



دانشکده مهندسی کامپیوتر

استفاده از یادگیری عمیق برای دسته‌بندی تصاویر با مجموعه داده

محدود

پروژه برای دریافت درجه کارشناسی در رشته مهندسی کامپیوتر گرایش

هوش مصنوعی

فاطمه عسکری

استاد راهنما:

دکتر محمدرضا محمدی

بهمن ۱۴۰۲

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

تاییدیه ی هیات داوران جلسه دفاع از پایان نامه / رساله

نام دانشکده: دانشکده مهندسی کامپیوتر

نام دانشجو: فاطمه عسکری جیره‌نده

عنوان پایان نامه یا رساله : استفاده از یادگیری عمیق برای دسته‌بندی تصاویر با

مجموعه داده محدود

تاریخ دفاع : بهمن ۱۴۰۲

رشته: مهندسی کامپیوتر

گرایش: هوش مصنوعی

ردیف	سمت	نام و نام خانوادگی	مرتبه دانشگاهی	دانشگاه یا موسسه	امضا
۱	استاد راهنما	دکتر محمدرضا محمدی	استادیار	دانشگاه علم و صنعت تهران	
۲	استاد داور داخلی	دکتر ناصر مزینی	دانشیار	دانشگاه علم و صنعت تهران	

تأییدیه‌ی صحت و اصالت نتایج

بسمه تعالی

اینجانب فاطمه عسکری به شماره دانشجویی ۹۸۴۷۱۴۱۴ دانشجوی رشته مهندسی کامپیوتر مقطع تحصیلی کارشناسی تأیید می‌نمایم که کلیه نتایج مندرج در این پایان‌نامه حاصل کار پژوهشی اینجانب تحت نظارت و راهنمایی عضو هیأت علمی دانشگاه علم و صنعت ایران بدون هر گونه دخل و تصرف انجام گرفته و به موارد نسخه‌برداری شده از آثار دیگران، مطابق مقررات و ضوابط، ارجاع داده شده و مشخصات کامل منابع را در فهرست منابع ذکر کرده‌ام. این پایان‌نامه قبلاً برای احراز هیچ مدرکی ارائه نگردیده است.

در صورت اثبات خالف مندرجات فوق، به تشخیص دانشگاه مطابق با ضوابط و مقررات حاکم (قانون حمایت از حقوق مولفان و منصفان و قانون ترجمه، تکثیر و نشریات و آثار صوتی، ضوابط و مقررات آموزشی و پژوهشی، انضباطی و غیره) با اینجانب رفتار خواهد شد و حق هرگونه اعتراض در خصوص احقاق حقوق مکسب و تشخیص و تعیین تخلف و مجازات را از خویش سلب می‌نمایم. درضمن، مسئولیت هرگونه پاسخگویی به اشخاص اعم از حقیقی و حقوقی و مراجع ذی‌صلاح (اعم از اداری و قضایی) به عهده اینجانب خواهد بود و دانشگاه هیچ گونه مسئولیتی در این خصوص نخواهد داشت.

کلیه نتایج و حقوق حاصل از این پایان‌نامه متعلق به دانشگاه علم و صنعت ایران است. هرگونه استفاده از نتایج علمی و عملی و واگذاری اطلاعات به دیگران یا چاپ و تکثیر، نسخه‌برداری ترجمه و اقتباس از این پایان‌نامه بدون موافقت کتبی دانشگاه علم و صنعت ایران ممنوع است. نقل مطالب با ذکر منبع بلامانع است.

نام و نام خانوادگی: فاطمه عسکری

امضا و تاریخ:

مجوز بهره برداری از پایان نامه

بهره برداری از این پایان نامه در چارچوب مقررات کتابخانه و با توجه به محدودیتی که توسط استاد راهنما به شرح زیر تعیین می شود، بلامانع است:

- بهره برداری از این پایان نامه برای همگان بلامانع است.
- بهره برداری از این پایان نامه با اخذ مجوز از استاد راهنما، بلامانع است.
- بهره برداری از این پایان نامه تا تاریخ ممنوع است.

استاد راهنما: دکتر محمدرضا محمدی

تاریخ:

امضا:

تقدیر و تشکر

از خانواده‌ام که در تمامی مراحل زندگی حمایت کردند، از اساتید گرانقدر، به ویژه آقای دکتر محمدرضا محمدی که در طول تحصیل و تحقیق، من را از تجارب با ارزش خودشون بهره‌مند ساختند، صمیمانه سپاسگزارم. همچنین برخورد لازم می‌دانم تا از حمایت‌ها و راهنمایی‌های بی‌دریغ آقا امیررضا فاتح در طول تحقیق نهایت تشکر و قدردانی بکنم.

چکیده

طبقه‌بندی تصاویر یکی از مسائل مهم و کلیدی در زمینه هوش مصنوعی می‌باشد. در سال‌های اخیر بیشتر روش‌های این حوزه از مدل‌های یادگیری عمیق برای حل مسئله طبقه‌بندی استفاده کرده‌اند. رسیدن به نتایج خوب در زمینه طبقه‌بندی با کمک یادگیری عمیق، رابطه مستقیمی با تعداد تصاویر آموزشی دارد. ولی دستیابی به یک مجموعه آموزشی بزرگ، در بسیاری از حوزه‌ها امکان‌پذیر نیست. هم‌چنین جمع‌آوری یک مجموعه داده مناسب، بسته به نوع مسئله می‌تواند هزینه و زمان زیادی صرف نماید. در این پایان‌نامه ما مدلی برای طبقه‌بندی تصاویر با مجموعه داده محدود ارائه داده‌ایم که پایه آن از مدل Prototypical است و تلاش کردیم آن را بهبود بدهیم. کارهایی که برای بالا بردن دقت مدل انجام داده‌ایم شامل تغییر backbone، استفاده از فضای تعبیه متعدد، دادن پارامتر وزن به هر یک بردارهای خروجی و استفاده از مکانیزم توجه به خود است. درنهایت، با مقایسه دقت در هر مرحله بهبود را نسبت به مرحله قبل نشان داده‌ایم و دقت مدل بر روی داده آموزشی MiniImagenet با سایر مدل‌های موجود در مقالات در دو تسک ۵ کلاس ۵ شات و ۵ کلاس تک شات مقایسه کرده‌ایم. در تسک ۵ کلاس ۵ شات به دقت ۸۴.۴۲ درصد و در تسک ۵ کلاس تک شات به دقت ۶۴.۴۶ درصد رسیده ایم. هم‌چنین برای اینکه نشان دهیم مدل تعمیم‌پذیری خوبی دارد مدل آموزش‌دیده با داده آموزشی MiniImagenet را بر روی داده آموزشی دیگری بدون آموزش بر روی آن تست کردیم و به دقت خوبی رسیدیم.

واژه‌های کلیدی: یادگیری عمیق، طبقه‌بندی تصاویر، یادگیری با مجموعه داده محدود، توجه به خود، فضای تعبیه متعدد

فهرست مطالب

فصل ۱	۱
مقدمه	۱
۱-۱ تعریف مسئله	۱
۲-۱ اهمیت موضوع	۲
۳-۱ اهداف پژوهش	۴
۴-۱ ساختار پایان نامه	۴
فصل ۲	۵
کارهای پیشین و مرتبط	۵
۱-۲ مقدمه	۵
۲-۲ حوزه غیر فرا یادگیری	۶
۱-۲-۲ یادگیری انتقالی	۶
۱-۲-۲-۱ طبقه بندی بر اساس معیار فاصله در یک شبکه پیش آموخته	۶
۲-۲-۲ آموزش یک طبقه کننده جدید با استفاده از یک شبکه پیش آموخته	۶
۳-۲-۲ استنتاج انتقالی با استفاده از تعبیه ها از یک شبکه پیش آموخته	۷
۳-۲ حوزه فرا یادگیری	۷
۱-۳-۲ یادگیری بدون نمونه	۸
۲-۳-۲ یادگیری با مجموعه دادگان محدود	۹
۱-۲-۳-۲ روش های مبتنی بر مدل	۹
۲-۲-۳-۲ روش های مبتنی بر بهینه سازی	۹
۳-۲-۳-۲ روش های مبتنی بر متریک	۱۰
۱-۳-۲-۳-۲ شبکه عصبی Siamese	۱۱
۲-۳-۲-۳-۲ شبکه تطبیق	۱۲
۳-۳-۲-۳-۲ شبکه prototypical	۱۳
۴-۳-۲-۳-۲ شبکه relation	۱۴
۵-۳-۲-۳-۲ یادگیری متمرکز بر وظیفه برای دسته بندی تصاویر با استفاده از استخراج ویژگی پویا و معیار فاصله قابل تطبیق (TADAM)	۱۶
۴-۲-۳-۲ روش های جدید در یادگیری با مجموعه داده محدود	۱۷
۴-۲ توابع ضرر در مجموعه دادگان محدود	۲۱

۲۱	۱-۴-۲ تابع ضرر Contrastive
۲۲	۲-۴-۲ تابع ضرر شبکه تطبیق
۲۲	۳-۴-۲ تابع ضرر Triplet
۲۳	۴-۴-۲ تابع ضرر Center
۲۴	۵-۲ معیارهای ارزیابی
۲۵	۱-۵-۲ معیار ROC-AUC
۲۶	فصل ۳
۲۶	روش‌های پیشنهادی
۲۶	۱-۳ مقدمه
۲۶	۲-۳ چالش‌های مربوط به دست بندی تصاویر با مجموعه دادگان محدود
۲۶	۱-۲-۳ تطبیق بیش از حد
۲۷	۲-۲-۳ تأثیر هاپر پارامترها بر دقت مدل
۲۷	۳-۲-۳ کافی نبودن یک بردار ویژگی
۲۷	۴-۲-۳ استخراج بردار ویژگی با کیفیت
۲۷	۳-۳ پیاده‌سازی مدل پیشنهادی
۲۸	۱-۳-۳ پیاده‌سازی اولیه Prototypical
۲۹	۲-۳-۳ استفاده از فضای تعبیه متعدد در مدل
۳۰	۳-۳-۳ دادن پارامتر وزن قابل یادگیری به هر خروجی
۳۱	۴-۳-۳ اضافه کردن مکانیزم توجه به خود در هر بردار خروجی
۳۵	فصل ۴
۳۵	نتایج آزمایش‌ها
۳۵	۱-۴ مقدمه
۳۵	۲-۴ نتایج مدل پیشنهادی در هر مرحله
۳۵	۱-۲-۴ نتایج مدل اولیه Prototypical
۳۷	۲-۲-۴ نتایج استفاده از فضای تعبیه متعدد
۳۸	۳-۲-۴ نتایج دادن پارامتر وزن قابل یادگیری به هر خروجی
۳۹	۴-۲-۴ نتایج اضافه کردن مکانیزم توجه به خود
۴۰	۳-۴ نتایج نهایی و مقایسه مدل پیشنهادی با سایر مدل‌ها
۴۲	۴-۴ تست مدل بر روی تصاویر هوایی

فصل ۵.....	۴۵
جمع بندی.....	۴۵
۱-۵ نتیجه گیری.....	۴۵
۲-۵ کارهای آینده.....	۴۶
۱-۲-۵ استفاده از یادگیری دو مرحله ای.....	۴۶
۲-۲-۵ خوشه بندی کردن داده ها.....	۴۶
۳-۲-۵ ترکیب تابع ضرر Cross Entropy با سایر توابع ضرر.....	۴۶
منابع:.....	۴۷

فهرست تصاویر

- شکل ۱-۱ : مساله طبقه‌بندی با داده جدید در کلاس آزمون..... ۲
- شکل ۲-۱: نمونه‌ای از مسئله طبقه‌بندی تصاویر..... ۳
- شکل ۳-۱: تأثیر اندازه دیتاست بر روی دقت مدل..... ۳
- شکل ۱-۲: نمونه‌ای از کاربرد فرا یادگیری در زمینه طبقه‌بندی تصاویر..... ۸
- شکل ۲-۲: مقایسه یادگیری شبکه بدون نمونه با تفکر انسان..... ۸
- شکل ۳-۲: نمودار الگوریتم فرا یادگیری آگنوستیک..... ۱۰
- شکل ۴-۲: نمونه‌ای از مسئله فرا یادگیری مبتنی بر متریک برای یک تسک ۴ کلاس ۱ تصویر..... ۱۱
- شکل ۵-۲: شبکه عصبی siamese..... ۱۲
- شکل ۶-۲: شبکه تطبیق..... ۱۳
- شکل ۷-۲: شبکه prototypical..... ۱۴
- شکل ۸-۲: معماری شبکه relation برای یادگیری با مجموعه دادگان محدود که از عناصری از جمله شبکه کانولوشنی تشکیل شده است..... ۱۵
- شکل ۹-۲: شبکه relation برای مسئله یادگیری ۵ کلاسه تک شات به همراه یک تصویر کوثری..... ۱۶
- شکل ۱۰-۲: معماری چند شات پیشنهادی بلوک‌های دارای پارامترهای مشترک دارای مرز چین‌دار هستند..... ۱۷
- شکل ۱۱-۲: نمای کلی شبکه SetFeat..... ۱۸
- شکل ۱۲-۲: شبکه setfeat وقتی از فاصله sum-min استفاده می‌شود..... ۱۹
- شکل ۱۳-۲: فرایند طبقه‌بندی دو مرحله‌ای..... ۲۰
- شکل ۱۴-۲: ساختار شبکه آموزش دو مرحله‌ای CFMA..... ۲۰
- شکل ۱۵-۲: شبکه تعبیه چند خروجی..... ۲۱
- شکل ۱۶-۲: آموزش شبکه Siamese با تابع ضرر Triplet..... ۲۲
- شکل ۱۷-۲: هدف تابع ضرر center نزدیک کردن ویژگی‌های هر کلاس به همدیگر است..... ۲۳
- شکل ۱۸-۲: تأثیر پارامتر λ بر روی طبق بندی کردن تصاویر..... ۲۴
- شکل ۱۹-۲: فضای ROC برای طبقه‌بندی بهتر و بدتر..... ۲۵
- شکل ۱-۳: نمونه‌ای از تصاویر مجموعه پشتیبان..... ۲۸
- شکل ۲-۳: نمونه‌ای از تصاویر مجموعه کوثری..... ۲۸
- شکل ۳-۳: ساختار اولیه شبکه Prototypical بر روی دیتاست Minilmagenet..... ۲۹
- شکل ۴-۳: ساختار فضا تعبیه متعدد در مدل Prototypical..... ۳۰
- شکل ۵-۳: اضافه کردن پارامتر وزن دهی قابل یادگیری در مدل تعبیه متعدد Prototypical..... ۳۱
- شکل ۶-۳: نگاشت دهنده..... ۳۳
- شکل ۷-۳: شکل مدل نهایی پیشنهادی..... ۳۴
- شکل ۱-۴: نمودار ضرر بر حسب تعداد instance ها برای تسک ۵ کلاسه ۵ شات در مدل اولیه Prototypical..... ۳۶
- شکل ۲-۴: نمودار ضرر بر حسب تعداد instance ها برای تسک ۵ کلاسه تک شات در مدل اولیه Prototypical..... ۳۶
- شکل ۳-۴: نمودار ضرر بر حسب تعداد instance ها برای تسک ۵ کلاسه ۵ شات در استفاده از فضای تعبیه متعدد..... ۳۷
- شکل ۴-۴: نمودار ضرر بر حسب تعداد instance ها برای تسک ۵ کلاسه تک شات در استفاده از فضای تعبیه متعدد..... ۳۷

- شکل ۴ - ۵: نمونه‌ای پیش‌بینی درست مدل در تسک ۵ کلاس ۵ شات ۴۱
- شکل ۴ - ۶: نمونه‌ای پیش‌بینی نادرست مدل در تسک ۵ کلاس ۵ شات ۴۱
- شکل ۴ - ۷: نمونه‌ای پیش‌بینی درست مدل در تسک ۵ کلاس ۵ شات ۴۲
- شکل ۴ - ۸: نمونه‌ای پیش‌بینی نادرست مدل در تسک ۵ کلاس ۵ شات ۴۲
- شکل ۴ - ۹: تصاویر مجموعه پشتیبان در تسک ۵ کلاس ۵ شات ۴۳
- شکل ۴ - ۱۰: تصاویر مجموعه کوثری در تسک ۵ کلاس ۵ شات ۴۴

فهرست جداول

جدول ۳-۱: وزن‌های مناسب برای هر از دو مسئله ۱ شات و ۵ شات.....	۳۱
جدول ۴-۲: مقایسه دقت نرخ یادگیری‌های مختلف بر روی مدل baseline prototypical	۳۶
جدول ۴-۲: مقایسه دقت نرخ یادگیری‌های مختلف در استفاده از فضای تعبیه متعدد.....	۳۸
جدول ۴-۳: بهترین دقت در دادن پارامتر وزن قابل یادگیری به هر خروجی.....	۳۸
جدول ۴-۴: مقایسه دقت در دو حالت در تسک ۵ کلاسه ۵ شات.....	۳۹
جدول ۴-۵: مقایسه دقت در دو حالت در تسک ۵ کلاسه تک شات.....	۳۹
جدول ۴-۶: دقت مدل پس از اضافه کردن توجه به خود.....	۳۹
جدول ۴-۷: مقایسه مدل پیشنهادی با سایر مدل‌ها.....	۴۰
جدول ۴-۸: دقت مدل مرحله به مرحله.....	۴۰
جدول ۴-۹: دقت مدل پیشنهادی آموزش داده شده بر روی Minilmagenet روی تصاویر هوایی.....	۴۴

فصل ۱

مقدمه

۱-۱ تعریف مسئله

امروزه هوش مصنوعی و یادگیری عمیق توسعه چشمگیری پیدا کرده و تأثیر آن در زندگی انسان بیشتر شده است. یکی از حوزه‌هایی که از مدل یادگیری عمیق بهره می‌برد طبقه‌بندی^۱ تصاویر است. به‌طور کلی طبقه‌بندی در حوزه بینایی کامپیوتر فرایندی است که در آن داده‌ها به کلاس‌های مختلف تقسیم می‌شوند. هدف اصلی طبقه‌بندی، تشخیص برچسب یا کلاس متعلق به یک داده است. یکی از چالش‌های اساسی در این زمینه تهیه داده و برچسب‌گذاری آن هست که ممکن است فرایندی زمان‌بر و پرهزینه باشد. برای مواجهه با این مشکل رویکرد یادگیری با مجموعه داده محدود^۲ به‌عنوان یک‌راه حل پیشنهاد شده است. در یادگیری با مجموعه داده محدود، مدل با تعداد کمی نمونه آموزشی برای هر کلاس آموزش داده می‌شود. سپس، با استفاده از اطلاعات کلی که از این نمونه‌ها استخراج می‌شود، مدل با دقت قابل قبولی قادر به تشخیص و طبقه‌بندی داده‌های جدید است. از مزایای این مدل‌ها این است که نیاز به جمع‌آوری داده‌های آموزشی بسیار بزرگ و برچسب‌گذاری آن‌ها به‌صورت کامل کاهش می‌یابد که طبعاً منجر به صرفه جویی در زمان و هزینه است.

نحوه آموزش با مجموعه داده محدود نسبت به روش‌هایی که در طبقه‌بندی تصاویر در یادگیری عمیق استفاده می‌شود متفاوت است. مثلاً در یک مساله ۵ کلاسه به ازای هر کلاس ۱ نمونه داریم (5 way 1 shot) یا ۵ نمونه (5 way 5 shot) داریم. اگر با این تعداد داده آموزشی مدل را آموزش دهیم به احتمال زیاد دچار تطبیق بیش از حد^۳ می‌شویم و مدل نیز قابلیت تعمیم‌پذیری^۴ نیز نخواهد داشت. این تعداد داده محدود تأثیر یادگیری انتقالی^۵ را بسیار کم می‌کند. هم‌چنین داده افزایشی نیز کمک خیلی زیادی نخواهد کرد

¹ Classification

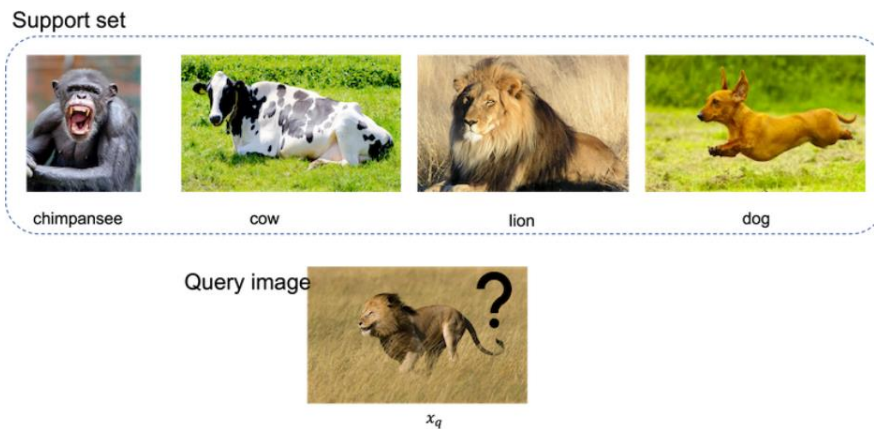
² Few-shot learning

³ overfitting

⁴ generalization

⁵ Transfer Learning

چون داده افزایی اطلاعات جدیدی به داده‌ها اضافه نمی‌کند. لذا مسائل یادگیری با مجموعه دادگان محدود را با درک این مشکلات به صورت دیگری تعریف می‌کنند.



شکل ۱-۱ : مساله طبقه‌بندی با داده جدید در کلاس آزمون

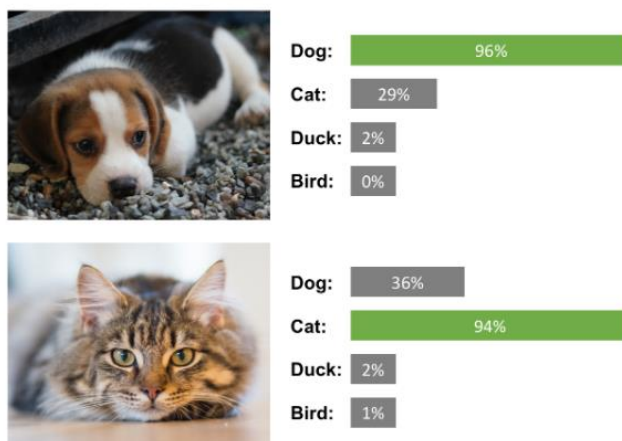
در مسائل با مجموعه دادگان محدود نمی‌توانیم همچون مسائلی که داده آموزشی خوبی داریم برخورد کنیم به همین علت تسک طبقه‌بندی بر اساس دادگان آموزشی را نداریم، بلکه با کمک دادگان آموزشی می‌خواهیم تفاوت‌ها و شباهت‌های بین کلاس‌های مختلف را به دست بیاوریم. در واقع مدل پس از آموزش قادر خواهد بود تشخیص دهد دو تصویر از یک کلاس هستند یا از دو کلاس متفاوت هستند. دو مجموعه داده به نام‌های پشتیبان^۱ و کوئری به مدل داده می‌شود حالا مدلی که آموزش دیده و تفاوت‌ها و شباهت‌ها را یاد گرفته است می‌تواند تشخیص دهد که تصویر کوئری متعلق به کدام یک از تصاویر پشتیبان است.

در این پژوهش، قصد داریم در ابتدا مدل‌های موجود در حوزه دسته‌بندی تصاویر با مجموعه داده محدود را بررسی کنیم و همچنین در صورت توان در عملکرد آن‌ها بهبودی ایجاد کنیم.

۲-۱ اهمیت موضوع

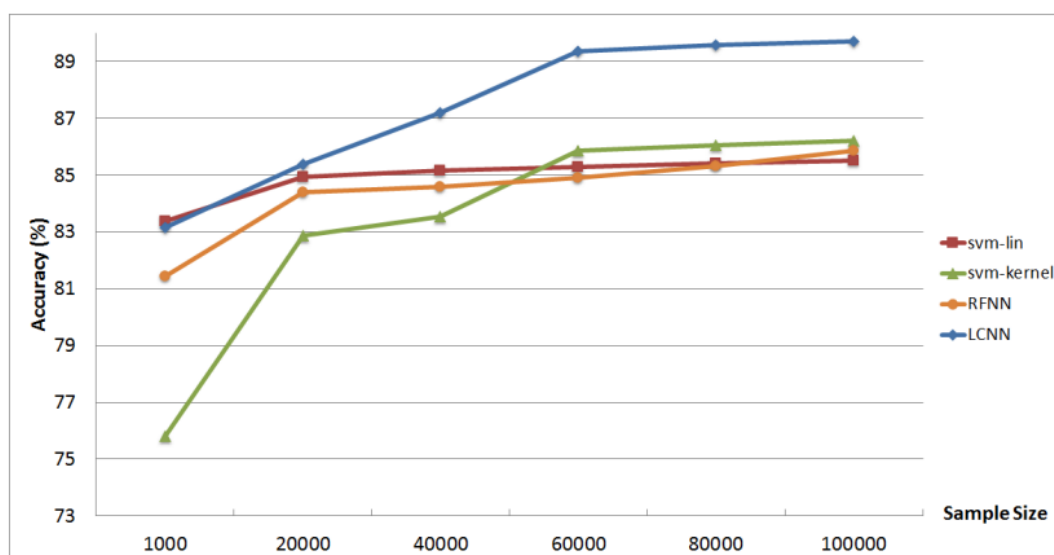
روش‌های مختلفی برای طبقه‌بندی تصاویر پیشنهاد شده است. در سال‌های اخیر شبکه‌های یادگیری عمیق در این حوزه ممتاز بوده‌اند و عملکرد بالایی داشتند. شبکه‌هایی همچون VGGNet, AlexNet, ResNet و GoogleNet با ارائه ساختارهای جدید تحولات اساسی در حوزه طبقه‌بندی تصاویر ایجاد کردند.

^۱ Support Set



شکل ۱-۲: نمونه‌ای از مسئله طبقه‌بندی تصاویر

همان‌طور که بیان شد برای حل مسئله طبقه‌بندی تصاویر به سمت یادگیری عمیق رفته‌اند. حل این مسئله با استفاده از یادگیری عمیق نیاز به آموزش دارد و برای اینکه پس از آموزش مدل بتواند به درستی طبقه‌بندی کند، به مقدار قابل توجهی تصویر با برچسب نیاز داریم.



شکل ۱-۳: تأثیر اندازه دیتاست بر روی دقت مدل

به عنوان مثال همان‌طور که در شکل ۱-۳ مشاهده می‌شود اکثر مدل‌ها زمانی قابل قبول هستند که با حجم خوبی از تصاویر آموزش داده شوند. اما در بسیاری از حوزه‌ها ما، با تعداد کافی داده برچسب خورده برخوردار نیستیم. از جمله حوزه‌هایی که با کمبود داده برچسب خورده مواجه هستیم، حوزه پزشکی است.

با توجه به هزینه‌بر بودن جمع‌آوری، برچسب‌زنی و نبود دادگان کافی، نیاز به مدلی که بتواند با استفاده از همین مجموعه دادگان کم به خوبی آموزش ببیند و بتواند قابلیت تعمیم بالایی داشته باشد. در کنار روش‌هایی مانند یادگیری انتقالی و یادگیری بدون نمونه، یادگیری با مجموعه دادگان محدود روشی است

که با درک مناسب ، توانست، با کمبود دادگان مواجهه خوبی داشته باشد و به دقت‌های قابل قبول‌تری نسبت به سایر روش‌ها برسد.

۳-۱ اهداف پژوهش

در این پایان‌نامه ما به بررسی استفاده از تکنیک‌های یادگیری عمیق و بینایی کامپیوتر در طبقه‌بندی تصاویر با استفاده از مجموعه دادگان محدود خواهیم پرداخت . همچنین مزایا و معایب مدل‌های موجود را مورد بررسی قرار خواهیم داد. در ادامه با انتخاب یکی از مدل‌های موجود با اعمال یکسری از تغییرات سعی در بهبود عملکرد آن با حفظ قدرت تعمیم‌پذیری آن می‌پردازیم و در نهایت تحلیل‌های به دست آمده از مدل نهایی ارائه می‌شود.

۴-۱ ساختار پایان‌نامه

این پایان‌نامه در پنج فصل به شرح زیر ارائه می‌شود:

- فصل دوم به بررسی مفاهیم اولیه و کارهای پیشین مرتبط با این پایان‌نامه می‌پردازد.
- فصل سوم به توضیح روش‌های پیشنهادی برای حل مساله می‌پردازد.
- فصل چهارم به بیان نتایج به دست آمده از اجرای روش‌های پیشنهادی می‌پردازد.
- فصل پنجم، جمع‌بندی کارهای انجام شده در این پژوهش و ارائه پیشنهادهایی برای انجام کارهای آتی خواهد پرداخت.

فصل ۲

کارهای پیشین و مرتبط

۲-۱ مقدمه

در این بخش، پیشینه تحقیقات و مطالعات قبلی درباره یادگیری با مجموعه داده‌های محدود را مورد بررسی قرار خواهیم داد. به طور کلی، دو رویکرد اصلی برای یادگیری با مجموعه داده‌های محدود وجود دارد: رویکرد فرا یادگیری^۱ و رویکرد غیر فرا یادگیری [۱] در ادامه به بررسی هر یک از این رویکردها خواهیم پرداخت.

فرا یادگیری، فرآیندی است که در آن از اطلاعات و دانشی که به دست آورده‌ایم در یادگیری‌های قبلی استفاده می‌کنیم تا عملکرد و کارایی یادگیری در تسک‌های جدید را بهبود بخشیم. به عبارت دیگر، فرا یادگیری به ما کمک می‌کند تا با یادگیری و تجربه‌های گذشته، روش‌های بهتری برای یادگیری مسائل جدید پیدا کنیم. یکی از مزایای فرا یادگیری این است که ما بدون نیاز به تلاش و هزینه‌های زیاد، قادر خواهیم بود اطلاعات جدید را به راحتی و با سرعت بیشتری فرا بگیریم.

غیرفرا یادگیری، رویکردی است که در آن مدل با استفاده از مجموعه داده‌های آموزشی آموزش می‌بیند و قوانین و الگوهای لازم برای حل مسئله را از روی این داده‌ها یاد می‌گیرد. در این رویکرد، هر مسئله به صورت جداگانه و مجدد حل می‌شود و هیچ دانش یادگیری قبلی به مسئله جدید منتقل نمی‌شود. به عبارت دیگر، یادگیرنده برای حل هر مسئله باید از ابتدا شروع کرده و بدون داشتن اطلاعات پیشین، آن را یاد بگیرد. این رویکرد در مواردی از مفید است که مجموعه داده‌ها به اندازه کافی بزرگ است و امکان بهره‌گیری از دانش یادگیری قبلی وجود ندارد یا ممکن است داده‌های جدید و تغییر کننده باشند.

^۱ Meta learning

۲-۲ حوزه غیر فرا یادگیری

در این بخش به توضیح روش‌های غیر فرا یادگیری که می‌تواند به یادگیری مدل در مجموعه دادگان محدود کمک کند، به طور بسیاری از این روش‌ها در حوزه یادگیری انتقالی قرار می‌گیرند. در حد مختصر به این روش‌ها می‌پردازیم.

۱-۲-۲ یادگیری انتقالی

یادگیری انتقالی یک روش در حوزه یادگیری ماشین است که با استفاده از دانش ویژگی‌هایی که در یک مسئله یاد گرفته شده‌اند، به بهبود عملکرد در یک مسئله جدید کمک می‌کند. به طور کلی، یادگیری انتقالی به این صورت عمل می‌کند که یک مدل یادگیری عمیق با استفاده از داده‌های موجود در یک مسئله (معمولاً مسئله منبع) آموزش داده می‌شود. سپس، وزن‌های مدل آموزش دیده شده در مسئله منبع را می‌توان به مسئله هدف منتقل کرده و با استفاده از داده‌های موجود در مسئله هدف، آموزش مدل را ادامه داد. اگر مجموعه دادگان محدود باشند می‌تواند منجر به تطبیق بیش از حد یا تعمیم‌پذیری ضعیف شود. در ادامه تعدادی روش برای مقابله با این مشکل مطرح می‌کنیم.

۱-۲-۲-۱ طبقه‌بندی بر اساس معیار فاصله در یک شبکه پیش آموخته

در مسئله طبقه‌بندی با استفاده از معیار فاصله، هدف این است که بر اساس فاصله میان بردارهای ویژگی داده‌ها، آن‌ها را در دسته‌های مختلف تقسیم‌بندی کنیم. برای این منظور، ابتدا یک شبکه عصبی عمیق پیش آموزش دیده روی یک مجموعه دادگان بزرگ آموزش داده می‌شود تا بردارهای ویژگی را به دست بیاوریم. سپس، با استفاده از بردارهای تولید شده، می‌توان از معیارهای فاصله همچون فاصله اقلیدسی، فاصله کسینوسی و فاصله منتهن استفاده کرد. برای مثال مقاله [۲]، از فاصله اقلیدسی برای طبقه‌بندی استفاده می‌کند. در این مقاله برای هر دسته آموزشی، میانگین بردارهای ویژگی تمام نمونه‌های آموزشی آن دسته محاسبه می‌شود. این میانگین به عنوان نماینده دسته استفاده می‌شود. سپس، برای داده‌های تست، بردار ویژگی متناظر استخراج می‌شود و با استفاده از فاصله اقلیدسی، نزدیک‌ترین نمونه‌ای که به آن میانگین نماینده تعلق دارد، پیدا می‌شود. این روش از لحاظ دقت قابل مقایسه با روش‌های فرا یادگیری هست.

۲-۱-۲-۲ آموزش یک طبقه‌کننده جدید با استفاده از یک شبکه پیش آموخته

وقتی تعداد نمونه‌های آموزشی محدود است، آموزش یک طبقه‌کننده از ابتدا بدون استفاده از شبکه آموخته به دلیل بازنمایی^۱ نامناسب، امکان‌پذیر نیست. با این حال، می‌توان بازنمایی‌ها را از یک شبکه آموخته استخراج کرده و سپس با استفاده از آن‌ها یک طبقه‌کننده جدید آموزش داد. در مقاله [۳] این روش را نشان می‌دهند. این مقاله از بازنمایی‌هایی که از یک شبکه پیش آموزش دیده به دست می‌آیند و از نرمال‌سازی L2

^۱ representation

برای آموزش یک طبقه‌بند جدید بر روی مسئله یادگیری با مجموعه داده‌های محدود استفاده می‌کند. این رویکرد به دنبال یافتن نزدیک‌ترین همسایه با استفاده از بارگیری ویژگی‌های استخراج شده از شبکه پیش آموزش دیده، بهبود یافته است.

۲-۱-۳ استنتاج انتقالی با استفاده از تعبیه‌ها از یک شبکه پیش آموخته

روش‌هایی هستند که تلاش می‌کنند با استخراج اطلاعات از مجموعه کوئری، مجموعه کوئری را طبقه‌بندی کنند. از این روش‌ها به عنوان استنتاج انتقالی یاد می‌شود. در این روش‌ها هم از نمونه پشتیبان و هم نمونه کوئری برای تنظیم پارامترها استفاده می‌کنیم. برای مثال در مقاله [۴] یک تابع تخصیص باینری را مینیمم می‌کند که از دو ترم به صورت زیر تشکیل شده است:

$$\varepsilon(Y) = N(Y) + \frac{\lambda}{2} l(Y)$$

که هر کدام از دو ترم به صورت زیر تعریف می‌شوند:

$$N(Y) = \sum_{q=1}^N \sum_{c=1}^C y_{q,c} d(x_q - m_c)$$

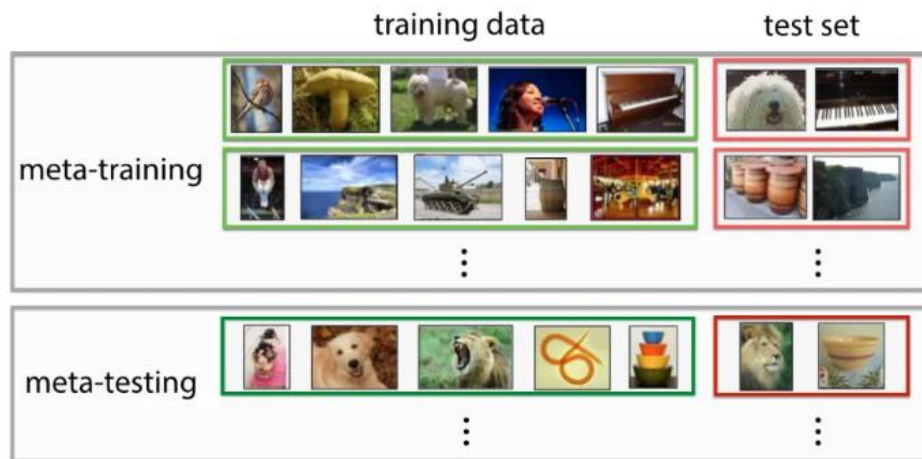
ترم بالا زمانی مینیمم می‌شود که هر مجموعه کوئری به نزدیک‌ترین پروتوتایپ تخصیص داده شود.

$$L(Y) = \frac{1}{2} \sum_{q,p} w(x_q, x_p) \|y_q - y_p\|^2$$

ترم بالا تلاش می‌کند تا دو جفت که به یک لیبل تخصیص دارند را بردار فیچرهای آن دو را به هم نزدیک کند.

۲-۳ حوزه فرا یادگیری

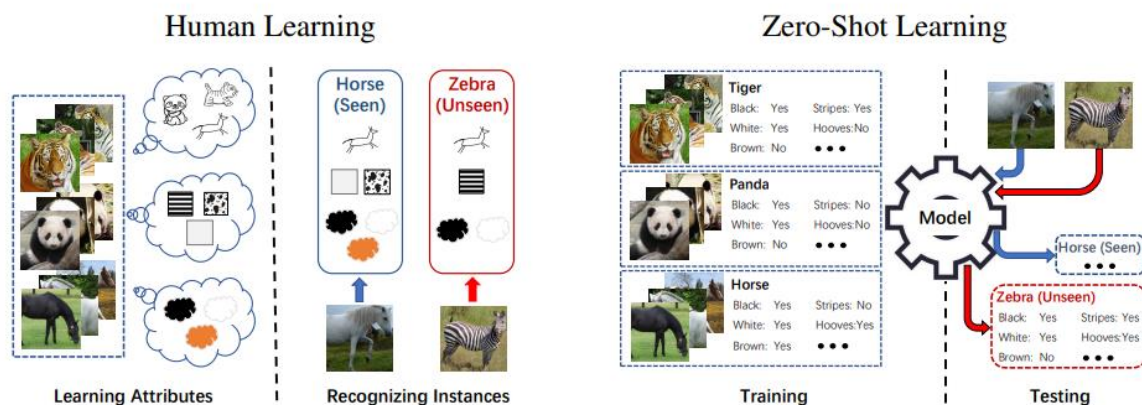
در روش‌های مبتنی بر یادگیری عمیق، استفاده از تعداد زیادی نمونه برچسب گذاری شده برای دستیابی به دقت بالا و عملکرد مناسب نیاز است. این در حالی است که رویکردهای فرا یادگیری در حل مسائل یادگیری بدون نمونه و یادگیری با مجموعه داده‌های محدود استفاده می‌شوند. در واقع با مجموعه داده محدود سعی در یادگیری برای یادگیری دارند. فرا یادگیری یک الگوی یادگیری ماشین است که یاد می‌گیرد که چگونه اطلاعات پیشین را به تسک‌های جدید تعمیم دهد. با توجه به چالش‌های زیادی که در جمع‌آوری مجموعه‌ی داده‌های مناسب وجود دارد، فرا یادگیری بسیار ضروری است. شکل زیر نمونه‌ای از کاربرد فرا یادگیری را نشان می‌دهد که فرا آزمون و فرا آموزش با هم‌پوشانی ندارند.



شکل ۲-۱: نمونه‌ای از کاربرد فرا یادگیری در زمینه طبقه‌بندی تصاویر

۲-۳-۱ یادگیری بدون نمونه

یادگیری بدون نمونه، یک تسک در یادگیری ماشین است که مدل قادر است دسته‌هایی که در طول آموزش ندیده است را طبقه‌بندی کند. ایده یادگیری بدون نمونه برآمده از ذهن انسان است. فرض کنید برای یک کودک ویژگی‌های یک گورخر را توصیف کنید احتمالاً او وقتی عکس‌های متفاوتی از حیوانات می‌بیند می‌تواند گورخر را شناسایی کند. بر اساس اطلاعات کمکی که می‌توانیم برای هر دسته از تصاویر به دست بیاوریم و برخی از نمونه‌های مربوط به آن می‌توان مدلی را برای ارتباط بین نمونه‌ها و اطلاعات کمکی آموزش داد.



شکل ۲-۲: مقایسه یادگیری شبکه بدون نمونه با تفکر انسان

۲-۳-۲ یادگیری با مجموعه دادگان محدود

در این بخش به صورت مفصل به روش‌های مختلف در یادگیری با مجموعه دادگان محدود می‌پردازیم. به طور معمول یادگیری با مجموعه دادگان محدود به سه روش مبتنی بر متریک^۱، مبتنی بر مدل^۲ و مبتنی بر بهینه‌سازی^۳ تقسیم می‌شوند. در ادامه نیز به مکانیزم‌های توجه در این حوزه می‌پردازیم.

۲-۳-۲-۱ روش‌های مبتنی بر مدل

مدل‌های مبتنی بر مدل در تلاش هستند تا با تمرکز بر معماری مدل، پارامترهای مدل را بر اساس وظایف ارائه شده تنظیم کنند و به یادگیری سریع دست یابند. در این روش‌ها، از چندین نوع معماری رایج مدل استفاده می‌شود، از جمله شبکه‌های عصبی کانولوشنی^۴، شبکه‌های عصبی بازگشتی^۵ می‌باشد.

در مقاله [5] مدل پیشنهادی شامل دو بخش اصلی است: شبکه کانولوشنی و لایه توجه به خود، شبکه کانولوشنی که وظیفه اصلی آن تبدیل ورودی به بردارهای ویژگی است. این بردارهای ویژگی، با استفاده از لایه توجه به خود^۶، بر روی بردارهای ویژه تمرکز می‌کند و اطلاعات مناسب‌تری را به دست می‌آورند.

در مقاله [6] الگوریتم جدیدی به نام "Robust-dist" را پیشنهاد می‌دهد که چندین الگوریتم را با هم ادغام می‌کند و نتیجه نهایی را با میانگین‌گیری بین نتایج به دست می‌آورد.

۲-۳-۲-۲ روش‌های مبتنی بر بهینه‌سازی

به دلیل تعداد محدود نمونه‌های آموزشی، در تسک‌های طبقه‌بندی با مجموعه دادگان محدود، یادگیرنده معمولاً دچار تطبیق بیش از حد شود. به علاوه، در فرآیند آموزش، معمولاً یادگیرنده بارها به صورت تکراری آموزش می‌بیند تا در نهایت همگرا شود و به نتیجه مطلوب برسد. این مشکلات نه تنها بر عملکرد یادگیرنده، بلکه بر کارایی مدل برای طبقه‌بندی نیز تأثیر می‌گذارد. روش‌های فرا یادگیری مبتنی بر بهینه‌سازی شاخه مهمی در زمینه یادگیری با مجموعه دادگان محدود هستند. این نوع الگوریتم‌ها سعی می‌کنند از طریق فرا یادگیری، مدل اولیه بهتری را به دست آورده و با استفاده از فرا یادگیرندها، پارامترهای اولیه بهتری را به دست آورند تا یادگیرنده فقط با دیدن تعداد کمی نمونه آموزشی، بتواند در تسک مربوطه سریع‌تر همگرا شود.

در مقاله [7] یک الگوریتم جدید به نام "MAML" ارائه داده است. تعدادی تسک داریم که هر تسک از تعدادی تصاویر پشتیبان و تصویر کوئری تشکیل شده است. در ابتدا پارامترهای شبکه را با توجه به تصاویر

¹ Metric-based

² Model-based

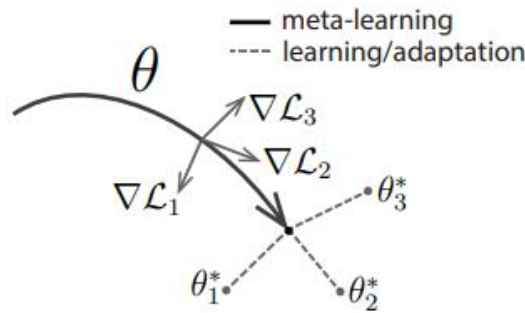
³ Optimization-based

⁴ convolution neural network

⁵ recurrent neural network

⁶ Attention layer

پشتیبان در هر تسک به روزرسانی می‌کنیم و با توجه به مدلی که پارامترهای آن توسط تصاویر پشتیبان آن تسک به روزرسانی شده بر روی تصویر کوئری پارامترهای نهایی مدل را با استفاده از گرادینان نزولی^۱ به روزرسانی می‌کنیم.



شکل ۲-۳: نمودار الگوریتم فرا یادگیری آگنوستیک

از مزیت‌های این الگوریتم می‌توان سازگاری بالا آن هست و می‌توان به مقادیر مناسب‌تری در پارامترهای مدل رسید. ولی یکسری معایب نیز دارد از جمله استفاده از شبکه‌های عمیق در آن ممکن است دشوار باشد، به خصوص زمانی که تعداد مراحل داخلی بیشتر است. در این الگوریتم، هدف اصلی این است که یک مدل را در حین فرآیند آموزش دوباره آموزش دهیم تا بتواند با سرعت بیشتر به مسئله‌های جدیدی که در محیط‌های مختلف ظاهر می‌شوند، سازگار شود. برای این کار، باید گرادینان‌ها را از مسئله‌ی هدف به مدل اصلی منتقل کنیم تا تغییرات لازم را در پارامترها ایجاد کنیم. این فرایند به عنوان یک برگشت عکسی در شبکه‌های عمیق انجام می‌شود. اگر تعداد مراحل داخلی بیشتر شود، برگشت عکسی نیازمند انجام محاسبات بیشتری در عمق شبکه است. به این معنی که گرادینان‌ها باید از طول تعداد مراحل داخلی بیشتری عبور کنند. این موضوع باعث می‌شود که محاسبات مورد نیاز برای آموزش مدل افزایش یابد و به تبع آن، زمان و منابع مورد نیاز برای آموزش نیز افزایش می‌یابد.

۲-۳-۲ روش‌های مبتنی بر متریک

روش‌های مبتنی بر متریک بر اساس فاصله یاد می‌گیرند در واقع تصویر را به یک بردار ویژگی نگاشت می‌کنیم. فاصله بردارهای ویژگی تصاویر از همدیگر می‌تواند بیانگر شباهت‌ها و تفاوت‌های بین تصاویر باشد. فرض کنید دو تصویر به صورت (x_1, y_1) و (x_2, y_2) داریم که x_i ها بیانگر بردارهای ویژگی و y_i ها بیانگر برچسب تصاویر هستند. حالا یک تصویر کوئری که بردار ویژگی آن x_3 هست را در نظر بگیرید اگر d تابع فاصله باشد در آن صورت $d(x_1, x_3)$ و $d(x_2, x_3)$ را محاسبه می‌کنیم و برچسب تصویر کوئری را به

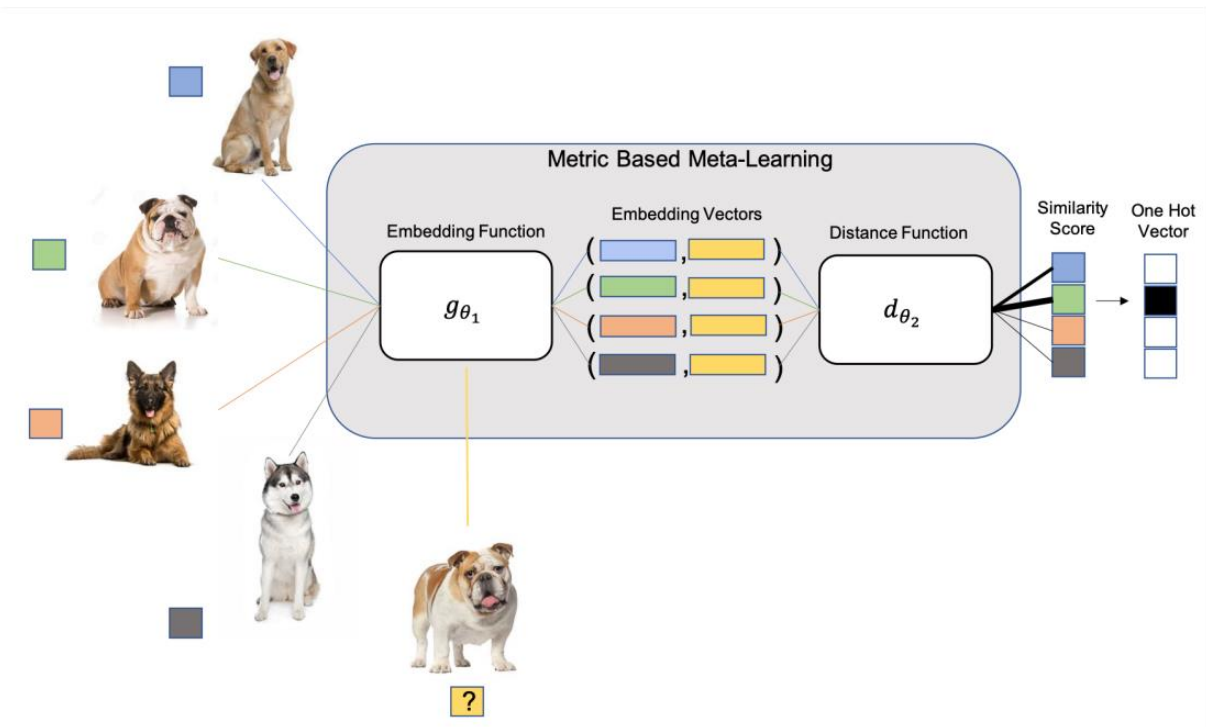
^۱ gradient descent

برچسب با فاصله کوتاه‌تر نسبت می‌دهیم. این همان ایده اصلی نزدیک‌ترین همسایه (KNN) هست. برای ورودی‌های با بعد بیشتر مثل تصاویر از یک تابع تعبیه g استفاده می‌کنیم که تعداد بعد را کاهش دهیم و قبل از محاسبه فاصله استفاده می‌شود:

$$g: R^n \rightarrow R^m \text{ where } n > m$$

هدف اصلی یادگیری مبتنی بر متریک یادگیری تابع تعبیه $g(\cdot; \theta_1)$ که پارامترهای مدل است بر اساس تابع فاصله d می‌باشد. همچنین به جای استفاده از تابع فاصله مشخص مانند تابع اقلیدسی می‌توان آن را نیز یاد گرفت.

در ابتدا از مجموعه آموزشی به صورت M کلاس K تصویر (M way K shot) نمونه‌برداری می‌کنیم و سپس شروع به آموزش مدل می‌کنیم. در هر تکرار برای هر کدام یک از تسک‌ها، خطای شبکه بر روی مجموعه کوئری حساب می‌شود. سپس میانگین خطاها را حساب می‌کنیم پارامترهای تابع تعبیه و تابع فاصله را به‌روزرسانی می‌کنیم. در نهایت مدل را بر روی مجموعه تست که به صورت M کلاس K تصویر هست ارزیابی می‌کنیم [۸]. در ادامه به بررسی چندین روش در حوزه روش‌های مبتنی بر متریک می‌پردازیم.

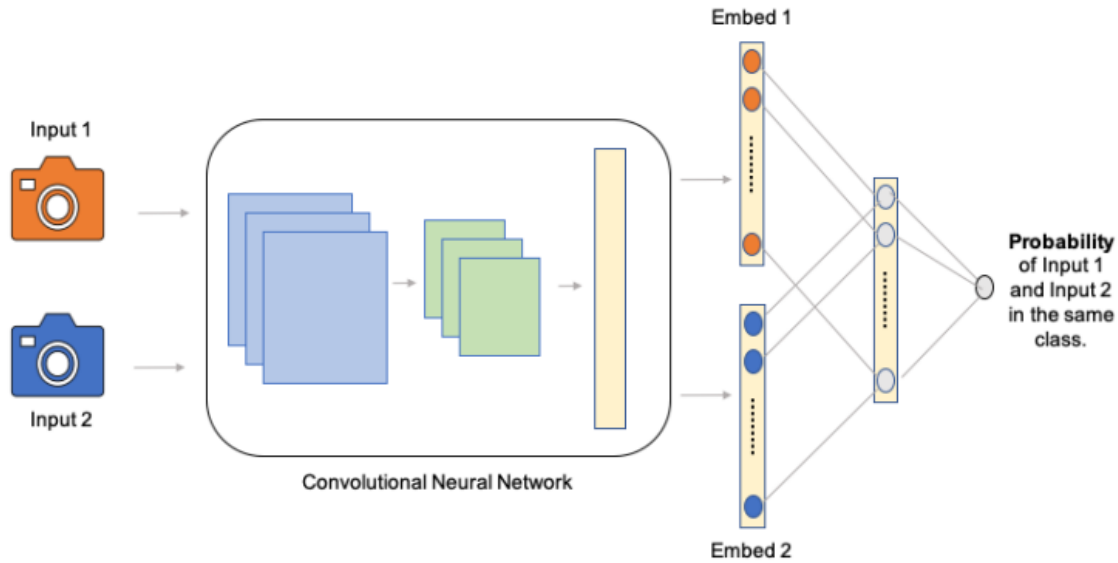


شکل ۲-۴: نمونه‌ای از مسئله فرا یادگیری مبتنی بر متریک برای یک تسک ۴ کلاس ۱ تصویر.

۲-۳-۱ شبکه عصبی Siamese

در شبکه Siamese طبق مقاله [۹] یک جفت شبکه عصبی با وزن‌های یکسان هستند. دو تصویر به این شبکه عصبی داده می‌شود و دو بردار ویژگی مربوط به هر کدام از تصاویر به دست می‌آید. در ادامه فاصله بین این دو بردار ویژگی به دست می‌آید. عددی که به دست می‌آید نشان دهنده این است که دو تصویر چه

میزان به هم شباهت دارند. در واقع خروجی شبکه ایده آل به این صورت است که اگر دو تصویر به یک دسته تعلق داشتند برابر با یک و اگر نه برابر با صفر باشد. این مسئله که یادگیری تک شات با محاسبه میزان شباهت بود بسیار مورد توجه قرار گرفت. همان طور که در شکل زیر مشخص است تابع تعبیه ما یک شبکه کانولوشنی است. و پس از محاسبه تابع فاصله دو بردارهای ویژگی یک لایه تماماً متصل^۱ و پس از آن نیز یک تابع sigmoid داریم که خروجی آن عددی بین صفر و یک است. برای تابع ضرر این شبکه، از تابع آنروپی متقاطع دودویی^۲ استفاده شده است.



شکل ۲-۵: شبکه عصبی siamese

۲-۳-۲-۳-۲ شبکه تطبیق

در شبکه عصبی تطبیق طبق مقاله [۱۰] یک مجموعه پشتیبان به صورت $S = \{x_i, y_i\}_{i=1}^K$ و یک کوثری به صورت \hat{x} داریم و با استفاده از یک کرنل توجه به نام $a(\hat{x}, x_k)$ توزیع احتمال را بر روی برچسب y محاسبه می کند. کرنل توجه شباهت کسینوسی^۳ بین بردارهای تعبیه را مجموعه پشتیبان و کوثری را حساب می کند و سپس با استفاده از فرمول softmax آن را به صورت زیر نرمالیزه می کند:

$$a(\hat{x}, x_k) = \frac{e^{\cos(f(\hat{x}), g(x_k))}}{\sum_{k=1}^t e^{\cos(f(\hat{x}), g(x_k))}}$$

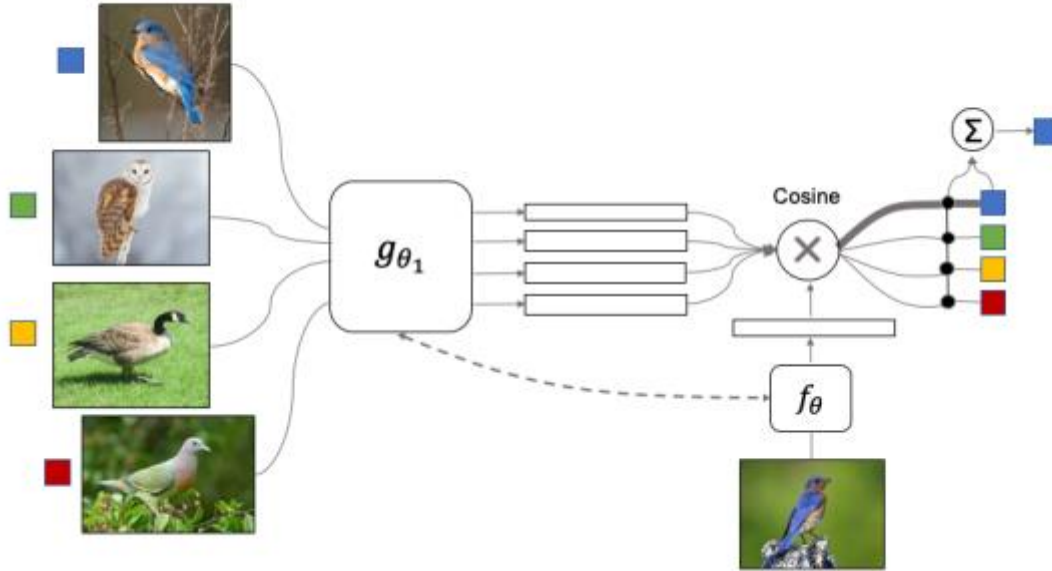
خروجی نهایی هم به صورت ضرب کرنل توجه در برچسب ها به صورت زیر محاسبه می شود:

^۱ Dense layer

^۲ Binary cross-entropy

^۳ cosine similarity

$$P(y|\hat{x}, S) = \sum_{k=1}^K a(\hat{x}, x_k) y_k$$



شکل ۲-۶: شبکه تطبیق

۲-۳-۲-۳-۳ شبکه prototypical

شبکه prototypical طبق مقاله [۱۱] به این صورت هست اگر تعداد تصاویر در هر مجموعه پشتیبان بیشتر از یک بود مثلاً ۵ شات بود می‌توان از آن استفاده کرد. از هر مجموعه پشتیبان یک بردار نماینده استخراج می‌شود. در واقع بردارهای ویژگی از هر عضو مجموعه پشتیبان استخراج می‌شود و بین بردارهای ویژگی میانگین گرفته می‌شود تا بردار نماینده به دست بیاید مطابق فرمول زیر:

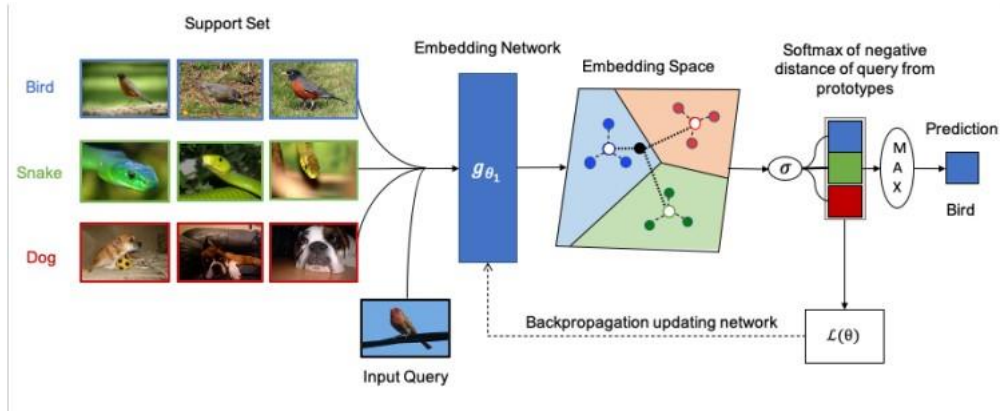
$$v_c = \frac{1}{|S^c|} \sum_{(x_k, y_k) \in S^c} g_{\theta_1}(x_k)$$

و بعد فاصله بین بردار ویژگی کوئری تا هر یک از بردارهای نماینده حساب می‌شود. در این شبکه از فاصله اقلیدسی استفاده شده است. احتمال تعلق هر تصویر کوئری به بردارهای نماینده با استفاده از فرمول softmax به صورت زیر محاسبه می‌شود:

$$P(y = c|\hat{x}) = \text{softmax} \left(-d(g_{\theta_1}(\hat{x}), v_c) \right) = \frac{e^{(-d(g_{\theta_1}(\hat{x}), v_c))}}{\sum_{\hat{c} \in C} e^{(-d(g_{\theta_1}(\hat{x}), v_{\hat{c}}))}}$$

همچنین میزان پس از انتشار^۱ بر اساس تابع ضرر به صورت زیر محاسبه می‌شود:

$$L(\theta_1) = -\log P_{\theta_1}(y = c|\hat{x})$$



شکل ۷-۲: شبکه prototypical

۴-۳-۲-۳-۲ شبکه relation

شبکه relation طبق مقاله [۱۲] دیگر تابع فاصله‌ای به صورت جدا نداریم. در واقع بازنمایی مجموعه پشتیبان و کوئری به هم متصل می‌شوند و در شبکه عصبی کانولوشن میزان شباهت به دست می‌آید. خروجی شبکه عددی بین صفر و یک هست. اگر v_c را نماینده مجموعه پشتیبان در نظر بگیریم و \hat{x} نیز کوئری ما باشد خروجی به صورت زیر محاسبه می‌شود:

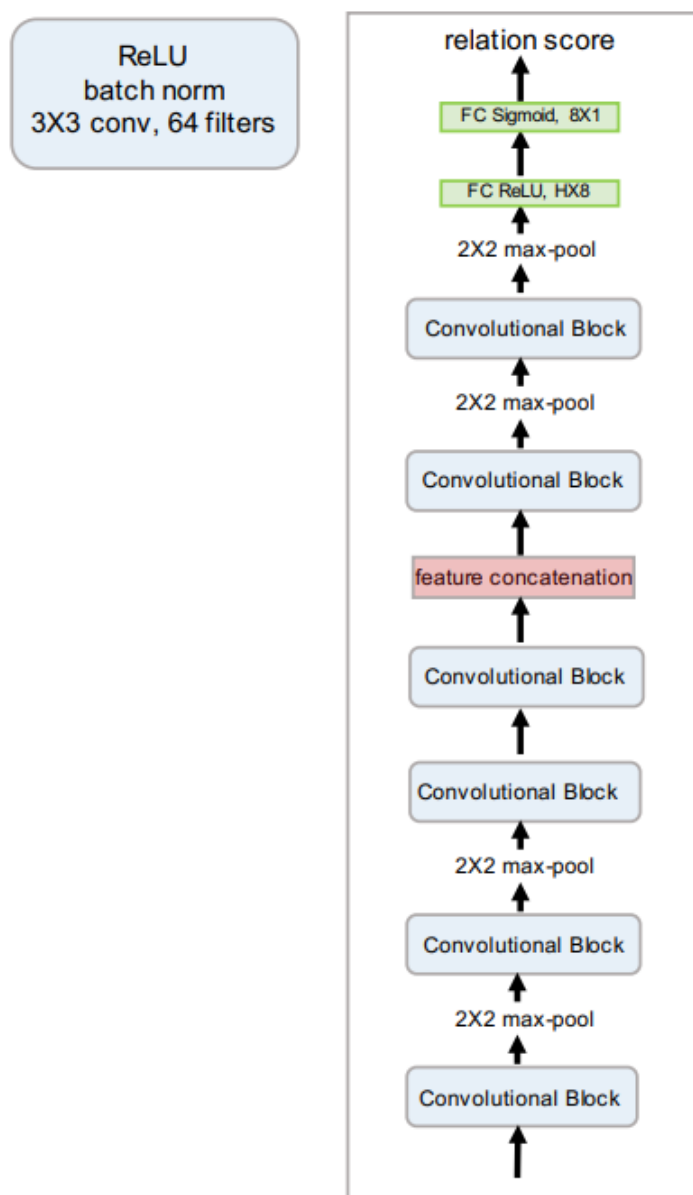
$$r_c = d_{\theta_2}(g_{\theta_1}(\hat{x}) \cup v_c)$$

و بهینه‌ساز نیز به صورت زیر است:

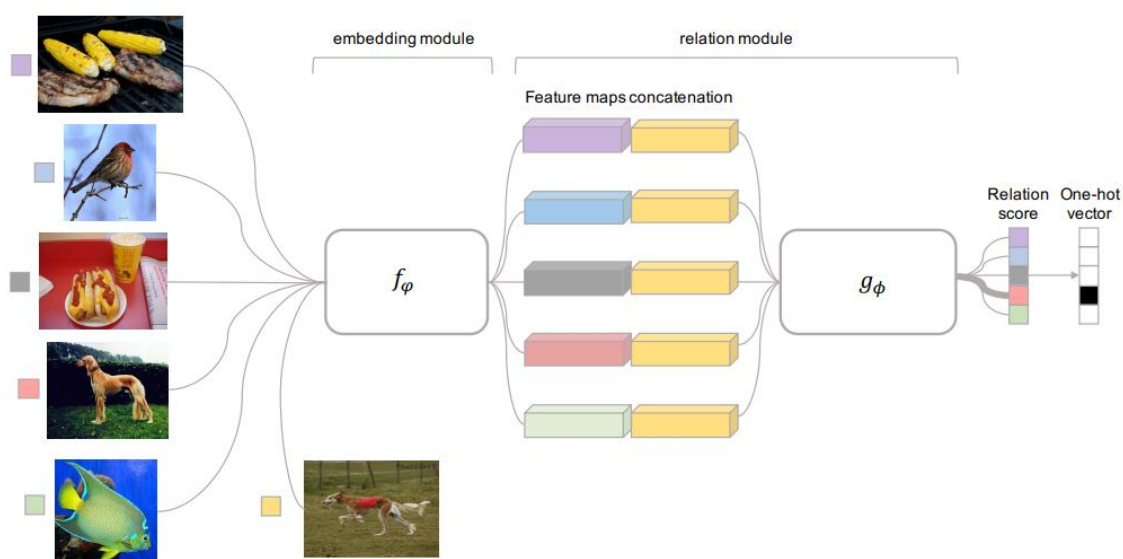
$$\theta_1, \theta_2 \leftarrow \arg \min_{\theta_1, \theta_2} \sum_{c \in C} (r_c - 1(y == c))^2$$

شکل زیر ساختار کانولوشنی شبکه relation را نشان می‌دهد:

^۱ Back-propagation



شکل ۲-۸: معماری شبکه *relation* برای یادگیری با مجموعه دادگان محدود که از عناصری از جمله شبکه کانولوشنی تشکیل شده است.



شکل ۲ - ۹: شبکه *relation* برای مسئله یادگیری ۵ کلاسه تک شات به همراه یک تصویر کوثری

۲-۳-۲-۵ یادگیری متمرکز بر وظیفه برای دسته‌بندی تصاویر با استفاده از استخراج ویژگی پویا و معیار فاصله قابل تطبیق (TADAM)

طبق مقاله [۱۳] یک روش یادگیری متمرکز بر وظیفه برای دسته‌بندی تصاویر است. در این روش، یادگیری مسیرهای جداگانه برای هر وظیفه^۱ انجام می‌شود و معیار فاصله نیز با توجه به وظیفه تعیین می‌شود.

مراحل اصلی TADAM عبارت‌اند از:

۱. استخراج ویژگی^۲: ابتدا تصاویر ورودی با استفاده از یک شبکه عصبی عمیق مانند شبکه کانولوشنی بردار ویژگی آن به دست می‌آید. این نمایش ویژگی برای هر تصویر به عنوان ورودی به مراحل بعدی استفاده می‌شود.

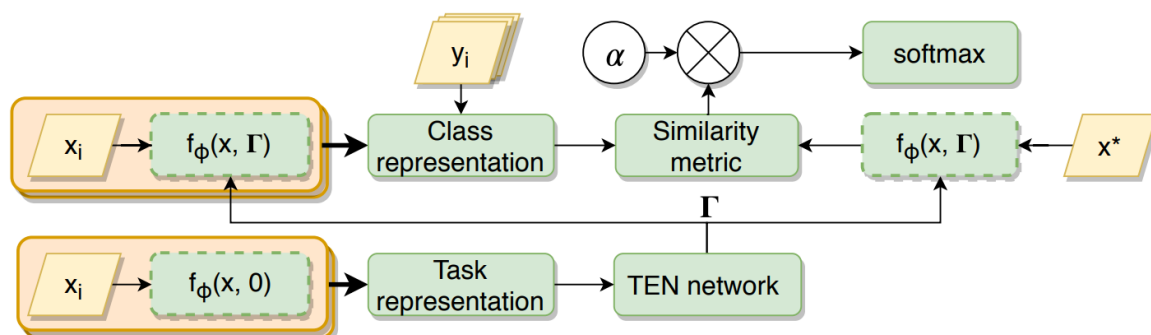
۲. استخراج نمایش وظیفه: با استفاده از یک استخراج‌کننده ویژگی پویا، نمایش وظیفه برای هر تصویر ایجاد می‌شود. این استخراج‌کننده ویژگی پویا توسط یک مدل مانند یک شبکه بازگشتی با حافظه کوتاه مدت پیاده‌سازی می‌شود و با استفاده از اطلاعات وظیفه موجود، نمایش وظیفه مربوطه را تولید می‌کند.

۳. محاسبه فاصله و تصمیم‌گیری: با استفاده از نمایش ویژگی تصاویر و نمایش وظیفه، یک معیار فاصله برای هر تصویر محاسبه می‌شود. این معیار فاصله با توجه به وظیفه مشخص شده است که می‌تواند معیار فاصله اقلیدسی یا معیار فاصله کسینوسی باشد. معیار فاصله با استفاده از یک دمای یادگیری پذیر نیز مقیاس‌بندی می‌شود.

^۱ Task

^۲ Feature extraction

۴. آموزش و بهینه‌سازی: با استفاده از تابع هزینه وظیفه مشخص شده و معیار فاصله محاسبه شده، شبکه عصبی آموزش داده می‌شود تا بتواند به درستی دسته‌بندی تصاویر را در وظایف مختلف انجام دهد. این شبکه با استفاده از یک الگوریتم بهینه‌سازی مانند نزول گرادیانی بهینه‌سازی می‌شود.



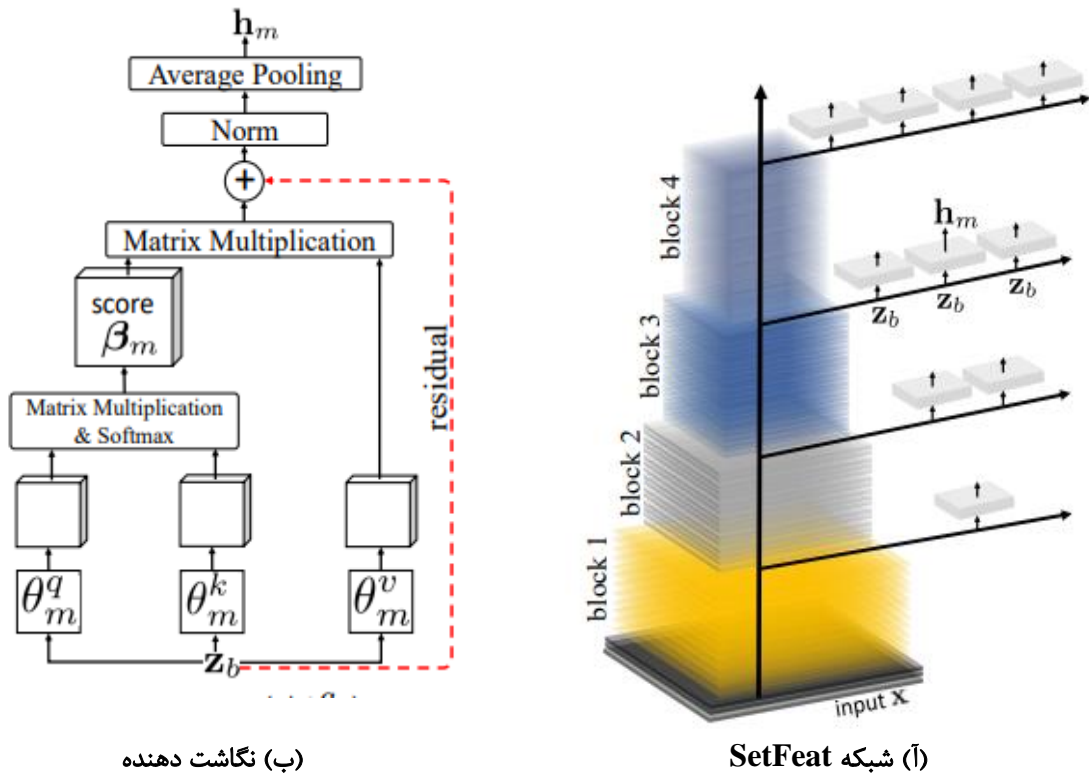
شکل ۲ - ۱۰: معماری چند شات پیشنهادی بلوک‌های دارای پارامترهای مشترک دارای مرز چین‌دار هستند

۲-۳-۲ روش‌های جدید در یادگیری با مجموعه داده محدود

در سال‌های اخیر روش‌های متعددی برای افزایش عملکرد شبکه‌های مجموعه داده‌گان محدود داده شده است. بردار فیچر یکی از نکاتی است که در این شبکه‌ها بسیار حائز اهمیت است. تلاش شده است که با استفاده از روش‌هایی بردار فیچرهای بهتری استخراج شود. برای بردار فیچر معمولاً از شبکه‌های پیش‌آمورخته‌ای همچون Resnet و VGG استفاده می‌شود. یکی از راه‌حل‌هایی که عملکرد شبکه را بهتر می‌کند مکانیزم توجه به خود بعد از استخراج بردار فیچر از شبکه‌های پیش‌آمورخته است در ادامه به بررسی این روش می‌پردازیم.

یکی از روش‌های جدید شبکه SetFeat است که طبق مقاله [۱۴] در ابتدا از یک شبکه پیش‌آمورخته یا یک شبکه کانولوشنی استفاده می‌کند و بردارهای ویژگی را در هر stage استخراج می‌کند و آن‌ها را به یک نگاشت دهنده^۱ می‌دهد. در واقع هر نگاشت دهنده خود یک بلوک توجه به خود هست و باعث می‌شود ویژگی‌های بهتری استخراج شود. نمای کلی آن به صورت زیر است:

^۱ mapper



شکل ۲ - ۱۱: نمای کلی شبکه SetFeat

برای محاسبه تابع فاصله با توجه به اینکه از مجموعه پشتیبان و مجموعه کوئری از هر یک چندین بردار ویژگی استخراج می‌شود در مقاله چندین روش پیشنهاد شده است که به صورت زیر است:

۱. جمع-نگاشت^۱

فاصله بین نگاشت دهنده و کوئری را جمع می‌کند مطابق فرمول زیر:

$$d_{ms}(x_q, S^n) = \sum_{i=1}^M d(h_i(x_q), \bar{h}_i(S^n))$$

۲. مینیمم-مینیمم^۲

از حداقل فاصله ممکن استفاده می‌کند بین نماینده هر مجموعه پشتیبان و کوئری مطابق فرمول

زیر:

$$d_{mm}(x_q, S^n) = \min_{i=1}^M \min_{j=1}^M d(h_i(x_q), \bar{h}_j(S^n))$$

¹ Match-sum

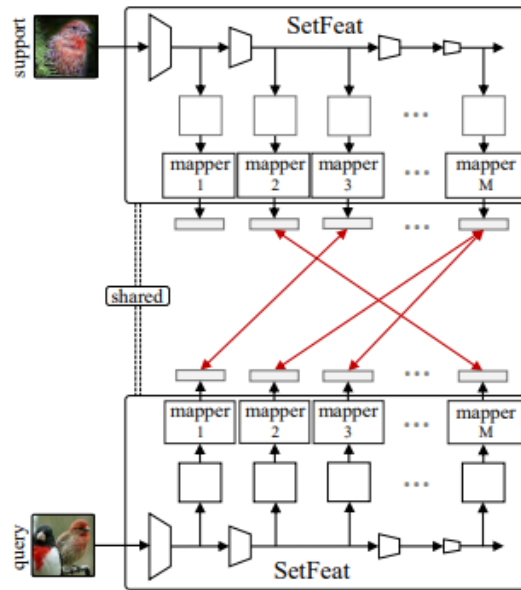
² Min-min

۳. جمع-مینیمم^۱

جمع حداقل فاصله ممکن بین نماینده هر مجموعه پشتیبان و کوئری که مطابق فرمول زیر محاسبه می‌شود:

$$d_{sm}(x_q, S^n) = \sum_{i=1}^M \min_{j=1}^M d(h_i(x_q), \bar{h}_j(S^n))$$

که اگر از این تابع فاصله استفاده کنیم شکل کلی شبکه به صورت زیر است:



شکل ۲ - ۱۲: شبکه *setfeat* وقتی از فاصله *sum-min* استفاده می‌شود

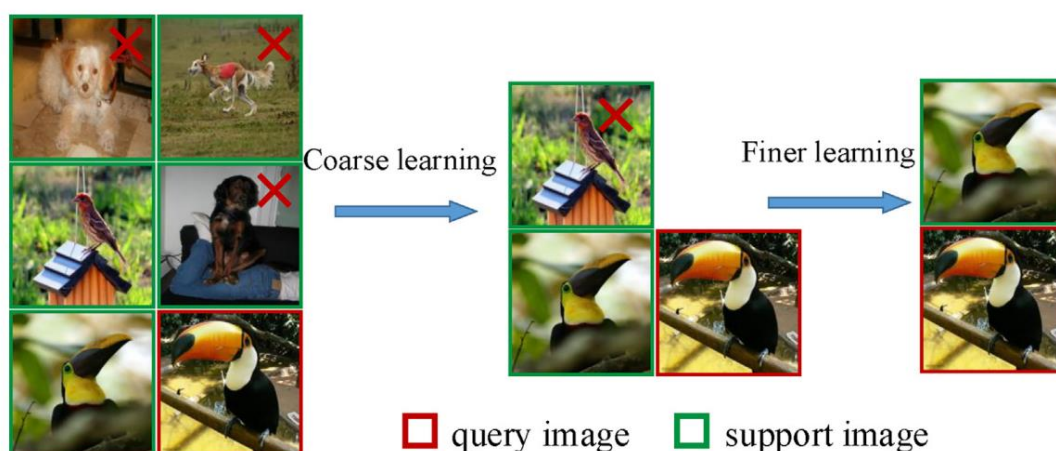
و در نهایت اینکه هر تصویر کوئری متعلق به کدام مجموعه پشتیبان است با استفاده از فرمول softmax به صورت زیر محاسبه می‌شود:

$$p(y = n | x_q, S) = \frac{\exp(-d_{set}(x_q, S^n))}{\sum_{S^i \in S} \exp(-d_{set}(x_q, S^n))}$$

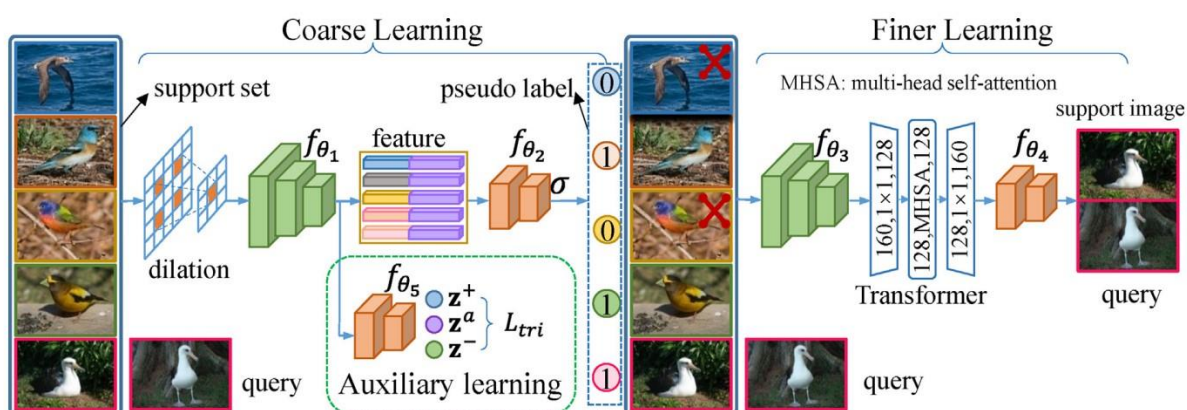
یکی دیگر از روش‌ها یادگیری دو مرحله‌ای هست. که در ابتدا با یک دید کلی تصاویری در مجموعه پشتیبان که فاصله زیادی با تصویر کوئری دارند را حذف می‌کند. در مرحله بعد با یک دید دقیق‌تر سعی بر دسته‌بندی صحیح تصویر کوئری دارد. طبق مقاله [۱۵] در ابتدا پس از انجام یک گسترش بر روی تصاویر با بهره‌گیری از Resnet12 بردارهای ویژگی برای تصاویر پشتیبان و کوئری استخراج می‌کند. در این مقاله برای بهبود توانایی انتقال دانش از یک ماژول کمکی بهره گرفته است. این ماژول از یک تابع ضرر triplet

^۱ Sum-min

استفاده می‌کند پس از حذف تعدادی از دسته‌ها در مرحله اول، در مرحله دوم از ترانسفورمر در میان بلوک‌های ResNet12 استفاده می‌کند تا برچسب صحیح از مجموعه پشتیبان انتخاب شود.



شکل ۲- ۱۳: فرایند طبقه‌بندی دو مرحله‌ای

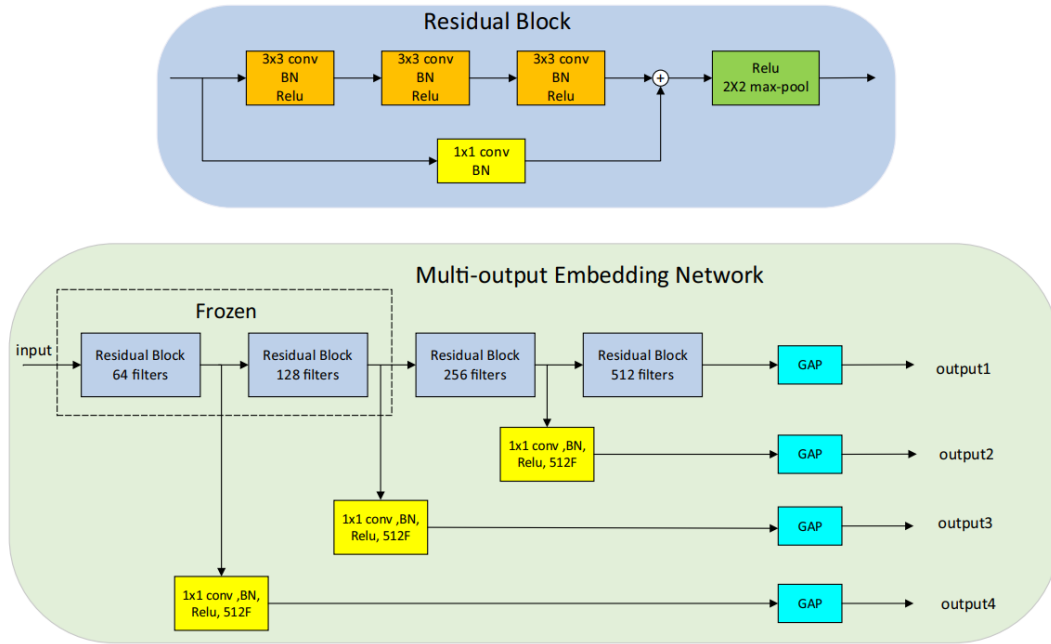


شکل ۲- ۱۴: ساختار شبکه آموزش دو مرحله‌ای CFMA

یکی دیگر از روش‌های معمول استفاده از فضاها^۱ تعبیه‌متعدد هست. در این روش به جای گرفتن فقط یک خروجی از تابع تعبیه، از stage های مختلف خروجی گرفته می‌شود که هر stage دید متفاوتی نسبت به تصویر دارد. مقاله [۱۶] از ResNet12 که دارای چهار بلوک Residual هست استفاده کرده است. بعد از خروجی گرفتن از هر بلوک یک استخراج میانگین سراسری^۲ می‌زند تا بردارهایی با دید متفاوت از تصویر داشته باشیم. همچنین برای اینکه تعداد کانال‌های هر بردار ویژگی برابر یک کانولوشن 1×1 زده می‌شود.

^۱ Embedding Spaces

^۲ global average pooling



شکل ۲-۱۵: شبکه تعبیه چند خروجی

۴-۲ توابع ضرر در مجموعه دادگان محدود

یکی از پارامترهایی که بر روی مدل‌های یادگیری عمیق تأثیر می‌گذارد، توابع ضرر هست. تابع ضرر عملکرد مدل را با مقایسه خروجی پیش‌بینی شده و خروجی واقعی اندازه‌گیری می‌کند. در واقع هدف این است که تابع ضرر را به کمترین مقدار ممکن برسانیم. انتخاب تابع ضرر به تسک مورد نظر بستگی دارد در این بخش به تعدادی از معروف‌ترین توابع ضرر در حوزه دسته‌بندی تصاویر با مجموعه داده محدود می‌پردازیم.

۱-۴-۲ تابع ضرر Contrastive

تابع ضرر contrastive معمولاً در شبکه Siamese مورد استفاده قرار می‌گیرد. به طور معمول زمانی که هدف پیدا کردن تصاویر مشابه بر اساس بردار ویژگی است از تابع ضرر contrastive استفاده می‌شود. در شبکه Siamese دو تصویر به صورت هم‌زمان وارد تابع تعبیه یکسان می‌شوند. پس در خروجی دوتا بردار ویژگی خواهیم داشت. تابع ضرر فاصله بین دو بردار ویژگی را اندازه‌گیری می‌کند. معیار اندازه‌گیری بستگی به مسئله دارد اما معیارهایی که معمولاً استفاده می‌شوند فاصله اقلیدسی و شباهت کسینوسی است. ایده اصلی این تابع ضرر تشویق به تولید بردارهای ویژگی یکسان اگر متعلق به یک کلاس هستند و ایجاد بردارهای ویژگی متفاوت اگر متعلق به یک کلاس نیستند. فرمول ریاضی آن به صورت زیر است:

$$L_{con} = \frac{1}{2} (1 - y) \times D_w^2 + \frac{1}{2} y \times \max(0, m - D_w)^2$$

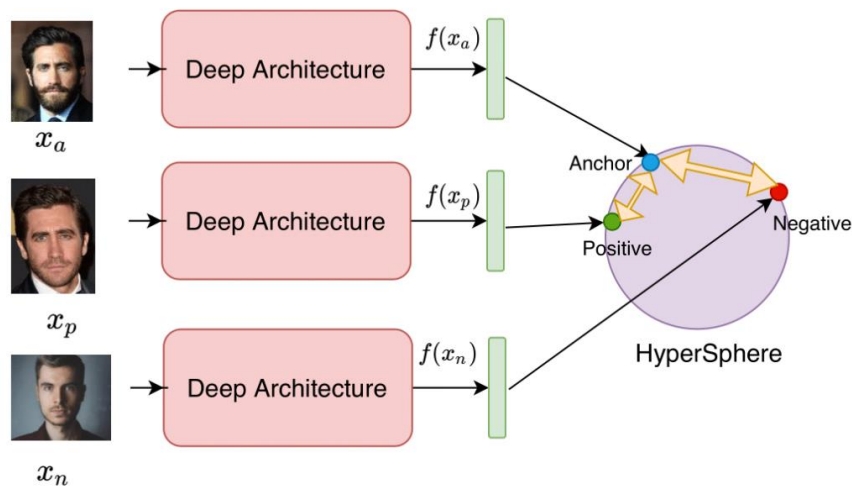
طبق فرمول بالا تابع ضرر تلاش می کند تا فاصله تصاویر مشابه به یک کلاس را کمینه و فاصله بین تصاویری که متعلق به یک کلاس نیستند را بیشینه کند. در رابطه بالا D_W فاصله بین دو بردار ویژگی و γ اگر دو نمونه مشابه باشند برابر با ۰ و اگر غیر مشابه باشند برابر با ۱ هست. همچنین m یک حد آستانه است که تعیین می کند نمونه های غیر مشابه چقدر بردار ویژگی های آنها باید از هم فاصله داشته باشد.

۲-۴-۲ تابع ضرر شبکه تطبیق

تابع ضرر تطبیق یکی دیگر از توابع ضرر هست که در یادگیری با مجموعه داده گان محدود مورد استفاده قرار می گیرد. این تابع برای طبق بندی نمونه کوئری از طریق مقایسه آن با مجموعه پشتیبان و تعیین میزان شباهت با هر کدام از تصاویر پشتیبان مورد استفاده قرار می گیرد. در این شبکه از یک مکانیزم توجه برای محاسبه جمع وزن دار بردارهای نماینده متعلق به هر کلاس استفاده می شود و پس از آموزش میزان شباهت بردار ویژگی کوئری با هر یک از بردارهای نماینده کلاس ها سنجیده می شود و پس از آن با یک تابع softmax میزان شباهت ها نرمال می شوند.

۳-۴-۲ Triplet تابع ضرر

ایده تابع ضرر Triple همانند Contrastive است و همچنین در شبکه های Siamese نیز استفاده می شود. این تابع ضرر مانند Contrastive مدل را تشویق به تولید بردارهای ویژگی مشابه برای نمونه های متعلق به یک کلاس و ایجاد بردارهای ویژگی متفاوت برای دو کلاس متفاوت می کند. ولی Triple هر دو مورد را به صورت همزمان انجام می دهد. برخلاف Contrastive ورودی شبکه سه تصویر است. یک تصویر لنگر، یک تصویر از کلاس لنگر^۱ و یک تصویر نیز از کلاس دیگر است. در حالت ایده آل، فاصله تصویر لنگر از تصویر مشابه باید کمینه و با تصویر از یک کلاس دیگر بیشینه باشد.



شکل ۲-۱۶: آموزش شبکه Siamese با تابع ضرر Triplet

^۱ anchor

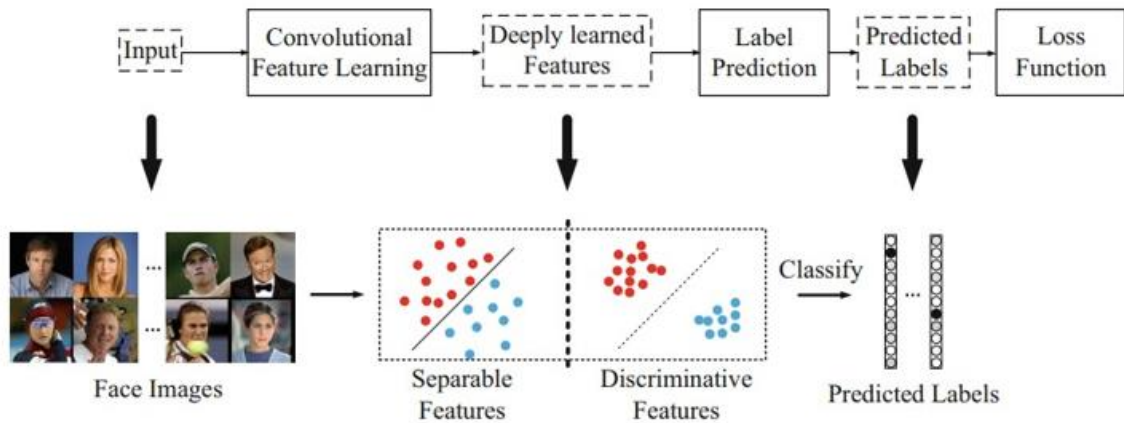
فرمول تابع ضرر Triplet نیز به صورت زیر است:

$$L_{triplet} = \max (\|f(A) - f(p)\|^2 - \|f(A) - f(N)\|^2 + m, 0)$$

که f بیانگر تابع تعبیه هست و همچنین m یک حد آستانه است که مشخص می‌کند تعبیه های نمونه های غیرمشابه و لنگر چقدر باید از هم فاصله داشته باشند.

۲-۴-۴ تابع ضرر Center

طبق مقاله [17] علاوه بر اینکه تلاش می‌شود تا تصاویر را دسته بندی شود تلاش می‌شود تا بردار فیچرهای هر کلاس به هم نزدیک شوند به همین علت علاوه بر cross entropy از تابع ضرر دیگری به نام Center نیز استفاده می‌شود شکل زیر شهود خوبی از این تابع ضرر می‌دهد:



شکل ۲-۱۷: هدف تابع ضرر center نزدیک کردن ویژگی‌های هر کلاس به همدیگر است

هدف اصلی تابع ضرر Center، ایجاد فضایی است که نماینده‌های هر دسته به طور جداگانه و دسته‌ها از یکدیگر به طور کامل تمایز پیدا کنند. برای این منظور، از مفهوم مرکز برای هر دسته استفاده می‌شود.

مرکز یک دسته برابر با میانگین نمونه‌های آن دسته در فضای ویژگی‌ها است. به عبارت دیگر، برای هر دسته یک مرکزی تعیین می‌شود که نماینده آن دسته است. در واقع مرکزها نیز در فرایند آموزش آپدیت میشوند. تابع ضرر Center معمولاً با تابع ضرر cross entropy که در آموزش شبکه عصبی عمومی استفاده می‌شود، ترکیب می‌شود. تابع ضرر Center قصد دارد فاصله بین ویژگی‌های استخراج شده از تصاویر و مراکز متناظر دسته‌ها را کاهش دهد. با کاهش این فاصله، نماینده‌های هر دسته به طور جداگانه و دسته‌ها از یکدیگر به طور کامل تمایز پیدا می‌کنند. فرمول آن به صورت زیر است:

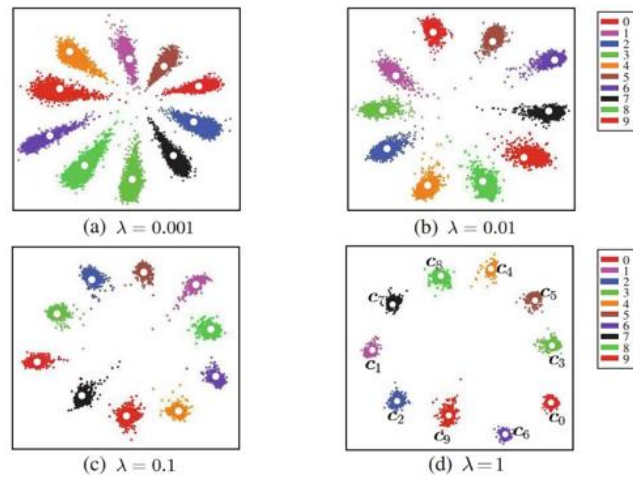
$$L_c = \frac{1}{2} \sum_{i=1}^m \|x_i - c_{y_i}\|_2^2$$

منتها تابع ضر center با cross entropy ترکیب می شود به این علت که اگر فقط از فرمول بالا استفاده کنیم احتمالاً همه مرکزها رو هم می افتند و طبقه بندی تصاویر انجام نمی شود و اگر فقط از cross entropy استفاده کنیم جداسازی ویژگی های هر کلاس به صورت کامل اتفاق نمی افتد فرمول ترکیب این دو تابع ضرر به صورت زیر است:

$$L = L_S + \lambda$$

$$= \sum_{i=1}^m \log \frac{e^{W_{y_i}^T x_i + b_{y_i}}}{\sum_{j=1}^n e^{W_j^T x_i + b_j}} + \frac{\lambda}{2} \sum_{i=1}^m \|x_i - c_{y_i}\|_2^2$$

پارامتر λ اهمیت بسیار زیادی در جداسازی ویژگی های هر کلاس دارد در تصویر زیر اهمیت آن مشخص است:



شکل ۲-۱۸: تأثیر پارامتر λ بر روی طبق بندی کردن تصاویر

۵-۲ معیارهای ارزیابی

در این بخش به بررسی برخی از معیارهای مهم در مسائل طبقه بندی با مجموعه دادگان محدود می پردازیم. ماتریس درهم ریختگی^۱ یکی از معیارهای بسیار مهم است که شامل چهار مقدار مختلف به صورت زیر است:

✓ مثبت درست: دسته هایی که برچسب آن ها مثبت است و مدل نیز به درستی پیش بینی کرده است.

✓ مثبت نادرست: دسته هایی که برچسب آن ها منفی است و مدل آن ها را مثبت پیش بینی کرده است.

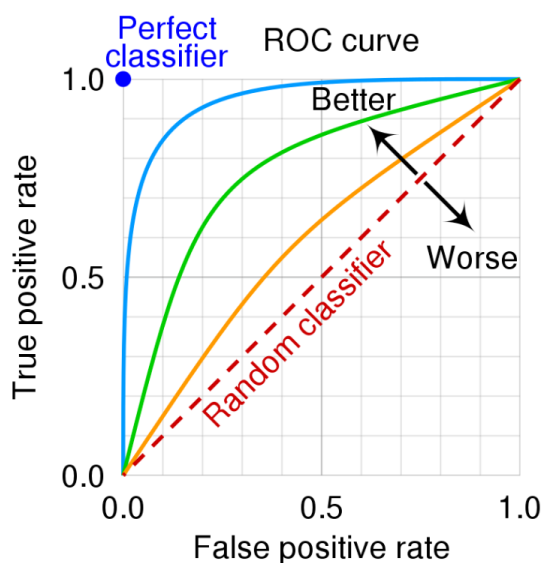
^۱ Confusion matrix

✓ منفی نادرست: دسته هایی که برچسب آن ها مثبت است و مدل آن ها را منفی پیش بینی کرده است.

✓ منفی درست: دسته هایی که برچسب آن ها منفی است و مدل نیز به درستی پیش بینی کرده است.

۲-۵-۱ معیار ROC-AUC

معیار ROC-AUC یک معیار ارزیابی عملکرد برای مدل های دسته بندی باینری است. این معیار بر اساس منحنی ROC ساخته شده توسط مدل بر روی مجموعه داده تست محاسبه می شود. منحنی ROC یک نمودار دوبعدی است که نشان می دهد که مدل در آستانه های مختلف، چه تعادلی بین نرخ تشخیص درست و نرخ تشخیص نادرست دارد. محور افقی منحنی ROC نمایش دهنده False Positive Rate است که نسبت تعداد نمونه هایی که اشتباهاً به عنوان مثبت تشخیص داده شده اند به تعداد کل نمونه های منفی است. محور عمودی نمایش دهنده True Positive Rate یا همان نرخ تشخیص درست است که نسبت تعداد نمونه هایی که به درستی به عنوان مثبت تشخیص داده شده اند به تعداد کل نمونه های مثبت است. معیار AUC نیز مساحت زیر منحنی ROC را نشان می دهد و ارزیابی می کند که مدل چقدر توانایی تفکیک بین نمونه های مثبت و منفی را دارد. مقدار AUC بین ۰ تا ۱ قرار می گیرد، که یک مدل با AUC برابر با ۱ به معنای دقت بسیار بالا در تشخیص بین دو کلاس است، در حالی که AUC برابر با ۰.۵ نشان دهنده یک مدل تصادفی است و AUC کمتر از ۰.۵ به معنای یک مدلی است که عملکرد نسبت به تصادف بدتر است. بنابراین، ROC-AUC معیاری است که ارزیابی می کند که مدل چقدر توانایی تمایز بین کلاس های مثبت و منفی را دارد، و با محاسبه مساحت زیر منحنی ROC، این توانایی را به صورت عددی بیان می کند.



شکل ۲-۱۹: فضای ROC برای طبقه بندی بهتر و بدتر

فصل ۳

روش‌های پیشنهادی

۳-۱ مقدمه

در این بخش ابتدا اقدام به پیاده‌سازی مدل Prototypical بر روی دیتاست MiniImagenet شد. سپس تلاش شد با کارهایی همچون تغییر backbone ، multiscale کردن ، دادن وزن به هر یک از stage ها و اضافه کردن مکانیزم توجه به خود دقت مدل را افزایش دهیم. سپس مدل را بر روی دیتاست تصاویر هوایی نیز تست کردیم که نشان دهیم مدل تعمیم‌پذیری خوبی دارد. همچنین در ابتدا به بررسی چالش‌های موجود در زمینه طبقه‌بندی تصاویر با مجموعه دادگان محدود و نحوه مقابله با آن می‌پردازیم.

۳-۲ چالش‌های مربوط به دست بندی تصاویر با مجموعه

دادگان محدود

با توجه به اینکه یکی از ارکان اصلی مسائل یادگیری عمیق تعداد داده آموزشی هست در هنگامی که تعداد داده‌ها برچسب خورده کم هست با چالش‌هایی رو به رو هستیم در این بخش ابتدا به بررسی چالش‌ها و نحوه مقابله مدل‌های دست بندی تصاویر با مجموعه داده محدود می‌پردازیم.

۳-۲-۱ تطبیق بیش از حد

وقتی تعداد داده محدود است احتمالاً اگر با مدل‌های یادگیری عمیق داده را آموزش دهیم مدل بر روی داده آموزشی بایاس می‌شود. برای مقابله با این مشکل راهکارهایی پیشنهاد داده‌اند از جمله:

۱. استفاده از شبکه پیش آموخته : با استفاده از شبکه‌های عمیق پیش آموزش دیده شده روی مجموعه دادگان بزرگ‌تر، می‌توان از اطلاعات استخراج شده توسط این شبکه‌ها بهره‌برداری کرده و آن‌ها را برای دسته‌بندی تصاویر با مجموعه دادگان محدود استفاده کرد. البته باید این را در نظر داشت اگر داده آموزشی که داریم متفاوت از داده آموزشی باشد که شبکه‌های پیش آموخته روی آن آموزش

دیده‌اند نیاز است تا شبکه را fine-tune کنیم که این کار وقتی داده آموزشی کم باشد نمی‌تواند مانع تطبیق بیش از حد شود.

۲. افزایش تعداد داده‌ها: می‌توان با استفاده از روش‌های افزایش داده تعداد داده‌های آموزشی را افزایش داد. این روش‌ها شامل تغییر اندازه تصاویر، چرخاندن، برگرداندن، اعمال افکت‌های نورپردازی و... می‌شوند. البته باید این نکته را در نظر داشت که افزایش داده اطلاعات جدیدی به داده‌های آموزشی اضافه نمی‌کند به همین علت نمی‌تواند به طور کامل با تطبیق بیش از حد مقابله کند.

۳-۲-۲ تأثیر هایپر پارامترها بر دقت مدل

یکی از مواردی در هنگام کار با شبکه‌های مجموعه دادگان محدود متوجه شدیم تأثیر زیاد هایپر پارامترها بر دقت مدل بود. هایپر پارامترهایی مانند نرخ آموزش، تعداد نمونه‌هایی که از هر کدام از مجموعه‌های پشتیبان و کوئری می‌سازیم و تعداد اپک‌هایی که شبکه را با آن آموزش می‌دهیم.

۳-۲-۳ کافی نبودن یک بردار ویژگی

همان‌طور که در قسمت ۳-۲-۳-۲ گفته شد برای اینکه بتوانیم از ویژگی‌های محلی تصاویر نیز استفاده کنیم از فضاهای تعبیه متعدد استفاده می‌کنیم تا چندین خروجی داشته باشیم. تنها یک بردار خروجی نمی‌تواند تمام اطلاعات لازم برای مقایسه را داشته باشد.

۳-۲-۴ استخراج بردار ویژگی با کیفیت

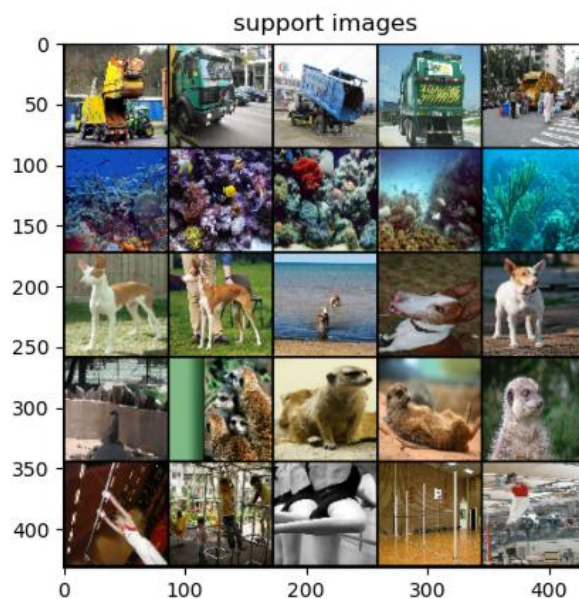
جدا از بحث استفاده از فضای تعبیه متعدد اینکه بردارهای ویژگی با کیفیت داشته باشیم نیز حائز اهمیت است. می‌توانیم بردارهای ویژگی که از شبکه‌های پیش‌آمخته‌ای همچون VGG یا Resnet استخراج می‌کنیم را با کیفیت‌تر بکنیم یکی از راه‌ها استفاده از مکانیزم توجه به خود است مشابه روشی که در قسمت ۳-۲-۴-۲ درباره شبکه SetFeat گفته شد.

۳-۳ پیاده‌سازی مدل پیشنهادی

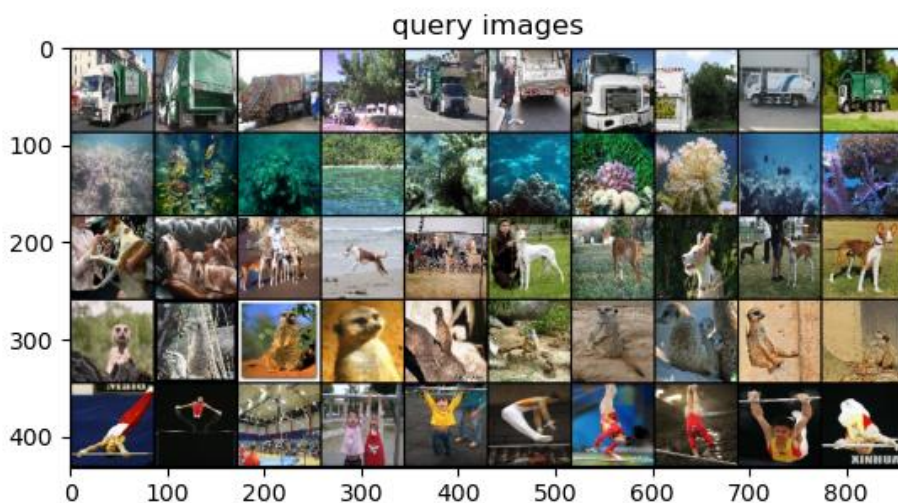
در این قسمت به پیاده‌سازی مدل پیشنهادی می‌پردازیم. در ابتدا مدل Prototypical بر روی دیتاست MiniImagenet پیاده‌سازی می‌کنیم در ادامه با توجه به چالش‌هایی که گفته شد چندین بردار ویژگی از ورودی استخراج می‌کنیم و به هر بردار ویژگی وزن می‌دهیم. برای اینکه بردار ویژگی‌های با کیفیت‌تری داشته باشیم از مکانیزم توجه به خود در هر خروجی استفاده می‌کنیم.

۳-۳-۱ پیاده‌سازی اولیه Prototypical

مدل Prototypical را مشابه بخش ۳-۳-۲ بر روی دیتاست MiniImagenet پیاده‌سازی می‌کنیم. دیتاست MiniImagenet شامل ۶۰۰۰ تصویر به ابعاد 84×84 است که به ۱۰۰ کلاس با ۶۰ نمونه در هر کلاس تقسیم شده‌اند. طبق تقسیم‌بندی که در مقالات استفاده می‌شود ۶۴ کلاس مربوط به آموزش، ۱۶ کلاس مربوط به اعتبارسنجی و ۲۰ کلاس مربوط به داده تست است. در ابتدا تصاویر مجموعه کوئری و پشتیبان را می‌سازیم شکل زیر نمونه‌ای از تصاویر مجموعه پشتیبان و کوئری را بر روی دیتاست MiniImagenet نشان می‌دهد.

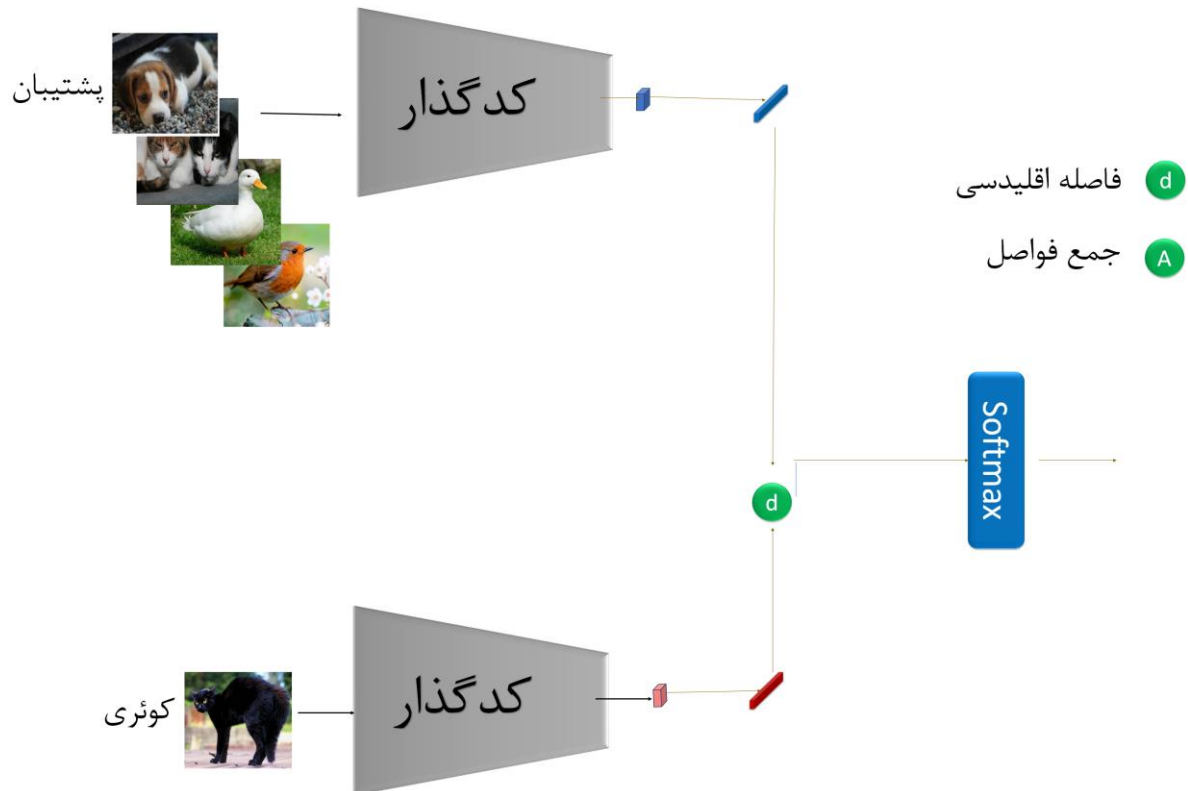


شکل ۳-۱: نمونه‌ای از تصاویر مجموعه پشتیبان



شکل ۳-۲: نمونه‌ای از تصاویر مجموعه کوئری

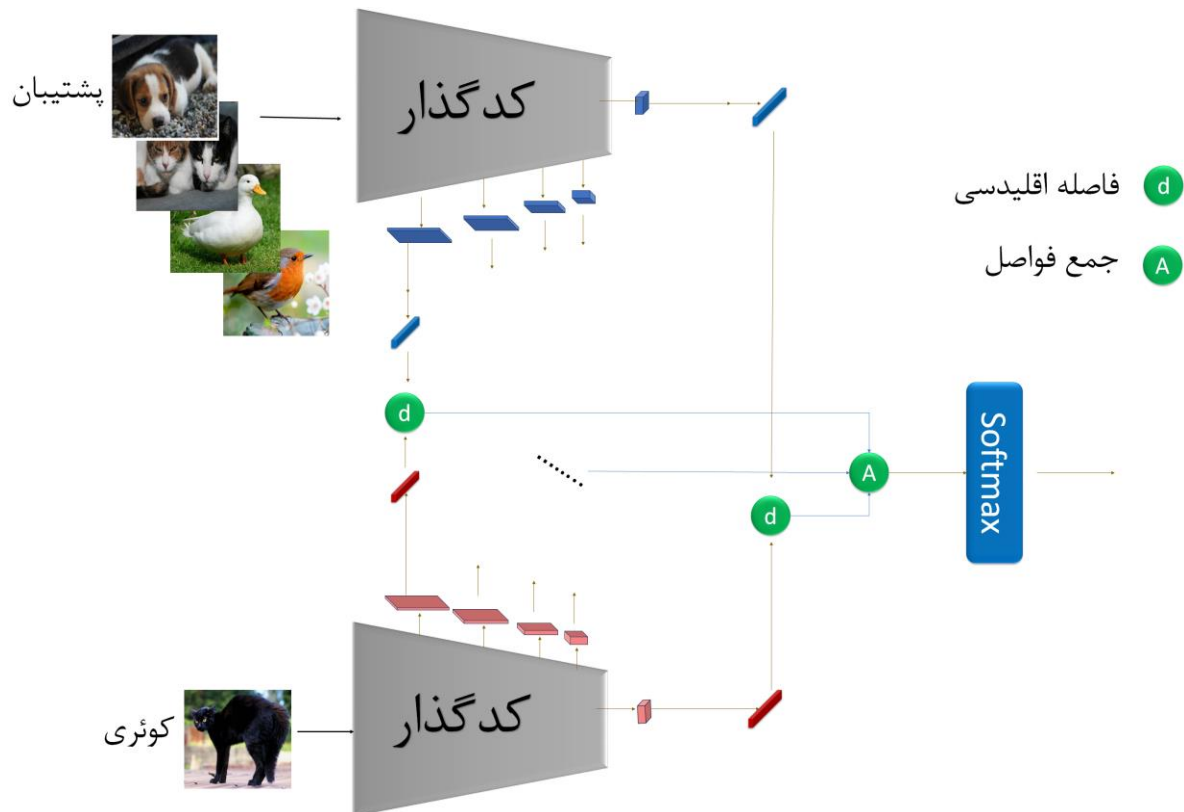
دو تصویر بالا یک instance از مجموعه کوثری و پشتیبان است که ۵ کلاسه هست و در شبکه آموزش داده می شود. همان طور که مشاهده می کنید باید هر تصویر از مجموعه کوثری را به یکی از ۵ کلاس مجموعه پشتیبان نسبت دهیم. در خود مقاله اصلی از شبکه ConvNet-4 که شامل چهار بلوک کانولوشنی هست استفاده کرده است. ما برای پیاده سازی از شبکه پیش آموخته ResNet18 استفاده کردیم. ساختار کلی شبکه در این مرحله به صورت زیر است:



شکل ۳-۳: ساختار اولیه شبکه Prototypical بر روی دیتاست Minilmagenet

۳-۲-۳ استفاده از فضای تعبیه متعدد در مدل

همان طور که در قسمت چالش نیز گفته شد برای اینکه خروجی های متعدد داشته باشیم و از تمام ویژگی های تصویر استفاده کنیم از فضای تعبیه متعدد استفاده کردیم. در واقع stage های اول شامل اطلاعات محلی تصویر هستند و stage های پایانی شامل اطلاعاتی هستند بیشتر سمت طبقه بندی تصاویر هستند. در جاهایی که سایز نصف می شود بردار کانولوشنی را استخراج می کنیم و سپس میانگین سراسری می زنیم تا یک بردار تک بعدی داشته باشیم و این کار را روی هر دو مجموعه تصاویر کوثری و ساپورت انجام می دهیم و با توجه به اینکه از Resnet18 استفاده می کنیم ۵ بردار خروجی داریم که هر بردار خروجی در مجموعه ساپورت فاصله اقلیدسی آن را با بردار متناظر آن در مجموعه کوثری حساب می کنیم و سپس تمامی فاصله ها را با هم جمع می زنیم تصویر کوثری به مجموعه ای تعلق دارد که فاصله آن نسبت به بقیه کمتر شود ساختار کلی به صورت شکل زیر است:



شکل ۳-۴: ساختار فضا تعبیه متعدد در مدل *Prototypical*

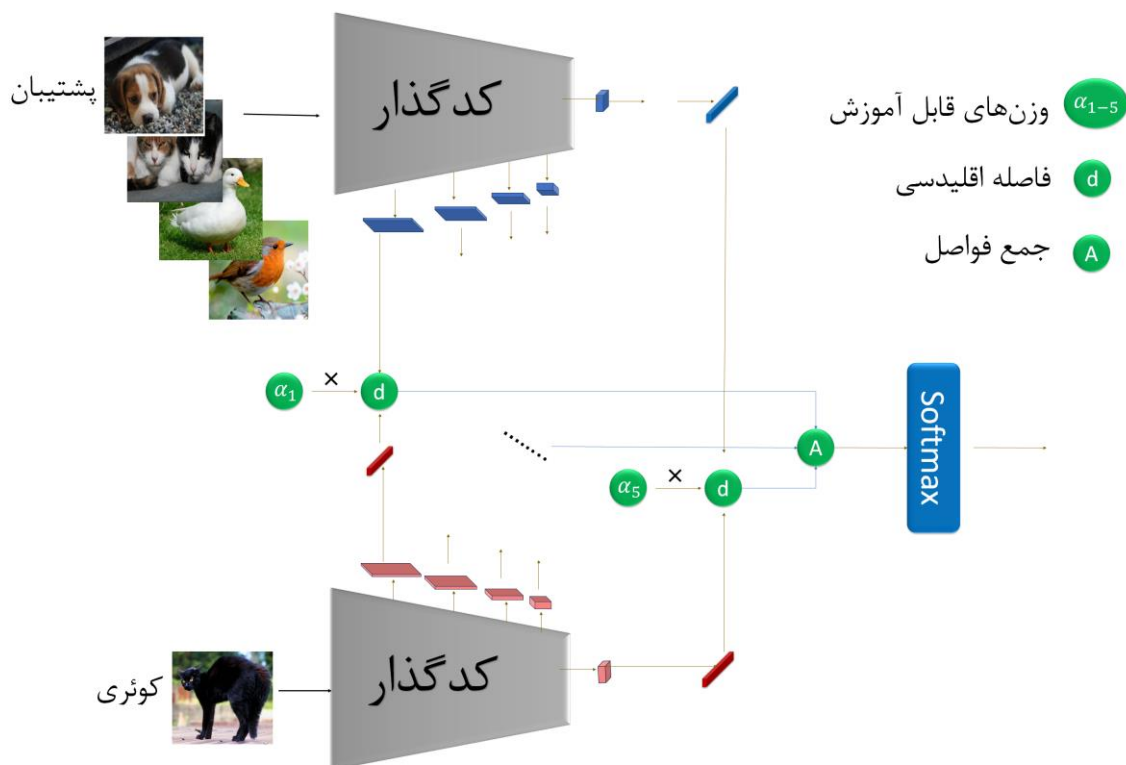
۳-۳-۳ دادن پارامتر وزن قابل یادگیری به هر خروجی

یکی از ایراداتی که به نظرم ساختار شبکه مدل فضای تعبیه متعدد در بخش ۲-۳-۲-۴ داشت اینکه به همه بردارهای خروجی به یک نسبت اهمیت می داد. با اینکه بردارهای خروجی انتهایی با توجه به اینکه ویژگی‌های مهم‌تری نسبت به بردارهای ابتدایی دارند چون بردار خروجی ابتدایی معمولاً ویژگی‌هایی همچون لبه‌های تصویر را تشخیص می‌دهند تصمیم گرفتیم به هر بردار خروجی وزن بدهیم در ابتدا وزن‌ها هاپیرپارامتر در نظر گرفتیم ولی در ادامه تصمیم گرفتیم وزن‌ها نیز پارامتر باشند و در شبکه آموزش داده شوند به همین علت در ابتدا تعدادی وزن اولیه به هر خروجی دادیم و بعد گذاشتیم با آن مقادیر ابتدایی در شبکه آموزش داده شوند. با وزن اولیه‌های متعددی امتحان کردیم تا به مقدار مناسب اولیه مناسب برای هر بردار خروجی رسیدیم که مقادیر اولیه مناسب آن برای 5 way 1 shot و 5 way 5 shot به صورت زیر است:

جدول ۳-۱: وزن های مناسب برای هر از دو مسئله ۱ شات و ۵ شات

	W1	W2	W3	W4	W5
5 way 1 shot	1	1.2	1.4	1.6	1.8
5 way 5 shot	1	1.1	1.2	1.3	1.4

ساختار شبکه نیز مطابق شکل زیر است که پس از استخراج هر بردار خروجی در مجموعه پشتیبان و کوئری به آن و وزن های اولیه می دهیم تا در شبکه آموزش دیده شوند.

شکل ۳-۵: اضافه کردن پارامتر وزن دهی قابل یادگیری در مدل تعبیه متعدد *Prototypical*

۳-۳-۴ اضافه کردن مکانیزم توجه به خود در هر بردار خروجی

همان طور که در قسمت چالش ها نیز گفته شد اینکه بردارهای خروجی با کیفیت داشته باشیم مکانیزم توجه به خود است. مکانیزم توجه به خود یکی از مهم ترین اجزای شبکه های عصبی مبتنی بر ترنسفورمر است. این مکانیزم به مدل امکان می دهد تا از روابط مکانی و وابستگی های موجود در داده های ورودی خود آگاه شود. با استفاده از توجه به خود، مدل می تواند بدون وابستگی به موقعیت مطلق هر عنصر ورودی، ارتباطات مهم را شناسایی کرده و بر آنها تمرکز کند.

ما پس از اینکه بردارهای کانولوشنی را در هر stage استخراج کردیم از یک نگاشت دهنده استفاده می‌کنیم که شامل یک توجه به خود و میانگین سراسری است. اگر بردار کانولوشنی که استخراج می‌کنیم را به صورت زیر در نظر بگیریم:

$$x \in R^{C \times N}$$

که C تعداد کانال‌های و N نیز حاصل ضرب بقیه بعدها هست. برای مکانیزیم توجه مراحل زیر را طی می‌کنیم:

- یک کانولوشن 1×1 را بر روی x اعمال می‌کنیم و بردارهای کانولوشنی h ، g و f را به دست می‌آوریم این کار را برای این انجام می‌دهیم تا تعداد کانال‌ها را از C به C^* کاهش دهیم.

$$f(x) = W_f x, g(x) = W_g x$$

$$h(x) = W_h x$$

$$W_f, W_g, W_h \in R^{C^* \times C}$$

- پس از ضرب $f(x)$ و $g(x)$ softmax را اعمال می‌کنیم.

$$\beta_{j,i} = \frac{\exp(S_{ij})}{\sum_{i=1}^N \exp(S_{ij})}, \text{ where } S_{ij} = f(x_i)^T g(x_j)$$

وزن‌ها که نقشه توجه نامیده می‌شوند و اهمیت پیکسل‌های j را نسبت به پیکسل‌های i تعیین می‌کنند. از آنجایی که این وزن‌ها (β) در تمام ارتفاع و عرض مجموعه ویژگی محاسبه می‌شوند، میدان دید^۱ دیگر به اندازه یک هسته کوچک محدود نمی‌شود.

- خروجی لایه توجه به خود به صورت زیر محاسبه می‌شود:

$$o_j = v\left(\sum_{i=1}^N \beta_{j,i} h(x_i)\right)$$

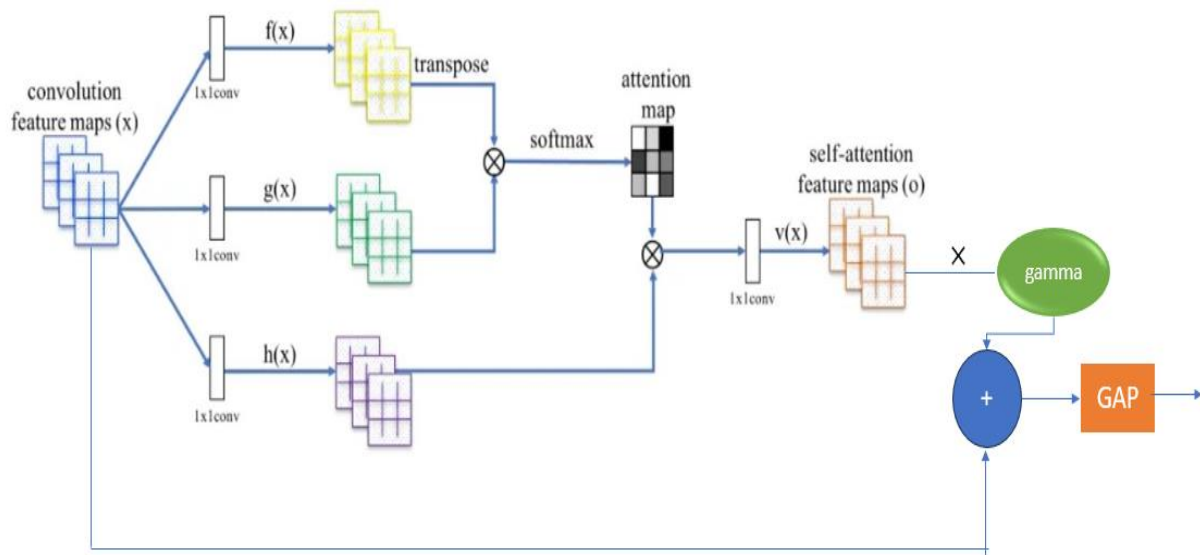
$$v(x) = W_v x_1, W_v \in R^{C \times C^*}$$

^۱ receptive field

در اینجا v خروجی کانولوشن 1×1 هست که تعداد کانال های ورودی یعنی x با تعداد کانال های خروجی برابر باشد. چون یک بلاک Residual نیز داریم که خروجی را با ضریب $gamma$ با ورودی جمع کند مطابق فرمول زیر:

$$y_i = gamma \times o_i + x_i$$

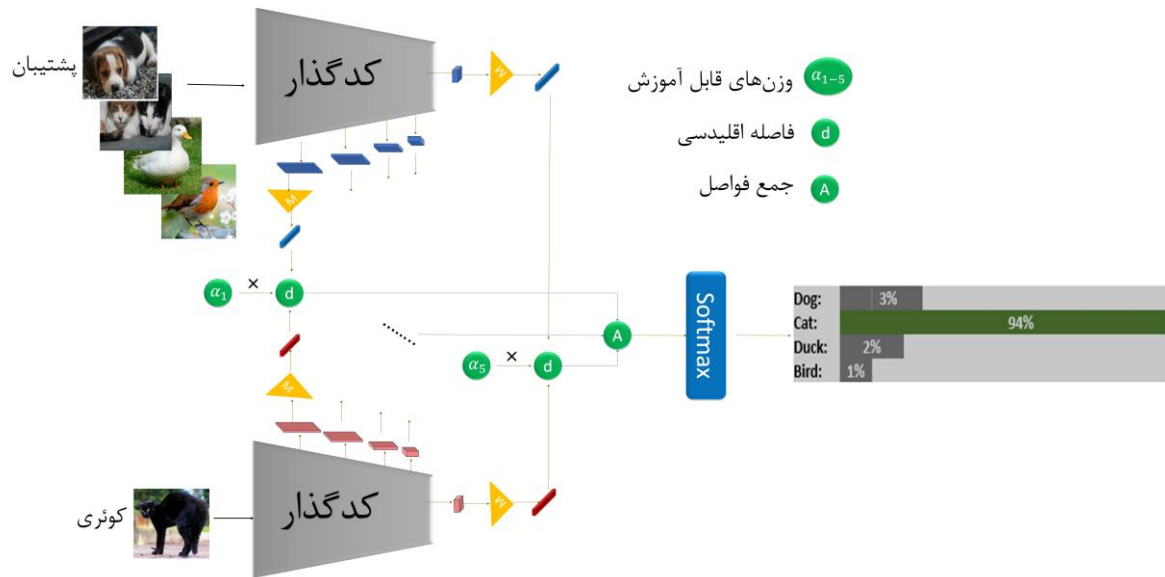
نکته: $gamma$ نیز یک پارامتر قابل یادگیری در شبکه است. در کد نیز $C^*/8$ قرار دادیم. پس از اینکه خروجی نهایی را به دست آوردیم یک میانگین سراسری می زنیم. شکل ساختار نگاشت دهنده ای که استفاده کردیم به صورت زیر است:



شکل ۳-۶: نگاشت دهنده

پس از اینکه فاصله های متناظر بردارهای خروجی را حساب کردیم و با هم جمع کردیم برای اینکه نرمالیزه کنیم از $softmax$ استفاده می کنیم تا احتمال تعلق هر تصویر کوئری به تصاویر پشتیبان به دست آید. تابع ضرری که در این شبکه Cross Entropy مشابه شبکه اصلی Prototypical است.

شکل کلی مدلی که پیشنهاد کردیم نیز به صورت زیر است:



شکل ۳-۷: شکل مدل نهایی پیشنهادی

فصل ۴

نتایج آزمایش‌ها

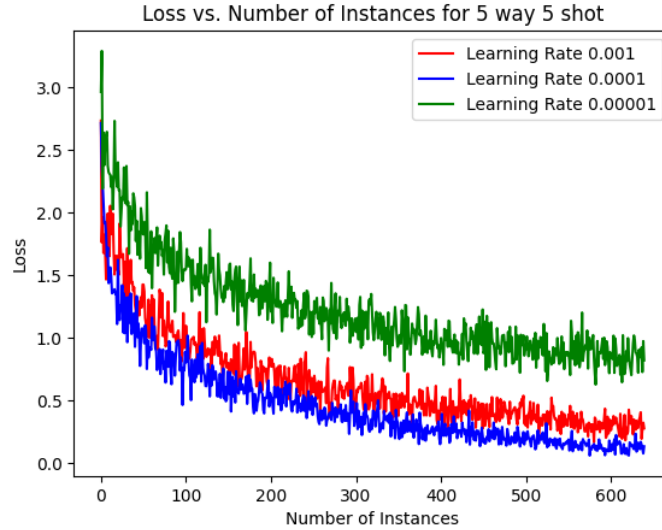
۴-۱ مقدمه

در این فصل نتایج مراحل‌ی که در فصل ۳ گفته شد ارائه شده است. در واقع با کنار هم قرار دادن نتایج هر کدام از بخش‌ها در کنار هم نشان می‌دهیم هر مرحله باعث بهبود دقت شده است. همچنین در آخر مدلی که پیشنهاد دادیم را با سایر مدل‌ها در مقالات دیگر مقایسه می‌کنیم.

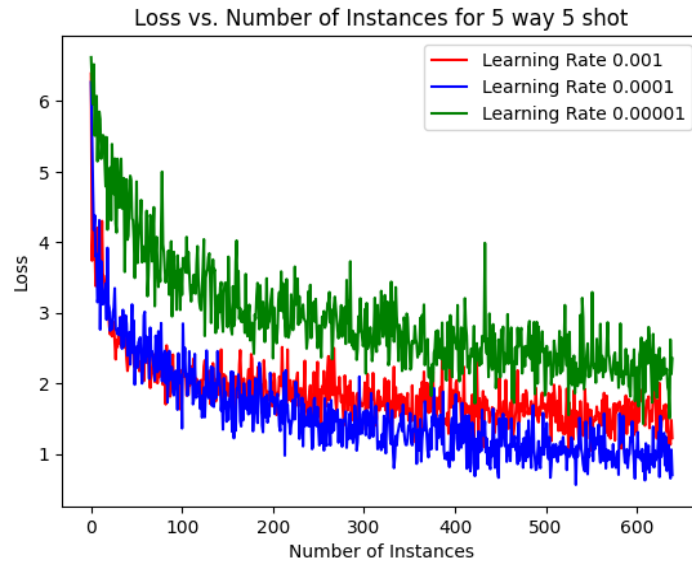
۴-۲ نتایج مدل پیشنهادی در هر مرحله

۴-۲-۱ نتایج مدل اولیه Prototypical

در بخش ۳-۳-۱ این روش معرفی شد. در این بخش نتایج این مدل با سه نرخ یادگیری ۰.۰۰۰۱، ۰.۰۰۰۰۱ و ۰.۰۰۰۰۰۱ بر روی تسک ۵ کلاسه ۵ شات و ۵ کلاسه تک شات آورده شده است. در مقایسه از معیار دقت استفاده شده است. این مدل با استفاده از ۶۴۰ مجموعه، که هر مجموعه متشکل از تصاویر پشتیبان و کوئری است آموزش دیده است. در شکل‌های زیر نمودار loss بر حسب تعداد instance ها برای هر دو تسک نمایش داده شده است:



شکل ۴-۱: نمودار ضرر بر حسب تعداد *instance* ها برای تسک ۵ کلاسه ۵ شات در مدل اولیه *Prototypical*



شکل ۴-۲: نمودار ضرر بر حسب تعداد *instance* ها برای تسک ۵ کلاسه تک شات در مدل اولیه *Prototypical*

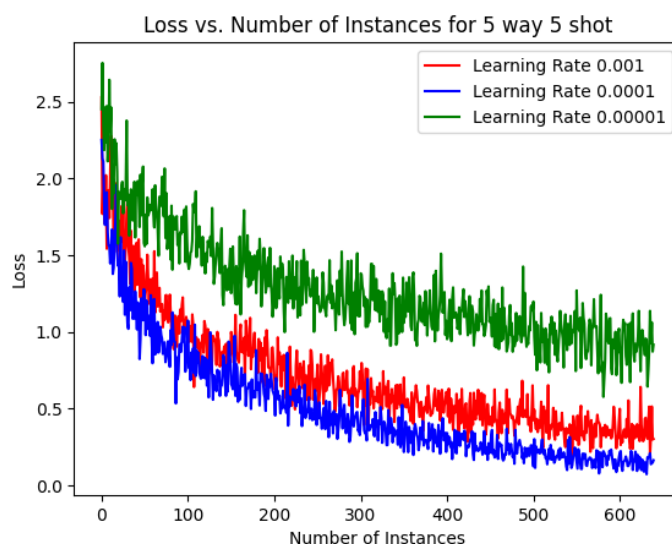
نتایج دقت نیز به صورت زیر است:

جدول ۴-۲: مقایسه دقت نرخ یادگیری‌های مختلف بر روی مدل *baseline prototypical*

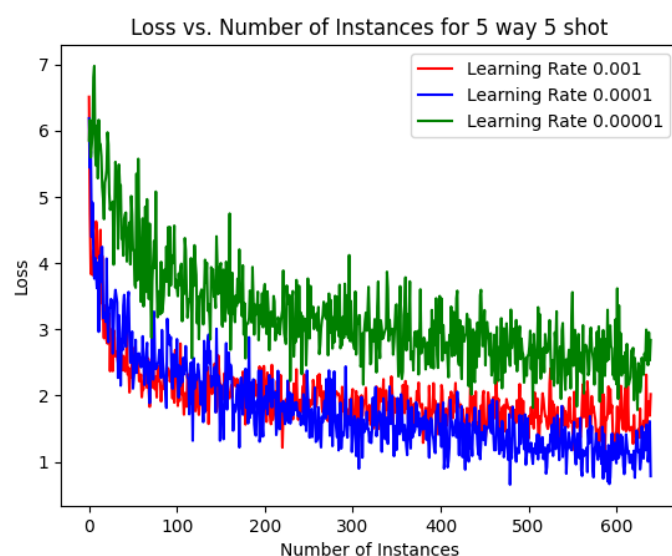
Learning rate on baseline	5 way 1 shot	5 way 5 shot
lr=0.001	47.039	68.88
lr=0.0001	62.67	82.06
lr=0.00001	54.17	82.64

۴-۲-۲ نتایج استفاده از فضای تعبیه متعدد

در بخش ۳-۳-۲ این روش گفته شد. در این بخش در به هر کدام از بردارهای خروجی وزن ۱ دادیم و با نرخ یادگیری‌های متفاوت شبکه را آموزش داده‌ایم. نتایج به صورت زیر است:



شکل ۴-۳: نمودار ضرر بر حسب تعداد *instance* ها برای تسک ۵ کلاسه ۵ شات در استفاده از فضای تعبیه متعدد



شکل ۴-۴: نمودار ضرر بر حسب تعداد *instance* ها برای تسک ۵ کلاسه تک شات در استفاده از فضای تعبیه متعدد

نتایج دقت نیز به ازای نرخ یادگیری متفاوت به صورت زیر است:

جدول ۴-۲: مقایسه دقت نرخ یادگیری‌های مختلف در استفاده از فضای تعبیه متعدد

Learning rate on multiscale without weight learnable	5 way 1 shot	5 way 5 shot
lr=0.001	52.834	71.7
lr=0.0001	63.64	۸۳.۰۲
lr=0.00001	57.314	82.86

همان‌طور که مشاهده می‌شود استفاده از فضا تعبیه متعدد باعث افزایش دقت در هر دو دقت و هم در تمامی نرخ‌های یادگیری متفاوت شده است.

با توجه به اینکه با نرخ یادگیری ۰.۰۰۰۱ نتایج بهتری به دست آوردیم در بقیه مراحل فقط از همین نرخ یادگیری استفاده کردیم.

۴-۲-۳ نتایج دادن پارامتر وزن قابل یادگیری به هر خروجی

در بخش ۳-۳-۳ این روش توضیح داده شد. در ابتدا ۵ وزن اولیه به هر خروجی نسبت می‌دهیم و این وزن‌ها در شبکه آموزش می‌بینند. ما ۲۰ وزن اولیه مختلف را امتحان کردیم از وزن‌های ۱.۳، ۱.۲، ۱.۱، ۱.۰ و ۱.۴ شروع کردیم و فاصله وزن‌های هر خروجی را ۰.۱ افزایش دادیم به‌طوری که در مرحله آخر وزن‌های ما برابر با ۸.۶ و ۱.۲، ۲.۹، ۴.۸، ۶.۷ بود بهترین دقتی که توانستیم برای دو تسک بگیریم با وزن‌های زیر است:

جدول ۴-۳: بهترین دقت در دادن پارامتر وزن قابل یادگیری به هر خروجی

	Weight1	Weight2	Weight3	Weight4	Weight5	After train Weight2	After train Weight2	After train Weight3	After train Weight4	After train Weight5	accuracy
5 way 5 shot	1	1.1	1.2	1.3	1.4	0.96	1.06	1.16	1.25	1.38	83.5
5 way 1 shot	1	1.1	1.2	1.3	1.4	0.94	1.05	1.14	1.24	1.37	65.14

هم‌چنین حالتی که وزن‌های اولیه خروجی‌های اول بیشتر باشد را نیز امتحان کردیم ولی بهبودی در نتایج ایجاد نشد. نتایج آن به صورت زیر است:

جدول ۴-۴: مقایسه دقت در دو حالت در تسک ۵ کلاسه ۵ شات

Weight1	Weight2	Weight3	Weight4	Weight5	accuracy
1	0.9	0.8	0.7	0.6	80.94
1	0.8	0.6	0.4	0.2	80.84
1	1	1	1	1	82.62
1	1.1	1.2	1.3	1.4	83.5
1	1.2	1.4	1.6	1.8	83.24

جدول ۴-۵: مقایسه دقت در دو حالت در تسک ۵ کلاسه تک شات

Weight1	Weight2	Weight3	Weight4	Weight5	accuracy
1	0.9	0.8	0.7	0.6	61.78
1	0.8	0.6	0.4	0.2	63.26
1	1	1	1	1	63.42
1	1.1	1.2	1.3	1.4	65.14
1	1.2	1.4	1.6	1.8	64.56

۴-۲-۴ نتایج اضافه کردن مکانیزم توجه به خود

در بخش ۳-۳-۴ این روش توضیح داده شد. در واقع به مرحله قبل در هر خروجی آن، یک توجه به خود اضافه می‌کنیم. هم‌چنین پارامتر γ نیز مقدار اولیه آن را برابر با ۰ می‌گذاریم نتایج به صورت زیر است:

جدول ۴-۶: دقت مدل پس از اضافه کردن توجه به خود

	accuracy
5 way 1 shot	64.46
5 way 5 shot	84.42

همان طور که مشاهده می‌کنید در تسک ۵ شات بهبود دقت خوبی داشتیم ولی در تسک تک شات بهبود دقت نداشتیم. (البته باید چند مدل را اجرا گرفت و میانگین دقت را گرفت شاید در تک شات در میانگین بهبود داشته باشیم.)

۳-۴ نتایج نهایی و مقایسه مدل پیشنهادی با سایر مدل‌ها

در این قسمت در ابتدا مدل پیشنهادی با سایر مدل‌ها در مقالات مقایسه می‌کنیم و سپس دقت در تمامی مراحل که در قسمت پیش گفته شد را در یک جدول نمایش می‌دهیم.

جدول ۴-۷: مقایسه مدل پیشنهادی با سایر مدل‌ها

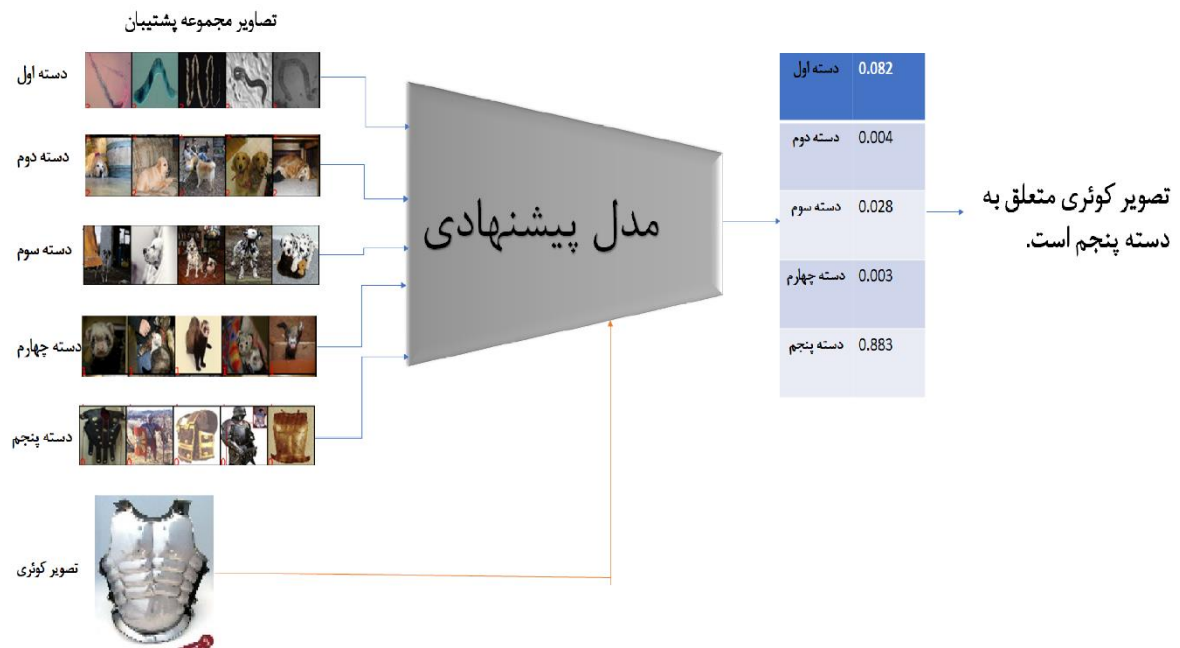
Method	1-shot 5 way	5-shot 5 way
AdaResNet	56.88	71.94
TADAM	58.50	76.70
MetaOptNet	62.64	78.63
Neg-Margin	63.85	81.57
MixtFSL	63.98	82.04
Meta-Baseline	63.17	79.26
Distill	64.82	82.14
ProtoNet	62.39	80.53
Set Feat	68.32	82.71
Our model	64.46	84.42

جدول ۴-۸: دقت مدل مرحله به مرحله

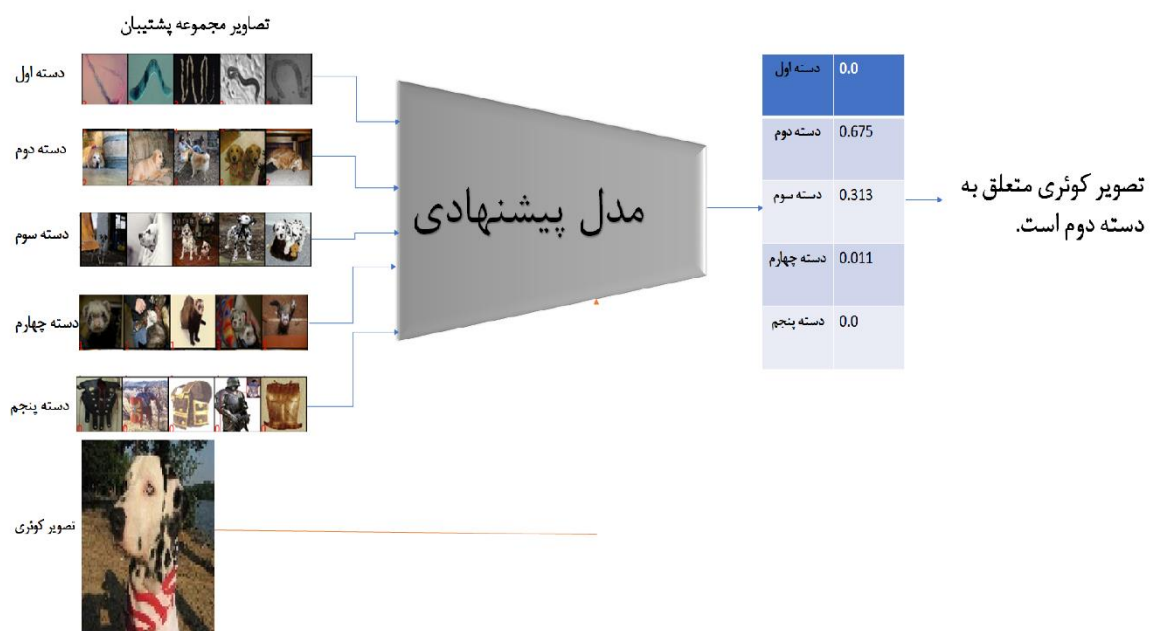
	1-shot 5 way	5-shot 5 way
baseline	62.67	82.06
Baseline + multiscale	63.64	83.02
Baseline + weighted multiscale	65.14	83.5
Baseline + weighted multiscale + attention	64.46	84.42

در ادامه چند نمونه از نتایج مدل را بر روی تصاویر کوئری مشاهده می‌کنید.

تسک ۵ کلاس ۵ شات:

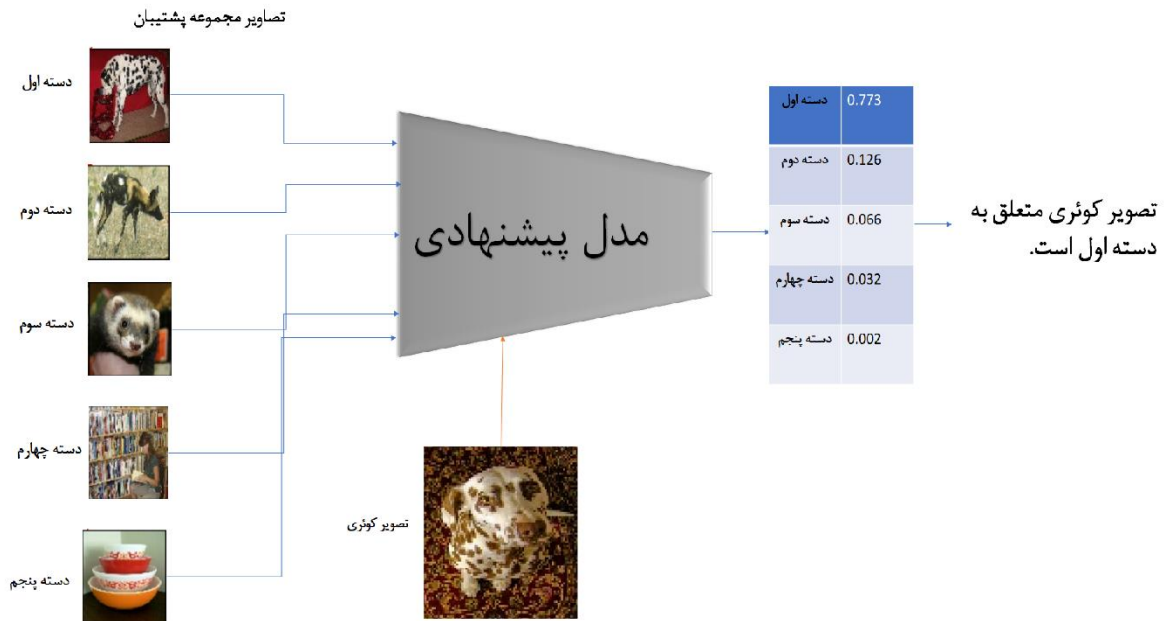


شکل ۴-۵: نمونه‌ای پیش‌بینی درست مدل در تسک ۵ کلاس ۵ شات

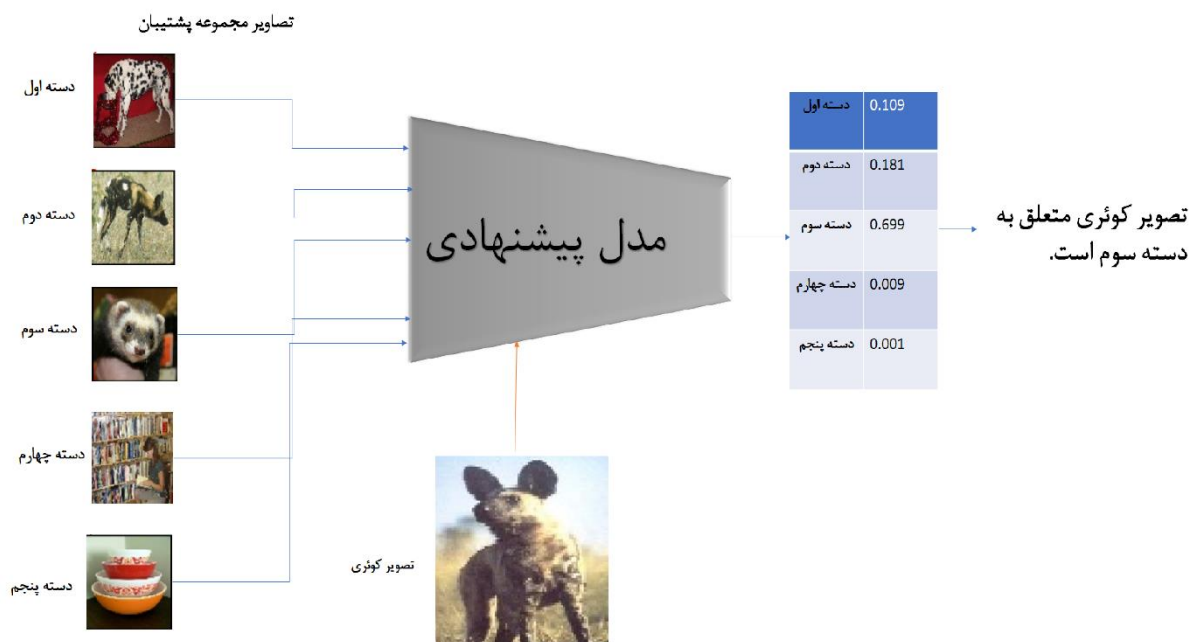


شکل ۴-۶: نمونه‌ای پیش‌بینی نادرست مدل در تسک ۵ کلاس ۵ شات

تسک ۵ کلاس ۵ تک شات:



شکل ۴-۷: نمونه‌ای پیش‌بینی درست مدل در تسک ۵ کلاسه تک شات



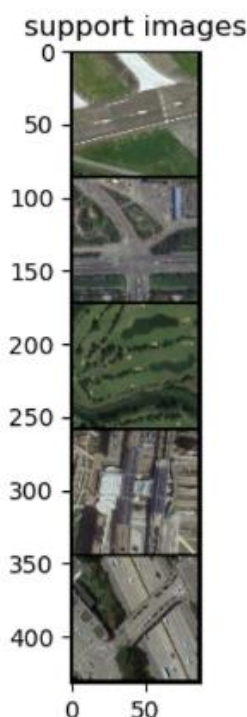
شکل ۴-۸: نمونه‌ای پیش‌بینی نادرست مدل در تسک ۵ کلاسه تک شات

۴-۴ تست مدل بر روی تصاویر هوایی

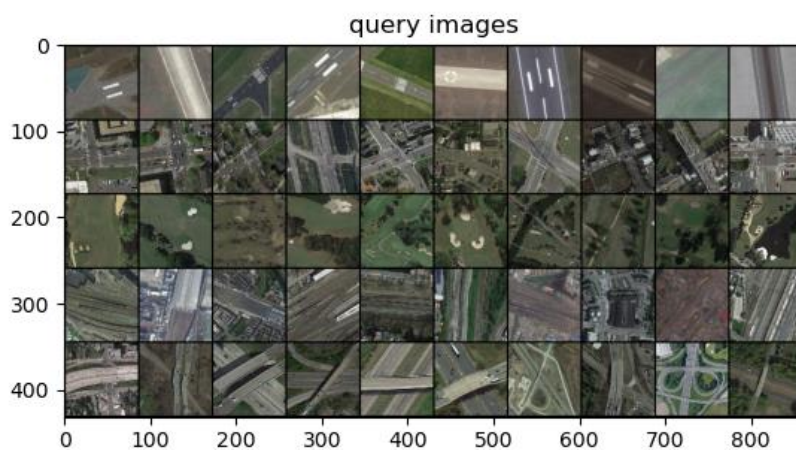
برای اینکه اثبات کنیم که مدلی که ارائه دادیم تعمیم‌پذیری خوبی دارد و به جز دیتاست MiniImagenet که بر روی آن آموزش داده شده است، روی سایر دیتاست‌ها نیز عملکرد خوبی دارد از دیتاست NWPU-RESISC45 استفاده کردیم. دیتاست NWPU-RESISC45 یک مجموعه داده بسیار مهم و گسترده در زمینه شناسایی تصاویر زمین‌شناسی است. این دیتاست شامل تصاویر هوایی از

مناطق مختلف جهان است و برای دسته‌بندی و طبقه‌بندی تصاویر به ۴۵ کلاس مختلف طراحی شده است. هر کلاس در این دیتاست نماینده یک نوع محیط زمینی و منظره است.

با استفاده از این دیتاست، می‌توان به صورت خودکار تصاویر را در دسته‌های مختلفی مانند هواپیما، زمین خالی، ساحل، پل، ساختمان‌ها، جنگل و بسیاری دیگر دسته‌بندی کرد. این دیتاست به عنوان یک منبع قدرتمند برای آموزش و ارزیابی الگوریتم‌های یادگیری ماشین و شبکه‌های عصبی در زمینه شناسایی منظره و تصاویر زمینی استفاده می‌شود. چون هدف مدل‌های یادگیری با مجموعه داده‌گان محدود این بود که شباهت‌ها و تفاوت‌های بین تصاویر را بتواند تشخیص دهد سراغ دیتاستی رفتیم که با MiniImagenet متفاوت باشد. درواقع مدلی که بر روی MiniImagenet آموزش داده شد بود را ذخیره کردیم و بر روی دیتاست NWPU-RESISC45 فقط تست کردیم. یعنی مدل هیچ کدام از کلاس‌های این دیتاست را در حین آموزش ندیده بود. نمونه‌ای از تصاویر مجموعه پشتیبان و کوئری را در شکل زیر مشاهده می‌کنید.



شکل ۴-۹: تصاویر مجموعه پشتیبان در تسک ۵ کلاسه تک شات



شکل ۴-۱۰: تصاویر مجموعه کوئری در تسک ۵ کلاسه تک شات

در ادامه نیز نتایج اجرا گرفتن مدل بر روی این دیتاست را مشاهده می کنید:

جدول ۴-۹: دقت مدل پیشنهادی آموزش داده شده بر روی MiniImagenet روی تصاویر هوایی

	accuracy
5 way 1 shot	55.49
5 way 5 shot	78.44

فصل ۵

جمع بندی

در این فصل در ابتدا نتیجه این پایان نامه به صورت خلاصه در قسمت ۵-۱ گفته می شود. سپس در قسمت ۵-۲، ایده هایی مطرح می گردد که می تواند به عنوان ادامه پژوهش بر روی مبحث بیان شده در این پایان نامه، در نظر گرفته شود.

۵-۱ نتیجه گیری

در این پایان نامه مدل های مختلفی برای طبقه بندی تصاویر با مجموعه داده محدود معرفی شد. سپس برای روش پیشنهادی مدل Prototypical را به عنوان مدل پایه انتخاب کردیم تا با یکسری از اقدامات بتوانیم دقت را افزایش دهیم. در ابتدا با تغییر backbone و نرخ یادگیری توانستیم به بهبود خوبی برسیم. در ادامه چون می خواستیم از ویژگی های محلی تصویر نیز در مدل استفاده کنیم از فضای تعبیه متعدد استفاده کردیم که در این مرحله نیز افزایش دقت داشتیم. چون بردار ویژگی هایی که مربوط به خروجی های آخر بودند به نظرمان با توجه به اینکه بیشتر سمت طبقه بندی تصویر بودند مهم تر بودند تصمیم گرفتیم وزن بیشتری به آنها بدهیم و در ادامه وزن را نیز به عنوان یک پارامتر در نظر گرفتیم که در شبکه آموزش ببیند. در این مرحله نیز کمی بهبود دقت داشتیم. هم چنین برای اینکه مطمئن شویم فرض اینکه بردار ویژگی های انتهایی مهم تر هستند به صورت امتحانی چندبار وزن بردارهای ابتدایی را بیشتر دادیم که دیدیم دقت کمی کاهش پیدا کرد. چون خود کیفیت بردارهای ویژگی نیز در مدل بسیار مهم بودند تصمیم گرفتیم از مکانیزم توجه به خود پس از استخراج بردارها استفاده کنیم که در این مرحله رو تسک ۵ شات افزایش دقت داشتیم. در انتها برای اینکه اثبات کنیم مدل تعمیم پذیری خوبی دارد مدلی که بر روی دیتاست MiniImagenet آموزش دیده بود را بر روی دیتاست تصاویر هوایی تست کردیم و به دقت خوبی رسیدیم. علاوه بر نتایج خوبی که به دست آوردیم، مواردی هنوز برای پژوهش های بیشتر وجود دارد که در ادامه به آنها اشاره می شود.

۵-۲ کارهای آینده

۵-۲-۱ استفاده از یادگیری دو مرحله ای

یکی از ایده های بسیار جالب استفاده از یادگیری دو مرحله ای هست که در قسمت ۲-۳-۲-۴ توضیح داده شد. اینکه در ابتدا تعدادی از تصاویر مجموعه پشتیبان که فاصله زیادی با تصویر کوئری دارند را حذف می کنیم و در مرحله بعد با نگاهی دقیق تر دسته متعلق به تصویر کوئری را انتخاب می کنیم می تواند خیلی مفید باشد. با توجه به اینکه در چند نمونه از پیش بینی های مدل در فصل چهار دیدیم که پیش بینی های نادرست مدل که دسته متعلق به تصویر کوئری را انتخاب می کرد خیلی بی مرتبط به تصویر کوئری نبود اگر از یادگیری دو مرحله ای استفاده می شد شانس اینکه مدل در این حالات درست پیش بینی کند افزایش پیدا می کرد.

۵-۲-۲ خوشه بندی^۱ کردن داده ها

در ۵ شات می توانیم از این ایده استفاده کنیم. در همان مدل Prototypical که میانگین هر دسته مجموعه پشتیبان را می گرفتیم می توانستیم همان دسته هایی که متعلق به یک کلاس هستند را خوشه بندی کنیم و از هر خوشه یک نماینده استخراج کنیم. این ایده برای تک شات قابل استفاده نیست ولی برای ۵ شات می تواند ایده خوبی باشد.

۵-۲-۳ ترکیب تابع ضرر Cross Entropy با سایر توابع ضرر

یکی از مواردی که بر روی دقت مدل بسیار تاثیرگذار است تابع ضرر است. در مدلی که پیشنهاد دادیم از تابع ضرر Cross Entropy استفاده کردیم ولی این تابع ضرر را می توان با تعدادی از توابع ضرر همچون Triplet و Center و... ترکیب کرد.

^۱ clustering

- [1] A. Parnami and M. Lee, "Learning from few examples: A summary of approaches to few-shot learning," *arXiv preprint arXiv:2203.04291*, 2022.
- [2] Y. Wang, W.-L. Chao, K. Q. Weinberger, and L. Van Der Maaten, "Simpleshot: Revisiting nearest-neighbor classification for few-shot learning," *arXiv preprint arXiv:1911.04623*, 2019.
- [3] 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 *Computer Vision—ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIV 16*, 2020: Springer, pp. 266-282.
- [4] I. Ziko, J. Dolz, E. Granger, and I. B. Ayed, "Laplacian regularized few-shot learning," in *International conference on machine learning*, 2020: PMLR, pp. 11660-11670.
- [5] N. Mishra, M. Rohaninejad, X. Chen, and P. Abbeel, "A simple neural attentive meta-learner," *arXiv preprint arXiv:1707.03141*, 2017.
- [6] N. Dvornik, C. Schmid, and J. Mairal, "Diversity with cooperation: Ensemble methods for few-shot classification," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 3723-3731.
- [7] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *International conference on machine learning*, 2017: PMLR, pp. 1126-1135.
- [8] B. Kulis, "Metric learning: A survey," *Foundations and Trends® in Machine Learning*, vol. 5, no. 4, pp. 287-364, 2013.
- [9] G. Koch, R. Zemel, and R. Salakhutdinov, "Siamese neural networks for one-shot image recognition," in *ICML deep learning workshop*, 2015, vol. 2, no. 1: Lille.
- [10] O. Vinyals, C. Blundell, T. Lillicrap, and D. Wierstra, "Matching networks for one shot learning," *Advances in neural information processing systems*, vol. 29, 2016.
- [11] J. Snell, K. Swersky, and R. Zemel, "Prototypical networks for few-shot learning," *Advances in neural information processing systems*, vol. 30, 2017.
- [12] F. Sung, Y. Yang, L. Zhang, T. Xiang, P. H. Torr, and T. M. Hospedales, "Learning to compare: Relation network for few-shot learning," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 1199-1208.
- [13] B. Oreshkin, P. Rodríguez López, and A. Lacoste, "Tadam: Task dependent adaptive metric for improved few-shot learning," *Advances in neural information processing systems*, vol. 31, 2018.

- [14] A. Afrasiyabi, H. Larochelle, J.-F. Lalonde, and C. Gagné, "Matching feature sets for few-shot image classification," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 9014-9024.
- [15] P. Li, G. Zhao, and X. Xu, "Coarse-to-fine few-shot classification with deep metric learning," *Information Sciences*, vol. 610, pp. 592-604, 2022.
- [16] F. Gao, L. Cai, Z. Yang, S. Song, and C. Wu, "Multi-distance metric network for few-shot learning," *International Journal of Machine Learning and Cybernetics*, vol. 13, no. 9, pp. 2495-2506, 2022.
- [17] Y. Wen, K. Zhang, Z. Li, and Y. Qiao, "A discriminative feature learning approach for deep face recognition," in *Computer Vision—ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part VII 14*, 2016: Springer, pp. 499-515.

Abstract

Image classification is one of the important and key issues in the field of artificial intelligence. In recent years, most methods in this field have utilized deep learning models to solve classification problems. Achieving good results in image classification with the help of deep learning has a direct relationship with the number of training images. However, obtaining a large training dataset is not feasible in many domains. Additionally, collecting an appropriate dataset can be time-consuming and costly depending on the nature of the problem. In this thesis, we have presented a model for few shot image classification with a, based on the Prototypical model, and made efforts to improve it. The techniques we employed to improve the model's accuracy include changing the backbone, using multiple embedding spaces, assigning weight parameters to each output vector, and utilizing self-attention mechanisms. Finally, we have demonstrated the improvements in accuracy compared to the previous stages and evaluated the model's performance on the MiniImagenet training dataset in comparison with other existing models in the papers, in two tasks: 5-way 5-shot and 5-way 1-shot. In the 5-way 5-shot task, we achieved an accuracy of 84.42%, and in the 5-way 1-shot task, we achieved an accuracy of 64.46%. Furthermore, to demonstrate the model's generalizability, we tested the trained model with the MiniImagenet training data on another unseen dataset without training on it, and achieved good accuracy.

Keywords: Deep learning, Image classification, Few-shot learning, self-attention, Multiple embedding spaces.



Iran University of Science and Technology
School of Computer Engineering

Using Deep Learning for Few-Shot Image Classification

Bachelor of Science Thesis in Computer Engineering

By:
Fatemeh Askari

Supervisor:
Dr. Mohammad Reza Mohammadi

February 2024