دانشگاه صنعتی شریف
دانشکده مهندسی کامپیوتر

پایان‌نامه کارشناسی ارشد
گرایش هوش مصنوعی

عنوان

دسته‌بندی در دسته‌های زیاد با تعداد نمونه محدود

نگارش
محمدرضا فریدونی

استاد راهنما
دکتر مهدیه سلیمانی باغشاه

زمستان ۱۴۰۲

بسم الله الرحمن الرحیم

دانشگاه صنعتی شریف
دانشکده مهندسی کامپیوتر

پایان‌نامه کارشناسی ارشد

دسته‌بندی در دسته‌های زیاد با تعداد نمونه محدود

نگارش: محمدرضا فریدونی

استاد راهنما:	دکتر مهدیه سلیمانی باغشاه	امضاء:
استاد ممتحن داخلی:	دکتر حمیدرضا ربیعی	امضاء:
استاد ممتحن خارجی:	اسم داور	امضاء:

اظهارنامه
(اصالت متن و محتوای پایان‌نامه کارشناسی‌ارشد)

عنوان پایان‌نامه: دسته‌بندی در دسته‌های زیاد با تعداد نمونه محدود

نام استاد راهنما: دکتر مهدیه سلیمانی باغشاه نام استاد راهنمای هم‌کار: - - - نام استاد مشاور: - - -

این‌جانب محمدرضا فریدونی اظهار می‌دارم:

۱. متن و نتایج علمی ارائه شده در این پایان‌نامه اصیل بوده و منحصرا توسط این‌جانب زیر نظر استادان (راهنما، همکار و مشاور) نام‌برده شده در بالا تهیه شده است.

۲. متن پایان‌نامه به این صورت در هیچ جای دیگری منتشر نشده است.

۳. متن و نتایج مندرج در این پایان‌نامه، حاصل تحقیقات این‌جانب به عنوان دانشجوی کارشناسی‌ارشد دانشگاه شریف است.

۴. کلیه مطالبی که از منابع دیگر در این پایان‌نامه مورد استفاده قرار گرفته، با ذکر منبع مشخص شده است.

نام دانشجو: محمدرضا فریدونی
تاریخ: ۱۴۰۲/۱۱/۱۱
امضا

نتایج تحقیقات مندرج در این پایان‌نامه و دستاوردهای مادی و معنوی ناشی از آن (شامل فرمول‌ها، توابع کتابخانه‌ای، نرم‌افزارها، سخت‌افزارها و مواردی که قابلیت ثبت اختراع دارد) متعلق به دانشگاه صنعتی شریف است. هیچ شخصیت حقیقی یا حقوقی بدون کسب اجازه از دانشگاه صنعتی شریف حق فروش و ادعای مالکیت مادی یا معنوی بر آن یا ثبت اختراع از آن را ندارد. همچنین کلیه حقوق مربوط به چاپ، تکثیر، نسخه‌برداری، ترجمه، اقتباس و نظائر آن در محیط‌های مختلف اعم از الکترونیکی، مجازی یا فیزیکی برای دانشگاه صنعتی شریف محفوظ است. نقل مطالب با ذکر ماخذ بلامانع است.

نام استاد راهنما: دکتر مهدیه سلیمانی باغشاه
تاریخ: ۱۴۰۲/۱۱/۱۱
امضا

نام دانشجو: محمدرضا فریدونی
تاریخ: ۱۴۰۲/۱۱/۱۱
امضا

قدردانی

خدا را سپاس گزارم که توفیق کسب علم را به من عطا کرد و در طول سالیان تحصیل، من را در این راه ثابت قدم نگه داشت.

از پدر و مادرم و خواهر مهربانم به جهت حمایت های بی پایان و دلگرمی هایشان ممنونم.

از سرکار خانم دکتر سلیمانی عزیز که دانش و وقت ارزشمندشان را در اختیار بنده قرار دادند و در طول انجام این پروژه من را از راهنمایی ها و حمایت هایشان بهره مند ساختند، تشکر و قدردانی می کنم.

همچنین از همکاری اعضای آزمایشگاه یادگیری ماشین دانشگاه صنعتی شریف سپاس گزارم و خرسندم مدتی چند را در جمع این عزیزان گذراندم. از دوستان عزیزم آقایان حسین حسنی و علی رازقندی که برای به ثمر نشستن این پروژه زحمات ارزشمندی کشیدند، تشکر ویژه می کنم.

در نهایت پیشاپیش از استاد محترم جناب آقای دکتر ربیعی برای قبول زحمت داوری پایان نامه و نظرات سازنده ی ایشان سپاسگزارم.

هرگز نمیرد آن که دلش زنده شد به عشق ثبت است بر جریده ی عالم دوام ما

چکیده

در دنیای واقعی، غالبا با مسائل با تعداد دسته زیاد و دادگان کم در هر دسته روبرو هستیم. به بیان دیگر، مسائل در دنیای واقعی دسته‌بندی بزرگ مقیاس با نمونه محدود هستند. در حالی که پیشرفت و موفقیت مدل‌های یادگیری ژرف عمدتا در مسئله‌هایی با تعداد دسته محدود، ثابت و با دادگان بسیار زیاد برای هر کلاس است.

روش‌های نمونه محدود در سال‌های اخیر با معرفی رویکردهای نوین در قالب متایادگیری به کارایی قابل توجهی رسیده‌اند. با این حال، هنوز این روش‌ها در تنظیمات مقیاس بزرگ که چند هزار کلاس برای دسته‌بندی وجود دارد، به چالش می‌خورند.

در این پژوهش، مسئله در قالب دو سطح دسته‌بندی با ریزدانگی متفاوت در قالب یک یادگیری سلسله‌مراتبی در قالب یک چارچوب بیزی برای ترکیب اطلاعات مدل شده است. در سطح درشت‌دانه، یک روش تشخیص شی که مقاوم به تغییر زیر جمعیت است، معرفی شده است.

در سطح ریزدانه، دسته‌بندهایی با رویکرد مولد طراحی شده‌اند که متخصص دسته‌بندی در حیطه‌ی یک کلاس درشت‌دانه هستند. این دسته‌بندها، یادگیرهای نمونه محدودی هستند که در فرآیند یادگیری خود، از اطلاعات سطح درشت‌دانه استفاده می‌کنند و در تنظیمات متایادگیری آموزش می‌بینند تا خواص تعمیم‌پذیری پیدا کنند.

همچنین، از مدل‌های بنیادی پیش‌آموزش دیده برای رسیدن به بازنمایی‌های غنی و بهینگی محاسبات بهره برده شده است. در واقع این پژوهش یک تنظیمات بدیع برای مسئله‌ی یادگیری بزرگ مقیاس و نمونه محدود با وجود چالش‌هایی مانند تغییر زیر جمعیت ارائه داده است. به وسیله‌ی آزمایش‌ها، کارایی و تاثیر این روش نشان داده شده است.

کلمات کلیدی: یادگیری ژرف، یادگیری سلسله‌مراتبی، دسته‌بندي بزرگ مقیاس، یادگیری بانمونه‌ی محدود، متایادگیری، تعمیم‌پذیری

فهرست مطالب

۱ مقدمه ۱

۱-۱ تعریف مسئله ۱

۱-۲ رویکرد حل مسئله ۵

۱-۲-۱ معماری مدل سطح درشت‌دانه ۵

۱-۲-۲ معماری مدل سطح ریزدانه ۵

۱-۲-۳ تنظیمات متاآموزش و متاتست ۶

۱-۳ چالش‌ها ۷

۱-۳-۱ محدودیت تعداد کلاس در روش‌های نمونه محدود ۷

۱-۳-۲ مجموعه‌باز بودن دسته‌بندی بزرگ مقیاس ۷

۱-۳-۳ انتشار خطا در سلسله‌مراتب ۸

۱-۴ اهمیت و کاربرد ۸

۱-۵ هدف پژوهش ۹

۱-۶ کد پیاده‌سازی و منابع جانبی ۹

۱-۷ ساختار پایان‌نامه ۱۰

۲ پژوهش‌های پیشین ۱۱

۲-۱ یادگیری نمونه محدود با داده‌افزایی ۱۱

۲-۱-۱ داده‌افزایی در سطح بردار ویژگی ۱۲

۲-۱-۲ داده‌افزایی در سطح نمونه ۱۳

۲-۲ متایادگیری ۱۴

۲-۱.۲ ساختار کلی الگوریتم متایادگیری ۱۶
۲-۲.۲ روش‌های مبتنی بر بهینه‌سازی ۱۶
۲-۳.۲ روش‌های جعبه سیاه [1] ۱۸
۲-۴.۲ روش‌های مبتنی بر متر [2] ۱۹
۲-۵.۲ اشتراک تجربه‌ی احتمالاتی ۲۱
۲-۳ استفاده از بدنه [3] بیانگر [4] برای رسیدن به کارایی نمونه [5] ۲۲
۲-۴ خاصیت مجموعه باز بودن ۲۲
۲-۱.۴ رویکرد سلبی؛ شناسایی دسته‌های بدیع ۲۳
۲-۲.۴ رویکرد ایجابی؛ تنظیم مدل با اضافه کردن توانایی طبقه‌بندی دسته‌های جدید . . ۲۵
۲-۵ یادگیری سلسله‌مراتبی نمونه محدود ۲۷
۲-۶ جمع‌بندی ۳۰

۳ راهکار پیشنهادی ۳۲
۳-۱ نمادگذاری مسئله و پیش‌نیازها ۳۲
۳-۲ دسته‌بند سطح درشت‌دانه ۳۵
۳-۳ یادگیرنده نمونه محدود در سطح ریزدانه ۳۶
۳-۴ جمع‌بندی روش و ارائه‌ی شبه‌کد ۴۰

۴ ارزیابی ۴۲
۴-۱ معرفی مجموعه داده‌های مورد استفاده ۴۲
۴-۲ معرفی بدنه استخراج‌گر ویژگی ۴۴
۴-۳ جزییات معماری سطح درشت‌دانه ۴۵
۴-۴ جزییات معماری سطح ریزدانه ۴۵
۴-۵ روش‌های پایه [6] و نحوه‌ی مقایسه ۴۶
۴-۶ نتایج ۴۷

[1] Black Box
[2] Metric Based
[3] Backbone
[4] Expressive
[5] Sample Efficiency
[6] Baselines

۴-۷ منابع سخت‌افزاری ۵۱
۴-۸ جمع‌بندی ۵۱

۵ جمع‌بندی و کارهای آتی **۵۳**
۵-۱ جمع‌بندی و نقاط قوت روش ۵۳
۵-۲ کارهای آتی ۵۴

مراجع **۵۶**

فهرست تصاویر

۲-۱ قسمت خیال‌باف با G و دسته‌بند با h نشان داده شده است. برای یک اپیزود متایادگیری، به نمونه‌های پشتیبان هر دسته تعدادی داده اضافه می‌شود. مسیرهایی که گردیان از آن‌ها برمی‌گردد با نقطه‌چین قرمز مشخص شده‌اند [۱]. ۱۲

۲-۲ تنظیم توزیع به وسیله‌ی نمونه‌برداری از توزیع‌های تخمین‌زده؛ ابتدا برای ویژگی‌های دسته‌های نمونه محدود، نزدیک ترین دسته‌ها که تعداد کافی نمونه دارند پیدا می‌شود و سپس با تنظیم پارامترهای توزیع مفروض، از توزیع جدید به‌دست آمده برای هر دسته نمونه‌های جدید برداشته می‌شود که با مثلث نشان داده شده‌است [۲]. ۱۳

۲-۳ آموزش به شیوه‌ی متایادگیری؛ هر سطر یک اپیزود یا وظیفه است که یک مسئله یادگیری مستقل است. کادرهای سبز نمونه‌های پشتیبان و کادرهای قرمز نمونه‌های پرسش است و در طول یک سطر یادگیری سریع یا حلقه‌ی داخلی انجام می‌شود. در طول سطرها وظایف مختلف یاد گرفته می‌شوند و به وسیله‌ی آن‌ها متاپارامترها یادگرفته می‌شوند [۳]. ۱۵

۲-۴ متایادگیری مستقل از مدل؛ در این روش، با تعداد کمی گام بهینه‌سازی روی متاپارامترها، به پارامترهای خاص هر وظیفه می‌رسیم [۴]. ۱۷

۲-۵ متایادگیری جعبه سیاه؛ در شکل سمت چپ که تنظیمات یادگیری با ناظر آمده است، می‌توان از $X_{t-۳}$ تا $X_{t-۱}$ را به همراه برچسب هر کدام به عنوان مجموعه‌ی داده‌های پشتیبان در نظر گرفت و X_t را داده‌ی پرسش در نظر گرفت. به دلیل ماهیت شبکه‌های بازگشتی، با به روز رسانی فضای نهان، به پارامترهای خاص وظیفه می‌رسیم [۵]. ۱۸

۲-۶ متایادگیری مبتنی بر متر؛ در این دسته از روش‌ها برای هر دسته یک مرکز از روی نمونه‌های پشتیبان هر دسته یا متادیتای آن محاسبه می‌شود و برای یک نمونه پرسش، با سنجش فاصله از مرکز دسته‌ها، طبقه‌بندی انجام می‌شود. در سمت چپ محاسبه‌ی نماینده دسته از روی نمونه‌های پشتیبان و در سمت راست محاسبه‌ی نماینده دسته از روی متادیتای دسته‌ها انجام شده است [۶]. ۱۹

۲-۷ انواع مختلف انتقال تجربه از متاپارامترها به پارامترهای خاص وظیفه؛ در شکل سمت چپ، متاپارامترها به صورت یکسان در اختیار تمامی وظیفه‌ها قرار می‌گیرد. در شکل وسط، به ازای هر وظیفه خاص، یک متاپارامتر خاص نمونه‌گیری می‌شود و به آن وظیفه انتقال می‌یابد. در شکل سمت راست، وظایف خوشه‌بندی می‌شوند و به هر خوشه، تجربیات یکسانی انتقال می‌یابد. [۷] ۲۱

۲-۸ مدل CLIP سعی می‌کند بازنمایی تصاویر و برچسب‌های متناظر که روی قطر اصلی قرار دارند را به هم نزدیک کند. در نهایت از این ویژگی برای دسته‌بندی بدون نمونه استفاده می‌کند [۸]. ۲۳

۲-۹ استفاده از یک خودکدگذار و تولید احتمالات دسته‌بندی برای یک ورودی که بر روی داده‌های درون توزیع آموزش داده شده است (قسمت آبی) و آموزش قسمت قرمز که وظیفه‌ی تشخیص داده‌هایی که از دسته‌های بدیع آمده اند را دارد [۹]. ۲۴

۲-۱۰ مصور سازی تابع انرژی روی نمونه‌های واقعی، دستکاری شده و خارج از توزیع به ازای هر دسته. با توجه به این شکل، مشخص است در داده‌هایی که از دسته‌ی بدیع می‌آیند، خطوط سبز پرتکرار است و به این ترتیب می‌توان آن‌ها را تشخیص داد [۱۰]. ۲۵

۲-۱۱ مصورسازی به کمک t-SNE برای مقادیر فعال‌سازی هر دسته قبل از لایه آخر یک شبکه‌ی عصبی تمام متصل در سمت چپ و پارامترهای مربوط به هر دسته در لایه‌ی آخر شبکه‌ی تمام متصل در سمت راست. هر نقطه‌ی آبی یک دسته را نشان می‌دهد. نقاط مشخص شده با شکل یکسان و رنگ یکسان مربوط به یک دسته اند. از طرفی اشکال مشابه دسته‌های درشت‌دانه را نشان می‌دهند. به عنوان مثال مثلث‌ها دسته‌های مربوط به پرندگان اند [۱۱]. ۲۶

۲-۱۲ یک شبکه‌ی عصبی پایه که با دسته‌های درون دامنه آموزش داده شده است و وزن‌های لایه‌ی آخر آن با مستطیل سبز نشان داده شده است. سپس برای داده‌های خارج از توزیع که از دسته‌های بدیع هستند، از این شبکه پیش‌آموزش دیده به عنوان استخراج‌گر تعبیه استفاده می‌شود و این تعبیه بدست آمده که با رنگ نارنجی نشان داده شده است، به عنوان وزن‌های کلاس جدید اضافه و به نوعی حک می‌شود. [۱۲]. ۲۶

۲-۱۳ متایادگیری مبتنی بر متر سلسله‌مراتبی؛ در ریزدانه‌ترین حالت، نماینده‌های دسته‌ها محاسبه می‌شود و به صورت بازگشتی، از روی آن‌ها نماینده‌ی دسته‌های سطوح بالاتر یا متانماینده‌ها محاسبه می‌شود. در زمان استنتاج، دسته‌بندی در همه‌ی سطوح انجام می‌شود و خطا برای سطوح ریزدانه و درشت‌دانه محاسبه و وزن‌ها به روز رسانی می‌شوند [۱۳]. ۲۷

۲-۱۴ با توجه به اینکه مدل CLIP در برچسب‌های درشت‌دانه عملکرد مناسبی ندارد، برای هر دسته فرزندان با برچسب‌های ریزدانه به‌دست می‌آید و دسته‌بندی در آن سطح انجام می‌شود و سپس حاصل دسته‌بندی به سطح درشت‌دانه منتشر می‌شود [۱۴]. ۲۹

۲-۱۵ تشکیل درخت سلسله‌مراتب با رویکرد داده محور؛ از بازنمایی زبانی دسته‌ها که مفاهیم معنایی در خود دارند استفاده می‌کنیم و با یک روش خوشه‌بندی، این درخت را تشکیل می‌دهیم. منظور از Source دسته‌های مبدا و Target دسته‌های مقصد اند و همینطور هاSuperclass ابردسته‌ها را نشان می‌دهند [۱۵]. ۲۹

۲-۱۶ وارد کردن اطلاعات سلسله‌مراتب در شبکه‌ی بازنمایی مشترک بین مبدا و مقصد که با رنگ قرمز مشخص شده است. با استفاده از داده‌های دسته‌های مبدا که تعداد بالای نمونه دارند، دسته‌بندهای موجود در مستطیل بنفش آموزش داده می‌شوند و با گرادیانی که از آن‌ها به شبکه‌ی بازنمایی مشترک در مستطیل قرمز برمی‌گردد، تعبیه‌هایی با قابلیت انتقال بین مبدا و مقصد تولید می‌شود [۱۵]. ۳۰

۳-۱ تنظیمات مسئله پیش‌رو؛ در سطح درشت‌دانه، ابردسته‌ها بین مجموعه‌ی دادگان مبدا و مقصد مشترک هستند چرا که دسته‌های درشت‌دانه دچار اضافه یا کم شدن نمی‌شوند. اما در سطح ریزدانه، دسته‌ها و نمونه‌های موجود در هر دسته، بین دادگان مبدا و مقصد ناسازگار و بدون هیچ اشتراکی طراحی شده‌اند. ۳۴

۳-۲ نمای کلی روش پیشنهادی؛ در قسمت بالا، دسته‌بند یک پرسش q را به عنوان ورودی گرفته‌است. دسته‌بندهای درشت‌دانه‌ی g_ϕ و f_φ که با رنگ‌آبی مشخص شده‌اند، محدوده‌ی دسته‌های پایه را محدود می‌کنند تا با یک رویکرد بالا به پایین، دقت دسته‌بندهای ریز دانه‌ی g_θ و f_ϑ را با محدودکردن دسته‌های ممکن، بالا ببرند. در شکل پایین سمت چپ، به ازای داده‌های پشتیبان یک دسته، شبه‌برچسب مشابه روندی که در تنظیم ترکیبی داشتیم، به‌دست می‌آید و با مشروط کردن شبکه‌ی ریزدانه روی شبه‌برچسب، نماینده‌ی یک دسته ساخته می‌شود. در شکل پایین سمت راست، نماینده دسته‌های فعال که مربوط به w ابردسته‌ی برنده هستند با رنگ قرمز نشان داده شده‌اند و باقی نماینده دسته‌های موجود در کانتور ابردسته‌های دیگر غیرفعال اند. ۳۹

۴-۱ دسته‌بندی زیستی موجودات مطابق با Taxonomic Rank ؛ از بالا به پایین سطوح درشت‌دانه تا ریزدانه چیده شده‌اند و سلسله‌مراتب مربوط به روباه قرمز در طول درخت سلسله‌مراتب مشخص شده‌است [۱۶]. ۴۴

۴-۲ مقایسه‌ی دقت‌ها در سه حالت پایه (تخت) با رنگ سبز، دسته‌بندی سلسله‌مراتبی طبق روش پیشنهادی با رنگ نارنجی و دسته‌بندی سلسله‌مراتبی با پیشگویی سطح درشت‌دانه با رنگ آبی؛ کارایی و افزایش صحت در این نمودارها مشخص است. نمودارهای آ و ب مربوط به iNat Animalia و نمودارهای ج و د مربوط به iNat Plantae است. همچنین ستون سمت راست رویکرد بدون آموزش را نشان می‌دهد. ۵۰

۳-۴ مصورسازی بازنمایی‌های تولید شده برای پروتوتایپ‌های هر دسته‌ی پایه در سطح ریزدانه؛ با توجه به این شکل‌ها مشخص است که دسته‌های مبدا و مقصد متعلق به یک ابردسته در نزدیکی یکدیگر قرار می‌گیرند چرا که شبکه‌ی کدگذار در سطح ریزدانه، بر روی اطلاعات ابردسته مشروط شده است و طوری یادگرفته است که محدوده‌ی هر ابردسته را مشخص کند. ۵۲

فهرست جداول

۱-۱ دسته‌بندی مسائل یادگیری و تمرکز پژوهش فعلی ۳

۱-۲ بررسی شاخصه‌های پژوهش پیش‌رو در کارهای پیشین؛ اگر دسته‌ای به صورت ذاتی و یا با اعمال روندی ساده در اکثریت مقالاتش بتواند یک ویژگی را داشته باشد، برگزیده شده‌است. در غیر اینصورت، اگر نیاز به ترکیب با رویکردهای دیگری برای داشتن یکی از ویژگی‌های مشخص شده داشته باشد، انتخاب نشده است. ۳۱

۱-۳ نمادهای اختصاری مورد استفاده در این فصل و توضیح هر نماد ۳۴

۱-۴ جزییات کمی مجموعه دادگان iNat Animalia و iNat Plantae و تقسیم‌بندی آن‌ها در قالب تنظیمات مسئله. ۴۴

۴-۲ صحت شبکه‌ی سطح درشت‌دانه متشکل از بدنه‌ی g_ϕ و قسمت قابل آموزش f_φ بعد از آموزش بر روی داده‌های $\mathcal{D}_{super,source}$ و بعد از تنظیم ترکیبی به صورت نیمه‌نظارتی روی داده‌های $\mathcal{D}_{super,target}$ نمایش داده شده‌است. دقت‌ها چه قبل و چه بعد از تنظیم ترکیبی، بر روی دادگان مجموعه‌ی مقصد و با تحمیل تغییر توزیع زیرجمعیت محاسبه شده‌است که از تنظیمات آزمون عادی سخت‌تر است. ۴۶

۴-۳ عملکرد روش پیشنهادی و مقایسه با مدل‌های پایه روی مجموعه دادگان iNat Animalia ؛ در این جدول، منظور از مدل پایه (تخت) استفاده از روش پروتوتیپیکال در ریزدانه‌ترین سطح است. مدل‌هایی که پسوند Cond دارند به این معنی است که طبق روش ارائه شده، از اطلاعات سطح درشت‌دانه به صورت مشروط استفاده می‌کنند. اعداد ستون آخر یعنی پیشگویی ابردسته برای زمانی است که دقت سطح درشت‌دانه صد درصد باشد و در واقع برچسب‌های ابردسته‌ها را به صورت صحیح بدانیم. ۴۸

۴-۴ عملکرد روش پیشنهادی و مقایسه با مدل‌های پایه روی مجموعه دادگان iNat Plantae ؛ در این جدول، منظور از مدل پایه (تخت) استفاده از روش پروتوتیپیکال در ریزدانه‌ترین سطح است. مدل‌هایی که پسوند Cond دارند به این معنی است که طبق روش ارائه شده، از اطلاعات سطح درشت‌دانه به صورت مشروط استفاده می‌کنند. اعداد ستون آخر یعنی پیشگویی ابردسته برای زمانی است که دقت سطح درشت‌دانه صد درصد باشد و در واقع برچسب‌های ابردسته‌ها را به صورت صحیح بدانیم. ۴۹

۴-۵ عملکرد روش دسته‌بندی سلسله‌مراتبی یکی از همه برنده در تنظیمات بدون آموزش نسبت به حالت پایه و داشتن برچسب سطح درشت‌دانه به صورت پیشگو مشخص شده‌است. این آزمایش روی دادگان iNat Animalia صورت پذیرفته‌است. ۴۹

۴-۶ عملکرد روش دسته‌بندی سلسله‌مراتبی یکی از همه برنده در تنظیمات بدون آموزش نسبت به حالت پایه و داشتن برچسب سطح درشت‌دانه به صورت پیشگو مشخص شده‌است. این آزمایش روی دادگان iNat Plantae صورت پذیرفته‌است. ۵۰

فصل ۱

مقدمه

مطالب این فصل

۱-۱ تعریف مسئله ۱
۱-۲ رویکرد حل مسئله ۵
۱-۳ چالش‌ها ۷
۱-۴ اهمیت و کاربرد ۸
۱-۵ هدف پژوهش ۹
۱-۶ کد پیاده‌سازی و منابع جانبی ۹
۱-۷ ساختار پایان‌نامه ۱۰

در این فصل ابتدا به معرفی مساله‌ی یادگیری بزرگ مقیاس، نمونه محدود [۱] پرداخته و ضمن معرفی دقیق، اهمیت و کاربردهای آن را مورد بررسی قرار می‌دهیم. در ادامه چالش‌های این حوزه را مورد بررسی قرار داده و بعد از معرفی هدف پژوهشی، ساختار این پایان‌نامه را شرح می‌دهیم.

۱-۱ تعریف مسئله

یادگیری عمیق در سالیان اخیر به طرز شگرفی توسعه پیدا کرده است و درهای جدیدی را در هوش مصنوعی گشوده است. امروزه بسیاری از مسائلی که تا قبل از معرفی یادگیری عمیق، چالش برانگیز بوده اند، دارای راه حل بدیهی هستند. در چند سال اخیر، رویکرد یادگیری مولد [۲] یک گام بزرگ دیگر در جهت پیشبرد اهداف هوش مصنوعی برداشته است. شاید بتوان گفت هوش مصنوعی و یادگیری عمیق در بین همه‌ی شاخه‌های علمی، به مرکز توجه پژوهشگران تبدیل شده است. از حدود دهه‌ی پنجاه میلادی که اولین الگوی [۳] هوش مصنوعی شکل گرفت و به الگوی سیستم‌های خبره [۴] ، هوش مصنوعی نمادین [۵] یا همان تفکر منطقی و در نهایت الگوی یادگیری [۶] که

[1] Large-Sclae Few-Shot
[2] Generative Learning
[3] Paradigm
[4] Expert Systems
[5] Symbolic AI
[6] Learning

به صورت عملکرد منطقی تعریف می‌شود رسید، در تعریف هدف هوش مصنوعی دچار ابهام بوده ایم و به پدیده‌ای به نام اثر هوش مصنوعی [1] برخورد کرده ایم. به این معنی که همواره، رسیدن به یکسری اهداف مشخص به عنوان هوش مصنوعی تعریف شده است و وقتی به آن هدف رسیده‌ایم، گفته شده که آن موضوع دیگر هدف هوش مصنوعی نیست و یک هدف متعالی در نظر گرفته شده است. با این حال اکنون هدفی که برای هوش مصنوعی متصور هستیم رسیدن به هوش مصنوعی عمومی [2] است که می‌توان آن را معادل هوش انسانی در نظر گرفت. در چند سال اخیر، محدودیت‌هایی که الگوریتم‌های یادگیری عمیق دارند بیشتر مورد توجه قرار گرفت که حاصل آن پدید آمدن رویکردهای یادگیری با الهام از یادگیری انسان است. یعنی علاوه بر اینکه در سطح عملکرد می‌خواهیم مشابه هوش انسانی عمل کنیم، در سطح الگوریتم یادگیری نیز فرآیندهای یادگیری انسان را وارد کنیم. به این ترتیب می‌توانیم محدودیت‌هایی که در فرآیند آموزش یا استنتاج یک شبکه‌ی عمیق وجود دارند را تا حدی کاهش دهیم.

از جمله‌ی این محدودیت‌ها داده ناکارآمدی [3] و تعمیم‌پذیری [4] است. همواره یکی از مشکلات الگوریتم‌های یادگیری ژرف، بازدهی کم از لحاظ داده بوده است. یعنی برای آموزش یک مدل برای وظیفه‌ای خاص، مدل در صورتی به عملکرد قابل قبول می‌رسد که داده‌ی با حجم بالایی از آن مسئله را دیده باشد. این در حالی است که انسان در یادگیری همان وظیفه [5] می‌تواند با مشاهده‌ی تعداد محدودی نمونه، مسئله را به خوبی یاد بگیرد. در نتیجه علی رغم تمامی پیشرفت‌های شگرف هوش مصنوعی، به خصوص از زمان پیدایش مجموعه داده‌ی بزرگ ایمیجنت [6] که افزایش توان محاسباتی به همراه عمیق شدن شبکه‌ها، روند هوش مصنوعی را به نقطه‌ی فعلی رساند، همواره دغدغه‌ی مجموعه‌ی دادگان آموزش برای یک مسئله مطرح بوده است.

یک رویکردی که در ابتدا برای پاسخ‌گویی به مشکل داده ناکارآمدی شکل گرفت، گسترش مجموعه‌ی دادگان به طرق مختلف بود. یعنی در ابتدا به عنوان یک اصل پذیرفته شد که مدل‌های یادگیری ژرف پیچیدگی نمونه‌ای [7] بالایی دارند، اما سعی شد از روی مجموعه‌ی دادگان برچسب‌خورده‌ی فعلی، داده‌های بیشتری تولید و به مدل برای آموزش داده شود [۱۷] . به این دسته روش‌های مبتنی بر داده‌افزایی [8] می‌گویند. نکته‌ی قابل توجه این است که در عصر امروزه، برای اکثر مسائل، داده‌ی خام به تعداد زیاد در دسترس است اما تمیز کردن این داده و برچسب‌زدن آن برای آموزش یک شبکه در یک مسئله‌ی خاص مشکل و هزینه‌بر است. در طول سالیان گذشته، رویکردهای دیگری برای حل این مشکل توسعه داده شده است. یک رویکرد استفاده از شبکه‌های مولد است [۱،۱۸] . در شبکه‌های مولد سعی می‌شود با یادگیری توزیع توأمان [9] داده‌ها و برچسب‌ها، قدرت تولید نمونه‌ی جدید را بدست آورد. در نتیجه از آن می‌توان به عنوان یک راه مقابله با کمبود داده بهره برد. راه حل دیگر جدایی از یادگیری نظارت شده است. در این زمینه روشهایی مثل یادگیری نیمه‌نظارتی [10] و یادگیری خودنظارتی [11] مطرح شده اند [۱۹–۲۲] .

در یادگیری نیمه‌نظارتی سعی می‌شود از ترکیب یادگیری نظارت‌شده و یادگیری بدون ناظر استفاده شود. به این صورت که بخش کوچکی از داده برچسب‌گذاری شود و در کنار بخش بزرگتری از داده‌ی برچسب نخورده، در آموزش

[1] AI Effect
[2] Artificial General Intelligence
[3] Data Inefficiency
[4] Generalization
[5] Task
[6] ImageNet Moment
[7] Data Complexity
[8] Data Augmentation
[9] Joint Distribution
[10] Semi-Supervised
[11] Self-Supervised

مدل استفاده شود. یک روش مشهور در این حوزه خوشه‌بندی [1] داده‌ها در فضا و تعیین برچسب هر خوشه با توجه به نزدیکی به داده‌های برچسب‌دار است [۲۳،۲۴] . در یادگیری خودنظارتی، مدل سعی می‌کند یک علامت نظارتی از درون ساختار داده بدست آورد. یک روش مطرح در این حوزه یادگیری تباینی [2] است. در این نوع یادگیری، با بهره‌بردن از داده‌افزایی، نمونه‌های مثبت از روی یک داده تولید می‌شود. سایر نمونه‌های موجود در بسته [3] به عنوان نمونه‌ی منفی در نظر گرفته می‌شوند و تابع خطا [4] برای مدل به نوعی تعریف می‌شود که فاصله‌ی بین نمونه‌های مثبت را کمینه کند و همزمان فاصله‌ی بین یک نمونه‌ی مثبت و منفی بیشینه شود [۲۵،۲۶] . همه‌ی این تلاش‌ها برای مقابله با کمبود داده‌های برچسب‌دار یک نقطه‌ی ضعف مشترک دارند. وقتی تعداد اندکی داده در اختیار باشد، گسترش این دادگان به هر نحوی موجب به وجود آمدن سوگیری [5] است. چرا که یک نمونه‌ی کوچک از یک توزیع توانایی بازنمایی بخش کوچکی از آن توزیع را دارد و وقتی برپایه‌ی آن دادگان جدید تولید شود، سوگیری به سمت آن بخش از توزیع به وجود می‌آید [۲] .

از طرف دیگر، شبکه‌های عمیق یک سیستم خبره هستند و روی حل یک مسئله‌ی خاص تمرکز دارند. این موضوع نه تنها با هدف هوش مصنوعی عمومی در تناقض است، بلکه با مسائل دنیای واقعی تطابق کامل ندارد. چرا که برای حل یک مسئله‌ی حقیقی، باید آن را به بخش‌های کوچک‌تر خرد کرد و برای هر کدام از این بخش‌ها با فراهم کردن مجموعه دادگان برچسب‌گذاری شده، برای آن وظیفه‌ی خاص یک مدل خبره آموزش داد. رویکرد آموزش شبکه‌های ژرف به صورت احتمالاتی است و در نگاه تمایزگذار [6] همواره سعی میکند احتمال یک برچسب به شرط داده‌ی فعلی را تخمین بزند. بنابراین با فرض وجود دادگان کافی، وقتی تعداد دسته‌ها را زیاد میکنیم، احتمال‌های تخصیص داده شده به هر دسته با فاصله‌ی کمی از هم قرار می‌گیرند و با کاهش مرز تصمیم‌گیری [7] برای مدل، خطای آن بالا می‌رود. در نتیجه با بالا رفتن تعداد دسته‌ها، دقت مدل‌ها به طور محسوسی افت میکند. دسته‌بندی بزرگ مقیاس همچنان به عنوان یکی از چالش های مهم این حوزه مطرح است.

جدول ۱-۱: دسته‌بندی مسائل یادگیری و تمرکز پژوهش فعلی

	تعداد کلاس کم	تعداد کلاس زیاد
نمونه محدود	روش‌های Few-Shot	**تمرکز این پژوهش**
نمونه زیاد	تمرکز مدل‌های متداول یادگیری ژرف	

اگر برای مسائل یک افراز مطابق جدول ۱-۱ در نظر بگیریم، تمرکز این پژوهش روی مسائلی با تعداد کلاس زیاد و بزرگ مقیاس به همراه تعداد کم نمونه‌ی آموزش برای هر کلاس است. برای مسائل نمونه محدود، در چند سال اخیر راهکارهای مختلفی پیشنهاد داده شده است که یکی از مهم‌ترین آن‌ها رویکرد متایادگیری [8] است. در این رویکرد سعی می‌شود یادگیری برای فرآیند یادگیری اتفاق بیفتد و به نوعی تمرکز روی استفاده از تجربیات مشترک در مسائل یادگیری از قبل دیده شده است. در این دسته از راهکارها، معمولا تعداد کلاس‌های هر مسئله‌ی یادگیری

[1] Clustering
[2] Contrastive Learning
[3] Batch
[4] Loss Function
[5] Bias
[6] Discriminative
[7] Decision Boundary
[8] Meta-Learning

را کمتر از ۲۰ در نظر می‌گیرند، چرا که این روش‌ها در مسائل با تعداد دسته‌ی بالا نمی‌توانند به عملکرد مطلوب برسند. از طرف دیگر، در مسائلی که نمونه زیاد آموزش وجود دارد، رویکردهای مرسوم یادگیری ژرف قابل استفاده است. اما در این حوزه هم توجه به مسائلی مانند خاصیت ذاتی مجموعه باز [۱] بودن دسته‌بندی بزرگ مقیاس حائز اهمیت است. چرا که وقتی با یک مسئله‌ی دسته‌بندی چند هزار دسته‌ای روبرو هستیم، احتمال بالایی وجود دارد که به صورت مستمر به دسته‌های موجود اضافه شود. بنابراین یکی از چالش‌های اصلی این دسته از مسائل، بحث تعمیم‌پذیری و داشتن نگاه مولد به جای نگاه تمایزگذار است.

در مسائل بزرگ مقیاس، به دلیل تعدد دسته‌ها، می‌توان آن‌ها را در قالب یک سلسله‌مراتب [۲] از درشت دانه [۳] به ریزدانه [۴] مرتب کرد و از اطلاعات موجود در سلسله‌مراتب ذاتی دسته‌ها استفاده کرد. همان‌طور که انسان دنیا را به صورت سلسله‌مراتبی درک می‌کنند [۲۷–۲۹] ، چنین خاصیت ذاتی سلسله‌مراتبی در محتوای بصری مورد بررسی قرار گرفته‌است تا در طراحی معماری مدل‌های مختلف کمک کند. بنابراین، استدلال ما این بوده که در تنظیمات پرچالشی مثل دسته‌بندی بزرگ مقیاس، نمونه محدود، که نیازمند درک و استدلال ظریف است، با کمبود داده، نیاز حیاتی به استفاده از ماهیت سلسله‌مراتبی محتوای بصری است تا از اطلاعات ضمنی موجود بتوان بیشترین استفاده را برد.

در پژوهش پیش رو، تمرکز اصلی روی حل کردن یک مسئله‌ی دسته‌بندی بزرگ مقیاس با تعداد نمونه‌ی محدود آموزش برای هر دسته با نگاه سلسله مراتبی است. به صورت خلاصه، مهمترین دستاوردهای پژوهش ما را می‌توان در موارد زیر خلاصه نمود:

- ما یک روش یادگیری نوین برای استفاده از سلسله‌مراتب ذاتی موجود در دسته‌ها با طراحی یک سوگیری القایی [۵] مبتنی بر آن، برای حل مسئله‌ی یادگیری بزرگ مقیاس، نمونه محدود پیشنهاد داده‌ایم.
- از مدل‌های بنیادی [۶] پیش‌آموزش دیده [۷] برای بدست‌آوردن بازنمایی [۸] غنی برای رسیدن به داده کارآمدی و افزایش مقاومت در برابر تغییر توزیع [۹] استفاده کرده‌ایم.
- یک رویکرد کاربردی برای کاهش تغییر زیر جمعیت [۱۰] در سطح کلاس‌های درشت‌دانه یا همان ابرکلاس [۱۱] ها ارائه داده‌ایم.
- مدل معرفی شده در این پژوهش حاوی المان‌های سبک و کارآمد است که هم امکان پیاده‌سازی روی سخت‌افزار محدود در شرایط خاص مسئله یادگیری بزرگ مقیاس را فراهم می‌کند و هم نسبت به روش‌های پیشین از بهینگی محاسباتی [۱۲] بهتری بهره می‌برد.

[1] Open-Set
[2] Hierarchy
[3] Coarse Grained
[4] Fine Grained
[5] Inductive Bias
[6] Foundation
[7] Pre-Trained
[8] Representation
[9] Distribution Shift
[10] Sub-Population Shift
[11] Superclass
[12] Computational Efficiency

- ما از یک نگاه مولد استفاده کرده‌ایم و در قالب یک چارچوب بیزی [1] به ترکیب اطلاعات حاصل از سطوح مختلف سلسله‌مراتب می‌پردازیم.
- ما نشان داده‌ایم که راهکار پیشنهادی می‌تواند به سادگی در تنظیمات بدون یادگیری [2] به کار گرفته شود.

۱-۲ رویکرد حل مسئله

مسئله‌ی تعریف شده را در قالب یک یادگیری سلسله‌مراتبی در دو سطح مجزا تعریف می‌کنیم. در سطح اول که سطح درشت‌دانه است، ابردسته‌ها قرار دارند و در سطح دوم هم کلاس‌های ریزدانه که به آن‌ها دسته پایه [3] می‌گوییم قرار می‌گیرند. برای هر سطح یک دسته‌بند مجزا آموزش داده می‌شود و در استنتاج دسته‌بند سطح دوم، از نتایج دسته‌بند سطح اول استفاده می‌شود. در واقع کل مسئله یادگیری سلسله‌مراتبی را در قالب یک مسئله‌ی دسته‌بندی یکی از همه برنده [4] تعریف می‌کنیم. در سطح اول، در دامنه‌ی مدنظر، ابرکلاس‌هایی قرار می‌گیرند که هیچ‌گاه جمعیت آن‌ها تغییری نخواهد کرد. به عنوان مثال در یک مسئله‌ی دسته‌بندی گونه‌های جانوری، با یک دسته‌بندی بزرگ مقیاس مواجه هستیم. در سطح اول دسته‌هایی مانند پرندگان و خزندگان و ... قرار می‌گیرند. در حالی که در سطح دوم ریزدانه‌ترین گونه‌هایی که برای حیوانات متصور هستیم قرار می‌گیرند. با توجه به نمونه محدود بودن مسئله، وقتی آن را در این قالب بازتعریف می‌کنیم، برای دسته‌های سطح اول به تعداد قابل توجهی نمونه آموزش تجمیع می‌شود. چرا که نمونه‌های آموزشی هر ابردسته به اندازه‌ی دسته‌های پایه فرزند آن ضربدر تعداد نمونه‌های آموزشی هر کدام از آن کلاس‌های ریزدانه است.

۱.۲-۱ معماری مدل سطح درشت‌دانه

برای سطح اول، یک دسته‌بند تمایزگذار در نظر گرفته‌ایم که علاوه بر آموزش نظارت شده، به صورت نیمه نظارتی هم تنظیم بهینه [5] می‌شود. علت این انتخاب ثابت بودن ابردسته‌ها در یک دامنه است. در این صورت دسته‌ای به آن‌ها به صورت مستمر اضافه نمی‌شود و به همین دلیل نیاز به باز آموزش یک شبکه‌ی تمایزگذار یا استفاده از رویکرد مولد نیست. از طرفی، با تجمیع نمونه‌های آموزشی دسته‌های پایه و عملکرد بهتر دسته‌بند تمایزگذار نسبت به مولد به صورت کلی، این انتخاب صورت گرفته است.

۲.۲-۱ معماری مدل سطح ریزدانه

برای سطح دوم، از یک رویکرد متایادگیری به نام شبکه‌های پروتوتیپیکال [۳۰] استفاده کرده‌ایم. این شبکه یک دسته‌بند مولد است که می‌تواند مشکلات ناشی از اضافه شدن کلاس‌های ریزدانه و مجموعه‌باز بودن را حل کند. از طرفی از خانواده‌ی الگوریتم‌های متایادگیری است که که برای تنظیمات نمونه محدود مناسب است و دارای

[1] Bayesian
[2] Free-Training
[3] Base Class
[4] Winner Takes All
[5] Fine-Tune

ویژگی‌هایی مانند تطبیق سریع، تعمیم‌پذیری، کشف ساختارهای مشترک و اساسی است. شبکه‌ی پروتوتیپیکال یک دسته‌بند ناپارامتری [1] است که مرحله‌ی یادگیری سریع [2] در الگوریتم‌های متایادگیری را بدون آپدیت خاصی و تنها با محاسبه‌ی پروتوتایپ‌ها بدست می‌آورد. به همین دلیل از لحاظ محاسباتی کارآمد است و نیازی به نگه داشتن گراف محاسباتی [3] سنگین و محاسبه‌ی مشتقات مراتب بالاتر و ماتریس هسین [4] ندارد. در سطح دوم، شبکه‌ی پروتوتیپیکال را به صورت مشروط [5] روی پیش‌بینی شبکه‌ی سطح اول آموزش می‌دهیم و به صورت مشروط از آن استنتاج می‌گیریم.

۳.۲-۱ تنظیمات متاآموزش و متاتست

تنظیمات آموزش و استنتاج [6] را به صورت مراحل متاآموزش [7] و متاآزمون [8] می‌شکنیم. در مسائل بزرگ مقیاس، تغییر زیر جمعیت متداول است. یعنی همانطور که گفته شد، در سطح ابردسته‌ها کلاس‌ها ثابت می‌مانند اما در سطح دسته‌های پایه، دسته‌ها کاملا بدیع [9] و مجزا [10] از دسته‌های پایه‌ی اولیه می‌شوند. این تنظیمات برای به‌دست آمدن مقاومت [11] نسبت به مجموعه‌باز بودن و پیدایش دسته‌های پایه جدید در نظر گرفته شده‌است. به این ترتیب، در دو فاز متاآموزش و متاتست، در سطح ابردسته‌ها، دسته‌هایی کاملا یکسان وجود دارد اما در سطح ریزدانه و دسته‌های پایه، دسته‌ها کاملا مجزا و متفاوت از هم هستند.

برای آموزش سطح اول، با توجه به داده‌های موجود در فاز متاآموزش، یک دسته‌بند به صورت یادگیری با ناظر آموزش داده می‌شود. در واقع در مرحله‌ی متاآموزش، باید برای هر داده برچسب ریزدانه و درشت‌دانه‌ی آن را بدانیم تا بتوانیم دسته‌بند ابرکلاس‌ها را به صورت با ناظر آموزش دهیم. اما در فاز متاتست، برای داده‌های کلاس‌های بدیع، هیچ اجباری روی دانستن برچسب درشت‌دانه داده‌ها نداریم و از این برچسب هیچ استفاده‌ای نمی‌کنیم. اما با توجه به اینکه تعداد محدود نمونه‌ی پشتیبان [12] هر کلاس پایه را در اختیار داریم، از آن‌ها برای تنظیم بهینه دسته‌بند سطح اول می‌توانیم استفاده کنیم. چرا که به صورت ضمنی وقتی یک تعدادی داده متعلق به یک کلاس ریزدانه هستند، حتما متعلق به یک کلاس درشت‌دانه هم هستند اما برچسب اصلی ابرکلاس آن‌ها را نمی‌دانیم. به همین دلیل با یک روش نیمه نظارتی به تنظیم بهینه وزن‌های دسته‌بند سطح اول در مرحله‌ی متاتست می‌پردازیم.

نکته‌ی حائز اهمیت این است که در معماری دسته‌بندهای هر دو سطح، از یک شبکه‌ی پایه [13] پیش آموزش دیده از مدل‌های بنیادی استفاده می‌کنیم که به استفاده از دانش و بازنمایی بیانگر [14] و داده کارآمدی مناسب در عین توان محاسباتی پایین در مرحله‌ی آموزش شبکه‌ها برسیم.

[1] Non-Parametric
[2] Fast Learning
[3] Computational Graph
[4] Hessian
[5] Conditioned
[6] Inference
[7] Meta Train
[8] Meta Test
[9] Novel
[10] Disjoint
[11] Robustness
[12] Support
[13] Backbone
[14] Expressive

در آموزش دسته‌بند سطح دوم، در فاز متاآموزش، یک شبکه‌ی کاملا متصل [1] را به صورت ایپزودیک و با ساختن وظیفه‌های متعدد مشابه روش پروتوتیپیکال آموزش می‌دهیم. ورودی به شبکه، حاوی پیش‌بینی احتمال تعلق داده‌ی فعلی به هر یک از ابردسته‌ها است. یعنی در هنگام آموزش، شبکه‌ی سطح دوم روی پیش‌بینی سطح اول مشروط می‌شود. در فاز متاآزمون، در یک وظیفه که به اندازه‌ی دسته‌های ریزدانه راه [2] دارد، از این سطح به صورت مشروط بر تخمین سطح اول استنتاج می‌گیریم. خروجی این سطح رویکرد مولد دارد و در واقع به شرط هر کدام از کلاس‌های ریزدانه، احتمال وقوع داده‌ی فعلی را خروجی می‌دهد. به همین دلیل است که بدون آموزش مجدد در فاز متاآزمون و با وجود پیدایش کلاس‌های کاملا بدیع می‌توانیم از آن استنتاج بگیریم.

در نهایت در یک تنظیمات [3] بدون آموزش [4] آزمایش‌هایی انجام داده‌ایم که نشان داده شده است از این چارچوب ارائه شده می‌توان بدون هیچ آموزشی برای دسته‌بند سطح دوم در فاز متاآزمون و تنها با استفاده از یک شبکه‌ی پایه پیش‌آموزش داده شده هم استفاده کرد.

۱-۳ چالش‌ها

در این بخش چالش‌های اصلی که الگوریتم‌های یادگیری مستمر با آن‌ها درگیر هستند را معرفی می‌کنیم.

۱-۳-۱ محدودیت تعداد کلاس در روش‌های نمونه محدود

در روش‌های نمونه محدود متداول، مثل روش MAML [۳۱] یا SNAIL [۵] بر روی تعداد دسته‌های هر وظیفه محدودیت وجود دارد. در این روش‌ها، حداکثر دسته‌هایی که برای یک وظیفه در نظر می‌گیرند حداکثر حدود ۲۰ کلاس است. بنابراین در مسئله‌ی تعریف شده که نیاز به دسته‌بندی نمونه محدود بین چند هزار دسته هست، نمی‌توان از این روش‌ها به صورت مستقیم استفاده کرد.

۱-۳-۲ مجموعه‌باز بودن دسته‌بندی بزرگ مقیاس

وقتی مسئله دارای تعداد زیادی کلاس است، به صورت ذاتی قابلیت اضافه شدن دسته‌های جدید وجود دارد. از طرفی یکی از تنظیمات متداول در مسائل بزرگ مقیاس، تغییر زیر جمعیت [5] است. این موضوع با تغییر توزیع متداول کاملا متفاوت است. در تغییر توزیع، مثلا از حیطه‌ی تصاویر طبیعی به تصاویر طراحی یا نقاشی شده می‌رویم. اما در تغییر زیر توزیع، در همان توزیع هدف، با یک سری دسته‌ی دیگر روبرو می‌شویم که دقیقا از همان توزیع می‌آیند و تغییرات کمی دارند. بنابراین در اینگونه مسائل باید یک رویکرد مولد در نظر گرفت.

[1] Fully Connected
[2] Way
[3] Setting
[4] Free Training
[5] Sub Population Shift

۱-۳.۳ انتشار خطا در سلسله‌مراتب

وقتی از سلسله‌مراتب استفاده می‌کنیم، در پیدا کردن تعداد سطوح بهینه دچار یک مصالحه [۱] هستیم. چرا که با بالارفتن سطوح سلسله‌مراتب، خطای یک سطح به تمامی سطوح پایینی انتشار پیدا می‌کند. یعنی در واقع در مسیر رسیدن به یک برگ، هر کدام از دسته‌بند های سطوح بالایی اشتباه کنند، پیش‌بینی دسته‌بندهای دیگر هر چه باشد غلط است چون مسیر استنتاج در مسیر غلطی می‌افتد. این مشکل وقتی نمود پیدا می‌کند که در هنگام استفاده از روش یکی از همه برنده، دسته‌بند به ازای هر گره داشته باشیم. برای حل این مشکل راهکاری که ما اتخاذ کردیم، استفاده از دسته‌بند در هر سطح است. یعنی به جای اینکه نگاه کنیم اکنون در کدام گره هستیم و فقط بین فرزندان آن گره دسته‌بندی انجام دهیم، به ازای هر سطح، یک دسته‌بند داریم که بین گره‌های موجود در آن سطح دسته‌بندی می‌کند. حال برای ترکیب اطلاعات حاصل از دسته‌بندی سطح قبلی، با یک دیدگاه بیزی به ترکیب این اطلاعات می‌پردازیم که در واقع مشروط شدن تصمیم‌گیری دسته‌بند هر سطح براساس خروجی دسته‌بند سطح قبلی است. با این اوصاف، به این نتیجه رسیدیم که تعداد بهینه سطوح برای رسیدن به بالاترین دقت، ۲ سطح است. این انتخاب در کنار در نظر گرفتن جوانب دیگر مثل ثابت بودن ابردسته‌ها در عین داشتن تغییر زیرجمعیت بوده است. یعنی برای سطوح بالایی، که می‌خواهیم دیتا به اندازه‌ی کافی تجمیع شود و به صورت تمایزگذار آموزش داده شود، یک سطح ابردسته کافی است.

از طرف دیگر، تمییز دادن کلاس‌های مختلف در فضای ویژگی آموخته شده با صدها دسته با نمونه‌های اندک، ما را به سمتی برد که از سلسله‌مراتب استفاده کنیم. چرا که استفاده از سلسله‌مراتب، تعداد بالای دسته را به سمت مسائل دسته‌بندی متوازن‌تر با مرتبه‌ی لگاریتمی می‌برد.

۱-۴ اهمیت و کاربرد

اهمیت رسیدن به یک دسته‌بند بزرگ مقیاس، نمونه محدود را از دو جنبه می‌توان بررسی کرد. در نگاه اول، می‌توان به مسائلی که نیاز به همچنین دسته‌بندی دارند نگاه کرد. معمولاً این مسائل با اشیا خاص‌تری برخورد می‌کنند که تنوع کمی در هر کلاس وجود دارد. دسته‌بندی‌های ظریف اشیاء به طور طبیعی می‌توانند در خوشه‌های بزرگ‌تری گروه‌بندی شوند و ابردسته‌های درشت‌دانه‌تری را با تنوع بیشتر اشیا تشکیل دهند. این ابردسته‌ها همچنین می‌توانند به صورت بازگشتی گروه‌بندی شوند و سلسله‌مراتبی از کلاس‌ها را در یک ساختار درختی‌شکل ایجاد کنند. این ساختار سلسله‌مراتبی در مجموعه داده‌های بزرگ مقیاس فراهم شده است و می‌تواند به عنوان یک سوگیری القایی اضافی برای بهبود عملکرد مسائل طبقه‌بندی با تعداد نمونه کم در مقیاس بزرگ مورد استفاده قرار گیرد. با در نظر گرفتن این ماهیت سلسله‌مراتبی دسته‌بندی اشیا، انسان‌ها می‌توانند یک شی را با درنظر گرفتن سطح ریزدانگی متفاوتی متناسب با سطوح مختلف این سلسله‌مراتب طبقه‌بندی کنند. از جمله‌ی این مسائل می‌توان به موارد زیر اشاره کرد.

- **دسته‌بندی جانوران**
- **دسته‌بندی گیاهان**

[۱] Trade Off

- **دسته‌بندی اقلام در فروشگاه‌های زنجیره‌ای**
- **دسته‌بندی صفحات وب**

اما در نگاه دیگر، اهمیت کلی یک دسته‌بند بزرگ مقیاس نمونه محدود را می‌توان مورد بررسی قرار داد. رسیدن به این مدل معادل با دستیابی به یک مدل بنیادین پایه همه‌منظوره [1] است. چرا که قابلیت تعمیم به دامنه‌ی دسته‌های بدیع را دارد و از طرفی با تعداد نمونه‌ی محدود می‌تواند به آن دامنه تطبیق پیدا کند. با داشتن یک دسته‌بند بزرگ مقیاس که قابلیت تعمیم‌پذیری برای دسته‌های بدیع دارد، یک مدل بنیادین در حوزه‌ی بینایی کامپیوتر داریم که قادر به دسته‌بندی هر تصویری در هر دامنه‌ای است.

۱-۵ هدف پژوهش

مسئله شناسایی بصری اشیا طی دهه گذشته به عملکرد انسانی رسیده است و حتی از آن پیشی گرفته‌است. با این حال، راه حل‌های سنتی نیازمند مجموعه داده‌های بزرگ با صدها نمونه برای هر شی هستند. علاوه بر این، توانایی تعمیم این روش‌ها محدود است و آن‌ها نمی‌توانند به راحتی به کلاس‌های دیده نشده اشیا تعمیم یابند. در مقابل، انسان‌ها می‌توانند با تعداد نمونه‌های بسیار کمتری در مورد اشیا جدید یاد بگیرند و تعمیم دهند. این چالش تا حدودی در پارادایم یادگیری با تعداد نمونه کم با استفاده از مکانیزم‌های انتقال دانش مانند متایادگیری مورد توجه قرار گرفته است. اکثر روش‌های موجود یادگیری با تعداد نمونه کم، بر مسائلی با تعداد محدودی دسته هدف متمرکز بوده‌اند. با این حال، اخیراً علاقه فزاینده‌ای به یادگیری با تعداد نمونه کم در مقیاس بزرگ وجود دارد [۱۵، ۳۲، ۳۳] که شرایط واقع‌گرایانه‌تری با تعداد زیادی دسته‌ی هدف و تعداد کم نمونه‌ها را پوشش می‌دهد. این چارچوب می‌تواند چالش‌برانگیز باشد زیرا تمییز دادن دسته‌های مختلف در فضای ویژگی آموخته شده دشوار می‌شود وقتی صدها دسته از اشیا با تعداد نمونه کم وجود دارد.

با توجه به مباحث ذکر شده، در این پژوهش ما به دنبال ارائه یک روش یادگیری هستیم که چالش‌های مطرح شده در دسته‌بندی بزرگ مقیاس نمونه محدود را پاسخ دهد و به عملکرد مطلوبی برسد. با الهام از سیستم شناختی انسان، یک رویکرد بیزی برای ادغام اطلاعات ابرکلاس به عنوان یک احتمال پیشین [2] برای استنتاج دقیق‌تر در سطح دسته پایه توسعه داده‌ایم.

۱-۶ کد پیاده‌سازی و منابع جانبی

مخزن کد پیاده‌سازی این پژوهش از طریق این پیوند [3] قابل دسترسی است. همچنین نتایج این پژوهش در قالب یک مقاله‌ی ورکشاپی در کنفرانس NeurIPS ۲۰۲۳ پذیرفته و به چاپ رسیده‌است. این مقاله به اسم "Divide and Conquer: Two-Level Problem Remodeling for Large-Scale Few-Shot Learning" از طریق این لینک قابل مطالعه است.

[1] Versatile
[2] Prior
[3] https://github.com/mohamadreza99/divide_and_conquer

۱-۷ ساختار پایان‌نامه

در ادامه‌ی مطالب بیان شده، در فصل بعدی به تشریح کارهای پیشین در زمینه‌ی یادگیری نمونه محدود، متایادگیری [۱] ، یادگیری تحت شرایط مجموعه باز بودن و یادگیری سلسله‌مراتبی نمونه محدود پرداخته‌ایم. به این منظور به توضیح رویکرد کلی هر کدام از این موارد در قالب مقالات مهم آن حوزه اشاره شده‌است. در فصل سوم، راهکار پیشنهادی خود برای مسئله‌ی دسته‌بندی بزرگ مقیاس نمونه محدود را پیشنهاد داده‌ایم که تلاش می‌کند تمامی چالش‌های موجود در این مسئله را پوشش دهد. در فصل چهارم، به توضیح جزییات مجموعه‌داده‌ها و معماری مدل‌ها و همچنین بیان نتایج آزمایش‌ها و مقایسه‌ی آن‌ها با مدل‌های دیگر پرداخته‌ایم. در نهایت در فصل آخر، به جمع‌بندی و بیان کارهای آتی پرداخته‌ایم.

[1] Meta Learning

فصل ۲

پژوهش‌های پیشین

مطالب این فصل

۲-۱ یادگیری نمونه محدود با داده‌افزایی ۱۱
۲-۲ متایادگیری ۱۴
۲-۳ استفاده از بدنه [1] بیانگر [2] برای رسیدن به کارایی نمونه [3] ۲۲
۲-۴ خاصیت مجموعه باز بودن ۲۲
۲-۵ یادگیری سلسله‌مراتبی نمونه محدود ۲۷
۲-۶ جمع‌بندی ۳۰

در این فصل ابتدا به مرور کارهای مهم در زمینه یادگیری نمونه محدود با داده‌افزایی پرداخته، سپس پژوهش‌ها و رویکردهای کلی در زمینه‌ی متایادگیری را معرفی کرده و بعد از آن کارهای پیشین در حوزه‌های استفاده از بدنه‌های پیش‌آموزش دیده برای رسیدن به ویژگی‌های بیانگر، مقالات حوزه‌ی یادگیری در شرایط مجموعه باز بودن و یادگیری سلسله‌مراتبی نمونه محدود بررسی شده‌اند.

۲-۱ یادگیری نمونه محدود با داده‌افزایی

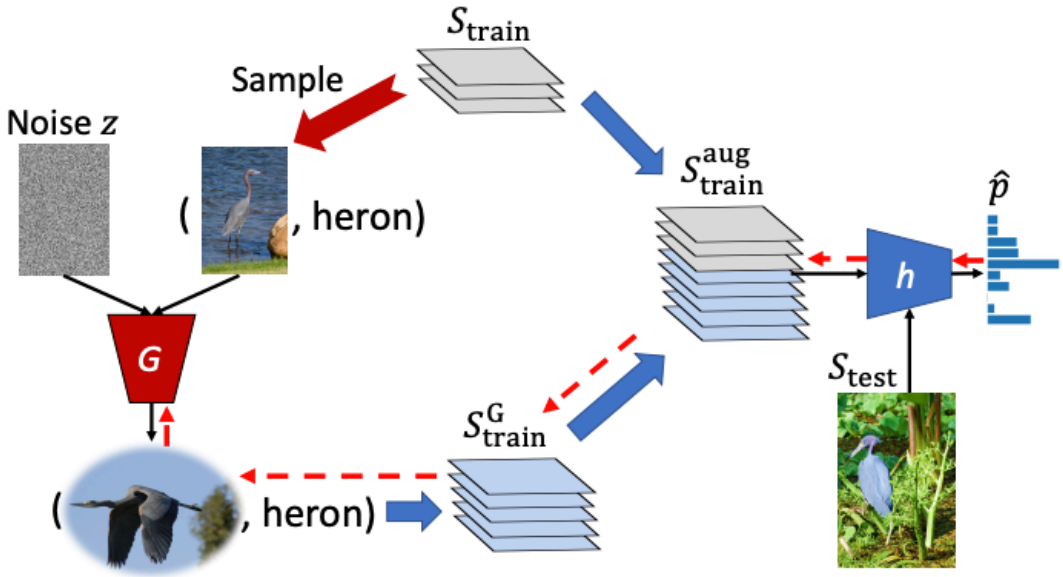
به طور کلی، داده‌افزایی در دو سطح بردار ویژگی‌ها و در سطح نمونه‌ها تعریف می‌شود. هدف داده‌افزایی تولید و گسترش مجموعه‌ی دادگان است به طوری که از توزیع کلی داده‌های آموزش خارج نشویم. چالش‌هایی مثل یافتن تبدیل‌های مناسب برای تولید نمونه‌های معنادار و داشتن سوگیری نمونه‌های تولید شده روی مجموعه‌ی داده‌های آموزش اولیه، از مشکلات این روش است.

[1] Backbone
[2] Expressive
[3] Sample Efficiency

۲-۱.۱ داده‌افزایی در سطح بردار ویژگی

در این دسته، سعی می‌شود تبدیل‌هایی روی بازنمایی‌های بدست آمده از یک شبکه‌ی عمیق اعمال شود تا به بازنمایی داده‌هایی معنادار از همان دسته برسیم. یکی از روش‌های مورد استفاده، داده‌افزایی به وسیله‌ی خیال‌بافی [۱] است [۱۸] . در این روش، تبدیل‌های انتزاعی که حاصل تغییرات بین نمونه‌های متعلق به یک دسته‌ی واحد است، از روی بردار ویژگی‌های آن‌ها محاسبه می‌شود. این تغییرات بدست آمده، ممکن است منحصر به یک دسته‌ی خاص باشد. برای مقابله با این موضوع، بین تمامی تغییرات درون دسته‌ای، یک معیار شباهت محاسبه می‌شود و متداول‌ترین آن‌ها انتخاب می‌شود.

به عنوان مثال، اگر در فضای ویژگی‌ها، بازنمایی دو نمونه مربوط به دسته‌ی اول را داشته باشیم و تغییرات بین آن‌ها را به صورت $z_1^a - z_2^a$ محاسبه کنیم، اگر همین تغییرات را برای دو نمونه از یک دسته‌ی دیگر به صورت $z_1^b - z_2^b$ محاسبه کنیم، در صورتی این تفاضل به تبدیل‌های قابل اعمال اضافه می‌شود که شباهت کسینوسی این دو تفاضل بالا باشد. در نتیجه تبدیل‌هایی ذخیره می‌شوند که مستقل از دسته باشند. ویژگی کلی داده‌افزایی در سطح ویژگی، داشتن امکان تبدیلات پیچیده‌تر نسبت به فضای تصویر است. به عنوان مثال، در این حالت ممکن است تبدیلی را اعمال کنیم که بازنمایی یک پرنده‌ی ساکن روی درخت را به بازنمایی یک پرنده در حال پرواز مبدل سازد.



شکل ۲-۱: قسمت خیال‌باف با G و دسته‌بند با h نشان داده شده است. برای یک اپیزود متایادگیری، به نمونه‌های پشتیبان هر دسته تعدادی داده اضافه می‌شود. مسیرهایی که گردیان از آن‌ها برمی‌گردد با نقطه‌چین قرمز مشخص شده‌اند [۱].

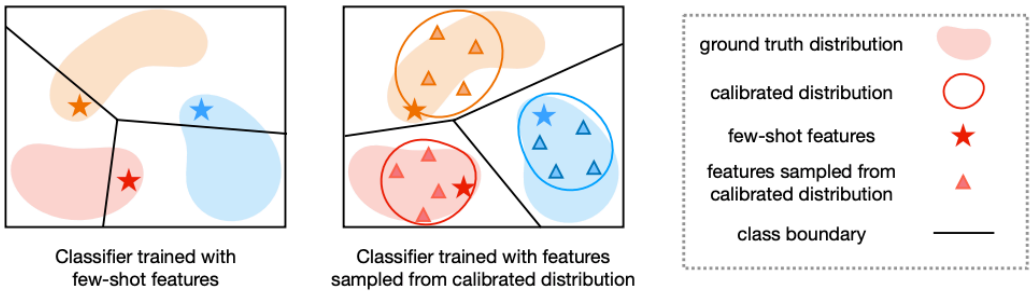
مقاله‌ی دیگری با الهام از انسان می‌خواهد وقتی یک وضعیت از یک شی را می‌بیند، بتواند آن تصویر را از زاویه‌های دیگر تصور کند و با دیدن یک نمونه، به توانایی شناخت برسد. در واقع به دنبال یادگیری مفاهیم است تا بتواند تبدیل‌های صحیح را پیاده کند. البته بیان می‌کند هر تبدیل معناداری ممکن است به بهتر شدن دقت دسته‌بندی شبکه‌ی عمیق کمک نکند. به همین دلیل، به دنبال تبدیل‌هایی است که تغییراتی به وجود آورند تا دقت مدل بیشتر شود [۱] .

[1] Hallucination

با روندی که در شکل ۲-۱ مشخص شده، پیشنهاد داده‌اند که یک قسمت خیال‌باف [1] با الگوریتم متایادگیری ترکیب شود و هر دو به صورت توأمان آموزش ببینند. به این منظور، از یک شبکه‌ی G استفاده می‌کند که با ورودی گرفتن یک نمونه از داده‌های پشتیبان و یک نویز تصادفی، به صورت قطعی [2] یک داده‌ی جدید تولید کند. این داده‌های تولیدی را با نمونه‌های پشتیبان یک وظیفه ترکیب می‌کند و به وسیله‌ی هر دوی این‌ها یک طبقه‌بند h را آموزش می‌دهد. گرادیان تابع هزینه‌ی این دسته‌بند به G هم می‌رسد و به این ترتیب قسمت خیال‌باف یاد می‌گیرد تابع قطعی‌ای را تخمین بزند که به بهبود دقت دسته‌بند کمک کند.

در کار یانگ و همکارانش، مشاهده شده آماره‌های دسته‌هایی که از لحاظ بصری به هم شبیه‌اند، مشابهت زیادی دارند [۲] . این روش روی مدل‌های مبتنی بر متر پیشنهاد شده است که برای هر دسته در فضا یک نماینده در نظر می‌گیرند و روی آن توزیعی متصور هستند. آن‌ها مشاهده کرده‌اند که آماره‌هایی مثل میانگین و پراکندگی توزیع دو دسته مثل روباه و گرگ که از لحاظ بصری به هم شبیه‌اند، حدود ۹۷ درصد مشابه است. در حالی که این حد از مشابهت با آماره‌های دسته‌های دیگر دیده نشد.

آن‌ها فرض کرده اند که فضای بردار ویژگی‌ها حول هر دسته از توزیع گاوسی [3] می‌آید. در نتیجه، به تعداد محدودی دسته‌ی با تعداد نمونه‌ی بالا را جمع‌اوری می‌کنند و آماره‌های مربوط به آن را حساب می‌کنند. سپس برای هر دسته‌ی نمونه محدود، نزدیک‌ترین دسته‌ها با تعداد نمونه‌ی بالا را پیدا می‌کنند. به دلیل مشابهتی که آماره‌های این دو دسته دارد، برای کلاس نمونه محدود آماره‌اش را تنظیم [4] می‌کنند و از آن توزیع نمونه‌برداری می‌کنند تا داده‌های بیشتری به مجموعه‌ی آموزش اضافه شود. روند این کار در شکل ۲-۲ مشخص است.



شکل ۲-۲: تنظیم توزیع به وسیله‌ی نمونه‌برداری از توزیع‌های تخمین‌زده؛ ابتدا برای ویژگی‌های دسته‌های نمونه محدود، نزدیک ترین دسته‌ها که تعداد کافی نمونه دارند پیدا می‌شود و سپس با تنظیم پارامترهای توزیع مفروض، از توزیع جدید به‌دست آمده برای هر دسته نمونه‌های جدید برداشته می‌شود که با مثلث نشان داده شده‌است [۲].

۲-۱.۲ داده‌افزایی در سطح نمونه

منظور از داده‌افزایی در سطح نمونه اعمال تبدیل‌هایی در فضای تصویرهای ورودی به شبکه‌ی عمیق است [۱۷] . به این منظور از تبدیل‌هایی نظیر تغییرات در فضای رنگ، پاک کردن تصادفی بخشی از تصویر، ترکیب بخش‌های

[1] Hallucinator
[2] Deterministic
[3] Gaussian
[4] Calibrate

مختلف از دو تصویر، چرخش تصویر، انتقال سبک با شبکه‌های ژرف [1] و تولید داده با مدل‌های مولد برای داده‌افزایی استفاده می‌شود [۳۴–۳۶] .

۲-۲ متایادگیری

اولین واژه‌ای که برای متایادگیری میتوان متصور شد، یادگیری برای یادگیری است. اگر روی یادگیری انسانها تمرکز کنیم، نوعی استفاده از تجربیات پیشین برای یادگیری یک وظیفه‌ی جدید در رفتار آن‌ها دیده می‌شود. این مسئله یادگیری را برای انسان بسیار داده کارآمد می‌کند. در واقع انسان‌ها به عنوان موجوداتی هوشمند، از تجربیات پیشین در یادگیری وظیفه‌ها به نوعی استفاده می‌کنند که با کشف ساختارهای مشترک [2] ، می‌توانند عملکرد بهینه در یادگیری وظیفه‌ی جدید داشته باشند.

انسان برای یکبار قوانین فیزیک حاکم بر محیط اطرافش را یاد میگیرد و هر بار در انجام یک فعالیت فیزیکی الگویی مشابه را می‌یابد، با الهام گرفتن از تجربیات پیشین، آن مسئله را حل میکند. وقتی یک فرد ورزشی مثل والیبال را تمرین می‌کند، در ورزشی مانند بسکتبال، در مقایسه با فردی که تجربه‌ی والیبال بازی کردن را ندارد، معمولا سریع‌تر یاد می‌گیرد. این به دلیل یادگرفتن چگونگی یادگیری در شرایطی با ساختار مشابه توسط انسان است. در شبکه‌های عصبی، معنای ساده‌ی یادگیری این است که پارامترهای بهینه [3] برای حل یک وظیفه مشخص شود. اما وقتی میخواهیم یادگیری برای یادگیری انجام دهیم، یعنی شبکه توانایی چگونگی بهینه یاد گرفتن را پیدا کند. در این صورت ویژگی‌هایی مانند داده کارآمد بودن، تطبیق سریع [4] و تعمیم‌پذیری بالا به شبکه اضافه می‌شود.

ما در مسئله‌ی متایادگیری، با دسته‌ای از مسائل یادگیری مواجه هستیم که هر کدام از آن‌ها را یک وظیفه می‌نامیم و می‌خواهیم در طول یادگرفتن وظایف مختلف، خود فرآیند یادگیری را بهتر کنیم. در زیر مجموعه وظایف یادگیری در قالب $\mathcal{D}_{meta}$ نشان داده شده است.

$$\mathcal{D}_{meta} = \{\mathcal{T}^{(1)}, \mathcal{T}^{(2)}, ..., \mathcal{T}^{(n)}\} \tag{۱.۲}$$

هر وظیفه‌ی یادگیری برای هر دسته، داده‌های آموزش دارد که به آن‌ها داده‌های پشتیبان می‌گوییم و با $\mathcal{S}$ نشان می‌دهیم. همچنین به داده‌های آزمون، داده‌ی پرسش می‌گوییم و با $\mathcal{Q}$ نمایش می‌دهیم.

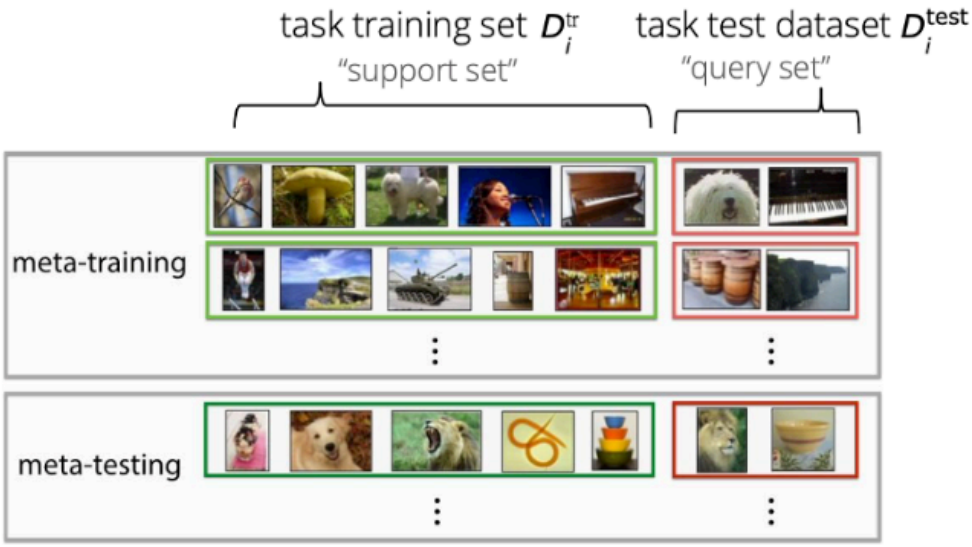
$$\mathcal{T}^{(i)} = \{S^{(i)}, Q^{(i)}\} \tag{۲.۲}$$

در مسائل متایادگیری، دو دسته پارامتر وجود دارد. پارامترهای θ که به آن‌ها متاپارامتر می‌گوییم و پارامترهای کلی مدل هستند و برای تمام وظایف و مسائل در $\mathcal{D}_{meta}$ مشترک‌اند. دسته‌ی دیگر پارامترهای خاص وظیفه [5] هستند و همانطور که از اسم آن پیداست، مختص هر یک از وظایف هستند. این دسته از پارامترها را معمولا با ϕ در ادبیات متایادگیری نمایش می‌دهند.

[1] Neural Style Transfer
[2] Shared Structures
[3] Optimum
[4] Early Adaptation
[5] Task Specific

در واقع متاپارامترها همان پارامترهایی هستند که تجربیات مشترک حاصل از دیدن مسائل و وظایف مختلف یادگیری را در خود جای داده‌اند. با یادگرفتن متاپارامترهاست که فرآیندهای یادگیری جدید بهینه می‌شود و وقتی متاپارامترها آموزش داده می‌شوند، مدل یاد می‌گیرد که چگونه یاد بگیرد. اما ϕ در قالب یکی از وظیفه‌ها یا همان یکی از مسائل یادگیری تعریف می‌شود.

در واقع پارامترهای خاص وظیفه، برای یک وظیفه‌ی یادگیری i ، از روی $S^{(i)}$ و با کمک و استفاده از اطلاعات متاپارامترها، یعنی θ یاد گرفته می‌شوند. بعد از اینکه ϕ یاد گرفته شد، به سراغ داده‌های آزمون یا پرسش یک وظیفه یعنی $Q^{(i)}$ می‌رویم و با استفاده از آن‌ها، از مدل استنتاج می‌گیریم و به وسیله‌ی تعریف یک تابع هزینه، خطایی که بدست می‌آید را برای به روزرسانی متاپارامترها استفاده می‌کنیم. با تکرار این روند در طول وظیفه‌های مختلف، متاپارامترهای θ به مقدار مناسبی همگرا می‌شوند به نحوی که از روی آن‌ها و داده‌های پشتیبان، با سرعت بالا و با دیدن نمونه‌های کم می‌توان پارامترهای خاص وظیفه را بدست آورد. به فرآیند یادگیری متاپارامترها که در طول وظایف مختلف صورت می‌گیرد، حلقه‌ی بیرونی [1] یادگیری و به مرحله‌ی بهینه‌سازی برای بدست آمدن پارامترهای خاص یک وظیفه، حلقه‌ی درونی [2] یادگیری یا مرحله‌ی یادگیری سریع [3] می‌گویند. در شکل ۲-۳ کلیت این روش آمده است.



شکل ۲-۳: آموزش به شیوه‌ی متایادگیری؛ هر سطر یک اپیزود یا وظیفه است که یک مسئله یادگیری مستقل است. کادرهای سبز نمونه‌های پشتیبان و کادرهای قرمز نمونه‌های پرسش است و در طول یک سطر یادگیری سریع یا حلقه‌ی داخلی انجام می‌شود. در طول سطرها وظایف مختلف یاد گرفته می‌شوند و به وسیله‌ی آن‌ها متاپارامترها یادگرفته می‌شوند [۳].

همانطور که در یک مسئله‌ی یادگیری، برای سنجیدن کیفیت پارامترهای یادگرفته شده، از مرحله‌ی آزمون و مجموعه‌ی دادگان آزمون استفاده می‌کنیم، برای سنجیدن کیفیت متاپارامترهای یادگرفته شده هم از یک فاز متاآزمون مطابق شکل ۲-۳ استفاده می‌کنیم.

[1] Outer Loop
[2] Inner Loop
[3] Fast Learning

برای این منظور، وظایف جدیدی می‌سازیم که دسته‌های موجود در آن‌ها، در مرحله‌ی قبلی یعنی متایادگیری وجود نداشته باشند. یعنی دسته‌های موجود در مرحله‌ی متاآزمون با دسته‌های موجود در مرحله‌ی متایادگیری ناسازگار [1] اند (برای یک وظیفه در مرحله‌ی متاآزمون $\mathcal{T} \notin \mathcal{D}_{meta}$) . به طور خلاصه، الگوریتم متایادگیری از دو مرحله تشکیل شده است. در مرحله‌ی متایادگیری مدل به دنبال بهینه‌سازی و یادگیری متاپارامترها θ است و می‌خواهد تجربیات حاصل از مسائل این مرحله را در خود جای دهد. این دانش پیشین در مرحله‌ی متاآزمون مورد بهره‌برداری قرار می‌گیرد و به کمک آن مدل باید پارامترهای خاص هر وظیفه‌ی جدید را با دیدن نمونه‌های محدود به خوبی بدست آورد. در ادامه، ابتدا نمادگذاری دقیق‌تر مسئله‌ی متایادگیری را بیان می‌کنیم و سپس به معرفی تقسیم‌بندی مسائل در این حیطه می‌پردازیم [۳۷] .

۲-۱.۲ ساختار کلی الگوریتم متایادگیری

معمولا الگوریتم یادگیری که برای یادگیری پارامترهای خاص تسک استفاده می‌شود را با $\mathcal{B}$ نشان می‌دهند. در مرحله‌ی یادگیری سریع یا حلقه‌ی داخلی، یک مدل یا به طور معادل الگوریتم یادگیری را انتخاب می‌کنیم که به آن یادگیر پایه [2] می‌گوییم. برای آموزش این یادگیر پایه، با استفاده از داده‌های پشتیبان یک وظیفه و همینطور دسترسی به متاپارامترها یک مسئله‌ی بهینه‌سازی به صورت زیر را حل می‌کنیم.

$$\phi = \mathcal{B}\left(\mathcal{S};\theta\right) = \arg\min_{\phi} \mathcal{L}^{base}\left(\mathcal{S};\theta,\phi\right) \tag{۳.۲}$$

در این معادله، منظور از $\mathcal{L}^{base}$ تابع هزینه‌ی مختص یک وظیفه است. با بدست آمدن پارامترهای ϕ که در واقع وزن‌های یادگیر پایه اند، یک مدل $\mathcal{M}$ که نشان‌دهنده‌ی متایادگیر است آموزش می‌دهیم. برای این منظور مسئله‌ی بهینه‌سازی زیر را حل می‌کنیم.

$$\theta^{\star} = \mathcal{M}(\mathcal{D}_{meta}) = \arg\min_{\theta} \mathbb{E}_{(\mathcal{S},\mathcal{Q})\in\mathcal{D}_{meta}}[\mathcal{L}^{meta}\left(\mathcal{Q};\theta,\phi\right),\ where\ \phi = \mathcal{B}(\mathcal{S};\theta)] \tag{۴.۲}$$

در معادله‌ی بالا منظور از $\mathcal{L}^{meta}$ تابع هزینه‌ی است که مختص ارزیابی متاپارامترها است. در ادبیات متایادگیری، به در محدوده‌ی یک وظیفه، به تعداد نمونه‌های پشتیبان به ازای هر دسته تعداد شات [3] و به تعداد دسته‌های موجود راه [4] می‌گویند.

در رویکرد متایادگیری، انواع پیاده‌سازی‌های متفاوت از این مفهوم وجود دارد. این پیاده‌سازی‌ها گستره‌ای از روش‌های مختلف را در بر می‌گیرد که هر کدام ویژگی‌های خاصی دارند و در ادامه به معرفی آن‌ها می‌پردازیم.

۲-۲.۲ روش‌های مبتنی بر بهینه‌سازی

در این دسته از روش‌ها در یک فرآیند بهینه‌سازی دو سطحی، در فاز متاآموزش، در هر وظیفه پارامترهای خاص آن وظیفه را یاد می‌گیریم. سپس با بدست آوردن خطای داده‌های پرسش، بهینه‌سازی روی متاپارامترها صورت

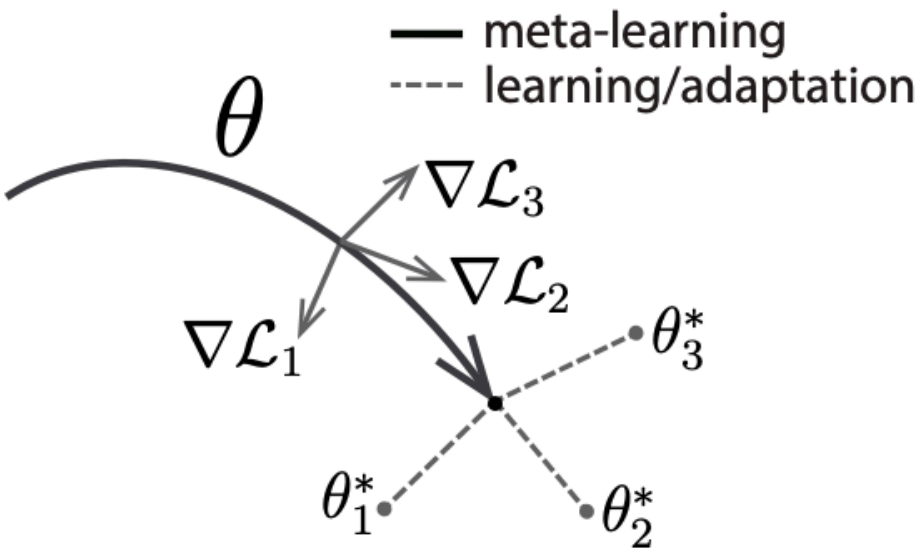
[1] Disjoint
[2] Base Learner
[3] Shot
[4] Way

میگیرد.

مقاله‌ی MAML به عنوان شاخص‌ترین روش این دسته مطرح است [۴] . تابع هدف حاصل از این روش در معادله‌ی ۵.۲ آمده است. در این معادله، $\mathcal{T}_i$ یک وظیفه‌ی نمونه‌برداری شده از توزیع مجموعه وظایف مسئله‌ی هدف است. تابع خطا با $\mathcal{L}$ مشخص شده و از α برای نمایش نرخ یادگیری استفاده شده است. همچنین $\mathcal{D}_i^{support}$ مجموعه‌ی داده‌های پشتیبان برای یک وظیفه و $\mathcal{D}_i^{query}$ مجموعه‌ی داده‌های پرسش برای وظیفه‌ی i است. همانطور که قبلا بیان شد، θ متاپارامترها را نشان می‌دهد و منظور از θ' پارامترهای خاص وظیفه یا همان ϕ است که در این روش با بهینه‌سازی و آپدیت متاپارمترها بدست می‌آید.

$$\min_{\theta} \sum_{\mathcal{T}_i \sim p(\mathcal{T})} \mathcal{L}_{\mathcal{T}_i}(f_{\theta'_i}) = \sum_{\mathcal{T}_i \sim p(\mathcal{T})} \mathcal{L}(\theta - \alpha \nabla_\theta \mathcal{L}(\theta, \mathcal{D}_i^{support}), \mathcal{D}_i^{query}) \tag{۵.۲}$$

در شکل ۲-۴ شهود این روش آمده است.



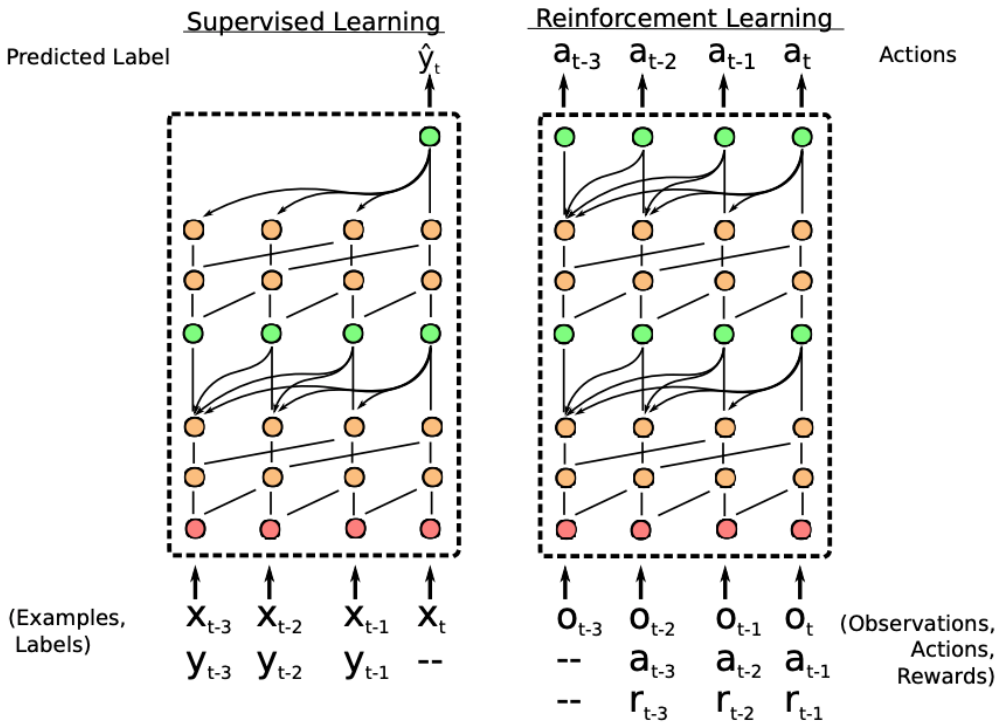
شکل ۲-۴: متایادگیری مستقل از مدل؛ در این روش، با تعداد کمی گام بهینه‌سازی روی متاپارامترها، به پارامترهای خاص هر وظیفه می‌رسیم [۴].

در واقع متاپارامترها یک نقطه‌ی شروع اولیه برای پارامترهای خاص هر وظیفه اند و با داشتن اطلاعات مشترک بین وظایف، سبب می‌شوند با تعداد کمی گام بهینه‌سازی، به پارامترهای مناسب برای حل یک وظیفه برسیم. این روش قادر به یادگیری توابع با پیچیدگی بالا است اما به دلیل نیاز به بهینه‌سازی دو سطحی [۱] و ظهور مشتق دوم، مشکلاتی از لحاظ حافظه به جهت نیاز به نگه داشتن تمامی مشتق‌های مرتبه اول و همینطور پیچیدگی فرآیند یادگیری را به دنبال دارد. برای حل این مشکلات راهکارهایی از قبیل استفاده از تقریب مرتبه‌اول مشتق‌گیری و همچنین بهینه‌سازی محدب با فرم بسته پیشنهاد شده است [۳۸، ۳۹] .

[1] Second Order Optimization

۲-۳.۲ روش‌های جعبه سیاه [1]

در این دسته از روش‌ها، یک مدل جداگانه آموزش می‌دهیم که با دریافت توصیف هر وظیفه، بخشی یا تمامی پارامترهای مدل مربوط به آن وظیفه را خروجی بدهد. در واقع یک مدل داریم با پارامترهای θ که به عنوان ورودی توصیف یک وظیفه یا همان داده‌های پشتیبان را می‌گیرد و سپس در خروجی یک آماره‌ی کافی [2] از پارامترهای خاص تسک یا همان ϕ را تولید می‌کند.



شکل ۲-۵: متایادگیری جعبه سیاه؛ در شکل سمت چپ که تنظیمات یادگیری با ناظر آمده است، می‌توان از $X_{t-۳}$ تا $X_{t-۱}$ را به همراه برچسب هر کدام به عنوان مجموعه‌ی داده‌های پشتیبان در نظر گرفت و X_t را داده‌ی پرسش در نظر گرفت. به دلیل ماهیت شبکه‌های بازگشتی، با به روز رسانی فضای نهان، به پارامترهای خاص وظیفه می‌رسیم [۵].

بخش‌های اساسی این دسته از مدل‌ها، g_ϕ یا همان مدل خاص وظیفه و f_θ یا همان مدل جعبه سیاه است که مدل خاص وظیفه را با دریافت داده‌های پشتیبان یک وظیفه می‌سازد. عملکرد این دو مدل مطابق فرمول ۶.۲ است.

$$\begin{aligned} \phi &= \mathcal{B}\left(\mathcal{S};\theta\right) = f_\theta(\mathcal{S}) \\ \hat{y} &= g_\phi(x) \end{aligned} \tag{۶.۲}$$

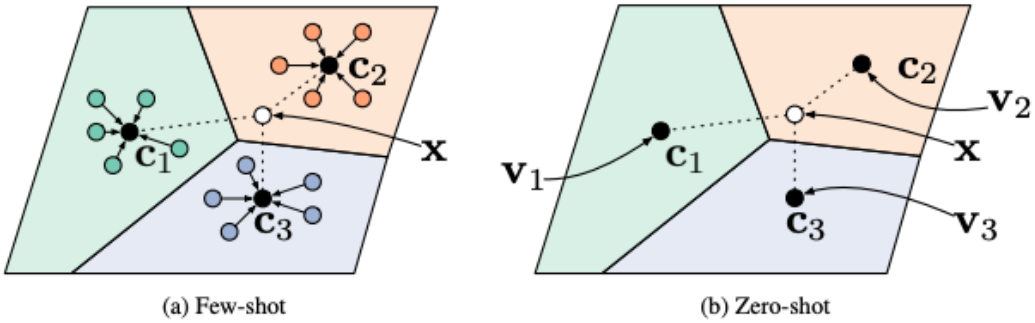
معماری‌های رایج در این دسته شامل مجموعه‌ی عمیق [3] [۴۰] ، مکانیزم توجه [4] [۴۱] و شبکه‌های بازگشتی

[1] Black Box
[2] Sufficient Statistic
[3] Deep Set
[4] Attention Mechanism

[۱] [۴۲] است. این دسته از مدل‌ها در صورت بزرگ بودن تعداد پارامترهای مدل خاص وظیفه، عملکرد مناسبی ندارند.

۲-۴.۲ روش‌های مبتنی بر متر [۲]

با توجه به پیچیدگی‌های روش دسته‌های قبل در تخمین پارامترهای خاص وظیفه، روش مبتنی بر متر یا بدون پارامتر پیشنهاد شده است. در این دسته از روش‌ها، سعی می‌شود در یک فضای معنایی، مبتنی بر یک معیار فاصله دسته‌بندی انجام پذیرد. نکته‌ی حائز اهمیت این است که مرحله‌ی متاآموزش همچنان دارای پارامتر است و در واقع در این مرحله متاپارامترها یاد گرفته می‌شوند. اما در فاز استنتاج، دسته‌بندی مبتنی بر متر و فاصله انجام میگیرد. تمامی کارهای پیشین این دسته بر پایه‌ی شبکه‌ی مبتنی بر نماینده [۳] است [۶] .



(a) Few-shot (b) Zero-shot

شکل ۲-۶: متایادگیری مبتنی بر متر؛ در این دسته از روش‌ها برای هر دسته یک مرکز از روی نمونه‌های پشتیبان هر دسته یا متادیتای آن محاسبه می‌شود و برای یک نمونه پرسش، با سنجش فاصله از مرکز دسته‌ها، طبقه‌بندی انجام می‌شود. در سمت چپ محاسبه‌ی نماینده دسته از روی نمونه‌های پشتیبان و در سمت راست محاسبه‌ی نماینده دسته از روی متادیتای دسته‌ها انجام شده است [۶].

در این روش، با همان تنظیماتی که به طور کلی برای متایادگیری تعریف کردیم، ابتدا یک شبکه‌ی نگاشت فضای ویژگی‌ها آموزش داده می‌شود. در واقع پارامترهای این شبکه در طول وظیفه‌های مختلف یاد می‌گیرند چگونه یک داده‌ی ورودی را به فضایی نگاشت کنند که در آن دوری و نزدیکی نقاط از هم معنا داشته باشد. پارامترهای این شبکه θ را تشکیل می‌دهند. در چنین فضایی، وقتی دو داده از یک دسته نگاشت می‌شوند، باید فاصله‌ی آنها نزدیک به هم باشد. به طور قرینه، اگر دو داده از دسته‌های متفاوتی باشند، باید به نقاطی دور از هم نگاشت شوند. با پیاده‌سازی این ایده، اگر برای هر دسته یک نماینده در فضای ویژگی‌ها محاسبه کنیم، در صورت دیدن یک داده‌ی مجهول، می‌توان با استفاده از شبکه‌ی آموزش دیده، نگاشت روی فضای معنایی را انجام داد و سپس با یک معیار فاصله مانند فاصله‌ی اقلیدسی، تعلق داده به نزدیکترین نماینده را مشخص کرد. در واقع متاپارامترها در این مسئله، همان پارامترهای شبکه‌ی نگاشت به فضای معنایی است.

[1] Recurrent Neural Networks
[2] Metric Based
[3] Prototype

$$
\begin{aligned}
c_k &= \frac{۱}{|S_k|} \sum_{(x_i,y_i)\in k} f_\theta(x) \\
p_\theta(y=k|x) &= \frac{\exp(-d(f_\theta(x), c_k))}{\sum_{k'} \exp(-d(f_\theta(x), c_{k'})} \\
minJ(\theta) &= -\log p_\theta(y=k|x)
\end{aligned}
\tag{۷.۲}
$$

در معادله‌ی ۷.۲ و برای دسته‌ی k که به تعداد $|S_k|$ عضو دارد، ابتدا نماینده هر دسته توسط شبکه‌ی نگاشت به فضای ویژگی f محاسبه و در c_k ذخیره شده است. سپس با یک معیار فاصله‌ی d نزدیکی یک داده‌ی ورودی با هر یک از نماینده دسته‌ها محاسبه شده است و احتمال تعلق داده به هر دسته بدست آمده است. تابع هدف این روش پیدا کردن متاپارامترهای θ به منظور کمینه‌کردن خطای منفی بیشینه‌ی درست‌نمایی [۱] تعریف می‌شود که با J نشان داده شده است.

Algorithm 1 Training episode loss computation for prototypical networks. N is the number of examples in the training set, K is the number of classes in the training set, $N_C \leq K$ is the number of classes per episode, N_S is the number of support examples per class, N_Q is the number of query examples per class. RandomSample(S, N) denotes a set of N elements chosen uniformly at random from set S, without replacement.

Input: Training set $D = \{(x_1, y_1), \ldots, (x_N, y_N)\}$, where each $y_i \in \{1, \ldots, K\}$. D_k denotes the subset of D containing all elements (x_i, y_i) such that $y_i = k$.

Output: The loss J for a randomly generated training episode.

```
V ← RandomSample({1, ..., K}, N_C)                    //Select class indices for episode
for k in {1, ..., N_C} do
    S_k ← RandomSample(D_{V_k}, N_S)                  // Select support examples
    Q_k ← RandomSample(D_{V_k} \ S_k, N_Q)            //Select query examples
    c_k ← (1/N_C) Σ_{(x_i,y_i)∈S_k} f_φ(x_i)          //Compute prototype from support examples
end for
J ← 0                                                 //Initialize loss
for k in {1, ..., N_C} do
    for (x, y) in Q_k do
        J ← J + (1/(N_C·N_Q)) (d(f_φ(x), c_k) + log Σ_{k'} exp(−d(f_φ(x), c_{k'})))   //Update loss
    end for
end for
```

انتخاب تابع فاصله‌ی مناسب یکی از چالش‌های این روش است. در اغلب مواقع تابع فاصله، یک تابع بدون پارامتر و ساده است [۴۳] . اما در برخی پژوهش‌ها، از تابع‌های فاصله‌ی پارامتری بسته به کاربرد مسئله، بهره برده اند [۴۴] . مزیت این روش‌ها علاوه بر سادگی و سرعت بالا در فاز استنتاج، نتایج بهتر آن نسبت به روش‌های

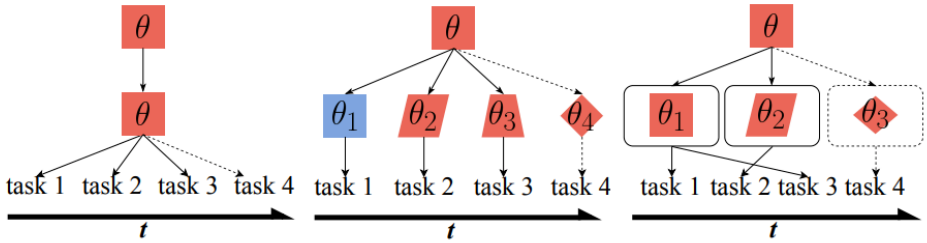
[1] negative log-likelihood

دیگر متایادگیری است. به طور عمده در آزمایش‌های مختلف نشان داده شده است روش‌های مبتنی بر نماینده بر روش‌های دیگر برتری دارد. به این جهت، می‌توان دریافت پیدا کردن یک نگاشت مناسب به فضای معنایی ویژگی‌ها که در آن یک فاصله مناسب جهت دسته‌بندی وجود دارد، برای حل مسائل بزرگ کافی است.

از مزایای این روش، بدون پارامتری بودن مرحله‌ی استنتاج است. این موضوع ما را از بدست آوردن پارامترهای خاص هر وظیفه به صورت یک فرآیند یادگیری بی‌نیاز می‌کند و دسته‌بندی با رویکرد مولد انجام می‌شود. به همین دلیل اگر دسته‌ی جدیدی در فاز آزمون اضافه شود، نیاز به به روز رسانی پارامترها نیست و فقط کافی است نماینده‌ی آن دسته محاسبه شود. همچنین این دسته از روش‌های متایادگیری در مقابل سایر رویکردها به دقت‌های بالاتری در مسائل گوناگون رسیده است. در پژوهش حاضر، از این رویکرد متایادگیری استفاده کرده‌ایم. به همین جهت نیاز به دقت بیشتر در شیوه‌ی یادگیری آن داریم که به صورت شبه کد در الگوریتم ۱ آمده است.

۲-۵.۲ اشتراک تجربه‌ی احتمالاتی

در بررسی شبکه‌های متایادگیری، یک نقطه‌ی ضعف مشترک بین روش‌های متایادگیری دیده شد. تمامی این شبکه‌ها فرض می‌کردند اطلاعات موجود در متاپارامترها باید بین همه‌ی وظیفه‌ها به اشتراک گذاشته شود [۴،۶،۴۵] . این در حالی است که الگوی یادگیری انسان، تجربیات حاصل از تجربه روی وظیفه‌های مختلف را با همه‌ی وظیفه‌ها به اشتراک نمی‌گذارد. در واقع عملکرد به نحوی است که ابتدا یک مشابهت‌یابی بین وظیفه‌ی در حال انجام با تجربیات قبلی صورت می‌پذیرد و از هر کدام از آن‌ها بخشی از تجربیات مرتبط به کمک یادگیری بهتر و سریع‌تر وظیفه‌ی فعلی می‌آید. جهت اصلاح این روند، کارهایی جهت اصلاح روند دانش قابل انتقال به هر وظیفه صورت پذیرفته است. یک روند رویکرد متایادگیری احتمالاتی بوده است که برای هر وظیفه، یک مدل از توزیع تمامی مدل‌های ممکن نمونه گرفته می‌شود [۴۶–۴۸] . در این طیف روش‌ها، تمرکز بر روی انتقال دانش به وظیفه هدف است. به همین دلیل تعمیم‌پذیری دانش بین وظیفه‌های با همبستگی بالا [1] اتفاق نمی‌افتد. در واقع در لحظه‌ی اشتراک دانش، فقط به وظیفه‌ی فعلی توجه می‌شود و ارتباط آن با وظایف دیگر در نظر گرفته نمی‌شود.



شکل ۲-۷: انواع مختلف انتقال تجربه از متاپارامترها به پارامترهای خاص وظیفه؛ در شکل سمت چپ، متاپارامترها به صورت یکسان در اختیار تمامی وظیفه‌ها قرار می‌گیرد. در شکل وسط، به ازای هر وظیفه خاص، یک متاپارامتر خاص نمونه‌گیری می‌شود و به آن وظیفه انتقال می‌یابد. در شکل سمت راست، وظایف خوشه‌بندی می‌شوند و به هر خوشه، تجربیات یکسانی انتقال می‌یابد. [۷]

برای بهبود این مسئله، باید به یک ساختار متایادگیری رسید که بتوان تعمیم‌پذیری و شخصی‌سازی دانش را برای

[1] Highly Correlated

هر وظیفه به انجام رساند. به این جهت در دسته‌ای دیگر از روش‌های پیشنهادی، با الهام از مفاهیم روانشناسی، هنگام مواجهه با یک وظیفه انسان به خوشه‌بندی آن می‌پردازد و به این وسیله، شباهت وظیفه با تجربیات گذشته سنجیده می‌شود. در نتیجه انتقال تجربه برای یک وظیفه، از خوشه‌ی مرتبط با آن سرچشمه می‌گیرد. البته برخی از تجربیات هم به صورت جهانی [1] به همه‌ی وظایف انتقال می‌یابند [۷] . در شکل ۲-۷ به ترتیب از چپ به راست سه رویکرد مطرح شده جهت استفاده از متاپارامترها برای رسیدن به پارامترهای خاص وظیفه مشخص شده است.

۲-۳ استفاده از بدنه [2] بیانگر [3] برای رسیدن به کارایی نمونه [4]

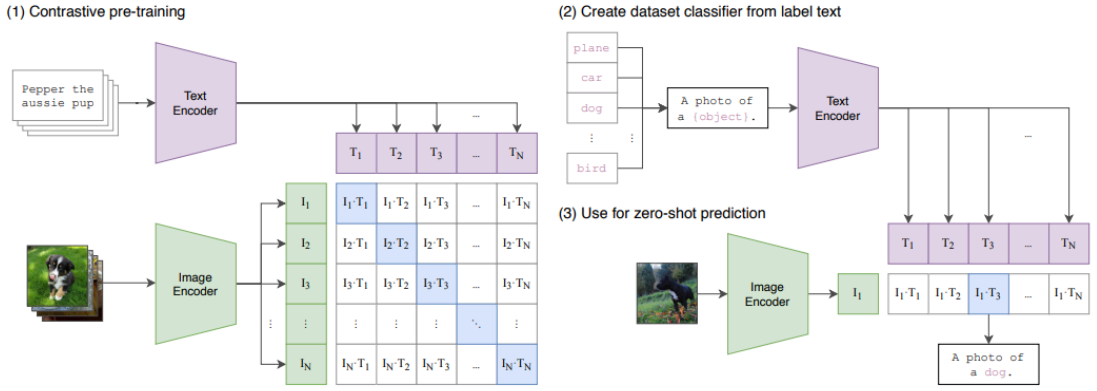
در تنظیمات نمونه محدود، داده کارآمدی مدل اهمیت به‌سزایی دارد. یعنی مدل برای یادگرفتن از حداقل تعداد نمونه به ازای هر دسته استفاده کند. به این منظور، یک رویکرد مهم در پژوهش‌های اخیر و گذشته استفاده از یادگیری انتقالی [5] و استفاده از بدنه [6] های بزرگ است که بازنمایی‌های بیانگر خروجی می‌دهند. این بدنه‌ها معمولا بر روی مجموعه دادگان بزرگ از قبل پیش‌آموزش دیده‌اند [۴۹–۵۱] .

علاوه بر یادگیری نظارت‌شده، یادگیری بدون ناظر [۵۰،۵۱] و یادگیری خودنظارتی [۵۲] نشان داده‌اند که می‌توانند روش‌های مناسبی برای آموزش یک بدنه‌ی مناسب برای یادگیری نمونه محدود باشند. اخیرا، مدل‌های بنیادی بزرگ مانند CLIP [۸] و DINO [۵۳] با استقبال گسترده‌ای در حیطه‌ی یادگیری نمونه محدود و بدون نمونه [7] مواجه شده‌اند [۵۴،۵۵] . در شدیدترین حالت یادگیری، هیچ نمونه‌ای برای یادگیری در اختیار نداریم و باید برای یک داده‌ی آزمون، استنتاج را انجام دهیم. در این حالت نیاز به ترکیب اطلاعات زبانی و تصویری داریم. به نحوی که بازنمایی برچسب‌های هر دسته، در جایی از فضا نگاشت شوند که نزدیک به بازنمایی تصاویر همان دسته باشد. برای حل این دسته از مسائل، از یادگیری تباینی استفاده می‌شود و مهم‌ترین معماری آن CLIP است که در شکل ۲-۸ آمده است.

۲-۴ خاصیت مجموعه باز بودن

برای رویارویی با مجموعه باز بودن دسته‌بندی دو رویکرد اساسی پیشنهاد شده است. یک رویکرد سلبی، دادن خودآگاهی [8] به مدل است. یعنی یک مدل آگاهی این را داشته باشد که روی چه ورودی می‌تواند خروجی تولید کند. در نتیجه وقتی یک ورودی خارج از حیطه‌ی توانایی‌های مدل است، از اظهار نظر روی آن خودداری کند. البته داشتن خودآگاهی، گروه بزرگتری از مسائل را در بر می‌گیرد که به سه دسته‌ی عمده تقسیم می‌شوند. گروه اول تشخیص خارج از توزیع [9] است. در این گروه تصاویری که از توزیع دادگان آموزش نیستند یا از دامنه‌های دیگر

[1] Globallly
[2] Backbone
[3] Expressive
[4] Sample Efficiency
[5] Transfer Learninig
[6] Backbone
[7] Zero Shot
[8] Self Awareness
[9] Out of Distribution



شکل ۲-۸: مدل CLIP سعی می‌کند بازنمایی تصاویر و برچسب‌های متناظر که روی قطر اصلی قرار دارند را به هم نزدیک کند. در نهایت از این ویژگی برای دسته‌بندی بدون نمونه استفاده می‌کند [۸].

مانند حملات تخاصمی [۱] می‌آیند شناسایی می‌شود [۹،۵۶] . گروه دوم طبقه‌بندی واقع‌بینانه [۲] است که وقتی یک نمونه برای طبقه‌بندی سخت به نظر می‌رسد، از دسته‌بندی آن خودداری می‌شود [۵۷] . گروه آخر هم مسئله‌ی شناسایی مجموعه باز است که وقتی نمونه‌ای از کلاس بدیع می‌بینند، آن را طبقه‌بندی نکند.

اما رویکرد دیگر که به نوعی ایجابی است، در حیطه‌ای است که مدل، خود را با ذات مجموعه باز بودن مسئله تطبیق دهد و برای تعمیم دانسته‌هایش روی دسته‌های جدید آماده باشد. اگر مدل مولد باشد با تنظیم توزیعی که یادگرفته میتواند به سرعت با اضافه شدن دسته‌های جدید انطباق پیدا کرد. این خاصیت در مدل‌های بدون پارامتر هم نمود دارد. اما در مدل‌های تمایزگذار با اضافه شدن دسته‌های بدیع، نیاز به آموزش مدل از ابتدا بر روی تمامی دسته‌ها است. به همین جهت، در مسائل دسته‌بندی بزرگ مقیاس باید رویکردی را انتخاب کنیم که به کمترین میزان آموزش مجدد شبکه نیاز باشد. البته در مدل‌های تمایزگذار، روش‌هایی پیشنهاد شده است که بتوان وزن های مرتبط با دسته‌های جدید را تنها با طی کردن گذر جلو رو [۳] و بدون آموزش مجدد، بدست آورد.

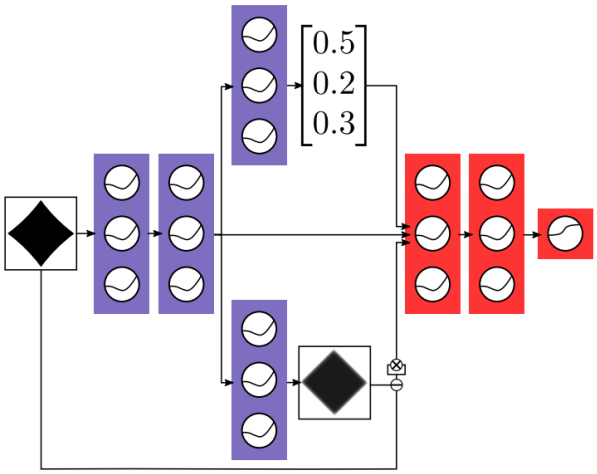
۲-۴.۱ رویکرد سلبی؛ شناسایی دسته‌های بدیع

در این رویکرد، سعی بر تشخیص دسته‌های بدیع و به نوعی عدم استنتاج مدل روی آن‌ها است. یعنی مدل فرآیندی برای تطابق با دسته‌های جدید و پذیرش آن‌ها طی نمی‌کند. عمده‌ی تمرکز پژوهش‌ها در این دسته روی تشخیص کلاس‌های بدیع در تنظیمات بزرگ مقیاس بوده است. اما در سال‌های اخیر روش‌هایی ارائه شده‌اند که علاوه بر مسائل بزرگ مقیاس با تعداد نمونه کافی، در مسائل نمونه محدود با تعداد دسته‌ی محدود هم عملکرد مناسبی داشته باشند. در یک روش، به این نتیجه رسیده‌اند که دسته‌بندی مبتنی بر سافت‌مکس [۴] چون روی دسته‌های آموزش دچار بیش برازش [۵] است، نمی‌تواند راه حل خوبی برای تشخیص دسته بدیع باشد. به همین دلیل با الهام از رویکرد متایادگیری، در هر وظیفه دسته‌های دیده شده و بدیع را جای داده‌اند و علاوه بر تابع هزینه اصلی، یک

[1] Adverserial Attacks
[2] Realistic
[3] Forward Pass
[4] Softmax
[5] Overfit

جمله برای بیشینه کردن آنتروپی توزیع پسین [1] برای دسته‌های بدیع قرار داده‌اند [۴۳] .

در یک کار دیگر، ابتدا یک خودکدگذار [2] به وسیله‌ی داده‌های درون توزیع آموزش داده می‌شود. بر روی این قسمت، یک لایه‌ی دسته‌بند هم اضافه می‌شود تا احتمال تعلق یک نمونه به دسته‌های درون توزیع را بدهد. سپس یک قسمت دیگر به وسیله‌ی داده‌های درون توزیع و دسته‌های بدیع آموزش داده می‌شود که باید تشخیص دهد آیا یک داده از کلاس بدیع هست یا خیر. عملکرد این مدل در شکل ۲-۹ آمده است [۹] .

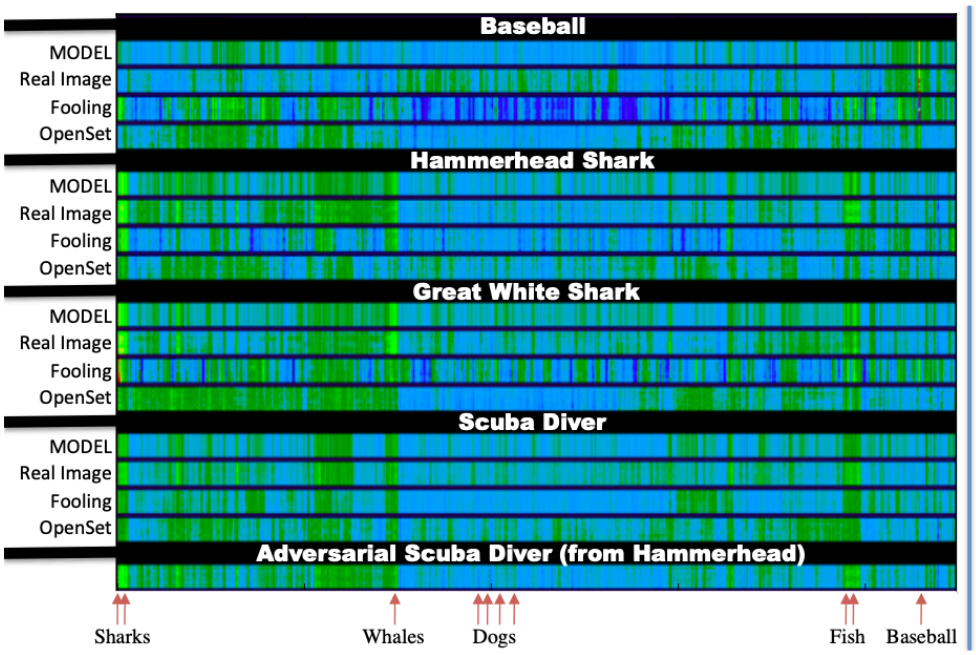


شکل ۲-۹: استفاده از یک خودکدگذار و تولید احتمالات دسته‌بندی برای یک ورودی که بر روی داده‌های درون توزیع آموزش داده شده است (قسمت آبی) و آموزش قسمت قرمز که وظیفه‌ی تشخیص داده‌هایی که از دسته‌های بدیع آمده اند را دارد [۹].

در کار دیگری با اشاره به اینکه نمی‌توان در تشخیص مجموعه باز، یک دسته ناشناخته اضافه کرد و اگر داده‌ای از دسته‌ی بدیع بود به آن اختصاص داد، یک لایه به نام OpenMax معرفی می‌کند و با اعمال آن در لایه‌ی ماقبل آخر [3] یک شبکه‌ی عصبی، سعی در تشخیص مجموعه باز دارد [۱۰] . همچنین با بررسی‌هایی که انجام داده، نشان داده است که با اعمال یک آستانه [4] بر روی خروجی‌های احتمالاتی نمی‌توان به تشخیص مجموعه باز پرداخت چرا که لایه‌ی سافت‌مکس یک ماهیت مجموعه بسته دارد.

در این کار با تحلیل انرژی یا همان تابع امتیاز لایه قبل از سافت‌مکس در شبکه‌های عصبی، نشان داده اند که تمایز واضحی بین داده‌ی واقعی با داده‌های خارج از توزیع هر دسته وجود دارد که نمونه آن در شکل ۲-۱۰ آمده است. با استفاده از همین موضوع، برای دسته‌ها میانگین بردارهای فعالسازی لایه‌ی آخر را حساب کرده اند و برای هر دسته نماینده‌ی انرژی آن را بدست آورده اند. سپس برای یک نمونه‌ی جدید با محاسبه‌ی فاصله‌ی بردار فعال‌سازی آن نمونه تا میانگین، راجع به خارج از توزیع بودن داده اظهار نظر می‌کنند.

[1] Posterior
[2] Auto Encoder
[3] Penultimate
[4] Threshold



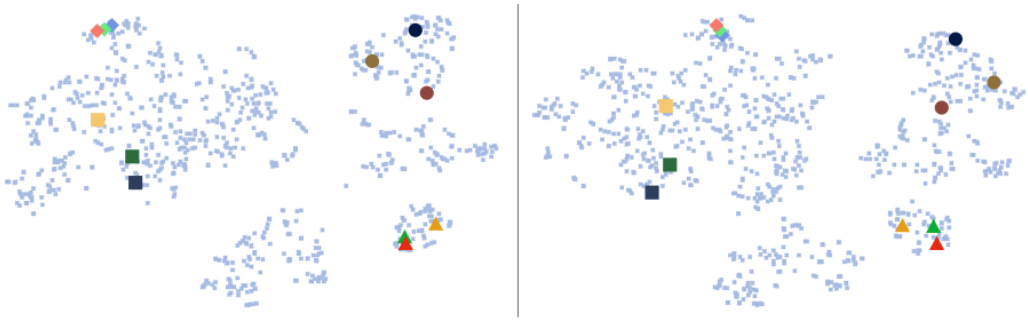
شکل ۲-۱۰: مصور سازی تابع انرژی روی نمونه‌های واقعی، دستکاری شده و خارج از توزیع به ازای هر دسته. با توجه به این شکل، مشخص است در داده‌هایی که از دسته‌ی بدیع می‌آیند، خطوط سبز پرتکرار است و به این ترتیب می‌توان آن‌ها را تشخیص داد [۱۰].

۲-۴ رویکرد ایجابی؛ تنظیم مدل با اضافه کردن توانایی طبقه‌بندی دسته‌های جدید

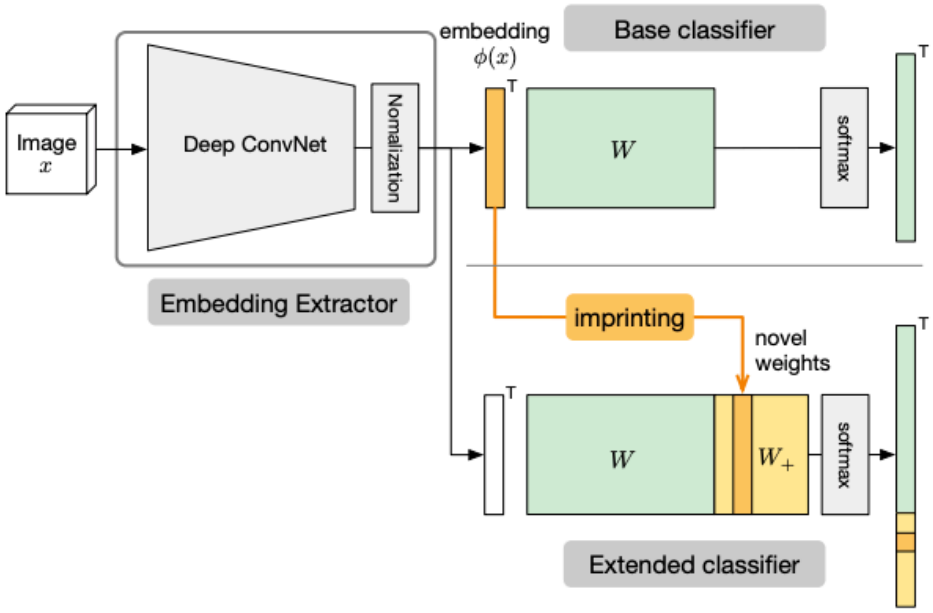
در این روش‌ها سعی می‌شود وزن‌های دسته‌ی جدید در مدل را از روی تعداد نمونه‌ی محدودی که برای دسته‌ی بدیع تشخیص داده شده است، حساب کنند. به طور کلی یک راه‌حل متداول حکاکی [۱] وزن‌ها است [۱۱،۱۲] .

شهود این مسئله در شکل ۲-۱۱ مشخص است. در این مصورسازی نشان داده شده که وزن‌های لایه‌ی آخر یک شبکه‌ی عصبی با تعبیه [۲] یک داده از همان دسته، در جای نزدیک به هم در فضای تعبیه‌ها قرار می‌گیرند. در واقع ارتباط مستقیم بین مقادیر فعال‌سازی و پارامترهای همان دسته وجود دارد. این موضوع با عملکرد شبکه‌های عمیق سازگار است. چرا که برای محاسبه‌ی امتیاز یک دسته، مقادیر فعال‌سازی مربوط به داده‌ی فعلی را در وزن‌های مربوط به نورون آن دسته ضرب داخلی می‌کنیم یا به نوعی شباهت کسینوسی می‌گیریم. در نتیجه مقادیر فعال‌سازی به وزن‌های هر دسته‌ای که شبیه‌تر باشد، به آن دسته اختصاص می‌یابد. با در نظر گرفتن این موضوع، می‌توان برای دسته‌های بدیع، با میانگین‌گیری روی مقادیر فعال‌سازی نمونه‌های آن دسته که از یک شبکه‌ی پیش‌آموزش دیده به‌دست می‌آیند، وزن‌های دسته جدید را حکاکی کرد. این موضوع در شکل ۲-۱۲ نشان داده شده است.

[1] Imprinting
[2] Embedding



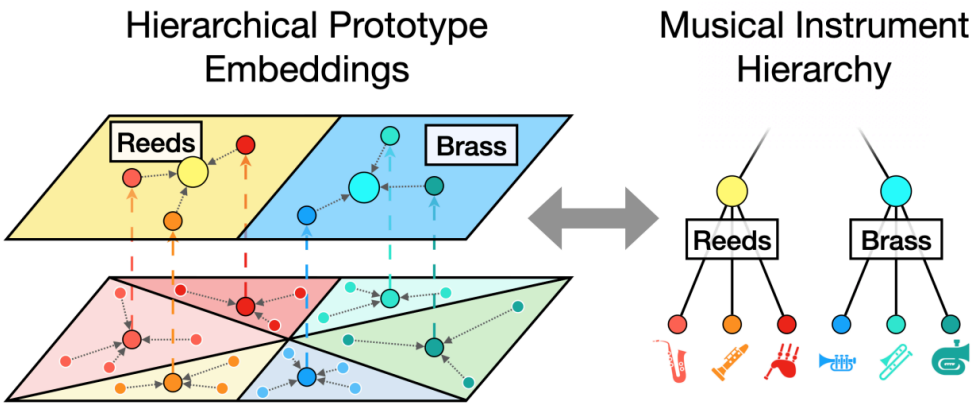
شکل ۲-۱۱: مصورسازی به کمک t-SNE برای مقادیر فعال‌سازی هر دسته قبل از لایه آخر یک شبکه‌ی عصبی تمام متصل در سمت چپ و پارامترهای مربوط به هر دسته در لایه‌ی آخر شبکه‌ی تمام متصل در سمت راست. هر نقطه‌ی آبی یک دسته را نشان می‌دهد. نقاط مشخص شده با شکل یکسان و رنگ یکسان مربوط به یک دسته اند. از طرفی اشکال مشابه دسته‌های درشت‌دانه را نشان می‌دهند. به عنوان مثال مثلث‌ها دسته‌های مربوط به پرندگان اند [۱۱].



شکل ۲-۱۲: یک شبکه‌ی عصبی پایه که با دسته‌های درون دامنه آموزش داده شده است و وزن‌های لایه‌ی آخر آن با مستطیل سبز نشان داده شده است. سپس برای داده‌های خارج از توزیع که از دسته‌های بدیع هستند، از این شبکه پیش‌آموزش دیده به عنوان استخراج‌گر تعبیه استفاده می‌شود و این تعبیه بدست آمده که با رنگ نارنجی نشان داده شده است، به عنوان وزن‌های کلاس جدید اضافه و به نوعی حک می‌شود. [۱۲].

۲-۵ یادگیری سلسله‌مراتبی نمونه محدود

در رویکرد متایادگیری مبتنی بر نماینده، کارهایی در زمینه‌ی استفاده از ساختار سلسله‌مراتبی در داده‌ها مطرح شده است. یک روش تحمیل ساختار سلسله‌مراتب روی یادگیری فضای ویژگی‌ها است [۱۳] . در این روش، از یک مجموعه دادگان که از قبل ساختار سلسله‌مراتبی روی مسئله‌ی هدف دارد، استفاده می‌کنند. ساختار مدل به این صورت است که در ابتدا برای تمامی دسته‌ها در ریزدانه‌ترین حالت ممکن، نماینده‌ها محاسبه می‌شوند. این نماینده‌ها در درخت سلسله‌مراتب پایین‌ترین سطح اند و در واقع برگ‌های درخت را نشان می‌دهند. سپس با استفاده از سلسله‌مراتب از پیش تعیین شده (مانند استفاده از یک مجموعه دادگان سلسله‌مراتبی)، نماینده‌های یک سطح بالاتر در درخت سلسله‌ی مراتب، با میانگین‌گیری روی نماینده‌های زیرمجموعه‌ی هر دسته به‌دست می‌آیند. این طریقه‌ی محاسبه در شکل ۲-۱۳ آمده است.



شکل ۲-۱۳: متایادگیری مبتنی بر متر سلسله‌مراتبی؛ در ریزدانه‌ترین حالت، نماینده‌های دسته‌ها محاسبه می‌شود و به صورت بازگشتی، از روی آن‌ها نماینده‌ی دسته‌های سطوح بالاتر یا متانماینده‌ها محاسبه می‌شود. در زمان استنتاج، دسته‌بندی در همه‌ی سطوح انجام می‌شود و خطا برای سطوح ریزدانه و درشت‌دانه محاسبه و وزن‌ها به روز رسانی می‌شوند [۱۳].

در واقع این روش یک بسط روی متایادگیری مبتنی بر نماینده است. در پایین‌ترین سطح درخت با در نظر گرفتن دسته‌های مسئله در ریزدانه‌ترین حالت ممکن، با اعمال متایادگیری مبتنی بر متر، نماینده‌ها را بدست می‌آوریم. سپس به ترتیب به سطح‌های بالاتری می‌رویم و هر بار با اعمال روش متایادگیری مبتنی بر متر، نماینده‌هایی از نماینده‌های دسته‌های سطح پایین می‌سازیم که می‌توان به آن‌ها واژه‌ی متانماینده [1] را نسبت داد. در معادله‌ی ۸.۲ تابع خطای این مدل آمده است.

$$\mathcal{L}_{hierarchical} = \sum_{h=۰}^{H} e^{-\alpha.h} \mathcal{L}_{CE}^{(h)} \tag{۸.۲}$$

در این تابع خطا، یک ابرپارامتر [2] برای تنظیم مقدار توجه به هر سطح از سلسله‌ی مراتب وجود دارد که با α نشان

[1] Meta Prototype
[2] Hyperparameter

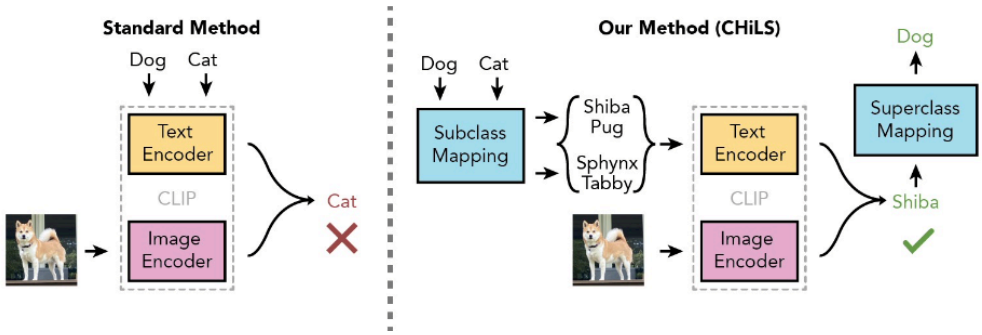
داده شده است. از ضریب حاوی این ابرپارامتر برای تنظیم میزان اثردهی تابع خطای بی‌نظمی متقابل [1] در سطوح مختلف درخت سلسله‌مراتب استفاده می‌شود. مقدار h برابر صفر نشان دهنده‌ی پایین‌ترین سطح در سلسله‌مراتب (برگ‌ها) است. بنابراین مقدار α اگر مثبت باشد، توجه را روی دسته‌بندی در سطوح با ریزدانگی بیشتر منعطف می‌کند و خطای بیشتری برای آن سطوح در نظر می‌گیرد. اگر مقدار آن برابر صفر باشد، در تمامی سطوح رفتار مشابهی دارد و اگر مقدار منفی داشته باشد، توجه بیشتری به سطوح با ریزدانگی کمتر دارد. به طور کلی اعمال سلسله‌مراتب در متایادگیری مبتنی بر متر سبب می‌شود فضای معنایی که توسط شبکه‌ی نگاشت ویژگی‌ها یاد گرفته می‌شود، این ساختار سلسله‌مراتبی را داشته باشد و بنابراین اگر یک نمونه‌ی جدیدی که در کلاس‌های پایه مسئله وجود ندارد را برای ارزیابی ورودی دهیم، در نزدیکی نماینده‌ای قرار می‌گیرد که متعلق به آن دسته است.

در یک روش دیگر، با تعریف یک خطای مبتنی بر سلسله‌مراتب، به هدایت نماینده‌های هر دسته پرداخته شده است [۵۸] . در این روش، این خطا بر اساس فاصله در سلسله‌مراتب عمل می‌کند. به این صورت که اگر یک داده در یک دسته‌ی اشتباه توسط مدل قرار گیرد، خطای پایه در شبکه‌ی مبتنی بر نماینده در یک ضریب که از ساختار سلسله‌مراتب به‌دست می‌آید، ضرب می‌شود. به طور مثال، با دسته‌بندی اشتباه یک داده، فاصله‌ی آن دسته در یک سلسله‌مراتب از پیش تعیین شده با دسته‌ی صحیح، به صورت کوتاه‌ترین مسیر در درخت سلسله‌مراتب محاسبه می‌شود و این ضریب در خطای پایه شبکه ضرب می‌شود. در نتیجه اگر اشتباه به نحوی باشد که دو دسته‌ی صحیح و اشتباه زیرمجموعه‌ی یک گره والد در فاصله‌ی نزدیک باشند، اثر خطا کمتر است و در نماینده‌های یادگرفته شده، ویژگی سلسله‌مراتبی بودن منعکس می‌شود.

اخیرا در حوزه‌ی یادگیری بدون نمونه، استفاده از مجموعه برچسب‌های سلسله‌مراتبی مورد توجه قرارگرفته‌است [۱۴] . همانطور که گفته شد، در دسته‌بندی بدون نمونه نیاز به یک مدل تصویری-زبانی [2] داریم که مهم‌ترین آن‌ها CLIP است [۸] . در بررسی‌های انجام شده، مشخص شد این مدل‌ها در یک ریزدانگی خاص می‌توانند عملکرد مناسبی داشته‌باشند و اگر برچسب‌های یک مجموعه داده درشت‌دانه باشد، مدل با خطاهای زیادی مواجه می‌شود. به همین دلیل، از مجموعه دادگان سلسله‌مراتبی استفاده کرده‌اند و به جای برچسب یک دسته‌ی درشت‌دانه، برچسب‌های فرزندان ریزدانه‌ی آن را به کار برده‌اند. همچنین اگر یک مجموعه دادگان دارای برچسب‌های سلسله‌مراتبی نبود، از یک مدل زبانی بزرگ [3] استفاده کرده‌اند و به این وسیله برچسب‌های فرزندان ریزدانه را بدست آورده‌اند. با اعمال این فرآیند، دقت‌های دسته‌بندی بدون نمونه افزایش قابل توجهی داشته‌است. نحوه‌ی کار این رویکرد در شکل ۲-۱۴ مشخص شده‌است. البته باز بودن نسبت به ریزدانگی [4] هنوز یک چالش باز در این دسته از مدل‌هاست.

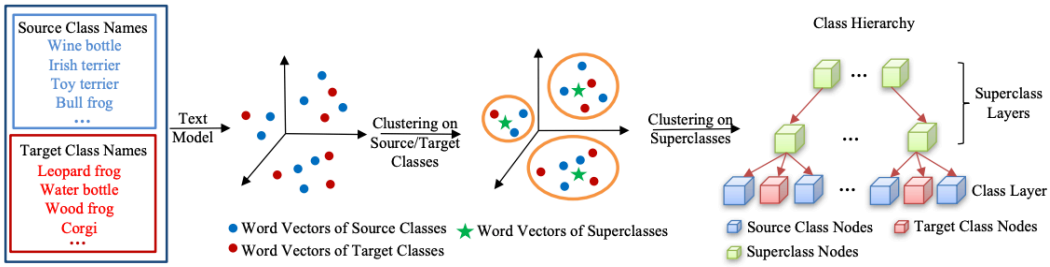
یک رویکرد دیگر، استفاده از دو دسته مبدا و مقصد و انتقال دانش است [۱۵،۳۳] . یعنی در تنظیمات بزرگ مقیاس، یک سری دسته‌ی مبدا داریم که از آن‌ها به تعداد زیاد نمونه در دسترس است. اما در دسته‌های مقصد، با دسته‌های کاملا متفاوت و جدیدی روبرو می‌شویم که از آن‌ها نمونه‌های محدودی در دسترس است. حال می‌خواهیم یک ساز و کار مناسب برای انتقال دانش بین مبدا و مقصد برقرار کنیم. به همین منظور از سلسله‌مراتبی بودن ذاتی مسائل بزرگ مقیاس استفاده می‌کنند و دسته‌های مبدا و مقصد را در قالب یک درخت سلسله‌مراتب و تحت والدهای مشترک به عنوان ابردسته‌ها، تقسیم‌بندی می‌کنند. برای بدست آوردن این درخت سلسله‌مراتب، نیازی به داشتن ارتباط والد و فرزندی بین دسته‌ها نیست و با استفاده از رویکردهای داده محور مانند استفاده از بازنمایی

[1] Cross Entropy
[2] Vision-Language Model (VLM)
[3] Large Language Model
[4] Open-Granular



شکل ۲-۱۴: با توجه به اینکه مدل CLIP در برچسب‌های درشت‌دانه عملکرد مناسبی ندارد، برای هر دسته فرزندان با برچسب‌های ریزدانه به‌دست می‌آید و دسته‌بندی در آن سطح انجام می‌شود و سپس حاصل دسته‌بندی به سطح درشت‌دانه منتشر می‌شود [۱۴].

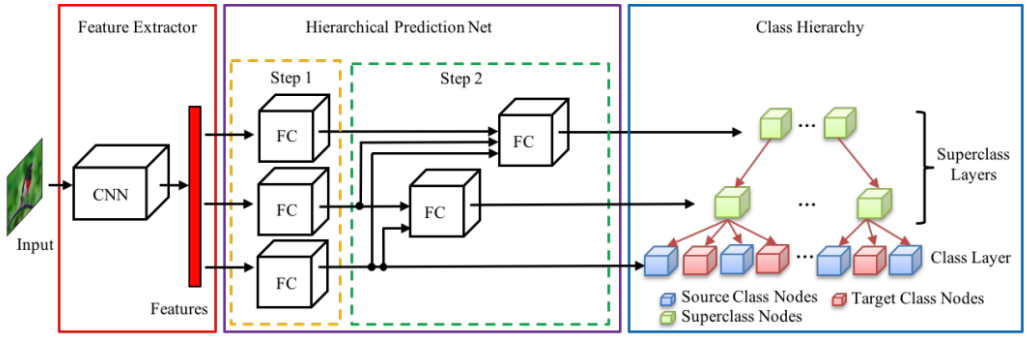
زبانی دسته‌ها، مانند آنچه در شکل ۲-۱۵ آمده است، می‌توان این کار را انجام داد.



شکل ۲-۱۵: تشکیل درخت سلسله‌مراتب با رویکرد داده محور؛ از بازنمایی زبانی دسته‌ها که مفاهیم معنایی در خود دارند استفاده می‌کنیم و با یک روش خوشه‌بندی، این درخت را تشکیل می‌دهیم. منظور از Source دسته‌های مبدا و Target دسته‌های مقصد اند و همینطور هاSuperclass ابردسته‌ها را نشان می‌دهند [۱۵].

با داشتن این درخت سلسله‌مراتب، می‌توان یک مدل بازنمایی ساخت که تعبیه‌های با قابلیت انتقال [۱] بالاتر بین دسته‌های مبدا و مقصد داشته باشند. به این منظور، مشابه شکل ۲-۱۶ از دسته‌های مبدا استفاده می‌کنیم و برای هر سطح در سلسله‌مراتب، یک دسته‌بند اختصاص می‌دهیم. نکته‌ی قابل توجه این است که دسته‌بندهای سطوح بالاتر را روی اطلاعات دسته‌بندهای ریزدانه مشروط می‌کنیم و به نوعی این اطلاعات را ورودی می‌دهیم. در نتیجه مدل یاد می‌گیرد اطلاعات ریزدانه را طوری تولید کند تا قابل استفاده در سطوح درشت‌دانه باشد که بین مبدا و مقصد مشترک اند. در نهایت در زمان آزمون و روی دسته‌های مقصد، از بازنمایی مشترک که با رنگ قرمز در شکل ۲-۱۶ نشان داده شده‌است، استفاده می‌شود. این بازنمایی اطلاعات سطوح درشت‌دانه‌ی مشترک را در خودش دارد و با یک دسته‌بندی مبتنی بر متر مانند نزدیک‌ترین همسایه، دسته‌بندی انجام می‌شود.

[1] Transferablility



شکل ۲-۱۶: وارد کردن اطلاعات سلسله‌مراتب در شبکه‌ی بازنمایی مشترک بین مبدا و مقصد که با رنگ قرمز مشخص شده است. با استفاده از داده‌های دسته‌های مبدا که تعداد بالای نمونه دارند، دسته‌بندهای موجود در مستطیل بنفش آموزش داده می‌شوند و با گرادیانی که از آن‌ها به شبکه‌ی بازنمایی مشترک در مستطیل قرمز برمی‌گردد، تعبیه‌هایی با قابلیت انتقال بین مبدا و مقصد تولید می‌شود [۱۵].

۲-۶ جمع‌بندی

این فصل به معرفی کارهای پیشین مهم و تاثیرگذار از جنبه‌های مختلف یادگیری بزرگ مقیاس، نمونه محدود اختصاص داشت. در ابتدا به معرفی یادگیری نمونه محدود با استفاده از روش‌های سنتی داده‌افزایی در سطح ویژگی و در سطح نمونه پرداختیم. سپس به معرفی رویکرد متایادگیری که یکی از رویکردهای مورد توجه بسیار در دسته‌بندی نمونه محدود است، پرداختیم. در متایادگیری، ساختار کلی روش‌ها بیان شد و سپس الگوریتم‌های مهم این حوزه مانند متایادگیری مبتنی بر بهینه‌سازی و متایادگیری مبتنی بر متر معرفی شد.

در ادامه به بررسی بدنه‌های بنیادی و مدل‌های پیش‌آموزش دیده برای رسیدن به کارایی داده پرداختیم. سپس به بررسی کارهای پیشین در حوزه‌ی مجموعه باز بودن که یکی از خواص ذاتی مسئله‌ی یادگیری بزرگ مقیاس است پرداختیم. در نهایت به معرفی و بررسی کارهای مهم در زمینه‌ی یادگیری سلسله‌مراتبی در تنظیمات نمونه‌محدود و بزرگ مقیاس پرداختیم.

هر کدام از این سرفصل‌ها به جنبه‌ی خاصی از تنظیمات بزرگ مقیاس، نمونه محدود اختصاص دارند و روی وجهه‌ی خاصی از مسئله تمرکز کرده اند. در فصل بعدی قصد داریم تا روش پیشنهادی خود را معرفی کنیم که بتواند تمامی چالش‌های این مسئله را در بر بگیرد. برای مقایسه‌ی شفاف بین هر کدام از سرفصل‌ها، جدول ۲-۱ در ادامه آمده‌است که بیان می‌کند مقالات بررسی شده در هر سرفصل چه ویژگی‌هایی از مسئله‌ی مدنظرمان را دارند.

جدول ۲-۱: بررسی شاخصه‌های پژوهش پیش‌رو در کارهای پیشین؛ اگر دسته‌ای به صورت ذاتی و یا با اعمال روندی ساده در اکثریت مقالاتش بتواند یک ویژگی را داشته باشد، برگزیده شده‌است. در غیر اینصورت، اگر نیاز به ترکیب با رویکردهای دیگری برای داشتن یکی از ویژگی‌های مشخص شده داشته باشد، انتخاب نشده است.

	مولد بودن	داده کارآمدی	در نظر گرفتن تغییرات ریزدانه (مرزهای تصمیم‌گیری باریک) بین دسته‌های بزرگ مقیاس	استفاده از سلسله‌مراتب ذاتی در دسته‌بندی بزرگ مقیاس	توجه به تغییر زیرجمعیت
یادگیری نمونه محدود با داده افزایی	✗	✓	✗	✗	✗
متایادگیری مبتنی بر بهینه‌سازی	✗	✓	✗	✗	✗
متایادگیری جعبه سیاه	✗	✓	✗	✗	✗
متایادگیری مبتنی بر متر	✓	✓	✗	✗	✗
استفاده از بدنه‌ی بیانگر	✗	✓	✓	✗	✗
مجموعه باز بودن	✓	✗	✗	✗	✓
یادگیری سلسله‌مراتبی نمونه محدود	✗	✓	✗	✓	✗

فصل ۳

راهکار پیشنهادی

مطالب این فصل

۳-۱ نمادگذاری مسئله و پیش‌نیازها ۳۲
۳-۲ دسته‌بند سطح درشت‌دانه ۳۵
۳-۳ یادگیرنده نمونه محدود در سطح ریزدانه ۳۶
۳-۴ جمع‌بندی روش و ارائه‌ی شبه‌کد ۴۰

همانطور که در فصل قبل مطرح شد، مسئله‌ی دسته‌بندی بزرگ مقیاس نمونه محدود زوایای مختلفی دارد که کارهای پیشین، هر کدام بر روی قسمت خاصی از حل چالش این مسئله متمرکز بوده‌اند. در نتیجه استفاده‌ی مستقیم از هر یک از این رویکردها دارای نقاط ضعفی در این مسئله است. در این فصل، با الهام از فواید و نقصان‌های هر کدام از این روش‌ها، یک راه‌حل جامع برای حل مسئله‌ی دسته‌بندی بزرگ مقیاس، نمونه محدود ارائه می‌دهیم. در ابتدا به نمادگذاری مسئله و بیان پیش‌نیازها می‌پردازیم.

۳-۱ نمادگذاری مسئله و پیش‌نیازها

در مسئله‌ی دسته‌بندی بزرگ مقیاس، نمونه محدود به دو مجموعه دادگان [۱] بزرگ‌مقیاس $\mathcal{D}_{source}$ و $\mathcal{D}_{target}$ دسترسی داریم. از این دو مجموعه به ترتیب با نام‌های دادگان مبدا و مقصد یا دادگان فاز متاآموزش [۲] و متاآزمون [۳] یاد میکنیم. هر یک از این مجموعه دادگان که به صورت $\mathcal{D} = \{(x^i, y^i_{super}, y^i_{base})\}_{i=1}^{|\mathcal{D}|}$ هستند، متشکل از سه مجموعه هستند.

۱. نمونه‌های داده در فضای ورودی تصویر $\mathcal{X} = \{x^i\}$

۲. برچسب هر نمونه در سطح درشت‌دانه (ابردسته [۴]) $\mathcal{Y}_{super} = \{y^i_{super}\}$

[1] Dataset
[2] Meta-Train
[3] Meta-Test
[4] Superclass

۳. برچسب هر نمونه در سطح ریزدانه (دسته پایه [1]) $\mathcal{Y}_{base} = \{y^i_{base}\}$

در سطح ابردسته فرض می‌کنیم $\mathcal{Y}_{super,target} \subseteq \mathcal{Y}_{super,source}$ که نشان می‌دهد برچسب‌های مقصد زیرمجموعه‌ی برچسب‌های دادگان مبدا است. همانطور که در ۱-۳.۲ بیان شد، با توجه به وجود تغییر زیرجمعیت در مسئله‌ی دسته‌بندی بزرگ مقیاس، این موضوع رایج است که دسته‌های ریزدانه متفاوت باشند. اما در سطح درشت‌دانه با توجه به وجود سلسله‌مراتب ذاتی در داده‌ها، قادر به تعریف یک‌سری دسته‌ی کلی هستیم که بین دادگان مبدا و مقصد همواره مشترک‌اند. به عنوان مثال در گونه‌های حیوانات، ممکن است گونه‌ای به تازگی شناسایی شود اما دسته‌هایی در سطح پرندگان و آبزیان و ... همواره ثابت هستند. با توجه به این موارد، برای سنجش خاصیت مجموعه باز بودن و تحمل تغییر زیرجمعیت [2] ، برچسب‌های سطح‌ریزدانه بین مجموعه دادگان مبدا و مقصد را ناسازگار و کاملا مجزا از هم در نظر گرفتیم و چالش‌برانگیزترین تنظیمات ممکن را اعمال کردیم $\mathcal{X}_{source} \cap \mathcal{X}_{target} = \emptyset$ و $\mathcal{Y}_{base,source} \cap \mathcal{Y}_{base,target} = \emptyset$. یعنی هیچ اشتراکی بین نمونه‌ها و همچنین برچسب‌های ریزدانه وجود ندارد.

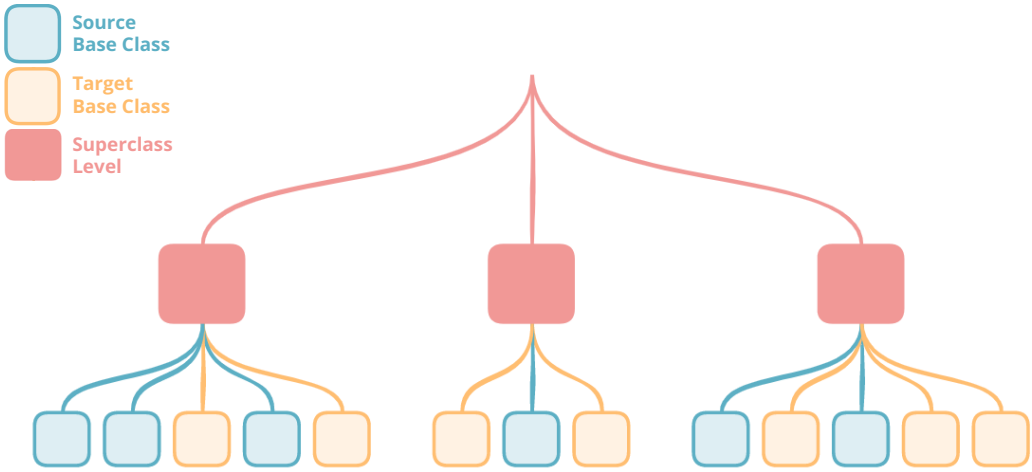
با توجه به اینکه قید اصلی مسئله، نمونه محدود بودن آن است، هر y_c یعنی برچسب هر دسته، دارای تعدادی نمونه‌ی محدود چه در سمت دادگان مبدا و چه مقصد است. علاوه بر این، روش پیشنهادی نشان داده‌است که هیچ نیازی به داده‌ی آموزش زیاد نداریم چرا که با استفاده از مدل‌های بنیادی بزرگ توانسته‌ایم به داده‌کارایی [3] و داشتن ویژگی‌های بیانگر برسیم. با استفاده از این رویکرد، نشان داده‌ایم حتی در تنظیمات بدون آموزش [4] توانسته‌ایم به کارایی مناسب دست یابیم.

برای یادگیری نمونه محدود، شبکه‌های پروتوتیپیکال [۶] را از بین رویکردهای متایادگیری انتخاب کرده‌ایم. این رویکرد در ۲-۴.۲ به طور کامل شرح داده‌شد. این انتخاب به دلیل مولد بودن این شبکه است و می‌تواند با دسته‌های پایه کاملا بدیع [5] در دادگان مقصد کار کند. فازهای متاآموزش و متاتست، به ترتیب روی دادگان مبدا و مقصد به صورت اپیزودیک انجام می‌شود. در هر اپیزود، یک مسئله‌ی یادگیری مستقل داریم که به عنوان یک وظیفه‌ی $\mathcal{T}$ از بین همه‌ی وظایف قابل تعریف روی $\mathcal{D}$ نمونه برداشته‌ایم و در نتیجه در هر اپیزود، یک مسئله‌ی K راه [6] N شات [7] داریم. این نمادگذاری به معنی این است که در یک اپیزود K دسته داریم و به ازای هر دسته N نمونه‌ی پشتیبان وجود دارد که N تعداد محدودی نمونه است.

در هر وظیفه، داده‌های مربوط به هر دسته به داده‌های پشتیبان و پرسش تقسیم می‌شوند که به ازای یک دسته‌ی c آن‌ها را به ترتیب با $\mathcal{S}^c$ و $\mathcal{Q}^c$ نمایش می‌دهیم. هدف متایادگیری رسیدن به دقت قابل قبول روی $\mathcal{Q}$ است که این هدف را با تطبیق سریع با به کار بردن نمونه‌های محدود پشتیبان $\mathcal{S}$ دنبال می‌کند.

ما از بدنه‌های بنیادی بزرگ برای تولید بازنمایی در مسئله‌ی دسته‌بندی نمونه محدود استفاده کرده‌ایم. این بدنه‌های بنیادی که وظیفه‌ی تولید تعبیه با وزن‌های منجمد [8] را دارند، در سطح درشت‌دانه و ابردسته‌ها با g_ϕ و در سطح‌ریزدانه و دسته‌های پایه با g_θ نمایش داده شده‌اند. همچنین شبکه‌های کاملا متصلی که برای انجام دسته‌بندی بر روی بدنه‌ها سوار می‌شوند در سطح ابردسته‌ها و دسته‌های پایه به ترتیب با f_φ و f_ϑ نشان داده شده‌است که در

[1] Base Class
[2] Sub-Population Shift
[3] Data Efficiency
[4] Free Training
[5] Novel
[6] Way
[7] Shot
[8] Freeze



شکل ۳-۱: تنظیمات مسئله پیش‌رو؛ در سطح درشت‌دانه، ابردسته‌ها بین مجموعه‌ی دادگان مبدا و مقصد مشترک هستند چرا که دسته‌های درشت‌دانه دچار اضافه یا کم شدن نمی‌شوند. اما در سطح ریزدانه، دسته‌ها و نمونه‌های موجود در هر دسته، بین دادگان مبدا و مقصد ناسازگار و بدون هیچ اشتراکی طراحی شده‌اند.

جدول ۳-۱: نمادهای اختصاری مورد استفاده در این فصل و توضیح هر نماد

نماد	توضیح
$\mathcal{D}_{source}$	مجموعه دادگان مبدا
$\mathcal{D}_{target}$	مجموعه دادگان مقصد
$\mathcal{X}$	مجموعه نمونه‌ها در فضای تصویر
$\mathcal{Y}_{base}$	برچسب دسته‌های پایه (ریزدانه)
$\mathcal{Y}_{super}$	برچسب ابردسته‌ها (درشت‌دانه)
ϕ, φ	زیروند مربوط به سطح درشت‌دانه
θ, ϑ	زیروند مربوط به سطح ریزدانه
g	بدنه منجمد مدل‌های بنیادی بزرگ
f	سر قابل آموزش شبکه‌ها
$k - way$	تعداد دسته‌ها در یک وظیفه
$N - shot$	تعداد نمونه‌ها به ازای هر دسته
$\mathcal{T}$	یک وظیفه از بین مجموعه وظایف ممکن
$\mathcal{S}^c$	مجموعه دادگان پشتیبان برای یک دسته خاص
$\mathcal{Q}^c$	مجموعه دادگان پرسش برای یک دسته خاص

واقع بخشی است که ما آموزش می‌دهیم.

به طور خلاصه، با توجه به بایسته‌ها و چالش‌های مسئله‌ی دسته‌بندی بزرگ مقیاس نمونه محدود، مطابق شکل ۳-۱ یک تنظیمات جدید در سخت‌ترین حالت ممکن تعریف کرده ایم که در سطح ریزدانه بین دادگان مبدا و مقصد هیچ اشتراکی از لحاظ داده و برچسب نداریم. از طرفی علاوه بر دادگان مقصد، در سمت مبدا مجموعه دادگان نمونه محدود است و مثل سایر روش‌ها نیازی به جمع‌آوری دادگان بزرگ در این مرحله نداریم. در فاز متاآزمون و داده‌های مقصد نیاز به برچسب‌گذاری در سطح درشت‌دانه را با اعمال یک تکنیک خاص از بین برده‌ایم و بنابراین روش‌ما نیازی به دانستن $\mathcal{Y}_{super,source}$ برای داده‌های پشتیبان مقصد ندارد. نمادگذاری‌های مورد استفاده در جدول ۳-۱ آمده‌است.

۳-۲ دسته‌بند سطح درشت‌دانه

به دلیل ثابت بودن ابردسته‌ها در سطح درشت‌دانه، از یک رویکرد یادگیری نظارت‌شده استاندارد استفاده می‌کنیم. چرا که در شبکه‌های ژرف، رویکرد تمایزگذار نسبت به رویکرد مولد به دقت‌های بالاتری می‌رسد و اگر امکان استفاده از آن باشد، انتخاب اول است. در گام اول، با استفاده از تابع خطای بی‌نظمی متقابل [1] به صورت استاندارد، یک دسته‌بند f_φ را که بر روی یک بدنه‌ی منجمد g_ϕ که از یک مدل بنیادی پیش‌اموزش دیده آمده است، سوار می‌کنیم و آموزش می‌دهیم.

از طرفی چون از بدنه‌ی g_ϕ به عنوان تولید کننده‌ی بازنمایی استفاده می‌کنیم، نیازی نیست مجموعه دادگان به تعداد زیادی نمونه داشته باشد چرا که نمونه‌های دسته‌های پایه متعلق به یک ابردسته تجمیع می‌شوند. در نتیجه اگر مجموعه دادگان اولیه در سطح ریزدانه مناسب دسته‌بندی نمونه محدود باشند، این داده‌های تجمیع شده می‌تواند برای آموزش یک مدل دسته‌بندی به صورت متداول مورد استفاده قرار گیرد. مدل مورد استفاده برای f_φ تنها یک تبدیل خطی ساده به وسیله‌ی یک لایه‌ی Linear است. این انتخاب به جهت این است که پیچیدگی مدل را تا حد ممکن کم کنیم تا با داده‌ی محدود به راحتی و با کمترین پیچیدگی محاسباتی و زمان به یک مدل قابل اتکا برای سطح درشت‌دانه برسیم. علاوه بر آن، برچسب‌های درشت‌دانه بسیار متداول اند و در کلاس‌های آموزشی بدنه‌ی بنیادین احتمالا دیده شده اند. به همین جهت شبکه‌ی g_ϕ می‌تواند یک بازنمایی بیانگر برای این ابردسته‌ها تولید کند که با یک تبدیل خطی آن‌ها را به فضای دسته‌بندی مسئله‌ی خود می‌آوریم. از طرفی به دلیل درشت‌دانه بودن ابردسته‌ها، با پیچیده شدن مدل f_φ به سمت بیش برازش [2] روی نمونه‌های مبدا می‌رویم. چرا که یک تغییر توزیع زیرجمعیت بین مبدا و مقصد وجود دارد.

علی رغم اینکه ابردسته‌های فاز متاآزمون بدیع نیست و با ابردسته‌های مبدا مشترک است، تمایز صد درصدی دسته‌های پایه موجود در هر ابردسته بین مبدا و مقصد چالش ایجاد می‌کند و موجب افت دقت می‌شود. برای تخفیف این اثر، ما یک تکنیک یادگیری نیمه نظارتی را طراحی کرده‌ایم. همانطور که گفته شد، در فاز متاآزمون نیاز به برچسب ابردسته‌ی داده‌های پشتیبان نداریم. اما با استفاده از این واقعیت که نمونه‌های متعلق به یک دسته‌ی پایه، حتما متعلق به یک ابردسته هستند، دسته‌بند درشت‌دانه f_φ را در فاز متاآزمون تنظیم بهینه می‌کنیم.

[1] Cross Entropy
[2] Overfitting

به همین منظور، با برداشتن k شات نمونه‌های پشتیبان از یک دسته‌ی پایه و با کنار هم قرار دادن احتمالات تعلق هر کدام از این نمونه‌ها که توسط شبکه‌ی f_φ خروجی داده می‌شود، با در نظر گرفتن استقلال شرطی داده‌ها و استفاده از قانون بیز داریم

$$P(y|\mathcal{S}) = P(y|x^1, \dots, x^K) = \frac{P(y)\prod_{i=1}^{K} P(x^i|y)}{P(x^1, \dots, x^K)} = \frac{P(y)\prod_{i=1}^{K} P(y|x^i)P(x^i)}{P(y)^K P(x^1, \dots, x^K)}. \quad (۱.۳)$$

در نتیجه می‌توان شبه‌برچسب [1] ابردسته که حاصل از این اطلاعات نیمه‌نظارتی برای k شات نمونه است را بدست آورد. می‌دانیم شبکه‌ی f_φ احتمالات $P(y|x^i)$ را خروجی می‌دهد. برای تعیین شبه‌برچسب این نمونه‌ها، کافی است لگاریتم درست‌نمایی [2] های شرطی را به ازای همه‌ی ابردسته‌ها با هم مقایسه کنیم و عبارت حائز بیشترین مقدار را به عنوان ابردسته انتخاب کنیم. با توجه به عدم تناسب جمعیت ابردسته‌ها در واقعیت، ما از احتمالات پیشین [3] مربوط به ابردسته‌ها صرف نظر نمی‌کنیم. در نتیجه رابطه‌ی توضیح داده شده به صورت زیر بدست می‌آید.

$$\underset{y_{super}}{\operatorname{argmax}} \sum_{i=1}^{K} \log P(y_{super}|x^i) - (K-1)\log P(y_{super}) \quad (۲.۳)$$

احتمال پیشین هر ابردسته، متناسب با تعداد دسته‌های پایه‌ی آن در دادگان مبدا محاسبه شده است. در نهایت با شبه‌برچسب‌های بدست آمده که حاصل از دادگان پشتیبان در فاز متاآزمون هستند، به تنظیم مجدد وزن‌های شبکه‌ی f_φ می‌پردازیم. با توجه به اینکه از یک دسته داده برای تعیین برچسب و در نهایت تنظیم وزن‌ها استفاده می‌کنیم، اسم این روش را تنظیم ترکیبی [4] گذاشته‌ایم. پارامترهای دسته‌بند درشت‌دانه بعد از اعمال تنظیم ترکیبی را با ϕ' و φ' نشان می‌دهیم.

۳-۳ یادگیرنده نمونه محدود در سطح ریزدانه

برای دسته‌بندی در سطح دسته‌های پایه از متایادگیری مبتنی بر متر ۲-۴.۲ و به طور خاص شبکه‌ی پروتوتیپیکال [۶] استفاده کرده‌ایم. اما در این شبکه، اطلاعات تخمین دسته‌بند درشت‌دانه را ترکیب می‌کنیم. استفاده از اطلاعات حاصل از دسته‌بند درشت دانه توسط دو رویکرد اعمال شده است.

۱. شبکه‌ی MLP ریزدانه یعنی f_ϑ بر روی پیش‌بینی نهایی شبکه‌ی دسته‌بند درشت‌دانه یعنی f_φ مشروط می‌شود. با توجه به این اطلاعات اضافه، یادگیری نمونه محدود در سطح ریزدانه به صورت یک یادگیری چند وظیفه‌ای [5] انجام می‌شود. در واقع پیش‌بینی f_φ که تعیین می‌کند یک نمونه متعلق به چه ابردسته‌ای است، به عنوان شناسه‌ی وظیفه [6] عمل می‌کند و به شبکه‌ی ریزدانه می‌فهماند باید روی کدام حوزه‌ی درشت‌دانه تمرکز کند. یعنی از یک رویکرد ترکیب خبره‌ها [7] در یادگیری چند وظیفه‌ای استفاده می‌کنیم

[1] Pseudo Label
[2] Likelihood
[3] Prior
[4] Mixed Tuning
[5] Multi Task
[6] Task Identifier
[7] Mixture of Experts

و هر ابردسته را یک وظیفه‌ی خاص در نظر می‌گیریم که برای حل آن از یک خبره استفاده می‌کنیم. برای رسیدن به این هدف، از یک شبکه‌ی مشترک f_ϑ استفاده می‌کنیم که بتواند اطلاعات متداول بین خبره‌ها را یاد بگیرد و علاوه بر آن، با مشروط شدن روی یک مشخصه‌ی وظیفه که از سطح درشت‌دانه می‌آید، به دسته‌بندی نمونه محدود در سطح یک ابردسته خاص بپردازد.

۲. برای کاهش خطر [1] دسته‌بندی اشتباه در تنظیمات بزرگ مقیاس که تعداد زیادی دسته‌ی هدف داریم، برای هر داده‌ی پرسش در ابتدا محتمل‌ترین ابردسته‌اش را مشخص می‌کنیم و به این وسیله دسته‌های پایه‌ای که می‌توانیم تخمین بزنیم را محدود به فرزندان آن ابردسته می‌کنیم. به همین منظور، ابتدا دسته‌بندی درشت‌دانه برای یک داده‌ی پرسش انجام می‌شود و w ابردسته‌ی برنده مشخص می‌شوند. در نتیجه‌ی آن، می‌توان دسته‌های پایه که متعلق به این w ابردسته نیستند را تحت یک راهبرد [2] یکی از همه برنده [3] ، کنار گذاشت.

به طور ساده شده، ما به دنبال $P(y_{base} = c|q)$ هستیم. اما با توجه به بدیع بودن دسته‌های مقصد، از یک رویکرد مولد برای رسیدن به این احتمال استفاده می‌کنیم. به همین جهت و با به کار گرفتن قانون بیز و استقلالات شرطی، به رابطه‌ی زیر می‌رسیم.

(۳.۳)

$$P(y_{base} = c|q, \{\mathcal{S}\}, f_\vartheta, g_\theta, f_{\varphi'}, g_{\phi'}) \propto \sum_{\hat{y}_{super}=1}^{|\mathcal{Y}_{super}|} P(q|\mathcal{S}^c, f_\vartheta, g_\theta, \hat{y}^i_{super}) P(\hat{y}^i_{super}|q, f_{\varphi'}, g_{\phi'}).$$

مراحل دقیق و روابطی که از آن‌ها معادله‌ی ۳.۳ بدست می‌آید را گام به گام طی می‌کنیم.

$$\begin{aligned}
&P(y_{base} = c|q, \{\mathcal{S}\}, f_\vartheta, g_\theta, f_{\varphi'}, g_{\phi'}) = \\
&\quad \sum_{\hat{y}_{super} \in \mathcal{Y}_{super}} P(y_{base} = c, \hat{y}_{super}|q, \{\mathcal{S}\}, f_\vartheta, g_\theta, f_{\varphi'}, g_{\phi'}) = \\
&\quad \sum_{\hat{y}_{super} \in \mathcal{Y}_{super}} P(y_{base} = c|q, \{\mathcal{S}\}, f_\vartheta, g_\theta, f_{\varphi'}, g_{\phi'}, \hat{y}_{super}) P(\hat{y}_{super}|q, \{\mathcal{S}\}, f_\vartheta, g_\theta, f_{\varphi'}, g_{\phi'}) \propto \\
&\quad \sum_{\hat{y}_{super} \in \mathcal{Y}_{super}} P(y_{base} = c|q, \{\mathcal{S}\}, f_\vartheta, g_\theta, \hat{y}_{super}) P(\hat{y}_{super}|q, f_{\varphi'}, g_{\phi'}) \propto \\
&\quad \sum_{\hat{y}_{super} \in \mathcal{Y}_{super}} P(q|\mathcal{S}^c, f_\vartheta, g_\theta, \hat{y}_{super}) P(\hat{y}_{super}|q, f_{\varphi'}, g_{\phi'}).
\end{aligned}$$

(۴.۳)

در معادله‌ی ۴.۳ ابتدا در خط اول احتمال شرطی که با داشتن یک داده‌ی پرسش q به دنبال آن هستیم را آورده‌ایم و در خط دوم، یک احتمال توأمان با وارد کردن متغیر تصادفی $\hat{y}_{super}$ نوشته‌ایم و برای حذف اثر آن، روی تمامی حالات این متغیر تصادفی جمع زده‌ایم و به اصطلاح marginalize out کرده‌ایم. در خط سوم با استفاده از قاعده‌ی

[1] Risk
[2] Strategy
[3] Winnet Takes All

زنجیره‌ای [1] احتمالی که از قانون بیز به‌دست می‌آید، متغیر تصادفی $\hat{y}_{super}$ را به قسمت شرطی برده‌ایم. سپس از استقلال‌های شرطی موجود استفاده کرده‌ایم و به خط چهارم این معادله رسیده‌ایم. به عنوان مثال، با داشتن $\hat{y}_{super}$ یک زنجیره‌ی علی [2] طبق گراف احتمالاتی این مسئله داریم و طبق قواعد D-Seperation از شبکه‌های قسمت ϕ' مستقل می‌شویم. برای رسیدن به خط آخر فرض کرده‌ایم احتمالات دسته‌های پایه یعنی $P(y_{base} = c)$ به صورت یکنواخت [3] است و به همین جهت می‌شود از آن صرف نظر کرد. این موضوع را به طور دقیق‌تر در معادله‌ی زیر نشان داده‌ایم.

$$\begin{aligned} P(y_{base} = c|q, \{\mathcal{S}\}, f_\vartheta, g_\theta, f_{\varphi'}, g_{\phi'}) &\propto P(q|y_{base} = c, \{\mathcal{S}\}, f_\vartheta, g_\theta, f_{\varphi'}, g_{\phi'})P(y_{base} = c) \\ &\propto P(q|\mathcal{S}^c, f_\vartheta, g_\theta, f_{\varphi'}, g_{\phi'}). \end{aligned} \tag{۵.۳}$$

در نتیجه به رابطه‌ی بدست آمده در معادله‌ی ۳.۳ می‌رسیم.

در طول آزمایش‌ها مشاهده کردیم که توزیع احتمالاتی $P(\hat{y}_{super}|q, f_{\varphi'}, g_{\phi'})$ برای یک ابردسته به صورت شدیدی بر ابردسته‌های دیگر غلبه می‌کند. بنابراین برای رسیدن به پیچیدگی محاسباتی کمتر، در معادله‌ی ۳.۳ احتمال شرطی برای ابردسته‌های مغلوب را برابر صفر در نظر گرفته‌ایم. به همین دلیل، رابطه‌ی ۳.۳ را با احتمال ابردسته‌ی برنده یا w ابردسته‌ی برنده تقریب می‌زنیم و سیگما را برای کل $\hat{y}_{super}$ ها محاسبه نمی‌کنیم. با این تقریب به رابطه‌ی زیر می‌رسیم که احتمال مولد یک نمونه‌ی پرسش با داشتن اطلاعات ابردسته‌ی برنده است.

$$P(q|\mathcal{S}^c, f_\vartheta, g_\theta, f_\varphi, g_\phi, \hat{y}_{super,winner}) \tag{۶.۳}$$

فرض می‌کنیم هر دسته‌ی پایه یک توزیع احتمال گاوسی کروی دارد و به این ترتیب درست‌نمایی وقوع یک پرسش به ازای یک دسته‌ی پایه را مدل می‌کنیم. با داشتن $\mathcal{S}^c$ که نمونه‌های پشتیبان یک دسته در یک ایپزود هست، نماینده‌ی هر دسته را طبق مدل پروتوتیپیکال به صورت زیر محاسبه می‌کنیم.

$$\mathcal{P}_{\mathcal{S}^j} = \frac{۱}{|\mathcal{S}^j|} \sum_{x^i \in \mathcal{S}^j} f_\vartheta \left(g_\theta(x^i), f_{\varphi'}(g_{\phi'}(x^i))\right) \tag{۷.۳}$$

در نهایت پیش‌بینی دسته‌ی پایه با در نظر گرفتن مدل‌سازی گاوسی و اعمال نرمال‌سازی سافت‌مکس به صورت زیر محاسبه می‌شود.

$$\begin{aligned} &P(y_{base} = c|q, \{\mathcal{S}\}, f_\vartheta, g_\theta, f_\varphi, g_\phi) \propto \\ &\frac{P(q|\mathcal{S}^c, f_\vartheta, g_\theta, f_\varphi, g_\phi, \hat{y}_{super,winner})}{\sum\limits_{j \in \mathcal{Y}_{base}} P(q|\mathcal{S}^j, f_\vartheta, g_\theta, f_\varphi, g_\phi, \hat{y}_{super,winner})} = \\ &\frac{\exp\left(-\|f_\vartheta\left(g_\theta(q), f_\varphi(g_\phi(q))\right) - \mathcal{P}_{\mathcal{S}^c}\|^۲\right)}{\sum\limits_{j \in \mathcal{Y}_{base}} \exp\left(-\|f_\vartheta\left(g_\theta(q), f_\varphi(g_\phi(q))\right) - \mathcal{P}_{\mathcal{S}^j}\|^۲\right)}. \end{aligned} \tag{۸.۳}$$

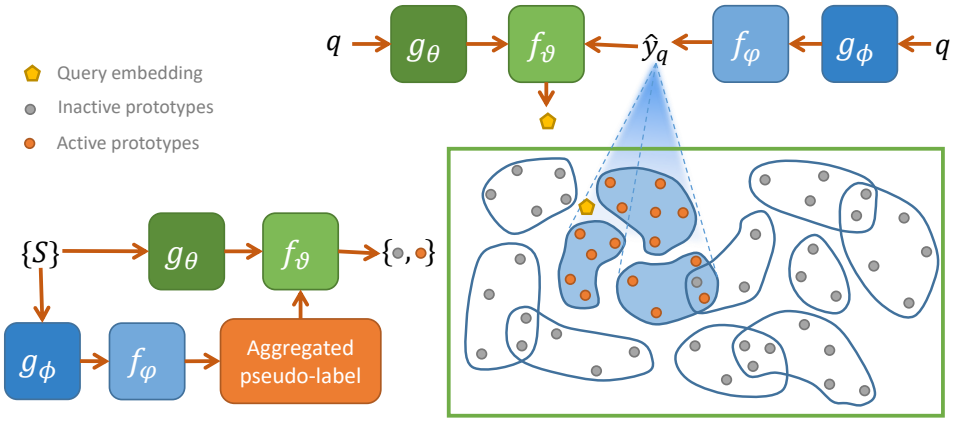
[1] Chain Rule
[2] Causal Chain
[3] Uniform

با محاسبه‌ی احتمال بدست‌آمده در رابطه‌ی ۸.۳ برای همه‌ی دسته‌های پایه، پیش‌بینی نهایی برابر است با

$$\underset{y_{base}}{\operatorname{argmax}}\ P(y_{base}|q, \{\mathcal{S}\}, f_\vartheta, g_\theta, f_\varphi, g_\phi) \tag{۹.۳}$$

در فاز متاآموزش و برای آموزش شبکه با اعمال بهینه‌سازی براساس تابع خطای بی‌نظمی متقابل [1] روی احتمالات بدست آمده از ۸.۳ ، پارامترهای شبکه‌ی f_ϑ به روز رسانی می‌شوند. نکته‌ی قابل توجه در معماری f_ϑ این است که مشابه معماری قسمت قابل آموزش در سطح درشت‌دانه، از دو لایه تبدیل خطی و یک تابع فعال‌ساز [2] GELU در بین آن‌ها استفاده کرده‌ایم. این انتخاب مشابه قسمت درشت‌دانه، به دلیل داشتن پیچیدگی محاسباتی پایین و سرعت بالا در آموزش سطح ریزدانه بوده‌است. از طرفی، به دلیل نمونه محدود بودن مسئله، با اضافه شدن پارامترهای شبکه و افزایش قدرت و پیچیدگی آن، دچار بیش برازش روی مجموعه داده‌های آموزش می‌شویم. این موضوع به دلیل بزرگ مقیاس بودن مسئله که سبب تغییرات بسیار جزیی بین دسته‌های ریزدانه می‌شود، نمود بیشتری دارد.

در ادامه به جمع‌بندی روش ارائه شده و بیان شبه‌کدهای مراحل آموزش سطوح درشت‌دانه و ریزدانه پرداخته‌ایم. جهت شهود بهتر، روش پیشنهادی در شکل ۳-۲ آمده‌است.



شکل ۳-۲: نمای کلی روش پیشنهادی؛ در قسمت بالا، دسته‌بند یک پرسش q را به عنوان ورودی گرفته‌است. دسته‌بندهای درشت‌دانه‌ی g_ϕ و f_φ که با رنگ‌آبی مشخص شده‌اند، محدوده‌ی دسته‌های پایه را محدود می‌کنند تا با یک رویکرد بالا به پایین، دقت دسته‌بندهای ریز دانه‌ی g_θ و f_ϑ را با محدودکردن دسته‌های ممکن، بالا ببرند. در شکل پایین سمت چپ، به ازای داده‌های پشتیبان یک دسته، شبه‌برچسب مشابه روندی که در تنظیم ترکیبی داشتیم، به‌دست می‌آید و با مشروط کردن شبکه‌ی ریزدانه روی شبه‌برچسب، نماینده‌ی یک دسته ساخته می‌شود. در شکل پایین سمت راست، نماینده دسته‌های فعال که مربوط به w ابردسته‌ی برنده هستند با رنگ قرمز نشان داده شده‌اند و باقی نماینده دسته‌های موجود در کانتور ابردسته‌های دیگر غیرفعال اند.

[1] Cross Entropy Loss
[2] Activation Function

۳-۴ جمع‌بندی روش و ارائه‌ی شبه‌کد

در این فصل به ارائه‌ی روش پیشنهادی برای حل مسئله‌ی دسته‌بندی بزرگ مقیاس نمونه محدود پرداختیم. هدف این روش پوشش تمامی چالش‌های موجود در این مسئله اعم از مجموعه باز بودن، توجه به تغییر توزیع زیرجمعیت، داده کارآمدی و استفاده از سلسله‌مراتب ذاتی داده‌ها بوده است. برای این منظور، یک تنظیمات جدید مطابق شکل ۳-۱ پیشنهاد داده‌ایم که دسته‌های ریزدانه در مبدا و مقصد هیچ اشتراکی در سطح داده و دسته نداشته باشند تا چالشی‌ترین حالت مسئله را حل کنیم. برای ترکیب اطلاعات سطوح درشت‌دانه و ریزدانه از یک رویکرد بیزی استفاده کردیم. نیاز به داشتن برچسب درشت‌دانه برای داده‌های پشتیبان مجموعه‌ی مقصد را با یک رویکرد نیمه‌نظارتی از بین بردیم. با توجه به مشاهدات و آزمایش‌ها، از تکنیک‌هایی برای رسیدن به پیچیدگی محاسباتی و زمان پایین‌تر آموزش بهره بردیم. از یک رویکرد مولد برای دسته‌بندی نهایی با توجه به بدیع بودن تمامی دسته‌ها در فاز متاآزمون استفاده کردیم. در نهایت روش ارائه شده به نحوی است که علاوه بر فاز متاآزمون، در متاآموزش هم برخلاف روش‌های دیگر نیازی به مجموعه دادگان با نمونه‌های زیاد ندارد.

در ادامه، برای مشخص شدن بهتر جزییات آموزش دسته‌بندهای سطح درشت‌دانه و ریزدانه، شبه‌کد آن‌ها را آورده‌ایم.

Algorithm 2 Coarse-grained classifier (training)

Input: dataset $\mathcal{D}_{source}$ and $\mathcal{D}_{target}$, backbone g_ϕ, classification head f_φ

Output: backbone $g_{\phi'}$, classification head $f_{\varphi'}$

while not converged **do** *//Supervised learning*
  sample batch $\{(x^i, y^i_{super})\}_{i=1}^{B} \sim \mathcal{D}_{source}$
  $\hat{y}^i_{super} \leftarrow f_\varphi(g_\phi(x^i))$ *//Do this for all of the batch samples*
  update ϕ and φ based on $\frac{1}{B}\sum_{i=1}^{B} \mathcal{L}(y^i_{super}, \hat{y}^i_{super})$
end while
initialize $\phi' \leftarrow \phi, \varphi' \leftarrow \varphi$
for $c \in \mathcal{Y}_{base,target}$ **do** *//Semi-supervised learning*
  take all support samples $\mathcal{S}^c$
  $\tilde{y}_{super} \leftarrow \operatorname{argmax} \prod_{x^i \in \mathcal{S}^c} f_{\varphi'}(g_{\phi'}(x^i))$
  $\hat{y}^i_{super} \leftarrow f_{\varphi'}\left(g_{\phi'}\left(x^i\right)\right)$ *//Do this for all of the support samples*
  update ϕ' and φ' based on $\frac{1}{|\mathcal{S}^c|}\sum_{i=1}^{|\mathcal{S}^c|} \mathcal{L}(\tilde{y}_{super}, \hat{y}^i_{super})$
end for

Algorithm 3 fine-grained classifier (training)

Input: dataset $\mathcal{D}_{source}$ and $\mathcal{D}_{target}$, backbone g_θ, classification head f_ϑ

Output: backbone $g_{\theta'}$, classification head $f_{\vartheta'}$

1: **while** not converged **do** *//Conditional prototypical learning*

2: sample $\{\mathcal{S}^j, \mathcal{Q}^j\}_j \sim \mathcal{D}_{source}$ and create a K-way N-shot task $\mathcal{T}$

3: calculate class conditional probabilities based on Eq. 3.3

4: calculate class probabilities based on Eq. 3.8

5: update θ' and ϑ' based on cross-entropy loss

6: **end while**

فصل ۴

ارزیابی

مطالب این فصل

۴-۱ معرفی مجموعه داده‌های مورد استفاده ۴۲
۴-۲ معرفی بدنه استخراج‌گر ویژگی ۴۴
۴-۳ جزییات معماری سطح درشت‌دانه ۴۵
۴-۴ جزییات معماری سطح ریزدانه ۴۵
۴-۵ روش‌های پایه [1] و نحوه‌ی مقایسه ۴۶
۴-۶ نتایج ۴۷
۴-۷ منابع سخت‌افزاری ۵۱
۴-۸ جمع‌بندی ۵۱

در این فصل با انجام ارزیابی‌ها و آزمایش‌های متعدد کارایی مدل پیشنهادی را نشان می‌دهیم. این آزمایش‌ها را بر روی گستره‌ای از بدنه‌های پیش‌آموزش دیده انجام داده‌ایم تا عملکرد روش را در حضور بدنه‌های ضعیف تا قوی بسنجیم. علاوه بر آن، با مقایسه عملکرد مدل از جنبه‌های مختلف و در تنظیمات مختلف، جامعیت راه‌حل نشان داده شده‌است. از طرفی، توضیح داده‌ایم چگونه مجموعه دادگان بزرگ مقیاس را در قالب تنظیمات رویکرد خود، تطابق داده‌ایم.

۴-۱ معرفی مجموعه داده‌های مورد استفاده

در مسائل بزرگ مقیاس، فراهم کردن یک مجموعه دادگان با تعداد دسته‌ی زیاد زحمت زیادی را می‌طلبد. دلیل این موضوع سختی فرآیند برچسب‌زنی [1] و مناسب‌سازی [2] داده‌ها است. در نتیجه‌ی این موارد، در مسائل حوزه‌ی بینایی کامپیوتر تعداد محدودی مجموعه داده با تعداد دسته‌ی بیش از ۱۰۰۰ وجود دارد. علاوه بر این، چالش‌های دیگری در ساختار و فرآیند شکل‌گیری مجموعه دادگان بزرگ مقیاس دیده می‌شود. اینگونه مجموعه دادگان، معمولا

[1] Baselines
[1] Annotation
[2] Curation

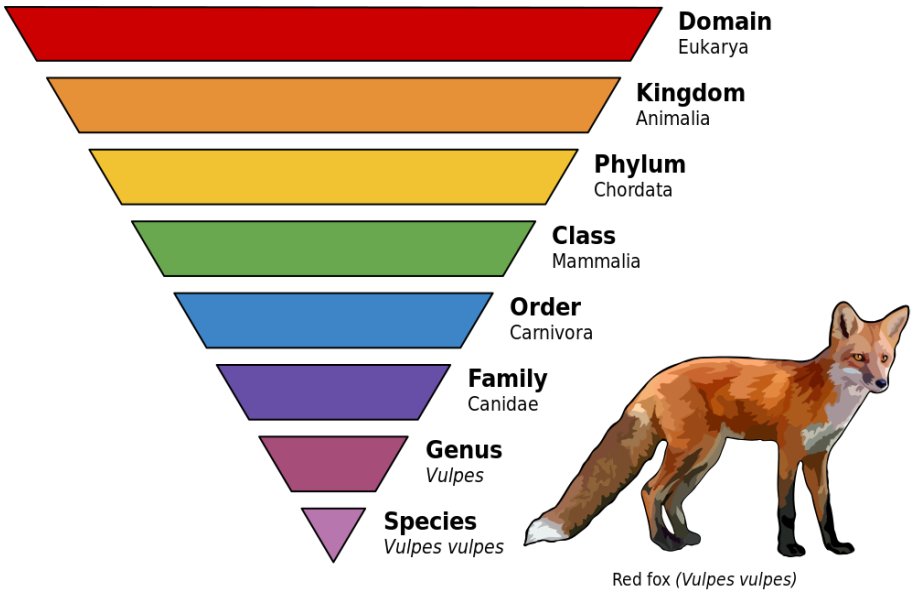
به صورت عام منظوره جمع‌اوری شده‌اند و بین نمونه‌های دسته‌ها، انسجام کافی وجود ندارد و در واقع بین دسته‌ها وابستگی متقابل [1] دیده می‌شود.

روش پیشنهادی ما به وجود سلسله‌مراتب در فاز متاآموزش نیاز دارد، به همین جهت تمرکزمان روی مجموعه‌داده‌هایی بوده که از پیش یک سلسله‌مراتب از پیش تعریف شده داشته‌اند. همانطور که در بخش ۳-۱ بیان شد، دسته‌های پایه در مبدا و مقصد (فازهای متاآموزش و متاآزمون) ناسازگار اند و از لحاظ داده و برچسب هیچ اشتراکی ندارند، اما ابردسته‌های متاآزمون زیرمجموعه‌ای از ابردسته‌های فاز متاآموزش هستند. به همین جهت، ما یک تغییر توزیع زیرجمعیت [2] را تحمیل کرده‌ایم که مقاومت مدل را در تطابق نمونه محدود به داده‌های مقصد بسنجیم. از این رو، ما از دو مجموعه داده که حاوی سطوح ریزدانگی متفاوت [3] اند، بهره برده‌ایم و آن‌ها را با تنظیمات خود تطابق داده‌ایم. مجموعه دادگانی که استفاده کردیم عبارت اند از iNat Animalia و iNat Plantae [۵۹] . نکته‌ی بسیار مهمی که در استفاده از بدنه‌های پیش‌آموزش دیده‌ی بنیادی باید رعایت کنیم، این است که مجموعه دادگانی انتخاب شوند که در فرآیند پیش‌اموزش آن بدنه‌ها به کار نرفته باشند و در واقع نمونه‌ها از پیش دیده نشده باشند؛ چرا که می‌خواهیم قابلیت یادگیری نمونه محدود مدل در حضور دسته‌های بدیع را به صورت واقعی بسنجیم. در زیر به معرفی جزییات مجموعه دادگان مورد استفاده می‌پردازیم.

- **iNat Animalia** یک بخش از مجموعه دادگان بزرگ مقیاس iNaturalist-2021 است که در مجموع ۱۰k دسته دارد. در این مجموعه دادگان، دسته‌های مربوط به جانداران جمع‌آوری شده‌است. به ازای هر دسته‌ی پایه در ریزدانه‌ترین سطح ممکن، تمامی پدران آن براساس دسته‌بندی شناخته‌شده‌ی Taxonomic Rank ارائه شده‌است که در شکل ۴-۱ آمده‌است. این سلسله‌مراتب در iNat از سطح Kingdom به عنوان درشت‌دانه‌ترین شروع می‌شود و تا Species به عنوان ریزدانه‌ترین ادامه می‌یابد. در واقع iNat در سطح Kingdom شامل سه دسته‌ی Animalia ، Plantae و Fungi است که به ترتیب شامل جانوران، گیاهان و قارچ‌ها می‌شوند. ما دسته‌های مربوط به Fungi را وارد آزمایش نکرده‌ایم چرا که بسیار محدود اند و در حدی نیستند که به عنوان مجموعه بزرگ مقیاس به آن نگاه کنیم. به قسمت مرتبط با حیوانات این مجموعه دادگان به صورت iNat Animalia اشاره می‌کنیم که گستره‌ی وسیعی از گونه‌های جانوری را دربر می‌گیرد. به جهت تعریف سطح درشت‌دانه و ابردسته‌ها، پنجمین سطح درخت سلسله‌مراتب که به سطح Family مشهور است را به عنوان ابردسته‌ها مشخص کرده‌ایم که ۷۴ ابردسته حاصل شده‌است. از بین دسته‌های پایه در ریزدانه‌ترین سطح، ۱۷۲۸ دسته را به عنوان مجموعه‌ی مبدا و ۱۵۰۰ دسته را به عنوان مجموعه‌ی مقصد و برای فاز متاآزمون جدا کرده‌ایم. یعنی در مجموع ۳۲۲۸ دسته‌ی پایه داریم که بین متاآموزش و متاآزمون به گونه‌ای تقسیم شده اند که از هر ابردسته به تعداد تقریبا مساوی در هر دو طرف باشد. این موضوع به دلیل تحمیل تغییر توزیع زیرجمعیت است.

- **iNat Plantae** قسمتی از مجموعه داده‌های iNaturalist-2021 است که گونه‌های گیاهی را پوشش می‌دهد. این مجموعه شامل ۲۷۰۵ دسته‌ی پایه است که مطابق آنچه در قسمت قبل توضیح دادیم، بین مجموعه‌ی مبدا و مقصد به گونه‌ای تقسیم شده که هر ابردسته به تعداد تقریبا مساوی دسته‌ی پایه در فاز متاآموزش و متاآزمون داشته باشد. به این ترتیب به یک تقسیم‌بندی ۱۴۵۰ دسته برای مجموعه‌ی مبدا و ۱۲۵۵ دسته

[1] Interdependency
[2] Sub-Population Shift
[3] Multi Granular



شکل ۴-۱: دسته‌بندی زیستی موجودات مطابق با Taxonomic Rank ؛ از بالا به پایین سطوح درشت‌دانه تا ریزدانه چیده شده‌اند و سلسله‌مراتب مربوط به روباه قرمز در طول درخت سلسله‌مراتب مشخص شده‌است [۱۶].

برای مجموعه‌ی مقصد در قالب ۵۸ ابردسته‌ی مشترک رسیده‌ایم. ابردسته‌ها از سطح Family در درخت سلسله‌مراتب انتخاب شده‌اند. در جدول ۴-۱ جزییات مجموعه دادگان ساخته شده آمده‌است.

جدول ۴-۱: جزییات کمی مجموعه دادگان iNat Animalia و iNat Plantae و تقسیم‌بندی آن‌ها در قالب تنظیمات مسئله.

مشخصه‌ها / مجموعه دادگان	تعداد ابردسته‌ها	تعداد دسته‌های پایه در مجموعه مبدا	تعداد دسته‌های پایه در مجموعه مقصد
iNat Animalia	۷۴	۱۷۲۸	۱۵۰۰
iNat Plantae	۵۸	۱۴۵۰	۱۲۵۵

۴-۲ معرفی بدنه استخراج‌گر ویژگی

در قسمت ۳-۱ اشاره شد که بخشی از رویکرد پیشنهادی در قسمت‌های g_θ و g_ϕ از بدنه‌ی مدل‌های بنیادی پیش‌آموزش دیده برای رسیدن به کارایی داده استفاده می‌کند. در این قسمت، به معرفی بدنه‌های بنیادی استفاده شده می‌پردازیم.

Dino Giant بزرگترین مدل در سری مدل‌های Dino V۲ است [۶۰] . این سری از مدل‌ها با یک رویکرد یادگیری خودنظارتی با استفاده از فرآیند تقطیر دانش [1] آموزش داده شده است. این مدل بازنمایی ویژگی‌های با کارایی بسیار بالا تولید می‌کند و در عمل مورد توجه و استفاده‌ی زیادی قرار گرفته‌است. از این خانواده در مسائل

[1] Knowledge Distillation

با جنس گوناگون مثل دسته‌بندی، تخمین عمق تصاویر، قطعه‌بندی تصاویر و ... استفاده شده و نتایج بسیار خوبی گزارش شده‌است. مجموعه دادگان آموزشی مدل‌های Dino شامل ۱۴۲ میلیون تصویر است. یک مدل دیگر از این خانواده که ما از آن بهره برده‌ایم، مدل **Dino Large** است که فرآیند آموزش مشابهی را طی می‌کند اما از لحاظ تعداد پارامتر به اندازه‌ی $\frac{1}{3}$ مدل Giant است.

مدل **ConvNext** $V2$ دارای معماری تمام پیچشی [1] است که به عملکردی مشابه مبدل‌های تصویری [2] رسیده‌است [۶۱] . ما از این بدنه که بر روی مجموعه دادگان بزرگ ImageNet21K آموزش دیده بهره برده‌ایم. این مجموعه دادگان در واقع نسخه‌ی کامل شده‌ی مجموعه‌ی معروف ImageNet1K است [۶۲] . **MobileNet** $V3$ بدنه‌ی دیگری است که ما انتخاب کرده‌ایم تا کارایی روش خود را روی بدنه‌های به نسبت کوچک هم نشان دهیم [۶۳] . وزن‌های پیش‌آموزش دیده برای این مدل را از [۶۴] برداشته‌ایم. بدنه‌ی بنیادی دیگری که از آن بهره برده‌ایم مدل CLIP است که از یادگیری تباینی برای یاد گرفتن یک فضای مشترک بین متن و تصویر بهره می‌برد و در قسمت ۲-۸ معرفی شد.

۴-۳ جزییات معماری سطح درشت‌دانه

سطح درشت‌دانه متشکل از g_ϕ است که از بدنه‌ی منجمد Dino Giant تشکیل شده‌است و بر روی آن یک بخش قابل آموزش f_φ اضافه می‌شود که یک تبدیل خطی ساده به وسیله‌ی یک لایه‌ی Linear است. این لایه ابعاد خروجی حاصل از بدنه‌ی منجمد را به ابعاد $|\mathcal{Y}_{super}|$ می‌برد. قسمت f_φ بر روی دادگان $\mathcal{D}_{source}$ با سایز بسته [3] برابر ۶۴ آموزش داده می‌شود. این آموزش به صورت یادگیری با ناظر استاندارد با برچسب‌های سطح درشت‌دانه یعنی $\mathcal{Y}_{super}$ انجام می‌شود. بعد از این مرحله، طبق فاز دوم الگوریتم ۲ ، مرحله‌ی آموزش با داشتن اطلاعات نیمه‌نظارتی از دادگان $\mathcal{D}_{target}$ در مرحله‌ی متاازمون آغاز می‌شود. برای این منظور ۵ شات از دادگان ساپورت هر دسته برداشته می‌شود و به صورت گروهی یک شبه‌برچسب برای آن‌ها طبق روش تنظیم ترکیبی که در قسمت ۳-۲ ارائه دادیم، تعیین می‌شود. با استفاده از این شبه‌برچسب‌ها وزن‌های مدل f_φ به روز رسانی می‌شود. در جدول ۴-۲ تاثیر اعمال تنظیم ترکیبی به صورت نیمه نظارتی بر روی صحت [4] مدل سطح درشت‌دانه مشخص شده‌است. به دلیل اینکه دقت شبه‌برچسب‌ها از دقت ۱ـبهترین شبکه‌ها بالاتر است، تنظیم بهینه روی شبه‌برچسب‌ها موجب افزایش دقت مدل به خصوص روی مجموعه دادگان Plantae شده‌است.

۴-۴ جزییات معماری سطح ریزدانه

در این قسمت، از بدنه‌هایی که در قسمت ۴-۲ معرفی شد، در قالب g_θ استفاده کرده‌ایم. بر روی این بدنه، از دو لایه خطی با یک تابع فعال‌ساز [5] GELU در بین آن‌ها به عنوان شبکه‌ی f_ϑ بهره برده‌ایم. در طول فاز متاآموزش،

[1] Fully Convolutional
[2] Visual Transformers
[3] Batch Size
[4] Accuracy
[5] Activation Function

جدول ۴-۲: صحت شبکه‌ی سطح درشت‌دانه متشکل از بدنه‌ی g_ϕ و قسمت قابل آموزش f_φ بعد از آموزش بر روی داده‌های $\mathcal{D}_{super,source}$ و بعد از تنظیم ترکیبی به صورت نیمه‌نظارتی روی داده‌های $\mathcal{D}_{super,target}$ نمایش داده شده‌است. دقت‌ها چه قبل و چه بعد از تنظیم ترکیبی، بر روی دادگان مجموعه‌ی مقصد و با تحمیل تغییر توزیع زیرجمعیت محاسبه شده‌است که از تنظیمات آزمون عادی سخت‌تر است.

تنظیمات / مجموعه دادگان	۱-بهترین	۵-بهترین	صحت شبه‌برچسب	۱-بهترین تنظیم ترکیبی	۵-بهترین تنظیم ترکیبی	بهره نسبی ۱-بهترین	بهره نسبی ۵-بهترین
iNat Animalia	۸۷/۳۸	۹۷/۲۶	۹۵/۸۶	۸۸/۸۸	۹۷/۸۳	+۱/۷	+۰/۶
iNat Plantae	۷۹/۶۸	۹۲/۴۵	۹۰/۵۱	۸۷/۸۶	۹۷/۱	+۱۰/۳	+۵

بدنه‌های Dino و CLIP با وزن‌های منجمد شده استفاده شده‌اند اما شبکه‌ی MobileNet به علت سبک بودن، تنظیم بهینه شده‌است. در مورد شبکه‌ی ،ConvNext تمامی وزن‌های بدنه منجمد شده‌اند و تنها چیزی که به روز رسانی شده‌است، وضعیت لایه‌های نرمال‌سازی بر اساس داده‌های متاآموزش است.

در آموزش سطح ریزدانه، از روش یادگیری پروتوتیپیکال که اصلی‌ترین روش متایادگیری مبتنی بر متر است استفاده کرده‌ایم. به این منظور، یک یادگیری اپیزودیک را ترتیب داده‌ایم که که در هر اپیزود یک وظیفه‌ی $\mathcal{T}$ از $\mathcal{D}_{source}$ نمونه‌برداری می‌شود و یک وظیفه‌ی ۶۵۰ راه، ۵ شات را می‌سازد. در واقع در فاز متاآموزش، از وظیفه‌های با تعداد راه خیلی بالا نسبت به متایادگیری متداول استفاده کرده‌ایم. چرا که در فاز متاآزمون، همواره قدرت تعمیم‌پذیری شبکه با ساختن یک وظیفه شامل تمامی دسته‌های پایه از $\mathcal{D}_{target}$ را می‌سنجیم. به بیان دیگر، در فاز متاآزمون یک وظیفه‌ی بزرگ مقیاس و نمونه محدود با داشتن $|\mathcal{Y}_{base,target}|$ راه و ۵ شات نمونه داریم و دقت دسته‌بندی را با نمونه‌های پرسش مرتبط با هر دسته می‌سنجیم.

۴-۵ روش‌های پایه [1] و نحوه‌ی مقایسه

برای بررسی کارایی و تاثیر دو مورد بیان شده در ابتدای بخش ۳-۳ سه نوع مطالعه‌ی فرسایشی [2] با فراهم کردن مدل‌های پایه با تغییر در f_ϑ انجام داده‌ایم.

۱. در فاز متاآموزش، یک حالت را به صورت ورودی دادن بازنمایی‌های حاصل از g_θ به صورت مستقیم به سر دسته‌بند ساخته‌ایم. در حالت دوم، بردار one hot که نمایانگر ابردسته‌ی صحیح است را به صورت چسبیده [3] همراه با بازنمایی حاصل از بدنه، به f_ϑ ورودی می‌دهیم. البته توجه به این نکته لازم است که در فاز متاآزمون، به جای بردار one hot صحیح، از تخمین مدل درشت‌دانه برای ابردسته استفاده می‌کنیم. در واقع مقایسه‌ی این دو حالت تاثیر مشروط کردن اطلاعات ابردسته برای پیش‌بینی دسته‌ی پایه را نشان می‌دهد. به همین دلیل حالت دوم را در جداول با پسوند Cond به معنی مشروط شده نمایش داده‌ایم. برای جمع‌بندی، حالت اول مشابه دسته‌بندی یکی از همه برنده [4] معمولی می‌شود و حالت دوم همین دسته‌بندی با ترکیب اطلاعات ابردسته‌ها در یک فریمورک بیزی است.

[1] Baselines
[2] Ablation Study
[3] Concatenated
[4] Winner Takes All

۲. از طرفی برای سنجش موثر بودن دسته‌بندی سلسله‌مراتبی، مدل پایه اصلی را به صورت یادگیری پروتوتیپیکال به صورت تخت تعریف کرده‌ایم. به این صورت از سلسله‌مراتب و اطلاعات موجود در ابردسته‌ی هر داده هیچ استفاده‌ای نمی‌شود و دسته‌بندی در ریزدانه‌ترین سطح صورت می‌پذیرد.

۳. در نهایت برای سنجش میزان غنای بازنمایی‌های تولید شده توسط بدنه‌ها و تاثیر مستقیم سلسله‌مراتب، با حذف f_ϑ فقط از بازنمایی‌های حاصل از بدنه g_θ در تنظیمات بدون آموزش [1] استفاده کرده‌ایم و به صورت پروتوتیپیکال به دسته‌بندی پرداخته‌ایم. در این رویکرد بدون آموزش، دسته‌بندی در حالت یکی از همه برنده را انجام داده‌ایم و نتایج در حضور این استراتژی گزارش شده‌است.

۴-۶ نتایج

ارزیابی عملکرد روش پیشنهادی روی مجموعه دادگان معرفی شده به ازای بدنه‌های مختلف در این بخش آمده است. تنظیمات فاز تست به صورت متاآزمون است و با فاز آزمون متداول یک شبکه‌ی عصبی فرق دارد. هر عددی که گزارش شده‌است، بر روی دسته‌های بدیع در مجموعه‌ی مقصد است. یعنی در سطح دسته‌های پایه تغییر توزیع زیرمجموعه داریم و هیچ آموزشی روی دسته‌های پایه مقصد صورت نگرفته‌است. در جدول ۴-۳ اطلاعات مربوط به مجموعه دادگان iNat Animalia مشخص شده و در جدول ۴-۴ اطلاعات و ارقام مربوط به مجموعه دادگان iNat Plantae آمده‌است. همانطور که از این جداول مشخص است، نسبت به مدل پایه تخت در همه‌ی موارد افزایش داشته‌ایم. با اینکه این بهبود در بدنه‌های بنیادی خیلی بزرگ مانند Dino که تعداد پارامتر خیلی بالایی دارند، محدود بوده‌است اما در بدنه‌ی بنیادی دیگر مانند CLIP افزایش چشمگیری را شاهد هستیم. از طرفی بر روی بدنه‌های متوسط و کوچک با معماری‌های متفاوت مانند MobileNet و ConvNext هم بهبود قابل توجهی داشته‌ایم. از طرفی استراتژی ترکیب اطلاعات سطح درشت‌دانه با ریزدانه در یک قالب بیزی که با پسوند Cond مشخص شده‌است در همه‌ی موارد موثر بوده و با افزایش صحت مواجه هستیم.

برای سنجش مورد سوم مطرح شده در قسمت ۴-۵ ، آزمایش‌ها را با کنار گذاشتن سر f_ϑ بر روی تمامی بدنه‌ها تکرار کرده‌ایم. این نتایج برای هر مجموعه داده در جداول ۴-۵ و ۴-۶ آمده‌است. حالت پایه دسته‌بندی پروتوتیپیکال بر روی بازنمایی‌های حاصل از بدنه‌ها بدون هیچ آموزشی صورت پذیرفته‌است. همچنین با به کار بردن یک مدل درشت‌دانه آموزش دیده روی مجموعه‌ی مبدا، W ابردسته‌ی محتمل را حدس زده‌ایم و سپس با همان بدنه‌های بدون آموزش، به دسته‌بندی بین دسته‌های پایه این W ابردسته پرداخته‌ایم که در واقع همان حالت یکی از همه برنده است. در این مورد با توجه به جداول، تاثیر سلسله‌مراتبی کردن دسته‌بندی حتی در تنظیمات بدون آموزش مشهود است. با توجه به دقت ستون پیشگویی ابردسته، اگر بر روی دسته‌بند سطح درشت‌دانه بیشتر کار شود، این افزایش بهبود حتی بر روی بدنه‌های خیلی بزرگ مانند Dino هم اثرش را نشان می‌دهد.

همچنین در نمودار ۴-۲ ، به مقایسه‌ی عملکرد بدنه‌های مختلف در سه حالت پایه (تخت)، دسته‌بندی یکی از همه برنده با در نظر گرفتن ۵ ابردسته‌ی تخمینی و دسته‌بندی سلسله‌مراتبی با داشتن برچسب ابردسته به صورت پیشگویی پرداخته‌ایم. در این نمودار هم کارایی روش و افزایش دقت‌ها مشهود است.

[1] Free Training

جدول ۴-۳: عملکرد روش پیشنهادی و مقایسه با مدل‌های پایه روی مجموعه دادگان iNat Animalia ؛ در این جدول، منظور از مدل پایه (تخت) استفاده از روش پروتوتیپیکال در ریزدانه‌ترین سطح است. مدل‌هایی که پسوند Cond دارند به این معنی است که طبق روش ارائه شده، از اطلاعات سطح درشت‌دانه به صورت مشروط استفاده می‌کنند. اعداد ستون آخر یعنی پیشگویی ابردسته برای زمانی است که دقت سطح درشت‌دانه صد درصد باشد و در واقع برچسب‌های ابردسته‌ها را به صورت صحیح بدانیم.

مجموعه دادگان / بدنه	iNat Animalia						
	پایه (تخت)	W ابردسته برنده				بهره نسبی	پیشگویی ابردسته
		۱	۲	۵	۱۰		
DINO Large	۵۳/۵۲	۵۰/۹۵	۵۲/۸۳	۵۳/۵۵	**۵۳/۵۶**	+۶/۳	۵۷/۳۱
DINO Large Cond.		۵۴/۸۳	۵۶/۳۳	**۵۶/۸۷**	۵۶/۸		۶۰/۱۲
DINO Giant	۶۱/۶۸	۵۸/۶۴	۶۰/۷۷	۶۱/۵۵	**۶۱/۶۵**	+۲	۶۴/۴
DINO Giant Cond.		۶۰/۰۷	۶۲/۱۹	۶۲/۸۹	**۶۲/۹۲**		۶۵/۴۸
CLIP	۳۳/۸۸	**۳۹/۱۲**	۳۸/۹	۳۷/۳۶	۳۶/۲۹	+۱۶/۹	۴۴/۹۶
CLIP Cond.		۳۹/۴۸	۳۹/۵۳	۳۹/۵۷	**۳۹/۶۱**		۴۳/۴۸
Mobilenet V۳	۴۱/۳۳	۴۳/۷	**۴۴/۳۹**	۴۳/۷۷	۴۳/۱۵	+۸/۸	۵۰/۲
Mobilenet V۳ Cond.		۴۳/۶	۴۴/۳۹	۴۴/۸	**۴۴/۹۵**		۴۷/۸۸
ConvNext V۲	۴۶/۰۹	۴۶/۱۶	۴۷/۴۹	**۴۷/۶۹**	۴۷/۵۷	+۶/۷	۵۲/۸۳
ConvNext V۲ Cond.		۴۵/۸۵	۴۷/۷۵	۴۸/۸۱	**۴۹/۱۹**		۵۱/۶۸

جدول ۴-۴: عملکرد روش پیشنهادی و مقایسه با مدل‌های پایه روی مجموعه دادگان iNat Plantae ؛ در این جدول، منظور از مدل پایه (تخت) استفاده از روش پروتوتیپیکال در ریزدانه‌ترین سطح است. مدل‌هایی که پسوند Cond دارند به این معنی است که طبق روش ارائه شده، از اطلاعات سطح درشت‌دانه به صورت مشروط استفاده می‌کنند. اعداد ستون آخر یعنی پیشگویی ابردسته برای زمانی است که دقت سطح درشت‌دانه صد درصد باشد و در واقع برچسب‌های ابردسته‌ها را به صورت صحیح بدانیم.

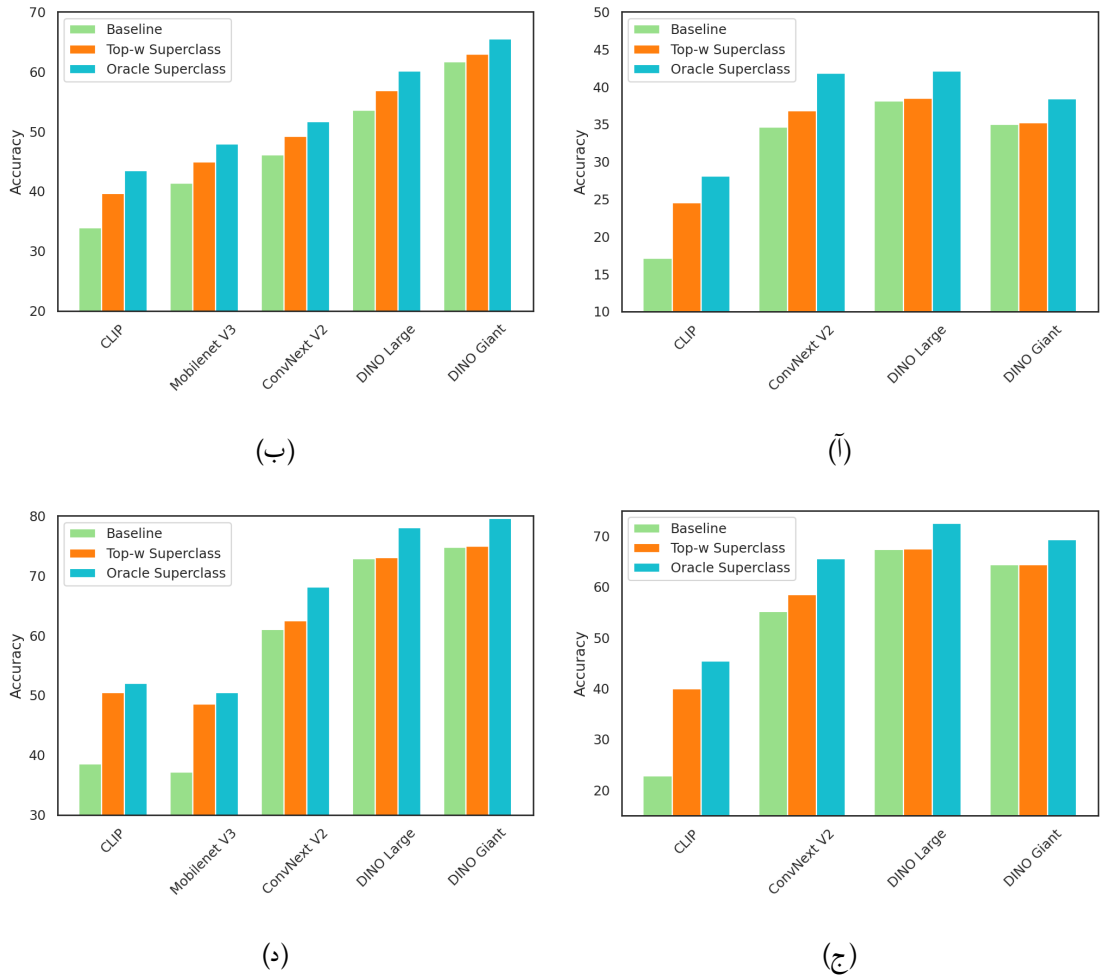
مجموعه دادگان / بدنه	iNat Plantae						
	پایه (تخت)	W ابردسته برنده				بهره نسبی	پیشگویی ابردسته
		۱	۲	۵	۱۰		
DINO Large	۷۲/۸۷	۶۸/۴۸	۷۱/۴۴	۷۲/۶۸	**۷۲/۹۴**	+۰/۲	۷۷/۹۱
DINO Large Cond.		۶۸/۵۷	۷۱/۷۱	**۷۳/۰۲**	۷۳/۰۲		۷۷/۹۹
DINO Giant	۷۴/۷۷	۶۹/۶۶	۷۲/۸۶	۷۴/۴۹	**۷۴/۷۴**	+۰/۳	۷۹/۱۱
DINO Giant Cond.		۶۹/۹۶	۷۳/۲۳	۷۴/۵۸	**۷۴/۹۶**		۷۹/۵۴
CLIP	۳۸/۴۸	**۵۱/۵۴**	۴۹/۶۳	۴۵/۷۵	۴۳/۱۹	+۳۱/۱	۵۸/۹
CLIP Cond.		۵۰/۲۶	۵۰/۴	**۵۰/۴۵**	۵۰/۴۵		۵۲
Mobilenet V۳	۳۷/۱۵	**۴۹/۸۸**	۴۷/۹۴	۴۴/۳۷	۴۱/۴	+۳۰/۵	۵۷/۷۲
Mobilenet V۳ Cond.		۴۸/۲۲	۴۸/۴۶	**۴۸/۴۹**	۴۸/۴۷		۵۰/۴۱
ConvNext V۲	۶۱/۰۲	۶۱/۹۴	**۶۳/۲**	۶۲/۶	۶۲/۱۸	+۲/۴	۷۰/۱۶
ConvNext V۲ Cond.		۶۱/۰۴	۶۲/۴۲	**۶۲/۴۷**	۶۲/۴۷		۶۸/۰۸

جدول ۴-۵: عملکرد روش دسته‌بندی سلسله‌مراتبی یکی از همه برنده در تنظیمات بدون آموزش نسبت به حالت پایه و داشتن برچسب سطح درشت‌دانه به صورت پیشگو مشخص شده‌است. این آزمایش روی دادگان iNat Animalia صورت پذیرفته‌است.

تنظیمات / بدنه	iNat Animalia						
	پایه (تخت)	W ابردسته برنده				بهره نسبی	پیشگویی ابردسته
		۱	۲	۵	۱۰		
DINO Large	۳۸/۱	۳۸/۱۷	**۳۸/۴۸**	۳۸/۲۸	۳۸/۱۷	+۱	۴۲/۱
DINO Giant	۳۵	۳۴/۶۵	**۳۵/۲۳**	۳۵/۱۳	۳۵/۱	+۰/۷	۳۸/۴۴
CLIP	۱۷/۰۹	**۲۴/۵۲**	۲۳/۰۴	۲۱	۱۹/۶۱	+۴۳/۵	۲۸/۱۲
ConvNext V۲	۳۴/۶۵	۳۶/۴۸	**۳۶/۸۴**	۳۶/۳۷	۳۶/۲	+۶/۳	۴۱/۸

جدول ۴-۶: عملکرد روش دسته‌بندی سلسله‌مراتبی یکی از همه برنده در تنظیمات بدون آموزش نسبت به حالت پایه و داشتن برچسب سطح درشت‌دانه به صورت پیشگو مشخص شده‌است. این آزمایش روی دادگان iNat Plantae صورت پذیرفته‌است.

تنظیمات / بدنه	iNat Plantae						
	پایه (تخت)	W ابردسته برنده				بهره نسبی	پیشگویی ابردسته
		۱	۲	۵	۱۰		
DINO Large	۶۷/۴۲	۶۳/۶۶	۶۶/۲۵	۶۷/۳۹	**۶۷/۵۲**	+۰/۲	۷۲/۵۵
DINO Giant	۶۴/۳۷	۶۰/۶۴	۶۳/۰۶	۶۴/۱۳	**۶۴/۴**	+۰/۱	۶۹/۲۶
CLIP	۲۲/۸۲	**۳۹/۹۷**	۳۵/۷۵	۳۰/۹۳	۲۸/۰۶	+۷۵/۲	۴۵/۴۵
ConvNext V۲	۵۵/۲	۵۸/۰۵	**۵۸/۴۹**	۵۷/۴۹	۵۶/۴	+۶	۶۵/۵۷



شکل ۴-۲: مقایسه‌ی دقت‌ها در سه حالت پایه (تخت) با رنگ سبز، دسته‌بندی سلسله‌مراتبی طبق روش پیشنهادی با رنگ نارنجی و دسته‌بندی سلسله‌مراتبی با پیشگویی سطح درشت‌دانه با رنگ آبی؛ کارایی و افزایش صحت در این نمودارها مشخص است. نمودارهای آ و ب مربوط به iNat Animalia و نمودارهای ج و د مربوط به iNat Plantae است. همچنین ستون سمت راست رویکرد بدون آموزش را نشان می‌دهد.

۴-۷ منابع سخت‌افزاری

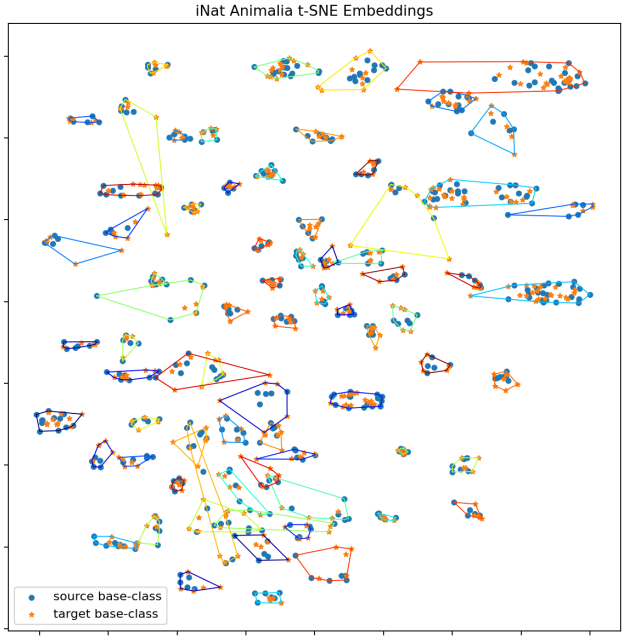
در این پژوهش از پردازنده‌های گرافیکی NVIDIA GeForce RTX 3080 Ti با ۱۲ گیگابایت حافظه‌ی RAM و NVIDIA Quadro RTX 8000 با ۴۸ گیگابایت حافظه‌ی RAM بهره برده‌ایم. برای آموزش مدل‌های سطوح درشت‌دانه از گرافیک ۳۰۸۰ و برای آموزش مدل‌های سطوح ریزدانه از گرافیک Quadro استفاده کرده‌ایم. با توجه به طراحی قسمت قابل آموزش یعنی f_φ و f_ϑ به صورت یک شبکه‌ی بسیار سبک خطی یک یا دو لایه، از بهینگی پیچیدگی محاسباتی برای آموزش با وجود تعدد دسته‌ها استفاده کرده‌ایم. همچنین در پیاده‌سازی کد، انواع روش‌های بهینگی محاسباتی از جمله Cache کردن بازنمایی‌ها به صورت گسترده پیاده‌سازی شده‌است.

۴-۸ جمع‌بندی

در این فصل به ارزیابی و ارائه‌ی نتایج حاصل از آزمایش‌های این پژوهش پرداختیم. در ابتدا مجموعه دادگان مورد استفاده و بدنه‌های بنیادین با وزن‌های پیش‌آموزش دیده معرفی شدند. در ادامه طراحی معماری مدل‌های مورد استفاده در دسته‌بندهای ریزدانه و درشت‌دانه معرفی شدند و در نهایت روش‌های پایه برای مقایسه و نحوه‌ی مقایسه در کنار سنجش کمی عملکرد روش ارائه شد.

نکته‌ی حائز اهمیت در مقایسه این است که ما یک تنظیمات جدید برای مسئله‌ی یادگیری بزرگ مقیاس نمونه محدود ارائه دادیم که تمامی چالش‌های معرفی شده در این حوزه مطابق با جدول ۲-۱ را دربرگیرد. بنابراین مقاله‌ای با این تنظیمات پیش از این وجود نداشت و این یک روش کاملا بدیع است که برای حل این مسئله از سوی ما مطرح شد.

به عنوان جمع‌بندی، طبق آزمایش‌های انجام شده مشخص شد که ساختن بازنمایی با داشتن آگاهی از اطلاعات درشت‌دانه می‌تواند مفید باشد و به افزایش دقت منجر شود. به این منظور بازنمایی‌های حاصل از سطح ریزدانه را با روش t-sne در شکل ۴-۳ آورده‌ایم. با توجه به این شکل، کاملا مشخص است که بازنمایی‌های تولید شده آگاه از اطلاعات ابردسته‌اند و در محدوده‌ی هر ابردسته جای گرفته و به خوبی از هم متمایز شده‌اند. به این ترتیب با مشخص شدن ابردسته‌های محتمل، دسته‌بندی فقط بین دسته‌های پایه‌ی متعلق به آن‌ها صورت می‌پذیرد که از پیچیدگی دسته‌بندی می‌کاهد و به دقت‌های بالاتری می‌انجامد. همچنین با ظهور و بروز دسته‌های پایه‌ی بدیع، می‌توان اظهار داشت که بازنمایی‌های آن‌ها در حیطه‌ی ابردسته‌ی مرتبط جای می‌گیرند که به حل کردن مسئله‌ی تغییر توزیع زیرجمعیت کمک می‌کند.



(آ)

(ب)

شکل ۴-۳: مصورسازی بازنمایی‌های تولید شده برای پروتوتایپ‌های هر دسته‌ی پایه در سطح ریزدانه؛ با توجه به این شکل‌ها مشخص است که دسته‌های مبدا و مقصد متعلق به یک ابردسته در نزدیکی یکدیگر قرار می‌گیرند چرا که شبکه‌ی کدگذار در سطح ریزدانه، بر روی اطلاعات ابردسته مشروط شده است و طوری یادگرفته است که محدوده‌ی هر ابردسته را مشخص کند.

فصل ۵

جمع‌بندی و کارهای آتی

مطالب این فصل

۵-۱ جمع‌بندی و نقاط قوت روش ۵۳

۵-۲ کارهای آتی ۵۴

در فصل ۱ به معرفی مسئله‌ی دسته‌بندی بزرگ مقیاس نمونه محدود و اهمیت آن پرداختیم و به رویکرد حل‌مسئله و چالش‌های موجود در این حوزه اشاره کردیم. در فصل ۲ رویکردها و چالش‌های اصلی این تنظیمات یادگیری را در قالب مرور مقالات مهم هر حوزه بیان کردیم. در واقع در این فصل با مرور پژوهش‌های پیشین، به نوعی ادبیات و دانش پایه مرتبط با این حوزه را در کنار بیان عملکرد هر کدام از مقالات معرفی شد.

در فصل ۳ راهکار پیشنهادی را به طور کامل توضیح دادیم. به این منظور مدل‌سازی تئوری که از مسئله داشتیم در قالب روابط ریاضی بیان شد و برای واضح شدن رویکرد یادگیری، شبه‌کدهای مرتبط ارائه شد. در نهایت در فصل ۴ به بررسی کیفیت روش پیشنهادی به صورت کمی و در مقایسه با سایر روش‌ها پرداختیم. از طرفی معماری دقیق مدل‌های دسته‌بند درشت‌دانه و ریزدانه در کنار مجموعه دادگانی که مورد استفاده قرار گرفتند، معرفی شدند. در این فصل، به جمع‌بندی موارد گفته‌شده می‌پردازیم و چالش‌های باقی مانده و حوزه‌هایی که هنوز جای کار دارند را معرفی می‌کنیم.

۵-۱ جمع‌بندی و نقاط قوت روش

در این قسمت مزیت‌های کلیدی رویکرد پیشنهادی در حل چالش‌های مسئله‌ی یادگیری بزرگ مقیاس نمونه محدود بیان شده‌اند.

- با ارائه‌ی یک تنظیمات جدید در مسئله‌ی یادگیری بزرگ مقیاس نمونه محدود، سخت‌ترین ستینگ ممکن را انتخاب کردیم که با ناسازگار بودن داده‌ها و برچسب‌های مجموعه دادگان مبدا و مقصد تشکیل می‌شود. در واقع این تنظیماتی است که به واقعیت نزدیک‌تر است چرا که چالش‌های مجموعه باز بودن و تغییر توزیع زیرجمعیت را دربرمی‌گیرد.

- در سطح درشت دانه، یک روش بدیع برای مقاومت در برابر تغییر توزیع زیرجمعیت به صورت نیمه‌نظارتی پیشنهاد دادیم که به آن تنظیم ترکیبی می‌گوییم.
- از بدنه‌های بنیادی بیانگر بهره بردیم و به همین جهت لایه‌هایی که آموزش می‌دهیم تا حد ممکن کوچک اند. در نتیجه هم بهینگی محاسباتی را سبب می‌شوند و هم در تنظیمات خاص این مسئله که تغییر توزیع و نمونه‌ی محدود داریم، از بیش برازش روی مجموعه‌ی مبدا جلوگیری می‌شود.
- از سلسله‌مراتب ذاتی در داده‌ها استفاده کردیم و در یک چارچوب بیزی، به ترکیب اطلاعات درشت‌دانه برای دسته‌بندی در سطح ریزدانه پرداختیم. به این ترتیب یک سوگیری القایی مبتنی بر وجود سلسله‌مراتب در یادگیری داریم که در بازنمایی داده‌ها بروز پیدا می‌کند و باعث منظم شدن فضای دسته‌ها در شرایط مجموعه باز می شود.
- با توجه به مشاهدات، از یک راهبرد یکی از همه برنده استفاده کردیم که تعداد دسته‌های هدف را محدود میکند و باعث بالا رفتن دقت می‌شود.
- با تعریف ساختار سلسله‌مراتبی، سبب شدیم در سطح درشت‌دانه با وجود نمونه محدود بودن دسته‌ها، داده‌ی کافی برای آموزش به صورت تمایزگذار فراهم شود و تمامی روش های متداول دسته‌بندی در یادگیری ژرف را بتوان در این سطح استفاده کرد.
- در مجموعه‌ی مبدا هم علاوه بر مقصد، دسته‌ها تعداد محدودی نمونه برچسب خورده دارند. برخلاف اکثر روش‌های دیگر که نیاز به یک مجموعه‌ی داده‌ی مبدا با تعداد بالای داده‌ی آموزشی دارند.
- در سطح ریزدانه، از یک رویکرد متایادگیری مولد استفاده کردیم که در عین داشتن مزایای متایادگیری، قابلیت کار در شرایط مجموعه باز و ظهور دسته‌های بدیع را دارد.
- رویکرد دسته‌بندی ما در سطح ریزدانه، یک یادگیری چند وظیفه‌ای [1] هست و در واقع چیزی که مدل بر روی آن مشروط می‌شود، یک شناسه‌ی وظیفه است. بنابراین سرهای قابل آموزش، از دانش مشترک بین ابردسته‌ها یاد می‌گیرند و یک رویکرد ترکیب خبره‌ها [2] داریم.
- با توجه به نتایج ارائه شده، رویکرد پیشنهادی در قالب دسته‌بندی سلسله‌مراتبی می‌تواند در تنظیمات بدون یادگیری [3] هم به کارایی لازم برسد. به این منظور لازم است از بدنه‌های بنیادی مناسبی استفاده شود.

۵-۲ کارهای آتی

همانطور که در جداول فصل ۴ دیدیم، رویکرد پیشنهادی بر روی بدنه‌های کوچک و متوسط منجر به افزایش دقت بسیار زیادی می‌شود. همچنین روی بدنه‌های بنیادی بزرگ مانند مدل CLIP که با حجم بسیار بالای داده آموزش دیده، به افزایش دقت قابل توجهی در تمامی شرایط تعریف شده اعم از با یادگیری و بدون یادگیری می‌رسیم. با این

[1] Multi Task
[2] Mixture of Experts
[3] Free Training

حال در بدنه‌های بنیادی بسیار قوی مانند Dino که از طرف شرکت‌های صنعتی بزرگ با تعداد پارامترهای بسیار بالا و آموزش روی مجموعه داده‌های بزرگ ارائه شده‌اند، افزایش دقت وجود دارد اما به اندازه‌ی موارد بیان شده‌ی دیگر نیست. با این وجود، اگر از دقت حاصل از مدل پیشگو [1] در سطح درشت‌دانه استفاده کنیم، دیدیم که در مدل‌های بنیادی خیلی بزرگ هم می‌توان افزایش دقت خوبی داشت. با توجه به این که دسته‌بند سطح درشت‌دانه‌ی ما، تنها یک لایه‌ی خطی بر روی بدنه‌ی پیش‌آموزش دیده آموزش می‌دهد، به نظر می‌رسد با کار بیشتر روی دسته‌بند این سطح بتوان به دقت حدود دسته‌بند پیشگو رسید. این موضوع با وجود تجمیع نمونه‌های دسته‌های پایه در سطح درشت دانه و تعداد کم دسته‌ها در حدود زیر ۱۰۰ ابردسته، قابل دستیابی است.

مورد دیگری که جای بهبود دارد، در نظر گرفتن اضافه شدن ابردسته‌های جدید در بعضی تنظیمات خیلی خاص است. به این منظور می‌توان روش‌های حکاکی وزن‌ها که در قسمت ۲-۲.۴ اشاره شد را مورد توجه قرار داد. در این صورت همچنان می‌توان از رویکرد یادگیری تمایزگذار و اکثر روش‌های متداول یادگیری ژرف مستقل از معماری و شرایط دیگر بهره‌برد.

[1] Oracle

مراجع

[1] Y.-X. Wang, R. B. Girshick, M. Hebert, and B. Hariharan, "Low-shot learning from imaginary data," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7278–7286, 2018.

[2] S. Yang, L. Liu, and M. Xu, "Free lunch for few-shot learning: Distribution calibration," *International Conference on Learning Representations*, 2021.

[3] S. Ravi and H. Larochelle, "Optimization as a model for few-shot learning," in *International Conference on Learning Representations*, 2016.

[4] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *International Conference on Machine Learning*, 2017.

[5] N. Mishra, M. Rohaninejad, X. Chen, and P. Abbeel, "A simple neural attentive meta-learner," in *International Conference on Learning Representations*, 2017.

[6] J. Snell, K. Swersky, and R. Zemel, "Prototypical networks for few-shot learning," in *Advances in Neural Information Processing Systems*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30. Curran Associates, Inc., 2017, pp. 4077–4087.

[7] H. Yao, Y. Wei, J. Huang, and Z. J. Li, "Hierarchically structured meta-learning," *International Conference on Machine Learning*, pp. 7045–7054, 2019.

[8] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," in *International Conference on Machine Learning*, 2021.

[9] D. Hendrycks and K. Gimpel, "A baseline for detecting misclassified and out-of-distribution examples in neural networks," *International Conference on Learning Representations, ICLR*, vol. abs/1610.02136, 2016.

[10] A. Bendale and T. E. Boult, "Towards open set deep networks," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1563–1572, 2015.

[11] S. Qiao, C. Liu, W. Shen, and A. L. Yuille, "Few-shot image recognition by predicting parameters from activations," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7229–7238, 2017.

[12] Qi, M. A. Brown, and D. G. Lowe, "Low-shot learning with imprinted weights," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5822–5830, 2017.

[13] H. F. Garcia, A. Aguilar, E. Manilow, and B. Pardo, "Leveraging hierarchical structures for few-shot musical instrument recognition," *International Society for Music Information Retrieval Conference*, pp. 220–228, 2021.

[14] Z. Novack, S. Garg, J. McAuley, and Z. C. Lipton, "Chils: Zero-shot image classification with hierarchical label sets," *Proceedings of the 40th International Conference on Machine Learning*, vol. 202, pp. 26 342–26 362, 23–29 Jul 2023.

[15] A. Li, T. Luo, Z. Lu, T. Xiang, and L. Wang, "Large-scale few-shot learning: Knowledge transfer with class hierarchy," *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7205–7213, 2019.

[16] Wikipedia, "Taxonomic rank," *https://en.wikipedia.org/wiki/Taxonomic_rank*, Accessed 28 Jan. 2024.

[17] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *Journal of Big Data*, vol. 6, pp. 1–48, 2019.

[18] B. Hariharan and R. B. Girshick, "Low-shot visual recognition by shrinking and hallucinating features," *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 3037–3046, 2016.

[19] L. Jing and Y. Tian, "Self-supervised visual feature learning with deep neural networks: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, pp. 4037–4058, 2019.

[20] X. Liu, F. Zhang, Z. Hou, Z. Wang, L. Mian, J. Zhang, and J. Tang, "Self-supervised learning: Generative or contrastive," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, pp. 857–876, 2020.

[21] Y. C. A. P. Reddy, P. Viswanath, and B. E. Reddy, "Semi-supervised learning: a brief review," *International journal of engineering and technology*, vol. 7, p. 81, 2018.

[22] D. P. Kingma, S. Mohamed, D. J. Rezende, and M. Welling, "Semi-supervised learning with deep generative models," *Advances in neural information processing systems 27*, 2014.

[23] E. Bair, "Semi-supervised clustering methods," *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 5, no. 5, pp. 349–361, 2013.

[24] Y. Qin, S. Ding, L. Wang, and Y. Wang, "Research progress on semi-supervised clustering," *Cognitive Computation*, vol. 11, pp. 599 – 612, 2019.

[25] T. Chen, S. Kornblith, K. Swersky, M. Norouzi, and G. E. Hinton, "Big self-supervised models are strong semi-supervised learners," *Advances in neural information processing systems 33*, pp. 22 243–22 255, 2020.

[26] T. Chen, S. Kornblith, M. Norouzi, and G. E. Hinton, "A simple framework for contrastive learning of visual representations," *International conference on machine learning*, pp. 1597–1607, 2020.

[27] D. H. Hubel and T. N. Wiesel, "Receptive fields and functional architecture of monkey striate cortex," *The Journal of Physiology*, vol. 195, 1968.

[28] K. Fukushima, "Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position," *Biological Cybernetics*, vol. 36, pp. 193–202, 1980.

[29] I. Kuzovkin, R. Vicente, M. Petton, J.-P. Lachaux, M. Baciu, P. Kahane, S. Rheims, J. R. Vidal, and J. Aru, "Activations of deep convolutional neural networks are aligned with gamma band activity of human visual cortex," *Communications Biology*, vol. 1, 2017.

[30] J. Snell, K. Swersky, and R. S. Zemel, "Prototypical networks for few-shot learning," *Advances in neural information processing systems 30*, 2017.

[31] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *International Conference on Machine Learning*, 2017.

[32] J. Chen, L. Zhuo, Z. Wei, H. Zhang, H. Fu, and Y. Jiang, "Knowledge driven weights estimation for large-scale few-shot image recognition," *Pattern Recognition*, vol. 142, p. 109668, 2023.

[33] J. Guan, M. Zhang, and Z. Lu, "Large-scale cross-domain few-shot learning," in *Asian Conference on Computer Vision*, 2020.

[34] L. A. Gatys, A. S. Ecker, and M. Bethge, "A neural algorithm of artistic style," *ArXiv*, vol. abs/1508.06576, 2015.

[35] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang, "Random erasing data augmentation," *AAAI conference on artificial intelligence*, vol. abs/1708.04896, 2017.

[36] R. Takahashi, T. Matsubara, and K. Uehara, "Data augmentation using random image cropping and patching for deep cnns," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, pp. 2917–2931, 2018.

[37] T. M. Hospedales, A. Antoniou, P. Micaelli, and A. J. Storkey, "Meta-learning in neural networks: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, pp. 5149–5169, 2020.

[38] A. Nichol, J. Achiam, and J. Schulman, "On first-order meta-learning algorithms," *ArXiv*, vol. abs/1803.02999, 2018.

[39] K. Lee, S. Maji, A. Ravichandran, and S. Soatto, "Meta-learning with differentiable convex optimization," *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10 649–10 657, 2019.

[40] M. Zaheer, S. Kottur, S. Ravanbakhsh, B. Póczos, R. Salakhutdinov, and A. Smola, "Deep sets," in *NIPS*, 2017.

[41] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *ArXiv*, vol. abs/1706.03762, 2017.

[42] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, pp. 1735–1780, 1997.

[43] B. Liu, H. Kang, H. Li, G. Hua, and N. Vasconcelos, "Few-shot open-set recognition using meta-learning," *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8795–8804, 2020.

[44] F. Sung, Y. Yang, L. Zhang, T. Xiang, P. H. S. Torr, and T. M. Hospedales, "Learning to compare: Relation network for few-shot learning," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1199–1208, 2017.

[45] O. Vinyals, C. Blundell, T. P. Lillicrap, K. Kavukcuoglu, and D. Wierstra, "Matching networks for one shot learning," *NIPS*, 2016.

[46] Y. Lee and S. Choi, "Gradient-based meta-learning with learned layerwise metric and subspace," *International Conference on Machine Learning*, 2018.

[47] T. Kim, J. Yoon, O. A. Dia, S. Kim, Y. Bengio, and S. Ahn, "Bayesian model-agnostic meta-learning," *Neural Information Processing Systems*, 2018.

[48] C. Finn, K. Xu, and S. Levine, "Probabilistic model-agnostic meta-learning," *Neural Information Processing Systems*, 2018.

[49] S. X. Hu, D. Li, J. Stuhmer, M. Kim, and T. M. Hospedales, "Pushing the limits of simple pipelines for few-shot learning: External data and fine-tuning make a difference," *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9058–9067, 2022.

[50] Y. Tian, Y. Wang, D. Krishnan, J. B. Tenenbaum, and P. Isola, "Rethinking few-shot image classification: a good embedding is all you need?" in *European Conference on Computer Vision*, 2020.

[51] H. Xu, J. Wang, H. Li, D. Ouyang, and J. Shao, "Unsupervised meta-learning for few-shot learning," *Pattern Recognit.*, vol. 116, p. 107951, 2021.

[52] Y. Lu, L. Wen, J. Liu, Y. Liu, and X. Tian, "Self-supervision can be a good few-shot learner," in *European Conference on Computer Vision*, 2022.

[53] M. Caron, H. Touvron, I. Misra, H. J'egou, J. Mairal, P. Bojanowski, and A. Joulin, "Emerging properties in self-supervised vision transformers," *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9630–9640, 2021.

[54] R. Zhang, X. Hu, B. Li, S. Huang, H. Deng, H. Li, Y. J. Qiao, and P. Gao, "Prompt, generate, then cache: Cascade of foundation models makes strong few-shot learners," *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15 211–15 222, 2023.

[55] R. Zhang, Z. Wei, R. Fang, P. Gao, K. Li, J. Dai, Y. J. Qiao, and H. Li, "Tip-adapter: Training-free adaption of clip for few-shot classification," *ArXiv*, vol. abs/2207.09519, 2022.

[56] S. H. Huang, N. Papernot, I. J. Goodfellow, Y. Duan, and P. Abbeel, "Adversarial attacks on neural network policies," *5th International Conference on Learning Representations, ICLR*, vol. abs/1702.02284, 2017.

[57] J. Lu, T. Issaranon, and D. A. Forsyth, "Safetynet: Detecting and rejecting adversarial examples robustly," *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 446–454, 2017.

[58] V. S. F. Garnot and L. Landrieu, "Leveraging class hierarchies with metric-guided prototype learning," *British Machine Vision Conference*, 2020.

[59] G. V. Horn, O. M. Aodha, Y. Song, Y. Cui, C. Sun, A. Shepard, H. Adam, P. Perona, and S. J. Belongie, "The inaturalist species classification and detection dataset," *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8769–8778, 2017.

[60] M. Oquab, T. Darcet, T. Moutakanni, H. Q. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, M. Assran, N. Ballas, W. Galuba, R. Howes, P.-Y. B. Huang, S.-W. Li, I. Misra, M. G. Rabbat, V. Sharma, G. Synnaeve, H. Xu, H. Jégou, J. Mairal, P. Labatut, A. Joulin, and P. Bojanowski, "Dinov2: Learning robust visual features without supervision," *ArXiv*, vol. abs/2304.07193, 2023.

[61] Z. Liu, H. Mao, C. Wu, C. Feichtenhofer, T. Darrell, and S. Xie, "A convnet for the 2020s," *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11 966–11 976, 2022.

[62] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. S. Bernstein, A. C. Berg, and L. Fei-Fei, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, pp. 211 – 252, 2014.

[63] A. G. Howard, M. Sandler, G. Chu, L.-C. Chen, B. Chen, M. Tan, W. Wang, Y. Zhu, R. Pang, V. Vasudevan, Q. V. Le, and H. Adam, "Searching for mobilenetv3," *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1314–1324, 2019.

[64] T. Ridnik, E. Ben-Baruch, A. Noy, and L. Zelnik-Manor, "Imagenet-21k pretraining for the masses," *ArXiv*, vol. abs/2104.10972, 2021.

واژه‌نامه

الف

آماره کافی ..... Sufficient Statistic
ابردسته ..... Superclass
الگو ..... Paradigm

ب

بازنمایی ..... Representation
بسته ..... Batch
بدنه ..... Backbone
بدون آموزش ..... Free Training
بدون نمونه ..... Zero Shot
بدیع ..... Novel
بزرگ مقیاس ..... Large Scale
بیانگر ..... Expressive
بیزی ..... Bayesian
بیش‌برازش ..... Overfitting
بی‌نظمی متقابل ..... Cross Entropy

پ

پارامترهای خاص وظیفه ..... Task-Specific Parameters
پایه ..... Baseline
پرسش ..... Query
نماینده ..... Prototype
پشتیبان ..... Support
پیش‌آموزش دیده ..... Pretrained
پیشین ..... Prior
پسین ..... Posterior

ت

تابع خطا ..... Loss Function
تباینی ..... Contrastive
تعبیه ..... Embedding
تعمیم‌پذیری ..... Generalization
تغییر توزیع ..... Distribution Shift
تغییر توزیع زیرجمعیت ..... Sub Population Shift
تغییر زیرجمعیت ..... Sub Population Shift
تمایزگذار ..... Discriminative
تنظیم بهینه ..... Fine Tune
تنظیم ترکیبی ..... Mixed Tune
تنظیمات ..... Setting

ج

جعبه‌سیاه ..... Black Box

چ

چند وظیفه‌ای ..... Multi Task

ح

حکاکی ..... Imprinting
حلقه‌ی بیرونی ..... Outer Loop
حلقه‌ی درونی ..... Inner Loop

خ

خود نظارتی ..... Self-Supervised
خیال‌بافی ..... Hallucination

د

داده کارآمدی ..... Data Efficiency
درشت‌دانه ..... Coarse Grained
درست‌نمایی ..... Likelihood
دسته‌ی پایه ..... Base Class

ر

راه ..... Way
ریزدانه ..... Fine Grained
ریزدانگی ..... Granularity

ز

زنجیره‌ی علی ..... Causal Chain

س

سلسله‌مراتب ..... Hierarchy
سوگیری ..... Bias
سوگیری القایی ..... Inductive Bias
سیستم‌های خبره ..... Expert Systems

ش

شبکه‌ی عصبی بازگشتی ..... Recurrent Neural Network
شناسه‌ی وظیفه ..... Task Identifier

ص

صحت ..... Accuracy

ف

فعال‌سازی ..... Activation

ق

قطعی ..... Deterministic

گ

گاوسی ..... Gaussian

م

متاآموزش ..... Meta Train
متاآزمون ..... Meta Test
متاپارامتر ..... Meta Parameters
متا یادگیر ..... Meta Learner
متا یادگیری ..... Meta Learning
مجموعه باز ..... Open Set
مجموعه دادگان ..... Dataset
مدل بنیادی ..... Foundation Model
مشروط ..... Conditioned
مصالحه ..... Trade Off
مطالعه‌ی فرسایشی ..... Ablation Study
مقاوم ..... Robust
مولد ..... Generative

ن

ناسازگار ..... Disjoint

و

وظیفه ..... Task

ه

همه‌کاره ..... Versatile
همبستگی ..... Correlation
هوش مصنوعی عمومی ... Artificial General Intelligence

ی

یادگیر پایه ..... Base Learner
یادگیری نمونه محدود ..... Few Shot Learning
یادگیری نظارت‌شده ..... Supervised Learning
یادگیری مبتنی بر متر ..... Metric Based Learning
یکی از همه برنده ..... Winner Takes All

Abstract

Few-shot learning methods have achieved notable performance in recent years. However, few-shot learning in large-scale settings with hundreds of classes is still challenging. In this dissertation, we tackle the problems of large-scale few-shot learning by taking advantage of pre-trained foundation models. We recast the original problem in two levels with different granularity.

At the coarse-grained level, we introduce a novel object recognition approach with robustness to sub-population shifts. At the fine-grained level, generative experts are designed for few-shot learning, specialized for different superclasses. A Bayesian schema is considered to combine coarse-grained information with fine-grained predictions in a winnertakes-all fashion.

Extensive experiments on large-scale datasets and different architectures show that the proposed method is both effective and efficient besides its simplicity and natural problem remodeling.

Keywords: *Deep Learning, Large-Scale Classification, Few-Shot Learning, Sub-Population Shift, Hierarchical Classification, Generalization, Robustness, Bayesian Learning*

Sharif University of Technology

Department of Computer Engineering

M.Sc. Thesis

Artificial Intelligence

Topic

Many-Class Few-Shot Classification

By

Mohamadreza Fereydooni

Supervisor

Dr. Mahdieh Soleymani Baghshah

Winter 2024