[23] می‌شود. هنگام کار با **CNN**ها استفاده از یادگیری انتقالی ضروری به نظر می‌رسد، چرا که با حرکت در دنباله **CNN**ها، جریان داده‌های آموزشی کاهش پیدامی‌کند، به این علت که وزن داده‌هایی که در همه **CNN**های قبلی درست دسته‌بندی شده‌اند، به شدت کاهش می‌یابد و درواقع با حرکت از یک **CNN** به **CNN** بعدی حجم داده‌های موثر در آموزش کاهش می‌یابد. بنابراین اگر بخواهیم شبکه‌ها را از ابتدا آموزش دهیم، به خوبی آموزش نمی‌بینند. در شبکه‌های **CNN** عمیق‌تر مثل **ResNet**، در **CNN** بعدی، لایه‌های ابتدایی را فریز می‌کنیم و فقط آموزش را روی لایه‌های انتهایی ادامه می‌دهیم.

بسته به پیچیدگی مجموعه داده مورد بررسی و قدرت **CNN** پایه، تعداد بهینه تخمین‌گرها متفاوت خواهد بود، بنابراین تعداد تخمین‌گرها و سرعت تغییر γ از یک **CNN** به **CNN** بعدی به عنوان دو هایپر پارامتر، برای مسائل مختلف قابل تنظیم است. اضافه کردن **CNN** بیشتر، همیشه به معنای رسیدن به دقت بیشتر نیست. از آنجا که با عبور جریان داده‌ها از هر **CNN** وزن داده-هایی که اشتباه دسته‌بندی شده‌اند افزایش پیدا می‌کند و وزن داده‌هایی که بدرستی دسته‌بندی شده کاهش پیدا می‌کند، بعد از عبور از چند **CNN**، وزن تعداد زیادی از داده‌ها بسیار کم خواهد شد و با حرکت به سمت **CNN**های انتهایی، حجم داده‌های موثر در آموزش **CNN** کاهش پیدا می‌کنند و برای آموزش **CNN**های انتهایی کافی نیستند، با استفاده از یادگیری انتقالی در این ساختار می‌توان حجم داده مورد نیاز برای آموزش **CNN**ها را تا حدی کاهش و تعداد بهینه **CNN**ها را افزایش داد.

✓ **تغییر بایاس در لایه آخر**

[23] fine tune

همانطور که در رابطه 11 و 12 دیده می‌شود، خروجی رگرسیون هر **CNN** با استفاده از یک تابع فعال‌ساز سافت‌مکس، به توزیع احتمال دسته تبدیل می‌شود،

$$s = \theta^o(F^{o-1})^T + b^o \quad (11)$$

$$p = softmax(s) \quad (12)$$

که دراین رابطه θ^o وزن لایه آخر، F^{o-1} بردار ویژگی ورودی به لایه آخر و b^o بایاس لایه آخر است. در ساختار پیشنهادی **FENIL** مقدار اولیه وزن‌های لایه‌های **CNN** مقادیر پیش‌فرض پایتورچ و بایاس همه لایه‌ها، بجز لایه آخر صفر در نظر گرفته‌شده‌است. از بایاس لایه آخر به عنوان ابزاری برای اهمیت بیشتر به دسته اقلیت استفاده کرده شده‌است. یقینا تعداد خطاها در هر دسته، در اولین **CNN** می‌تواند وزن آن دسته را در آموزش‌های بعدی افزایش دهد (چراکه وزن داده‌های آن دسته با دسته‌بندی اشتباه افزایش پیدا می‌کند)، بنابراین اگر ترفندی بکاربریم که تعداد خطاها در دسته اقلیت را افزایش دهیم، به نفع کلاس اقلیت خواهد بود، برای رسیدن به این منظور در شبکه پیشنهادی از تغییر در بایاس لایه آخر استفاده شده‌است.

تعداد عناصر بردار بایاس در لایه آخر برابر تعداد دسته‌هاست، بنابراین به ازای هر دسته یک مقدار معادل در بایاس وجود دارد، مقدار عنصر **i**ام بایاس برای هر کلاس **i**ام طوری در نظرگرفته‌شده‌است که برای دسته با تعداد کمتر (دسته اقلیت) تابع سافت‌مکس عدد کوچکتری تولید کند، (عدد کوچکتر، احتمال دسته‌بندی درست را کاهش می‌دهد)، برای مثال برای یک مساله سه دسته‌ای با تعداد نمونه‌های **[800, 500, 1000]**، یک انتخاب می‌تواند b^o = **[-0.5, -1.0 ,0.0]** باشد، هر عدد مشخص کننده بایاس برای یک دسته است، مثلا **0.5-** برای دسته با 800 نمونه و **1.0-** بایاس برای دسته با 500 نمونه است. این بایاس احتمال پیشینی (**prior probability**) برابر **p** = **[0.3072, 0.1863, 0.50]** تولید می‌کند(در واقع **0.5-** سافت مکس وارون **0.3072** است)، چنین بایاسی برای این مثال باعث می‌شود احتمال پیشبینی درست در دسته اقلیت (دسته دوم)، مخصوصا در دورهای اولیه، کاهش و در نتیجه وزن آن افزایش پیدا کند. این روش مشابه تغییر حد آستانه‌ی تصمیم‌گیری در مسائل دسته‌بندی دودویی است، با این تفاوت که روش تغییر بایاس، در دورهای اولیه آموزش بیشتر نمود پیدا می‌کند.

✓ **ترکیب CNNها در زمان تست برای داده‌های جدید**

برای دسته‌بندی داده‌های جدید توسط شبکه پیشنهادی **FENIL** از الگوریتم **SAMME.R** معرفی شده در [32] استفاده‌شده‌است. برای دسته‌بندی با **M** شبکه **CNN**، برای پیشبینی دسته داده ورودی (**C(x)**) از را بطه (13) استفاده‌شده‌است.

$$C(x) = arg\max_c \sum_{m=1}^{M} h_c^{(m)}(x) \quad c = 1.\cdots.C \quad (13)$$

که در این رابطه **M** تعداد **CNN**ها، **C** تعداد دسته‌ها است و $h_c^{(m)}(x)$ از رابطه زیر بدست می‌آید.

$$h_c^{(m)}(x) \leftarrow (C-1)\left(logp_c^{(m)}(x) - \frac{1}{C}\sum_{c'} logp_{c'}^{(m)}(x)\right) \quad c = 1.\cdots.C \quad (14)$$

6. جزئیات پیاده‌سازی و نتایج

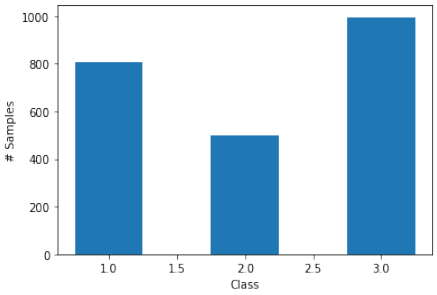
برای پیاده‌سازی راهکار پیشنهادی، از چهار مجموعه‌داده مختلف استفاده‌شده‌است، مجموعه‌داده‌مصنوعی[32]، مجموعه‌داده **Cifar-10 [44]** که مجموعه از تصاویر سایز کوچک در ده دسته مجزا تشکیل شده‌است، مجموعه داده تشخیص فعالیت **EGTEA gaze+** [45] که تصاویر ویدیویی اول شخص از فعالیت‌های مرتبط با آشپزخانه است و مجموعه داده‌های پیگیری فعالیت از طریق سیگنال‌های ذخیره‌شده در گوشی همراه، که **actitracker** [46] نام دارد. جزئیات پیاده‌سازی، شبکه پایه **CNN** پیشنهادی برای قرار گرفتن در شبکه تجمعی **FENIL** و نتایج برای هریک از مجموعه‌های داده در ادامه آمده‌است.

6-1. مجموعه داده مصنوعی

برای ایجاد دیتاست نامتوازن مصنوعی از توزیع نرمال استاندارد چند بعدی استفاده شده است. سه دسته در این دیتاست مصنوعی استفاده شده است که شامل 12300 نمونه است و هر نمونه یک بردار 10 بعدی است. داده‌ها به شکلی که در [1] بیان شده ایجاد شده است. رابطه زیر این داده‌ها را توصیف می‌کند. **c** مشخص کننده برچسب دسته است.

$$c = \begin{cases} 1. & 0 \le \sum x_j^2 \le X_{1/3}^2 \\ 2. & X_{1/3}^2 \le \sum x_j^2 \le X_{2/3}^2 \\ 3. & X_{2/3}^2 \le \sum x_j^2 \end{cases} \quad (15)$$

در رابطه (15) $X_{k/3}^2$ چندک[24]هایی هستند که مشخص کننده k/3 از کل داده‌های 10 بعدی تولید شده با توزیع X^2 هستند و $\sum x_j^2$ فاصله اقلیدسی است. درواقع رابطه (15) نشان دهنده داده‌های مربوط به داده‌های مختلف است که توسط کره‌های چندبعدی متحدالمرکز تودرتو جدا می‌شوند. همان‌طور که در رابطه (15) نشان‌داده‌شده‌است، نمونه‌های دسته 1 در اطراف مبدا یک کره قرار گرفته‌اند، نمونه‌های دسته 2 بین سطح دو کره توزیع شده‌اند و نمونه‌های دسته 3 خارج از کره‌ها قراردارند.



شکل 3- توزیع داده‌های مصنوعی در سه کلاس

برای ایجاد یک مجموعه داده‌ی آموزشی با نامتوازنی دسته-ای، در مجموع، 2300 نمونه به عنوان داده‌های آموزشی استخراج می‌شود، به طوری که 800، 500 و 1000 نمونه به ترتیب، متعلق به کلاس 1، 2 و 3 هستند(شکل 3شکل 3). مجموعه تست با استخراج 10000 نمونه مستقل ساخته می‌شود. هر نمونه در مجموعه آزمون متعلق به یکی از سه دسته با احتمال‌های مساوی است. تقریباً به تعداد مساوی داده در مجموعه تست برای سه دسته وجود دارد.

جدول 1-**پیکربندی CNN استفاده شده برای دیتاست داده‌های مصنوعی**

لایه‌های شبکه CNN پیشنهادی:
کانولووشن یک بعدی با 32 فیلتر و کرنل‌های ***3x1***
فعال ساز ***ReLU***
کانولووشن یک بعدی با 64 فیلتر و کرنل‌های ***2x1***
Max-Pooling *با کرنل‌های* ***2x1***
لایه تمام متصل با 128 نرون و فعال ساز ***ReLU***
Dropout 20%
لایه تمام متصل با 16 نرون و فعال ساز ***ReLU***
لایه تمام متصل با 3 نرون و فعال ساز ***SoftMax***

همانطور که در

جدول 1 نشان داده شده است، **CNN** پیشنهادی برای مجموعه داده مصنوعی دارای شش لایه است که در انتهای لایه‌ی خروجی آن فعال ساز **soft-max** قرار گرفته است. لایه اول یک لایه کانولوشن یک بعدی با کرنل‌های 3×1 است و لایه دوم هم یک لایه کانولوشن یک بعدی با هسته‌های 2×1 است و به دنبال آن یک لایه **max-pooling** 2×1 قرار گرفته است. این دو لایه به ترتیب دارای تعداد 32 و 64 فیلتر هستند. پس از آنها، دو لایه کاملاً متصل به ترتیب دارای 128 و 16 نرون قرار دارد و **dropout** 20 درصد در لایه 128 نورون کاملاً متصل استفاده شده است. تمام لایه‌ها با استفاده از تابع **ReLu** فعال می‌شوند. در نهایت، لایه خروجی یک لایه کاملا متصل با سه نرون است که با استفاده از تابع **softmax** فعال می‌شود. بهینه ساز **Adagard** با نرخ یادگیری برابر با 0.3 و **exponential scheduler** با گاما برابر با 0.9 استفاده شده است، وزن و بایاس در ابتدا با تنظیمات پیش فرض **PyTorch** استفاده شده است. لازم به ذکر است تعداد لایه‌ها و نحوه قرار گیری آنها با سعی و

[24] Quantile

خطا و بررسی پیکربندی‌های مختلف و مقایسه دقت نهایی آنها انتخاب شده است.

در ساختار شبکه **FENIL** پیشنهادی، تعداد بهینه تخمین‌گرها 8 و α در رابطه (9)، یک در نظر گرفته شده است، برای مقایسه نتایج از معیارهای دقت، **f1-score** و دقت متوازن‌شده استفاده شده است و در جدول 2 با دیگر روش‌های مقابله با نامتوازنی داده‌ها مقایسه شده است.

دقت آموزش و تست، معیار **F1-score** و **Balanced Accuracy** روی مجموعه داده مصنوعی در جدول 2 نشان داده شده است. نتیجه نشان می‌دهد که شبکه **FENIL** پیشنهادی بهترین نتایج را در مقایسه با از رویکردهای دیگر کسب کرده است، معیارهای مقایسه نسبت به روش قدیمی آدابوست که در [32] با تخمین‌گرهای درخت تصمیم پیشنهاد شده است، فاصله زیادی دارد و میزان افزایش کارایی 18/63، 19/61 و 19/45 بالاتر را به ترتیب برای دقت تست، **F1-score** و **Balanced accuracy** بدست‌آورده‌است. دقت آموزش شبکه پیشنهادی 2/73 بیشتر از **CNN** تنها با تابع ضرر آنتروپی متقابل استاندارد است و دقت تست، **F1-score** و **Balanced accuracy** در نمونه آزمایشی 10000 تایی هنگام استفاده از شبکه تجمعی پیشنهادی، به ترتیب2/53، 1/55 و 2/11 بیشتر است. در مقایسه تابع ضرر پیشنهادی با آنتروپی متقابل متوازن (وزن‌دار) که یک راه حل در مواجهه با نامتوازنی دسته‌ای است، دقت زمان آموزش و تست، **F1-score** و **Balanced accuracy** شبکه تجمعی پیشنهادی ما به ترتیب 1/33، 1/7، 1/43 و 1/25 بیشتر است.

روش پرطرفدار دیگر برای رسیدگی به نامتوازنی دسته‌ای، روش بیش نمونه برداری مصنوعی دسته اقلیت **[15] (SMOTE)** است، نتیجه **SMOTE** روی داده‌های مصنوعی در ردیف4 جدول 2 نشان داده شده‌است، روش ما دقت آموزش و تست بهتری را نسبت به **SMOTE** نشان‌داده‌است. به طور کلی استفاده از **SMOTE** دقت **CNN** استاندارد با تابع ضرر آنتروپی متقابل را در نمونه‌های آزمایشی بهبود بخشیده است. همچنین برای بررسی تاثیر تابع ضرر کانونی افزایشی، با تابع آنتروپی متقابل استاندارد جایگزین شده‌است (**CASCADE-CNN**، ردیف 5 جدول) و با روش پیشنهادی **FENIL** مقایسه شده‌است، روش ما در هر دو مرحله آموزش و تست، **f1.score** و **Balanced accuracy** به ترتیب با مقدار 1/55، 1/63، 1/16 و 1/22 بالاتر، دقت بهتری دارد.

جدول 2 - مقایسه نتایج تکنیک‌های مختلف روی داده‌های مصنوعی

Model name	Train acc %	Test acc %	F1-score %	Balanced_accuracy
AdaBoost-Decision-Tree	91.78	77.08	75.54	75.64
CNN+CE	94.43	93.18	93.6	92.98
CNN+balanced CE	95.83	94.01	93.72	93.84
SMOTE-CNN	95.48	93.45	93.89	93.22
CASCADE-CNN	95.61	94.08	93.99	93.87
روش پیشنهادی	97.16	95.71	95.15	95.09

2-6. مجموعه داده long tailed-Cifar10

مجموعه داده **Cifar-10[44]** متوازن است و شامل 10 دسته است که برای هر دسته 5000 داده آموزشی استفاده شده است. و 10000 تصویر برای داده‌های تست ذخیره شده است. برای بررسی کارایی شبکه **FENIL** پیشنهادی و ایجاد دیتاست نامتوازن، از نسخه **Long-Tailed Cifar** به نام (**Cifar-10-LT**) [47] استفاده‌شده‌است. تعداد دسته‌های مجموعه‌های داده **Cifar-10** استاندارد و **Cifar-10-LT** برابر است، اما تعداد نمونه‌های هر کلاس با توجه به تابع نمایی $n = n_t \times \mu^t$ مشخص می‌شود. که **t** ایندکس دسته و n_t تعداد نمونه‌های دسته در مجموعه داده اولیه و μ عددی بین **0** و 1 است. مجموعه آزمون تغییری نمی‌کند. فاکتور نامتوازنی (**imbalanced factor**) برابر است با حاصل تقسیم بزرگترین دسته بر کوچکترین دسته،

فصلنامهٔ علمی - پژوهشی فرماندهی و کنترل، سال ششم ، شمارهٔ دو، تابستان 1401

که بین 1 تا 100 متغیر است. برای آموزش **Cifar-10-LT** از شبکه **ResNet18**، **batch size=256**، **learning rate=0.01**، اپتیمایزر **SGD** با **momentum=0.9** و **weight_decay=0.01** به عنوان تخمین‌گر پایه استفاده شده است. تعداد بهینه تخمین گرها برابر 4 است و الفا (نرخ تغییر گاما با افزایش **i**) برابر یک است. آموزش اولین شبکه از ابتدا و به تعداد 80 اپوچ انجام می‌شود و در سه تای بعدی فقط لایه آخر آموزش می‌بینند و همه لایه‌ها دیگر فریز می‌شوند.

جدول 3- مقایسه کارایی شبکه تجمعی پیشنهادی با دیگر راهکارهای مقابله با نامتوازنی دسته ای با معیار **Balanced accuracy** برای نرخ‌های نامتوازنی مختلف برای **Cifar-10-LT**

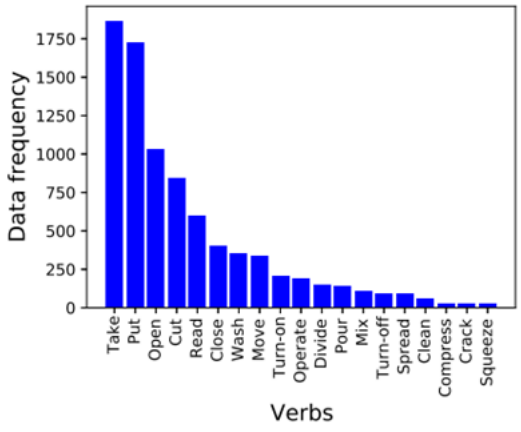
Imbalanced rate	**100**	**50**	**20**	**10**
ResNet+CE	**70.36**	**74.81**	**82.23**	**86.39**
CNN+Balanced CE	**70.22**	**75.82**	**82.90**	**86.84**
Resnet+Focal(γ=2)	**70.38**	**76.71**	**82.89**	**86.81**
SMOTE+ResNet	**71.99**	**77.1**	**82.76**	**86.85**
Voting ResNet(n=4)	**70.96**	**75.99**	**83.16**	**86.82**
Cascade ResNet(n=4)	**71.25**	**76.84**	**83.42**	**86.85**
روش FENIL پیشنهادی	**73.31**	**77.21**	**83.56**	**86.98**

روش پیشنهادی بر روی **Cifar-10-LT** با نرخ نامتوازنی مختلف بین 10 تا 100 اعمال شده‌است و معیار **Balanced accuracy** را برای هریک محاسبه شده‌است و در جدول 3 با سایر روش‌های مقابله با نامتوازنی دسته‌ای مقایسه شده است که شامل استفاده از تنها یک **ResNet** با تابع ضرر آنتروپی استاندارد، تابع ضرر متوازن شده (وزن‌دار)، استفاده از تکنیک افزایش مصنوعی داده ای کلاس اقلیت **SMOTE** و رای‌گیری از 4 شبکه **ResNet** که به طور متفاوت وزن‌دهی اولیه شده‌اند می‌باشد.

نتایج نشان می‌دهند روش پیشنهادی بهترین کارایی را بین تکنیک‌هاهی مطرح شده بدست آورده‌است.

3-6. مجموعه داده EGTEA

همچنین روش پیشنهادی روی مجموعه داده **EGTEA Gaze+** [45] اجرا شده است. این مجموعه داده شامل کلیپ-های اول شخص (ویدیوها توسط دوربین عینک جمع‌آوری شده است) از فعالیت‌های مربوط به آشپزخانه است. این کلیپ‌ها که خودشان بخش‌هایی از ویدیو بزرگتر هستند، در 32 موضوع مختلف جمع‌آوری شده‌اند. به هر کلیپ یک برچسب موضوعی اختصاص داده شده است، این مجموعه داده شامل 19 کلاس افعال مختلف است (به عنوان مثال، "برداشتن"، "گذاشتن"، "بریدن" و غیره). ما مشابه [48] از این 19 دسته افعال مختلف استفاده می‌کنیم. **EGTEA Gaze+** [45] ذاتاً نامتوازن دسته‌ای است .شکل**4** توزیع داده مجموعه داده **EGTEA** را نشان داده شده است، همانطور که مشاهده می‌شود این مجموعه داده براساس افعال، نا متوازن و **long tail** است.



شکل**4** - توزیع داده افعال در مجموعه داده **EGTEA gaze+**

ما برای بررسی کارایی راهکار پیشنهادی روی مجموعه داده **EGTEA Gaze+**، از **pre trained 3D Resnext-101** به عنوان دسته‌بند پایه استفاده کردیم که قبلا با **Kinetics-400** آموزش دیده شده است، **learning rate** برابر 0.001 تنظیم شده است و **optimizer** مورد استفاده **SGD** است با **momentum**

برابر 0.9 و **weight decay** برابر **0.0005**. از سه تخمین‌گر استفاده شده که اولین تخمین‌گر از ابتدا و به تعداد 80 ایپوچ آموزش دیده است و فقط لایه انتهایی بقیه شبکه‌ها، که وزن‌های آن‌ها بوسیله یادگیری انتقالی از شبکه قبلی گرفته‌شده‌اند، نشان داده شده است، روش پیشنهادی ما با بدست آوردن 68.75 و 64.76 نسبت به دیگر تکنیک‌ها برتری دارد.

4-6. مجموعه داده‌ی تشخیص فعالیت‌های انسان توسط گوشی هوشمند

برای انجام این کار از مجموعه‌داده **Actitracker[46]** که بوسیله آزمایشگاه **WISDM** (**Wireless Sensor Data Mining**) جمع آوری شده است استفاده کرده ایم. این مجموعه داده شامل اطلاعات جمع آوری شده از 36 فرد است که بوسیله گوشی همراهی که در جیبشان قرار داده شده و با سرعت 20 نمونه بر ثانیه گرفته‌شده‌است. کاربران 6 فعالیت مختلف را در محیط کنترل شده انجام می‌دهند و داده‌ها درواقع مقادیر شتاب سه بعدی در راستای محورهای **x**، **y** و **z** است. این فعالیت‌ها شامل پایین رفتن از پله (**Downstairs**) ، دویدن (**Jogging**)، بالا رفتن از پله‌ها (**Upstairs**)، ایستادن (**Standing**)، نشستن (**Sitting**) و پیاده روی (**Walking**) است.

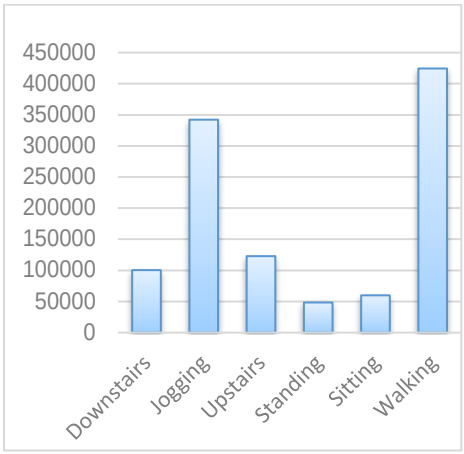
جدول**4** - مقایسه کارایی شبکه تجمعی پیشنهادی با دیگر تکنیک‌های مقابله با نامتوازنی داده‌ای روی مجموعه داده **EGTEA**

	Test accuracy	**F1-score**
CNN+CE	**67.41**	**63.21**
CNN +balanced CE	**66.86**	**63.32**
Voting CNN	**67.56**	**63.65**
Cascade CNN	**67.85**	**64.01**
روش پیشنهادیFENIL	**68.75**	**64.76**

هرکدام به تعداد 10 ایپوچ، آموزش دیده می‌شوند. کارایی روش پیشنهادی با معیارهای **F1-score** و دقت آزمون بررسی شده و در

جدول 5- ساختار **CNN** پیشنهادی برای دیتا ست **actitracker**

لایه‌ها	*تعداد فیلترها یا نرون‌ها*	*سایز کرنل*
1D Convolution ReLU	***100***	***10x1***
Max-Pooling	*--*	***3x1***
1D Convolution ReLU	***160***	***10x1***
Global-Average-Pooling Dropout 20%	*--*	*--*
Fully connected SoftMax	***6 Neurons***	*--*



شکل 5- توزیع دسته‌های مختلف در مجموعه‌داده **actitracker**

جدول 6–مقایسه نتایج راهکارهای مختلف مقابله با نامتوازنی داده‌ها روی مجموعه‌داده **actitracker**

	Test accuracy	Balanced_accuracy	F1-score
CNN+CE	96.25	90.14	88.77
CNN + weighted CE	96.42	91.86	88.65
Voting CNN	97.31	91.25	88.89
Cascade CNN	97.71	91.89	89.52
روش پیشنهادی	97.86%	92.90	90.01

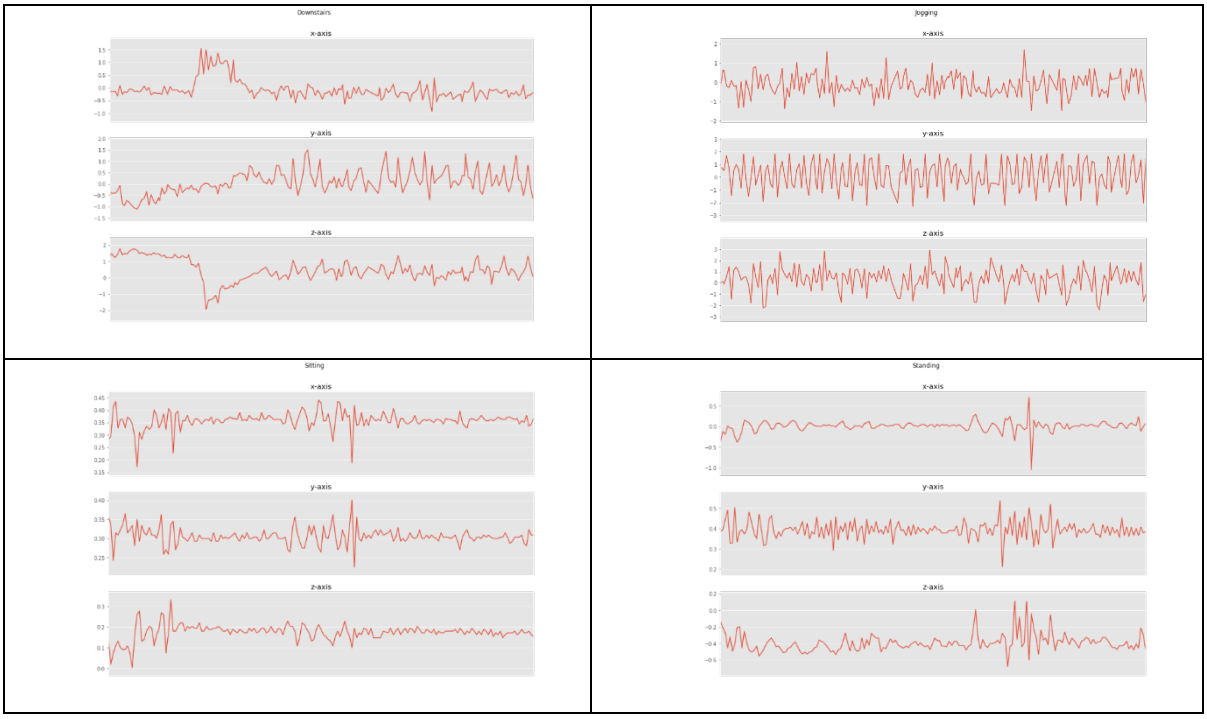
همانطور که درشکل 5 نشان داده شده است، مجموعه داده نسبت به فعالیت‌های انجام شده (برچسب‌های دسته) نامتوازن است. در ابتدا داده‌ها نرمال سازی شده و به توزیع نرمال استاندارد برده شده‌است، سپس این داده‌های استاندارد شده به برش‌های زمانی با اندازه پنجره 80 تقسیم می‌شوند که درواقع به مجموعه ای از قطعات 4 ثانیه ای داده تبدیل می‌شود. سپس این داده‌ها به طور تصادفی با نسبت 70 به 30 به مجموعه‌های آموزشی و تست تقسیم می‌شوند. نمونه‌ای از داده‌ها در شکل 6 نشان داده شده‌است.

داده‌های آموزشی به یک شبکه کانولووشنال یک بعدی داده شده است. ساختار شبکه در جدول 6 نشان داده‌شده‌است. این شبکه در ابتدا با یک لایه کانولوشنال یک بعدی با 100 فیلتر آغاز می‌شود و پس از آن یک لایه **Max-pooling 3×1** قرار گرفته، یک لایه کانولوشنال یک بعدی دیگر با 160 فیلتر به عنوان لایه سوم قرار گرفته است و پس از **average-pooling** و **dropout** 20 درصد، لایه تماما متصل با 6 نرون قرار گرفته است که یک تابع **SoftMax** در انتهای آن قرار داده شده است. بهینه ساز **Adam** با نرخ یادگیری 0.001 برای آموزش **CNN** استفاده‌شده‌است. برای پیاده‌سازی راهکار تجمعی پیشنهادی از 5 تخمین‌گر استفاده شده که اولین **CNN** 40 بار و **CNN** دوم 5 بار ، **CNN** سوم 3 بار و دو **CNN** آخر هرکدام یک اپوچ آموزش دیده‌اند. برای مقایسه راهکار پیشنهادی با تکنیک‌هایی که از فقط از یک **CNN** استفاده می‌کنند، 50 اپوچ در نظر گرفته شده است. در جدول 6 نتایج حاصل از هرکدام از راهکارها قرار داده شده است.

همانطور که از جدول 6 مشخص است با در نظر گرفتن دقت متوازن تست شبکه **FENIL** پیشنهادی، با فاصله 2/76، 1/04، 1/65 و 1/01 به ترتیب از آنتروپی متقابل و آنتروپی متقابل متوازن شده، رای گیری **CNN**ها و آبشار **CNN**ها (ترکیب **CNN** بدون در نظر گرفتن تابع ضرر کانونی) بهتر است و با توجه امتیاز **F1**، شبکه **FENIL** با کسب 90/01 بهترین امتیاز را کسب کرده است.

7. جمع‌بندی

تابع ضرر کانونی در کار با داده‌های نامتوازن در شناسایی اشیا بسیار موفق عمل کرده‌است اما زمانی که در مساله‌ی دسته‌بندی، تابع ضرر کانونی استفاده می‌شود، گرچه برای دسته اقلیت به‌خوبی عمل می‌کند اما کارایی آن در دسته اکثریت قابل قبول نیست، استفاده از گاماهای مختلف کارایی تابع ضرر کانونی بر روی داده‌های اقلیت و اکثریت را تغییر می‌دهد، انتخاب مقدار دقیق گاما به عنوان یک هایپرپارامتر به طوری که دقت در دسته اقلیت و اکثریت حفظ شود، مشکل است. در این مقاله شبکه تجمعی عمیق جدیدی بر اساس تابع ضرر کانونی با γهای مختلف پیشنهاد داده‌شده است که دسته‌بندهای پایه در این شبکه تجمعی، شبکه عصبی کانولووشنال است. به این ترتیب بازه‌ای از مقادیر گاما در آموزش شبکه استفاده می‌شوند. بنابراین شبکه‌ای ایجاد شده است که هم برای دسته اقلیت و هم برای دسته اکثریت کارایی خوبی دارد. در این شبکه، هر **CNN** نسبت به **CNN** قبلی به داده‌های آسان، اهمیت کمتری می‌دهد و تمرکز خود را روی اشتباهات شبکه قبلی بیشتر می‌کند، برای وزن دهی به داده-هایی که توسط **CNN** قبلی به درستی دسته بندی نشده‌اند از



شکل 6-نمونه‌هایی از داده‌های آموزشی نرمال شده.

الگوریتم **SAMME.R** استفاده شده است، و γ از یک **CNN** به **CNN** بعدی با نرخ α تغییر می‌کند. برای افزایش جریان داده به نفع دسته اقلیت در شبکه تجمعی، در اولین **CNN**، بایاس آخرین لایه طوری تنظیم می‌شود که خطای دسته‌بند در دسته اقلیت افزایش پیدا کند. در بخش 5 شبکه **FENIL** روی مجموعه داده‌های مختلف اجرا و نتایج بررسی شدند که بررسی نتایج کارایی شبکه **FENIL** روی داده‌های نامتوازن مختلف تصویری، ویدیو و سیگنال) را نشان می‌دهد. در این مقاله کارایی راهکار پیشنهادی با استفاده ار شبکه پایه **CNN** و در کاربرد دسته‌بندی داده‌ها در مجموعه داده‌های کوچک بررسی شده است به عنوان کارهای آتی پیشنهاد می‌شود کارایی روش در مجموعه های بزرگ‌تر و پیچیده‌تر بررسی شود و در ترکیب با دیگر راهکارهای مقابله با نامتوازنی داده‌ها میزان موفقیت آن سنجیده شود.

8. منابع

[1] F. Anowar and S. Sadaoui, "Incremental learning framework for real-world fraud detection environment," *Comput. Intell.*, vol. 37, no. 1, pp. 635-656, / 2021, doi: 10.1111/coin.12434.

[2] M. Xu and Y. Wang, "An Imbalanced Fault Diagnosis Method for Rolling Bearing Based on Semi-Supervised Conditional Generative Adversarial Network With Spectral Normalization," *IEEE Access*, vol. 9, pp. 27736-27747, / 2021, doi: 10.1109/ACCESS.2021.3058334.

[3] M. Zareapoor, P. Shamsolmoali, and J. Yang, "Oversampling Adversarial Network for Class-Imbalanced Fault Diagnosis," *CoRR*, vol. abs/2008.03071, / 2020. [Online]. Available: https://arxiv.org/abs/2008.03071.

[4] M. Tavallaee, N. Stakhanova, and A. A. Ghorbani, "Toward Credible Evaluation of Anomaly-Based Intrusion-Detection Methods," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 40, no. 5,

[13] D. Masko and P. Hensman, "The Impact of Imbalanced Training Data for Convolutional Neural Networks," 2015.

[14] H. Lee, M. Park, and J. Kim, "Plankton classification on imbalanced large scale database via convolutional neural networks with transfer learning," *2016 IEEE International Conference on Image Processing (ICIP),* pp. 3713-3717, 2016.

[15] N. Chawla, K. Bowyer, L. Hall, and W. Kegelmeyer, "SMOTE: Synthetic Minority Over-sampling Technique," *J. Artif. Intell. Res. (JAIR),* vol. 16, pp. 321-357, 01/01 2002, doi: 10.1613/jair.953.

[16] S. Pouyanfar *et al.*, "Dynamic Sampling in Convolutional Neural Networks for Imbalanced Data Classification," in *2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*, 10-12 April 2018 2018, pp. 112-117, doi: 10.1109/MIPR.2018.00027.

[17] M. Havaei *et al.*, "Brain tumor segmentation with Deep Neural Networks," *Medical Image Analysis,* vol. 35, pp. 18-31, 2017/01/01/ 2017, doi: https://doi.org/10.1016/j.media.2016.05.004.

[18] M. Buda, A. Maki, and M. A. Mazurowski, "A systematic study of the class imbalance problem in convolutional neural networks," *Neural Networks,* vol. 106, pp. 249-259, 2018/10/01/ 2018, doi: https://doi.org/10.1016/j.neunet.2018.07.011.

[19] S. Wang, W. Liu, J. Wu, L. Cao, Q. Meng, and P. J. Kennedy, "Training deep neural networks on imbalanced data sets," ed. Piscataway, NJ: Institute of Electrical and Electronics Engineers (IEEE), 2016, pp. 4368-4374.

[20] T. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal Loss for Dense Object Detection," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 22-29 Oct. 2017 2017, pp. 2999-3007, doi: 10.1109/ICCV.2017.324.

[21] S. H. Khan, M. Hayat, M. Bennamoun, F. A. Sohel, and R. Togneri, "Cost-Sensitive Learning of Deep Feature Representations From Imbalanced pp. 516-524, 2010, doi: 10.1109/TSMCC.2010.2048428.

[5] M. Rezaei, H. Yang, and C. Meinel, "Recurrent generative adversarial network for learning imbalanced medical image semantic segmentation," *Multim. Tools Appl.,* vol. 79, no. 21-22, pp. 15329-15348, / 2020, doi: 10.1007/s11042-019-7305-1.

[6] R. Singh, T. Ahmed, A. Kumar, A. K. Singh, A. K. Pandey, and S. K. Singh, "Imbalanced Breast Cancer Classification Using Transfer Learning," *IEEE ACM Trans. Comput. Biol. Bioinform.,* vol. 18, no. 1, pp. 83-93, / 2021, doi: 10.1109/TCBB.2020.2980831.

[7] M. Kubat, R. C. Holte, and S. Matwin, "Machine Learning for the Detection of Oil Spills in Satellite Radar Images," *Machine Learning,* vol. 30, no. 2, pp. 195-215, 1998/02/01 1998, doi: 10.1023/A:1007452223027.

[8] S. Yue, "Imbalanced Malware Images Classification: a CNN based Approach," 2017. [Online]. Available: http://arXiv.org/abs/.

[9] O. Sturman *et al.*, "Deep learning-based behavioral analysis reaches human accuracy and is capable of outperforming commercial solutions," *Neuropsychopharmacology,* vol. 45, no. 11, pp. 1942-1952, 2020/10/01 2020, doi: 10.1038/s41386-020-0776-y.

[10] G. E. A. P. A. Batista, R. C. Prati, and M. C. Monard, "A study of the behavior of several methods for balancing machine learning training data," *SIGKDD Explor. Newsl.,* vol. 6, no. 1, pp. 20–29, 2004, doi: 10.1145/1007730.1007735.

[11] R. Anand, K. G. Mehrotra, C. K. Mohan, and S. Ranka, "An improved algorithm for neural network classification of imbalanced training sets," *IEEE Transactions on Neural Networks,* vol. 4, no. 6, pp. 962-969, 1993, doi: 10.1109/72.286891.

[12] J. M. Johnson and T. M. Khoshgoftaar, "Survey on deep learning with class imbalance," *Journal of Big Data,* vol. 6, no. 1, p. 27, 2019/03/19 2019, doi: 10.1186/s40537-019-0192-5.

[29] Y. Freund and R. E. Schapire, "Experiments with a New Boosting Algorithm," 1996.

[30] J. Li, S. Fong, R. K. Wong, and V. W. Chu, "Adaptive multi-objective swarm fusion for imbalanced data classification," *Information Fusion,* vol. 39, pp. 1-24, 2018/01/01/ 2018, doi: https://doi.org/10.1016/j.inffus.2017.03.007.

[31] A. J. Ferreira and M. A. T. Figueiredo, "Boosting Algorithms: A Review of Methods, Theory, and Applications," in *Ensemble Machine Learning: Methods and Applications*, C. Zhang and Y. Ma Eds. Boston, MA: Springer US, 2012, pp. 35-85.

[32] T. J. Hastie, S. Rosset, J. Zhu, and H. Zou, "Multi-class AdaBoostB " *Statistics and Its Interface,* vol. 2, pp. 349-360, 2009.

[33] "ImageNet Large Scale Visual Recognition Challenge 2015. ImageNet." http://image-net.org/challenges/LSVRC/2015/ (accessed.

[34] D. Ciregan, U. Meier, and J. Schmidhuber, "Multi-column deep neural networks for image classification," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, 16-21 June 2012 2012, pp. 3642-3649, doi: 10.1109/CVPR.2012.6248110.

[35] H. Guo and H. L. Viktor, "Learning from imbalanced data sets with boosting and data generation: the DataBoost-IM approach," *ACM Sigkdd Explorations Newsletter,* vol. 6, no. 1, pp. 30–39, 2004, doi: 10.1145/1007730.1007736.

[36] I. Hwang, H.-M. Park, and J.-H. Chang, "Ensemble of deep neural networks using acoustic environment classification for statistical model-based voice activity detection," *Computer Speech & Language,* vol. 38, pp. 1-12, 2016.

[37] X. Zhou, L. Xie, P. Zhang, and Y. Zhang, "An ensemble of deep neural networks for object tracking," in *2014 IEEE International Conference on Image Processing (ICIP)*, 2014: IEEE, pp. 843-847.

Data," *IEEE Transactions on Neural Networks and Learning Systems,* vol. 29, no. 8, pp. 3573-3587, 2018, doi: 10.1109/TNNLS.2017.2732482.

[22] Q. Dong, S. Gong, and X. Zhu, "Imbalanced Deep Learning by Minority Class Incremental Rectification," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 41, no. 6, pp. 1367-1381, 2019, doi: 10.1109/TPAMI.2018.2832629.

[23] S. Ando and C. Y. Huang, "Deep Over-sampling Framework for Classifying Imbalanced Data," in *Machine Learning and Knowledge Discovery in Databases*, Cham, M. Ceci, J. Hollmén, L. Todorovski, C. Vens, and S. Džeroski, Eds., 2017// 2017: Springer International Publishing, pp. 770-785.

[24] Z. Zhang, L. Chen, Q. Liu, and P. Wang, "A Fraud Detection Method for Low-Frequency Transaction," *IEEE Access,* vol. 8, pp. 25210-25220, / 2020, doi: 10.1109/ACCESS.2020.2970614.

[25] H. Guo, Y. Li, J. Shang, G. Mingyun, H. Yuanyue, and G. Bing, "Learning from class-imbalanced data: Review of methods and applications," *Expert Syst. Appl.,* vol. 73, pp. 220-239, / 2017, doi: 10.1016/j.eswa.2016.12.035.

[26] B. Krawczyk, L. L. Minku, J. Gama, J. Stefanowski, and M. Wozniak, "Ensemble learning for data stream analysis: A survey," *Inf. Fusion,* vol. 37, pp. 132-156, / 2017, doi: 10.1016/j.inffus.2017.02.004.

[27] M. Galar, A. Fernández, E. B. Tartas, H. B. Sola, and F. Herrera, "A Review on Ensembles for the Class Imbalance Problem: Bagging-, Boosting-, and Hybrid-Based Approaches," *IEEE Trans. Syst. Man Cybern. Part C,* vol. 42, no. 4, pp. 463-484, / 2012, doi: 10.1109/TSMCC.2011.2161285.

[28] H. F. Jerome, "Greedy function approximation: A gradient boosting machine," *The Annals of Statistics,* vol. 29, no. 5, pp. 1189-1232, 10/1 2001, doi: 10.1214/aos/1013203451.

[38] C. Huang, Y. Li, C. C. Loy, and X. Tang, "Learning Deep Representation for Imbalanced Classification," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 27-30 June 2016 2016, pp. 5375-5384, doi: 10.1109/CVPR.2016.580.

[39] E. P. Ijjina and C. Krishna Mohan, "Hybrid deep neural network model for human action recognition," *Applied Soft Computing,* vol. 46, pp. 936-952, 2016/09/01/ 2016, doi: https://doi.org/10.1016/j.asoc.2015.08.025.

[40] M. Lyksborg, O. Puonti, M. Agn, and R. Larsen, "An Ensemble of 2D Convolutional Neural Networks for Tumor Segmentation," in *Image Analysis*, Cham, R. R. Paulsen and K. S. Pedersen, Eds., 2015// 2015: Springer International Publishing, pp. 201-211.

[41] N. Tajbakhsh, S. R. Gurudu, and J. Liang, "Automatic polyp detection in colonoscopy videos using an ensemble of convolutional neural networks," in *2015 IEEE 12th International Symposium on Biomedical Imaging (ISBI)*, 16-19 April 2015 2015, pp. 79-83, doi: 10.1109/ISBI.2015.7163821.

[42] J. Xie, B. Xu, and C. Zhang, "Horizontal and Vertical Ensemble with Deep Representation for Classification," *CoRR,* vol. abs/1306.2759, / 2013. [Online]. Available: http://arxiv.org/abs/1306.2759.

[43] M. Chen, C. Zhang, and S.-C. Chen, "Semantic Event Extraction Using Neural Network Ensembles," 2007. [Online]. Available: https://doi.org/10.1109/ICSC.2007.75

http://doi.ieeecomputersociety.org/10.1109/ICSC.2007.75.

[44] A. Krizhevsky, "Learning Multiple Layers of Features from Tiny Images," *University of Toronto,* 05/08 2012.

[45] Y. Li, M. Liu, and J. M. Rehg, "In the Eye of Beholder: Joint Learning of Gaze and Actions in First Person Video," in *Computer Vision – ECCV 2018*, Cham, V. Ferrari, M. Hebert, C. Sminchisescu, and Y. Weiss, Eds., 2018// 2018: Springer International Publishing, pp. 639-655.

[46] J. R. Kwapisz, G. M. Weiss, and S. A. Moore, "Activity Recognition Using Cell Phone Accelerometers," *SIGKDD Explor. Newsl.,* vol. 12, no. 2, pp. 74–82, 2011, doi: 10.1145/1964897.1964918.

[47] Y. Cui, M. Jia, T. Y. Lin, Y. Song, and S. Belongie, "Class-Balanced Loss Based on Effective Number of Samples," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 15-20 June 2019 2019, pp. 9260-9269, doi: 10.1109/CVPR.2019.00949.

[48] S. Sinha, H. Ohashi, and K. Nakamura, "Class-Wise Difficulty-Balanced Loss for Solving Class-Imbalance," in *Computer Vision – ACCV 2020*, Cham, H. Ishikawa, C.-L. Liu, T. Pajdla, and J. Shi, Eds., 2021// 2021: Springer International Publishing, pp. 549-565.