



دانشگاه صنعتی امیرکبیر
(پلی تکنیک تهران)
دانشکده مهندسی کامپیووتر

پایان نامه کارشناسی ارشد
گرایش هوش مصنوعی و رباتیکز

شناسایی فعالیت‌های انسانی در محیط‌های هوشمند با
استفاده از یادگیری خودناظارتی

نگارش
اردلان نهادوندی فرد

استاد راهنما
دکتر احسان ناظرفرد

شهریور ۱۴۰۴

بِسْمِ اللّٰهِ الرَّحْمٰنِ الرَّحِيْمِ

صفحه فرم ارزیابی و تصویب پایان نامه- فرم تأیید اعضاء کمیته
دفاع



دانشگاه صنعتی امیرکبیر
(پلی‌تکنیک تهران)

به نام خدا

تعهدنامه اصالت اثر

تاریخ: شهریور ۱۴۰۴

اینجانب اردلان نهادنی فرد متعهد می‌شوم که مطالب مندرج در این پایان‌نامه حاصل کار پژوهشی اینجانب تحت نظرارت و راهنمایی استادی دانشگاه صنعتی امیرکبیر بوده و به دستاوردهای دیگران که در این پژوهش از آنها استفاده شده است مطابق مقررات و روال متعارف ارجاع و در فهرست منابع و مأخذ ذکر گردیده است. این پایان‌نامه قبلاً برای احراز هیچ مدرک هم‌سطح یا بالاتر ارائه نگردیده است. در صورت اثبات تخلف در هر زمان، مدرک تحصیلی صادر شده توسط دانشگاه از درجه اعتبار ساقط بوده و دانشگاه حق پیگیری قانونی خواهد داشت.

کلیه نتایج و حقوق حاصل از این پایان‌نامه متعلق به دانشگاه صنعتی امیرکبیر می‌باشد. هرگونه استفاده از نتایج علمی و عملی، واگذاری اطلاعات به دیگران یا چاپ و تکثیر، ترجمه و اقتباس از این پایان‌نامه بدون موافقت کتبی دانشگاه صنعتی امیرکبیر ممنوع است. نقل مطالب با ذکر مأخذ بلامانع است.

اردلان نهادنی فرد

امضا

تقدیم به پدر، مادر و برادرم که چراغ راه من در تاریکی ها بودند و هرگز از حیات من دست نکشیدند.

و تقدیم به هراوه هم دل سخن نداشتم، که در تمام این مسیر پر فرازو نشیب، بترین همراه و حامی من بود و با بودنش به من قدرت
بنشید.

سپاسگزاری

لازم می‌دانم از تمامی عزیزانی که در طول دوره کارشناسی ارشد، از راهنمایی‌ها و حمایت‌های بی‌دریغ خود من را بهره‌مند ساختند، صمیمانه قدردانی کنم.
با کمال احترام و قدردانی، از استاد ارجمند، جناب آقای دکتر ناظرفرد، که با دانش و بینش خود، راهنمای مسیر علمی من بودند، سپاسگزاری می‌نمایم.

اردلان نساوندی فرد

شهریور ۱۴۰۴

چکیده

با گسترش روزافزون محیط‌های هوشمند و استفاده از حسگرهای مختلف مانند تلفن همراه، نیاز به سیستم‌هایی که بتوانند به صورت خودکار و دقیق فعالیت‌های انسانی را تشخیص دهنده، افزایش یافته است. یکی از چالش‌های اصلی در این حوزه، وابستگی شدید مدل‌های یادگیری ماشین به داده‌های برچسب‌خورده است که جمع‌آوری آن‌ها در مقیاس بزرگ، پرهزینه و زمانبر است. این موضوع، ضرورت بهره‌گیری از روش‌هایی را مطرح می‌سازد که بدون نیاز به برچسب‌گذاری دستی، بتوانند نمایش‌های مفهومی و قابل‌انتقال از داده‌های حسگر استخراج کنند. در این پژوهش، یک چارچوب یادگیری خودناظارتی طراحی شده که با بهره‌گیری از ترکیب دیدگاه‌های زمانی و فرکانسی، تلاش می‌کند نمایش‌هایی با کیفیت و عمومی از داده‌های خام فعالیت انسانی استخراج نماید. چارچوب پیشنهادی با هدف بهبود کیفیت بازنمایی داده، کاهش نیاز به داده‌های برچسب‌خورده، و افزایش قابلیت تعمیم‌پذیری مدل، در دو سناریوی مختلف مورد ارزیابی قرار گرفته است: نخست، آموزش و ارزیابی در یک محیط یکسان؛ و دوم، آموزش در یک محیط و ارزیابی در محیطی متفاوت، با هدف سنجش توانایی انتقال دانش و تعمیم‌پذیری مدل. نتایج حاصل از ارزیابی‌ها نشان می‌دهد که چارچوب ارائه شده در سناریوی آموزش مستقیم بر روی مجموعه داده HAPT به امتیاز F1 ۹۲.۳٪ (با ۰.۹٪ بهبود نسبت به مدل پایه) و در سناریوی یادگیری انتقالی از مجموعه داده MobiAct به HAPT به امتیاز F1 ۹۰.۲٪ (با ۱.۵٪ بهبود نسبت به مدل پایه) دست یافته است. این چارچوب توانسته با بهره‌گیری از ترکیب اطلاعات زمانی و فرکانسی، بازنمایی‌هایی استخراج کند که منجر به بهبود دقت در تشخیص فعالیت‌های انسانی شده‌اند. در مجموع، روش پیشنهادی گامی مؤثر در جهت کاهش وابستگی به داده‌های برچسب‌خورده و توسعه‌ی مدل‌های تعمیم‌پذیر برای کاربرد در محیط‌های متنوع و واقعی برداشته است.

واژه‌های کلیدی:

شناسایی فعالیت انسان، یادگیری خودناظارتی، تبدیل موجک، یادگیری تباینی، یادگیری انتقالی

فهرست مطالب

صفحه	عنوان
۱	۱ مقدمه
۷	۱-۱ دستاوردهای پژوهش
۷	۲-۱ ساختار مطالب
۸	۲ ادبیات موضوع و کارهای پیشین
۹	۲-۱ شناسایی فعالیت انسان
۱۰	۱-۱-۲ تعریف مسئله
۱۱	۲-۱-۲ سابقه پژوهش
۱۷	۲-۲ یادگیری خودنظرتی
۱۸	۱-۲-۲ تعریف یادگیری خودنظرتی
۲۰	۱-۱-۲-۲ فرمولبندی یادگیری خودنظرتی
۲۱	۲-۲-۲ سابقه پژوهش
۲۲	۱-۲-۲-۲ روش‌های زمینه محور
۲۷	۲-۲-۲-۲ روش‌های بازسازی محور
۳۱	۳-۲-۲-۲ یادگیری تباینی
۴۱	۴-۲-۲-۲ پردازش زبان طبیعی
۴۳	۳-۲ شناسایی فعالیت انسان با استفاده از یادگیری خودنظرتی
۴۴	۱-۳-۲ سابقه پژوهش
۴۴	۱-۱-۳-۲ شناسایی فعالیت مبتنی بر روش CPC
۴۵	۲-۱-۳-۲ شناسایی فعالیت مبتنی بر روش SimCLR
۴۶	۳-۱-۳-۲ شناسایی فعالیت مبتنی بر یادگیری مشارکتی
۴۸	۴-۲ جمعبندی
۵۰	۳ روش پیشنهادی
۵۱	۱-۳ تبدیل موجک
۵۳	۲-۳ روش پایه

۵۳	۱-۲-۳ کدگذار سیگنال
۵۵	۲-۲-۳ کدگذار اسکالولگرام
۵۶	۳-۲-۳ تنظیم دقیق مدل
۵۶	۳-۳ نوآوری‌های پیشنهادی
۵۷	۱-۳-۳ الگوریتم یادگیری تباینی SwAV
۵۸	۱-۱-۳-۳تابع هزینه پیش‌بینی تعویض شده
۶۳	۲-۱-۳-۳ راهبرد برش چندگانه
۶۴	۲-۳-۳ راهبرد داده‌افزایی
۶۶	۴-۳ جمع‌بندی
۶۷	۴ آزمایش‌ها و نتایج
۶۸	۱-۴ مجموعه داده
۶۸	۱-۱-۴ HAPT
۷۰	۲-۱-۴ MobiAct
۷۰	۲-۴ جزئیات پیاده‌سازی
۷۰	۱-۲-۴ پیش‌پردازش داده‌ها
۷۲	۲-۲-۴ آموزش مدل
۷۳	۳-۲-۴ معیارهای ارزیابی
۷۵	۴-۲-۴ ابیرپارامترها
۷۷	۳-۴ نتایج
۷۷	۱-۳-۴ نتایج آزمایش‌های درون‌مجموعه‌ای
۷۹	۲-۳-۴ نتایج آزمایش‌های بین‌مجموعه‌ای
۸۱	۴-۴ جزئیات اجرا
۸۲	۵-۴ جمع‌بندی
۸۴	۵ جمع‌بندی و نتیجه‌گیری و پیشنهادات
۸۵	۱-۵ جمع‌بندی
۸۶	۲-۵ کارهای آتی

فهرست تصاویر

صفحه

شکل

۱-۱	تفاوت داده‌های تولیدی توسط حسگر شتاب‌سنج قرار گرفته در نقاط مختلف بدن	۵
۱-۲	ساختار خودکدگذار عمیق بیش‌کامل	۱۴
۲-۲	معماری سیستم تشخیص فعالیت، ناهنجاری و پیش‌بینی فعالیت بعدی	۱۵
۳-۲	معماری سیستم توجه دوگانه بر روی داده‌های حسگر	۱۶
۴-۲	ساختار کلی سیستم‌های یادگیری خودناظارتی	۱۸
۵-۲	ساختار کلی سیستم‌های یادگیری خودناظارتی	۱۹
۶-۲	ساختار کلی شبکه‌ی پیش‌بینی چرخش	۲۳
۷-۲	ساختار کلی شبکه‌ی حل پازل	۲۵
۸-۲	ساختار کلی شبکه‌ی بررسی درستی ترتیب فریم‌های ویدیویی	۲۶
۹-۲	عملکرد کدگذار زمینه‌ای برای ترمیم تصاویر	۲۹
۱۰-۲	معماری کدگذار زمینه‌ای	۳۰
۱۱-۲	نمونه‌ای از فرایند یادگیری تباینی	۳۲
۱۲-۲	ساختار کلی روش CPC	۳۳
۱۳-۲	ساختار کلی روش MoCo	۳۶
۱۴-۲	روش‌های ایجاد داده‌ی افزوده	۳۸
۱۵-۲	دقت روش SimCLR	۴۰
۱۶-۲	انتخاب حسگرهای مثبت و منفی در ColloSSL	۴۷
۱-۳	نمونه اسکالوگرام برای یک موج متغیر در زمان	۵۳
۲-۳	معماری کلی پیش‌آموزش در روش پیشنهادی	۵۴
۳-۳	مقایسه‌ی الگوریتم SwAV (سمت راست) و SimCLR (سمت چپ)	۵۷
۴-۳	تأثیر ترتیب اعمال جایگشت و تبدیل موجک بر اسکالوگرام حاصل	۶۵
۱-۴	دسته‌های مختلف مجموعه داده HAPT	۶۹
۲-۴	تأثیر تبدیل موجک بر برد اسکالوگرام خروجی	۷۱

فهرست تصاویر

- ۳-۴ مقایسه امتیاز F1 (راست) و امتیاز کاپا (چپ) برای روش‌های مختلف بر اساس
درصدهای متفاوت داده‌های آموزشی. ۷۸
- ۴-۴ مقایسه امتیاز F1 (راست) و امتیاز کاپا (چپ) برای روش‌های مختلف بر اساس
درصدهای متفاوت داده‌های آموزشی. ۸۰

فهرست جداول

صفحه

جدول

۱-۲ مقایسه‌ی عملکرد روش پیش‌بینی چرخش	۲۴
۲-۲ دقت روش SimCLR در یادگیری انتقالی	۴۰
۳-۲ نتایج روش CPC در شناسایی فعالیت	۴۵
۴-۲ نتایج روش CSSHAR	۴۶
۱-۴ پیکربندی یادگیرنده‌ی سیگنال (سمت راست) و یادگیرنده‌ی اسکالوگرام (سمت چپ)	۷۶
۲-۴ نتایج عملکرد مدل در سناریوی درون‌مجموعه‌ای بر اساس درصدهای مختلف داده آموزشی	۷۸
۳-۴ نتایج عملکرد مدل در سناریوی بین‌مجموعه‌ای بر اساس درصدهای مختلف داده آموزشی	۸۰

فصل ۱

مقدمه

در سال‌های اخیر، با افزایش جمعیت سالمدان، فشارها بر روی مراکز بهداشتی و مراقبتی افزایش یافته است. این امر علاوه بر افزایش هزینه‌ها، می‌تواند باعث شود که نیروی انسانی نتواند به خوبی وظایف خود در مراقبت از افراد تحت مراقبت را به دلایل مختلفی مانند خستگی و یا فراموشی ایفا کند. به همین دلیل به سیستم‌های هوشمند مراقبتی نیاز است تا بتوانند بهتر، با هزینه‌ای کمتر و بهینه، وظیفه‌ی مراقبت از افراد کم‌توان را انجام دهند. وجود سیستم‌های هوشمند باعث می‌شود که سالمدان بدون نیاز به داشتن کمک از جانب افراد مختلف، بتوانند زندگی مستقل و در عین حال باکیفیتی را داشته باشند [۷].

پیشرفت‌ها در حوزه‌های متعدد فناوری مانند افزایش قدرت پردازشی ریزپردازنده‌ها، کاهش هزینه‌ی تولید و بهبود کیفیت خروجی حسگرها امکان ایجاد چنین سیستم‌هایی را فراهم کرده است. وظیفه‌ی کلی این سیستم‌ها این است که فعالیت افراد تحت مراقبت را در لحظه شناسایی کنند و در صورت لزوم به آن‌ها کمک کنند. به طور کلی فرایند شناسایی فعالیت‌های انسانی در بسیاری از موارد مانند خانه‌های هوشمند [۲]، بازی‌ها [۳]، محاسبات شهری [۴]، رباتیک [۵]، پیش‌بینی فعالیت بعدی^۱ [۱، ۶]، تشخیص ناهنجاری^۲ [۱، ۷] و اینترنت اشیا (IoT)^۳ [۸] کاربردهای بسیاری داشته‌اند و نتایج خوب و امیدوارکننده‌ای را از خود نشان داده‌اند.

روش‌های شناسایی فعالیت انسان^۴ را می‌توان بر اساس نوع داده‌ی مورد استفاده به دو دسته تقسیم کرد. دسته‌ی اول، شامل استفاده از داده‌های دوربین‌ها و پردازش تصویر و دسته‌ی دوم، شامل استفاده از داده‌های انواع حسگرها می‌باشد.

استفاده از داده‌های دوربین و تصاویر در محیط‌های عمومی کاربرد دارد. با استفاده از داده‌های دوربین به‌ویژه داده‌های دنباله‌ای از تصاویر پیوسته (مانند یک فیلم) می‌توان تعدادی فعالیت ابتدایی^۵ مانند جهت حرکت دست را کشف کرد و سپس با ترکیب این فعالیت‌ها، به عمل انجام شده توسط فرد پی‌برد [۹]. اما نکته‌ای که در رابطه با داده‌های تصویری وجود دارد این است که به دلایل مربوط به حریم شخصی، نمی‌توان از این داده‌ها برای هوشمندسازی شناسایی فعالیت در تمامی محیط‌ها، به‌ویژه خانه‌ها استفاده کرد چرا که ذخیره‌سازی اطلاعات تصاویر و پردازش آن‌ها حریم خصوصی افراد را مختل خواهد کرد. علاوه بر آن، دوربین‌ها نقاط کور دارند و برای عملکرد خوب دوربین‌ها نیاز است که در تمامی نقاط

¹Next Activity Prediction

²Anomaly Detection

³Internet of Things

⁴Human Activity Recognition

⁵Action Primitive

خانه دوربین قرار داده شود. اما این امر مقرر به صرفه نیست؛ چرا که علاوه بر هزینه‌ی زیاد قرار دادن تعداد زیادی دوربین در تمامی نقاط خانه، در تعدادی از نقاط خانه مانند حمام امکان قرار دادن دوربین وجود ندارد و در صورت رخ دادن هر گونه اتفاق غیرمنتظره‌ای برای افراد تحت مراقبت در این محیط‌ها، امکان شناسایی آن وجود ندارد و این امر خطرات جانی را به همراه خواهد داشت. به همین دلیل علیرغم عملکرد خوب روش‌های شناسایی فعالیت بر مبنای داده‌های تصویری^[۱۰]، بهتر است از روش‌هایی برای شناسایی فعالیت استفاده شود که قابل استفاده در انواع محیط‌های هوشمند، بهویژه خانه‌ها که افراد (بهخصوص سالمدان) زمان زیادی از روز را در آن سپری می‌کنند، باشد.

به همین دلیل، به روش‌های شناسایی فعالیت مبتنی بر داده‌های حسگرها روی می‌آوریم. روش‌های مبتنی بر داده‌های حسگر، خروجی یک یا مجموعه‌ای از حسگرها را به عنوان داده‌ی ورودی در نظر می‌گیرند و بر روی آن‌ها پردازش انجام می‌دهند. حسگرهای مورد استفاده می‌توانند از نوع محیطی^[۱۱]، پوشیدنی^[۱۲] و یا هر دو^[۱۳] باشند. وظیفه‌ی حسگرهای محیطی اندازه‌گیری تغییرات در محیط مانند لمس اجسام مختلف، تغییرات دما و مجاورت اجسام می‌باشد. از طرفی حسگرهای پوشیدنی حرکات کاربر را با دقت بهتری می‌توانند بررسی کنند و جزئیات بهتری از نحوه و جهت حرکت کاربر را به ما بدهند. حسگرهای حرکتی شتاب‌سنج^[۳]، ژیروسکوپ^[۴] و لختی^[۵] از جمله حسگرهای پوشیدنی می‌باشند. علاوه بر آن، امروزه به دلیل پیشرفت‌هه شدن تلفن‌های همراه، می‌توان به سادگی از یک تلفن همراه به عنوان انواع حسگرهای پوشیدنی به طور همزمان استفاده کرد^[۱۴]، و با توجه به این که در عصر حاضر کمتر کسی از تلفن‌های همراه هوشمند استفاده نمی‌کند، به سادگی و با کمترین هزینه‌ی سخت‌افزاری می‌توان فعالیت اشخاص را شناسایی کرد.

یادگیری عمیق^[۶] به دلیل دقت بالایی که از خود نشان داده است^[۱۵] (مانند حافظه کوتاه مدت بلند LSTM^[۷] و شبکه‌ی عصبی پیچشی^[۸]، به روش اصلی مسائل مربوط به شناسایی فعالیت انسان در سال‌های اخیر تبدیل شده است. این روش‌ها با بهره‌گیری از توابع فعال‌ساز غیرخطی^[۹] و الگوریتم

¹Ambient Sensors

²Wearable Sensors

³Accelerometer

⁴Gyroscope

⁵Inertia

⁶Deep Learning

⁷Long Short-Term Memory

⁸Convolutional Neural Network

⁹Non-linear Activation Functions

پس انتشار خطا^۱[۱۶] این امکان را فراهم می‌کنند که مدل‌های بسیار عمیق و دارای پارامترهای با تعداد بالا آموزش داده شوند که با انتخاب معماری مناسب و داده‌ی برچسب‌گذاری شده‌ی کافی، امکان کشف روابط پیچیده در داده‌ی ورودی را به شبکه‌های عصبی عمیق می‌دهد؛ بر خلاف روش‌های سنتی یادگیری ماشین که یا بدون پارامتر هستند (مانند روش K-نزدیک‌ترین همسایه (KNN)^۲) یا در صورت داشتن پارامتر، دارای تعداد پارامتر قابل یادگیری بسیار کمی هستند (مانند روش رگرسیون لجستیک^۳ که دارای تنها $d+1$ پارامتر هستند که d بیانگر تعداد ابعاد ورودی است). به همین دلیل شبکه‌های عصبی با داشتن تعداد زیادی پارامتر قابل یادگیری و پیچیدگی بالا، می‌توانند عملکرد بهتری نسبت به روش‌های سنتی یادگیری ماشین نشان دهند.

اما همانطور که اشاره شد، مدل‌های یادگیری عمیق نظارت شده^۴ برای عملکرد خوب خود نیازمند حجم زیادی داده‌ی برچسب‌گذاری شده می‌باشند. در صورت کافی نبودن داده‌های آموزشی، عملکرد مدل افت خواهد کرد و مدل دچار بیش‌برازش^۵ یا کمبرازش^۶ (بسته به پیچیدگی مدل و ماهیت داده‌ها) خواهد شد. علاوه بر آن، حتی هنگامی که داده‌های کافی برای آموزش وجود داشته باشد، مدل‌های یادگیری عمیق هنگامی عملکرد خوبی از خود نشان خواهند داد که توزیع آماری داده‌های آموزشی و ارزیابی شباهت زیادی با یکدیگر داشته باشند. در واقع قدرت تعمیم^۷ مدل‌های یادگیری عمیق نظارت شده به طور کلی خیلی بالا نیست و در شرایطی که داده‌های ارزیابی شبیه به داده‌های آموزشی نباشند، این مدل‌ها افت عملکرد خواهند داشت[۱۷]. امری که در داده‌های دنیای واقعی کاملاً مشهود است. برای مثال در وظیفه‌ی شناسایی فعالیت انسان، داده‌ی حسگرها برای افراد سالم‌مند و افراد جوان و کودکان و حتی افراد مختلف با سن و شرایط جسمانی یکسان تفاوت‌های بسیاری دارند چرا که نحوه انجام فعالیت‌ها توسط افراد مختلف متفاوت است و منجر به تولید داده‌ها از توزیع‌های متنوع می‌شود. همینطور با قرار دادن حسگر در نقاط مختلف بدن یا چرخش جهت حسگر (مثلاً قرار دادن تلفن همراه در خلاف جهت قرار داده شده در مجموعه داده) توزیع داده‌های تولیدی توسط حسگرها می‌تواند تغییراتی کند که منجر به افت کیفیت مدل در تولید خروجی شود[۱۸].

علاوه بر مشکل تعمیم مدل بر روی داده‌های جدید، چالش برچسب‌گذاری داده‌ها نیز وجود دارد.

¹Backpropagation Algorithm

²K-Nearest Neighbors

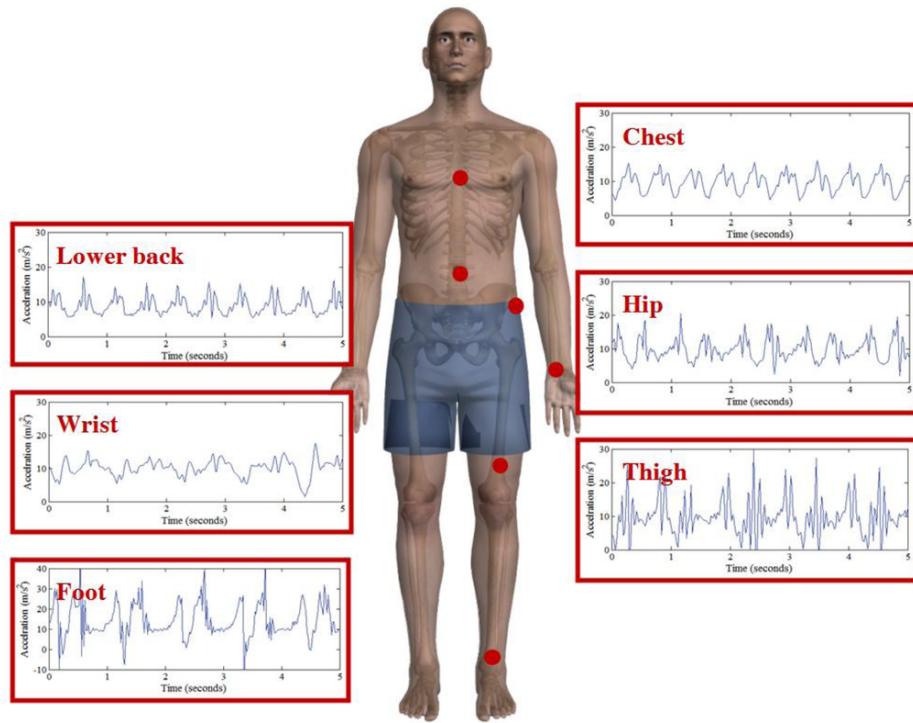
³Logistic Regression

⁴Supervised

⁵Overfitting

⁶Underfitting

⁷Generalization



شکل ۱-۱: تفاوت داده‌های تولیدی توسط حسگر شتاب‌سنج قرار گرفته در نقاط مختلف بدن

چرا که برای برچسب‌گذاری داده‌های خام، نیاز به نیروی انسانی وجود دارد. برای مثال، مجموعه داده‌ی ImageNet شامل ۱.۳ میلیون تصویر از ۱۰۰۰ کلاس مختلف می‌باشد که هر کدام از این تصاویر کاملاً توسط نیروی انسانی برچسب خورده‌اند [۱۹]. طبیعتاً برچسب‌گذاری این داده‌ها کاری بسیار طاقت‌فرسا و پرهزینه می‌باشد؛ اما برچسب‌گذاری داده‌های تصویری به مراتب ساده‌تر از برچسب‌گذاری داده‌های حسگرها می‌باشد. چرا که داده‌های تصویری برای افراد آشنا هستند و اشخاص می‌توانند با نگاه به تصاویر متوجه شوند که تصویر چه برچسبی را خواهد داشت. اما برای داده‌های حسگرها نمی‌توان صرفاً از روی داده‌ی خام حسگر، فعالیت انجام شده‌ی مربوطه را به دست آورد. بنابراین برای برچسب‌گذاری داده‌ی حسگرها بایستی یک متخصص در تمام مدت آزمایش حرکات شخص مورد آزمایش را زیر نظر بگیرد و برچسب‌گذاری کند. این امر باعث می‌شود که برای تولید حجم زیادی داده‌ی برچسب‌دار برای مسئله‌ی شناسایی فعالیت انسان، زمان و هزینه‌ی بسیار زیادی صرف شود و علاوه بر آن داده‌ها در محیط آزمایشگاهی که با دنیای واقعی متفاوت هستند تولید شوند.

با توجه به دلایل ذکر شده، نیاز به روش‌هایی است که وابستگی مدل‌ها را به داده‌های برچسب‌خورده کاهش داده و در عین حال قدرت تعمیم آن‌ها را افزایش دهند. یکی از رویکردهای نوین برای رسیدن

به این هدف، بهره‌گیری از یادگیری خودناظارتی^۱ (که زیرمجموعه‌ای از یادگیری بدون نظارت^۲ است) می‌باشد. در این رویکرد، مدل با استفاده از ساختارهای درونی و الگوهای موجود در داده‌های بدون برچسب، پیش‌وظایفی را تحت عنوان وظایف پوششی^۳ اجرا می‌کند و مدل از آن‌ها برای یادگیری بازنمایی‌های^۴ مفید بهره می‌برد. این پیش‌وظایف می‌تواند شامل مواردی مانند شناسایی چرخش [۲۰]، روش‌های مبتنی بر ایجاد داده مانند تکمیل بخش‌های حذف شده از داده [۲۱] و روش‌های مبتنی بر یادگیری تباینی^۵ [۲۲، ۲۳، ۲۴] باشند. همچنین در بسیاری از این روش‌ها، به منظور تقویت اثربخشی فرایند یادگیری، از تکنیک‌های داده‌افزایی^۶ (مانند ایجاد نسخه‌های متغیر از یک نمونه‌ی داده با حفظ معنای کلی آن از طریق روش‌هایی مانند افزودن نویز و ایجاد برش بر روی داده‌ها) بهره گرفته می‌شود. با انجام این وظایف و حل این‌گونه مسائل، یک تابع هزینه نیز برای مدل تعریف می‌شود که مدل در طی فرایند کمینه کردن این تابع هزینه، بازنمایی‌های خوبی از داده یاد می‌گیرد که از وزن‌های تصادفی اولیه‌ی مدل عملکرد بسیار بهتری دارند. سپس با استفاده از این بازنمایی‌ها می‌توان در مراحل بعدی برای حل وظایف دارای برچسب (مانند دسته‌بندی فعالیت‌ها) مدل را تنظیم دقیق^۷ کرد و عملکرد قابل قبولی را حتی با حجم کمی از داده‌های برچسب‌خورده به دست آورد [۲۰]. استفاده از یادگیری خودناظارتی، نه تنها هزینه‌های مربوط به برچسب‌گذاری را به‌طور چشم‌گیری کاهش می‌دهد، بلکه با بهره‌گیری از حجم زیاد داده‌های بدون برچسب، قابلیت تعمیم مدل را نیز در مواجهه با داده‌هایی از توزیع‌های متفاوت افزایش می‌دهد. بدین صورت که می‌توان مدل را بر روی یک مجموعه داده‌ی بسیار بزرگ اما بدون برچسب پیش‌آموزش داد و سپس به‌وسیله‌ی یادگیری انتقالی^۸، بر روی مجموعه داده‌ی کوچک تنظیم دقیق انجام داد. تحقیقات نشان داده‌اند که پیش‌آموزش مدل به‌روش خودناظارتی و سپس تنظیم دقیق بر روی تنها بخش کوچکی از مجموعه داده‌ی مقصد (یادگیری با نمونه‌ی اندک^۹، عملکرد بهتری را نسبت به پیش‌آموزش به روش نظارت‌شده نشان می‌دهد [۲۵، ۲۶]). این ویژگی به‌ویژه در مسئله‌ی شناسایی فعالیت انسان که در آن تنوع بالایی در نحوه‌ی انجام فعالیت وجود دارد و همچنین کوچک‌ترین تغییراتی مانند وجود حیوان خانگی می‌تواند عملکرد مدل را مختل کند، از اهمیت بالایی برخوردار است. از این

¹Self-Supervised Learning

²Unsupervised Learning

³Pretext tasks (Auxiliary tasks)

⁴Representation

⁵Contrastive Learning

⁶Data Augmentation

⁷Fine-tune

⁸Transfer Learning

⁹Few-shot learning

رو، استفاده از یادگیری خودناظارتی به عنوان یک راهکار جایگزین یا مکمل یادگیری نظارت شده گامی مؤثر در جهت ارتقای کارایی مدل‌ها در شرایط دنیای واقعی برداشته است.

۱-۱ دستاوردهای پژوهش

دستاوردهای کلی این پژوهش شامل موارد زیر می‌شوند:

- استفاده از یک رویکرد یادگیری خودناظارتی تباینی بر مبنای خوشبندی^۱
- استفاده از تبدیل موجک^۲ برای بهره‌گیری از مولفه‌های فرکانسی داده
- بهبود روش‌های تولید داده‌ی افزوده برای تبدیل موجک
- آزمایش و ارزیابی مدل ارائه شده در یادگیری انتقالی

۲-۱ ساختار مطالب

در این پایان‌نامه، فصل دوم به مرور پیشینه پژوهش و روش‌هایی که تاکنون در حوزه شناسایی فعالیت انسان، یادگیری خودناظارتی و کاربردهای یادگیری خودناظارتی در شناسایی فعالیت انسان ارائه شده‌اند اختصاص دارد. فصل سوم به معرفی مدل پیشنهادی این پژوهش پرداخته و ساختار مدل، الگوریتم‌های به کاررفته و نحوه استفاده از آن‌ها بررسی می‌گردد. در فصل چهارم، نتایج حاصل از آزمایش‌های انجام شده با استفاده از روش پیشنهادی در مقایسه با روش پایه ارائه می‌گردد و با تحلیل این نتایج، فصل خاتمه می‌یابد. در نهایت، فصل پنجم به جمع‌بندی و نتیجه‌گیری و ارائه‌ی پیشنهاداتی برای کارهای آینده اختصاص دارد.

¹Clustering

²Wavelet Transform

فصل ۲

ادبیات موضوع و کارهای پیشین

فصل دوم: ادبیات موضوع و کارهای پیشین

در این فصل ابتدا به معرفی و بررسی مسئله‌ی شناسایی فعالیت‌های انسانی و پژوهش‌های انجام شده در این زمینه می‌پردازیم. در ادامه، به بررسی روش‌های یادگیری خودناظارتی، کاربرد آن‌ها در حوزه‌ی شناسایی فعالیت انسان، و پژوهش‌های مرتبط در این زمینه خواهیم پرداخت.

شناسایی فعالیت‌های انسانی یکی از مسائل مهم و پرکاربرد در حوزه‌های مختلف از جمله سلامت، خانه‌های هوشمند و پایش رفتار کاربران بهشمار می‌رود. همان‌گونه که در فصل مقدمه نیز بیان شد، دو رویکرد کلی برای این مسئله وجود دارد: رویکرد مبتنی بر داده‌های تصویری و رویکرد مبتنی بر داده‌های حسگری. در این پژوهش، تمرکز اصلی بر استفاده از داده‌های حسگرها بوده و از این رو، روش‌های انتخاب شده نیز بر پایه‌ی این نوع داده‌ها طراحی و ارزیابی شده‌اند.

۱-۲ شناسایی فعالیت انسان

پیشرفت‌های مختلف در تکنولوژی، باعث شده‌اند که حسگرها با هزینه‌ی اندک تولید شوند و داده‌های تولید شده توسط آن‌ها با سرعت بالا پردازش شوند که در نتیجه‌ی آن کار با حسگرها در عمل ساده می‌شود و باعث تحقیقات متعددی در این حوزه شده است. علاوه بر آن، سیستم‌های هوشمند برای عملکرد مناسب نیازمند این هستند که فعالیت انجام شده توسط کاربر شناسایی شود تا سیستم بتواند به خوبی کار خود را انجام دهد. در ادامه چند مورد از کاربردهای کلی شناسایی فعالیت انسان در دنیای واقعی را شرح می‌دهیم:

• سیستم‌های مراقبتی

در بیشتر سیستم‌های سنتی مراقبتی، بیماران و افراد تحت مراقبت بایستی ارزیابی‌های دوره‌ای را انجام دهند که این موضوع علاوه بر زمان بر و هزینه بر بودن، دقیق نیست. چرا که این ارزیابی‌های دوره‌ای تنها وضعیت بیمار در لحظه را مورد سنجش قرار می‌دهند و ممکن است در یک زمانی بیمار مراجعه کند که علائم به خوبی دیده نشوند. به دلایل ذکر شده، سیستم‌های مراقبتی و پزشکی استقبال گسترده‌ای از روش‌های شناسایی فعالیت مبتنی بر انواع حسگرها کرده‌اند و تلاش‌ها در راستای هوشمندسازی هر چه بیشتر سیستم‌های مراقبتی ادامه دارد.

• دستیار زندگی در خانه‌های هوشمند

از سیستم‌های شناسایی فعالیت می‌توان برای مراقبت از بیماران یا سالمندان در منزل و همچنین به عنوان دستیار زندگی بهره گرفت. برای مثال، سامانه‌ای هوشمند برای شناسایی فعالیت توسط

الاقباری و همکاران طراحی شده که علاوه بر شناسایی فعالیت‌های روزمره، قابلیت تشخیص ناهنجاری‌ها^۱ (مانند زمین خوردن فرد یا هر گونه اختلال در داده‌های حسگرها) و پیش‌بینی فعالیت بعدی (مثلًاً پیش‌بینی ورود فرد به اتاق خواب و فعال‌سازی سیستم تهویه) را نیز دارد. چنین سیستمی می‌تواند به عنوان یک دستیار زندگی در خانه‌های هوشمند عملکرد مؤثری از خود نشان دهد [۱].

۱-۱-۲ تعریف مسئله

برای مسئله‌ی شناسایی فعالیت انسان دو تعریف کلی می‌توان ارائه داد.
تعریف اول:

فرض کنید مجموعه‌ای به صورت $\{s_0, s_1, \dots, s_{k-1}\} = S$ در اختیار داریم که شامل k دنباله زمانی از اندازه‌گیری‌های مربوط به ویژگی‌های مختلف است. این داده‌ها در بازه‌ی زمانی $I = [t_\alpha, t_\omega]$ ثبت شده‌اند. در مسئله‌ی تشخیص فعالیت، هدف این است که بازه‌ی زمانی I را به زیربازه‌هایی مانند I_0, I_1, \dots, I_{r-1} تقسیم کنیم و برای هر زیربازه، برچسبی که بیانگر نوع فعالیت است اختصاص دهیم، به‌گونه‌ای که این برچسب‌گذاری با داده‌های S تطابق داشته باشد.

بر اساس این تعریف، زیربازه‌ها باید ناتهی و بدون همپوشانی باشند و کل بازه زمانی I را پوشش دهند، یعنی $I = \bigcup_{i=0}^{r-1} I_i$. بنابراین، فرض می‌شود که در هر لحظه تنها یک فعالیت در حال انجام است. هرچند این فرض در برخی شرایط دنیای واقعی ممکن است برقرار نباشد. برای نمونه، یک فرد ممکن است همزمان در حال آشپزی و مصرف دارو باشد. این مدل ساده، چارچوبی پایه برای طراحی سیستم‌های تشخیص فعالیت فراهم می‌کند.

تعریف دوم:

مجموعه‌ای از m پنجره زمانی به صورت $\{w_0, w_1, \dots, w_{m-1}\} = W$ را در نظر بگیرید، که هر پنجره‌ی w_i شامل توالی زمانی از مشاهدات سنسورها است. برای هر پنجره‌ی زمانی، داده‌های مربوطه با مجموعه‌ای مانند $\{s_{i,0}, \dots, s_{i,k-1}\} = S_i$ نمایش داده می‌شوند. همچنین فرض می‌کنیم مجموعه‌ای از برچسب‌ها تحت عنوان $\{l_0, \dots, l_{m-1}\} = L$ موجود است که هر عنصر آن نشان‌دهنده‌ی یک نوع فعالیت در بازه‌ی متناظر با یک پنجره‌ی زمانی می‌باشد.

در این صورت، می‌خواهیم تابعی تعریف کنیم به صورت $L \xrightarrow{f} S_i$ که بتواند با دریافت داده‌های

¹Anomaly

فصل دوم: ادبیات موضوع و کارهای پیشین

هر پنجره S_i ، مناسب‌ترین فعالیت از L را انتخاب کند. در واقع، این تابع تلاش می‌کند تشخیص دهد که در پنجره‌ی زمانی w کدام فعالیت غالب بوده است. اگرچه این شیوه به فرض غالب بودن یک فعالیت در هر پنجره استوار است، اما با در نظر گرفتن همپوشانی جزئی یا نویز در داده‌ها، می‌تواند تخمینی مناسب از واقعیت ارائه دهد.

نکته‌ی مهم اینجاست که اگر دو پنجره زمانی پشت سر هم باشند، برچسب آن‌ها ممکن است مشابه باشد یا حتی در مرز پنجره‌ها، فعالیتی مشترک رخ دهد. با این حال، این روش باعث می‌شود مسئله به صورت برچسب‌گذاری دنباله‌ای از پنجره‌ها ساده‌سازی شده و امکان آموزش مدل‌های یادگیری ماشین فراهم گردد.

در مجموع، این مدل به ما اجازه می‌دهد با وجود پیچیدگی‌های ذاتی رفتارهای انسانی، از طریق تحلیل پنجره‌ای و نگاشت آن به برچسب‌ها، فرایند تشخیص فعالیت را قابل پیاده‌سازی و آموزش‌پذیر سازیم. این روش به‌ویژه در محیط‌های دارای داده‌های زیاد و پیوسته که فعالیت‌ها با یکدیگر همپوشانی دارند، بسیار کاربردی و موثر خواهد بود.

رویکرد کلی برای حل مسئله‌ی شناسایی فعالیت بدین صورت است که ابتدا نیاز به یک مجموعه داده‌ی برچسب‌دار داریم. سپس با استفاده از این مجموعه داده‌ی برچسب‌دار، سعی می‌کنیم ویژگی‌هایی را استخراج کنیم و از آن‌ها برای آموزش مدل استفاده کنیم. بدین منظور چالش‌های متعددی بوجود می‌آیند که برای حل آن‌ها رویکردهای مختلفی توسط محققان ارائه شده‌اند که در ادامه به بررسی تعدادی از آن‌ها می‌پردازیم.

۲-۱-۲ ساقه پژوهش

در زمینه‌ی شناسایی فعالیت‌های انسانی در کاربردهای مختلف کارهای متعددی انجام شده است. هم‌لا و همکاران [۲۷] حسگرهای پوشیدنی را به کفشهای ۳۴ شخص مختلف دارای بیماری پارکینسون^۱ متصل کردند. وظیفه‌ی این حسگرهای اندازه‌گیری سرعت و شیوه راه رفتن افراد بود. سپس با استخراج ویژگی‌ها از روی مقادیر خام تولیدی توسط حسگرهای ماشین بولتزمن محدود (RBM^۲) اقدام به تشخیص بیماری پارکینسون در این افراد کردند.

یک کاربرد دیگر سیستم‌های شناسایی فعالیت که به آن اشاره کردیم، استفاده از آن برای دستیار زندگی و نظارت در خانه‌ی هوشمند می‌باشد. می‌توان از این سیستم‌ها برای مراقبت از بیماران مبتلا

¹Parkinson's Disease

²Restricted Boltzmann Machine

فصل دوم: ادبیات موضوع و کارهای پیشین

به بیماری فراموشی^۱ و سالمندان استفاده کرد. محققان یک روش بر مبنای مدل‌های پنهان مارکوف^۲ ارائه دادند که می‌تواند فعالیت‌های فرد ساکن خانه را شناسایی کند و موارد اضطراری و موارد مربوط به سلامت را گزارش دهد^[۲۸].

به طور کلی، روش‌های مربوط به شناسایی فعالیت‌های انسانی به روش‌های مبتنی بر یادگیری ماشین و یادگیری عمیق روی آورده‌اند. مدل‌های یادگیری عمیق بسیاری برای شناسایی فعالیت پیشنهاد شده است و این مدل‌ها دقیق خوبی را در آموزش با داده‌های برچسب‌دار به اندازه‌ی کافی ارائه می‌کنند^[۲۹]. علاوه بر آن، روش‌های یادگیری عمیق در کاربردهای استخراج ویژگی در مسائل شناسایی فعالیت که داده‌های ورودی دارای ابعاد بالایی هستند به کار گرفته شده‌اند. روش‌های داده محور^[۳۰] و مدل محور چندوجهی^[۳۱] دو روش کاربرد مدل عمیق در مسائل تشخیص فعالیت هستند.

یک راهکار که در گذشته به نتایج خوبی دست یافت، استفاده از روش‌های سنتی یادگیری ماشین است. روش کا-نژدیک‌ترین همسایه (KNN) اگرچه یک روش ساده است، اما برای مثال توسط فوستر و همکاران^[۳۲] برای شناسایی وضعیت بدن و جهت حرکت با استفاده از داده‌های حسگر به کار گرفته شد و نتایج خوبی از خود نشان داد. از دیگر روش‌های مورد استفاده نیز می‌توان مدل‌های پنهان مارکوف^[۲۸]، جنگل تصادفی^[۳۳] و ماشین بردار پشتیبان (SVM)^[۳۴] را نام برد که در دسته‌بندی و شناسایی فعالیت‌های انسانی دارای عملکرد نسبتاً خوبی هستند. مزیت این روش‌ها این است که برخلاف روش‌های مبتنی بر یادگیری عمیق، نیازمند تعداد داده‌ی بسیار زیادی نیستند و با تعداد داده‌ی کم می‌توانند نتایج خوبی را از خود نشان دهند^[۱۵]. اما مشکلی که این روش‌ها دارند این است که نیازمند متخصص برای استخراج ویژگی دستی^[۶] هستند و علاوه بر نیاز به متخصص، ویژگی‌های استخراج شده به اندازه‌ی کافی انتزاعی^[۷] و پرکاربرد نیستند. بنابراین نمی‌توان به این روش‌ها در دنیای امروز که توزیع و ابعاد داده‌ها از پیچیدگی بالایی برخوردار هستند اکتفا کرد.

با پیشرفت شبکه‌های عصبی و فراهم شدن حجم زیاد داده‌ی برچسب‌دار و همه‌گیر شدن روش‌های مبتنی بر یادگیری عمیق، در سال‌های اخیر روش‌های شناسایی فعالیت نیز به استفاده از شبکه‌های عصبی روی آورده‌اند. چرا که داده‌ها در حوزه‌ی شناسایی فعالیت‌های انسانی عموماً پیچیده و دارای

¹Alzheimer's Disease

²Hidden Markov Models

³Multi-modal

⁴Random Forest

⁵Support Vector Machine

⁶Handcrafted Features

⁷Abstract

ابعاد بالا و در برخی موارد چند وجهی هستند. محققان نشان دادند که با در نظر گرفتن داده‌ها بصورت چندوجهی و ترکیب داده‌های انواع مختلفی از حسگرهای بهتری نسبت به حالتی که تنها از یک حسگر (مانند یک شتاب‌سنج) استفاده شده است می‌توان دست یافت^[۳۴]. علاوه بر آن، شبکه‌های عصبی عمیق به دلیل ساختار چند لایه و پیچیده‌ای که دارند، هر لایه می‌تواند ویژگی‌های مختلفی را از داده استخراج کند و نتیجتاً امکان کار با داده‌های پیچیده‌تر را برایمان فراهم می‌کنند. برای استخراج ویژگی توسط شبکه‌های عصبی دیگر نیاز چندانی به مهندسی ویژگی به صورت پیچیده و کاملاً دستی نخواهیم داشت و با ساختارهای خاصی می‌توان ویژگی‌ها را به صورت کاملاً خودکار استخراج کرد. برای مثال، از شبکه‌های پیچشی یک بعدی و دو بعدی می‌توان در استخراج ویژگی‌ها و وابستگی‌های مکانی^[۱] (مثلاً در تصاویر) و همینطور از شبکه‌های بازگشتی^[۲] مانند حافظه‌ی کوتاه‌مدت بلند (LSTM)^[۳] یا واحد بازگشتی دروازه‌ای (GRU)^[۴] در استخراج ویژگی‌ها و وابستگی‌های زمانی^[۵] استفاده کرد.

الاقباری و همکاران^[۱] در مقاله‌ی خود^[۳] روش مختلف برای شناسایی فعالیت، شناسایی ناهنجاری و پیش‌بینی فعالیت ارائه دادند. آن‌ها برای هر فعالیت، یک مجموعه از R ویژگی را تعریف می‌کنند که این ویژگی‌ها شامل موارد زیر می‌باشند:

۱. زمانی که برای انجام شدن فعالیت سپری شده است.

۲. تعداد حسگرهایی که در طول انجام فعالیت فعال بوده اند.

۳. تعداد دفعاتی که این فعالیت در طول روز انجام شده است.

۴. وضعیت (روشن/خاموش) تمام حسگرها

با استفاده از تمامی این ویژگی‌ها مسئله به جای تحلیل سری به یک مسئله‌ی ساده‌تر تبدیل می‌شود که توسط یک پرسپترون چند لایه (MLP)^[۶] عملیات دسته‌بندی و شناسایی فعالیت انجام می‌شود. در همین حین که شبکه‌ی پرسپترون چند لایه مشغول شناسایی فعالیت است، یک شبکه‌ی خودکدگذار بیش‌کامل (OCDAE)^[۷] وظیفه‌ی شناسایی ناهنجاری را دارد. بدین صورت که همان R ویژگی به این شبکه

¹Spatial Dependencies

²Recurrent Neural Network

³Long Short-Term Memory

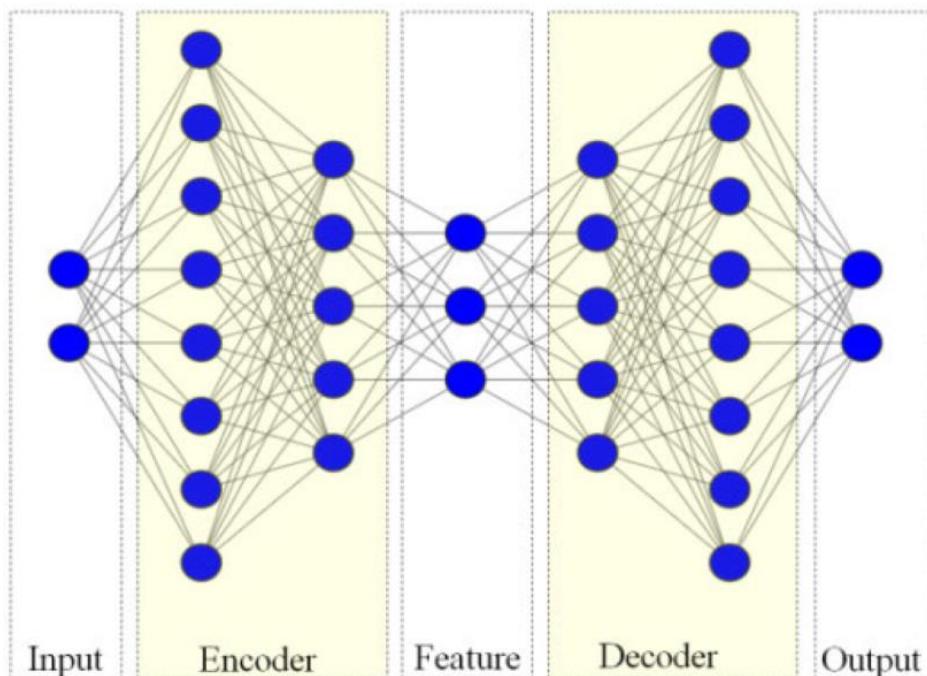
⁴Gate Recurrent Unit

⁵Temporal Dependencies

⁶Multi-Layer Perceptron

⁷Over-Complete Deep Auto-Encoder

نیز داده می‌شوند و این شبکه سعی می‌کند که خروجی را از روی ورودی بازسازی کند. اگر که خطای بازسازی^۱ کم بود، بدین معناست که فعالیت مربوطه مانند یکی از فعالیت‌هایی است که شبکه قبلاً بر روی آن آموزش دیده و در غیر اینصورت یک ناهنجاری شناسایی می‌شود. در عین حال که این^۲



شکل ۱-۲: ساختار خودکدگذار عمیق بیش‌کامل

شبکه کار می‌کنند، اگر که ناهنجاری شناسایی نشود خروجی شبکه‌ی شناسایی کننده‌ی فعالیت به عنوان فعالیت فعلی در نظر گرفته می‌شود و سپس توسط یک شبکه‌ی حافظه‌ی کوتاه‌مدت بلند، فعالیت بعدی پیش‌بینی می‌شود. شکل کلی ساختار پیاده شده در این مقاله به فرم شکل ۲-۲ می‌باشد.

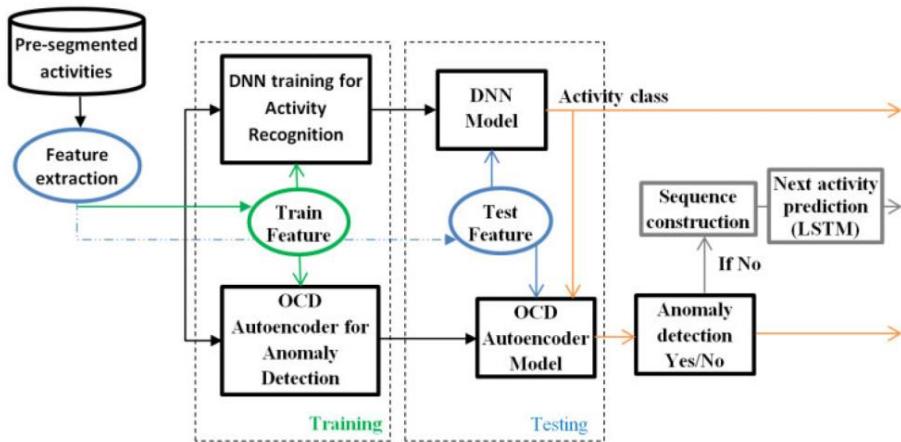
گائو و همکاران^۳^۴ [۲۵] یک روش بر مبنای سازوکار توجه^۲ و توجه دوگانه^۳ ارائه دادند. سازوکار توجه به شبکه‌ی عصبی این امکان را می‌دهد که به مرور یاد بگیرد که به کدام بخش‌های داده توجه بیشتری نشان دهد. برای مثال سازوکار توجه در داده‌ی تصویری به شبکه این امکان را می‌دهد که به بخش‌های مهم‌تر تصویر اهمیت بیشتری نشان بدهد و شبکه بتواند ویژگی‌های مهم‌تری را استخراج کند. توجه دوگانه بدین صورت عمل می‌کند که سازوکار توجه به طور همزمان برای دو نوع داده (مثلاً تصویر و داده‌ی دنباله‌ای مانند متن) به کار گرفته شود.

بدین ترتیب گائو و همکاران از توجه دوگانه برای مسئله‌ی شناسایی فعالیت انسان استفاده کردند.

¹Reconstruction Error

²Attention Mechanism

³Dual Attention



شکل ۲-۲: معماری سیستم تشخیص فعالیت، ناهنجاری و پیش‌بینی فعالیت بعدی

روش ارائه شده چند بخش اصلی دارد:

۱. ورودی داده‌های سنسورها: سیگنال‌های چند-کانالی حسگرها (مانند شتاب‌سنج و ژیروسکوپ) به عنوان ورودی به مدل داده می‌شوند.

۲. استخراج ویژگی با شبکه‌های پیچشی: با عبور داده‌ها از چندین لایه شبکه‌ی پیچشی، ویژگی‌های سطح پایین و میانی استخراج می‌شوند.

۳. مازول توجه دوگانه: پس از استخراج ویژگی، ویژگی‌های به‌دست‌آمده به مازول توجه دوگانه داده می‌شوند. این مازول شامل:

- **توجه کانالی^۱:** محاسبه‌ی وزن برای هر کanal از ویژگی‌ها به‌منظور تعیین اهمیت سنسورها و ویژگی‌های مختلف و تقویت ویژگی‌های مهم.

- **توجه زمانی^۲:** محاسبه‌ی وزن برای هر گام زمانی به‌منظور تمرکز روی بازه‌های زمانی مهم در طول فعالیت.

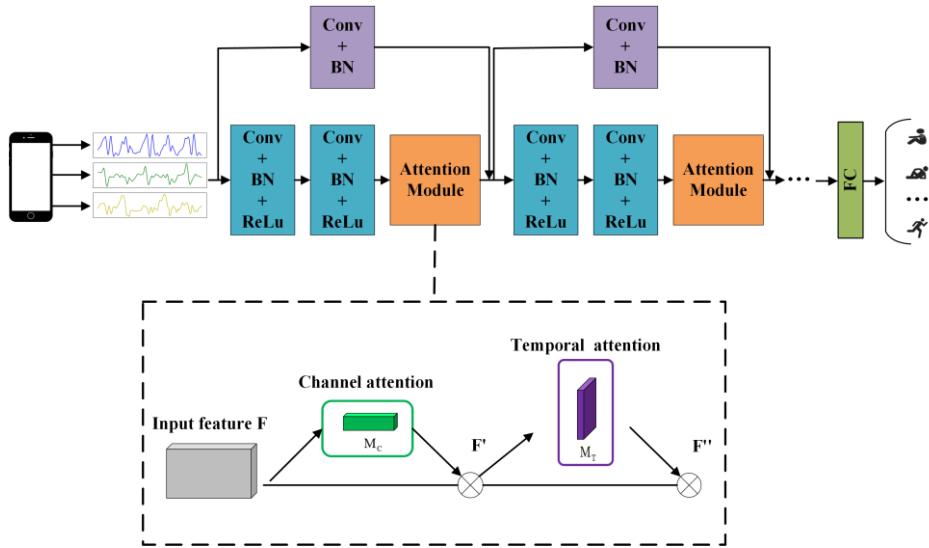
ویژگی‌های خروجی از این دو توجه به‌ترتیب روی ویژگی‌ها ضرب می‌شوند و بازنمایی غنی‌شده‌ای از داده تولید می‌شود.

۴. تکرار فرایند استخراج ویژگی و توجه: فرایند استخراج ویژگی و مازول توجه دوگانه یک بار دیگر تکرار می‌شود تا ویژگی‌های سطح بالاتری استخراج شوند.

¹Channel Attention

²Temporal Attention

۵. لایه‌های دسته‌بندی: در نهایت، ویژگی‌های استخراج و وزن‌دهی شده به لایه‌های تماماً متصل^۱ داده می‌شوند تا فعالیت مربوطه پیش‌بینی شود.



شکل ۲-۳: معماری سیستم توجه دوگانه بر روی داده‌های حسگر

وانگ و همکاران^[۳۶] یک روش مبتنی بر ترکیب شبکه پیچشی و شبکه حافظه کوتاه‌مدت بلند ارائه کردند که به دقت بالایی دست یافت. در این روش، از داده‌های مربوط به سری زمانی دو حسگر^۳ کاناله شتاب‌سنج و ژیروسکوپ ابتدا پنجره‌های لغزان^۴ دارای همپوشانی استخراج می‌شوند و سپس داده‌های هر ۶ کanal حسگرها به صورت یک تصویر در کنار هم قرار می‌گیرند. بدین صورت یک تصویر به طول پنجره‌ی لغزان و عرض ۶ خواهیم داشت. سپس بر روی این شبه تصاویر لایه‌های پیچشی اعمال می‌شوند و آرایه‌ای از این شبه تصاویر توسط لایه‌های پیچشی پردازش می‌شوند تا ویژگی‌های هر پنجره که با دیگر پنجره‌ها همپوشانی دارد استخراج شوند. سپس این دنباله ویژگی‌های استخراج شده به یک شبکه حافظه کوتاه‌مدت بلند داده می‌شود که خروجی آن مجموعه‌ای از بردارها است که شامل وابستگی‌های زمانی و توالی فعالیت‌ها می‌باشد. این بردارها به لایه‌ی تماماً متصل داده می‌شوند تا ویژگی‌های کلی فعالیت با یکدیگر ترکیب^۳ شده و به صورت یکپارچه استخراج شوند. ضمناً جهت پایداری بیشتر یادگیری و دستیابی به دقت بالاتر، نویسنده‌گان مقاله از نرمال‌سازی دسته‌ای^۴ استفاده کردند.

¹Fully Connected

²Sliding Windows

³Fusion

⁴Batch Normalization

۲-۲ یادگیری خودناظارتی

همانطور که در بخش قبل بررسی کردیم، شبکه‌های عصبی عمیق در مواردی که داده‌ها از پیچیدگی بالایی برخوردار هستند و ابعاد داده‌های ورودی بالا هستند، شدیداً از روش‌های سنتی یادگیری ماشین عملکرد بهتر و قدرتمندتری دارند و نتایج دقیق‌تری را از خود نشان می‌دهند. در واقع می‌توان گفت که در کاربردهای پیشرفته‌ی دنیاً امروز استفاده از یادگیری عمیق به روش‌های یادگیری ماشین سنتی در اکثر مواقع ترجیح داده می‌شوند. اما چالش اصلی روش‌های مبتنی بر یادگیری عمیق نیاز شدید این روش‌ها به حجم زیادی داده برچسب‌گذاری شده می‌باشد. آنچه که در دنیاً امروز شدیداً فراوان و در دسترس است، انواع داده بدون برچسب می‌باشد و به‌طور کلی جمع‌آوری داده‌ی خام کاری نسبتاً ساده و کم‌هزینه می‌باشد. اما برچسب‌گذاری داده‌ها کاری شدیداً پرهزینه و زمان‌بر می‌باشد. در واقع یکی از اصلی‌ترین گلوگاه‌های^۱ آموزش شبکه‌های عصبی عمیق، جمع‌آوری داده‌ی آموزشی دارای برچسب می‌باشد.

علاوه بر هزینه‌ها و چالش‌های ناشی از برچسب‌گذاری داده‌ها، یادگیری تحت نظارت دارای مشکلاتی مانند خطای تعمیم^۲، همبستگی‌های کاذب^۳ و برچسب‌گذاری‌های غلط غیر عمدى و یا عمدى ناشی از حملات خصم‌مانه^۴ می‌باشد [۳۷].

با توجه به تمامی چالش‌های ذکر شده، ضروری است که به سراغ روش‌هایی برویم که وابستگی ما را به مجموعه داده‌های برچسب‌دار کاهش دهند. هدف اصلی، دستیابی به قابلیت اجرای مانند دسته‌بندی تنها با اتکا به حجم به مراتب کمتری از داده‌های برچسب‌خورده است.

برای تحقق این هدف، رویکردهایی مانند یادگیری خودناظارتی بسیار کارآمد هستند. روش‌های مبتنی بر یادگیری خودناظارتی به ما اجازه می‌دهند تا از ظرفیت عظیم داده‌های بدون برچسب که به وفور یافت می‌شند و دسترسی به آن‌ها کم‌هزینه‌تر است بهره ببریم. در این شیوه، مدل ابتدا با استفاده از داده‌های خام و بدون برچسب، یک درک پایه‌ای و غنی از ساختار و ویژگی‌های داده‌ها پیدا می‌کند. سپس این مدل پیش‌آموخته را می‌توان با مقدار بسیار اندکی از داده‌های برچسب‌دار برای عملیات نهایی مورد نظر تنظیم دقیق کنیم. این رویکرد نه تنها باعث صرفه‌جویی چشمگیری در هزینه و زمان برچسب‌زنی می‌شود، بلکه به مدل اجازه می‌دهد تا با یادگیری از گسترده وسیع‌تری از داده‌ها، به تعمیم‌پذیری و عملکرد بهتری

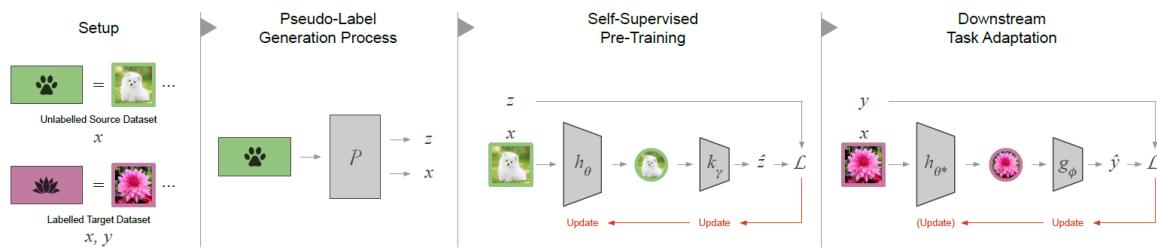
¹Bottleneck

²Generalization Error

³Spurious Correlations

⁴Adversarial Attacks

دست یابد.



شکل ۲-۴: ساختار کلی سیستم‌های یادگیری خودناظارتی

۱-۲-۲ تعریف یادگیری خودناظارتی

پیش از ارائه یک تعریف برای یادگیری خودناظارتی، برای درک بهتر مفاهیم بکار گرفته شده در این پایان‌نامه به بیان برخی از اصطلاحات که در این حوزه رایج هستند می‌پردازیم:

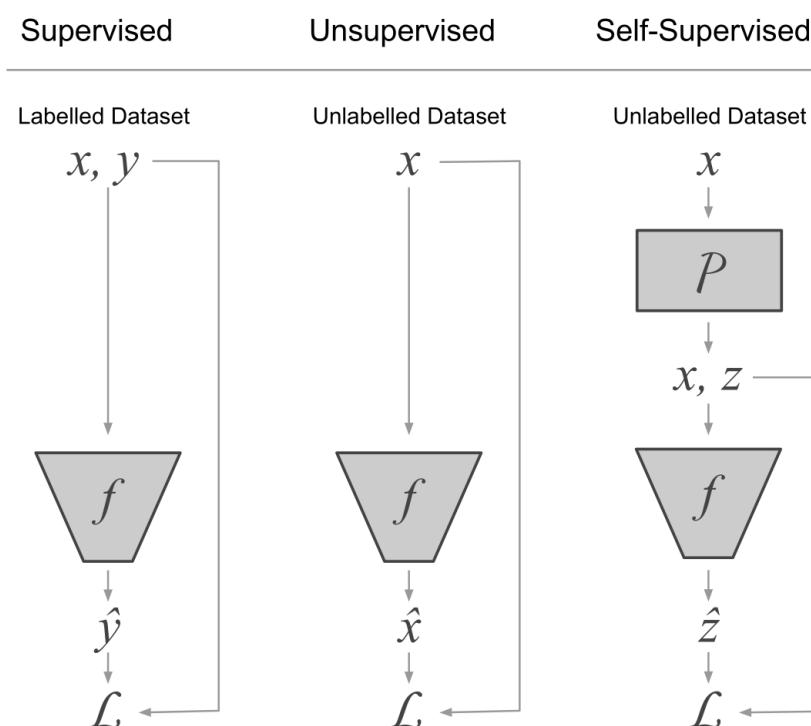
- **شبه برچسب:** در حقیقت، شبه برچسب‌ها برچسب‌هایی هستند که معمولاً به صورت خودکار و بر اساس ویژگی‌های داده‌ها در فاز اول آموزش (پیش‌آموزش) برای هر داده تولید می‌شوند. برای مثال در وظیفه پوششی شناسایی میزان چرخش تصویر، شبه برچسب مربوطه میزان چرخش اعمال شده به این تصویر می‌باشد.

- **وظیفه پوششی:** وظیفه پوششی در واقع وظیفه‌ای است که برای اجرای فاز پیش‌آموزش خودناظارتی طراحی شده و هدف آن یادگیری ویژگی‌های سطح بالا را روی داده‌های خام با کمک شبه برچسب‌ها می‌باشد. این وظیفه در حقیقت معماری شبکه و نحوه یادگیری ویژگی‌ها در فاز پیش‌آموزش خودناظارتی را تعیین می‌کند.

- **وظیفه پایین‌دستی^۱:** وظیفه پایین‌دستی در واقع همان وظیفه اصلی است که پس از فاز پیش‌آموزش خودناظارتی انجام می‌شود که به‌طور کلی به دو منظور انجام می‌شود. هدف اول، ارزیابی کیفیت ویژگی‌های استخراج شده توسط شبکه‌ی پیش‌آموزش دیده و هدف دوم، آموزش نهایی مدل برای هدف اصلی (مثلاً شناسایی فعالیت انسان یا دسته‌بندی تصاویر) می‌باشد. در واقع وظیفه پایین‌دستی شامل وظایف مستقل از پیش‌آموزش خودناظارتی است که مدل پیش‌آموزش دیده شده را به صورت کاربردی مورد ارزیابی قرار می‌دهد و آن را برای کاربردهای دنیای واقعی آماده می‌کند.

¹Downstream Task

یادگیری خودناظارتی، یک رویکرد یادگیری ماشین است که در آن مدل بدون استفاده از برچسب‌های داده‌ها و مجموعه داده‌ی برچسب‌دار، از داده‌های بدون برچسب برای یادگیری بازنمایی‌های مفید استفاده می‌کند. در این روش، با ایجاد شبه برچسب‌ها از داده‌های ورودی بدون برچسب (مثلًا حذف بخشی از داده و تلاش برای بازسازی آن) یک وظیفه پوششی تعریف می‌کنیم تا مدل بتواند ساختارها و الگوهای درونی داده را یاد بگیرد. سپس این بازنمایی‌های آموخته شده می‌توانند برای حل وظایف پایین‌دستی اصلی مانند دسته‌بندی مورد استفاده قرار بگیرند. نکته‌ی حائز اهمیت در اینجا این است که با تعداد بسیار کمتری داده‌ی برچسب‌دار می‌توان آموزش مدل را انجام داد و به قدرت تعمیم بالایی دست یافت.



شکل ۲-۵: ساختار کلی سیستم‌های یادگیری خودناظارتی

همانطور که در شکل ۵-۲ دیده می‌شود، یادگیری خودناظارتی در اصل نوعی از یادگیری بدون ناظارت است [۳۸]. یادگیری بدون ناظارت شامل الگوریتم‌هایی مانند انواع الگوریتم‌های خوشبندی^۱، شبکه‌ی مولد تخصصی (GAN)^۲ و خودکدگزار متغیر (VAE)^۳ می‌باشد. یادگیری خودناظارتی نیز در این موضوع که مجموعه داده بدون برچسب می‌باشد با یادگیری بدون ناظارت مشترک است. اما معماری کلی و بهینه‌سازی^۴ مدل‌های یادگیری خودناظارتی به روش‌های یادگیری ناظارت شده نزدیک‌تر است. چرا که

¹Clustering

²Generative Adversarial Network

³Variation Auto-Encoder

⁴Optimization

سیگنال نظارت را توسط شبیه برچسبها و وظایف پوششی اعمال می‌کنیم و با شبیه برچسبها می‌توان دقیقاً مانند یک برچسب واقعی رفتار کنیم و عمل دسته‌بندی و بهینه‌سازی مدل را با محاسبه خطای دسته‌بندی انجام دهیم.

۱-۱-۲-۲ فرمول بندی یادگیری خودنظراتی

در یادگیری خودنظراتی، برخلاف یادگیری با نظارت که به داده‌های جفت‌شده‌ی X_i و Y_i نیاز دارد (که Y_i توسط نیروی انسانی برچسب‌گذاری می‌شود)، از برچسب‌هایی استفاده می‌شود که به صورت خودکار و بدون نیاز به مداخله انسانی تولید می‌شوند. این برچسب‌های خودکار یا شبیه برچسب‌ها (P_i) مستقیماً از ویژگی‌های درونی داده‌ها (مانند تصاویر یا داده‌های سری زمانی) و با استفاده از قواعد و طراحی الگوریتمی مناسب استخراج می‌گردند. بنابراین با داشتن N نمونه داده‌ی آموزشی،تابع هزینه در یادگیری با نظارت طبق رابطه‌ی زیر محاسبه می‌شود [۳۹]:

$$loss(D) = \min_{\theta} \frac{1}{N} \sum_{i=1}^N loss(f(X_i), Y_i) \quad (1-2)$$

اما در فاز نخست یادگیری خودنظراتی که با هدف یادگیری بازنمایی‌های غنی از داده‌ها و استخراج ویژگی‌های باکیفیت بدون نیاز به برچسب انسانی انجام می‌شود،تابع هزینه‌ی رابطه‌ی ۱-۲ به شکل زیر تغییر پیدا می‌کند:

$$loss(D) = \min_{\theta} \frac{1}{N} \sum_{i=1}^N loss(f(X_i), P_i) \quad (2-2)$$

در این معادلات، $f(X_i)$ به این معنا است که ابتدا خروجی شبکه برای X_i محاسبه می‌گردد و سپس مقدار آن با برچسب‌ها یا شبیه برچسب‌ها مقایسه می‌شود و هزینه محاسبه می‌گردد. این فرایند باعث می‌شود که شبکه به تدریج پارامترهای خود را به گونه‌ای تنظیم نماید که خروجی‌های تولید شده بیشترین تطابق را با برچسب‌های موجود داشته باشند و در نتیجه مدل قادر به یادگیری الگوهای موثر و معنادار از داده‌های ورودی گردد.

پس از پایان این مرحله، مدل آموزش‌دیده در فاز پیش‌آموزش خودنظراتی آماده می‌شود تا در

مرحله‌ی بعدی، یعنی آموزش با نظارت در وظایف پایین‌دستی مورد استفاده قرار گیرد. در این مرحله، از بازنمایی‌های یادگرفته‌شده توسط مدل برای بهبود کارایی و کاهش نیاز به داده‌های برچسب‌خورده‌ی فراوان استفاده می‌شود و فرایند یادگیری انتقالی به اجرا درمی‌آید.

۲-۲-۲ ساقه پژوهش

به‌طور کلی کارهای انجام شده در حوزه‌ی یادگیری خودناظارتی را می‌توان بر حسب ماهیت وظیفه‌ی پوششی مورد استفاده به چند دسته‌ی کلی تقسیم‌بندی کرد:

۱. **روش‌های زمینه‌محور^۱:** در این دسته از روش‌ها، هدف مدل، یادگیری روابط میان اجزای مختلف داده با استفاده از زمینه‌ی محلی یا سراسری آن است. این روش‌ها معمولاً بر پایه‌ی روش‌هایی مانند پیش‌بینی موقعیت نسبی بخش‌های داده، ترتیب وقوع رویدادها، یا ویژگی‌های ساختاری مانند شناسایی جهت چرخش یا ترتیب قطعات بنا می‌شوند. از آنجا که این وظایف معمولاً نیاز به بازسازی کامل داده ندارند و تنها از اطلاعات ضمنی در خود داده استفاده می‌کنند، پیاده‌سازی نسبتاً ساده‌تری دارند و در حوزه‌هایی نظیر پردازش تصویر و تحلیل سیگنال کاربرد گسترده‌ای یافته‌اند.

۲. **روش‌های بازسازی‌محور^۲:** در این روش‌ها، شبکه تلاش می‌کند تا ورودی ناقص، دارای نویز یا کدگذاری‌شده را بازسازی کند. این دسته شامل خانواده‌ی گسترده‌ای از روش‌های مولد^۳ نیز می‌شود، از جمله خوکدگذارها، حذف نویز، رنگ‌سازی تصاویر، و پر کردن بخش‌های حذف‌شده از داده. تمرکز اصلی این رویکردها بر حفظ اطلاعات کامل از ورودی در بازنمایی‌های آموخته‌شده است. چنین بازنمایی‌هایی معمولاً ظرفیت بالایی برای انتقال به وظایف پایین‌دستی مانند طبقه‌بندی یا تشخیص دارند.

۳. **روش‌های برچسب معنایی‌محور^۴:** در این رویکردها، هدف از وظیفه‌ی پوششی، پیش‌بینی برچسب‌هایی است که به صورت خودکار و بدون دخالت انسان از داده استخراج شده‌اند، اما نمایانگر مفاهیم سطح بالای معنایی هستند. برای نمونه، دسته‌بندی داده‌ها بر اساس خوشه‌بندی

¹Context-based

²Reconstruction-based

³Generative

⁴Semantic Label-based

فصل دوم: ادبیات موضوع و کارهای پیشین

بازنمایی‌های اولیه یا پیش‌بینی ویژگی‌هایی که نمایانگر ساختار مفهومی داده هستند. در واقع، این روش‌ها سعی می‌کنند با اجرای الگوریتم‌هایی، یک یا چند برچسب برای داده‌ها استخراج کنند و یادگیری را با استفاده از این برچسب‌ها انجام می‌دهیم. به همین دلیل، معمولاً از مکانیزم‌هایی مانند خوشبندی، خودتقطیر^۱، یا یادگیری مبتنی بر نماینده‌ها بهره می‌برند. در این پایان‌نامه با جزئیات به آن نمی‌پردازیم اما نمونه‌ای از کاربرد این روش را می‌توان در مقاله‌ی ارائه شده توسط دتون و همکاران [۴۰] مشاهده نمود.

۴. روش‌های تباینی: این دسته از روش‌ها بر اساس اصل نزدیکسازی نمونه‌های مشابه و دورسازی نمونه‌های ناسازگار از یکدیگر عمل می‌کنند. در این رویکرد، نمونه‌های مثبت (مانند دو نمایش^۲ مختلف از یک داده) باید در فضای بازنمایی به یکدیگر نزدیک شوند و نمونه‌های منفی (مانند دو نمایش مختلف از دو داده م مختلف) از هم فاصله بگیرند. این مکانیزم باعث می‌شود که مدل، بازنمایی‌هایی مقاوم نسبت به تغییرات بی‌اهمیت یاد بگیرد. روش‌های تباینی نقش مهمی در موققتی یادگیری خودنظرارتی مدرن داشته‌اند و پایه‌گذار بسیاری از مدل‌های پیشرفتی در حوزه‌های تصویر، ویدیو و سیگنال هستند.

در ادامه، به بررسی دقیق‌تر هریک از دسته‌های یادگیری خودنظرارتی معرفی شده در بالا پرداخته و نمونه‌هایی از روش‌های بر جسته در هر دسته معرفی می‌شوند.

۱-۲-۲-۲ روش‌های زمینه‌محور

در روش‌های زمینه‌محور، ایده‌ی اصلی استفاده از اطلاعات زمینه‌ای موجود در خود داده برای تعریف یک وظیفه‌ی یادگیری است. این وظایف معمولاً بر پایه‌ی روابط مکانی، زمانی یا ساختاری میان اجزای مختلف یک نمونه شکل می‌گیرند. چنین روش‌هایی با بهره‌گیری از ساختار درونی داده، سعی در استخراج بازنمایی‌هایی دارند که بتوانند موقعیت، ترتیب، یا سایر روابط میان اجزا را درک کنند. این دسته از روش‌ها به‌ویژه در حوزه‌های بینایی ماشین و تحلیل سیگنال، نقطه‌ی آغاز پژوهش‌های جدی در یادگیری خودنظرارتی بوده‌اند و هنوز هم کاربرد گسترده‌ای دارند.

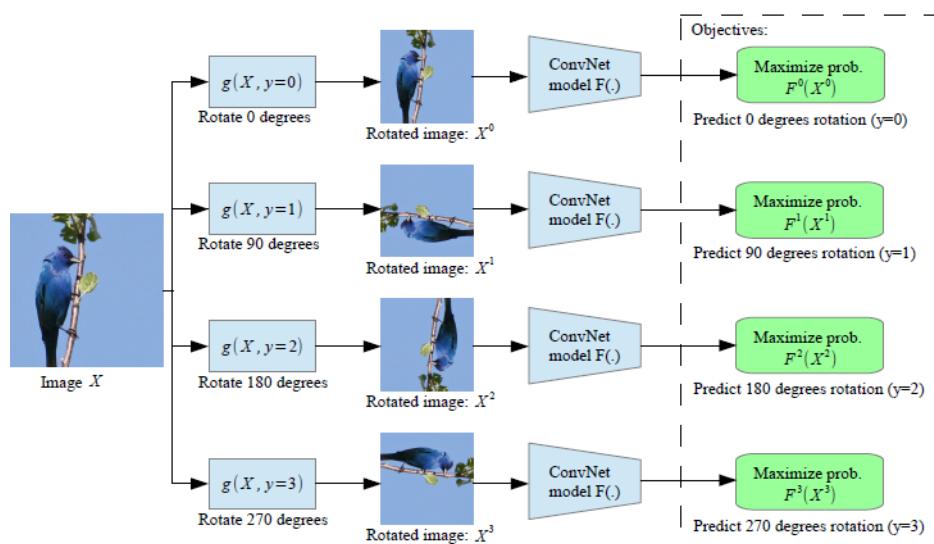
¹Self-distillation

²View

وظیفه‌ی پوششی پیش‌بینی چرخش:

یکی از وظایف پوششی مشهور در حوزه‌ی یادگیری خودناظارتی، پیش‌بینی میزان چرخش اعمال شده بر تصویر است. این وظیفه که نخستین بار در مقاله‌ی RotNet [۲۰] معرفی شد، بر این فرض استوار است که یک شبکه‌ی عصبی تنها در صورتی قادر به تشخیص زاویه‌ی چرخش یک تصویر خواهد بود که بتواند به درک عمیقی از ساختار درونی و مفاهیم معنایی موجود در تصویر دست یابد. از این رو، پیش‌بینی چرخش به عنوان یک وظیفه‌ی ساده و مشخص، در عمل منجر به یادگیری بازنمایی‌هایی می‌شود که برای بسیاری از وظایف پایین‌دستی نیز قابل انتقال هستند.

در پیاده‌سازی اولیه‌ی این ایده، تصویر ورودی به صورت تصادفی یکی از چهار چرخش صفر، ۹۰، ۱۸۰ یا ۲۷۰ درجه را دریافت می‌کند. مدل باید زاویه‌ی صحیح را از میان چهار گزینه تشخیص دهد. برای این منظور، ساختار مدل از چندین لایه‌ی پیچشی برای استخراج ویژگی استفاده می‌کند و در نهایت به یک لایه‌ی کاملاً متصل با چهار نورون خروجی منتهی می‌شود که هر نورون نمایانگر یکی از کلاس‌های زاویه‌ی چرخش است. این ساختار در شکل ۶-۲ نمایش داده شده است.



شکل ۶-۲: ساختار کلی شبکه‌ی پیش‌بینی چرخش

برتری اصلی این روش در آن است که بدون استفاده از هیچ‌گونه برچسب دستی، مدل را وادر می‌کند تا ساختار اشیاء، موقعیت اجزای تصویر و ویژگی‌های کلان معنایی را در بازنمایی‌های درونی خود بیاموزد. این بازنمایی‌ها در مراحل بعدی می‌توانند برای وظایفی نظیر طبقه‌بندی تصویر یا شناسایی اشیاء مورد استفاده قرار گیرند.

همان‌گونه که در جدول ۱-۲ دیده می‌شود، عملکرد مدل پیش‌آموزش‌دیده با استفاده از وظیفه‌ی

جدول ۱-۲: مقایسه‌ی عملکرد روش پیش‌بینی چرخش

	Classification (%mAP)	Detection (%mAP)	Segmentation (%mIoU)
Trained layers	fc6-8	all	all
ImageNet labels	78.9	79.9	56.8
Random		53.3	43.4
Random rescaled Krähenbühl et al. (2015)	39.2	56.6	45.6
Egomotion (Agrawal et al., 2015)	31.0	54.2	43.9
Context Encoders (Pathak et al., 2016b)	34.6	56.5	44.5
Tracking (Wang & Gupta, 2015)	55.6	63.1	47.4
Context (Doersch et al., 2015)	55.1	65.3	51.1
Colorization (Zhang et al., 2016a)	61.5	65.6	46.9
BIGAN (Donahue et al., 2016)	52.3	60.1	46.9
Jigsaw Puzzles (Noroozi & Favaro, 2016)	-	67.6	53.2
NAT (Bojanowski & Joulin, 2017)	56.7	65.3	49.4
Split-Brain (Zhang et al., 2016b)	63.0	67.1	46.7
ColorProxy (Larsson et al., 2017)		65.9	38.4
Counting (Noroozi et al., 2017)	-	67.7	51.4
RotNet	70.87	72.97	54.4
			39.1

پوششی پیش‌بینی چرخش و سپس تنظیم دقیق برای انجام وظیفه‌ی پایین‌دستی اصلی، نه تنها از سایر روش‌های خودناظارتی و بدون نظارت موجود در زمان خود پیشی گرفته، بلکه عملکردی نزدیک به مدل‌های نظارت شده نیز از خود نشان داده است. این امر نشان‌دهنده‌ی قدرت وظایف ساده‌ی زمینه‌محور در هدایت مدل به سمت درک معنایی از داده‌های ورودی است.

وظیفه‌ی پوششی حل پازل^۱:

یکی دیگر از روش‌های برجسته در یادگیری خودناظارتی زمینه‌محور، وظیفه‌ی حل پازل است که نخستین بار توسط نوروزی و فاوارو در مقاله‌ای تأثیرگذار ارائه شد [۴۱]. ایده‌ی اصلی این روش بر آن استوار است که مدل برای تشخیص نحوه‌ی قرارگیری صحیح اجزای تصویر، ناگزیر به درک دقیق ساختار داخلی تصویر، موقعیت اجزای اشیاء، و روابط مکانی بین آن‌ها خواهد بود. این درک ساختاری موجب می‌شود که مدل به بازنمایی‌هایی دست یابد که نه تنها ویژگی‌های محلی تصویر (مانند لبه‌ها و بافت‌ها) بلکه مفاهیم سطح بالای معنایی (مانند موقعیت اعضای یک شیء یا ارتباط بین اشیاء) را نیز منعکس کنند.

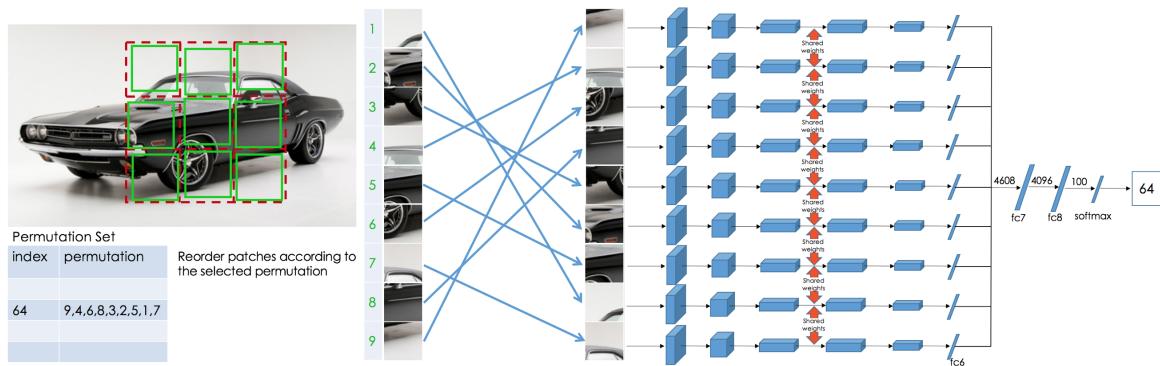
در فرایند این وظیفه‌ی پوششی، ابتدا تصویر اصلی به یک برش با ابعاد 225×225 تبدیل می‌شود. سپس این تصویر به ۹ قسمت مساوی 75×75 تقسیم شده و از هر کدام، یک برش تصادفی 64×64 استخراج می‌شود. هدف از این کار، حذف مرزهای دقیق بین قطعات و جلوگیری از وابستگی مدل به صرف تشخیص مرزها برای حل پازل است؛ به عبارت دیگر، مدل باید به جای تکیه بر نشانه‌های مصنوعی،

¹Jigsaw Puzzle

ویژگی‌های معنایی واقعی تصویر را فرا بگیرد.

پس از آن، قطعات تصویر به صورت یک بردار ۹ تایی مسطح‌سازی شده و یکی از ۱۰۰ جایگشت از پیش تعیین شده روی آن اعمال می‌شود. این ۱۰۰ جایگشت از میان! ۹۱ جایگشت ممکن به گونه‌ای انتخاب شده‌اند که دشوارترین حالات ممکن را پوشش دهند و بدین ترتیب شبکه را به یادگیری عمیق‌تر و ادار کنند. جایگشت اعمال شده به عنوان شبه‌برچسب در نظر گرفته می‌شود که تنها در موقعیت جایگشت صحیح مقدار یک و در سایر نقاط صفر است.

معماری مدل، مطابق شکل ۷-۲، از ۹ بار اجرای شبکه‌ی AlexNet (با وزن‌های مشترک) برای استخراج ویژگی از هر قطعه استفاده می‌کند. خروجی‌های حاصل از این شبکه‌ها سپس در کنار هم قرار گرفته و به یک یا چند لایه‌ی تماماً متصل داده می‌شوند تا جایگشت صحیح پیش‌بینی شود. تابع خروجی softmax بر بردار خروجی اعمال شده و هدف آموزش، کاهش خطای پیش‌بینی نسبت به شبه‌برچسب جایگشت است.



شکل ۷-۲: ساختار کلی شبکه‌ی حل پازل

مطالعه‌ی انجام شده در این مقاله نشان داد که مدل آموزش‌دیده با این وظیفه‌ی پوششی، قادر به یادگیری بازنمایی‌هایی با کیفیت بالا بوده که در وظایف پایین‌دستی همچون طبقه‌بندی تصویر و شناسایی اشیاء عملکرد قابل توجهی داشته‌اند. همچنین این روش راه را برای توسعه‌ی سایر وظایف زمینه‌محور در یادگیری خودناظارتی هموار ساخت و مبنای برای کارهای بعدی [۴۳، ۴۲] در این حوزه شد.

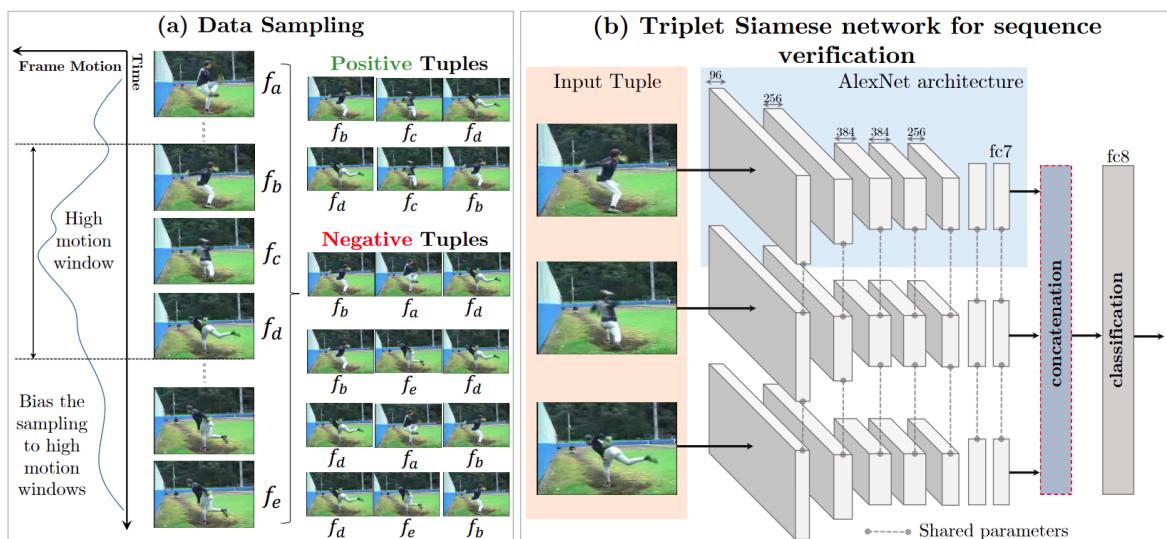
وظیفه‌ی پوششی پیش‌بینی ترتیب صحیح در دنباله:

یکی دیگر از وظایف پوششی مبتنی بر زمینه، وظیفه‌ی پیش‌بینی درستی یا نادرستی ترتیب زمانی فریم‌های یک ویدیو است. این وظیفه نخستین بار توسط میسرا و همکارانش در مقاله‌ای با عنوان *Shuffle and Learn* [۴۴] معرفی شد. ایده‌ی اصلی این روش آن است که فریم‌های استخراج شده از یک ویدیو را

فصل دوم: ادبیات موضوع و کارهای پیشین

به صورت یک دنباله‌ی تصویری به مدل می‌دهیم و از آن انتظار داریم که تشخیص دهد آیا ترتیب زمانی این فریم‌ها حفظ شده است یا به طور تصادفی به هم ریخته شده‌اند. در واقع، این روش یک مسئله‌ی طبقه‌بندی دودویی را تعریف می‌کند که خروجی آن مشخص می‌سازد آیا دنباله‌ی ورودی طبیعی و معنادار است یا نه.

برای افزایش دشواری این وظیفه و در نتیجه به دست آوردن بازنمایی‌های باکیفیت‌تر، در مرحله‌ی انتخاب فریم‌ها، از یک راهبرد هوشمندانه بهره گرفته می‌شود. همانطور که در شکل ۲-۲ نشان داده شده است، فریم‌هایی از ویدیو انتخاب می‌شوند که دارای بیشترین تفاوت بصری با یکدیگر هستند. این کار موجب می‌شود مدل نتواند صرفا بر پایه‌ی اطلاعات سطح پایین مانند رنگ یا بافت، تصمیم‌گیری کند، بلکه ناگزیر شود برای تشخیص ترتیب صحیح، به درک عمیق‌تری از محتوای ویدیویی و روابط زمانی میان فریم‌ها برسد.



شکل ۲-۲: ساختار کلی شبکه‌ی بررسی درستی ترتیب فریم‌های ویدیویی

ساختار کلی شبکه‌ی مورد استفاده نیز شباهت زیادی به روش حل پازل تصویری دارد که پیش‌تر معرفی شد (شکل ۲-۲). در اینجا نیز از یک شبکه‌ی عصبی برای استخراج ویژگی‌های هر فریم استفاده می‌شود و سپس ویژگی‌های استخراج شده در سطحی بالاتر با یکدیگر ترکیب می‌شوند تا تصمیم‌گیری نهایی صورت گیرد. اگرچه این دو روش در ظاهر شباهت‌های زیادی دارند، اما در عمل بر نوع داده‌های متفاوتی تکیه دارند (تصویر در برابر ویدیو) و همچنین اهداف طبقه‌بندی‌شان نیز تفاوت دارد (تشخیص ترتیب صحیح در برابر پیش‌بینی جایگشت دقیق).

علاوه بر داده‌های ویدیویی، این رویکرد در حوزه‌های غیرتصویری نیز مورد استفاده قرار گرفته است.

فصل دوم: ادبیات موضوع و کارهای پیشین

برای مثال، بنویل و همکاران [۴۵] در مطالعه‌ای بر روی سیگنال‌های مغزی ثبت شده با استفاده از الکتروانسفالوگرافی^۱ (EEG)، ایده‌ی تشخیص ترتیب زمانی را به داده‌های چندکانالهی EEG تعمیم دادند. این کار نشان می‌دهد که وظایف پوششی مبتنی بر ترتیب، نه تنها در داده‌های بصری، بلکه در داده‌های زمانی نیز قابل کاربرد و اثربخش هستند.

۲-۲-۲-۲ روش‌های بازسازی محور

حال به سراغ دسته‌ی دیگری از روش‌های وظایف پوششی تحت عنوان بازسازی محور می‌رویم. در این دسته، ایده‌ی اصلی آن است که مدل با مشاهده‌ی بخشی از داده، یا نسخه‌ای ناقص، فشرده یا مخدوش شده‌ی آن، تلاش کند نسخه‌ی کامل یا اصلی داده را بازسازی کند. این فرایند باعث می‌شود مدل ناگزیر به استخراج اطلاعات بنیادی و ساختاری از داده باشد. روش‌های بازسازی محور، به‌ویژه در یادگیری نمایش‌های عمیق و قابل انتقال، اهمیت زیادی دارند و بسیاری از آن‌ها با روش‌های مولد نیز همپوشانی دارند؛ چراکه در هر دو، بازتولید داده نقش کلیدی دارد.

خودکدگذارها:

یکی از نخستین و پایه‌ای ترین روش‌های بازسازی محور که پایه‌ی بسیاری از دیگر الگوریتم‌های یادگیری خودناظارتی به شمار می‌رود، روش خودرمزگذار است. در این چارچوب، یک شبکه‌ی عصبی دو بخشی طراحی می‌شود: بخش نخست، با عنوان کدگذار، داده‌ی ورودی را به یک نمایش فشرده یا بردار ویژگی در فضای نهان نگاشت می‌کند؛ و بخش دوم، یعنی رمزگشا، تلاش می‌کند تا داده‌ی اولیه را از روی همین نمایش بازسازی کند. هدف اصلی، یادگیری یک بازنمایی نهفته است که بتواند ساختارهای اصلی داده را در خود حفظ کرده و بازسازی دقیقی از ورودی ارائه دهد. بدین ترتیب، شبکه بدون نیاز به برچسب‌گذاری، تنها با مشاهده‌ی داده‌های خام و بازسازی آن‌ها، ویژگی‌هایی را می‌آموزد که برای فهم عمیق‌تر محتوا مفید هستند.

روش خودکدگذار را می‌توان یکی از مصادق‌های روشن یادگیری خودناظارتی دانست؛ زیرا وظیفه‌ی آموزشی آن (یعنی بازسازی ورودی) تنها با استفاده از داده‌های خام تعریف می‌شود، بی‌آنکه نیازی به برچسب‌های انسانی باشد. از آنجا که خود داده‌ی ورودی نقش برچسب را ایفا می‌کند، ساختار یادگیری

^۱ الکتروانسفالوگرافی یا نوار مغزی، روشی برای ثبت فعالیت الکتریکی مغز است. در این روش الکترودهایی بر روی پوست سر قرار داده می‌شوند تا امواج مغزی را ثبت کرده و به تشخیص اختلالاتی مانند صرع، مشکلات خواب و آسیب‌های مغزی کمک کنند

فصل دوم: ادبیات موضوع و کارهای پیشین

آن به طور طبیعی در قالب خودنظرارتی جای می‌گیرد. این چارچوب، نه تنها به عنوان یک روش مستقل برای استخراج ویژگی‌های قابل انتقال کاربرد دارد، بلکه پایه‌گذار بسیاری از دیگر رویکردهای پیشرفته‌تر مانند خودکدگذار متغیر و خودکدگذار حذف نویز^۱ بوده است.

یکی از گسترش‌های مهم خودکدگذار، خودزمگذار حذف نویز است که برای نخستین بار توسط ونسنت و همکاران [۴۶] معرفی شد. در این روش، ورودی اصلی به صورت مصنوعی دچار اختلال یا نویز می‌شود و مدل موظف است نسخه‌ی اصلی و بدون نویز را بازسازی کند. هدف از این فرایند، واداشتن مدل به یادگیری ویژگی‌هایی مقاوم و پایدار نسبت به اختلالات سطحی داده است. به عبارت دیگر، مدل نمی‌تواند صرفاً با حفظ جزئیات ظاهری داده موفق به بازسازی شود و ناچار است ساختارهای بنیادین تری از داده را درک کند. این ویژگی، خودکدگذار حذف نویز را به ابزاری مناسب برای یادگیری نمایش‌های قابل تعمیم در حوزه‌های مختلف مانند بینایی ماشین، پردازش سیگنال و یادگیری انتقالی تبدیل کرده است.

ترمیم تصویر:

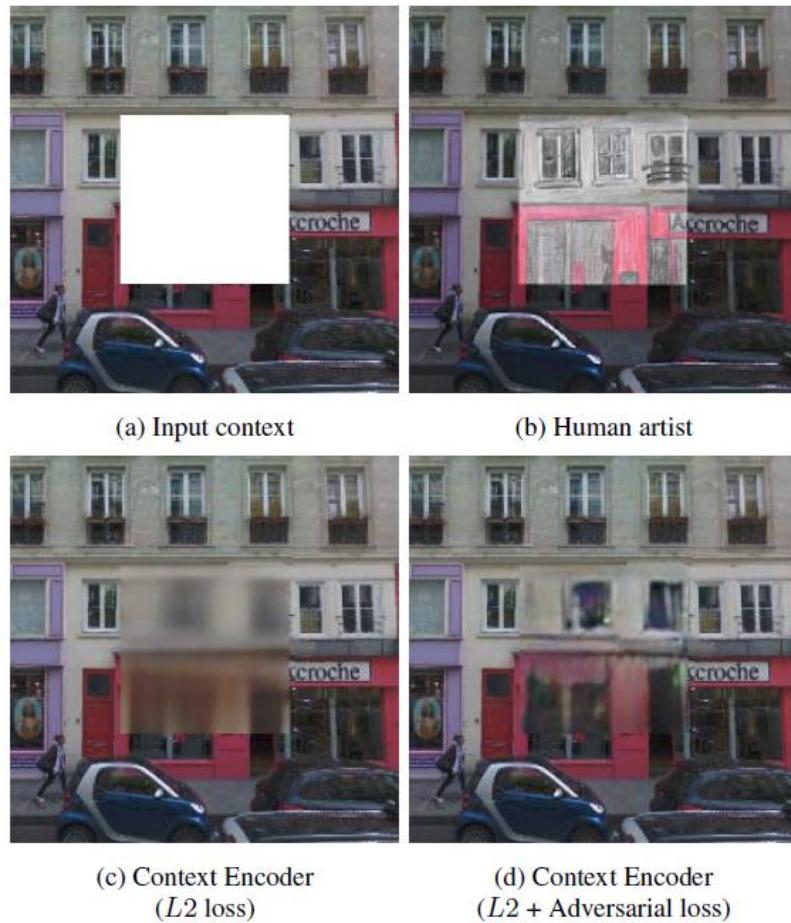
وظیفه پوششی ترمیم تصویر^۲ که توسط پاتاک و همکاران [۴۷] ارائه شد، از خودکدگذاری تحت عنوان کدگذار زمینه‌ای^۳ استفاده می‌کند تا بتواند یک شبکه‌ی پیچشی را طوری آموزش دهد که بخش‌های حذف شده و آسیب‌دیده از تصویر را بازسازی و ترمیم نماید. شبکه برای این که بتواند این وظیفه را به خوبی انجام دهد باید توانایی درک مفاهیم موجود در داده را داشته باشد. در شکل ۹-۲ یک نمونه از عملکرد این روش را با استفاده از دوتابع هزینه‌ی مختلف می‌توان دید.

معماری کلی کدگذار زمینه‌ای این‌گونه است که ابتدا یک یا چند ناحیه از تصاویر را به صورت تصادفی حذف می‌کنیم. این نواحی تصادفی می‌توانند انواع اشکال هندسی مانند دایره و مربع و یا حتی کاملاً تصادفی باشد. سپس این تصویر تخریب شده به ورودی شبکه‌ی رمگذار داده می‌شود که معماری کلی آن شباهت زیادی به شبکه‌ی AlexNet دارد. شبکه‌ی کدگذار یک ماتریس به فرم $C \times W \times H$ به ازای هر نمونه می‌دهد که در آن C بیانگر تعداد کانال‌ها و W و H بیانگر عرض و ارتفاع ماتریس خروجی هستند. در معماری شبکه‌ی کدگذار زمینه‌ای، برای افزایش ظرفیت یادگیری مدل، خروجی مدل مستقیماً به رمزگشا داده نمی‌شود. بلکه از یک لایه‌ی تمام متصل استفاده می‌شود. برای استفاده از لایه‌ی تماماً

¹Denoising Auto-Encoder

²Image Inpainting

³Context Encoder

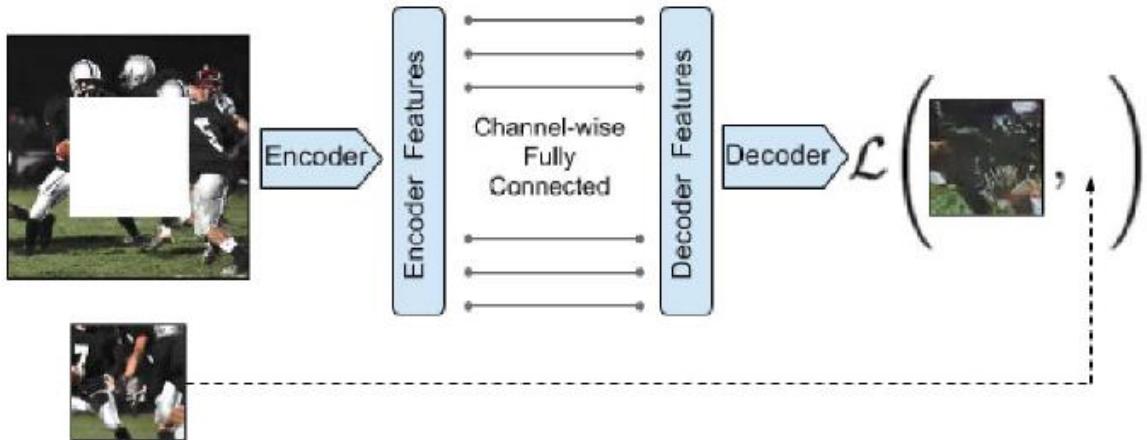


شکل ۹-۲: عملکرد کدگذار زمینه‌ای برای ترمیم تصاویر

متصل معمولاً از روش‌های ادغام^۱ مانند ادغام حداقل استفاده می‌شود. این‌گونه ابعاد از $C \times W \times H$ به C تغییر می‌کند. اما ایراد ادغام این است که می‌تواند اطلاعات بالرزش را از بین ببرد. علاوه بر آن اگر بخواهیم که بدون استفاده از ادغام از لایه‌ی تمام‌متصل استفاده کنیم، باید خروجی را به فرم برداری به طول $C \times W \times H$ درآوریم که حدود ۱۰۰ میلیون پارامتر به شبکه اضافه می‌کند. این کار علاوه بر پردازش پیچیده، مشکل بیش‌برازش را نیز به دنبال خواهد داشت. به همین منظور محققان در این مقاله از یک لایه‌ی تمام‌متصل درون‌کanalی^۲ استفاده کردند. بدین صورت که هر یک از C کanal را به یک بردار به طول $W \times H$ تبدیل می‌کنیم. سپس یک لایه خواهیم داشت که در دل خود C لایه‌ی تمام‌متصل مانند شکل ۱۰-۲ دارد. نتیجتاً خروجی کدگذار به ورودی رمزگشا همراه با یک لایه‌ی غیر خطی متصل خواهد شد، اطلاعاتی از بین نخواهد رفت و تعداد پارامترهای قابل یادگیری مقدار اندکی افزایش خواهد یافت که برای ما قابل تحمل هستند.

¹Pooling Methods

²Channel-wise Fully Connected



شکل ۲: معماری کدگذار زمینه‌ای

در رمزگشا با استفاده از لایه‌های پیچشی معکوس^۱، ویژگی‌های فشرده شده را به ابعاد تصویر ورودی می‌رسانیم. مثلاً ویژگی‌های ورودی به رمزگشا اگر دارای ۶۴ کانال با طول و عرض ۱۶ باشند، آن را به ۳ کانال با طول و عرض ۲۵۶ می‌رسانیم. سپس سیگنال نظارت که همان تصویر سالم است اعمال می‌شود و با استفاده از یک تابع هزینه، آموزش شبکه انجام می‌شود. تابع هزینه‌ی ابتدایی، میانگین مربعات خطای^۲ می‌باشد که در فرمول ۳-۲ قابل مشاهده است.

$$L_{rec}(x) = \|\hat{M} \odot (x - F((1 - \hat{M}) \odot x))\|_2 \quad (3-2)$$

اما همانطور که در شکل ۹-۲ بخش c قابل مشاهده است، این تابع هزینه یک خروجی محو شده به ما می‌دهد. محققان برای بهبود خروجی تولید شده از تابع هزینه‌ی تخاصمی استفاده کردند. در واقع از یک شبکه‌ی اضافه تحت عنوان تفکیک‌کننده^۳ بر پایه‌ی شبکه‌ی مولد تخاصمی استفاده کردند. در اینجا خروجی رمزگشا حکم مولد را خواهد داشت. بنابراین فرمول بندی آن به فرم فرمول ۴-۲ در می‌آید. البته در نهایت از ترکیب هزینه‌ی mse و تخاصمی استفاده می‌شود که به فرم فرمول ۵-۲ نهایی می‌شود. همانطور که در شکل ۹-۲ بخش d نیز دیده می‌شود، خروجی دقیق‌تر و بهتری به دست می‌آید.

$$L_{adv} = \max_D \mathbb{E}_{x \in X} [\log(D(x)) + \log(1 - D(F((1 - \hat{M}) \odot x)))] \quad (4-2)$$

^۱منظور از «لایه‌ی پیچشی معکوس»، همان عملیات deconvolution یا transposed convolution است که برای بزرگنمایی ویژگی‌ها در شبکه‌های مولد یا رمزگشا به کار می‌رود.

^۲Mean Squared Error

^۳Discriminator

$$L = \lambda_{rec} L_{rec} + \lambda_{adv} L_{adv} \quad (5-2)$$

در مقاله، مقدار λ_{adv} برابر با 0.001 و مقدار λ_{rec} برابر با 0.999 قرار داده شده است. بنابراین هدف اصلی باز تولید دقیق تصویر می‌باشد اما مقدار کمی هزینه‌ی تخصصی نیز برای وضوح تصاویر تولید شده استفاده شده است.

روش ترمیم تصویر با استفاده از کدگذار زمینه‌ای علاوه بر اینکه وظیفه‌ی اصلی خود یعنی بازسازی نواحی حذف شده را انجام می‌دهد، به عنوان یک سیستم پیش‌آموزش قدرتمند برای استخراج ویژگی‌های بصری نیز عمل می‌کند. در حقیقت پس از آموزش کدگذار روی تصاویر بدون برچسب، می‌توان آن را بر روی یک مجموعه داده‌ی دارای برچسب تنظیم دقیق کرد و وظایفی مانند دسته‌بندی را انجام داد.

۳-۲-۲-۲ یادگیری تباینی

با وجود پیشرفت‌هایی که روش‌های پیشین یادگیری خودناظارتی در زمان خود به همراه داشتند، همچنان عملکرد آن‌ها فاصله‌ی محسوسی با روش‌های نظارت شده کامل داشت. این شکاف عملکردی تا حد زیادی با ظهور یادگیری تباینی در چارچوب یادگیری خودناظارتی کاهش یافت. بهره‌گیری از ایده‌های تباینی منجر به رشد قابل توجهی در کیفیت بازنمایی‌های استخراج شده از داده‌ها شد، به طوری که عملکرد مدل‌های بدون نظارت در برخی وظایف به سطوح قابل مقایسه‌ای با روش‌های نظارت شده رسید. در سال‌های اخیر، تمرکز بسیاری از مقالات بر توسعه و بهبود روش‌های خودناظارتی تباینی بر روی انواع داده‌های مختلف از جمله تصویر و سیگنال بوده است. در ادامه، ابتدا به شرح مفهوم یادگیری تباینی پرداخته و سپس برخی از روش‌های شاخص مبتنی بر این رویکرد را مرور می‌کنیم.

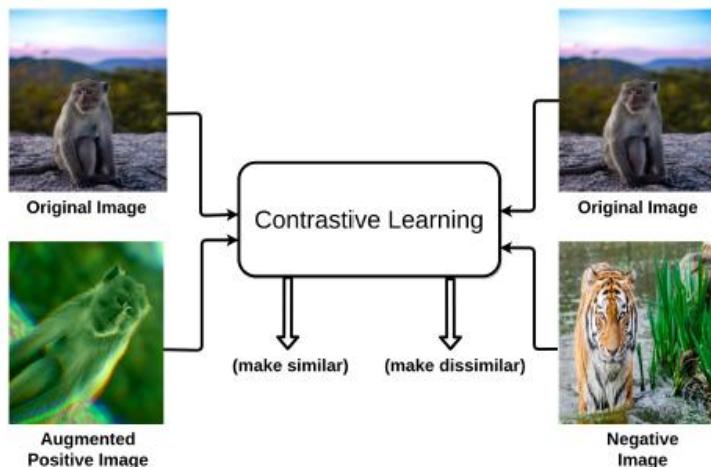
تعریف یادگیری تباینی

یادگیری تباینی در بستر یادگیری خودناظارتی، رویکردی مبتنی بر تمایز است که تلاش می‌کند بازنمایی‌های مشابه برای نمونه‌های با مفهوم یکسان و بازنمایی‌هایی متمایز برای نمونه‌های ناهم‌معنا ایجاد کند. این امر معمولاً با بهره‌گیری از تکنیک‌های داده‌افزایی محقق می‌شود؛ به این ترتیب که از یک داده‌ی واحد چندین نسخه‌ی تغییریافته تولید شده و به عنوان نمونه‌های «متبت» در نظر گرفته می‌شوند، در حالی که داده‌های متفاوت، نمونه‌های «منفی» تلقی می‌گردند. هدف از آموزش مدل، نزدیک کردن بازنمایی نمونه‌های مثبت به یکدیگر و دور ساختن آن‌ها از نمونه‌های منفی است.

برای مثال، فرض کنید از مجموعه داده، دو نمونه‌ی A و B انتخاب می‌شوند و سپس با اعمال

فصل دوم: ادبیات موضوع و کارهای پیشین

داده‌افزایی، نمونه‌های A_1 و B_1 از آن‌ها ساخته می‌شوند. مدل باید یاد بگیرد که A و A_1 را به عنوان نمونه‌های مشابه و A و B_1 را به عنوان نمونه‌های متفاوت در نظر بگیرد. این فرایند با تعریف یکتابع هزینه بر پایه‌ی بردارهای بازنمایی حاصل از یک کدگذار پیاده‌سازی می‌شود (شکل ۱۱-۲) [۴۸]. در نهایت، کدگذار آموزش‌دیده قادر خواهد بود بازنمایی‌های غنی و قابل انتقال برای استفاده در وظایف یادگیری پایین‌دستی فراهم آورد.



شکل ۱۱-۲: نمونه‌ای از فرایند یادگیری تباینی

در ادامه به بررسی تعدادی از روش‌ها و مقالات منتشر شده در این حوزه می‌پردازیم.

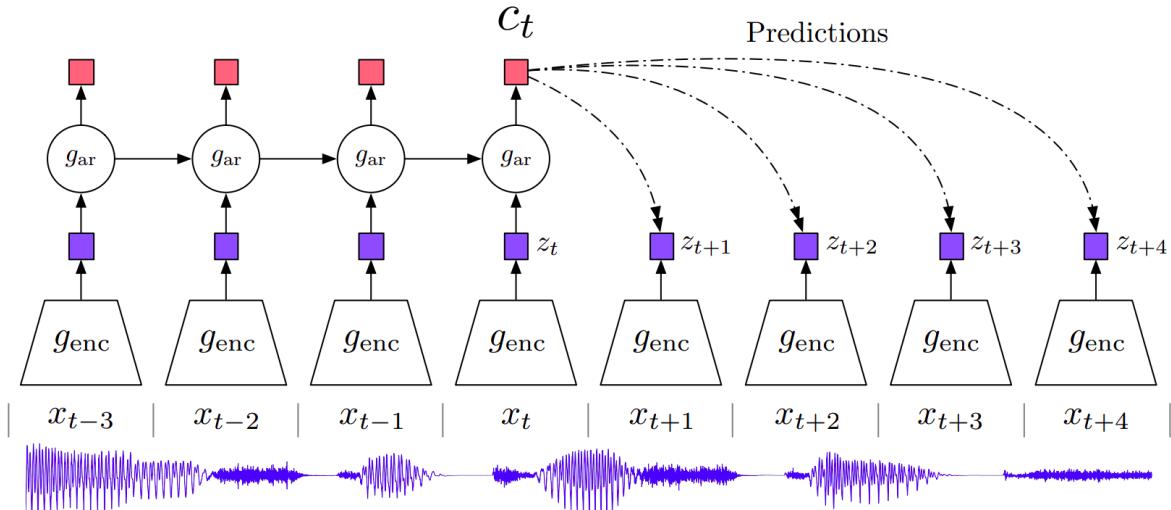
روش کدگذاری پیش‌بینی‌کننده‌ی تباینی (CPC^۱)

روش کدگذاری پیش‌بینی‌کننده‌ی تباینی یا به اختصار CPC که نخستین بار در سال ۲۰۱۸ توسط Oord و همکاران [۴۹] معرفی شد، یکی از اولین تلاش‌های موفق برای استفاده از یادگیری تباینی در استخراج بازنمایی‌های غنی و قابل انتقال از داده‌های بدون برچسب بود. این روش، به ویژه در داده‌های ترتیبی مانند صوت و سیگنال، عملکرد قابل توجهی از خود نشان داده و در حوزه‌هایی مانند تشخیص گفتار کاربرد یافته است.

ایده‌ی اصلی این روش بر پایه‌ی پیش‌بینی اطلاعات آینده از روی گذشته است. اما بر خلاف بسیاری از دیگر روش‌های خودهمبسته^۱ که آینده را با تولید داده‌ی اصلی پیش‌بینی می‌کنند، یک مدل مولد نیست. بلکه یک مدل تباینی است که هدف آن پیش‌بینی بازنمایی مربوط به آینده با استفاده از تابع هزینه تباینی است.

¹Contrastive Predictive Coding

¹Autoregressive



شکل ۱۲-۲: ساختار کلی روش CPC

با این فرض که مجموعه داده ورودی سری زمانی می‌باشد، فرض کنید در x_t هستیم که هر x_t بیانگر یک پنجره از سری زمانی می‌باشد که می‌تواند دارای همپوشانی با پنجره‌های مجاور باشد. با استفاده از یک شبکه‌ی پیچشی (g_{enc})، بردارهای بازنمایی برای پنجره‌های از x_{t-n} تا x_t می‌سازیم. سپس با استفاده از یک شبکه‌ی بازگشتی مانند واحد بازگشتی دروازه‌ای (g_{ar})، از روی $g_{enc}(x_t)$ تا $g_{enc}(x_{t-n})$ یک بردار زمینه تحت عنوان c_t تولید می‌کنیم.

حال به نوآوری این روش می‌رسیم. در روش CPC بهجای این که از بردار c_t برای تخمین مستقیم توزیع احتمالاتی k قدم آینده استفاده شود (یعنی $p_k(x_{t+k}|c_t)$ ، از یک نسبت چگالی^۲ استفاده می‌شود که هدف آن حفظ اطلاعات متقابل بین بازنمایی آینده z_{t+k} و بردار زمینه c_t است. این نسبت چگالی به صورت معادله ۶-۲ مدل‌سازی می‌شود که در آنتابع امتیازدهی f_k میزان شباهت بازنمایی آینده و زمینه را نشان می‌دهد. برای پیاده‌سازی، از یک مدل ساده به فرم معادله ۷-۲ استفاده شده است که در آن W_k یک ماتریس تبدیل خطی قابل آموزش برای گام زمانی k است. این مدل با محاسبه‌ی شباهت بین بردار z_{t+k} (که یک نماینده‌ی بازنمایی برای آینده است) و بردار زمینه‌ی c_t ، تلاش می‌کند که امتیاز را برای z_{t+k} های مثبت بیشینه و برای z_{t+k} های منفی کمینه کند

$$f_k(z_{t+k}, c_t) \propto \frac{p(z_{t+k}|c_t)}{p(z_{t+k})} \quad (6-2)$$

²Density Ratio

$$f_k(z_{t+k}, c_t) = \exp(z_{t+k}^T W_k c_t) \quad (7-2)$$

برای آموزش مدل، ازتابع هزینه‌ی ^۱ InfoNCE به فرم معادله‌ی ^۲ ۸-۲ استفاده می‌شود که در آن X شامل N نمونه‌ی تصادفی است که یکی از آن‌ها نمونه‌ی مثبت و $1 - N$ تای دیگر نمونه‌های منفی هستند. هدف مدل این است که امتیاز شباht f_k برای نمونه‌ی واقعی نسبت به نمونه‌های منفی بیشتر باشد. بدین ترتیب، مدل می‌آموزد که بازنمایی c_t شامل اطلاعات مفیدی برای پیش‌بینی آینده‌ی داده باشد.

$$\mathcal{L}_N = -\mathbb{E}_X \left[\log \frac{f_k(x_{t+k}, c_t)}{\sum_{x_j \in X} f_k(x_j, c_t)} \right] \quad (8-2)$$

در نهایت، پس از پایان مرحله‌ی پیش‌آموزش، می‌توان از کدگذار آموزش‌دیده به عنوان یک استخراج‌کننده‌ی ویژگی استفاده کرد و بازنمایی‌های به دست آمده را در وظایف پایین‌دستی مانند دسته‌بندی به کار برد.

استفاده از بانک حافظه ^۲ و تکانه ^۳:

یکی از چالش‌های اصلی در پیاده‌سازی موثر یادگیری تباینی، نیاز به مجموعه‌ای بزرگ و متنوع از نمونه‌های منفی است. برای اینکه مدل بتواند تمایز درستی بین نمونه‌های مثبت و منفی قائل شود، باید در هر تکرار آموزشی تعداد زیادی نمونه‌ی منفی داشته باشد. این در حالی است که در یک دسته از ورودی، تنها تعداد محدودی از نمونه‌ی منفی در دسترس است. به همین منظور، محققان روشنی تحت عنوان بانک حافظه ^۴ را ارائه دادند.

در این روش، به جای آن که فقط از داده‌های موجود در دسته‌ی ورودی فعلی استفاده شود، یک بردار بازنمایی برای هر نمونه در کل مجموعه داده محاسبه و در یک بانک حافظه ذخیره می‌شود. در طول آموزش، این بانک به طور تدریجی به روزرسانی شده و از آن برای استخراج نمونه‌های منفی استفاده می‌شود. فرض اصلی این روش (مانند بیشتر روش‌های دیگر تباینی) این است که هر نمونه در مجموعه داده به عنوان یک کلاس منحصر به فرد در نظر گرفته شود و بنابراین یادگیری بازنمایی به صورت تباینی

¹ Information Noise-Contrastive Estimation

² Memory Bank

³ Momentum

بر اساس تفاوت بین نمونه‌ها صورت می‌گیرد.

$$\mathcal{L}_i = -\log \frac{\exp(z_i^\top v_i / \tau)}{\sum_{j=1}^K \exp(z_i^\top v_j / \tau)} \quad (9-2)$$

تابع هزینه به کاررفته در این روش به فرم معادله‌ی ۹-۲ می‌باشد که همان تابع هزینه‌ی InfoNCE می‌باشد. در این فرمول:

- z_i بردار بازنمایی نمونه‌ی ورودی است که توسط شبکه کدگذار تولید شده است.
- v_i بازنمایی ذخیره شده‌ی همان نمونه در بانک حافظه است (مثبت).
- v_j دیگر نمونه‌های استخراج شده از بانک حافظه‌اند.
- τ پارامتر دما است که بسته به مقدار آن باعث هموارسازی یا تیز کردن نمودار خروجی می‌شود.

با وجود عملکرد قابل قبول، یکی از مشکلات این روش آن است که بانک حافظه به صورت مستقیم از پارامترهای کدگذار فعلی به روزرسانی نمی‌شود و در طول آموزش بین بازنمایی‌های فعلی و آنچه در حافظه ذخیره شده، ناسازگاری‌هایی به وجود خواهد آمد. این موضوع می‌تواند مانع از یادگیری پایدار شود. در راستای رفع مشکل ناهماهنگی بانک حافظه، روش تباین تکانه یا به اختصار^۱ Moco^۲ توسط He و همکاران [۵۱] معرفی شد. ایده‌ی اصلی این روش استفاده از دو شبکه کدگذار است؛ یک کدگذار جستار^۳ و یک کدگذار کلید^۴. کدگذار جستار مستقیماً از طریق پساننتشار به روزرسانی می‌شود؛ در حالی که کدگذار کلید با استفاده از به روزرسانی تکانه‌ای روی کدگذار جستار به روزرسانی می‌شود. فرم به روزرسانی تکانه‌ای برای پارامترهای کدگذار کلید به شکل زیر تعریف می‌شود:

$$\theta_k \leftarrow m \cdot \theta_k + (1 - m) \cdot \theta_q \quad (10-2)$$

که در آن:

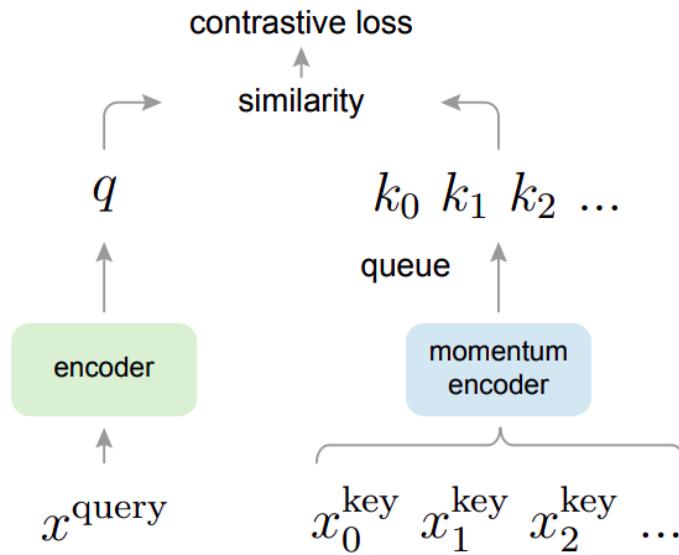
- پارامترهای کدگذار جستار است.
- پارامترهای کدگذار کلید است.

¹Momentum Contrast

²Query Encoder

³Key Encoder

- ضریب تکانه است که با قرار دادن مقادیر نزدیک به ۱ (در مقاله برابر با ۰.۹۹۹ می‌باشد) باعث می‌شود تغییرات پارامترهای کلید آهسته‌تر و پیوسته‌تر باشد.



شکل ۱۳-۲: ساختار کلی روش MoCo

از طرفی، MoCo به جای استفاده از یک بانک حافظه‌ی کامل که تمام داده‌ها را نگهداری می‌کند، از یک صف حافظه^۱ استفاده می‌کند. در این صف، بازنمایی‌های کلید تولید شده از نمونه‌های قبلی ذخیره می‌شوند. به مرور زمان، نمونه‌های قدیمی از صف خارج شده و نمونه‌های جدید جای آن‌ها را می‌گیرند. این ساختار حافظه‌ی پویا، ضمن صرفه‌جویی در حافظه، امکان دسترسی به هزاران نمونه‌ی منفی را در هر تکرار آموزشی فراهم می‌کند. تابع هزینه مورد استفاده در MoCo نیز نوعی از InfoNCE است:

$$\mathcal{L}_q = -\log \frac{\exp(q \cdot k^+ / \tau)}{\sum_{i=0}^K \exp(q \cdot k_i / \tau)} \quad (11-2)$$

که در آن:

- q بازنمایی حاصل از کدگذار جستار است.
- k^+ نمونه‌های مثبت تولید شده توسط کدگذار کلید است.
- k_i نمونه‌های منفی ذخیره شده در صف حافظه هستند.
- τ پارامتر دما است.

¹Queue

فصل دوم: ادبیات موضوع و کارهای پیشین

روش MoCo بهدلیل استفاده از کدگذار کلید با بهروزرسانی تکانهای، انسجام میان بازنمایی‌های فعلی و نمونه‌های منفی ذخیره شده را حفظ می‌کند و در عین حال بدون نیاز به پردازش مجدد تمام داده‌ها، بازنمایی‌های بهروز و موثری تولید می‌نماید. این مدل باعث پایداری بیشتر آموزش و بهبود دقت مدل‌های پیش‌آموزش یافته شد.

یادگیری تباینی ساده اما قدرتمند: روش SimCLR

روش یادگیری تباینی ساده که به اختصار SimCLR^۱ نامیده می‌شود، یکی از اثرگذارترین و پایه‌ای ترین روش‌های یادگیری تباینی خودنظرتی است که توسط چن و همکاران [۲۲] معرفی شد. برخلاف روش‌هایی مانند MoCo که به ساختارهای پیچیده‌ای نظری صفت حافظه و کدگذار تکانهای نیاز دارند، SimCLR با ساختاری بسیار ساده، توانست بازنمایی‌های قدرتمندی را برای تصاویر یاد بگیرد و عملکرد قابل مقایسه با روش‌های نظارت شده به دست آورد. سادگی معماری در کنار داده‌افزایی قوی و دسته‌های بزرگ داده آموزشی کلید عملکرد خوب این روش است.

فرایند آموزش در SimCLR از چهار جزء اصلی تشکیل شده است:

۱. داده‌افزایی قوی

۲. شبکه کدگذار

۳. شبکه نگاشت^۲

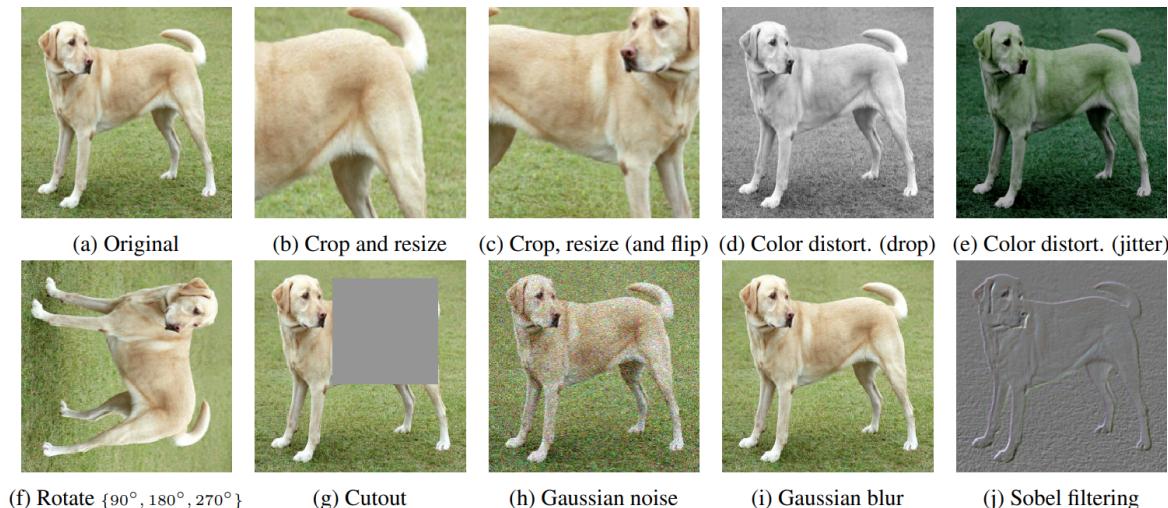
۴. تابع هزینه InfoNCE

در ادامه هر یک از این اجزا را بررسی می‌کنیم.

داده‌افزایی قوی: دو تابع داده‌افزایی به صورت تصادفی بر روی هر ورودی x_k اعمال می‌شود و در نتیجه‌ی آن دو نمایش x_{2k-1} و x_{2k} تولید می‌گردند. ترکیب‌های متعددی از تابع‌های تبدیل مختلف موجود در شکل ۱۴-۲ به صورت تصادفی می‌توانند استفاده شوند تا خروجی‌های مختلف و تصادفی برای هر داده ایجاد شوند. برای مثال یکی از توابع مورد استفاده می‌تواند به این صورت باشد که با یک احتمال برش انجام دهد، با یک احتمال تصویر را قرینه و بر عکس کند، نویز تصادفی گوسی با میانگین μ و انحراف معیار σ اعمال کند و رنگ تصویر را به صورت تصادفی تغییر دهد. نتیجتاً با دو بار اعمال این تابع تصادفی

¹Simple Framework for Contrastive Learning of Visual Representations

²Projection Head



شکل ۱۲-۲: روش‌های ایجاد داده‌ی افزوده

بر روی هر ورودی، دو داده‌ی افزوده خواهیم داشت که باید شباهتشان را با یکدیگر بیشینه کنیم. هر چه شدت اعمال روش‌های افزایش داده بیشتر باشد، تفکیک بین داده‌های مثبت و منفی را برای مدل سخت‌تر می‌کند؛ اما همین امر می‌تواند باعث شود که مدل وادار به یادگیری بازنمایی‌های مفیدتر و کاربردی‌تر شود. همچنین هر چه تعداد نمونه‌های منفی درون یک دسته‌ی آموزشی بیشتر باشد، مدل نمونه‌های بیشتری را می‌بیند و بهمین ترتیب تباین بهتری می‌تواند انجام دهد و یادگیری قوی‌تر و پایدارتر می‌شود.

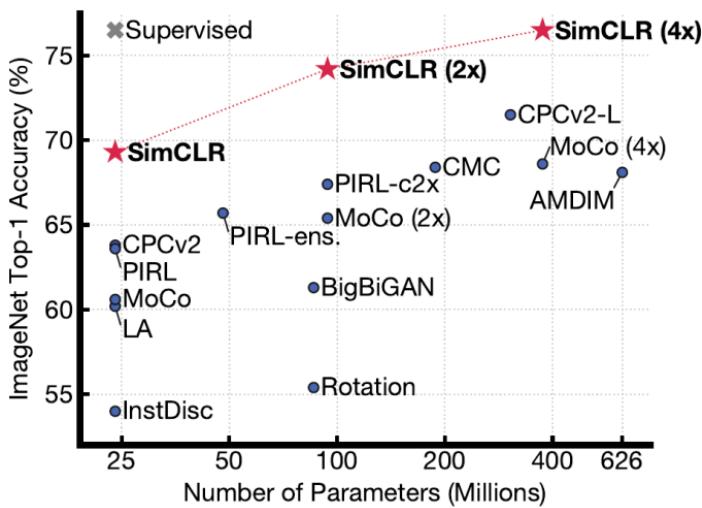
شبکه کدگذار: برای هر نما x_ℓ ، با استفاده از کدگذار f_θ مبتنی بر شبکه‌ی پیچشی، یک بردار بازنمایی h_ℓ به فرم معادله ۱۲-۲ می‌سازیم.

$$h_\ell = f_\theta(x_\ell) \in \mathbb{R}^d \quad (12-2)$$

در مقاله، از یک شبکه ResNet بدون لایه‌ی دسته‌بندی کننده استفاده شده است و h_ℓ بعد از تجمیع سراسری به دست می‌آید. در ارزیابی پایین‌دستی از همین h به عنوان بازنمایی نهایی برای هر نمونه استفاده می‌شود.

شبکه نگاشت: یکی از نوآوری‌های روش SimCLR، استفاده از یک شبکه تماماً متصل کم‌عمق (یک لایه پنهان) تحت عنوان شبکه نگاشت می‌باشد. با استفاده از این شبکه، خروجی h_ℓ تبدیل به z_ℓ می‌شود.

$$z_\ell = g_\phi(h_\ell) = \mathbf{W}^{(2)}\sigma(\mathbf{W}^{(1)}h_\ell) \quad (13-2)$$



شکل ۲-۱۵: دقت روش SimCLR

نویسندهان مقاله نشان دادند که استفاده از یک شبکه نگاشت و یک تابع فعال‌ساز غیر خطی و سپس اعمال هزینه‌ی تباینی بر روی z عملکرد بهتری را نسبت به استفاده از h برای محاسبات هزینه‌ی تباینی می‌دهد. ایده‌ی شهودی برای این کار این است که g_ϕ جذب‌کننده‌ی هزینه‌ی تباینی باشد تا بازنمایی‌های عمومی‌تری را بیاموزد. در معادله ۱۳-۲، $W^{(1)}$ و $W^{(2)}$ پارامترهای شبکه‌ی نگاشت و σ تابع فعال‌ساز غیر خطی می‌باشد که معمولاً از تابع ReLU^۱ استفاده می‌شود.

تابع هزینه تباینی: فرض کنید یک دسته آموزشی شامل N نمونه ورودی باشد. پس با دو نما از هر تصویر، $2N$ نمونه آموزشی خواهیم داشت. سپس برای دو نمونه‌ی مثبت i و j معادله ۱۴-۲ را خواهیم داشت.

$$\ell_{i,j} = -\log \frac{\exp(sim(z_i, z_j)/\tau)}{\sum_{k=1}^{2N} 1_{[k \neq i]} \exp(sim(z_i, z_k)/\tau)} \quad (14-2)$$

$$sim(u, v) = u^\top v / \|u\| \|v\| \quad (15-2)$$

در این معادله، $\{0, 1\} \in 1_{[k \neq i]}$ بیانگر تابع نشانگر^۲ می‌باشد که برای تمامی k ‌ها نابرابر با i برابر با یک می‌باشد. پارامتر τ تمرکز توزیع را کنترل می‌کند. هر چه τ کوچک‌تر، شیب‌ها تیزتر تابع sim نیز بیانگر

¹Rectified Linear Unit

²Indicator function

فصل دوم: ادبیات موضوع و کارهای پیشین

جدول ۲-۲: دقیقیت روش SimCLR در یادگیری انتقالی

	Food	CIFAR10	CIFAR100	Birdsnap	SUN397	Cars	Aircraft	VOC2007	DTD	Pets	Caltech-101	Flowers
<i>Linear evaluation:</i>												
SimCLR	76.9	95.3	80.2	48.4	65.9	60.0	61.2	84.2	78.9	89.2	93.9	95.0
Supervised	75.2	95.7	81.2	56.4	64.9	68.8	63.8	83.8	78.7	92.3	94.1	94.2
<i>Fine-tuned:</i>												
SimCLR	89.4	98.6	89.0	78.2	68.1	92.1	87.0	86.6	77.8	92.1	94.1	97.6
Supervised	88.7	98.3	88.7	77.8	67.0	91.4	88.0	86.5	78.8	93.2	94.2	98.0
Random init	88.3	96.0	81.9	77.0	53.7	91.3	84.8	69.4	64.1	82.7	72.5	92.5

یک معیار سنجش شباهت بین نمونه‌ها می‌باشد. در مقاله اصلی از تابع شباهت کسینوسی^۳ استفاده شده که فرمول آن به فرم معادله ۱۵-۲ می‌باشد. شباهت کسینوسی بیانگر کسینوس زاویه‌ی بین دو بردار می‌باشد. هر چه دو بردار در یک جهت باشند، کسینوس زاویه‌ی بین آن‌ها بیشینه و به یک نزدیک می‌شود و هر چه در خلاف جهت یکدیگر باشند، کسینوس زاویه‌ی بین آن‌ها کمینه و به منفی یک نزدیک می‌شود. بنابراین این تابع هزینه f_θ و به تبع آن g_ϕ را وادر می‌کند که نگاشت مربوط به نمونه‌های مثبت در یک جهت قرار گیرند و تا جای ممکن در جهت مخالف نسبت به دیگر نمونه‌های دسته باشند.

تابع هزینه استفاده شده در معادله ۱۴-۲، آنتروپی متقاطع نرمال شده با مقیاس دمایی (NT-Xent)^۱ است که فرمی از تابع InfoNCE می‌باشد. می‌توان نشان داد که با افزایش تعداد منفی‌ها و کمینه کردن این هزینه، کران پایین روی اطلاعات متقابل^۲ بین نماها افزایش می‌یابد. یعنی این که با کمینه کردن این تابع هزینه هنگامی که از تعداد نمونه‌های زیادی در یک دسته استفاده کردہایم، مدل یاد می‌گیرد که اطلاعات متقابل بین نمونه‌های مثبت را افزایش دهد و در واقع بازنمایی‌های پایدارتر و کاربردی‌تری را از روی داده‌ها بیاموزد. در نهایت پس از پایان پیش‌آموزش مدل، شبکه‌ی نگاشت به‌کل کنار گذاشته می‌شود چرا که برای وظیفه‌ی یادگیری تباینی و هزینه‌ی NT-Xent آموزش دیده بود. سپس یک شبکه‌ی تماماً متصل دیگر به عنوان یک دسته‌بند به جای شبکه‌ی نگاشت قرار داده می‌شود و یادگیری را بر روی مجموعه داده برچسب‌دار انجام می‌دهیم.

همانطور که در شکل ۱۵-۲ می‌توان دید، با افزایش پارامترها دقیقیت روش SimCLR بهبود بسیاری یافته؛ تا جایی که به دقیقیت روش نظارت‌شده رسیده و قابل مقایسه با آن شده است. اما نکته‌ی بسیار مهم روش SimCLR قابلیت تعمیم‌پذیری آن است. همانطور که در جدول ۲-۲ می‌توان دید، عملکرد روش SimCLR در یادگیری انتقالی با روش‌های نظارت‌شده برابر می‌کند و در بسیاری از موارد از آن‌ها

³Cosine Similarity

¹Normalized Temperature-scaled Cross-Entropy loss

²Mutual Information

پیشی گرفته است.

۴-۲-۲ پردازش زبان طبیعی

پردازش زبان طبیعی^۱ یکی از مهمترین شاخه‌های هوش مصنوعی است که هدف آن تعامل مؤثر میان انسان و ماشین از طریق زبان طبیعی می‌باشد. در سال‌های اخیر، یادگیری خودناظارتی پیشرفتهای شگرفی را در این حوزه رقم زده است. این رویکرد با بهره‌گیری از حجم عظیمی از داده‌های بدون برچسب و تعریف وظایف پیش‌بینی کمکی (مانند پیش‌بینی واژه‌های حذف شده یا پیش‌بینی جمله‌ی بعدی) و همچنین با بهره‌گیری از مدل مبدل^۲ [۵۲]، امکان یادگیری نمایش‌های زبانی قدرتمند و غنی را فراهم ساخته است. مزیت اصلی این روش در مقایسه با یادگیری نظارت شده، عدم نیاز به برچسب‌گذاری دستی داده‌ها و قابلیت تعمیم بهتر مدل به وظایف گوناگون زبان طبیعی است. ظهور مدل‌هایی همچون BERT^۳ و GPT^۴، که بر مبنای یادگیری خودناظارتی آموزش دیده‌اند، باعث ایجاد جهشی چشمگیر در کیفیت حل مسائل متنوع پردازش زبان طبیعی مانند ترجمه ماشینی، درک مطلب، و تولید متن شده است. علاوه بر این، استفاده از تمام داده‌های متنی موجود در آموزش، این امکان را فراهم می‌کند که اندازه و ظرفیت مدل (تعداد پارامترها) را به‌طور قابل توجهی افزایش دهیم، بی‌آن‌که به‌سادگی دچار بیش‌برازش شویم. این رویکرد منجر به پیدایش نسل جدیدی از مدل‌ها شده که با نام مدل‌های زبانی بزرگ (LLM^۵) شناخته می‌شوند و قادرند طیف گسترده‌ای از وظایف زبانی را تنها با یک فرایند آموزش عمومی، بدون نیاز به بازآموزی ویژه، انجام دهند. در ادامه، دو مدل BERT و GPT به عنوان نمونه‌های شاخص این رویکرد مورد بررسی قرار می‌گیرند.

BERT مدل

مدل BERT که توسط Devlin و همکاران^[۵۳] معرفی شد، یک معماری بر مدل مبدل است که با هدف یادگیری بازنمایی‌های زبانی عمیق و دوسویه طراحی شده است. بر خلاف مدل‌های پیشین که جهت پردازش را محدود به چپ‌به‌راست یا راست‌به‌چپ می‌کردند، BERT از خودتوجهی دوسویه^۱ بهره می‌گیرد و در هر لایه به تمام کلمات موجود در جمله، هم از سمت چپ و هم از سمت راست، توجه

¹Natural Language Processing - NLP

²Transformer

³Bidirectional Encoder Representations from Transformers

⁴Generative Pre-Trained Transformer

⁵Large Language Models

¹Bidirectional Self-attention

می‌کند. این ویژگی باعث می‌شود که مدل بتواند وابستگی‌های معنایی پیچیده را به‌شکل دقیق‌تری مدل‌سازی کند.

مدل BERT با استفاده از دو وظیفه‌ی پوششی آموزش داده می‌شود:

۱. **وظیفه مدل‌سازی زبان پوشیده^{۵۴}**: در این روش، درصدی از توکن‌های ورودی به صورت تصادفی

با یک نشانه ویژه جایگزین می‌شوند و مدل باید با استفاده از بافت دوطرفه، توکن‌های پوشیده

را پیش‌بینی کند. این کار باعث می‌شود که مدل به‌طور همزمان از اطلاعات گذشته و آینده در

جمله بهره ببرد.

۲. **وظیفه پیش‌بینی جمله بعدی**: در این وظیفه، به مدل دو جمله ارائه می‌شود و مدل باید

تشخیص دهد که آیا جمله دوم واقعاً در متن اصلی پس از جمله اول آمده یا خیر. این مرحله به

BERT کمک می‌کند تا روابط سطح جمله و انسجام متنی را بیاموزد.

پس از پیش‌آموزش، BERT می‌تواند برای طیف وسیعی از وظایف زبانی مانند دسته‌بندی متون، پاسخ

به پرسش، برچسب‌گذاری توالی و استنتاج معنایی تنظیم دقیق شود.

GPT مدل

مدل‌های GPT^{۵۵} که توسط OpenAI معرفی شدند، همانند مدل BERT بر پایه معماری مبدل

ساخته شده‌اند. اما برخلاف BERT که از کدگذار مبدل استفاده می‌کند، مدل GPT فقط از رمزگشایی

مدل مبدل استفاده می‌کند. ایده‌ی اصلی GPT این است که:

۱. یک مدل زبانی بزرگ و قدرتمند را به صورت پیش‌آموزش روی یک مجموعه‌داده بسیار عظیم و

بدون برچسب، با هدف پیش‌بینی کلمه بعدی آموزش دهد.

۲. مدل پیش‌آموزش یافته را با تنظیم دقیق روی داده‌های برچسب‌دار برای وظایف خاص مانند

پرسش و پاسخ تطبیق دهد.

نحوه آموزش GPT به فرم مدل‌سازی زبانی خودهمبسته است. یعنی احتمال یک توالی (x_1, x_2, \dots, x)

را به شکل معادله زیر مدل می‌کند:

$$P(x_1, x_2, \dots, x_T) = \prod_{t=1}^T P(x_t | x_1, \dots, x_{t-1}) \quad (16-2)$$

²Masked Language Modeling

بنابراین مدل در طول آموزش، سعی دارد که کلمات بعدی متن ورودی را صرفا با دانستن کلمات قبلی پیش‌بینی کند.

سپس مدل بر روی مجموعه داده برچسب‌دار تنظیم دقیق می‌شود و از آن برای وظایفی مانند پرسش و پاسخ استفاده می‌شود.

۳-۲ شناسایی فعالیت انسان با استفاده از یادگیری خودناظارتی

در فصل حاضر، ابتدا روش‌ها و رویکردهای متدالو در حوزه‌ی شناسایی فعالیت انسان با استفاده از داده‌های حسگر بررسی شد و سپس یادگیری خودناظارتی به عنوان یکی از رویکردهای نوین یادگیری ماشین که بدون نیاز به داده‌های برچسب‌خورده قادر به استخراج ویژگی‌های معنادار است، معرفی گردید. ترکیب این دو حوزه، یعنی به کارگیری یادگیری خودناظارتی در مسئله‌ی شناسایی فعالیت انسان، به دلیل پتانسیل بالای آن در کاهش وابستگی به داده‌های برچسب‌خورده و استفاده‌ی موثر از داده‌های خام، در سال‌های اخیر مورد توجه گسترده‌ی پژوهشگران قرار گرفته است.

با توجه به این که فرایند برچسب‌گذاری داده‌های حسگر، بهویژه در سناریوهای واقعی و مقیاس بزرگ، زمان‌بر، پرهزینه و مستعد خطا است، وجود رویکردهایی که بتوانند از داده‌های بدون برچسب بهره‌برداری کنند، اهمیت بالایی دارد. علاوه بر آن، چالش‌های دیگری نیز برای مسئله‌ی شناسایی فعالیت وجود دارند که به‌طور خلاصه شامل موارد زیر می‌باشد:

۱. وقتی که کاربران مختلف یک فعالیت را انجام می‌دهند، به‌دلیل تفاوت‌هایی که در فیزیولوژی افراد وجود دارد، ممکن است که برای یک فعالیت مشابه داده‌های دارای توزیع‌های نسبتاً متفاوت تولید شود.

۲. حرکت کاربر در طول زمان بسته به مواردی مانند خستگی ممکن است تغییر کند.

۳. ویژگی‌های مربوط به سالمدان و جوانان متفاوت هستند.

۴. در استفاده‌ی عملی از شناسایی فعالیت توسط یادگیری عمیق با تعداد زیادی کاربران جدید مواجه خواهیم شد.

یادگیری خودناظارتی با تعریف وظایف کمکی و استفاده از ساختار ذاتی داده‌ها، این امکان را فراهم می‌سازد که بازنمایی‌های غنی و تعمیم‌پذیری از داده‌های خام به‌دست آید، که در ادامه می‌توان آن‌ها

فصل دوم: ادبیات موضوع و کارهای پیشین

را در مدل‌های نظارت شده برای شناسایی فعالیت انسان به کار گرفت. از سوی دیگر، حجم بسیار بالای داده‌های خام حسگر که از دستگاه‌هایی نظیر تلفن‌های همراه، ساعت‌های هوشمند، مجندوهای سلامتی و سامانه‌های سنجش محیطی به دست می‌آید، فرصت کم‌نظیری برای بهره‌گیری از این رویکرد فراهم می‌کند. با این حال، به کارگیری یادگیری خودناظارتی در حوزه‌ی شناسایی فعالیت انسان با چالش‌های نیز همراه است؛ از جمله طراحی وظایف کمکی مناسب برای داده‌های زمانی چندحسگری و استفاده از روش‌های مناسب برای داده‌افزایی.

در ادامه به بررسی پژوهش‌های انجام شده در زمینه‌ی شناسایی فعالیت با استفاده از یادگیری خودناظارتی می‌پردازم.

۱-۳-۲ ساقه پژوهش

در حوزه‌ی شناسایی فعالیت انسان با استفاده از یادگیری خودناظارتی کارهای متعددی انجام شده است. در این بخش به بررسی تعدادی از روش‌های موفقیت‌آمیز در این حوزه می‌پردازم.

۱-۱-۳-۲ شناسایی فعالیت مبتنی بر روش CPC

با تکیه بر توضیحاتی که پیش‌تر در بخش ۳-۲-۲-۲ [۴۹] درباره‌ی روش CPC ارائه شد، در اینجا نحوه استفاده از آن در شناسایی فعالیت انسان بررسی می‌شود. Haresamudram و همکاران [۵۵] از روش CPC برای ایجاد بازنمایی ویژگی از داده‌های حسگر بهره برده‌اند. در این مطالعه، داده‌های خام حسگرها به قطعات زمانی همپوشان تقسیم می‌شوند. سپس یک شبکه‌ی کدگذار مبتنی بر شبکه‌های پیچشی، بازنمایی‌های سطح بالا را برای هر قطعه استخراج می‌کند. پس از آن، همانند روش CPC، از یک مدل مبتنی بر GRU به عنوان کدگذار خودهمبسته استفاده شده و هزینه‌ی تباینی بر روی پیش‌بینی بازنمایی‌ها اعمال می‌شود.

جدول ۳-۲: نتایج روش CPC در شناسایی فعالیت

Approach	Method type	Mobiact	Motionsense	UCI-HAR	USC-HAD
DeepConvLSTM CPC (end-to-end, 1D Conv Encoder)	Supervised	82.40	85.15	82.83	44.83
	Supervised	83.68	86.66	79.79	49.09
Multi-task self-supervised learning	Unsupervised	75.41	83.30	80.20	45.37
Convolutional autoencoder	Unsupervised	79.58	82.50	80.26	48.82
Masked reconstruction	Unsupervised	76.81	88.02	81.89	49.31
CPC (1D Conv Encoder)	Unsupervised	80.97	89.05	81.65	52.01

پس از اتمام مرحله‌ی پیش‌آموزش، کدگذار آموزش‌دیده با وزن‌های ثبت شده برای استخراج ویژگی به یک مدل دسته‌بندی کاملاً متصل منتقل می‌شود تا آموزش نظارت شده انجام شود. همان‌طور که در جدول ۳-۲ مشاهده می‌شود، عملکرد این روش در بیشتر معیارها قابل قبول است و در برخی موارد حتی از مدل‌های کاملاً نظارت شده پیشی گرفته است.

۲-۱-۳-۲ شناسایی فعالیت مبتنی بر روش SimCLR

روش SimCLR [۲۲] که پیش‌تر در بخش ۳-۲-۲-۲ معرفی شد، یکی از چارچوب‌های مطرح در یادگیری خودناظارتی مبتنی بر یادگیری تباینی است؛ اما این روش در اصل برای داده‌های تصویری ارائه شده و برای استفاده بر روی داده‌های حسگر نیازمند تغییرات و انطباق‌هایی است.

خارتدينف و همکاران [۵۶]^۱ چارچوبی با عنوان CSSHAR^۲ ارائه دادند که نسخه‌ی سازگارشده‌ی SimCLR برای شناسایی فعالیت انسان با داده‌های حسگر است. در این رویکرد، به جای شبکه‌های پیچشی، از یک معماری مبتنی بر مبدل (Transformer) برای استخراج ویژگی استفاده شده است. همچنین به دلیل ماهیت متفاوت سیگنال‌های زمانی نسبت به تصاویر، مجموعه‌ای از داده‌افزایی‌های اختصاصی برای حسگرها طراحی شده که شامل موارد زیر است:

- افزودن نویز: اضافه کردن نویز گوسی تصادفی به سیگنال.
- مقیاس‌گذاری^۳: ضرب دامنه‌ی سیگنال در ضریب تصادفی از یک توزیع گوسی.
- چرخاندن: معکوس کردن علامت نمونه‌های انتخاب شده به صورت تصادفی.
- جایگشت^۴: تقسیم سیگنال به چند بخش و جایه‌جایی تصادفی مقادیر در این بخش‌ها.

بسته به ویژگی‌های مجموعه‌داده، ممکن است همه یا تنها تعدادی از این تبدیلات به کار گرفته شوند که این انتخاب به صورت تجربی و با آزمون و خطا انجام می‌شود. پس از ایجاد نماهای مختلف از هر نمونه، آن‌ها به یک شبکه‌ی نگاشت (Projection Head) وارد می‌شوند و تابع هزینه‌ی تباینی NT-Xent برای یادگیری بازنمایی‌ها به کار می‌رود. در مرحله‌ی تنظیم دقیق، شبکه‌ی نگاشت حذف شده و یک دسته‌بند کاملاً متصل جایگزین آن می‌گردد. در این مرحله، وزن‌های کدگذار مبتنی بر مبدل ثابت نگه داشته می‌شوند تا سرعت آموزش افزایش یابد.

¹Contrastive Self-Supervised Human Activity Recognition

²Scaling

³Permutation

جدول ۲-۴: نتایج روش CSSHAR

Method	Type	Mean F1-Score	
		UCI-HAR	USC-HAD
DeepConvLSTM	Sup.	73.68	25.57
Transformer	Sup.	86.62	39.8
Multi-task SSL	SSL	73.89	31.35
CAE	SSL	84.15	51.66
Masked Reconstruction	SSL	81.37	46.19
CSSHAR	SSL	88.26	48.73

همان‌طور که در جدول ۴-۲ مشاهده می‌شود، ترکیب سازوکار یادگیری تباینی SimCLR با توانایی مبدل در مدل‌سازی وابستگی‌های طولانی‌مدت سیگنال، به بهبود قابل توجه دقت در شناسایی فعالیت منجر شده است. این روش به‌ویژه در شرایطی که داده‌های برچسب‌خورده محدود هستند، عملکردی رقابتی یا حتی برتر نسبت به مدل‌های نظرات‌شده نشان داده است.

۳-۱-۳-۲ شناسایی فعالیت مبتنی بر یادگیری مشارکتی

جين و همکاران [۵۷] یک چارچوب یادگیری خودنظراتی نوین و مشارکتی^۱ تحت عنوان ColloSSL را برای پیش‌آموزش مدل‌های بازشناسی فعالیت انسان ارائه دادند. این روش برای محیط‌هایی طراحی شده است که در آن چندین حسگر به‌طور همزمان داده‌های مربوط به یک فعالیت را ثبت می‌کنند. این محیط، یک سیستم چند دستگاهی همگام با زمان (TSMDS^۲) نامیده می‌شود که در آن، داده‌های ثبت‌شده توسط حسگرهای مختلف کاملاً همگام هستند.

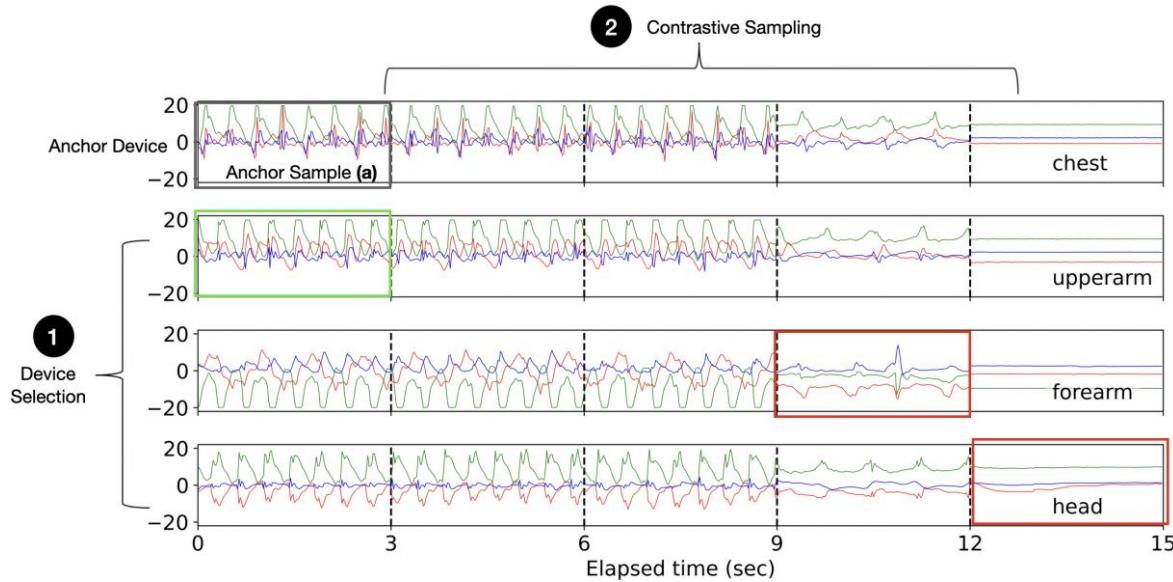
ایده‌ی اصلی ColloSSL این است که به جای تولید داده‌های افزوده به صورت مصنوعی (مانند افزودن نویز یا دوران)، از داده‌های حسگرهای مختلف به عنوان تبدیل‌های طبیعی^۳ از یکدیگر استفاده شود. به عبارت دیگر، داده‌ی ثبت‌شده توسط حسگر روی مج دست و حسگر روی قفسه‌ی سینه، دو «نما» یا «دیدگاه» متفاوت از یک فعالیت یکسان (مثل راه‌رفتن) هستند. هدف یادگیری تباینی در این روش، نزدیک کردن بازنمایی این نماهای مختلف از یک فعالیت و دور کردن آن‌ها از بازنمایی فعالیت‌های دیگر است.

برای مثال، همان‌طور که در شکل ۱۶-۲ نمایش داده شده است، فرض کنید می‌خواهیم مدل را با محوریت داده‌های حسگر قفسه سینه پیش‌آموزش دهیم. در این حالت، این حسگر به عنوان دستگاه

¹Collaborative

²Time-Synchronous Multi-Device System

³Natural Transformations



شکل ۲: انتخاب حسگرهای مثبت و منفی در ColloSSL

محوری^۱ انتخاب می‌شود. سپس، نمونه‌های داده از این دستگاه با نمونه‌های سایر دستگاه‌ها مقایسه می‌شوند. فرایند کلی یادگیری در این مقاله به صورت زیر است:

۱. ابتدا، یک استخراج‌کننده‌ی ویژگی (در این مقاله، یک شبکه‌ی پیچشی یک‌بعدی با سه لایه) با وزن‌های تصادفی مقداردهی اولیه می‌شود.
۲. یک دسته‌ی تصادفی از داده‌ها انتخاب می‌شود. این دسته شامل پنجره‌های زمانی (مثلاً به طول ۳ ثانیه) از تمام حسگرها است. نکته‌ی کلیدی این است که این پنجره‌ها از نظر زمانی با یکدیگر همگام هستند.
۳. برای شروع یادگیری تباینی، یک نمونه از یک دستگاه خاص (مثلاً قفسه سینه) به عنوان نمونه‌ی محوری^۲ انتخاب می‌شود.
۴. نمونه‌های زمانی متناظر با نمونه‌ی محوری از سایر دستگاه‌ها (مانند مچ، ران و غیره) به عنوان نمونه‌های مثبت در نظر گرفته می‌شوند. این نمونه‌ها دیدگاه‌های متفاوتی از همان فعالیت در همان لحظه هستند.
۵. تمام نمونه‌های دیگر در آن دسته (که متعلق به پنجره‌های زمانی دیگری هستند) به عنوان نمونه‌های منفی انتخاب می‌شوند.

¹Anchor Device

²Anchor Sample

۶. تمام این نمونه‌ها (محوری، مثبت و منفی) به استخراج‌کننده‌ی ویژگی داده می‌شوند تا بازنمایی نهفته‌ی آن‌ها استخراج شود. سپس با استفاده از یک تابع زیان تباینی، مدل آموزش داده می‌شود. هدف این تابع زیان، به حداقل رساندن شباهت بین بازنمایی نمونه‌ی محوری و نمونه‌های مثبت، و به حداقل رساندن شباهت آن با نمونه‌های منفی است.

۷. مراحل ۳ تا ۶ برای تمام نمونه‌های موجود در دسته تکرار می‌شوند تا مدل به طور کامل یاد بگیرد که چگونه بازنمایی‌های معناداری از فعالیت‌ها استخراج کند. این فرایند با انتخاب دسته‌های جدید ادامه می‌یابد.

در نهایت، مزیت کلیدی چارچوب ColloSSL در تعریف هوشمندانه‌ی وظیفه‌ی پیش‌آموزشی آن نهفته است. این روش با بهره‌گیری از داده‌های همگام حسگرهای مختلف، یک سیگنال ناظارتی غنی و طبیعی را بدون نیاز به هیچ‌گونه برچسب‌گذاری انسانی یا افزونه‌سازی مصنوعی ایجاد می‌کند. در نتیجه، استخراج‌کننده‌ی ویژگی حاصل، بازنمایی‌های عمیق و پایدار از فعالیت‌های انسانی را فرا می‌گیرد که می‌تواند به عنوان یک پایه‌ی قدرتمند برای بهبود عملکرد و کاهش نیاز به داده‌های برچسب‌دار در مدل‌های دسته‌بندی نهایی عمل کند.

۴-۲ جمع‌بندی

در این فصل، ادبیات موضوع و کارهای پیشین در حوزه‌ی شناسایی فعالیت انسان و یادگیری خودناظارتی مورد بررسی قرار گرفت. ابتدا به بررسی روش‌های شناسایی فعالیت انسان پرداختیم. سپس چالش اصلی روش‌های یادگیری عمیق، یعنی نیاز به حجم بالای داده‌های برچسب‌دار، تشریح شد. در ادامه، یادگیری خودناظارتی به عنوان یک راهکار موثر برای غلبه بر این محدودیت معرفی گردید که با تعریف وظایف پوششی بر روی داده‌های خام، به یادگیری بازنمایی‌های غنی می‌پردازد. در نهایت، با مرور پژوهش‌های ترکیبی در این دو حوزه، نشان داده شد که اقتباس روش‌های خودناظارتی، به ویژه رویکردهای تباینی، برای داده‌های حسگری به نتایج امیدوارکننده‌ای منجر شده و می‌تواند عملکردی قابل مقایسه با روش‌های کاملاً ناظارت‌شده، با نیاز به مراتب کمتری به داده‌های برچسب‌دار، ارائه دهد.

فصل ۳

روش پیشنهادی

در این فصل، به تشریح کامل روش پیشنهادی می‌پردازیم. نخست، به عنوان پیش‌نیاز، به تعریف تبدیل موجک^۱ و اسکالوگرام^۲ خواهیم پرداخت. سپس، روش پایه و معماری آن را به تفصیل مورد بررسی قرار می‌دهیم. در نهایت، با تحلیل این معماری، زمینه‌های مستعد بهبود در آن شناسایی شده و سپس نوآوری‌های ارائه شده در این پژوهش که برای رفع این چالش‌ها طراحی شده‌اند، به تفصیل تشریح خواهند شد.

۱-۳ تبدیل موجک

تبدیل موجک یکی از ابزارهای قدرتمند در پردازش سیگنال است که برای تحلیل سیگنال‌ها در هر دو حوزه‌ی زمان و فرکانس به صورت همزمان به کار می‌رود. این ویژگی، تبدیل موجک را از ابزارهای کلاسیک مانند تبدیل فوریه^۳ متمایز می‌سازد. تبدیل فوریه، سیگنال را به مولفه‌های فرکانسی تشکیل‌دهنده‌ی آن از طریق دو طیف دامنه و فاز تجزیه می‌کند. طیف دامنه نشان می‌دهد که هر مولفه‌ی فرکانسی با چه شدتی در کل سیگنال حضور دارد، اما اطلاعاتی در مورد زمان وقوع آن ارائه نمی‌دهد. اگرچه طیف فاز به صورت غیرمستقیم حاوی اطلاعات زمانی است، اما تفسیر آن برای محلی‌سازی رویدادها در زمان بسیار دشوار و غیرمستقیم است. به عبارت دیگر، تبدیل فوریه در نمایش همزمان رویدادها در حوزه‌ی زمان و فرکانس دارای محدودیت است. در مقابل، تبدیل موجک با استفاده از توابعی به نام موجک مادر^۴ که در زمان و فرکانس محدود هستند، این محدودیت را برطرف می‌سازد.

ایده‌ی اصلی در تبدیل موجک، مقایسه‌ی سیگنال با نسخه‌های جابه‌جا شده و مقیاس‌گذاری شده از یک موجک مادر است. جابه‌جایی به منظور محلی‌سازی تحلیل در زمان و مقیاس‌گذاری به منظور محلی‌سازی تحلیل در فرکانس انجام می‌شود. مقیاس‌های کوچک (فشرده‌سازی موجک) متناظر با فرکانس‌های بالا و مقیاس‌های بزرگ (کشیده‌سازی موجک) متناظر با فرکانس‌های پایین هستند.

تبدیل موجک به دو دسته‌ی اصلی گسسته (CWT^۵) و پیوسته (DWT^۶) تقسیم می‌شوند که در این پژوهش از تبدیل موجک پیوسته استفاده کرده‌ایم.

¹ Wavelet Transform

² Scalogram

³ Fourier Transform

⁴ Mother Wavelet

⁵ Discrete Wavelet Transform

⁶ Continuous Wavelet Transform

تبدیل موجک یک سیگنال زمانی ($x(t)$) به فرم زیر تعریف می‌گردد:

$$CWT(a, b) = \int_{-\infty}^{\infty} x(t)\psi_{a,b}(t)dt \quad (1-3)$$

که در این معادله، $\psi_{a,b}(t)$ موجک دختر^۱ نامیده می‌شود که نسخه‌ی مقیاس‌گذاری و جابه‌جا شده است و به فرم زیر تعریف می‌گردد:

$$\psi_{a,b}(t) = \frac{1}{\sqrt{|a|}}\psi\left(\frac{t-b}{a}\right) \quad (2-3)$$

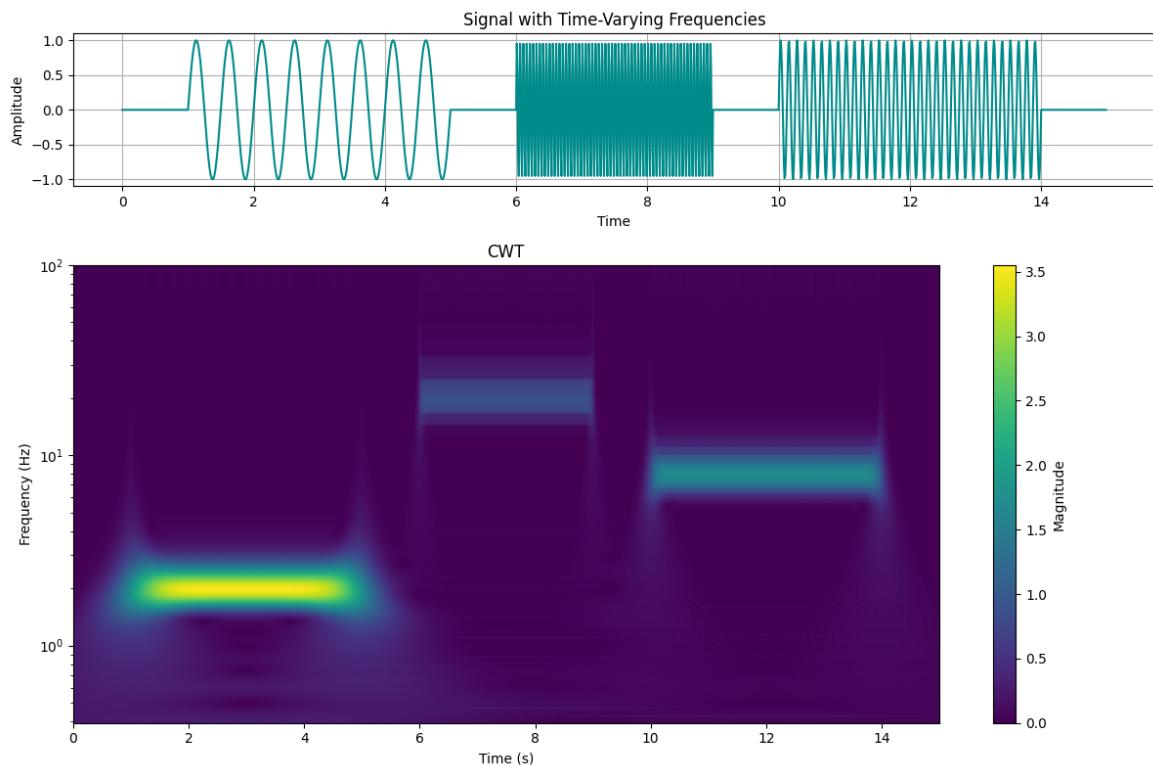
در این رابطه، پارامتر a مقیاس است که با فرکانس رابطه‌ی معکوس دارد. پارامتر b بیانگر میزان جابه‌جای در محور زمان است و ضریب $\frac{1}{\sqrt{|a|}}$ برای نرمال‌سازی انرژی موجک به کار می‌رود. $(t)\psi$ نیز یک تابع ریاضی است که باید دارای میانگین صفر و انرژی محدود در تمام دامنه باشد. برای مثال موجک مورلت^۲ که فرمول آن به فرم رابطه‌ی ۳-۳ می‌باشد، یک موجک بسیار کاربردی در حوزه‌ی سیگنال‌های دنیای واقعی، به خصوص سیگنال‌های مربوط به فعالیت انسان می‌باشد. چرا که در این سیگنال‌ها، معمولاً نوسانات و فرکانس‌های غیر ایستاداریم که به سرعت محو می‌شوند و موجک مورلت به دلیل دارا بودن $e^{-\frac{t^2}{2}}$ که به آن اثر محوشده‌گی را می‌دهد، برای این‌گونه سیگنال‌ها بسیار مناسب است.

$$\psi(t) = \frac{\cos(\omega_0 t)e^{-\frac{t^2}{2}}}{\pi^{\frac{1}{4}}} \quad (3-3)$$

خروجی CWT مجموعه‌ای از ضرایب است که میزان شباهت سیگنال ($x(t)$) را با موجک با مقیاس a و زمان b نشان می‌دهد. این ضرایب یک نمایش دو بعدی از سیگنال یک بعدی اولیه ارائه می‌دهند که به آن اسکالوگرام می‌گویند. در یک اسکالوگرام محور افقی بیانگر میزان جابه‌جای یا همان b و محور عمودی بیانگر مقیاس‌ها می‌باشد. یک نمونه اسکالوگرام در شکل ۳-۳ قابل مشاهده می‌باشد.

¹Daughter Wavelet

²Morlet Wavelet



شکل ۳-۱: نمونه اسکالوگرام برای یک موج متغیر در زمان

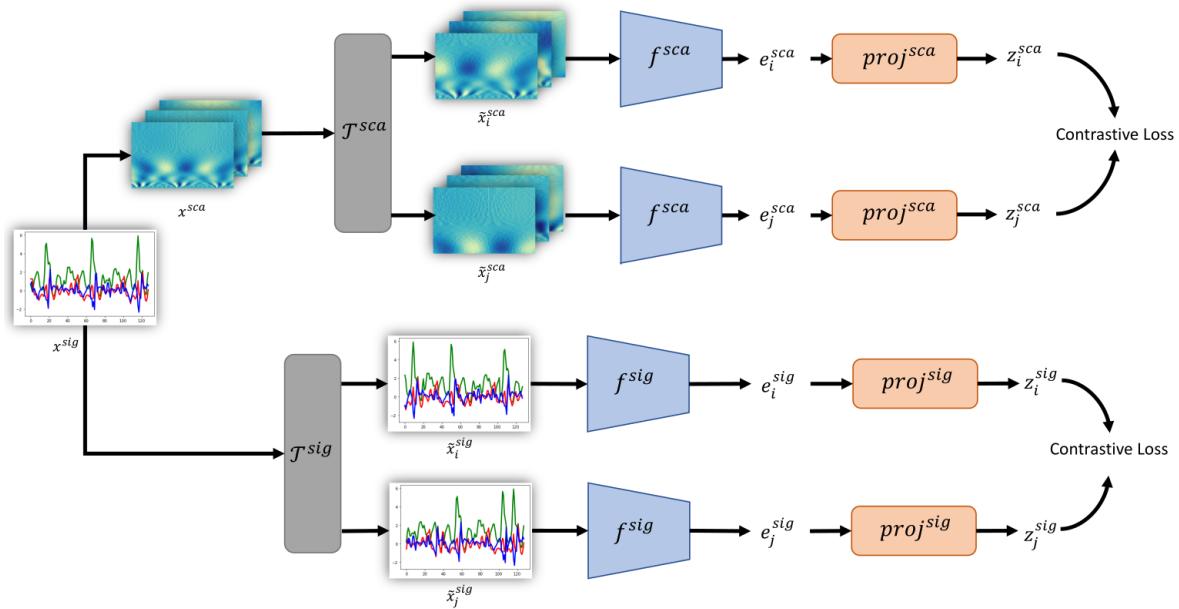
۲-۳ روش پایه

معماری کلی پیشآموزش خودناظارتی روش پیشنهادی در شکل ۲-۳ قابل مشاهده است. این معماری توسط طاقانکی و همکاران [۵۸] برای پیشآموزش یک مدل جهت استخراج بازنمایی‌های مفید از روی داده‌های سیگنال مربوط به شناسایی فعالیت انسان ارائه شد. این معماری با این فرضیه طراحی شده است که اطلاعات مفید مربوط به فعالیت در هر دو حوزه زمان و فرکانس نهفته است. به همین منظور، از دو مسیر پردازشی مجزا برای استخراج ویژگی از سیگنال ورودی استفاده می‌شود. یک کدگذار سیگنال که وظیفه‌ی آن پردازش داده‌های خام سیگنال می‌باشد و یک کدگذار اسکالوگرام که وظیفه‌ی آن پردازش اسکالوگرام‌های حاصل از تبدیل موجک بر روی داده‌های خام سیگنال می‌باشد.

در ادامه به بررسی جزئیات مربوط به هر دو مسیر آموزش کدگذار می‌پردازیم.

۱-۲-۳ کدگذار سیگنال

این بخش از معماری وظیفه‌ی یادگیری از روی داده‌های خام سیگنال در حوزه‌ی زمان را بر عهده دارد. برای آموزش این بخش، از چارچوب یادگیری تباینی SimCLR که در فصل قبل درباره‌ی جزئیات آن



شکل ۲-۳: معماری کلی پیشآموزش در روش پیشنهادی

توضیح دادیم استفاده شده است. کدگذار سیگنال یا \mathcal{H}_{sig} از یک شبکه عصبی شامل ۳ لایه پیچشی یک بعدی تشکیل شده است. سپس خروجی \mathcal{H}_{sig} به یک شبکه نگاشت داده می‌شود و در نهایت بر روی خروجی شبکه نگاشت تابع هزینه تباینی NT-Xent^۱ به فرم معادله ۱۴-۲ اعمال می‌شود. پیاده‌سازی چارچوب SimCLR مستلزم تعریف مجموعه‌ای از تبدیلات داده‌افزایی است. در این مسیر، تبدیلات زیر برای اعمال بر روی سیگنال خام زمانی به کار گرفته شده‌اند:

- **نویز:** یک نویز تصادفی گوسی با میانگین صفر و انحراف معیار ۰.۱ اعمال می‌شود.
- **مقیاس‌دهی^۲:** یک مقدار تصادفی با میانگین ۱ و انحراف معیار ۰.۲ در سیگنال ضرب می‌شود.
- **معکوس‌سازی زمانی^۳:** با یک احتمال ثابت، پنجره سیگنال ورودی نسبت به زمان معکوس می‌گردد. ایده‌ی پشت این تبدیل این است که یک فعالیت ممکن است در جهت برعکس انجام شود.
- **بر زدن کانال‌ها^۴:** با یک احتمال ثابت، کانال‌های سیگنال ورودی به صورت تصادفی با یکدیگر جابه‌جا می‌شوند. ایده‌ی پشت این تبدیل این است که ممکن است حسگر در یک جهتی عمود بر جهت استاندارد قرار گرفته باشد. مثلاً جهت قرار گرفتن تلفن همراه هوشمند در جیب شخص.

¹Scale

²Time Flip

³Channel Shuffle

نکته‌ی مهم در رابطه با این تبدیل این است که فقط کانال‌های مربوط به هر دستگاه باید با یکدیگر جایه‌جا شوند. مثلاً اگر کانال‌های اول تا سوم برای شتاب‌سنج و کانال‌های چهارم تا ششم برای ژیروسکوپ باشند، کانال‌های اول تا سوم با یکدیگر و کانال‌های چهارم تا ششم نیز با یکدیگر جایه‌جا می‌شوند.

• **جایگشت:** سیگنال ورودی به قطعاتی تقسیم می‌شود و این قطعات با یکدیگر جایه‌جا می‌شوند. ایده‌ی پشت این تبدیل این است که ممکن است ترتیب بخش‌های مربوط به انجام یک فعالیت تغییر کنند.

• **چرخش:** سیگنال ورودی حول یک محور تصادفی و به میزان درجه‌ی تصادفی چرخش داده می‌شود. در واقع چرخش، حالت کلی‌تر از بر زدن کانال‌ها می‌باشد.

انتخاب تبدیلات بهینه و ترکیب آن‌ها، فرایندی تجربی و وابسته به مشخصات هر مجموعه داده است. اما به‌طور کلی تبدیلات نه باید به‌قدرتی سخت و شدیداً تصادفی باشند که مدل کلا نتواند الگویی کشف کند، و نه باید به‌قدرتی ساده باشند که مدل به سراغ کشف بازنمایی‌های مفید و جامع نرود.

۲-۲-۳ کدگذار اسکالوگرام

این بخش از معماری وظیفه‌ی یادگیری از روی اسکالوگرام‌های حاصل از اعمال تبدیل موجک بر روی سیگنال‌ها را بر عهده دارد. همانند کدگذار سیگنال، در کدگذار اسکالوگرام نیز از روش SimCLR برای آموزش مدل استفاده شده است. کدگذار اسکالوگرام یا H_{sca} از یک شبکه عصبی شامل ۳ لایه پیچشی دو بعدی تشکیل شده است. قبل از این که یک پنجره‌ی چندکاناله از داده‌ها وارد این شبکه شوند، بر روی آن‌ها تبدیل موجک اعمال می‌شود و با اسکالوگرام حاصل می‌توان مانند یک تصویر رفتار کرد. روش‌های داده‌افزایی به کار رفته در کدگذار اسکالوگرام نیز به شرح زیر می‌باشند:

• **اعوجاج رنگ تصادفی:**^۱ در این تبدیل، هر کانال مربوط به هر حسگر را یک رنگ در نظر می‌گیریم و رنگ آن را به صورت تصادفی دچار اعوجاج و تغییرات می‌کنیم. مثلاً شدت رنگ‌ها را افزایش می‌دهیم و یا آن را به فرم سیاه و سفید در می‌آوریم.

• **برش تصادفی:** به صورت تصادفی اسکالوگرام‌ها را برش می‌زنیم.

• **معکوس‌سازی زمانی:** اسکالوگرام‌ها را به صورت افقی معکوس می‌کنیم.

^۱Random Color Distortion

۳-۲-۳ تنظیم دقیق مدل

پس از این که هر دو کدگذار فرایند پیشآموزش را پشت سر گذاشتند، مانند روش SimCLR شبکه نگاشت کنار گذاشته می‌شود و به جای آن یک شبکه مت Shank از دو لایه‌ی تماماً متصل برای دسته‌بندی قرار داده می‌شوند. این کار را برای هر دو کدگذار به صورت مستقل انجام می‌دهیم. سپس وزن‌های لایه‌های پیچشی هر دو کدگذار تثبیت می‌شوند. این عمل دو هدف اصلی را دنبال می‌کند: اولاً، از بیش‌برازش مدل بر روی داده‌های برچسبدار که معمولاً حجم کمتری دارند جلوگیری می‌کند و ثانياً، بازنمایی‌های کلی و مفیدی که در مرحله پیشآموزش یاد گرفته شده‌اند، حفظ می‌شوند. در نهایت دو دسته‌بندی کننده خواهیم داشت که هر یک جداگانه آموزش دیده‌اند؛ یکی بر روی داده‌های خام و دیگری بر روی اسکالوگرام‌ها. در نهایت برای به دست آمدن دسته‌بند نهایی، با استی خروجی دو دسته‌بند با یکدیگر ادغام شوند. در این روش، از راهبرد ادغام در سطح امتیاز^۱ که به آن ادغام دیرهنگام^۲ نیز گفته می‌شود، استفاده شده است. در این روش، بردارهای احتمال خروجی از هر دو دسته‌بند (به عنوان مثال از طریق میانگین‌گیری) با یکدیگر ترکیب شده و سپس دسته‌ی نهایی بر اساس بیشترین امتیاز در بردار حاصل انتخاب می‌گردد.

۳-۳ نوآوری‌های پیشنهادی

همان‌طور که تشریح شد، روش پایه یک چارچوب قدرتمند و منطقی برای یادگیری بازنمایی از سیگنال‌های فعالیت انسان ارائه می‌دهد. با این وجود، تحلیل دقیق این معماری نشان می‌دهد که چندین مولفه کلیدی در آن، ظرفیت بهبود و بهینه‌سازی را دارا هستند. در این پژوهش، سه حوزه اصلی برای ارتقای مدل پایه شناسایی و مورد هدف قرار گرفته است:

- **الگوریتم یادگیری تباینی:** با وجود این که چارچوب یادگیری تباینی SimCLR عملکرد خوبی از خود در حوزه‌های مختلف نشان داده است، می‌توان از رویکردهای دیگر یادگیری تباینی که در دیگر حوزه‌ها عملکرد بهتری از خود نشان داده‌اند استفاده کرد.

- **راهبرد داده‌افزایی:** روش‌های داده‌افزایی اعمال شده بر روی اسکالوگرام‌ها مستقیماً از حوزه بینایی کامپیوتر اقتباس شده‌اند و ممکن است بهترین گزینه برای داده‌های زمان-فرکانس نباشند. بنابراین، یک راهبرد داده‌افزایی جدید و متناسب با ماهیت این داده‌ها ارائه می‌شود.

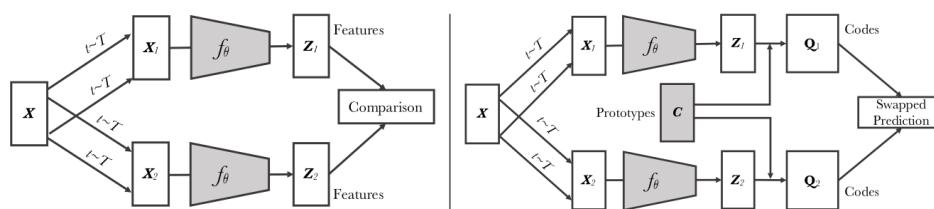
¹Score-Level Fusion

²Late Fusion

۱-۳-۳ الگوریتم یادگیری تباینی SwAV

الگوریتم یادگیری تباینی تعویض انتساب‌ها میان نماها یا به اختصار SwAV^۱، یک الگوریتم یادگیری تباینی مبتنی بر خوشبندی می‌باشد که توسط کارون و همکاران [۲۴] ارائه شد. این الگوریتم با هدف یادگیری بازنمایی‌های بصری قدرتمند بدون نیاز به برچسب‌های انسانی طراحی شده و توانسته است فاصله‌ی عملکردی میان روش‌های خودناظارتی و نظارت‌شده را به شکل چشمگیری کاهش دهد.

ایده‌ی اصلی این الگوریتم بر یک سازوکار پیش‌بینی تعویض‌شده^۲ استوار است. در این روش، به جای مقایسه‌ی مستقیم بردارهای ویژگی حاصل از نماهای مختلف یک تصویر با یکدیگر (کاری که در روش‌های متداولی مانند SimCLR انجام می‌شود)، مدل می‌آموزد که تخصیص خوشبینی یک نما را از روی بردار ویژگی نمای دیگر پیش‌بینی کند. به عبارت دیگر، اگر دو نمای مختلف از یک تصویر ورودی داشته باشیم، ویژگی‌های استخراج‌شده از نمای اول باید آنقدر غنی و معنادار باشند که بتوان با استفاده از آن‌ها، کد خوشبینی مربوط به نمای دوم را پیش‌بینی کرد و بالعکس. این فرایند، مدل را قادر می‌سازد تا بازنمایی‌هایی را بیاموزد که نسبت به تبدیلات و تغییرات اعمال‌شده بر روی تصویر (مانند برش، تغییر رنگ و دوران) نامتفاوت و پایدار باشند.



شکل ۱-۳-۳: مقایسه‌ی الگوریتم SwAV (سمت راست) و SimCLR (سمت چپ)

یکی از نوآوری‌های کلیدی روش SwAV، انجام فرایند خوشبندی به صورت برخط^۳ است. در روش‌های مبتنی بر خوشبندی پیشین (مانند روش خوشبینی عمیق^۴ [۵۹]), فرایند یادگیری به دو مرحله‌ی مجزا و غیر برخط تقسیم می‌شد: ابتدا تمام داده‌ها برای تخصیص به خوشبینی پردازش شده و سپس از این تخصیص‌ها به عنوان برچسب‌های کاذب برای آموزش شبکه استفاده می‌شد. این فرایند نیازمند عبورهای چندباره از کل مجموعه داده بود و مقیاس‌پذیری الگوریتم را با چالش مواجه می‌کرد. اما در SwAV تخصیص خوشبینی از داده‌ها و به صورت آنی

¹Swapping Assignments between Views

²Swapped Prediction

³Online

⁴Deep Cluster

انجام می‌شود. این ویژگی باعث می‌شود که SwAV بسیار کارآمد بوده و بتواند بر روی مجموعه‌داده‌های بسیار بزرگ نیز به راحتی آموزش ببیند.

۱-۳-۳ تابع هزینه پیش‌بینی تعویض شده

همان‌طور که اشاره کردیم، هسته‌ی اصلی الگوریتم SwAV بر مبنای یک تابع هزینه‌ی منحصر به فرد به نام «پیش‌بینی تعویض شده» قرار دارد. این تابع هزینه، سازگاری میان نماهای مختلف یک تصویر را نه از طریق مقایسه‌ی مستقیم ویژگی‌ها، بلکه از طریق مقایسه‌ی تخصیص خوش‌های آن‌ها می‌سنجد. فرض کنید برای یک تصویر ورودی، دو نمای مختلف با اعمال تبدیلات تصادفی ایجاد کرده‌ایم و پس از عبور آن‌ها از شبکه‌ی کدگذار، بردارهای ویژگی z_t و z_s را به دست آورده‌ایم. تابع هزینه‌ی SwAV برای این زوج ویژگی به صورت زیر تعریف می‌شود:

$$L(z_t, z_s) = \ell(z_t, q_s) + \ell(z_s, q_t) \quad (4-3)$$

در این رابطه:

- z_t و z_s بردارهای ویژگی نرم‌مال‌شده‌ی حاصل از دو نما هستند.
- q_t و q_s کدهای خوش‌های تخصیص نرم^۱ مربوط به هر یک از این ویژگی‌ها می‌باشند. این کدها که در عمل یک توزیع احتمال هستند، نشان می‌دهند که هر ویژگی تا چه حد به هر یک از خوش‌های موجود در مدل تعلق دارد.
- تابعی است که میزان اختلاف میان یک توزیع پیش‌بینی شده (که از z حاصل می‌شود) و یک کد هدف (q) را اندازه‌گیری می‌کند.

فرمول ۴-۳ دارای دو بخش متقاضن است. بخش اول، $\ell(z_t, q_s)$ مدل را وادار می‌کند تا با استفاده از ویژگی نمای اول (z_t) کد خوش‌های نمای دوم (q_s) را پیش‌بینی کند. بخش دوم نیز این فرایند را به صورت معکوس انجام می‌دهد. به همین دلیل این سازوکار، پیش‌بینی تعویض شده نام گرفته است. تابع $\ell(z, q)$ در عمل یک تابع هزینه آنتروپی متقاطع است که اختلاف میان دو توزیع احتمال را

می‌سنجد:

^۱Soft Assignment Vectors

- توزیع پیش‌بینی شده p : یک توزیع احتمال که از بردار ویژگی z به دست می‌آید و نشان‌دهنده‌ی پیش‌بینی مدل برای تخصیص آن ویژگی به خوش‌هاست.
- کد هدف q : یک توزیع احتمال که به عنوان برچسب نرم عمل می‌کند و از قبل برای نمای دیگر محاسبه شده است.

محاسبه توزیع پیش‌بینی شده (p): از ویژگی به احتمال

توزیع p , پیش‌بینی مدل از میزان تعلق یک بردار ویژگی z به هر یک از K خوش‌هه موجود را نشان می‌دهد. مراکز خوش‌هه را با بردارهایی تحت عنوان پیش‌نمونه^۱ نشان می‌دهیم که این مراکز خوش‌هه مجموعه‌ای از بردارهای قابل یادگیری $C = \{c_1, \dots, c_k\}$ در فضای ویژگی هستند.

فرایند تبدیل بردار ویژگی z به توزیع احتمال p در دو مرحله‌ی اصلی انجام می‌شود: ابتدا محاسبه امتیازات شباht و سپس تبدیل این امتیازات به یک توزیع احتمال معتر.

در گام نخست، مدل میزان شباht میان بردار ویژگی z_t و هر یک از K بردار مرکز خوش‌هه را محاسبه می‌کند. این شباht از طریق ضرب داخلی میان بردار ویژگی و هر بردار c_k به دست می‌آید. نتیجه‌ی این عملیات، یک بردار با K درایه است که هر درایه‌ی آن، که هر درایه‌ی آن، امتیاز شباht خام یا Logit نامیده می‌شود و نشان‌دهنده‌ی میزان تطابق ویژگی با آن بردار مرکز خوش‌هه خاص است.

امتیازات خام به دست آمده، مقادیری نامحدود هستند و مجموع آن‌ها لزوماً برابر با یک نیست. برای تبدیل آن‌ها به یک توزیع احتمال معتر، از تابع سافت‌مکس^۲ استفاده می‌شود. این تابع، هر امتیاز را به یک مقدار در بازه‌ی $(0, 1)$ نگاشت کرده و تضمین می‌کند که مجموع تمام احتمالات برابر با یک شود. در این فرایند، از یک پارامتر دما τ نیز برای کنترل تیزی یا همواری توزیع خروجی استفاده می‌شود. دمای پایین‌تر منجر به توزیعی تیزتر (با قطعیت بیشتر) و دمای بالاتر منجر به توزیعی هموارتر می‌گردد. بنابراین، احتمال تعلق ویژگی z_t به خوش‌هه k که با $p_t^{(k)}$ نمایش داده می‌شود، از طریق فرمول زیر محاسبه می‌گردد:

$$p_t^{(k)} = \frac{\exp(\frac{1}{\tau} z_t^\top c_k)}{\sum_{k'=1}^K \exp(\frac{1}{\tau} z_t^\top c_{k'})} \quad (5-3)$$

در این فرمول، $z_t^{k c_k}$ همان ضرب داخلی میان بردار ویژگی و بردار مرکز خوش‌هه k است. مخرج کسر نیز مجموع مقادیر صورت کسر برای تمام K مرکز خوش‌ه است که برای نرمال‌سازی به کار می‌رود. این توزیع

¹Prototype

²Softmax

فصل سوم: روش پیشنهادی

همان پیش‌بینی مدل برای بردار ویژگی z_t است که در تابع هزینه‌ی آنتروپویی متقطع با کد هدف p_t مقایسه خواهد شد.

محاسبه کدهای هدف q : تخصیص بهینه خوشها

اکنون به بخش پیچیده‌تر محاسبه‌ی تابع هزینه، یعنی نحوه‌ی تعیین کدهای هدف q می‌رسیم. برخلاف توزیع پیش‌بینی شده p که به سادگی از خروجی شبکه به دست می‌آمد، محاسبه‌ی q نیازمند یک سازوکار ویژه برای جلوگیری از یک مشکل اساسی است.

یک چالش مهم در روش‌های مبتنی بر خوشبندی، پدیدهای است که با عنوانی چون پاسخ بدیهی^۱ یا فروپاشی مدل^۲ شناخته می‌شود. این مشکل زمانی رخ می‌دهد که مدل یک راه حل ساده و بی‌ارزش برای کمینه کردن تابع هزینه پیدا کند. در مسئله‌ی ما، مدل می‌تواند یاد بگیرد که تمام بردارهای ویژگی را به یک خوشی یکسان تخصیص دهد. در این حالت، اگرچه تابع هزینه به سرعت کمینه می‌شود، اما بازنمایی‌های آموخته‌شده هیچ اطلاعات مفیدی در مورد تمایز میان تصاویر مختلف نخواهند داشت و عملابی‌ارزش خواهند بود.

برای مقابله با این پدیده، الگوریتم SwAV یک قید افزار برابر^۳ را بر روی فرایند تخصیص خوشها اعمال می‌کند. هدف این قید، وادار کردن مدل به استفاده از تمام ظرفیت بردارهای مربوط به مراکز خوشها است. این قید تضمین می‌کند که نمونه‌های موجود در یک دسته آموزشی از داده‌ها، تا حد امکان به صورت یکنواخت و برابر میان تمام K خوش موجود توزیع شوند.

مسئله‌ی تخصیص بهینه‌ی مجموعه‌ای از منابع (ویژگی‌های مربوط به داده‌ها) به مجموعه‌ای از مقاصد (خوشها) تحت یک سری قیود، یک مسئله‌ی کلاسیک در ریاضیات و علوم کامپیوتر است که با عنوان مسئله‌ی انتقال بهینه^۴ شناخته می‌شود. الگوریتم SwAV از این چارچوب قدرتمند ریاضی برای محاسبه‌ی ماتریس تخصیص خوشها (Q) استفاده می‌کند.

برای درک بهتر، می‌توان این مسئله را با یک مثال ملموس توصیف کرد. فرض کنید B انبار (بردارهای ویژگی) و K فروشگاه (خوشها) داریم. هدف، طراحی یک برنامه‌ی حمل و نقل (ماتریس Q) است که کالاها را از انبارها به فروشگاهها به بهینه‌ترین شکل ممکن ارسال کند. بهینه بودن در اینجا به دو معناست: اولاً، مجموع شباهت میان انبارها و فروشگاههای متناظرشان بیشینه شود و ثانیاً، قیود توزیع عادلانه (قید

¹Trivial Solution

²Model Collapse

³Equipartition Constraint

⁴Optimal Transport

افراز برابر) که در بخش قبل به آن اشاره شد، رعایت گردد.

به زبان ریاضی، این مسئله به صورت یافتن ماتریس Q از میان مجموعه‌ی تمام ماتریس‌های معتبر \mathcal{Q} (که در قیود افراز برابر صدق می‌کنند) تعریف می‌شود که عبارت زیر را بیشینه کند:

$$\max_{Q \in \mathcal{Q}} \text{Tr}(Q^T C^T Z) + \epsilon H(Q) \quad (6-3)$$

در این رابطه:

- عبارت $\text{Tr}(Q^T C^T Z)$ ، مجموع وزن دار شباهت‌ها میان تمام ویژگی‌ها و تمام مراکز خوشه‌هاست.
هدف اصلی، بیشینه کردن این مقدار است.

- عبارت $\epsilon H(Q)$ ، یک جمله‌ی تنظیم‌گر آنتروپی^۱ است. $H(Q)$ آنتروپی ماتریس تخصیص است و افزودن آن با ضریب کوچک باعث می‌شود که تخصیص‌ها نرم‌تر و هموارتر باشند و از تخصیص‌های بسیار قطعی و سخت جلوگیری می‌کند. این کار به پایداری فرایند بهینه‌سازی کمک شایانی می‌کند.

بنابراین، الگوریتم باید ماتریس Q را از میان تمام ماتریس‌های معتبر پیدا کند که تابع هدف فرمول ۶-۳ را بیشینه سازد. یافتن این ماتریس به صورت قطعی بسیار پیچیده است، اما الگوریتم‌های کارآمدی برای حل تقریبی آن وجود دارد.

برای حل مسئله‌ی بهینه‌سازی در فرمول ۶-۳، الگوریتم SwAV از یک روش تکرارشونده و بسیار کارآمد به نام الگوریتم سینک‌هورن-نک^۲[۶۰] بهره می‌برد. این الگوریتم به جای یافتن مستقیم ماتریس Q ، دو بردار مقیاس‌دهی u (به طول K) و v (به طول B) را پیدا می‌کند که با استفاده از آن‌ها می‌توان ماتریس بهینه‌ی Q^* را ساخت. این الگوریتم به دلیل سرعت همگرایی بالا، برای پیاده‌سازی برخط و در هر دسته از داده‌ها بسیار مناسب است. فرایند الگوریتم به شرح زیر است:

۱. آماده‌سازی: ابتدا ماتریس شباهت نرم‌شده $M = \exp(C^T Z / \epsilon)$ محاسبه می‌شود. سپس، بردار v با مقادیر اولیه (معمولًا تمامًا یک) مقداردهی می‌شود.

۲. حلقه‌ی تکرار: الگوریتم برای تعداد مشخصی تکرار (در مقاله‌ی اصلی، تنها ۳ تکرار کافی دانسته شده است) دو مرحله‌ی زیر را به تناوب انجام می‌دهد تا قیود افراز برابر به تدریج ارضاء شوند:

¹Entropy Regularization

²Sinkhorn-Knopp

فصل سوم: روش پیشنهادی

- بهروزرسانی u (نرمال‌سازی سطرها): بردار u به گونه‌ای به روز می‌شود که قید مربوط به مراکز خوشه‌ها (مجموع سطرها باید برابر $K/1$ شود) ارضاء گردد.

$$u \leftarrow \frac{\frac{1}{K} \mathbf{1}_K}{Mv} \quad (7-3)$$

در این رابطه، تقسیم به صورت عنصر به عنصر انجام می‌شود.

- بهروزرسانی v (نرمال‌سازی ستون‌ها): سپس، با استفاده از u جدید، بردار v برای اراضی قید مربوط به ویژگی‌ها (مجموع ستون‌ها باید برابر $B/1$ شود) به روز می‌شود.

$$v \leftarrow \frac{\frac{1}{B} \mathbf{1}_B}{M^\top u} \quad (8-3)$$

۳. ساخت ماتریس نهایی: پس از پایان حلقه‌ی تکرار، بردارهای نهایی u و v به دست می‌آیند. ماتریس تخصیص بهینه‌ی Q^* از طریق فرمول زیر ساخته می‌شود:

$$Q^* = \text{Diag}(u) M \text{Diag}(v) \quad (9-3)$$

که در آن $\text{Diag}(u)$ و $\text{Diag}(v)$ ماتریس‌های قطری هستند که عناصر قطری آن‌ها به ترتیب از بردارهای u و v گرفته شده‌اند.

ماتریس Q^* به دست آمده تضمین می‌کند که هر دو قید افزای برابر ارضاء شده‌اند. هر ستون از این ماتریس، یک کد هدف q است که به صورت یک توزیع احتمال نرم، میزان تعلق یک ویژگی به هر یک از K خوشه را مشخص می‌کند. این کدها سپس به عنوان هدف در تابع هزینه‌ی آنتروپی متقطع به فرم معادله ۱۰-۳ به کار گرفته می‌شوند.

$$\ell(\mathbf{z}_t, \mathbf{q}_s) = - \sum_k \mathbf{q}_s^{(k)} \log \mathbf{p}_t^{(k)} \quad (10-3)$$

تابع هزینه نهایی

با مشخص شدن نحوه‌ی محاسبه‌ی تمام اجزا، اکنون می‌توان تابع هزینه نهایی روش SwAV را تعریف کرد. معادله ۱۱-۳ شکل بسط‌یافته‌ی تابع هزینه‌ی آنتروپی متقطعی است که در معادله ۱۰-۳ معرفی

شد و بر روی تمام داده‌های درون یک دسته و جفت‌های افزوده شده اعمال می‌شود:

$$L_{Batch} = -\frac{1}{N} \sum_{n=1}^N \sum_{s,t \sim \mathcal{T}} \left[\frac{1}{\tau} \mathbf{z}_{nt}^\top \mathbf{C} \mathbf{q}_{ns} + \frac{1}{\tau} \mathbf{z}_{ns}^\top \mathbf{C} \mathbf{q}_{nt} \right. \\ \left. - \log \sum_{k=1}^K \exp \left(\frac{\mathbf{z}_{nt}^\top \mathbf{c}_k}{\tau} \right) - \log \sum_{k=1}^K \exp \left(\frac{\mathbf{z}_{ns}^\top \mathbf{c}_k}{\tau} \right) \right] \quad (11-3)$$

در این معادله، بازنمایی‌های \mathbf{z} پس از عبور داده‌ی ورودی از شبکه‌ی f_θ به دست می‌آیند که f_θ خود شامل یک کدگذار و یک شبکه نگاشت (مانند روش SimCLR) می‌باشد. اعمال این تابع هزینه باعث می‌شود که پارامترهای اصلی شبکه یعنی f_θ و ماتریس مراکز خوشه‌ها یعنی C به صورت مشترک آموزش یابند تا تابع هزینه خودناظارتی بهینه شود و مدل به بازنمایی‌های مفید از روی داده‌ها دست یابد.

ماتریس C که ستون‌های آن بیانگر مراکز خوشه‌ها هستند نیز به سادگی و با یک لایه‌ی تماماً متصل خطی (بدون استفاده از بایاس) پیاده‌سازی می‌شود. در این لایه، وزن‌های وارد شده به هر نورون (که تعداد آن‌ها برابر با ابعاد فضای ویژگی \mathbf{z} می‌باشد) بیانگر جهت هر بردار مرکز خوشه در فضای ویژگی‌ها می‌باشد. بنابراین می‌توان گفت که هر نورون این لایه، بیانگر مرکز یکی از خوشه‌ها می‌باشد.

۲-۱-۳-۳ راهبرد برش چندگانه

یکی دیگر از نوآوری‌های مهم معرفی شده در مقاله‌ی SwAV، یک راهبرد جدید برای داده‌افزایی به نام راهبرد برش چندگانه^۱ است. پژوهش‌ها نشان داده‌اند که افزایش تعداد نماهای مختلف از یک تصویر در فرایند یادگیری تباینی، به یادگیری بازنمایی‌های بهتر و قوی‌تر منجر می‌شود. با این حال، استفاده از چندین نمای باوضوح^۲ استاندارد، هزینه‌های محاسباتی و حافظه را به صورت چشمگیری افزایش می‌دهد. راهبرد برش چندگانه راه حلی کارآمد برای این مشکل ارائه می‌دهد.

در این روش، به جای دو نمای از نماها باوضوح متفاوت استفاده می‌شود:

- **دو برش باوضوح استاندارد:** این دو برش، نماهای سراسری از داده را ارائه می‌دهند و اطلاعات کلی را در بر می‌گیرند.

- **چندین برش باوضوح پایین:** این برش‌ها که تعداد آن‌ها با V نمایش داده می‌شود، نماهای محلی و کوچکتری از تصویر را ثبت می‌کنند و بر روی جزئیات تمرکز دارند.

¹Multi-Crop Strategy

²Resolution

فصل سوم: روش پیشنهادی

ایده‌ی اصلی در این راهبرد، وادار کردن مدل به یادگیری ارتباط میان جزئیات محلی و ساختار کلی داده است. با پیش‌بینی تخصیص خوش‌های یک نمای سراسری (برش بزرگ) از روی ویژگی‌های یک نمای محلی (برش کوچک)، مدل می‌آموزد که یک جزء کوچک (مانند چشم) به یک کل بزرگ‌تر (مانند صورت) تعلق دارد. این فرایند به یادگیری ویژگی‌های معنایی بسیار غنی‌تری کمک می‌کند. برای پیاده‌سازی این رویکرد، تابع هزینه به شکل زیر تعمیم داده می‌شود:

$$L = \sum_{i \in 1,2} \sum_{v=1}^{V+2} \mathbf{1}_{v \neq i; \ell(z_v, q_i)} \quad (12-3)$$

یک نکته‌ی بسیار مهم در این فرایند آن است که کدهای هدف (q_1 و q_2) تنها و تنها از دو برش با وضوح استاندارد (نمایهای سراسری) محاسبه می‌شوند. دلیل این کار آن است که برش‌های کوچک به دلیل نمایش اطلاعات جزئی و ناقص از داده، ممکن است منجر به تولید کدهای هدف بی‌کیفیت و مبهم شوند که به فرایند یادگیری آسیب می‌رساند و تنها باعث افزایش شدید بار محاسباتی می‌گردد. در مقابل، بردارهای ویژگی z_v و توزیع‌های پیش‌بینی‌شده‌ی p_v برای تمام $V + 2$ نما (اعم از بزرگ و کوچک) محاسبه شده و همگی در پیش‌بینی دو کد هدف اصلی مشارکت می‌کنند. در نهایت، راهبرد برش چندگانه به SwAV اجازه می‌دهد تا از مزایای مقایسه‌های متعدد بهره‌مند شود، ارتباط میان مقیاس‌های مختلف داده را بیاموزد و همه‌ی این‌ها را با افزایش اندکی در هزینه‌ی محاسباتی به دست آورد.

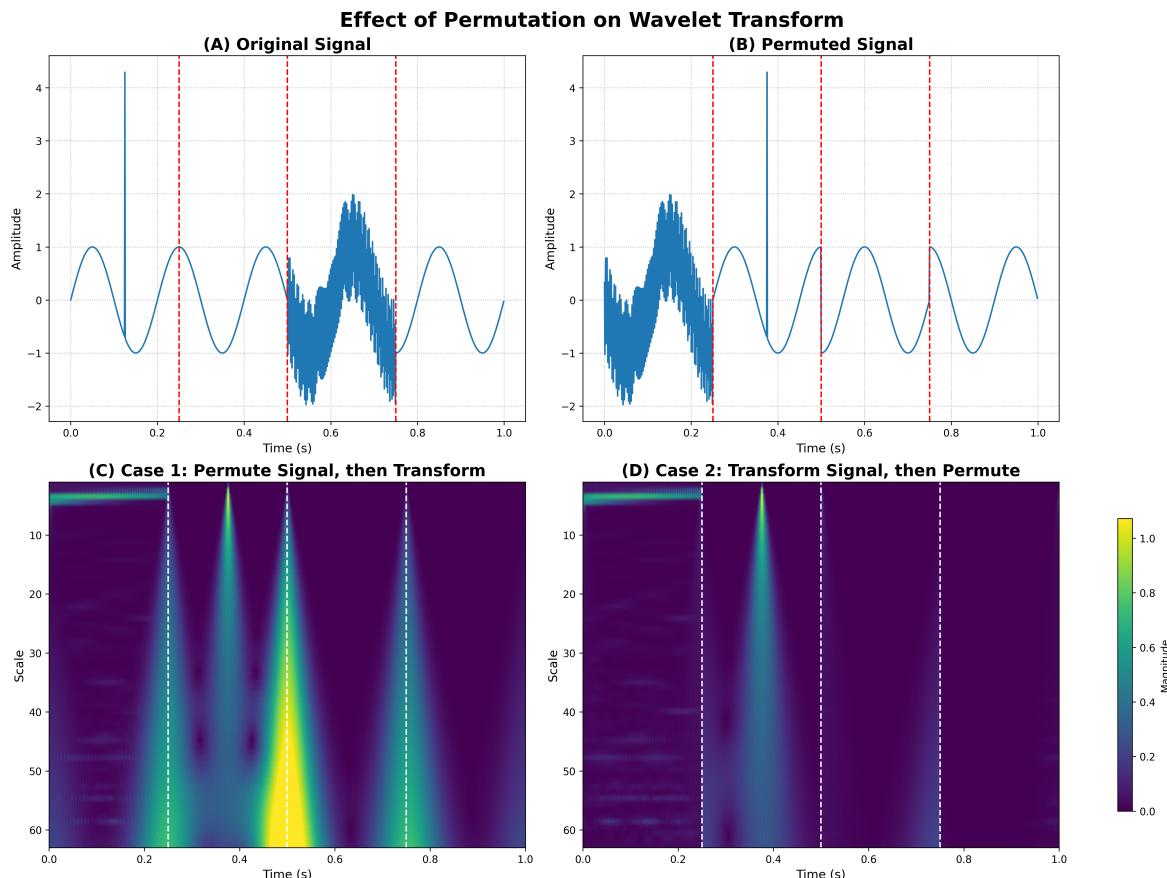
۲-۳-۳ راهبرد داده‌افزایی

راهبردهای داده‌افزایی به کاررفته برای سیگنال‌های خام، به‌خوبی توجیه شده و متناسب با تغییرات محتمل در دنیای واقعی هستند. با این حال، رویکرد اتخاذ شده برای اسکالولوگرام‌ها با چند چالش مواجه است. نخست آنکه مجموعه تبدیلات در نظر گرفته شده محدود است و مهم‌تر از آن، برخی از این تبدیلات، مانند اعوجاج رنگ تصادفی، فاقد معادل فیزیکی روش در کاربرد مورد نظر هستند. کانال‌های یک اسکالولوگرام، برخلاف تصاویر دیجیتال، ماهیت رنگ را ندارند و نمی‌توان با آن‌ها مانند کانال‌های رنگی رفتار کرد. برای مثال، عملیاتی مانند سیاه و سفید کردن تصویر که معادل میانگین‌گیری از کانال‌های رنگی است، در مورد اسکالولوگرام‌ها (مثلًا میانگین‌گیری از سه محور یک شتاب‌سنگ) به تحریفی منجر می‌شود که بازتاب‌دهنده‌ی هیچ پدیده‌ی فیزیکی محتملی نیست.

برای رفع این مشکل، می‌توان از همان تبدیلاتی که بر روی داده‌های خام سیگنال انجام داده‌ایم، بر

فصل سوم: روش پیشنهادی

روی اسکالوگرام‌ها نیز استفاده کنیم. با این حال، پیاده‌سازی این ایده با یک محدودیت عملی مهم روبرو است: تبدیل موجک یک عملیات محاسباتی زمان‌بر است. به همین دلیل، اعمال این تبدیل به صورت آنی بر روی هر پنجره از سیگنال در طول فرایند آموزش مدل، عملاً غیرممکن است. راه حل کارآمد، پیش‌پردازش کل مجموعه داده، محاسبه و ذخیره‌سازی اسکالوگرام‌ها، و سپس اعمال تبدیلات داده‌افزایی بر روی این اسکالوگرام‌های از پیش آماده شده است.



شکل ۴-۳: تاثیر ترتیب اعمال جایگشت و تبدیل موجک بر اسکالوگرام حاصل

این راهبرد یک پیش‌نیاز اساسی را به همراه دارد: تبدیلات منتخب باید به گونه‌ای باشند که ترتیب اعمال آن‌ها و تبدیل موجک، تفاوت چشمگیری در خروجی نهایی ایجاد نکند. به عبارت دیگر، نتیجه‌ی اعمال تبدیل داده‌افزایی بر روی اسکالوگرام باید تا حد امکان به نتیجه‌ی محاسبه‌ی اسکالوگرام از روی سیگنال داده‌افزایی شده نزدیک باشد. زیرا در دنیای واقعی، پدیده‌هایی مانند نویز یا چرخش حسگر، ابتدا بر سیگنال خام اثر می‌گذارند و سپس این سیگنال تغییریافته است که توسط ابزارهایی مانند تبدیل موجک تحلیل می‌شود.

خوبی‌ختانه، تبدیل موجک یک تبدیل خطی است. این ویژگی ریاضی به ما اجازه می‌دهد تا تبدیلات

خطی را با آن جابه‌جا کنیم بدون آنکه خروجی به شکل معناداری تغییر کند. به همین دلیل، استفاده از تبدیلات مقیاس‌دهی، معکوس‌سازی زمانی، بر زدن کانال‌ها و چرخش مشکلی ایجاد نمی‌کنند. اما نویز و جایگشت اسکالوگرام حاصل را به کلی تغییر می‌دهند. البته در رابطه با جایگشت، همانطور که در شکل ۴-۳ قابل مشاهده است، بخش‌های دارای فرکانس بالا (مقیاس پایین) که تغییرات محلی را در نظر می‌گیرند، تغییر چندانی ایجاد نمی‌شود اما در مقیاس‌های بالا که فرکانس‌های پایین و تغییرات سراسری را در نظر می‌گیرند، تغییرات شدیدی ایجاد می‌شود.

۴-۳ جمع‌بندی

در این فصل، روش پیشنهادی این پژوهش به‌طور جامع معرفی و تشریح گردید. ابتدا، معماری پایه که بر دو کدگذار مجزا برای حوزه‌های زمان و زمان-فرکانس استوار است، مورد بررسی قرار گرفت و نقاط ضعف و زمینه‌های مستعد بهبود در آن شناسایی شد. سپس، نوآوری‌های این پژوهش که برای رفع این چالش‌ها طراحی شده‌اند، ارائه گردید. نوآوری اصلی، جایگزینی چارچوب یادگیری تباینی SimCLR با الگوریتم SwAV بود که یک رویکرد مبتنی بر خوشه‌بندی است و با هدف یادگیری بازنمایی‌هایی پایدارتر و متمایزتر معرفی شد. علاوه بر این، راهبرد داده‌افزایی برای اسکالوگرام‌ها مورد بازنگری قرار گرفت و یک رویکرد جدید مبتنی بر تبدیلات معنادار فیزیکی که با ماهیت سیگنال‌ها سازگاری بیشتری دارد، جایگزین روش‌های پیشین شد. در فصل آتی، کارایی نوآوری‌های مطرح شده از طریق آزمایش‌های گسترشده مورد ارزیابی قرار گرفته و نتایج حاصل از آن به تفصیل بررسی خواهد شد.

فصل ۴

آزمایش‌ها و نتایج

فصل چهارم: آزمایش‌ها و نتایج

در این فصل ابتدا به معرفی مجموعه داده‌های استفاده شده در انجام آزمایش‌ها می‌پردازیم. سپس به بررسی آزمایش‌های انجام شده و نتایج به دست آمده از ارزیابی مدل پیشنهادی می‌پردازیم و نتایج حاصل را مورد بررسی قرار می‌دهیم.

به طور کلی، آزمایش‌های انجام شده در این پژوهش را می‌توان به دو دسته تقسیم کرد:

۱. پیشآموزش خودناظارتی بر روی مجموعه داده‌ی کوچک و تنظیم دقیق بر روی همان مجموعه.
۲. پیشآموزش خودناظارتی بر روی مجموعه داده‌ی بزرگ و تنظیم دقیق بر روی مجموعه داده‌ی کوچک.

۱-۴ مجموعه داده

برای ارزیابی عملکرد روش پیشنهادی در این پژوهش، از دو مجموعه داده^۱ HAPT^{۶۱} (مجموعه داده کوچک) و MobiAct^{۱۲} (مجموعه داده بزرگ) استفاده کردیم. در ادامه، جزئیات هر یک از این مجموعه داده‌ها به تفصیل تشریح می‌شود.

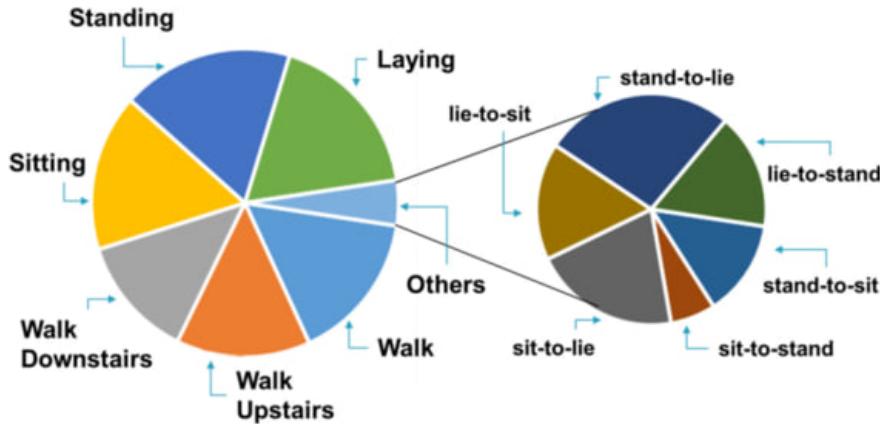
۱-۱-۴ مجموعه داده HAPT

مجموعه داده HAPT یکی از مجموعه داده‌های شناخته شده و پرکاربرد در حوزه شناسایی فعالیت انسان است که در دسترس عموم قرار دارد. این مجموعه داده نسخه توسعه یافته و کامل‌تری از مجموعه داده^۲ UCI-HAR^{۶۲} است و علاوه بر فعالیت‌های پایه، شامل گذارهای وضعیتی^۲ نیز می‌شود. هدف اصلی از ایجاد این مجموعه داده، فراهم کردن داده‌های خام و پردازش شده از حسگرهای اینرسی تعییه شده در گوشی‌های هوشمند برای ساخت و ارزیابی مدل‌های شناسایی فعالیت است. تمرکز ویژه این مجموعه داده بر تمایز قائل شدن بین فعالیت‌های ایستا و پویا و همچنین شناسایی حرکات کوتاه و انتقالی بین حالت‌های ایستا است.

داده‌های این مجموعه از ۳۰ داوطلب در بازه سنی ۱۹ تا ۴۸ سال جمع‌آوری شده است. هر شرکت‌کننده یک گوشی هوشمند Samsung Galaxy S II را بر روی کمر خود بسته بود و رویه مشخصی از فعالیت‌ها را انجام می‌داد. تمام آزمایش‌ها به صورت ویدیویی ضبط شدند تا برچسب‌گذاری داده‌ها با دقیق‌تری به صورت دستی انجام شود.

¹Smartphone-Based Recognition of Human Activities and Postural Transitions

²Postural Transitions



شکل ۱-۴: دسته‌های مختلف مجموعه داده HAPT

داده‌ها از دو حسگر اصلی گوشی هوشمند استخراج شده‌اند: حسگر شتاب‌سنج که سیگنال شتاب خطی سه‌محوره و حسگر ژیروسکوپ که سیگنال سرعت زاویه‌ای سه‌محوره را ثبت می‌کند. نرخ نمونه‌برداری برای هر دو حسگر 5° هرتز بوده است.

همانطور که در شکل ۱-۴ می‌توان دید، این مجموعه داده شامل ۱۲ کلاس فعالیت مجزا است که به دو دسته اصلی تقسیم می‌شوند:

- فعالیت‌های پایه که خود شامل سه فعالیت ایستادن، نشستن و دراز کشیدن و سه فعالیت پویای راه رفتن، بالا رفتن از پله و پایین آمدن از پله می‌باشد.
- گذارهای وضعیتی که شامل حرکات انتقالی بین فعالیت‌های ایستادن، نشستن به دراز کشیدن، ایستادن به نشستن، نشستن به ایستادن، دراز کشیدن به نشستن، ایستادن به دراز کشیدن و دراز کشیدن به ایستادن می‌باشد.

مجموعه داده HAPT هم به صورت داده‌های خام حسگر و هم به صورت ویژگی‌های استخراج شده ارائه می‌گردد. داده‌های خام شامل سیگنال‌های سه‌محوره شتاب‌سنج و ژیروسکوپ به صورت سری زمانی است که نتیجتاً شامل ۶ ویژگی می‌باشد. ویژگی‌های استخراج شده نیز بدین صورت می‌باشند که ابتدا سیگنال‌های خام با استفاده از یک پنجره لغزان به طول ۱۲۸ (۲.۵۶ ثانیه) و 5° درصد همپوشانی قطعه‌بندی شده‌اند. از هر قطعه، یک بردار ویژگی ۵۶۱ بعدی استخراج شده است. این ویژگی‌ها شامل محاسبات آماری در حوزه زمان و فرکانس مانند میانگین، انحراف معیار، تبدیل فوریه سریع و غیره هستند. در این پژوهش از سیگنال‌های خام حسگرها برای آموزش مدل استفاده کردیم.

۲-۱-۴ مجموعه داده MobiAct

مجموعه داده MobiAct یک مجموعه داده عمومی برای شناسایی فعالیت انسان است که با استفاده از حسگرهای گوشی هوشمند ایجاد شده و به طور خاص بر تشخیص فعالیتهای روزمره و انواع سقوط^۱ تمرکز دارد. این مجموعه داده شامل داده‌های ثبت شده از سه حسگر اصلی یک گوشی هوشمند Samsung Galaxy S III یعنی شتاب‌سنج، ژیروسکوپ و حسگر جهت‌یاب^۲ می‌باشد.

داده‌ها از ۵۷ داوطلب (۴۲ مرد و ۱۵ زن) با میانگین سنی ۲۶ سال جمع‌آوری شده است. از این تعداد، ۵۰ شرکت‌کننده تمام سناریوهای مربوط به فعالیت‌های روزمره و ۵۴ شرکت‌کننده تمام سناریوهای سقوط را با موفقیت به پایان رساندند. برای شبیه‌سازی هرچه بهتر شرایط واقعی، از هر شرکت‌کننده خواسته شد تا گوشی هوشمند را به صورت آزاد و با جهت‌گیری دلخواه در جیب شلوار خود قرار دهد.

فعالیت‌های ثبت شده در این مجموعه داده به دو گروه اصلی تقسیم می‌شوند:

- **نه نوع فعالیت روزمره:** این فعالیت‌ها شامل ایستادن، راه رفتن، دویدن، پریدن، بالا رفتن از پله، پایین آمدن از پله، نشستن روی صندلی، وارد شدن به ماشین و خارج شدن از ماشین است.
- **چهار نوع سقوط شبیه‌سازی شده:** این سقوط‌ها شامل سقوط به جلو، سقوط به جلو روی زانو، سقوط به پهلو و سقوط به عقب در حین تلاش برای نشستن روی صندلی می‌باشند.

۲-۴ جزئیات پیاده‌سازی

در این بخش به بررسی جزئیات مختلف پیاده‌سازی روش پیشنهادی (شامل پیش‌پردازش داده‌ها، آموزش مدل و معیارهای ارزیابی) می‌پردازیم.

۱-۲-۴ پیش‌پردازش داده‌ها

پیش‌پردازش داده‌ها یک مرحله مهم در آموزش مدل‌های شناسایی فعالیت انسان است و عملکرد چشم‌گیری در بهبود نتایج به همراه دارد. برای این امر، ابتدا بایستی که داده‌ها را نرمال‌سازی کنیم. برای نرمال‌سازی داده‌ها از روش استانداردسازی که به آن نرمال‌سازی Z-score نیز می‌گویند استفاده می‌کنیم.

¹Falls

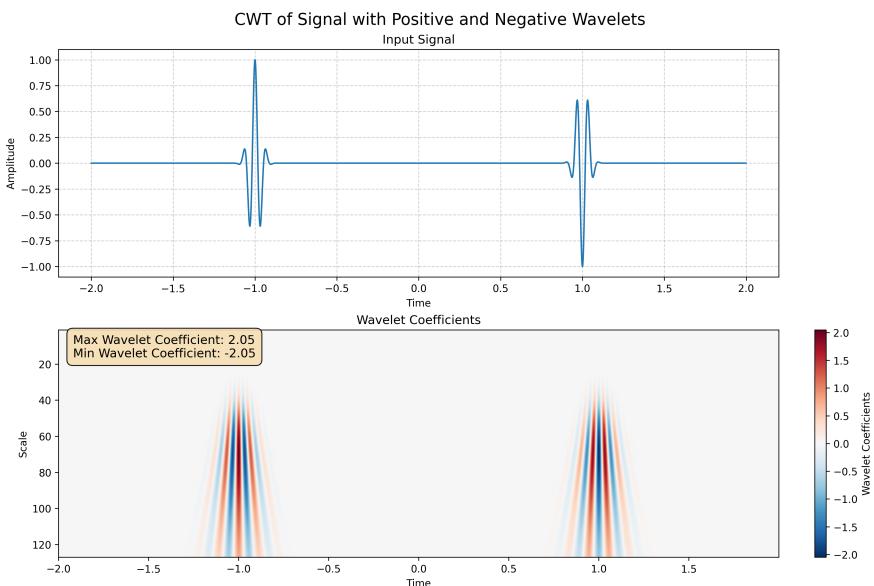
²Orientation Sensor

فرمول آن به فرم زیر است:

$$z = \frac{x - \mu}{\sigma} \quad (1-4)$$

در این رابطه x بیانگر مقدار هر داده، μ بیانگر میانگین توزیع داده‌ها و σ انحراف معیار توزیع داده‌ها می‌باشند. با استفاده از استانداردسازی، میانگین توزیع داده‌ها برابر با صفر و انحراف معیار آن‌ها برابر با یک می‌شود.

نکته‌ی مهم در رابطه با نرمال‌سازی داده‌ها این است که اگر ابتدا داده‌ی خام سیگنال‌ها را استانداردسازی کنیم و سپس تبدیل موجک را اعمال کنیم، تبدیل موجک حاصل همانطور که در شکل ۲-۴ می‌توان دید، در جاهایی که موجک استفاده شده با داده‌ی سری زمانی ورودی همبستگی بالا داشته باشد، بازه‌ی اسکالوگرام خروجی بزرگ می‌گردد و می‌تواند تا چند برابر بیشتر از داده‌ی خام سیگنال شود. به همین منظور، داده‌ی خام سیگنال‌ها استانداردسازی می‌شوند و داده‌ی اسکالوگرام‌ها به صورت مستقل از یکدیگر استانداردسازی می‌شوند. پس از استانداردسازی داده‌ها می‌توانیم از آن‌ها برای پیش‌آموزش و آموزش مدل استفاده کنیم.



شکل ۲-۴: تاثیر تبدیل موجک بر برد اسکالوگرام خروجی

پس از نرمال‌سازی داده‌ها، به سراغ پر کردن مقادیر از دست رفته‌ی سیگنال‌ها با استفاده از درون‌یابی^۱ می‌رویم. هر چند که هیچ یک از دو مجموعه داده‌ی استفاده شده در این پژوهش مقادیر از دست رفته

¹Interpolation

ندارند، اما در هر حال می‌توان با درونیابی به آن‌ها رسیدگی کرد. در قدم بعدی، بایستی نرخ نمونه‌برداری مجموعه داده‌ها را کنترل کنیم. در حالتی که هم پیش‌آموزش خودناظارتی و هم تنظیم دقیق بر روی یک مجموعه داده انجام می‌شود، نیازی به این موضوع نیست و نرخ نمونه‌برداری را همان ۵۰ هرتز برای مجموعه داده HAPT نگاه می‌داریم. اما هنگامی که پیش‌آموزش بر روی مجموعه داده MobiAct انجام می‌شود و تنظیم دقیق بر روی HAPT، بایستی که هر دو مجموعه داده دارای نرخ نمونه‌برداری برابر باشند. چرا که برای مثال اگر از یک پنجره به طول ۱۰۰ استفاده کنیم، در حالتی که نرخ نمونه‌برداری ۵۰ هرتز است، به فعالیت‌های انجام شده در یک بازه‌ی ۲ ثانیه‌ای نگاه می‌کنیم. اما هنگامی که نرخ نمونه‌برداری ۲۰ هرتز باشد (نرخ نمونه‌برداری مجموعه داده MobiAct)، به فعالیت‌های انجام شده در یک بازه‌ی ۵ ثانیه‌ای نگاه می‌کنیم. که این امر باعث افت عملکرد مدل می‌شود. به همین دلیل باید در حالتی که یادگیری انتقالی انجام می‌دهیم، نرخ نمونه‌برداری مجموعه داده HAPT را نیز به ۲۰ هرتز برسانیم تا با مجموعه داده MobiAct هماهنگ شود.

سپس داده‌ها را به پنجره‌های لغزان به طول ۱۲۸ و دارای همپوشانی تقسیم می‌کنیم. در رابطه با مجموعه داده HAPT به علت کم بودن تعداد داده‌ها میزان همپوشانی را برابر با ۹۰ درصد قرار دادیم. اما برای مجموعه داده MobiAct میزان همپوشانی را برابر با ۵۰ درصد قرار دادیم. در تنظیم دقیق نظارت‌شده نیز برچسب هر پنجره برابر با برچسبی است که بیشترین تعداد تکرار را دارد.

۲-۲-۴ آموزش مدل

آموزش مدل شامل دو بخش پیش‌آموزش خودناظارتی و تنظیم دقیق نظارت‌شده می‌باشد. در بخش پیش‌آموزش مدل، از تمام داده‌های موجود در مجموعه داده استفاده می‌کنیم. چرا که از برچسب داده‌ها استفاده‌ای نکرده‌ایم و صرفا هدف یادگیری توزیع داده‌ها و تولید بازنمایی مفید است. بنابراین نشت مدل^۱ رخ نمی‌دهد.

در آموزش نظارت‌شده که از مجموعه داده HAPT استفاده کرده‌ایم، مدل پیش‌آموزش دیده را با درصدی مختلفی از داده‌ی آموزش و ارزیابی آموزش دادیم. این درصدها شامل ۸۰ درصد آموزش، ۶۰ درصد آموزش، ۴۰ درصد آموزش و ۲۰ درصد آموزش می‌باشند. در واقع هدف این است که قدرت تعمیم مدل و آموزش با میزان پایین داده‌ی آموزشی را ارزیابی کنیم. روند انجام آزمایشات بدین صورت است که ابتدا کدگذارهای سیگنال و اسکالوگرام را با استفاده از

^۱Model leakage

فصل چهارم: آزمایش‌ها و نتایج

روشی که در فصل قبل ارائه دادیم را پیش‌آموزش می‌دهیم و وزن‌های کدگذارها را ذخیره می‌کنیم. سپس در مرحله‌ی بعد، دسته‌بندها را بر روی ویژگی‌های استخراج شده از کدگذارها با استفاده از درصدهای مختلف داده آموزشی آموزش می‌دهیم. به ازا هر درصد، ۵ بار آزمایش را به‌این صورت تکرار می‌کنیم:

۱. مجموعه داده را به ۵ بخش مساوی تقسیم می‌کنیم.
۲. بسته به درصد داده‌ی آموزشی و ارزیابی، تعدادی از این بخش‌ها آموزشی و تعدادی از آن‌ها ارزیابی می‌باشند.
۳. ۵ بار آزمایش را تکرار می‌کنیم و هر بار داده‌های آموزشی و ارزیابی متفاوت می‌باشند.
۴. ارزیابی نهایی عملکرد مدل برابر با میانگین ۵ بار اجرا مربوطه می‌باشد.

۳-۲-۴ معیارهای ارزیابی

ارزیابی عملکرد مدل‌های یکی از مهم‌ترین بخش‌های یادگیری ماشین می‌باشد. در این پژوهش از دو معیار ارزیابی امتیاز F1^۱ و امتیاز کاپا^۲ (که به آن کاپای کوهن^۳ نیز می‌گویند) استفاده کرده‌ایم. اما پیش از بررسی این دو معیار ارزیابی، بایستی که ۳ معیار ارزیابی دیگر شامل صحت^۴، دقت^۵ و فراخوانی^۶ را معرفی کنیم.

معیار صحت به عنوان یکی از معیارهای پرکاربرد برای ارزیابی یک مدل دسته‌بندی در مسائل مختلف یادگیری ماشین مورد استفاده قرار می‌گیرد. این معیار نسبت تعداد داده‌هایی را که به درستی توسط مدل دسته‌بندی شده‌اند به تعداد کل داده‌های موجود در داده‌های آزمون می‌سنجد. فرمول محاسبه صحت به فرم معادله ۲-۴ می‌باشد:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \quad (2-4)$$

¹F1 Score

²Kappa Score

³Cohen's Kappa

⁴Accuracy

⁵Precision

⁶Recall

فصل چهارم: آزمایش‌ها و نتایج

در این معادله، TP^1 به تعداد نمونه‌های مثبت واقعی که به درستی مثبت تشخیص داده شده‌اند، TN^2 به تعداد نمونه‌های منفی واقعی که به درستی منفی تشخیص داده شده‌اند، FP^3 به تعداد نمونه‌های منفی واقعی که به اشتباه مثبت تشخیص داده شده‌اند و FN^4 به تعداد نمونه‌های مثبت واقعی که به اشتباه به عنوان منفی تشخیص داده شده‌اند اشاره دارد.

معیار دقت بیانگر نسبت نمونه‌های مثبت واقعی که به درستی تشخیص داده شده‌اند به تعداد نمونه‌هایی که مدل مثبت تشخیص داده است می‌باشد:

$$\text{Precision} = \frac{TP}{TP + FP} \quad (3-4)$$

معیار فراخوانی نیز بیانگر نسبت نمونه‌های مثبت واقعی که به درستی تشخیص داده شده‌اند به تعداد کل نمونه‌های مثبت واقعی است:

$$\text{Recall} = \frac{TP}{TP + FN} \quad (4-4)$$

حال به بررسی معیارهای ارزیابی مورد استفاده در این پژوهش می‌پردازیم. معیار امتیاز F1 یک معیار جامع برای ارزیابی مدل‌های دسته‌بندی است که به صورت ترکیبی از دقت و فراخوانی محاسبه می‌شود. این معیار بیشتر منعکس‌کننده‌ی توازن میان دقت و فراخوانی مدل است. فرمول محاسبه امتیاز F1 به فرم معادله ۵-۴ می‌باشد:

$$\text{F1Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (5-4)$$

معیار ارزیابی دیگری که در این پژوهش مورد استفاده قرار گرفته است، امتیاز کاپا می‌باشد. این معیار، میزان توافق بین پیش‌بینی‌های انجام‌شده توسط مدل و برچسب‌های واقعی را با در نظر گرفتن احتمال توافق تصادفی، ارزیابی می‌کند. مزیت اصلی امتیاز کاپا این است که نشان می‌دهد عملکرد مدل تا چه حد از یک حدس کاملاً تصادفی بهتر است. این ویژگی، کاپا را به معیاری قابل اطمینان‌تر، بهویژه در هنگام مواجهه با مجموعه داده‌های نامتوازن تبدیل می‌کند. فرمول محاسبه امتیاز کاپا به فرم معادله ۶-۴

¹True Positive

²True Negative

³False Positive

⁴False Negative

می‌باشد:

$$\text{Kappa} = \frac{\text{Accuracy} - \text{Accuracy}_r}{1 - \text{Accuracy}_r} \quad (6-4)$$

در این معادله، Accuracy همان صحت مدل است و Accuracy_r نشان‌دهنده‌ی صحت توافق تصادفی است. این مقدار، نمایانگر صحت عملکرد یک مدل فرضی است که تعداد پیش‌بینی‌هاییش برای هر دسته، دقیقاً با تعداد پیش‌بینی‌های مدل اصلی ما یکسان است، اما این تخصیص برچسب‌ها را کاملاً به صورت تصادفی انجام می‌دهد. به عبارت دیگر، ما عملکرد مدل خود را با یک پیش‌بینی‌کننده‌ی تصادفی که از توزیع داده‌ها آگاه است، مقایسه می‌کنیم تا ببینیم یادگیری مدل چقدر فراتر از شанс بوده است. برای یک مسئله‌ی دسته‌بندی چندکلاسه با k دسته، فرمول کلی محاسبه‌ی Accuracy_r به صورت معادله

۷-۴ می‌باشد:

$$\text{Accuracy}_r = \frac{1}{N^2} \sum_{i=1}^k (A_i \times P_i) \quad (7-4)$$

در این معادله k تعداد دسته‌ها، N تعداد کل نمونه‌ها، A_i تعداد کل نمونه‌های واقعی متعلق به دسته‌ی i و P_i تعداد کل نمونه‌هایی است که توسط مدل به عنوان دسته‌ی i پیش‌بینی شده‌اند. مقدار امتیاز کاپا بین -1 و $+1$ قرار دارد. مقادیر مثبت به معنای عملکرد بهتر مدل از یک دسته‌بند تصادفی، مقدار صفر به معنای عملکرد کاملاً تصادفی مدل و مقادیر منفی به معنای عملکرد بدتر مدل از یک دسته‌بند تصادفی می‌باشد.

۴-۲-۴ ابرپارامترها

جزئیات پیکربندی مدل پیشنهادی در جدول ۱-۴ قابل مشاهده می‌باشد. مقادیر انتخابی بر اساس روش پایه [۵۸] انتخاب شده‌اند و جستجوی گسترده‌ای برای یافتن پیکربندی بهینه انجام نشده است. تعداد خوشه‌ها برای روش SwAV نیز به صورت تجربی باستی حدود 10° برابر تعداد دسته‌های مجموعه داده هدف باشند [۲۴]. بنابراین تعداد خوشه‌ها را برابر با ۱۲۸ قرار دادیم. نرخ یادگیری در بخش پیش‌آموزش خودنظرارتی بدین صورت می‌باشد:

۱. مقدار اولیه‌ی نرخ یادگیری را برابر با 0.001 قرار می‌دهیم.

فصل چهارم: آزمایش‌ها و نتایج

جدول ۴-۱: پیکربندی یادگیرنده‌ی سیگنال (سمت راست) و یادگیرنده‌ی اسکالولوگرام (سمت چپ)

جزئیات لایه	واحد	جزئیات لایه	واحد
پیچشی با هسته 8×8 کاناله	۳۲	پیچشی با هسته 1×12 کاناله	۳۲
پیچشی با هسته 4×4 کاناله	۶۴	پیچشی با هسته 8×1 کاناله	۶۴
رمزگذار		رمزگذار	
پیچشی با هسته 4×4 کاناله	۹۶	پیچشی با هسته 8×4 کاناله	۹۶
ادغام بیشینه سراسری		ادغام بیشینه سراسری	
تماما متصل 96 بعدی	شبکه نگاشت	تماما متصل 96 بعدی	شبکه نگاشت
تماما متصل 96 بعدی		تماما متصل 96 بعدی	
تماما متصل 256 بعدی	شبکه دسته‌بند	تماما متصل 256 بعدی	شبکه دسته‌بند
تماما متصل 12 بعدی		تماما متصل 12 بعدی	

۲. به صورت خطی طی 10 دوره^۱ اول آن را تا 0.05 افزایش می‌دهیم. مقادیر بزرگ‌تر موجب ناپایداری یادگیری می‌شوند.

۳. به صورت کسینوسی و تا آخرین دوره آموزش (100 دوره) مقدار نرخ یادگیری را تا 0.0001 کاهش می‌دهیم.

برای تنظیم دقیق مدل، از یک نرخ یادگیری ثابت برابر با 0.001 استفاده کردیم. تعداد تکرار الگوریتم سینکهورن (که در بخش ۱-۱-۳-۳ معرفی کردیم) برابر با 3 می‌باشد. مقدار ابرپارامتر ϵ (معادله ۶-۳) را برابر با 0.01 قرار دادیم. افزایش مقدار این پارامتر باعث می‌شود که تخصیص‌های روش سینکهورن برای تمامی داده‌ها به تمامی خوشها یکنواخت شود که در این حالت مدل بازنمایی‌های خوبی یاد نمی‌گیرد. پارامتر دما (τ در معادله ۵-۳) نیز می‌تواند تاثیری مشابه داشته باشد. زیاد کردن مقدار آن باعث می‌شود که توزیع تخصیص‌ها توسط مدل (نه الگوریتم سینکهورن) یکنواخت شود. اما بیش از حد کم کردن مقدار آن نیز شدیداً باعث ناپایداری آموزش خواهد شد. همانطور که در معادله ۵-۳ دیده می‌شود، مقدار موجود در نمای \exp تقسیم بر ابرپارامتر دما می‌شود. کوچک بودن بیش از حد این ابرپارامتر باعث می‌شود که مقدار این توان از لحاظ عددی بسیار بزرگ شود که به تبع آن کامپیوتر نمی‌تواند مقدار آن را محاسبه کند و فرایند آموزش دچار خطا می‌شود. بنابراین مقدار آن را برابر با مقدار 0.05 قرار دادیم که هم از یکنواخت شدن توزیع سینکهورن جلوگیری می‌کند و هم باعث ناپایداری آموزش نمی‌شود.

اندازه‌ی دسته برای داده‌های خام سیگنال را برابر با 1024 و برای اسکالولوگرام نیز برابر با 512 قرار

¹Epoch

دادیم. تعداد دوره‌ی آموزش در بخش پیش‌آموزش خودناظارتی را برای کدگذار سیگنال برابر با ۱۰۰ و برای کدگذار اسکالولوگرام برابر با ۷۵ قرار دادیم. اما برای تنظیم دقیق هر دو مدل، از روش توقف زودهنگام^۱ استفاده می‌کنیم. بدین صورت که بر روی داده‌های اعتبارسنجی، مقدار هزینه مدل را بررسی می‌کنیم و هنگامی که مدل دچار بیش‌برازش شد و هزینه‌ی اعتبارسنجی بالا رفت، آموزش را متوقف می‌کنیم و بهترین مدل را انتخاب می‌کنیم.

۳-۴ نتایج

در این بخش، به ارائه و تحلیل نتایج حاصل از آزمایش‌های انجام‌شده بر روی مدل پیشنهادی پرداخته می‌شود. همانطور که پیش‌تر تشریح شد، ارزیابی عملکرد مدل در دو سناریوی اصلی صورت گرفته است. ابتدا، نتایج حاصل از پیش‌آموزش خودناظارتی و تنظیم دقیق بر روی مجموعه داده واحد HAPT در قالب آزمایش‌های درون‌مجموعه‌ای^۲ بررسی می‌گردد. سپس، کارایی روش در سناریوی یادگیری انتقالی، که در آن پیش‌آموزش بر روی مجموعه داده MobiAct و تنظیم دقیق بر روی مجموعه داده HAPT انجام شده، تحت عنوان آزمایش‌های بین‌مجموعه‌ای^۳ ارائه می‌گردد. عملکرد مدل در تمامی آزمایش‌ها با استفاده از معیارهای امتیاز F1 و امتیاز کاپا سنجیده شده و نتایج به تفکیک مورد بحث قرار خواهند گرفت.

۱-۳-۴ نتایج آزمایش‌های درون‌مجموعه‌ای

در این بخش، به بررسی نتایج حاصل از سناریوی اول، یعنی پیش‌آموزش خودناظارتی و تنظیم دقیق مدل بر روی مجموعه داده HAPT می‌پردازیم. هدف از این آزمایش، ارزیابی عملکرد مدل در مقایسه با مدل پایه و یادگیری نظارت‌شده و ایجاد یک خط معیار برای مقایسه با نتایج یادگیری انتقالی است که در بخش بعد ارائه خواهد شد.

نتایج دقیق این آزمایش‌ها در جدول ۲-۴ ارائه شده است. این جدول، میانگین و انحراف معیار امتیاز F1 و امتیاز کاپا را به ازای استفاده از ۲۰، ۴۰، ۶۰ و ۸۰ درصد از داده‌های آموزشی برچسب‌دار، پس از ۵ بار تکرار آزمایش، نمایش می‌دهد.

¹Early Stopping

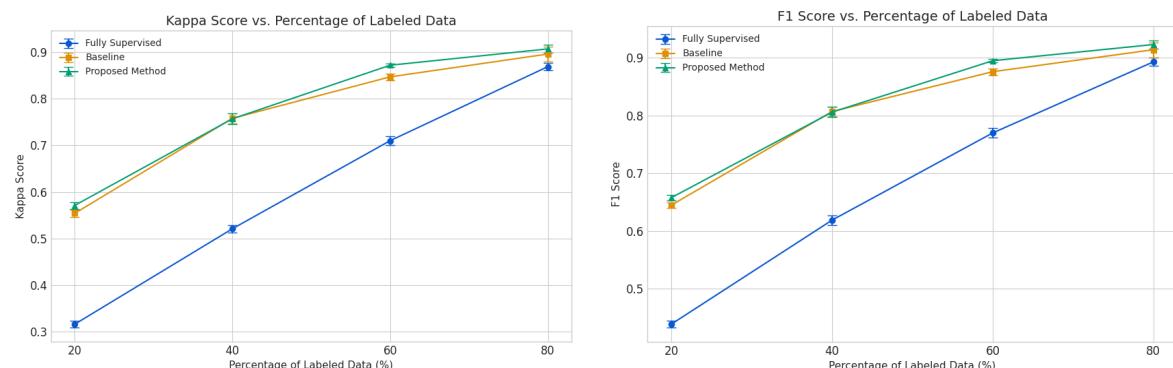
²Intra-Dataset

³Inter-Dataset

فصل چهارم: آزمایش‌ها و نتایج

جدول ۴-۲: نتایج عملکرد مدل در سناریویی درون‌مجموعه‌ای بر اساس درصدهای مختلف داده آموزشی

درصد داده	روش	F1	امتیاز کاپا
۲۰٪	کاملاً نظارت شده	۰.۴۳۹ ± ۰.۰۰۶	۰.۳۱۶ ± ۰.۰۰۷
	روش پایه [۵۸]	۰.۶۴۵ ± ۰.۰۰۵	۰.۵۵۴ ± ۰.۰۰۸
	روش پیشنهادی	۰.۶۵۸ ± ۰.۰۰۵	۰.۵۷۱ ± ۰.۰۰۷
۴۰٪	کاملاً نظارت شده	۰.۶۱۹ ± ۰.۰۰۸	۰.۵۲۱ ± ۰.۰۰۸
	روش پایه [۵۸]	۰.۸۰۷ ± ۰.۰۰۸	۰.۷۵۸ ± ۰.۰۱۱
	روش پیشنهادی	۰.۸۰۶ ± ۰.۰۰۹	۰.۷۵۷ ± ۰.۰۱۱
۶۰٪	کاملاً نظارت شده	۰.۷۷۰ ± ۰.۰۰۸	۰.۷۱۰ ± ۰.۰۱۰
	روش پایه [۵۸]	۰.۸۷۶ ± ۰.۰۰۶	۰.۸۴۷ ± ۰.۰۰۷
	روش پیشنهادی	۰.۸۹۵ ± ۰.۰۰۳	۰.۸۷۲ ± ۰.۰۰۴
۸۰٪	کاملاً نظارت شده	۰.۸۹۳ ± ۰.۰۰۷	۰.۸۶۹ ± ۰.۰۰۸
	روش پایه [۵۸]	۰.۹۱۴ ± ۰.۰۱۳	۰.۸۹۶ ± ۰.۰۱۶
	روش پیشنهادی	۰.۹۲۳ ± ۰.۰۰۷	۰.۹۰۷ ± ۰.۰۰۹



شکل ۴-۳: مقایسه امتیاز F1 (راست) و امتیاز کاپا (چپ) برای روش‌های مختلف بر اساس درصدهای متفاوت داده‌های آموزشی.

اولین و واضح‌ترین نتیجه‌گیری که از روی جدول ۴-۲ و شکل ۴-۳ می‌توان استنباط کرد، عملکرد به مراتب بهتر هر دو روش خودناظارتی (روش پایه و روش پیشنهادی) در مقایسه با روش کاملاً نظارت شده است؛ بهویژه در شرایط کمبود داده‌های برچسب‌دار. برای مثال، در حالت استفاده از تنها ۲۰٪ داده‌ها، روش پیشنهادی بهبودی در حدود ۲۲٪ در امتیاز F1 نسبت به مدل کاملاً نظارت شده نشان می‌دهد. این شکاف عمیق، ارزش مرحله پیش‌آموزش را به اثبات می‌رساند؛ جایی که مدل با استفاده از کل داده‌های برچسب‌نخورده، بازنمایی‌های غنی و مفیدی را می‌آموزد و با یک دانش اولیه قدرتمند وارد مرحله تنظیم دقیق می‌شود.

با تمرکز بر مقایسه دو روش خودناظارتی، مشاهده می‌شود که روش پیشنهادی در اکثر سناریوها ۲۰٪، ۴۰٪ و ۶۰٪ داده عملکرد بهتری از خود نشان داده است.

- در شرایط داده بسیار کم (۲۰٪): روش پیشنهادی با اختلاف کمی، بهترین عملکرد را ثبت کرده است. این موضوع اهمیت ویژه‌ای دارد، زیرا نشان می‌دهد مدل پیشنهادی در چالش‌برانگیزترین حالت (کمبود داده) کارآمدتر است. این موضوع نشان می‌دهد که مدل پیشنهادی قابلیت تعمیم‌پذیری بیشتری می‌تواند از خود نشان بدهد.
- در شرایط داده متوسط (۴۰٪): عملکرد دو روش بسیار نزدیک و رقابتی است و روش پایه برتری جزئی و از نظر آماری نامحسوسی دارد. این نشان می‌دهد که هر دو روش در این سطح از داده به پایداری عملکردی خوبی رسیده‌اند.
- در شرایط داده زیاد (۶۰٪ و ۸۰٪): با افزایش حجم داده‌های برچسبدار، برتری روش پیشنهادی مجددأ خود را نشان می‌دهد و به ویژه در سطح ۶۰٪، شکاف قابل توجهی ایجاد می‌کند. این امر نشان‌دهنده ظرفیت بالاتر مدل پیشنهادی برای بهره‌برداری از داده‌های بیشتر و رسیدن به دقت بالاتر است.

در مجموع، نتایج این بخش به وضوح نشان می‌دهد که روش پیشنهادی نه تنها بر محدودیت‌های یادگیری کاملا نظارت شده در شرایط کمبود داده فائق می‌آید، بلکه در مقایسه مستقیم با روش پایه‌ی اخیر نیز در اکثر موارد عملکردی بهتر ارائه می‌دهد. این نتایج، پتانسیل بالای مدل را در سناریوی درون-مجموعه‌ای تأیید می‌کند. در بخش بعد، این ارزیابی را یک گام فراتر برده و عملکرد مدل را در سناریوی چالش‌برانگیزتر یادگیری انتقالی خواهیم سنجید.

۲-۳-۴ نتایج آزمایش‌های بین‌مجموعه‌ای

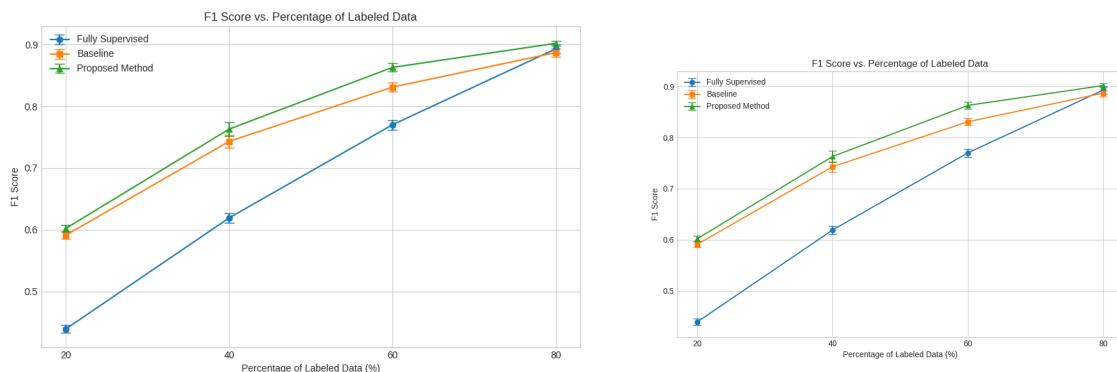
در این بخش، به بررسی نتایج حاصل از سناریوی دوم، یعنی پیش‌آموزش خودناظارتی بر روی مجموعه داده MobiAct و تنظیم دقیق مدل بر روی مجموعه داده‌ی HAPT می‌پردازیم. نتایج دقیق این آزمایش‌ها در جدول ۳-۴ ارائه شده است. این جدول، میانگین و انحراف معیار امتیاز F1 و امتیاز کاپا را به ازای استفاده از ۲۰، ۴۰، ۶۰ و ۸۰ درصد از داده‌های آموزشی برچسبدار، پس از ۵ بار تکرار آزمایش، نمایش می‌دهد

تحلیل نتایج ارائه شده در جدول ۳-۴، به ویژه هنگامی که در کنار نتایج آزمایش‌های درون-مجموعه‌ای (جدول ۲-۴) قرار می‌گیرد، یافته‌های بسیار مهمی را در مورد قدرت تعمیم و پایداری مدل‌ها آشکار می‌سازد.

فصل چهارم: آزمایش‌ها و نتایج

جدول ۴-۳: نتایج عملکرد مدل در سناریوی بین‌مجموعه‌ای بر اساس درصدهای مختلف داده آموزشی

درصد داده	روش	امتیاز کاپا	F1
۲۰٪	کاملا نظارت شده	۰.۳۱۶ ± ۰.۰۰۷	۰.۴۳۹ ± ۰.۰۰۶
	روش پایه [۵۸]	۰.۴۸۹ ± ۰.۰۰۷	۰.۵۹۱ ± ۰.۰۰۶
	روش پیشنهادی	۰.۵۶۰ ± ۰.۰۰۸	۰.۶۰۲ ± ۰.۰۰۶
۴۰٪	کاملا نظارت شده	۰.۵۲۱ ± ۰.۰۰۸	۰.۶۱۹ ± ۰.۰۰۸
	روش پایه [۵۸]	۰.۶۷۶ ± ۰.۰۱۱	۰.۷۴۳ ± ۰.۰۱۰
	روش پیشنهادی	۰.۷۰۱ ± ۰.۰۱۵	۰.۷۶۳ ± ۰.۰۱۱
۶۰٪	کاملا نظارت شده	۰.۷۱۰ ± ۰.۰۱۰	۰.۷۷۰ ± ۰.۰۰۸
	روش پایه [۵۸]	۰.۷۸۹ ± ۰.۰۰۹	۰.۸۳۱ ± ۰.۰۰۷
	روش پیشنهادی	۰.۸۳۰ ± ۰.۰۰۹	۰.۸۶۳ ± ۰.۰۰۷
۸۰٪	کاملا نظارت شده	۰.۸۶۹ ± ۰.۰۰۸	۰.۸۹۳ ± ۰.۰۰۷
	روش پایه [۵۸]	۰.۸۶۳ ± ۰.۰۰۸	۰.۸۸۷ ± ۰.۰۰۷
	روش پیشنهادی	۰.۸۸۶ ± ۰.۰۰۶	۰.۹۰۲ ± ۰.۰۰۴



شکل ۴-۴: مقایسه امتیاز F1 (راست) و امتیاز کاپا (چپ) برای روش‌های مختلف بر اساس درصدهای متفاوت داده‌های آموزشی.

اولین مشاهده در تحلیل این جدول، برتری کامل و مداوم روش پیشنهادی بر هر دو روش دیگر در تمامی سطوح داده‌ی برچسبدار است. برخلاف سناریوی قبلی که در سطح ۴۰٪ داده، رقابت بسیار نزدیک بود، در این سناریوی بین-مجموعه‌ای، مدل پیشنهادی در تمام آزمون‌ها با اختلاف معناداری بهترین عملکرد را به ثبت رسانده است. این موضوع نشان می‌دهد که بازنمایی‌های آموخته شده توسط روش پیشنهادی از مجموعه داده MobiAct، برای انتقال به دامنه جدید (مجموعه داده HAPT) کارآمدتر و قابل استفاده‌تر بوده‌اند.

نکته کلیدی و بسیار قابل تأمل، مقایسه مستقیم عملکرد مدل‌ها بین دو سناریو است. مشاهده می‌شود که امتیازات F1 و کاپا برای هر دو روش خودناظارتی در سناریوی بین-مجموعه‌ای به طور کلی کمتر از سناریوی درون-مجموعه‌ای است. مجموعه داده‌های MobiAct و HAPT اگرچه هر دو به شناسایی فعالیت

انسان می‌پردازند، اما در جزئیاتی نظری نوع و محل قرارگیری سنسورها، مجموعه فعالیتهای ثبت شده و مشخصات شرکت‌کنندگان تفاوت‌هایی دارند. این تفاوت‌ها باعث ایجاد اختلاف در توزیع آماری داده‌ها شده و باعث می‌شود بازنمایی‌های آموخته شده از یک دامنه، به طور کامل به دامنه‌ی دیگر قابل انتقال نباشند.

با این حال، با وجود اینکه هر دو روش خودناظارتی تحت تاثیر تغییر دامنه دچار افت عملکرد شدند، اما میزان این افت برای روش پایه به مراتب شدیدتر بوده است. این یافته نشان می‌دهد که روش پیشنهادی مقاومت بیشتری در برابر تغییر دامنه دارد و قادر به یادگیری بازنمایی‌های عمومی‌تر و مستقل از دامنه است. در حالی که روش پایه ممکن است ویژگی‌هایی را آموخته باشد که بیش از حد به خصوصیات مجموعه داده MobiAct وابسته بوده‌اند، روش پیشنهادی توانسته است الگوهای کلی‌تری را استخراج کند که حتی پس از انتقال به یک دامنه جدید نیز کارایی بالاتری از خود نشان می‌دهند.

در نهایت، نتایج این بخش نشان می‌دهد که اگرچه فرضیه ساده "پیش‌آموزش روی داده بیشتر همیشه بهتر است" به دلیل وجود شکاف دامنه به چالش کشیده می‌شود، اما همین چالش، نقطه قوت اصلی روش پیشنهادی را آشکار می‌سازد: پایداری^۱ و قابلیت تعمیم بالاتر در شرایط واقعی که داده‌های آموزش و آزمون ممکن است از توزیع‌های متفاوتی آمده باشند. این ویژگی، ارزش عملیاتی مدل پیشنهادی را به شدت افزایش می‌دهد.

۴-۴ جزئیات اجرا

برای آموزش مدل‌ها از یک کارت گرافیک NVIDIA Tesla P100 استفاده شده است. جزئیات زمان اجرای پیش‌آموزش مدل‌ها به شرح زیر می‌باشد:

• **مجموعه داده HAPT:** هر دوره از پیش‌آموزش کدگذار سیگنال حدود ۱۲۵ ثانیه و هر دوره از پیش‌آموزش کدگذار اسکالولگرام حدود ۲۶۰ ثانیه به‌طول انجامید. نتیجتاً پیش‌آموزش کامل کدگذار سیگنال حدود ۳ و نیم ساعت و کدگذار اسکالولگرام حدود ۵ و نیم ساعت به‌طول انجامید.

• **مجموعه داده MobiAct:** هر دوره از پیش‌آموزش کدگذار سیگنال حدود ۱۳۰۰ ثانیه و هر دوره از پیش‌آموزش کدگذار اسکالولگرام حدود ۲۸۰۰ ثانیه به‌طول انجامید. نتیجتاً پیش‌آموزش کامل کدگذار سیگنال حدود ۳۶ ساعت و کدگذار اسکالولگرام حدود ۵۸ ساعت به‌طول انجامید.

^۱Robustness

یکی از معیارهای کلیدی برای نظارت بر فرایند پیش‌آموزش و اطمینان از یادگیری بازنمایی‌های مفید، رصد کردن مقدار تابع هزینه روش SwAV بود. اگر مقدار این تابع هزینه به عدد ثابت $\log(K)$ (که در آن K تعداد خوشها است) نزدیک شود، این امر نشان‌دهنده‌ی عدم یادگیری بازنمایی‌های معنادار توسط مدل است. دلیل این پدیده به شرح زیر است:

تابع هزینه SwAV، همانطور که در معادله ۱۵-۳^۱ نشان داده شد، بر اساس آنتروپی متقاطع عمل می‌کند. در این میان، بردار احتمال p از خروجی یک تابع سافت‌مکس بر روی امتیازات شباخت بین بازنمایی شبکه (z) و مراکز خوشها به دست می‌آید (معادله ۵-۳). در صورتی که مدل در حال یادگیری نباشد (مثلاً به دلیل تنظیم نامناسب ابرپارامترها)، بازنمایی خروجی آن حاوی اطلاعات مفیدی نخواهد بود. در نتیجه، امتیاز شباخت این بازنمایی با تمام K مرکز خوشه تقریباً یکسان خواهد بود. هنگامی که ورودی‌های یک تابع سافت‌مکس مقادیر یکسانی داشته باشند، خروجی آن یک توزیع احتمال یکنواخت خواهد بود، یعنی $\frac{1}{K} \approx p_k$ برای تمام خوشها. با جایگزین کردن p_k در فرمول هزینه داریم:

$$L = - \sum_{k=1}^K q_k \log \left(\frac{1}{K} \right) = \log(K) \sum_{k=1}^K q_k \quad (8-4)$$

و از آنجا که q نیز خود یک توزیع احتمال است، جمع تمام q_k برابر با ۱ می‌شود و مقدار نهایی هزینه به فرم $L = \log(K)$ در می‌آید. بنابراین، وقتی مدل یادگیری را به درستی انجام نمی‌دهد، هزینه به پایین‌ترین حد ممکن برای یک حدس تصادفی، یعنی $\log(K)$ می‌رسد.

بنابراین، مقدار $\log(K)$ به عنوان خط پایه برای یک مدل تصادفی^۱ عمل می‌کند. در یک آموزش موفق، انتظار می‌رود که مقدار هزینه به سرعت از این عدد فاصله گرفته و به سمت مقادیر کمتر کاهش یابد. نزدیک شدن یا باقی ماندن هزینه در این سطح، هشداری مبنی بر عدم یادگیری مدل است.

۴-۵ جمع‌بندی

در این فصل، عملکرد مدل پیشنهادی از طریق مجموعه‌ای از آزمایش‌های جامع مورد ارزیابی قرار گرفت. ابتدا، مجموعه داده‌های مورد استفاده به عنوان بستر آزمایش‌ها معرفی شدند و سپس جزئیات پیاده‌سازی، شامل مراحل پیش‌پردازش داده‌ها، فرایند آموزش دومرحله‌ای، ابرپارامترها و معیارهای ارزیابی تشریح گردید. نتایج در دو سناریوی اصلی ارائه شد: درون‌مجموعه‌ای و بین‌مجموعه‌ای. در سناریوی

^۱Random Baseline

فصل چهارم: آزمایش‌ها و نتایج

درون‌مجموعه‌ای، برتری آشکار رویکرد خودناظارتی پیشنهادی نسبت به یادگیری کاملاً نظارت شده، به ویژه در شرایط کمبود داده، به اثبات رسید و در اکثر موارد عملکرد بهتری نسبت به روش پایه داشت. مهم‌تر از آن، در سناریوی چالش‌برانگیز بین‌مجموعه‌ای (یادگیری انتقالی)، مدل پیشنهادی پایداری و قابلیت تعمیم بالاتری از خود به نمایش گذاشت و با اختلاف معناداری نسبت به روش پایه، برتری خود را در تمام سطوح داده‌های برچسبدار ثبت کرد. این یافته‌ها نشان می‌دهد که مدل پیشنهادی قادر به یادگیری بازنمایی‌های غنی، عمومی و مقاوم در برابر تغییر دامنه است.

فصل ۵

جمع‌بندی و نتیجه‌گیری و پیشنهادات

در این فصل، ابتدا به جمع‌بندی و مرور کلی پژوهش انجام‌شده، اهداف، رویکردها و دستاوردهای کلیدی آن می‌پردازیم. سپس، با تکیه بر نتایج به دست آمده و محدودیت‌های موجود، پیشنهاداتی برای پژوهش‌های آتی در این حوزه ارائه خواهیم داد.

۱-۵ جمع‌بندی

مسئله‌ی شناسایی فعالیت انسان با استفاده از داده‌های حسگر، یکی از ارکان اساسی در توسعه‌ی سیستم‌های هوشمند مراقبتی، خانه‌های هوشمند و سلامت دیجیتال به شمار می‌رود. با وجود موفقیت‌های چشمگیر روش‌های یادگیری عمیق در این حوزه، وابستگی شدید آن‌ها به حجم زیادی از داده‌های برچسب‌دار، یک چالش اساسی و پرهزینه محسوب می‌شود. فرایند برچسب‌گذاری داده‌های حسگری نه تنها زمان‌بر و نیازمند نیروی متخصص است، بلکه مدل‌های آموزش‌دیده به روش نظارت‌شده اغلب در تعمیم به افراد جدید یا شرایط متغیر دنیای واقعی، با افت عملکرد مواجه می‌شوند.

این پژوهش با هدف غلبه بر این محدودیت‌ها، به بررسی و پیاده‌سازی یک چارچوب یادگیری خودناظارتی برای استخراج بازنمایی‌های غنی و تعمیم‌پذیر از سیگنال‌های مربوط به فعالیت انسان پرداخت. معماری پایه، که بر دو کدگذار مجزا برای تحلیل همزمان داده‌ها در حوزه‌ی زمان (سیگنال خام) و حوزه‌ی زمان-فرکانس (اسکالوگرام) استوار بود، به عنوان نقطه شروع انتخاب شد. با این حال، با تحلیل این معماری، دو حوزه اصلی برای بهبود شناسایی و مورد هدف قرار گرفت.

نوآوری نخست، جایگزینی الگوریتم یادگیری تباینی SimCLR با روش پیشرفته‌تر و مبتنی بر خوشبندی SwAV بود. این الگوریتم با بهره‌گیری از یک سازوکار پیش‌بینی تعویض‌شده و خوشبندی برخط، مدل را به یادگیری بازنمایی‌هایی پایدارتر و متمایزتر وادار می‌کند که نسبت به تغییرات و تبدیلات داده‌افزایی، نامتغیر هستند. نوآوری دوم، بازنگری و بهبود راهبرد داده‌افزایی برای اسکالوگرام‌ها بود. به جای استفاده از تبدیلات اقتباس‌شده از حوزه‌ی بینایی کامپیوتر که فاقد معنای فیزیکی روشی برای داده‌های حسگری هستند، مجموعه‌ای از تبدیلات معنادارتر و سازگارتر با ماهیت سیگنال‌ها به کار گرفته شد.

برای ارزیابی عملکرد روش پیشنهادی، دو سناریوی آزمایشی جامع طراحی گردید. در سناریوی درون‌مجموعه‌ای، که پیش‌آموزش و تنظیم دقیق بر روی مجموعه داده‌ی واحد HAPT انجام شد، نتایج نشان داد که رویکرد خودناظارتی پیشنهادی، به ویژه در شرایط کمیود داده‌های برچسب‌دار، برتری چشمگیری نسبت به یادگیری کاملاً نظارت‌شده دارد و در اکثر موارد، عملکردی بهتر از روش پایه ارائه

فصل پنجم: جمع‌بندی و نتیجه‌گیری و پیشنهادات

می‌دهد. مهم‌تر از آن، در سناریوی چالش‌برانگیزتر بین مجموعه‌ای که به منظور سنجش قدرت یادگیری انتقالی طراحی شده بود (پیش‌آموزش بر روی مجموعه داده‌ی بزرگ MobiAct و تنظیم دقیق بر روی HAPT)، برتری مدل پیشنهادی کاملاً مشهود بود. در این سناریو که شبیه‌ساز شرایط واقعی‌تر با وجود تغییر دامنه بین داده‌های آموزشی و ارزیابی است، مدل پیشنهادی پایداری و قابلیت تعیین بالاتری از خود به نمایش گذاشت. این یافته‌ی کلیدی نشان می‌دهد که بازنمایی‌های آموخته‌شده توسط روش پیشنهادی، عمومی‌تر، بنیادی‌تر و در برابر تغییر دامنه مقاوم‌تر هستند. این یافته‌ی کلیدی نشان می‌دهد که بازنمایی‌های آموخته‌شده توسط روش پیشنهادی، عمومی‌تر، بنیادی‌تر و در برابر تغییر دامنه مقاوم‌تر هستند.

در مجموع، نتایج این پژوهش تأیید می‌کند که ترکیب الگوریتم یادگیری خودناظارتی SwAV با راهبردهای داده‌افزایی متناسب، منجر به توسعه‌ی مدلی قدرتمندتر و قابل‌اطمینان‌تر برای شناسایی فعالیت انسان می‌شود که گامی مؤثر در جهت کاهش وابستگی به داده‌های برچسب‌دار و افزایش کارایی مدل‌ها در کاربردهای دنیای واقعی است.

۲-۵ کارهای آتی

با توجه به نتایج امیدوارکننده‌ی این پژوهش، مسیرهای متعددی برای توسعه و بهبود کارهای آتی قابل تصور است که در ادامه به برخی از مهم‌ترین آن‌ها اشاره می‌شود:

- **به کارگیری معماهای پیشرفته‌تر:** کدگذارهای استفاده‌شده در این پژوهش مبتنی بر شبکه‌های پیچشی نسبتاً ساده بودند. می‌توان با بهره‌گیری از معماهای پیشرفته‌تر مانند مدل CNN-توانایی بالایی در مدل‌سازی وابستگی‌های طولانی‌مدت دارند، یا معماهای ترکیبی مانند LSTM، بازنمایی‌های دقیق‌تری از الگوهای زمانی پیچیده استخراج کرد.
- **راهبردهای ادغام چندوجهی:** در این پژوهش از راهبرد ادغام دیرهنگام برای ترکیب خروجی دو کدگذار استفاده شد. تحقیق بر روی راهبردهای ادغام زودهنگام یا میانی، که در آن‌ها بازنمایی‌های حوزه‌ی زمان و زمان-فرکانس در لایه‌های پایین‌تر با یکدیگر ترکیب می‌شوند، می‌تواند به مدل اجازه دهد تا وابستگی‌های متقابل میان این دو حوزه را بهتر بیاموزد.
- **شخصی‌سازی و یادگیری مداوم:** مدل فعلی به صورت عمومی آموزش دیده است. یک مسیر تحقیقاتی ارزشمند، توسعه‌ی روش‌هایی برای شخصی‌سازی سریع مدل برای یک کاربر جدید با

فصل پنجم: جمع‌بندی و نتیجه‌گیری و پیشنهادات

استفاده از حجم بسیار کمی داده یا تطبیق مداوم مدل با تغییرات تدریجی در الگوهای حرکتی یک فرد در طول زمان (یادگیری برخط) است.

- ارزیابی در سناریوهای پیچیده‌تر: عملکرد مدل را می‌توان در مجموعه داده‌های پیچیده‌تر و نزدیک‌تر به واقعیت ارزیابی کرد؛ مجموعه داده‌هایی که شامل فعالیت‌های همزمان و همپوشان، تعداد بیشتری از دسته‌های فعالیت، یا داده‌های حاصل از منابع متنوع‌تری (مانند حسگرهای محیطی در کنار حسگرهای پوشیدنی) هستند.

منابع و مراجع

- [1] Alaghbari, Khaled A, Saad, Mohamad Hanif Md, Hussain, Aini, and Alam, Muhammad Raisul. Activities recognition, anomaly detection and next activity prediction based on neural networks in smart homes. *IEEE Access*, 10:28219–28232, 2022.
- [2] Liao, Jing, Stankovic, Lina, and Stankovic, Vladimir. Detecting household activity patterns from smart meter data. in 2014 International Conference on Intelligent Environments, pp. 71–78. IEEE, 2014.
- [3] Almeida, Alexandre and Alves, Ana. Activity recognition for movement-based interaction in mobile games. in Proceedings of the 19th International Conference on Human-Computer Interaction with Mobile Devices and Services, pp. 1–8, 2017.
- [4] Zambonelli, Franco. Pervasive urban crowdsourcing: Visions and challenges. in 2011 IEEE international conference on pervasive computing and communications workshops (PERCOM Workshops), pp. 578–583. IEEE, 2011.
- [5] Pereyda, Christopher, Raghunath, Nisha, Minor, Bryan, Wilson, Garrett, Schmitter-Edgecombe, Maureen, and Cook, Diane J. Cyber-physical support of daily activities: A robot/smart home partnership. *ACM Transactions on Cyber-Physical Systems*, 4(2):1–24, 2019.
- [6] Jaouedi, Neziha, Perales, Francisco J, Buades, José María, Boujnah, Noureddine, and Bouhlel, Med Salim. Prediction of human activities based on a new structure of skeleton features and deep learning model. *Sensors*, 20(17):4944, 2020.

- [7] Zhu, Chun, Sheng, Weihua, and Liu, Meiqin. Wearable sensor-based behavioral anomaly detection in smart assisted living systems. *IEEE Transactions on automation science and engineering*, 12(4):1225–1234, 2015.
- [8] Lu, Wei, Fan, Fugui, Chu, Jinghui, Jing, Peiguang, and Yuting, Su. Wearable computing for internet of things: A discriminant approach for human activity recognition. *IEEE Internet of Things Journal*, 6(2):2749–2759, 2018.
- [9] Dhillon, Jagwinder Kaur, Kushwaha, Alok Kumar Singh, et al. A recent survey for human activity recognition based on deep learning approach. in 2017 fourth international conference on image information processing (ICIIP), pp. 1–6. IEEE, 2017.
- [10] Mathew, Sheryl, Subramanian, Annapoorani, MS, Balamurugan, Rajagopal, Manoj Kumar, et al. Human activity recognition using deep learning approaches and single frame cnn and convolutional lstm. *arXiv preprint arXiv:2304.14499*, 2023.
- [11] Cook, Diane J, Crandall, Aaron S, Thomas, Brian L, and Krishnan, Narayanan C. Casas: A smart home in a box. *Computer*, 46(7):62–69, 2012.
- [12] Vavoulas, George, Chatzaki, Charikleia, Malliotakis, Thodoris, Pediaditis, Matthew, and Tsiknakis, Manolis. The mobiact dataset: Recognition of activities of daily living using smartphones. in International conference on information and communication technologies for ageing well and e-health, vol. 2, pp. 143–151. SciTePress, 2016.
- [13] Roggen, Daniel, Calatroni, Alberto, Rossi, Mirco, Holleczeck, Thomas, Förster, Kilian, Tröster, Gerhard, Lukowicz, Paul, Bannach, David, Pirkl, Gerald, Wagner, Florian, et al. Walk-through the opportunity dataset for activity recognition in sensor rich environments. Helsinki, Finland, May, 2010.
- [14] Reyes-Ortiz, Jorge-L, Oneto, Luca, Samà, Albert, Parra, Xavier, and Anguita, Davide. Transition-aware human activity recognition using smartphones. *Neurocomputing*, 171:754–767, 2016.

- [15] Chen, Kaixuan, Zhang, Dalin, Yao, Lina, Guo, Bin, Yu, Zhiwen, and Liu, Yunhao. Deep learning for sensor-based human activity recognition: Overview, challenges, and opportunities. *ACM Computing Surveys (CSUR)*, 54(4):1–40, 2021.
- [16] Rumelhart, David E, Hinton, Geoffrey E, and Williams, Ronald J. Learning representations by back-propagating errors. *nature*, 323(6088):533–536, 1986.
- [17] Recht, Benjamin, Roelofs, Rebecca, Schmidt, Ludwig, and Shankar, Vaishaal. Do imagenet classifiers generalize to imagenet? in *International conference on machine learning*, pp. 5389–5400. PMLR, 2019.
- [18] Cleland, Ian, Kikhia, Basel, Nugent, Chris, Boytsov, Andrey, Hallberg, Josef, Synnes, Kåre, McClean, Sally, and Finlay, Dewar. Optimal placement of accelerometers for the detection of everyday activities. *Sensors*, 13(7):9183–9200, 2013.
- [19] Deng, Jia, Dong, Wei, Socher, Richard, Li, Li-Jia, Li, Kai, and Fei-Fei, Li. Imagenet: A large-scale hierarchical image database. in *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.
- [20] Gidaris, Spyros, Singh, Praveer, and Komodakis, Nikos. Unsupervised representation learning by predicting image rotations. *arXiv preprint arXiv:1803.07728*, 2018.
- [21] He, Kaiming, Chen, Xinlei, Xie, Saining, Li, Yanghao, Dollár, Piotr, and Girshick, Ross. Masked autoencoders are scalable vision learners. in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 16000–16009, 2022.
- [22] Chen, Ting, Kornblith, Simon, Norouzi, Mohammad, and Hinton, Geoffrey. A simple framework for contrastive learning of visual representations. in *International conference on machine learning*, pp. 1597–1607. PmLR, 2020.
- [23] Grill, Jean-Bastien, Strub, Florian, Altché, Florent, Tallec, Corentin, Richemond, Pierre, Buchatskaya, Elena, Doersch, Carl, Avila Pires, Bernardo, Guo, Zhaohan,

- Gheshlaghi Azar, Mohammad, et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in neural information processing systems*, 33:21271–21284, 2020.
- [24] Caron, Mathilde, Misra, Ishan, Mairal, Julien, Goyal, Priya, Bojanowski, Piotr, and Joulin, Armand. Unsupervised learning of visual features by contrasting cluster assignments. *Advances in neural information processing systems*, 33:9912–9924, 2020.
- [25] Chen, Ting, Kornblith, Simon, Swersky, Kevin, Norouzi, Mohammad, and Hinton, Geoffrey E. Big self-supervised models are strong semi-supervised learners. *Advances in neural information processing systems*, 33:22243–22255, 2020.
- [26] Yuan, Hang, Chan, Shing, Creagh, Andrew P, Tong, Catherine, Acquah, Aidan, Clifton, David A, and Doherty, Aiden. Self-supervised learning for human activity recognition using 700,000 person-days of wearable data. *NPJ digital medicine*, 7(1):91, 2024.
- [27] Hammerla, Nils, Fisher, James, Andras, Peter, Rochester, Lynn, Walker, Richard, and Plötz, Thomas. Pd disease state assessment in naturalistic environments using deep learning. in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 29, 2015.
- [28] Asghari, Parviz and Nazerfard, Ehsan. Activity recognition using hierarchical hidden markov models on streaming sensor data. in *2018 9th International Symposium on Telecommunications (IST)*, pp. 416–420. IEEE, 2018.
- [29] Cook, Diane, Feuz, Kyle D, and Krishnan, Narayanan C. Transfer learning for activity recognition: A survey. *Knowledge and information systems*, 36:537–556, 2013.
- [30] Chen, Yuqing and Xue, Yang. A deep learning approach to human activity recognition based on single accelerometer. in *2015 IEEE international conference on systems, man, and cybernetics*, pp. 1488–1492. IEEE, 2015.

- [31] Ha, Sojeong, Yun, Jeong-Min, and Choi, Seungjin. Multi-modal convolutional neural networks for activity recognition. in 2015 IEEE International conference on systems, man, and cybernetics, pp. 3017–3022. IEEE, 2015.
- [32] Foerster, Friedrich, Smeja, Manfred, and Fahrenberg, Jochen. Detection of posture and motion by accelerometry: a validation study in ambulatory monitoring. Computers in human behavior, 15(5):571–583, 1999.
- [33] Attal, Ferhat, Mohammed, Samer, Dedabirshvili, Mariam, Chamroukhi, Faicel, Oukhellou, Latifa, and Amirat, Yacine. Physical human activity recognition using wearable sensors. Sensors, 15(12):31314–31338, 2015.
- [34] Guo, Haodong, Chen, Ling, Peng, Liangying, and Chen, Gencai. Wearable sensor based multimodal human activity recognition exploiting the diversity of classifier ensemble. in Proceedings of the 2016 ACM international joint conference on pervasive and ubiquitous computing, pp. 1112–1123, 2016.
- [35] Gao, Wenbin, Zhang, Lei, Teng, Qi, He, Jun, and Wu, Hao. Danhar: Dual attention network for multimodal human activity recognition using wearable sensors. Applied Soft Computing, 111:107728, 2021.
- [36] Wang, Huaijun, Zhao, Jing, Li, Junhuai, Tian, Ling, Tu, Pengjia, Cao, Ting, An, Yang, Wang, Kan, and Li, Shancang. Wearable sensor-based human activity recognition using hybrid deep learning techniques. Security and communication Networks, 2020(1):2132138, 2020.
- [37] Liu, Xiao, Zhang, Fanjin, Hou, Zhenyu, Mian, Li, Wang, Zhaoyu, Zhang, Jing, and Tang, Jie. Self-supervised learning: Generative or contrastive. IEEE transactions on knowledge and data engineering, 35(1):857–876, 2021.

- [38] Ericsson, Linus, Gouk, Henry, Loy, Chen Change, and Hospedales, Timothy M. Self-supervised representation learning: Introduction, advances, and challenges. *IEEE Signal Processing Magazine*, 39(3):42–62, 2022.
- [39] Jing, Longlong and Tian, Yingli. Self-supervised visual feature learning with deep neural networks: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 43(11):4037–4058, 2020.
- [40] DeTone, Daniel, Malisiewicz, Tomasz, and Rabinovich, Andrew. Superpoint: Self-supervised interest point detection and description. in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pp. 224–236, 2018.
- [41] Noroozi, Mehdi and Favaro, Paolo. Unsupervised learning of visual representations by solving jigsaw puzzles. in *European conference on computer vision*, pp. 69–84. Springer, 2016.
- [42] Li, Ru, Liu, Shuaicheng, Wang, Guangfu, Liu, Guanghui, and Zeng, Bing. Jigsawgan: Auxiliary learning for solving jigsaw puzzles with generative adversarial networks. *IEEE Transactions on Image Processing*, 31:513–524, 2021.
- [43] Park, Wongi and Ryu, Jongbin. Fine-grained self-supervised learning with jigsaw puzzles for medical image classification. *Computers in Biology and Medicine*, 174:108460, 2024.
- [44] Misra, Ishan, Zitnick, C Lawrence, and Hebert, Martial. Shuffle and learn: unsupervised learning using temporal order verification. in *European conference on computer vision*, pp. 527–544. Springer, 2016.
- [45] Banville, Hubert, Albuquerque, Isabela, Hyvärinen, Aapo, Moffat, Graeme, Engemann, Denis-Alexander, and Gramfort, Alexandre. Self-supervised representation learning from electroencephalography signals. in *2019 IEEE 29th International Workshop on Machine Learning for Signal Processing (MLSP)*, pp. 1–6. IEEE, 2019.

- [46] Vincent, Pascal, Larochelle, Hugo, Bengio, Yoshua, and Manzagol, Pierre-Antoine. Extracting and composing robust features with denoising autoencoders. in Proceedings of the 25th international conference on Machine learning, pp. 1096–1103, 2008.
- [47] Pathak, Deepak, Krahenbuhl, Philipp, Donahue, Jeff, Darrell, Trevor, and Efros, Alexei A. Context encoders: Feature learning by inpainting. in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 2536–2544, 2016.
- [48] Jaiswal, Ashish, Babu, Ashwin Ramesh, Zadeh, Mohammad Zaki, Banerjee, Debapriya, and Makedon, Fillia. A survey on contrastive self-supervised learning. Technologies, 9(1):2, 2020.
- [49] Oord, Aaron van den, Li, Yazhe, and Vinyals, Oriol. Representation learning with contrastive predictive coding. arXiv preprint arXiv:1807.03748, 2018.
- [50] Wu, Zhirong, Xiong, Yuanjun, Yu, Stella X, and Lin, Dahua. Unsupervised feature learning via non-parametric instance discrimination. in Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 3733–3742, 2018.
- [51] He, Kaiming, Fan, Haoqi, Wu, Yuxin, Xie, Saining, and Girshick, Ross. Momentum contrast for unsupervised visual representation learning. in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp. 9729–9738, 2020.
- [52] Vaswani, Ashish, Shazeer, Noam, Parmar, Niki, Uszkoreit, Jakob, Jones, Llion, Gomez, Aidan N, Kaiser, Łukasz, and Polosukhin, Illia. Attention is all you need. Advances in neural information processing systems, 30, 2017.
- [53] Devlin, Jacob, Chang, Ming-Wei, Lee, Kenton, and Toutanova, Kristina. Bert: Pre-training of deep bidirectional transformers for language understanding. in Proceedings of the 2019 conference of the North American chapter of the association for computa-

- tional linguistics: human language technologies, volume 1 (long and short papers), pp. 4171–4186, 2019.
- [54] Radford, Alec, Wu, Jeffrey, Child, Rewon, Luan, David, Amodei, Dario, Sutskever, Ilya, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.
- [55] Haresamudram, Harish, Essa, Irfan, and Plötz, Thomas. Contrastive predictive coding for human activity recognition. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 5(2):1–26, 2021.
- [56] Khaertdinov, Bulat, Ghaleb, Esam, and Asteriadis, Stylianos. Contrastive self-supervised learning for sensor-based human activity recognition. in 2021 IEEE International Joint Conference on Biometrics (IJCB), pp. 1–8. IEEE, 2021.
- [57] Jain, Yash, Tang, Chi Ian, Min, Chulhong, Kawsar, Fahim, and Mathur, Akhil. Collossl: Collaborative self-supervised learning for human activity recognition. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 6(1):1–28, 2022.
- [58] Taghanaki, Setareh Rahimi, Rainbow, Michael, and Etemad, Ali. Self-supervised human activity recognition with localized time-frequency contrastive representation learning. *IEEE Transactions on Human-Machine Systems*, 53(6):1027–1037, 2023.
- [59] Caron, Mathilde, Bojanowski, Piotr, Joulin, Armand, and Douze, Matthijs. Deep clustering for unsupervised learning of visual features. in *Proceedings of the European conference on computer vision (ECCV)*, pp. 132–149, 2018.
- [60] Cuturi, Marco. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26, 2013.

- [61] Reyes-Ortiz, Jorge, Anguita, Davide, Oneto, Luca, and Parra, Xavier. Smartphone-based recognition of human activities and postural transitions. UCI Machine Learning Repository, 2015.
- [62] Anguita, Davide, Ghio, Alessandro, Oneto, Luca, Parra, Xavier, Reyes-Ortiz, Jorge Luis, et al. A public domain dataset for human activity recognition using smartphones. in Esann, vol. 3, pp. 3–4, 2013.

Abstract

With the increasing expansion of smart environments and the use of various sensors such as mobile phones, the need for systems that can automatically and accurately recognize human activities has grown. One of the main challenges in this field is the heavy reliance of machine learning models on labeled data, the collection of which on a large scale is costly and time-consuming. This issue highlights the necessity of utilizing methods that can extract conceptual and transferable representations from sensor data without the need for manual labeling. In this research, a self-supervised learning framework has been designed that, by leveraging a combination of temporal and frequency perspectives, attempts to extract high-quality and general representations from raw human activity data. The proposed framework, aiming to improve data representation quality, reduce the need for labeled data, and enhance model generalizability, has been evaluated in two different scenarios: first, training and evaluation in the same environment; and second, training in one environment and evaluating in a different one, with the goal of assessing the model's knowledge transfer and generalization capabilities. The evaluation results show that the proposed framework has achieved **92.3% F1 score** (with a **0.9% improvement** over the base model) in the direct training scenario on the HAPT dataset, and **90.2% F1 score** (with a **1.5% improvement** over the base model) in the transfer learning scenario from the MobiAct dataset to the HAPT dataset. By utilizing the combination of temporal and frequency information, this framework has been able to extract representations that have led to improved accuracy in human activity recognition. Overall, the proposed method takes an effective step towards reducing dependency on labeled data and developing generalizable models for application in diverse and real-world environments.

Key Words:

Human Activity Recognition, Self-Supervised Learning, Wavelet Transform, Contrastive Learning, Transfer Learning



**Amirkabir University of Technology
(Tehran Polytechnic)**

Department of Computer Engineering

M. Sc. Thesis

Human Activity Recognition in Smart Environments Using Self-supervised Learning

By
Ardalan Nahavandi Fard

Supervisor
Dr. Ehsan Nazerfard

September 2025