



دانشگاه صنعتی امیرکبیر  
(پلی تکنیک تهران)  
دانشکده ...

پایان نامه کارشناسی ارشد  
گرایش ...

## عنوان پایان نامه - دستورالعمل و راهنمای نگارش پایان نامه

نگارش

نام و نام خانوادگی کامل نویسنده

استاد راهنما

نام کامل استاد راهنما

استاد مشاور

نام کامل استاد مشاور

ماه و سال

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

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

در این صفحه فرم دفاع یا تأیید و تصویب پایان نامه موسوم به فرم کمیته دفاع - موجود در پرونده آموزشی - را قرار دهید.

## نکات مهم:

- نگارش پایان نامه/رساله باید به **زبان فارسی** و بر اساس آخرین نسخه دستورالعمل و راهنمای تدوین پایان نامه های دانشگاه صنعتی امیرکبیر باشد.(دستورالعمل و راهنمای حاضر)
- رنگ جلد پایان نامه/رساله چاپی کارشناسی، کارشناسی ارشد و دکترا باید به ترتیب مشکی، طوسی و سفید رنگ باشد.
- چاپ و صحافی پایان نامه/رساله بصورت **پشت و رو(دورو)** بلامانع است و انجام آن توصیه می شود.



دانشگاه صنعتی امیرکبیر  
(پلی تکنیک تهران)

به نام خدا

## تعهدنامه اصالت اثر

تاریخ: ماه و سال

اینجانب نام و نام خانوادگی کامل نویسنده متعهد می‌شوم که مطالب مندرج در این پایان‌نامه حاصل کار پژوهشی اینجانب تحت نظارت و راهنمایی اساتید دانشگاه صنعتی امیرکبیر بوده و به دستاوردهای دیگران که در این پژوهش از آنها استفاده شده است مطابق مقررات و روال متعارف ارجاع و در فهرست منابع و مآخذ ذکر گردیده است. این پایان‌نامه قبلاً برای احراز هیچ مدرک هم‌سطح یا بالاتر ارائه نگردیده است.

در صورت اثبات تخلف در هر زمان، مدرک تحصیلی صادر شده توسط دانشگاه از درجه اعتبار ساقط بوده و دانشگاه حق پیگیری قانونی خواهد داشت.

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

نام و نام خانوادگی کامل نویسنده

امضا

نویسنده پایان نامه، در صورت تمایل میتواند برای پاسخگویی پایان نامه خود را به شخص یا اشخاص و یا ارگان خاصی تقدیم نماید.

## پاس کزاری

نویسنده پایان نامه می تواند مراتب امتنان خود را نسبت به استاد راهنما و استاد مشاور و یا دیگر افرادی که طی انجام پایان نامه به نحوی او را یاری و یا با او همکاری نموده اند ابراز دارد.

نام و نام خانوادگی کامل نویسنده

ماه و سال

## چکیده

در این قسمت چکیده پایان نامه نوشته می‌شود. چکیده باید جامع و بیان‌کننده خلاصه‌ای از اقدامات انجام‌شده باشد. در چکیده باید از ارجاع به مرجع و ذکر روابط ریاضی، بیان تاریخچه و تعریف مسئله خودداری شود.

## واژه‌های کلیدی:

کلیدواژه اول، ...، کلیدواژه پنجم (نوشتن سه تا پنج واژه کلیدی ضروری است)

# فهرست مطالب

عنوان

صفحه

۱	راهنمای استفاده از الگوی لاتک دانشگاه صنعتی امیرکبیر (پلی تکنیک تهران) . . . .	۱
۱-۱	مقدمه . . . . .	۲
۲-۱	این همه فایل؟! . . . . .	۲
۳-۱	از کجا شروع کنم؟ . . . . .	۳
۴-۱	مطالب پایان نامه را چگونه بنویسم؟ . . . . .	۴
۱-۴-۱	نوشتن فصل ها . . . . .	۴
۲-۴-۱	مراجع . . . . .	۵
۳-۴-۱	واژه نامه فارسی به انگلیسی و برعکس . . . . .	۵
۵-۱	اگر سوالی داشتم، از کی بپرسم؟ . . . . .	۵
۲	مرور کارهای پیشین . . . . .	۶
۱-۲	مقدمه . . . . .	۷
۲-۲	یادگیری پیوسته . . . . .	۷
۱-۲-۲	رویکرد مبتنی بر تنظیم . . . . .	۹
۲-۲-۲	رویکرد مبتنی بر بازپخش . . . . .	۹
۳-۲-۲	رویکرد مبتنی بر بهینه سازی . . . . .	۱۰
۴-۲-۲	رویکرد مبتنی بر معماری . . . . .	۱۰
۵-۲-۲	رویکرد ترکیب رویکردها و سناریوها . . . . .	۱۰
۶-۲-۲	کاربردها . . . . .	۱۰
۳-۲	یادگیری پیوسته در بینایی کامپیوتر . . . . .	۱۱
۱-۳-۲	دسته بندی تصویر . . . . .	۱۱
۲-۳-۲	تشخیص عمل . . . . .	۱۳
۴-۲	مدل های بینایی-زبان . . . . .	۱۵
۱-۴-۲	پیش آموزش مدل های بینایی-زبان . . . . .	۱۶
۲-۴-۲	یادگیری انتقالی مدل های بینایی-زبان . . . . .	۱۶
۳-۴-۲	تقطیر دانش . . . . .	۱۸



۱۹	۵-۲ یادگیری پیوسته در مدل‌های بینایی-زبان
۱۹	۱-۵-۲ روش‌های مبتنی بر حافظه
۲۰	۲-۵-۲ روش‌های مبتنی بر تنظیم
۲۰	۳-۵-۲ روش‌های مبتنی بر تقطیر دانش
۲۰	۴-۵-۲ روش‌های مبتنی بر معماری
۲۵	۶-۲ جمع‌بندی
۲۷	۳ روش پیشنهادی
۲۸	۱-۳ مقدمه
۲۹	۲-۳ روش ProActionCLIP
۲۹	۳-۳ مدل Open-VCLIP
۳۰	۱-۳-۳ تبدیل CLIP مبتنی بر تصویر به CLIP مبتنی بر ویدیو
۳۱	۲-۳-۳ منظم سازی مبتنی بر درونیابی وزن‌ها
۳۲	۳-۳-۳ میانگین‌گیری تصادفی وزن‌ها
۳۳	۴-۳ مدل L2P
۳۴	۱-۴-۳ انتخاب پرامپت
۳۵	۲-۴-۳ یادگیری پرامپت
۳۶	۵-۳ مرحله‌ی آموزش ProActionCLIP
۳۶	۱-۵-۳ کدگذار ویدیو
۳۸	۲-۵-۳ یادگیری پرامپت
۳۹	۶-۳ مرحله‌ی آزمون ProActionCLIP
۳۹	۷-۳ جمع‌بندی
۴۱	۴ مشخصات یک پایان نامه و گزارش علمی
۴۲	۱-۴ مقدمه
۴۲	۲-۴ معیارهای ارزیابی
۴۲	۱-۲-۴ میانگین صحت
۴۳	۲-۲-۴ میزان فراموشی
۴۴	۳-۴ مجموعه داده

۴۴	۱-۳-۴ مجموعه داده‌ی UCF101
۴۵	۲-۳-۴ مجموعه داده‌ی HMDB51
۴۶	۴-۴ تنظیمات آزمایش
۴۶	۱-۴-۴ آزمایش با مجموعه داده‌ی UCF101
۴۷	۲-۴-۴ آزمایش با مجموعه داده‌ی HMDB51
۴۸	۵-۴ ارزیابی نتایج
۴۹	۵ جمع‌بندی و نتیجه‌گیری و پیشنهادات
۵۰	۱-۵ پیشنهادات
۵۱	منابع و مراجع
۵۹	پیوست
۶۰	واژه‌نامه‌ی فارسی به انگلیسی
۶۲	واژه‌نامه‌ی انگلیسی به فارسی

## فهرست شکل‌ها

صفحه

شکل

## فهرست جدول‌ها

صفحه

جدول

## فهرست نمادها

نماد	مفهوم
$\mathbb{R}^n$	فضای اقلیدسی با بعد $n$
$\mathbb{S}^n$	کره $n$ یکه بعدی
$M^m$	خمینه $m$ -بعدی $M$
$\mathfrak{X}(M)$	جبر میدان‌های برداری هموار روی $M$
$\mathfrak{X}^1(M)$	مجموعه میدان‌های برداری هموار $1$ یکه روی $(M, g)$
$\Omega^p(M)$	مجموعه $p$ -فرمی‌های روی خمینه $M$
$Q$	اپراتور ریچی
$\mathcal{R}$	تانسور انحنای ریمان
$ric$	تانسور ریچی
$L$	مشتق لی
$\Phi$	$2$ -فرم اساسی خمینه تماسی
$\nabla$	التصاق لوی-چویتای
$\Delta$	لاپلاسیین ناهموار
$\nabla^*$	عملگر خودالحاق صوری القا شده از التصاق لوی-چویتای
$g_s$	متر ساساکی
$\nabla$	التصاق لوی-چویتای وابسته به متر ساساکی
$\Delta$	عملگر لاپلاس-بلترامی روی $p$ -فرم‌ها

## فصل اول

راهنمای استفاده از الگوی لاتک دانشگاه

صنعتی امیرکبیر (پلی تکنیک تهران)

## ۱-۱ مقدمه

حروفچینی پروژه کارشناسی، پایان نامه یا رساله یکی از موارد پرکاربرد استفاده از زی پرشین است. از طرفی، یک پروژه، پایان نامه یا رساله، احتیاج به تنظیمات زیادی از نظر صفحه آرایی دارد که ممکن است برای یک کاربر مبتدی، مشکل باشد. به همین خاطر، برای راحتی کار کاربر، یک کلاس با نام AUTthesis برای حروفچینی پروژه ها، پایان نامه ها و رساله های دانشگاه صنعتی امیرکبیر با استفاده از نرم افزار زی پرشین، آماده شده است. این فایل به گونه ای طراحی شده است که کلیه خواسته های مورد نیاز مدیریت تحصیلات تکمیلی دانشگاه صنعتی امیرکبیر را برآورده می کند و نیز، حروفچینی بسیاری از قسمت های آن، به طور خودکار انجام می شود.

کلیه فایل های لازم برای حروفچینی با کلاس گفته شده، داخل پوشه ای به نام AUTthesis قرار داده شده است. توجه داشته باشید که برای استفاده از این کلاس باید فونت های PGaramond, B Nazanin و IranNastaliq روی سیستم شما نصب شده باشد.

## ۲-۱ این همه فایل؟!

از آنجایی که یک پایان نامه یا رساله، یک نوشته بلند محسوب می شود، لذا اگر همه تنظیمات و مطالب پایان نامه را داخل یک فایل قرار بدهیم، باعث شلوغی و سردرگمی می شود. به همین خاطر، قسمت های مختلف پایان نامه یا رساله داخل فایل های جداگانه قرار گرفته است. مثلاً تنظیمات پایه ای کلاس، داخل فایل AUTthesis.cls، تنظیمات قابل تغییر توسط کاربر، داخل commands.tex، قسمت مشخصات فارسی پایان نامه، داخل fa\_title.tex، مطالب فصل اول، داخل chapter1 و ... قرار داده شده است. نکته مهمی که در اینجا وجود دارد این است که از بین این فایل ها، فقط فایل AUTthesis.tex قابل اجرا است. یعنی بعد از تغییر فایل های دیگر، برای دیدن نتیجه تغییرات، باید این فایل را اجرا کرد. بقیه فایل ها به این فایل، کمک می کنند تا بتوانیم خروجی کار را ببینیم. اگر به فایل AUTthesis.tex دقت کنید، متوجه می شوید که قسمت های مختلف پایان نامه، توسط دستورهای مانند input و include به فایل اصلی، یعنی AUTthesis.tex معرفی شده اند. بنابراین، فایلی که همیشه با آن سروکار داریم، فایل AUTthesis.tex است. در این فایل، فرض شده است که پایان نامه یا رساله شما، از ۵ فصل و یک پیوست، تشکیل شده است. با این حال، اگر پایان نامه یا رساله شما، بیشتر از ۵ فصل و یک پیوست است، باید خودتان فصل های بیشتر را به این فایل، اضافه کنید. این کار، بسیار ساده است. فرض کنید بخواهید

یک فصل دیگر هم به پایان‌نامه، اضافه کنید. برای این کار، کافی است یک فایل با نام chapter6.tex و با پسوند .tex بسازید و آن را داخل پوشه AUTthesis قرار دهید و سپس این فایل را با دستور \include{chapter6} داخل فایل AUTthesis.tex و بعد از دستور \include{chapter6} قرار دهید.

## ۳-۱ از کجا شروع کنم؟

قبل از هر چیز، بدیهی است که باید یک توزیع تک مناسب مانند TeX Live و یک ویرایش‌گر تک مانند Texmaker را روی سیستم خود نصب کنید. نسخه بهینه شده Texmaker را می‌توانید از سایت [پارسی لاتک<sup>۱</sup>](http://www.parsilatex.com) و TeX Live را هم می‌توانید از [سایت رسمی آن<sup>۲</sup>](http://www.tug.org/texlive) دانلود کنید.

در مرحله بعد، سعی کنید که یک پشتیبان از پوشه AUTthesis بگیرید و آن را در یک جایی از هارددیسک سیستم خود ذخیره کنید تا در صورت خراب کردن فایل‌هایی که در حال حاضر، با آن‌ها کار می‌کنید، همه چیز را از دست ندهید.

حال اگر نوشتن پایان‌نامه اولین تجربه شما از کار با لاتک است، توصیه می‌شود که یک‌بار به طور سری، کتاب «[مقدمه‌ای نه چندان کوتاه بر L<sup>A</sup>T<sub>E</sub>X 2<sub>ε</sub>](http://www.parsilatex.com)»<sup>۳</sup> ترجمه دکتر مهدی امیدعلی، عضو هیات علمی دانشگاه شاهد را مطالعه کنید. این کتاب، کتاب بسیار کاملی است که خیلی از نیازهای شما در ارتباط با حروف‌چینی را برطرف می‌کند.

بعد از موارد گفته شده، فایل AUTthesis.tex و fa\_title را باز کنید و مشخصات پایان‌نامه خود مثل نام، نام خانوادگی، عنوان پایان‌نامه و ... را جایگزین مشخصات موجود در فایل fa\_title کنید. دقت داشته باشید که نیازی نیست نگران چینش این مشخصات در فایل پی‌دی‌اف خروجی باشید. فایل AUTthesis.cls همه این کارها را به طور خودکار برای شما انجام می‌دهد. در ضمن، موقع تغییر دادن دستورهای داخل فایل fa\_title کاملاً دقت کنید. این دستورها، خیلی حساس هستند و ممکن است با یک تغییر کوچک، موقع اجرا، خطا بگیرید. برای دیدن خروجی کار، فایل fa\_title را Save، (نه Save As) کنید و بعد به فایل AUTthesis.tex برگشته و آن را اجرا کنید. حال اگر می‌خواهید مشخصات انگلیسی پایان‌نامه را هم عوض کنید، فایل en\_title را باز کنید و مشخصات

<sup>۱</sup><http://www.parsilatex.com>

<sup>۲</sup><http://www.tug.org/texlive>

<sup>۳</sup><http://www.tug.ctan.org/tex-archive/info/lshort/persian/lshort.pdf>



داخل آن را تغییر دهید.<sup>۴</sup> در اینجا هم برای دیدن خروجی، باید این فایل را Save کرده و بعد به فایل AUTthesis.tex برگشته و آن را اجرا کرد.

برای راحتی بیشتر، فایل AUTthesis.cls طوری طراحی شده است که کافی است فقط یکبار مشخصات پایان نامه را وارد کنید. هر جای دیگر که لازم به درج این مشخصات باشد، این مشخصات به طور خودکار درج می شود. با این حال، اگر مایل بودید، می توانید تنظیمات موجود را تغییر دهید. توجه داشته باشید که اگر کاربر مبتدی هستید و یا با ساختار فایل های cls آشنایی ندارید، به هیچ وجه به این فایل، یعنی فایل AUTthesis.cls دست نزنید.

نکته دیگری که باید به آن توجه کنید این است که در فایل AUTthesis.cls، سه گزینه به نام های msc، bsc و phd برای تایپ پروژه، پایان نامه و رساله، طراحی شده است. بنابراین اگر قصد تایپ پروژه کارشناسی، پایان نامه یا رساله را دارید، در فایل AUTthesis.tex باید به ترتیب از گزینه های msc، bsc و phd استفاده کنید. با انتخاب هر کدام از این گزینه ها، تنظیمات مربوط به آنها به طور خودکار، اعمال می شود.

## ۴-۱ مطالب پایان نامه را چطور بنویسم؟

### ۱-۴-۱ نوشتن فصل ها

همان طور که در بخش ۱-۲ گفته شد، برای جلوگیری از شلوغی و سردرگمی کاربر در هنگام حروف چینی، قسمت های مختلف پایان نامه از جمله فصل ها، در فایل های جداگانه ای قرار داده شده اند. بنابراین، اگر می خواهید مثلاً مطالب فصل ۱ را تایپ کنید، باید فایل های AUTthesis.tex و chapter1 را باز کنید و محتویات داخل فایل chapter1 را پاک کرده و مطالب خود را تایپ کنید. توجه کنید که همان طور که قبلاً هم گفته شد، تنها فایل قابل اجرا، فایل AUTthesis.tex است. لذا برای دیدن حاصل (خروجی) فایل خود، باید فایل chapter1 را Save کرده و سپس فایل AUTthesis.tex را اجرا کنید. یک نکته بدیهی که در اینجا وجود دارد، این است که لازم نیست که فصل های پایان نامه را به ترتیب تایپ کنید. می توانید ابتدا مطالب فصل ۳ را تایپ کنید و سپس مطالب فصل ۱ را تایپ کنید.

نکته بسیار مهمی که در اینجا باید گفته شود این است که سیستم TEX، محتویات یک فایل تک

<sup>۴</sup> برای نوشتن پروژه کارشناسی، نیازی به وارد کردن مشخصات انگلیسی پروژه نیست. بنابراین، این مشخصات، به طور خودکار، نادیده گرفته می شود.

را به ترتیب پردازش می‌کند. به عنوان مثال، اگر فایلی، دارای ۴ خط دستور باشد، ابتدا خط ۱، بعد خط ۲، بعد خط ۳ و در آخر، خط ۴ پردازش می‌شود. بنابراین، اگر مثلاً مشغول تایپ مطالب فصل ۳ هستید، بهتر است که دو دستور `\include{chapter1}` و `\include{chapter2}` را در فایل `AUTthesis.tex` غیرفعال<sup>۵</sup> کنید. زیرا در غیر این صورت، ابتدا مطالب فصل ۱ و ۲ پردازش شده (که به درد ما نمی‌خورد؛ چون ما می‌خواهیم خروجی فصل ۳ را ببینیم) و سپس مطالب فصل ۳ پردازش می‌شود و این کار باعث طولانی شدن زمان اجرا می‌شود. زیرا هر چقدر حجم فایل اجرا شده، بیشتر باشد، زمان بیشتری هم برای اجرای آن، صرف می‌شود.

## ۲-۴-۱ مراجع

برای وارد کردن مراجع به فصل ۲ مراجعه کنید.

## ۳-۴-۱ واژه‌نامه فارسی به انگلیسی و برعکس

برای وارد کردن واژه‌نامه فارسی به انگلیسی و برعکس، بهتر است مانند روش بکار رفته در فایل‌های `dicen2fa` و `dicfa2en` عمل کنید.

## ۵-۱ اگر سوالی داشتیم، از کی بپرسیم؟

برای پرسیدن سوال‌های خود در مورد حروف چینی با زی‌پرشین، می‌توانید به **تالار گفتگوی پارسی لاتک**<sup>۶</sup> مراجعه کنید. شما هم می‌توانید روزی به سوال‌های دیگران در این تالار، جواب بدهید.

---

<sup>۵</sup> برای غیرفعال کردن یک دستور، کافی است پشت آن، یک علامت `%` بگذارید.

<sup>۶</sup><http://www.forum.parsilatex.com>

## فصل دوم

### مرور کارهای پیشین

## ۱-۲ مقدمه

در این فصل به معرفی یادگیری پیوسته در دو حوزه تصویر و ویدیو و مدل‌های بینایی-زبان در این حوزه‌ها می‌پردازیم. روش‌های یادگیری پیوسته به بخش‌های مبتنی بر تنظیم، بازپخش، بهینه‌سازی و معماری تقسیم می‌شوند که به مرور هرکدام از دسته‌های فوق می‌پردازیم.

## ۲-۲ یادگیری پیوسته

یادگیری پیوسته به توانایی یک سامانه‌ی هوشمند برای کسب، به‌روزرسانی، جمع‌آوری و بهره‌برداری از دانش در طول عمر آن اشاره دارد. این، شامل یادگیری یک دنباله از مطالب یا وظایف یکی پس از دیگری و سازگاری با اطلاعات جدید بدون فراموشی دانش قبلاً آموخته شده است. به علت تدریجی اضافه شدن مطالب، یادگیری مداوم به عنوان یادگیری افزایشی<sup>۱</sup>، یادگیری مستمر<sup>۲</sup> و یادگیری مادام‌العمر<sup>۳</sup> نیز شناخته می‌شود. هدف یادگیری پیوسته حفظ تعادل بین یادگیری اطلاعات جدید و حفظ دانش قبلاً کسب شده، با غلبه بر چالش فراموشی فاجعه‌بار است. در ادامه به معرفی چالش‌ها، سناریوها، رویکردها و کاربردهای این حوزه می‌پردازیم.

چالش اصلی که در این نوع یادگیری به وجود می‌آید، چالشی با نام فراموشی فاجعه‌بار است. علت این امر این است که وقتی داده‌های جدید برای یادگیری اضافه می‌شوند، سامانه مجبور می‌شود اطلاعات قبل را تا حدی به فراموشی بسپرد و شرایط جدید را در نظر بگیرد. برای همین به نوعی یک تعادل بین حالتی که حافظه ثابت است و حالتی که یادگیری انعطاف‌پذیر است، نیاز می‌باشد. در واقع هدف بر این است که یادگیری مستمر تعمیم‌پذیری خوبی برای تطبیق در شرایط مختلف با داده‌های جدید و با توزیع‌های جدید داشته باشد، همچنین کارایی منابع را نیز تضمین کند [۱، ۲].

همچنین، بر اساس نوع داده‌ها و وظیفه‌هایی که باید انجام شود، سناریوهای مختلفی برای یادگیری پیوسته ارائه شده است [۱]:

<sup>1</sup>Incremental learning

<sup>2</sup>Continuous learning

<sup>3</sup>Lifelong learning

### • یادگیری نمونه‌ای افزایشی

در یادگیری نمونه‌ای-افزاینده<sup>۴</sup> نمونه‌های داده‌ی جدید به طور پیوسته به مدل معرفی می‌شوند که هر یک نقطه داده جدید را نشان می‌دهد. این مدل باید با توزیع داده‌های در حال تکامل سازگار شود و در عین حال نمونه‌های جدید را نیز در نظر بگیرد.

### • یادگیری افزایشی دامنه

در یادگیری افزایشی دامنه<sup>۵</sup>، دامنه تغییر می‌کند و به معنای افزایش یا تغییر در توزیع داده‌ها، ویژگی‌ها یا محیط‌ها در مسئله‌ای که مدل در حال یادگیری آن است، می‌باشد. این تغییر می‌تواند به صورت افزودن داده‌های جدید از دامنه‌های جدید، تغییر در ویژگی‌ها، یا حتی تغییر مفهوم برخی اجزاء از داده‌ها (مثلاً تغییر تعبیر برچسب‌ها) اتفاق بیافتد. چالش این است که اطمینان حاصل شود که مدل می‌تواند با این حوزه‌های جدید سازگار شود، بدون اینکه عملکرد آن در حوزه‌هایی که قبلاً دیده شده‌اند، کاهش یابد [۲، ۳، ۴].

### • یادگیری افزایشی وظیفه

در یادگیری افزایشی وظیفه<sup>۶</sup> وظایف جدید در طول زمان معرفی می‌شوند و مدل باید یاد بگیرد که در هر وظیفه‌ی جدید به خوبی عمل کند و در عین حال عملکرد خود را در وظایفی که قبلاً آموخته‌است حفظ کند. در این نوع یادگیری، معمولاً شناسه‌ی وظیفه<sup>۷</sup> قبل از ورود داده به مدل مشخص می‌شود. به عنوان مثال، مشخص می‌شود که کدام وظیفه از مدل، باید با داده‌ی فعلی پیش‌بینی انجام دهد. [۳].

### • یادگیری افزایشی دسته‌ای

دسته‌های جدید به تدریج به داده‌های آموزشی مدل اضافه می‌شوند. مدل باید بیاموزد که دسته‌های جدید را تشخیص دهد و بین آنها تمایز قائل شود بدون اینکه دانش خود را در مورد دسته‌های آموخته شده قبلی فراموش کند [۲، ۳، ۵، ۶].

<sup>4</sup>Instance incremental learning

<sup>5</sup>Domain incremental learning

<sup>6</sup>Task incremental learning

<sup>7</sup>Task identity

همانطور که پیش تر گفته شد، رویکردهای متفاوتی برای یادگیری پیوسته ارائه شده است. ونگ و همکاران [۱]، به صورت زیر رویکردها را تقسیم بندی کرده اند:

## ۱-۲-۲ رویکرد مبتنی بر تنظیم

این رویکرد بر اساس این است که از تکنیک های مبتنی بر تنظیم<sup>۸</sup> برای ایجاد تعادل در وظایف قدیم و جدید استفاده کند. در حالت کلی نیز تکنیک های منظم سازی با ایجاد تغییراتی در محاسبات باعث جلوگیری از بیش برآزش مدل می شوند و در این جا نیز هدف این است که بین تاثیر مدل های قبلی و جدید تعادل ایجاد کند. برای رسیدن به این هدف نیز نیاز است که یک کپی از مدل های قبلی داشته باشد تا باعث فراموشی نشود. بر اساس اینکه چه نوع روشی استفاده شود، منظم سازی به دو دسته تقسیم می شود: منظم سازی وزن ( که وزن هایی که اهمیت بیشتری در نتیجه دارند را بیشتر نگه می دارد و وزن های بی اهمیت به مدل جدید با ضریب خیلی کم یا صفر انتقال می یابند). و منظم سازی تابع ( مدل جدید که با نام شاگرد از آن یاد می کند، تلاش می کند از اطلاعات خروجی مدل قبلی که با نام معلم از آن یاد می شود، برای ایجاد خروجی وظیفه ی خودش راهنمایی بگیرد) [۷، ۸].

## ۲-۲-۲ رویکرد مبتنی بر بازپخش

هدف این رویکرد در واقع این است که بخشی از داده های قبلی را در حافظه ذخیره کرده و آن ها را در وظیفه ی جدید با داده های جدید آموزش دهد و به این صورت هم مدل جدید آموزش داده می شود و هم اطلاعات قبلی فراموش نمی شوند. این رویکرد نیز به سه دسته ی بازپخش تجربه<sup>۹</sup> ( انتقال بخشی از داده های قبلی به حافظه)، بازپخش مولد<sup>۱۰</sup> (ایجاد یک مدل مولد اضافی برای بازپخش داده های تولید شده)، بازپخش ویژگی<sup>۱۱</sup> (انتقال ویژگی های مهم داده های قبلی به وظیفه ی جدید و جلوگیری از فراموشی فاجعه بار). به عنوان مثال شین و همکاران [۹]، مدلی به نام مدل مولد عمیق ارائه داده اند که چارچوبی است از یک معماری مدل دوگانه تعاونی، با یک مدل مولد عمیق (مولد) و یک مدل حل وظیفه (حل کننده) تشکیل شده است. مولد، مسئول تولید ورودی های جعلی است که شبیه داده های گذشته است، در حالی که حل کننده برای حل وظیفه فعلی آموزش دیده است. با درهم آمیختن داده های

<sup>8</sup>Regularization based

<sup>9</sup>Experience replay

<sup>10</sup>Generative replay

<sup>11</sup>Feature replay

آموزشی برای وظایف قبلی با داده‌های مربوط به وظیفه جدید، مدل می‌تواند بدون فراموش کردن دانش وظایف قدیمی، وظایف جدید را یاد بگیرد. این رویکرد از ماهیت مولد هیپوکامپ، یک سامانه حافظه کوتاه مدت در مغز پستانداران الهام گرفته شده است.

### ۳-۲-۲ رویکرد مبتنی بر بهینه‌سازی

رویکرد مبتنی بر بهینه‌سازی<sup>۱۲</sup> به جای تغییر در تابع خطا، به دستکاری برنامه‌های بهینه‌سازی می‌پردازد مثلاً به روزرسانی پارامترها<sup>۱۳</sup> به گونه‌ای صورت گیرد که با جهت‌هایی مانند فضای ورودی قبلی عمود یا هم تراز باشند تا اطلاعات وظایف قبلی نیز حفظ شود.

### ۴-۲-۲ رویکرد مبتنی بر معماری

رویکرد مبتنی بر معماری<sup>۱۴</sup> به ساخت پارامترهای خاص هر وظیفه برای جلوگیری از دخالت وظیفه‌ها در یکدیگر می‌پردازد و در واقع هدف بر کاهش محدودیت استفاده از پارامترهای مشترک وظیفه‌هاست زیرا با وجود این محدودیت‌ها، اطلاعات کمتری از وظیفه‌ی قبل به بعد منتقل می‌شود و فراموشی فاجعه‌بار با احتمال بیشتری رخ می‌دهد. در این رویکرد نیز تکنیک‌های مختلفی ارائه شده است مانند تخصیص پارامتر، تجزیه مدل و شبکه پیمانه‌ای.

### ۵-۲-۲ رویکرد ترکیب رویکردها و سناریوها

یکی از راه‌حل‌ها این است که از ترکیب رویکردهای ذکر شده با یکدیگر یا با دیگر رویکردهای شبکه عصبی استفاده شود. مثلاً استفاده از ترکیب رویکردهای مبتنی بر منظم‌سازی و مبتنی بر بازپخش.

### ۶-۲-۲ کاربردها

یادگیری پیوسته در حوزه بینایی ماشین کاربردهای گسترده‌ای دارد که از مهم‌ترین آن‌ها می‌توان به تشخیص چهره، دسته‌بندی تصویر، تشخیص اعمال در ویدیو، بخش‌بندی معنایی و تلفیق زبان و بینایی اشاره کرد. این قابلیت‌ها امکان آموزش پیوسته مدل‌ها را بدون فراموشی اطلاعات قبلی فراهم می‌سازند و

<sup>12</sup>Optimization based

<sup>13</sup>Parameters

<sup>14</sup>Architecture based

باعث می‌شوند سامانه‌های هوشمند در مواجهه شدن با داده‌های جدید، ضمن حفظ دانش گذشته، عملکرد خود را ارتقا دهند.

## ۳-۲ یادگیری پیوسته در بینایی کامپیوتر

یادگیری پیوسته در زمینه‌های مختلفی مورد استفاده قرار گرفته است. یکی از این زمینه‌ها بینایی کامپیوتر است که دو حوزه کاربرد به نام دسته‌بندی تصویر و تشخیص عمل را شامل شده و در ادامه به بررسی مقالات مطرح درباره آن‌ها می‌پردازیم.

### ۱-۳-۲ دسته‌بندی تصویر

مای و همکاران [۲]، مقایسه‌ای بین رویکردهای مطرح ارائه شده برای دسته‌بندی تصویر انجام داده‌اند که در ادامه به معرفی مختصر رویکردها و مقایسه آن‌ها می‌پردازیم.

#### روش تثبیت وزن کشسان

جیمز و همکاران [۷] مقاله‌ای در رابطه با روش جدیدی در زمینه‌ی منظم‌سازی ارائه داده‌اند. تثبیت وزن کشسان<sup>۱۵</sup> یک الگوریتم است که به شبکه‌های عصبی عمیق اجازه می‌دهد تا مجموعه‌ای از وظایف پیچیده را بدون فراموش کردن فاجعه‌آمیز یاد بگیرند. این کار را با کاهش انتخابی انعطاف‌پذیری وزن انجام می‌دهد. این روش با محدود کردن هر وزن با یک جریمه درجه دوم کار می‌کند که آن را به مقداری متناسب با اهمیت آن برای عملکرد در وظایفی که قبلاً یاد گرفته شده، به سمت مقادیر قدیمی خود می‌کشاند. در واقع از ماتریسی به نام ماتریس اطلاعات فشر برای محاسبه‌ی اهمیت وزن‌ها در مدل قبلی استفاده می‌کند.

#### روش یادگیری بدون فراموشی

لی و همکاران [۸]، الگوریتم یادگیری بدون فراموشی<sup>۱۶</sup> را ارائه داده‌اند. این الگوریتم برای یادگیری بدون فراموشی یک الگوریتم پیوسته است که هدف آن یادگیری وظایف جدید بدون فراموشی دانش وظایف قبلی می‌باشد. با آموزش یک مدل اولیه بر روی یک مجموعه وظایف اولیه کار می‌کند.

<sup>15</sup>Elastic weight consolidation (EWC)

<sup>16</sup>Learning without forgetting (LWF)



این مدل اولیه به عنوان مدل استاد نامیده می‌شود. هنگامی که با یک وظیفه جدید مواجه می‌شود، یک مدل جدید با نام مدل شاگرد، با استفاده از داده‌های وظیفه جدید، آموزش می‌بیند. مدل شاگرد همچنین با استفاده از داده‌های وظیفه‌های قدیمی، آموزش می‌بیند تا از مدل استاد پیروی کند. این به صورت محاسبه خروجی مدل استاد برای داده‌های جدید و سپس استفاده از آن خروجی به عنوان هدف برای آموزش مدل شاگرد انجام می‌شود. این روند به مدل شاگرد کمک می‌کند تا دانش وظایف قدیمی را حفظ کند، در حالی که در عین حال برای وظیفه جدید نیز بهینه می‌شود. از این روند به عنوان تقطیر دانش<sup>۱۷</sup> نیز یاد می‌کنند. عملکرد آن می‌تواند در صورتی که وظیفه جدید بسیار متفاوت از وظایف قدیمی باشد، کاهش یابد.

### روش میانگین حافظه رخدادی گرادیان

چودری و همکاران [۱۰]، الگوریتمی به نام الگوریتم میانگین حافظه رخدادی گرادیان<sup>۱۸</sup> ارائه کردند که با استفاده از یک حافظه ذخیره شده برای ذخیره اطلاعات مربوط به وظایف قبلی، از فراموشی فاجعه بار جلوگیری می‌کند. در هر مرحله از یادگیری، مدل از حافظه ذخیره شده برای تولید یک زیرمجموعه تصادفی از تجارب استفاده می‌کند. این تجربیات سپس برای محاسبه گام‌های گرادیان استفاده می‌شوند. گام‌های گرادیان فعلی سپس با گام‌های گرادیان زیرمجموعه تصادفی از حافظه ذخیره شده میانگین گیری می‌شوند. این میانگین گیری به مدل کمک می‌کند تا یک دیدگاه کلی تر از وظیفه فعلی به دست آورد. این دیدگاه کلی تر به مدل کمک می‌کند تا دانش وظایف قبلی خود را حفظ کند، حتی زمانی که در حال یادگیری یک وظیفه جدید است.

### روش طبقه بندی افزایشی و یادگیری بازنمایی

ربوفی و همکاران [۱۱]، روش طبقه بندی افزایشی و یادگیری بازنمایی<sup>۱۹</sup> در یادگیری پیوسته ارائه دادند که مدل با استفاده از نمونه‌هایی از همه دسته‌ها، از جمله دسته‌های جدید و قدیمی، آموزش داده می‌شود. تابع ضرر شامل یک ضرر طبقه بندی برای تشویق مدل به پیش بینی برچسب‌های صحیح برای دسته‌های جدید و یک ضرر تقطیر دانش برای تشویق مدل به بازتولید خروجی‌های مدل قبلی برای دسته‌های قدیمی است. همچنین یک روش به روزرسانی حافظه را پیشنهاد می‌دهد که بر اساس فاصله در فضای ویژگی‌های نهفته است. این روش برای انتخاب زیرمجموعه‌ای از نمونه‌ها از هر دسته

<sup>17</sup>Knowledge distillation (KD)

<sup>18</sup>Averaged Gradient Episodic Memory (A-GEM)

<sup>19</sup>Incremental Classifier and Representation Learning (iCaRL)

استفاده می شود که میانگین ویژگی های نهفته آنها به میانگین همه نمونه ها در این دسته نزدیک ترین است.

### روش حداکثر تداخل ارزیابی

الجانندی و همکاران [۱۲]، روشی به نام حداکثر تداخل ارزیابی<sup>۲۰</sup> را ارائه داده اند که یک روش مبتنی بر بازپخش است که اخیراً با هدف بهبود راهبرد بازیابی حافظه پیشنهاد شده است. این روش، نمونه های بازپخش را با توجه به افزایش ضرر با داشتن به به روزرسانی پارامتر تخمین زده شده بر اساس دسته های کوچک ورودی انتخاب می کند. نمونه های حافظه را که بیشترین تداخل (افزایش ضرر) را با به روزرسانی پارامتر با دسته ورودی جدید دارند، انتخاب می کند. همچنین نمونه گیری مخزن را در به روزرسانی حافظه اعمال می کند و نمونه های حافظه انتخابی را با نمونه های جدید در به روزرسانی مدل دوباره پخش می کند.

### ۲-۳-۲ تشخیص عمل

تشخیص عمل یکی از مباحث پیشرفته امروزی است که به علت اضافه شدن پارامتر زمان، پیچیدگی های بیشتری نسبت به تصویر پیدا کرده است. همچنین به علت حجیم بودن داده ها در این مسائل، یادگیری مداوم ضرورت پیدا می کند. زیرا در طول زمان مثلاً دسته ها و داده های بیشتری به مدل اضافه می شوند و مدل نمی تواند دوباره از ابتدا این حجم داده را آموزش دهد. پس چند سال اخیر، مطالعه هایی نیز در این زمینه شده و روش های جدید با مجموعه داده های مختلف ارائه شده است. که به چندین مقاله در ادامه می پردازیم.

### روش مین هاس

مین هاس و همکاران [۱۳]، یک روش یادگیری پیوسته برای تشخیص اعمال انسان ارائه داده اند. این روش بر پایه یک چارچوب است که دو تکنیک، یعنی تقریب شکل و یادگیری تحلیلی، را با هم ترکیب می کند. تقریب شکل برای ثبت شکل بازیگر در ویدیو استفاده می شود. به این صورت که با تغییراتی که به شکل می دهیم، از تصاویر شدت نور بهره برداری می کنیم تا ویژگی های مربوط به جهت گرایان ها را استخراج کنیم. هنگامی که حرکت در ویدیو پیش می رود، شکل با تغییر و تنظیم چندین قطعه کوچک داخل یک پنجره ی پیگیری، به طور دقیق به تغییرات مرزها پیگیری می کند. به منظور یادگیری دینامیک های غیرخطی حرکت ها، از یادگیری تحلیلی استفاده می شود. این فرآیند یادگیری به شکل

<sup>20</sup>Maximally Interfered Retrieval (MIR)

بازگشتی انجام می‌شود و از طریق آن آموزش به نمایش خطی ساده تبدیل می‌شود. این روش دو مزیت دارد: کمینه کردن خطاها و کاهش قابل توجه زمان محاسباتی، و از بین بردن محدودیت‌های آموزش به صورت دسته‌ای برای تشخیص اعمال. این روش یادگیری پیوسته اجازه می‌دهد که مدل به تدریج با ورودی داده‌های جدید به‌روز شود. این روش مقابل یادگیری دسته‌ای است که در آن برای آموزش دسته‌بند تمام مجموعه داده آموزشی استفاده می‌شود.

## روش الزنت

لی و همکاران [۱۴] روشی به نام الزنت را ارائه کردند که با انتخاب و به‌روزرسانی پویاترین بلوک‌های یادگیری از فراموشی فاجعه‌بار در شناسایی عمل جلوگیری می‌کند. هنگام یادگیری اعمال جدید، الزنت به دنبال بلوک‌های یادگیری می‌گردد که بیشترین ارتباط را با عمل فعلی دارند و پارامترهای آن‌ها را به‌روزرسانی می‌کند، در حالی که پارامترهای بلوک‌های غیرانتخابی را حفظ می‌کند. این راهبرد به‌روزرسانی انتخابی به حفظ دانش حرکات قبلاً یادگیری شده کمک می‌کند و مشکل فراموشی را کاهش می‌دهد. با به‌روزرسانی فقط بلوک‌های مرتبط، از وارد کردن نویز و اختلالات غیرمرتبط به دانش قبلی جلوگیری می‌کند، که منجر به عملکرد بهتر در یادگیری اعمال جدید می‌شود.

## روش تعبیه همسایه تی تصادفی موقت تحت نظارت

چنگ و همکاران [۱۵] یک روش برای تشخیص حرکات انسان با استفاده از تعبیه‌سازی همسایگی تصادفی زمانی نظارت شده و یادگیری پیوسته ارائه کرده‌اند. الگوریتم برای یادگیری ارتباط بین قاب‌های عمل به‌کار می‌رود و اطلاعات دسته و زمانی را تلفیق می‌کند. یادگیری پیوسته برای تعبیه‌سازی کم‌بعدی داده‌های جدید با استفاده از رویکردهایی نظیر تعبیه خطی محلی و پیش بینی حفظ محلی استفاده می‌شود. همچنین سه روش برای یادگیری پیوسته در زمینه تشخیص حرکات انسان توصیف می‌کند.

## روش پاریزی

پاریزی و همکاران [۱۶] رویکردی را ارائه کرده‌اند که باعث جلوگیری از فراموشی دانش با شبکه‌ی سلسله مراتبی خودسازمانده می‌شود. در این شبکه‌ی سلسله مراتبی هر لایه به صورت شبکه رشد هنگام نیاز است به این صورت که نورون‌های جدید را تخصیص می‌دهد یا نورون‌های موجود را بر اساس اختلاف بین توزیع ورودی و وزن‌های نورون‌های نمونه‌ای به‌روز می‌کند.

## ۴-۲ مدل‌های بینایی-زبان

مدل‌های زبانی بزرگ<sup>۲۱</sup>، شبکه‌های عصبی ترنسفورمر مقیاس‌پذیری هستند که با آموزش بر حجم عظیمی از داده‌های متنی، قادرند زبان طبیعی را تولید، درک و تحلیل کنند. این مدل‌ها به دلیل ظرفیت بالای خود در یادگیری، نقش اساسی در پیشرفت‌های اخیر پردازش زبان طبیعی<sup>۲۲</sup> ایفا کرده‌اند. به دنبال این پیشرفت‌ها پژوهش‌های زیادی انجام شده است که از این مدل‌ها در کاربردهای بینایی ماشین نیز استفاده کرده‌اند [۱۷]. مدل‌های بینایی-زبان<sup>۲۳</sup> دسته‌ای از مدل‌های هوش مصنوعی هستند که به طور هم‌زمان قادر به تحلیل و درک داده‌های بصری (تصویر یا ویدیو) و زبانی (متن) می‌باشند. این مدل‌ها با استفاده از حجم انبوهی از داده‌های تصویر-متن که به صورت گسترده در وب موجود است، آموزش می‌بینند. ایده اصلی پشت این مدل‌ها، یادگیری هم‌بستگی میان نمایش‌های تصویری و متنی در یک فضای مشترک نهفته<sup>۲۴</sup> است. به عنوان نمونه، مدل CLIP<sup>۲۵</sup> [۱۸] که توسط OpenAI ارائه شده است، با بهره‌گیری از صدها میلیون جفت تصویر و متن، توانسته است عملکرد قابل قبولی در وظایف مختلف بینایی و زبانی ارائه دهد. مدل‌های بینایی-زبان به دلایل متعددی مورد توجه پژوهشگران قرار گرفته‌اند که به برخی از آن‌ها در ادامه اشاره می‌کنیم [۱۷]:

- توانایی پیش‌بینی در حالت یادگیری بدون نمونه: این مدل‌ها قادرند وظایف جدید را بدون نیاز به بازآموزی<sup>۲۶</sup>، انجام دهند. به این حالت، یادگیری بدون نمونه<sup>۲۷</sup> گفته می‌شود.
- چندکاربردی بودن: یک مدل واحد می‌تواند در وظایف متنوعی همچون دسته‌بندی تصویر، تشخیص اشیاء، بازیابی تصویر بر اساس متن و تولید توضیح برای تصویر به کار گرفته شود.
- قابلیت مقیاس‌پذیری بالا: امکان آموزش بر روی میلیاردها جفت تصویر-متن و دستیابی به تعمیم‌پذیری قابل توجه در دامنه‌های گوناگون را دارد.

آموزش این مدل‌ها هزینه‌ی محاسباتی بالایی دارد اما طریقه‌ی استفاده از آن‌ها به صورتی است که این چالش تعدیل شود. استفاده از این مدل‌ها به سه مرحله‌ی اصلی تقسیم می‌شود: پیش‌آموزش، یادگیری انتقالی و تقطیر دانش که در ادامه بررسی می‌گردد.

<sup>21</sup> Large Language Models (LLMs)

<sup>22</sup> Natural language processing

<sup>23</sup> Vision language models (VLMs)

<sup>24</sup> Embedding space

<sup>25</sup> Contrastive Language-Image Pre-training (CLIP)

<sup>26</sup> Retraining

<sup>27</sup> Zero-shot learning

## ۲-۴-۱ پیش آموزش مدل‌های بینایی-زبان

در مرحله‌ی پیش آموزش مدل‌های بینایی-زبان<sup>۲۸</sup> مدل با بهره‌گیری از حجم انبوهی از داده‌های تصویر-متن بدون برچسب، به گونه‌ای آموزش می‌بیند که توانایی درک هم‌زمان مفاهیم زبانی و تصویری را کسب کند. سه نوع هدف آموزشی عمده در این بخش عبارت‌اند از:

### اهداف تقابلی

در روش اهداف تقابلی<sup>۲۹</sup> مدل یاد می‌گیرد تا نمایش جفت‌های صحیح تصویر-متن را به یکدیگر نزدیک و جفت‌های نادرست را از هم دور کند. به عنوان مثال مدل CLIP که با هدف تقابلی و داده‌های وب‌مقیاس آموزش داده شد و توانست در بیش از ۳۰ وظیفه نمونه-صفر عملکرد موفق‌تری ارائه دهد.

### اهداف مولد

در اهداف مولد<sup>۳۰</sup> مدل به بازسازی بخش‌های حذف‌شده از تصویر یا متن می‌پردازد یا توصیف متنی برای تصویر تولید می‌کند. به عنوان نمونه، مدل FLAVA<sup>[۱۹]</sup> با بهره‌گیری هم‌زمان از ماسک‌کردن تصویر و زبان، دانش چندحالتی‌ای را در یک مدل واحد می‌آموزد.

### اهداف هم‌ترازی

اهداف هم‌ترازی<sup>۳۱</sup> بر هم‌خوانی معنایی میان تصویر و متن، به صورت کلی (تطابق تصویر-متن<sup>۳۲</sup>) یا حتی به صورت محلی (تطابق واژه-ناحیه<sup>۳۳</sup>) تمرکز دارند. مدل GLIP<sup>[۲۰]</sup> با هم‌ترازی زبان-ناحیه توانست به شناسایی اشیای واژگان-باز<sup>۳۴</sup> دست یابد.

## ۲-۴-۲ یادگیری انتقالی مدل‌های بینایی-زبان

برای استفاده از مدل‌های بینایی-زبان در وظایف خاص مانند دسته‌بندی تصویر، تشخیص اشیاء یا بازیابی تصویر، لازم است که مدل با روش‌هایی کم‌هزینه و تطبیقی انتقال یابد. مهم‌ترین روش‌ها عبارت‌اند از:

<sup>28</sup>Vision-language model pre-training

<sup>29</sup>Contrastive objectives

<sup>30</sup>Generative objectives

<sup>31</sup>Alignment objectives

<sup>32</sup>image-text matching

<sup>33</sup>region-word matching

<sup>34</sup>Open-vocabulary

## تنظیم پرامپت

در روش تنظیم پرامپت <sup>۳۵</sup>، به جای تغییر ساختار داخلی مدل یا بازآموزی کامل آن، تلاش می‌شود تا ورودی‌های متنی (و در برخی موارد تصویری) به گونه‌ای هوشمندانه طراحی یا بهینه شوند که مدل بتواند عملکرد بهتری در وظیفه مورد نظر ارائه دهد. در واقع، مدل اصلی ثابت می‌ماند و تنها شکل ورودی‌هایی که به آن داده می‌شود، به کمک الگوریتم‌هایی قابل یادگیری، تغییر می‌کند. این رویکرد به ویژه برای وظایف با یادگیری محدود بسیار کارآمد است؛ زیرا به مدل اجازه می‌دهد با استفاده از اطلاعات آموخته شده قبلی، خود را با وظیفه جدید تطبیق دهد بدون آنکه پارامترهایش خیلی تغییر کند. لیو و همکاران [۲۱]، تشخیص حرکت را به مسئله تطبیق ویدیو-متن تبدیل کرده‌اند تا از قدرت نمایش‌های زبانی بهره ببرند. پرامپت‌سازی نقش کلیدی در نزدیک‌سازی وظیفه هدف به ساختار داده‌های پیش‌تمرین شده دارد و موجب بهبود عملکرد مدل در شرایط یادگیری بدون نمونه می‌شود. در مدل CoOp [۲۲] به جای استفاده از پرامپت‌های متنی دستی مانند "a photo of a [class name]" پرامپت‌هایی از کلمات قابل یادگیری طراحی شدند. این کلمات به صورت بردارهایی آموزش‌پذیر به مدل داده می‌شوند و نقش آنها تقویت معنای دسته‌بندی برای مدل است. این روش، که در بخش بعدی بیشتر به آن پرداخته می‌شود، باعث شد صحت مدل CLIP در دسته‌بندی چنددسته به ویژه در شرایط یادگیری محدود به طور قابل توجهی بهبود یابد.

## وفق دهنده‌ی ویژگی

وفق دهنده‌ی ویژگی <sup>۳۶</sup> یکی از روش‌های مؤثر برای انتقال مدل‌های بینایی-زبان به وظایف جدید بدون نیاز به بازآموزی کامل شبکه است. در این رویکرد، به جای تغییر پارامترهای اصلی مدل، لایه‌هایی سبک و کم‌پارامتر به انتهای یا میانه‌ی شبکه اضافه می‌شود تا ویژگی‌های استخراج شده از تصویر یا متن را با نیازهای وظیفه‌ی خاص منطبق کند. این تطبیق دهنده‌ها می‌توانند به صورت افزونه‌هایی جدا از معماری اصلی عمل کنند، بنابراین هسته‌ی مدل بدون تغییر باقی می‌ماند. در روش CLIP-Adapter [۲۳] مجموعه‌ای از لایه‌های سبک وزن به مدل CLIP افزوده شد تا ویژگی‌های استخراج شده از تصویر و متن پیش از تصمیم‌گیری نهایی پردازش و تطبیق یابند. این کار برای وظایف یادگیری کم‌نمونه نیز مؤثر بود، زیرا بدون نیاز به تغییر در مدل پایه، عملکرد بسیار مناسبی حاصل شد. این روش انتقالی به دلیل کم‌هزینه بودن و عدم نیاز به تنظیم مجدد کل مدل، برای بسیاری از کاربردهای عملی مناسب است.

<sup>35</sup>Prompt Tuning<sup>36</sup>Feature adapter

## سایر روش‌ها

در کنار تنظیم پرامپت و تطبیق‌دهنده‌های ویژگی، برخی روش‌ها نیز با تغییر مستقیم در پارامترهای مدل، در بهبود عملکرد مدل برای وظایف خاص، نقش دارند. این روش‌ها معمولاً شامل تنظیم دقیق کامل یا تلفیق مدل‌های یادگرفته‌شده با مدل اولیه هستند. در روش Wise-FT [۲۴]، یک رویکرد ساده اما مؤثر ارائه شده است که در آن وزن‌های مدل پایه و مدل تنظیم دقیق شده به صورت میانگین‌گیری وزنی ترکیب می‌شوند. این تکنیک باعث می‌شود که مدل هم از تعمیم‌پذیری مدل اولیه بهره‌برد و هم بتواند دانش خاص وظیفه‌ی جدید را بیاموزد، بدون آنکه دچار بیش‌برازش شود. در توسعه‌ی این روش، ونگ و همکاران [۲۵]، روش Open-VCLIP را برای تطبیق مدل CLIP برای داده‌های ویدیویی ارائه دادند به صورتی که دانش مدل CLIP نیز حفظ شود. این مدل در فصل بعد به صورت مفصل‌تری توضیح داده خواهد شد.

## ۳-۴-۲ تقطیر دانش

در تقطیر دانش، دانش مدل بینایی-زبان به یک مدل سبک‌تر منتقل می‌شود تا بتوان از آن در کاربردهای خاص و با منابع محدود استفاده کرد. دو کاربرد اصلی عبارت‌اند از:

### تقطیر دانش برای تشخیص شیء

در حوزه‌ی بینایی کامپیوتر، یکی از چالش‌های مهم، شناسایی اشیائی است که در داده‌های آموزش مدل پایه وجود نداشته‌اند. روش‌های متداول تشخیص شیء نیازمند برچسب‌گذاری دقیق و پرهزینه‌ی داده‌ها برای هر دسته هستند. در این میان، مدل‌های بینایی-زبان مانند CLIP که از داده‌های وب‌مقیاس و متن‌های توصیفی متنوع آموزش دیده‌اند، دارای دانش گسترده‌ای درباره‌ی مفاهیم بصری و زبانی هستند که می‌توان از آن‌ها برای توسعه‌ی مدل‌های تشخیص شیء استفاده کرد. در مدل VILD [۲۶] نمونه‌ای برجسته از این رویکرد است. این مدل با استفاده از تقطیر دانش از CLIP یک آشکارساز دو مرحله‌ای توسعه داده است که می‌تواند اشیاء خارج از مجموعه‌ی برچسب‌گذاری‌شده‌ی اولیه را شناسایی کند؛ به این صورت که ویژگی‌های بصری استخراج‌شده از تصاویر با تعبیه‌های متنی مدل CLIP مقایسه می‌شوند تا به جای اتکا به دسته‌های از پیش تعریف‌شده، اشیاء جدید نیز قابل شناسایی باشند. این روش نوعی تشخیص شیء واژگان-باز را ممکن می‌سازد که در بسیاری از کاربردهای دنیای واقعی اهمیت بالایی دارد.

## تقطیر دانش برای بخش‌بندی معنایی

بخش‌بندی معنایی به معنای اختصاص یک برچسب معنایی به هر پیکسل از تصویر است و از وظایف کلیدی در درک صحنه محسوب می‌شود. پیاده‌سازی موفق این وظیفه معمولاً نیازمند مجموعه داده‌های پر حجم و برچسب‌خورده در سطح پیکسل است، که تولید آن‌ها بسیار پرهزینه و زمان‌بر است. با این حال، مدل‌های بینایی-زبان که از داده‌های ضعیف برچسب‌خورده یا بدون برچسب بهره می‌برند، می‌توانند دانش انتزاعی خود را به مدل‌های سبک‌تر انتقال دهند تا نیاز به برچسب‌گذاری کاهش یابد. در CLIPSeg [۲۷] از ویژگی‌های استخراج‌شده توسط CLIP برای هر تصویر استفاده می‌کند و با افزودن یک رمزگشای سبک<sup>۳۷</sup> امکان پیش‌بینی نقشه‌های بخش‌بندی معنایی را تنها بر اساس توصیف متنی (prompt) فراهم می‌کند؛ برای مثال، با دادن جمله‌ای مانند «گربه در تصویر کجاست؟»، مدل قادر به تولید نقشه‌ای است که نواحی مربوط به گربه را برجسته کند. نکته قابل توجه این است که این مدل به یادگیری بدون نمونه دست یافته و برای انجام این کار نیازی به آموزش مجدد بر روی داده‌های هدف ندارد، که آن را برای کاربردهای در دنیای واقعی بسیار کارآمد و مقیاس‌پذیر می‌سازد.

## ۵-۲ یادگیری پیوسته در مدل‌های بینایی-زبان

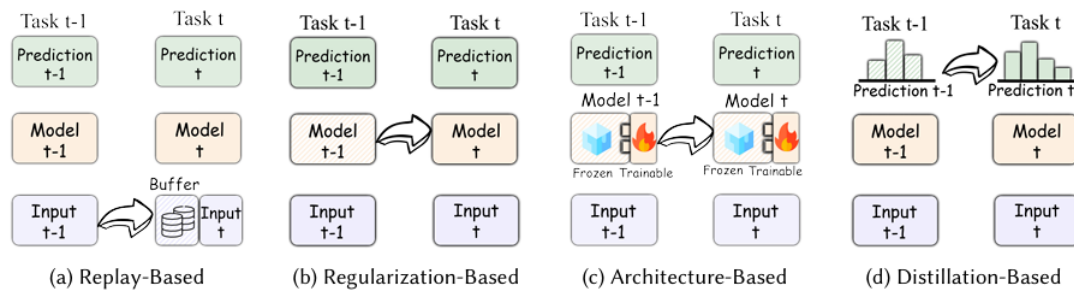
بیشتر مدل‌های زبانی بزرگ و بینایی-زبان، در شرایط ایستا آموزش می‌بینند و توانایی کمی در انطباق با داده‌های جدید، بدون بازآموزی کامل، دارند. این محدودیت، باعث توسعه و پیاده‌سازی روش‌های متنوع یادگیری پیوسته در این مدل‌ها شده است. در این زمینه، ژنگ و همکاران [۲۸] طبقه‌بندی شکل ۱-۲ را برای روش‌های متفاوت یادگیری پیوسته ارائه داده‌اند که در ادامه به آن پرداخته می‌شود.

### ۲-۵-۱ روش‌های مبتنی بر حافظه

در این روش، مدل بخشی از داده‌های قدیمی را ذخیره کرده و در کنار داده‌های جدید برای آموزش مجدد استفاده می‌کند. این رویکرد با هدف کاهش پدیده‌ی فراموشی طراحی شده است، که در آن مدل، دانش قبلی خود را هنگام یادگیری اطلاعات جدید از دست می‌دهد. محدودیت در ذخیره‌سازی داده‌های قدیمی چالش اصلی این روش است [۲۸]. گارگ و همکاران [۲۹]، روشی برای آموزش پیوسته‌ی مدل‌های بینایی-زبان مانند CLIP، در مواجهه شدن با داده‌های وب‌مقیاس و در حال تغییر زمانی، ارائه کرده‌اند.

<sup>37</sup>Lightweight decoder





شکل ۲-۱: طبقه‌بندی روش‌های یادگیری پیوسته در مدل‌های بینایی بزرگ و بینایی-زبان [۲۸]

این روش با بهره‌گیری از بازپخش داده‌های گذشته و استفاده از مدل پیش‌آمورخته به‌عنوان نقطه شروع، امکان به‌روزرسانی کارآمد مدل را، بدون نیاز به بازآموزی کامل، فراهم می‌کند.

## ۲-۵-۲ روش‌های مبتنی بر تنظیم

در این روش، مدل از مکانیزم‌هایی مانند جریمه یا محدودسازی برای حفظ اطلاعات قبلی استفاده می‌کند. هدف این است که پارامترهایی که در یادگیری گذشته مهم بوده‌اند، هنگام آموزش جدید کمتر تغییر کنند. در این روش، در صورت حجم زیاد وظایف، اثربخشی کاهش می‌یابد [۲۸].

## ۳-۵-۲ روش‌های مبتنی بر تقطیر دانش

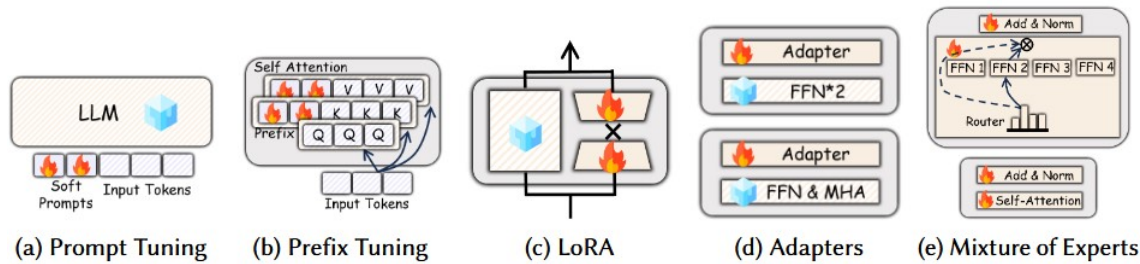
مدل دانش خود را از مدل‌های قبلی یا معلم یاد می‌گیرد و توجه به مواردی مانند حساسیت به صحت مدل معلم و انتخاب صحیح داده‌ها برای تقطیر حائز اهمیت است [۲۸]. لو و همکاران [۳۰]، با هدف کاهش فراموشی در یادگیری افزایشی ویدیو، از روشی به نام تقطیر توجه<sup>۳۸</sup> استفاده کرده‌اند که در آن ویژگی‌های توجه از خروجی کدگشای ترنسفورمر CLIP، به مدل جدید منتقل می‌شود. این رویکرد به مدل کمک می‌کند تا دانش مراحل قبلی را حفظ کرده و در عین حال بتواند دسته‌های جدید را بدون نیاز به آموزش کامل مجدد یاد بگیرد.

## ۴-۵-۲ روش‌های مبتنی بر معماری

در این رویکرد، معماری مدل برای جذب وظایف جدید، بدون تداخل با وظایف قبلی، تغییر می‌کند؛ مانند افزودن پیمانه<sup>۳۹</sup> استفاده از تطبیق‌دهنده‌ها و ... ژنگ و همکاران [۲۸]، با بررسی روش‌های مختلف

<sup>۳۸</sup>Attention distillation

<sup>۳۹</sup>Module



شکل ۲-۲: روش‌های یادگیری پیوسته مبتنی بر معماری در مدل‌های زبانی بزرگ و بینایی-زبان [۲۸]

ارائه‌شده در این رویکرد، طبقه‌بندی مطابق با شکل ۲-۲ ارائه داده اند که در ادامه به بررسی آن‌ها پرداخته می‌شود.

### تنظیم پرامپت

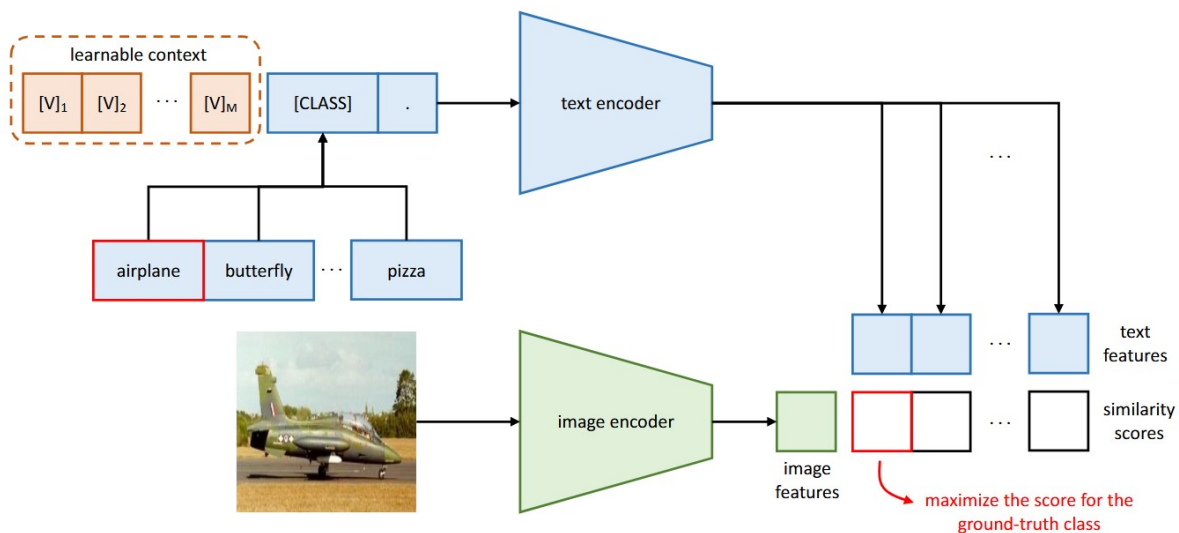
همانطور که در بخش یادگیری انتقالی مدل‌های بینایی-زبان ذکر شد، در روش تنظیم پرامپت، به جای بازآموزی کامل یا تغییر در پارامترهای اصلی مدل، مجموعه‌ای از بردارهای قابل آموزش به عنوان پرامپت به ابتدای نشانه‌های ورودی<sup>۴۰</sup> اضافه می‌شوند. این بردارها بدون دست‌کاری در ساختار درونی مدل، نقش راهنما را ایفا کرده و جهت‌گیری مدل در تفسیر داده‌های جدید را مشخص می‌سازند. در واقع، مدل با همان دانش قبلی خود به تحلیل ورودی می‌پردازد، اما به واسطه‌ی پرامپت‌های جدید، قادر به تطبیق با وظایف تازه می‌شود [۲۸]. این روش به دلیل مصرف کم منابع محاسباتی و عدم نیاز به تغییر در پارامترهای اصلی، به‌ویژه برای سناریوهایی با دسترسی محدود به مدل یا منابع، بسیار مناسب است و الگوی مورد استفاده در این روش، می‌تواند به خوبی برای مسائل یادگیری پیوسته نیز استفاده شود [۲۸]. ژو و همکاران [۲۲]، رویکردی به نام بهینه‌سازی بافت<sup>۴۱</sup> (که در بخش قبل اشاره شد) ارائه کرده‌اند که به جای پرامپت‌های ثابت، از پرامپت به صورت بردارهای بافت یادگرفتنی<sup>۴۲</sup> که در کنار برچسب متنی داده‌ها قرار می‌گیرند، استفاده می‌کند. مطابق شکل ۲-۳، تمام وزن‌های مدل CLIP، ثابت نگه داشته شده و تنها بردارهای پرامپت، قابل آموزش هستند. به دنبال توسعه‌ی روش‌های پرامپت گذاری، روش L2P<sup>۴۳</sup> که توسط وانگ و همکاران [۳۱] ارائه شده است، یک چارچوب نوآورانه برای یادگیری پیوسته بدون نیاز به شناسایی وظیفه در زمان آزمون می‌باشد. همان‌طور که در شکل ۲-۴ مشاهده می‌شود، این روش به جای تغییر وزن‌های مدل پیش‌آموخته، از مجموعه‌ای از پرامپت‌های یادگرفتنی بهره می‌برد که در یک

<sup>۴۰</sup>Input tokens

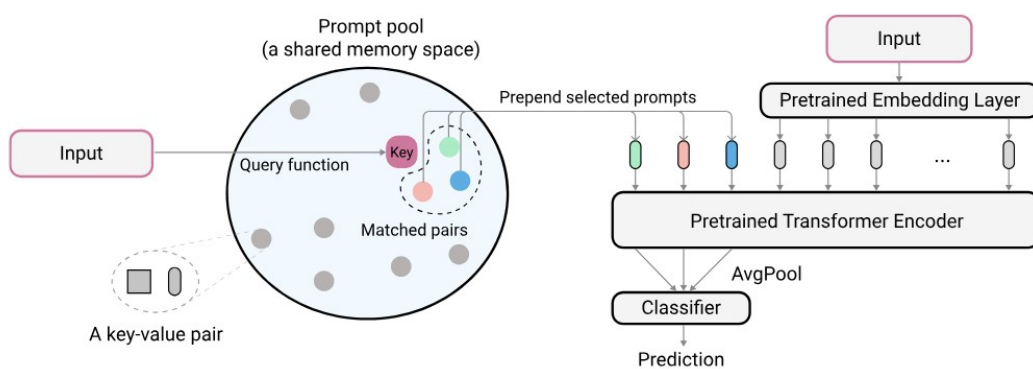
<sup>۴۱</sup>Context Optimization (CoOp)

<sup>۴۲</sup>Learnable context vectors

<sup>۴۳</sup>Learning to Prompt for Continual Learning



شکل ۲-۳: نمایش روش CoOp [۲۲]



شکل ۲-۴: نمایش روش L2P در زمان آزمون [۳۱]

فضای حافظه اشتراکی به نام استخر پرامپت<sup>۴۴</sup> نگهداری می‌شوند. L2P از یک مکانیزم پرس‌وجوی مبتنی بر جفت‌های کلید-مقدار بهره می‌برد تا به صورت پویا و متناسب با ورودی، پرامپت‌های مرتبط را انتخاب کرده و به نشانه‌های ورودی مدل، اضافه کند. سپس این نشانه‌های توسعه‌یافته به مدل پیش‌آمोخته تزریق شده و پیش‌بینی انجام می‌شود. در این روش، پرامپت‌ها، دانش خاص هر وظیفه یا دانش مشترک بین وظایف را به صورت فشرده ذخیره می‌کنند و باعث کاهش چشمگیر فراموشی مخرب در یادگیری وظایف متوالی می‌شوند. ساختار طراحی شده در L2P همان‌طور که در شکل ۲-۴ نشان داده شده، از یک بخش انتخاب پرامپت، لایه‌های کدگذار پیش‌آمोخته<sup>۴۵</sup>، و دسته‌بند نهایی تشکیل شده است. جهت بهبود و مقاوم‌سازی نسبت به فراموشی در انتخاب پرامپت روش‌های پیشین، مارتین و همکاران [۳۲]، روش STAR-Prompt را معرفی کرده‌اند که از رویکردی دوسطحی برای تنظیم پرامپت، پیروی می‌کند. ابتدا از CLIP، برای تولید پرامپت‌های متنی و ساخت نمونه‌های اولیه<sup>۴۶</sup> پایدار دسته‌ها، استفاده می‌شود و سپس این نمونه‌های اولیه به عنوان کلید برای بازیابی پرامپت‌های تصویری در ترنسفورمر تصویر به کار می‌روند. هم‌چنین روش‌های DualPrompt [۳۳] و H-prompts [۳۴]، مانند مطالعات مذکور، در زمینه‌ی تولید پرامپت‌های مشترک و خاص وظایف ارائه شده‌اند. داهوین و همکاران [۳۵]، از پرامپت اختصاصی برای هر نمونه به جای هر دسته، استفاده کرده‌اند. برخلاف روش‌های پرامپت‌گذاری قبلی، هنگ و همکاران [۳۶]، علاوه بر استفاده از یک پرامپت به جای چند پرامپت، از نمونه‌های پرت مصنوعی برای ایجاد مرز دسته‌بندی بهتر استفاده کرده‌اند. در مطالعات دیگری مانند روش یکپارچه‌سازی دانش بدون تداخل و آگاه از توزیع [۳۷]<sup>۴۷</sup>، به رفع چالش مداخله‌ی پرامپت در تصمیم‌گیری مکانیزم توجه پرداخته شده است. به دنبال پیشرفت‌های روش‌های پرامپت در حوزه‌ی تصویر، ویلا و همکاران [۳۸]، روشی به نام PIVOT را معرفی کرده‌اند که با بهره‌گیری از دانش پیش‌آمोخته مدل تصویر-متن CLIP و استفاده از پرامپت‌های مکانی<sup>۴۸</sup> و زمانی، وابستگی‌های زمانی و مکانی ویدیوها را مدل کرده است. وانگ و همکاران [۳۹] نیز با هدف ارتقای عملکرد مدل CLIP در تشخیص حرکت‌های انسانی در ویدیوها، چارچوبی معرفی کرده‌اند که با استفاده از مدل‌سازی حرکتی و پرامپت‌های پویا، به شکلی مؤثر اطلاعات حرکتی را وارد فرآیند یادگیری می‌کند بدون اینکه به تغییر پارامترهای اصلی CLIP نیاز باشد. در مواردی نیز مانند روش ViLT-CLIP [۴۰]، از هر دو نوع پرامپت برای تصویر و متن برای درک ویژگی‌های ویدیویی استفاده

<sup>44</sup> Prompt pool<sup>45</sup> Pretrained encoder layers<sup>46</sup> Prototypes<sup>47</sup> Distribution-aware Interference-free Knowledge Integration (DIKI)<sup>48</sup> Spatial prompts

شده است.

### تنظیم پیشوند

در روش تنظیم پیشوند<sup>۴۹</sup>، مجموعه‌ای از پارامترهای قابل آموزش به عنوان پیشوند به ابتدای هر لایه‌ی ترنسفورمر افزوده می‌شود تا رفتار مدل را در انجام وظایف خاص تنظیم کند. این پیشوندها نقش تغییرات زمینه‌ای را ایفا کرده و برخلاف تنظیم پرامپت، چندین لایه از مدل را تحت تأثیر قرار می‌دهند [۲۸]. روی و همکاران [۴۱]، روشی معرفی کرده‌اند که با استفاده از پیشوندهای قابل یادگیری در هر لایه‌ی مدل، امکان یادگیری وظایف جدید را بدون فراموشی وظایف قبلی فراهم می‌کند. این پیشوندها با ترکیب کانولوشن و اطلاعات مشترک بین وظایف، باعث انتقال بهتر دانش و کاهش تعداد پارامترهای لازم در یادگیری پیوسته می‌شوند.

### سازگاری رتبه‌پایین

با وارد کردن ماتریس‌های رتبه‌پایین در لایه‌های معینی از مدل پیش‌آمخته و منجمد، روش سازگاری رتبه‌پایین<sup>۵۰</sup>، امکان تنظیم هدفمند بخش‌هایی از مدل را بدون بازآموزی کامل فراهم می‌سازد [۲۸]. مارتین و همکاران [۴۲]، به نتیجه رسیدند که در زمینه‌ی یادگیری پیوسته، جایگزینی روش مذکور به جای تنظیم پرامپت، بدون افزایش چشمگیر در تعداد پارامترها، منجر به بهبود عملکرد مدل می‌گردد.

### وفق دهنده

همانطور که در بخش یادگیری انتقالی مدل‌های بینایی-زبان ذکر شد، وفق دهنده‌ها شبکه‌های عصبی کوچک با ساختار فشرده‌ای هستند که بین لایه‌های مدل اصلی قرار می‌گیرند و به مدل امکان می‌دهند ویژگی‌های جدید را یاد بگیرد، بدون آن‌که نیازی به تغییر پارامترهای اصلی از پیش آموزش دیده باشد [۲۸]. به همین دلیل، می‌توان این روش را نیز در یادگیری پیوسته استفاده نمود. در برخی روش‌ها مانند DIA [۴۳] و EASE [۴۴]، از وفق دهنده‌های سبک‌وزن و اختصاصی برای هر وظیفه‌ی جدید استفاده می‌شود تا مدل بتواند بدون بازآموزی کامل یا ذخیره داده‌های قدیمی، دانش جدید را جذب کند. این دو روش، امکان به‌روزرسانی مدل را بدون آسیب به دانش قبلی فراهم کرده و تصمیم‌گیری ترکیبی میان دسته‌های قدیم و جدید را ممکن می‌سازند. دانگ و همکاران [۴۵]، در مطالعه‌ای دیگر، روشی به نام

<sup>49</sup>Prefix tuning

<sup>50</sup>Low-Rank Adaptation (LoRA)

C-ADA، ارائه داده‌اند که با استفاده از وفق دهنده‌هایی با قابلیت گسترش عاملی، یادگیری وظایف جدید را بدون نیاز به ذخیره داده‌های گذشته ممکن می‌سازد. این روش با حفظ پارامترهای قبلی و افزودن وزن‌های جدید، از تداخل دانش جلوگیری کرده و عملکرد و سرعت آموزش را به‌طور محسوسی بهبود می‌دهد. برای رفع چالش تداخل پارامترهای دسته‌های مشابه در یادگیری پیوسته، هانگ و همکاران [۴۶]، با استفاده از وفق دهنده‌های قابل تنظیم، ابتدا بازنمایی‌های متنی را متناسب با تأثیر دسته‌های جدید بر دسته‌های قدیمی اصلاح می‌کنند و سپس با یک راه‌برد تجزیه و ادغام پارامترها، فراموشی مدل در حین تنظیم وفق دهنده‌ها را کاهش می‌دهند. در حوزه‌ی ویدیو نیز، پن و همکاران [۴۷]، روشی به نام ST-Adapter پیشنهاد داده‌اند که با افزودن قابلیت زمانی-مکانی به مدل تصویری از پیش‌آموخته، آن را برای وظایف ویدیویی قابل استفاده کرده است.

### مخلوط خبره‌ها

روش مخلوط خبره‌ها<sup>۵۱</sup>، با استفاده از یک مکانیزم دروازه‌ای، به‌صورت پویا تعدادی از شبکه‌های عصبی خبره را برای انجام هر وظیفه فعال می‌کند. این ساختار باعث می‌شود مدل بخش‌های مختلف خود را به وظایف متنوع اختصاص دهد و عملکرد بهتر و مقیاس‌پذیری بیشتری پیدا کند. ونگ و همکاران [۴۸]، در این زمینه رویکردی را ارائه داده‌اند که مدل به‌صورت خودکار تصمیم می‌گیرد که بسته به تغییر داده یا وظیفه، از کدام وفق دهنده‌های موجود استفاده کند یا وفق دهنده‌ی جدیدی اضافه نماید، تا تعادلی میان حفظ دانش قبلی و یادگیری دانش جدید ایجاد شود. در ادامه نیز، یو و همکاران [۴۹]، با استفاده از همین روش و با گسترش تدریجی مدل CLIP و استفاده از مسیرهای انتخابی میان وفق دهنده‌های خبره و مدل اصلی، قابلیت تشخیص یادگیری بدون نمونه حفظ شده و در عین حال بار محاسباتی به شکل چشم‌گیری کاهش می‌یابد.

## ۶-۲ جمع‌بندی

در این فصل، به بررسی یادگیری پیوسته با تمرکز بر حوزه‌های تصویر، ویدیو و مدل‌های بینایی-زبان پرداخته شد. ابتدا مفهوم یادگیری پیوسته معرفی شد که هدف آن توانایی سیستم‌های هوشمند در کسب و به‌روزرسانی دانش در طول زمان بدون فراموش کردن دانش پیشین است. این حوزه با چالش اصلی «فراموشی فاجعه‌بار» روبه‌روست؛ مشکلی که هنگام ورود داده‌های جدید، باعث تضعیف یا حذف

<sup>51</sup>Mixture of Experts (MoE)

دانش قبلی مدل می‌شود.

رویکردهای اصلی مقابله با این چالش در چهار دسته‌ی مبتنی بر تنظیم، بازپخش، بهینه‌سازی و معماری طبقه‌بندی شدند. هر یک از این روش‌ها مزایا و محدودیت‌های خاص خود را دارند، از جمله استفاده از حافظه خارجی برای بازپخش داده‌های قدیمی، یا تنظیم پارامترها برای حفظ دانش پیشین. همچنین، مدل‌های بینایی-زبان معرفی و بررسی شدند. این مدل‌ها با ترکیب اطلاعات تصویری و متنی، قادر به درک عمیق‌تری از محتوای چندرسانه‌ای هستند و در وظایفی مانند توصیف تصاویر، پرسش‌وپاسخ تصویری و جستجوی مبتنی بر تصویر نقش مهمی ایفا می‌کنند. ادغام این مدل‌ها با رویکردهای یادگیری پیوسته، نیازمند راهکارهایی برای مدیریت همزمان دانش در هر دو حوزه بینایی و زبان است.

در مجموع، این فصل اهمیت یادگیری پیوسته در توسعه سامانه‌های هوشمند را برجسته کرد و نشان داد که طراحی روش‌های نوآورانه برای مدیریت داده‌های جدید و حفظ اطلاعات پیشین، به‌ویژه در مدل‌های بینایی-زبان، یکی از محورهای کلیدی پیشرفت در هوش مصنوعی آینده است.

# فصل سوم

## روش پیشنهادی



## ۱-۳ مقدمه

روش‌های موجود در زمینه یادگیری پیوسته برای داده‌های ویدیویی با وجود پیشرفت‌های اخیر، همچنان با مشکلات اساسی روبه‌رو هستند. برخی رویکردها به مدل‌های از پیش آموزش‌دیده متکی‌اند، اما برای انطباق با داده‌های ویدیویی نیاز به آموزش یا تنظیم مجدد کدگذارهای زمانی دارند، که فرآیندی زمان‌بر، پرهزینه و وابسته به منابع سخت‌افزاری سنگین است [۳۸]. هم‌چنین، روش‌های مختلف (تصویر یا ویدیو)، برای مقابله با فراموشی فاجعه‌بار به استفاده از بافرهای بازپخش یا ذخیره‌سازی داده‌های قبلی متکی هستند که نیازمند حافظه بالا و ناسازگار با محدودیت‌های حریم خصوصی است [۳۸، ۹، ۵۰]. علاوه بر این، برخی از روش‌ها از ساختارهای وابسته به وظیفه<sup>۱</sup> استفاده می‌کنند که مدیریت و نگهداری آن‌ها در سناریوهای واقعی و وظایف متوالی دشوار بوده و باعث کاهش تعمیم‌پذیری می‌شود [۵۱، ۵۲]. با توجه به این محدودیت‌ها، نیاز به رویکردهایی احساس می‌شود که بتوانند بدون وابستگی به ذخیره‌سازی وسیع داده‌های گذشته یا آموزش سنگین کدگذارها، عملکرد بهتری در داده‌های ویدیویی ارائه دهند و در عین حال ماهیت مستقل از وظیفه<sup>۲</sup> داشته باشند. هدف چنین رویکردهایی این است که از ظرفیت مدل‌های بزرگ و از پیش آموزش‌دیده استفاده کرده و با اضافه کردن لایه‌های سبک یا پرامپت‌های یادگیرنده، بدون تغییر مستقیم پارامترهای اصلی مدل، دانش قبلی را حفظ کنند و با داده‌های جدید تطبیق یابند.

در راستای رفع محدودیت‌های مذکور، در این فصل به ارائه روشی با عنوان ProActionCLIP<sup>۳</sup>، پرداخته می‌شود که با ترکیب قابلیت‌های روش Open-VCLIP [۲۵] و ایده‌های روش L2P [۳۱]، از پرامپت‌های سبک و پویا، برای انطباق با وظایف جدید، استفاده می‌کند. بنابراین در این روش، نیازی به طی کردن فرآیند آموزش کدگذارهای زمانی، که از نظر محاسباتی هزینه‌بر است، وجود ندارد. روش ProActionCLIP، با بهره‌گیری بهینه از دانش مدل‌های پیش‌آموزش‌دیده، کاهش فراموشی فاجعه‌بار و حفظ کارایی در سناریوهای واقعی را هدف قرار داده است. این روش می‌تواند با بهره‌گیری بهینه از منابع محاسباتی و نیاز کمتر به سخت‌افزار، توانایی یادگیری پیوسته وظایف جدید را داشته باشد و بدون وابستگی به شناسه وظایف، عملکردی کارآمد و مقیاس‌پذیر ارائه دهد.

<sup>۱</sup>Task specific<sup>۲</sup>Task-agnostic

<sup>۳</sup>این نام مخفف Prompt Action recognition CLIP می‌باشد که به استفاده از روش پرامپت‌گذاری برای تشخیص حرکت توسط مدل کلیپ، اشاره می‌کند.

## ۲-۳ روش ProActionCLIP

روش ProActionCLIP، که برای یادگیری پیوسته‌ی تشخیص حرکت انسان معرفی شده، بر پایه‌ی ترکیب دو رویکرد Open-VCLIP و L2P، بنا شده است. ایده‌ی اصلی این روش آن است که از قابلیت‌های Open-VCLIP برای استخراج ویژگی‌های چندماهیتی (تصویر-متن) بهره گرفته و در عین حال از سازوکار پرامپت‌های یادگیرنده در L2P استفاده می‌کند. به این ترتیب مدل می‌تواند بدون نیاز به تغییر مستقیم پارامترهای کدگذار اصلی، خود را با وظایف متوالی تطبیق دهد. این ترکیب باعث می‌شود که مشکل فراموشی فاجعه‌بار کاهش یافته، حافظه‌ی مورد نیاز برای ذخیره‌سازی نمونه‌ها به حداقل برسد و مدل به صورت مستقل از وظیفه، وظایف جدید را پردازش کند. به عبارت دیگر، با این رویکرد سعی شده است مزیت‌های هر دو روش با هم ترکیب شود که عبارت‌اند از: قدرت تعمیم‌دهی و دانش وسیع Open-VCLIP و انعطاف‌پذیری L2P در مدیریت وظایف پیوسته. مدل ProActionCLIP، شامل دو مرحله‌ی آموزش و آزمون می‌باشد که پس از معرفی مدل‌های پایه‌ی بکار برده شده در ProActionCLIP، به توضیح آن‌ها پرداخته می‌شود.

## ۳-۳ مدل Open-VCLIP

مدل‌های بینایی-زبان مانند CLIP، به دلیل توانایی یادگیری بازنمایی‌های مشترک تصویر و متن، عملکرد قابل توجهی در وظایف بینایی و زبانی داشته‌اند. با این حال، این مدل‌ها در حالت پایه برای داده‌های ایستا (تصاویر) طراحی شده‌اند. Open-VCLIP با گسترش معماری CLIP و افزودن قابلیت درک اطلاعات زمانی، این محدودیت را برطرف می‌کند و روشی کارآمد برای تحلیل ویدیو ارائه می‌دهد. مدل Open-VCLIP هم‌چنین به منظور حفظ قابلیت یادگیری بدون نمونه‌ی مدل پایه‌ی CLIP و جلوگیری از فراموشی آن در فرایند یادگیری داده‌های ویدیویی، از دو تکنیک منظم‌سازی وزن‌های درون‌یابی<sup>۴</sup> و میانگین تصادفی وزن‌ها<sup>۵</sup> استفاده می‌کند. تکنیک اول با بکارگیری روش ترکیب مدل قدیمی و جدید و تغییراتی در آن، از فراموشی دانش قبلی جلوگیری کرده و هم‌زمان باعث یادگیری دانش جدید می‌شود. در تکنیک دوم، با میانگین‌گیری از وزن‌های مدل در نقاط مختلف، باعث بهبود وزن‌های نهایی از لحاظ تعمیم‌پذیری می‌شوند. در ادامه، ابتدا نحوه‌ی تعمیم مدل CLIP برای داده‌های ویدیویی بررسی شده و سپس تکنیک‌های منظم‌سازی وزن‌های درون‌یابی و میانگین تصادفی وزن‌ها شرح داده خواهد شد.

<sup>۴</sup>Interpolation Weight Regularization (IWR)

<sup>۵</sup>Stochastic Weight Averaging (SWA)

### ۱-۳-۳ تبدیل CLIP مبتنی بر تصویر به CLIP مبتنی بر ویدیو

برای تعمیم مدل CLIP برای داده‌های ویدیویی، ورودی ویدیویی به دنباله‌ای از قاب‌ها تبدیل می‌شود و هر قاب توسط کدگذار تصویری CLIP به یک بردار ویژگی تبدیل می‌گردد. سپس این ویژگی‌ها در قالب دنباله‌ای زمانی قرار گرفته و با سازوکار توجه ترکیب می‌شوند. مفهومی به نام وصله<sup>۶</sup> تصویر که در ادامه ذکر شده است، ناحیه‌ای مستطیلی و پیوسته از تصویر اصلی با ابعاد مشخص است که به‌منظور بررسی جزئیات محلی و ویژگی‌های بخش‌های مختلف تصویر استفاده می‌شود (مانند نمونه‌ی تیره شده در شکل ۱-۳). در سازوکار توجه فرمول محاسبه‌ی خروجی مطابق رابطه‌ی (۱-۳) می‌باشد:

$$y_{s,t} = \text{Softmax} \left( \frac{q_{s,t} K_t^T}{\sqrt{d}} \right) V_t. \quad (1-3)$$

در این رابطه،  $q_{s,t}$  بردار پرسمان<sup>۷</sup> برای یک وصله از تصویر،  $K_t$  ماتریس کلید و  $V_t$  ماتریس مقدار برای قاب یا تصویر  $t$  هستند که از طریق سازوکار توجه ترکیب می‌شوند. در این حالت، ارتباط یک وصله از تصویر با خودش و بقیه‌ی وصله‌های تصویر مشخص می‌شود. برای تعمیم مدل برای داده‌های ویدیویی در مدل Open-VCLIP، بردار پرسمان وصله‌ی قاب فعلی را در ماتریس کلید قاب فعلی، بعدی و قبلی ( $K_{(t-1) \sim (t+1)}$ ) ضرب می‌کند و پس از اجرای تابع softmax نیز در ماتریس مقدار قاب فعلی، بعدی و قبلی ( $V_{(t-1) \sim (t+1)}$ ) ضرب می‌کند و سازوکار توجه را براساس رابطه‌ی (۲-۳)، محاسبه می‌کند. به این ترتیب ارتباط وصله قاب فعلی با سایر وصله‌های این قاب، و قاب‌های قبل و بعد از قاب فعلی در نظر گرفته می‌شود (مطابق با شکل ۱-۳). این راهکار به ظاهر ساده، توانست تحول خوبی در زمینه‌ی سازگاری مدل CLIP با داده‌ی ویدیویی ایجاد کند [۲۵].

$$y_{s,t} = \text{Softmax} \left( \frac{q_{s,t} [K_{(t-1) \sim (t+1)}]^T}{\sqrt{d}} \right) [V_{(t-1) \sim (t+1)}]. \quad (2-3)$$

<sup>۶</sup>Patch

<sup>۷</sup>Query



شکل ۳-۱: در سازوکار تغییریافته‌ی توجه، وصله‌های در نظر گرفته شده برای هر وصله از قاب (مانند وصله‌ی تیره در شکل) شامل وصله‌های قاب فعلی و بعدی و قبلی می‌باشند.

### ۳-۳-۲ منظم سازی مبتنی بر درونیابی وزن‌ها

همانطور که ذکر شد، برای جلوگیری از فراموشی دانش پیش‌آموزش و در عین حال سازگار کردن مدل با داده‌های جدید، منظم‌سازی وزن‌های درونیابی معرفی شده است. در این روش، وزن‌های مدل به صورت ترکیبی از وزن‌های اولیه (پیش‌آموزش) و وزن‌های به‌روزرسانی شده در وظیفه جدید، تنظیم می‌شوند. این درونیابی به مدل کمک می‌کند تا در حین یادگیری، تعادلی میان دانش قدیمی و اطلاعات تازه برقرار کرده و از بیش‌برازش<sup>۸</sup> جلوگیری کند. روش مذکور، تعمیم ایده‌ی گابریل و همکاران [۵۳]، که مطابق با رابطه‌ی (۳-۳) است، می‌باشد:

$$\theta = \lambda \theta_A + (1 - \lambda) \theta_B. \quad (3-3)$$

در این رابطه،  $\theta$  از ترکیب خطی وزن‌های مدل پایه  $\theta_A$  و مدل به‌روزرسانی شده  $\theta_B$  با ضریب  $\lambda$  تشکیل می‌شود تا صحت مدل در وظایف جدید افزایش یابد، بدون آنکه عملکرد آن در سایر وظایف که از پیش بهینه بوده‌اند، کاهش پیدا کند. با توجه به این که  $\lambda$  یک ابرپارامتر<sup>۹</sup> است و در رابطه‌ی (۳-۳) هیچ فرایند بهینه‌سازی مستقیمی روی مدل نهایی انجام نمی‌شود، مدل ترکیبی ممکن است روی داده‌های جدید ( $\theta_B$ ) عملکرد ضعیفی داشته باشد و دچار زیربرازش<sup>۱۰</sup> شده و همچنین، کیفیت مدل به شدت به مقدار پارامتر تعادلی  $\lambda$  وابسته شود [۲۵]. پس در مدل Open-VCLIP راه حلی ارائه شد که عملکرد مدل ترکیبی را در برابر بازه‌ای از مقادیر  $\lambda$  بهینه کند. مطابق با رابطه‌ی (۴-۳)، وزن‌های جدید در آموزش

<sup>8</sup>Overfitting

<sup>9</sup>Hyper parameter

<sup>10</sup>Underfitting

به سمتی می‌رود که هم زیان مدل جدید و هم زیان مدل ترکیبی با ضریب  $\alpha$  روی داده‌های جدید حداقل شود. در این حالت، ترکیب‌های مختلف مدل قبلی و جدید در مراحل مختلف آموزش در نظر گرفته می‌شود که در واقع، عملکرد بهتری در تحلیل داده‌های نادیده خواهد داشت. در انتها با استفاده از رابطه‌ی (۳-۳)، مدل جدید و قدیم ترکیب خواهند شد. با این تفاوت که وزن‌های مدل جدید، در طول آموزش با در نظر گرفتن عدم فراموشی مدل قبلی، یاد گرفته شده‌اند:

$$\arg \min_{\theta_B} \mathcal{L} = L(\theta_B; D_B) + \beta L(\alpha \theta_A + (1 - \alpha) \theta_B; D_B). \quad (۴-۳)$$

پارامتر  $\alpha$  در رابطه‌ی (۴-۳)، از یک توزیع یکنواخت در بازه  $(0, \lambda)$  نمونه‌برداری می‌شود و ضریب  $\beta$  به‌عنوان یک پارامتر تنظیم‌کننده برای کنترل میزان تاثیر عبارت درونی‌یابی تعریف شده است. همچنین مقدار  $\beta$  به‌صورت  $\beta = C \frac{1}{1-\alpha}$  محاسبه می‌شود که در آن  $C$  یک مقدار ثابت برای کنترل بزرگی  $\beta$  است.

### ۳-۳-۳ میانگین‌گیری تصادفی وزن‌ها

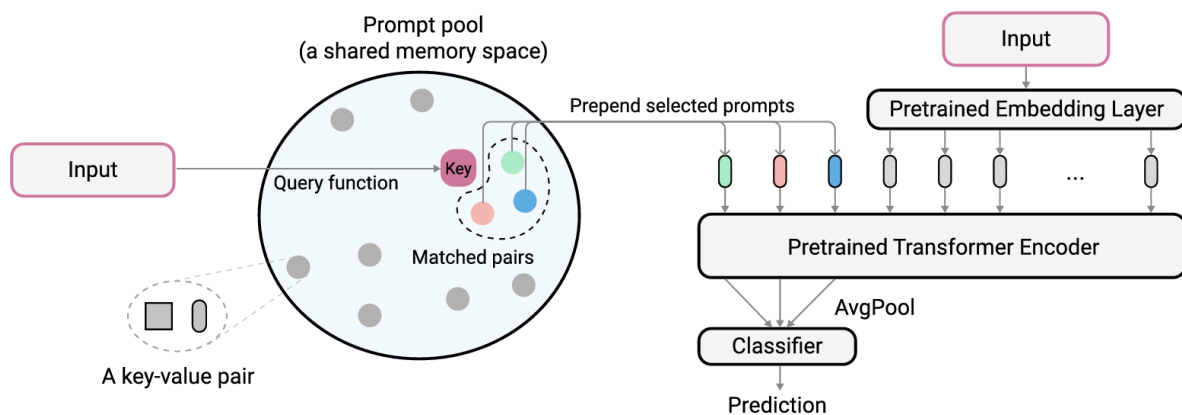
در فرایند آموزش شبکه‌های عصبی، مدل‌ها معمولاً در نواحی کمینه تیز<sup>۱۱</sup> در فضای وزن‌ها قرار می‌گیرند؛ این نقاط اگرچه روی داده‌های آموزش خطای پایینی دارند، اما به دلیل حساسیت زیاد به تغییرات وزن، معمولاً تعمیم خوبی روی داده‌های جدید ندارند. به‌منظور بهبود قابلیت تعمیم مدل، پاول و همکاران [۵۴]، روش میانگین‌گیری تصادفی وزن‌ها<sup>۱۲</sup> را معرفی کردند که با میانگین‌گیری از وزن‌های مدل در نقاط مختلف طی فرایند آموزش، مدل را به سمت کمینه مسطح<sup>۱۳</sup> هدایت می‌کند. این نواحی از فضای وزن‌ها تغییرات کمتری در تابع زیان دارند و باعث کاهش خطای آزمون و بهبود توانایی تعمیم مدل می‌شوند. به‌منظور بهبود تعمیم‌پذیری مدل Open-VCLIP نیز، از روش مذکور استفاده شده است. فرمول نهایی مدل Open-VCLIP با احتساب روش میانگین‌گیری تصادفی وزن‌ها، در رابطه‌ی (۵-۳) نشان داده شده است:

$$\sum_i^N \frac{\lambda \theta_A + (1 - \lambda) \theta_i}{N} = \lambda \theta_A + (1 - \lambda) \left( \frac{1}{N} \sum_i^N \theta_i \right). \quad (۵-۳)$$

<sup>۱۱</sup> Sharp minimum

<sup>۱۲</sup> Stochastic Weight Averaging (SWA)

<sup>۱۳</sup> Flat minimum



شکل ۳-۲: طرح کلی مدل L2P [۳۱]

در نهایت همان‌طور که ذکر شد، مدل Open-VCLIP نهایی، نسخه‌ی تنظیم دقیق‌شده از مدل CLIP با سازوکار توجه اصلاح‌شده است که به‌منظور جلوگیری از فراموشی دانش مدل اولیه و در عین حال حفظ قابلیت یادگیری بدون نمونه، دو تکنیک مکمل منظم‌سازی مبتنی بر درونیابی وزن‌ها و میانگین‌گیری تصادفی وزن‌ها، نیز به کار گرفته است.

## ۴-۳ مدل L2P

روش ارائه‌شده در این مقاله، با هدف بهبود یادگیری پیوسته، از سازوکاری مبتنی بر «استخر پرامپت» استفاده می‌کند. در این رویکرد، به‌جای تغییر پارامترهای اصلی مدل، مجموعه‌ای از پرامپت‌های قابل آموزش طراحی می‌شود که مدل با استفاده از آن‌ها قادر به استخراج اطلاعات مهم از داده‌های ورودی است. در هر مرحله یادگیری، پرامپت‌های مناسب بر اساس شباهت با داده‌های جدید انتخاب می‌شوند و این امر باعث می‌شود مدل بتواند دانش جدید را یاد بگیرد، بدون آنکه دانش قبلی را فراموش کند. این روش با بهره‌گیری از معماری ترنسفورمر، توانسته است تعادل موثری میان حفظ دانش گذشته و یادگیری وظایف جدید برقرار کند. همان‌طور که در شکل ۳-۲ قابل مشاهده است، این مدل دارای دو بخش انتخاب پرامپت‌ها و یادگیری و به‌روزرسانی پرامپت‌ها است که هر یک در ادامه توضیح داده می‌شود.

## ۱-۴-۳ انتخاب پرامپت

در بخش انتخاب پرامپت<sup>۱۴</sup>، از استخر پرامپت تعدادی پرامپت متناسب با ورودی تصویری انتخاب می‌شود. استخر پرامپت شامل تعدادی پرامپت به تعداد  $M$  بوده که به صورت  $\mathbf{P} = \{P_1, P_2, \dots, P_M\}$  نمایش داده می‌شوند. هر  $P_j \in \mathbb{R}^{L_p \times D}$  یک پرامپت منفرد با طول نشانه‌ی  $L_p$  و اندازه تعبیه‌ی  $D$ <sup>۱۵</sup> مشابه تعبیه‌ی ورودی است. همچنین هر پرامپت، دارای یک کلید است و فرآیند انتخاب بر اساس شباهت این کلیدها با بردار ویژگی مرتبط با کلاس ورودی انجام می‌گیرد. بردار ویژگی موردنظر از طریق استخراج‌کننده‌ی ویژگی، به‌دست می‌آید و سپس با کلیدهای موجود در استخر مقایسه می‌شود. در نهایت، تعدادی از پرامپت‌های کلیدهایی که بیشترین شباهت را با بردار ویژگی دارند انتخاب می‌شوند (مطابق با شکل ۲-۳). مجموعه‌ای از کلیدها به صورت  $\mathbf{K} = \{k_j\}_{j=1}^M$  نمایش داده می‌شود که هر  $k_i \in \mathbb{R}^{D_k}$  است. استخراج‌کننده‌ی ویژگی به صورت  $q: \mathbb{R}^{H \times W \times C} \rightarrow \mathbb{R}^{D_k}$  معرفی می‌شود که بردار ویژگی  $x$  را به همان بعد کلیدها نگاشت می‌کند. در واقع از مدل ازپیش‌آموزش‌دیده به‌عنوان استخراج‌کننده ویژگی منجمد<sup>۱۶</sup> استفاده می‌شود:  $q(x) = f(x)[0, :]$  (که در آن از بردار ویژگی متناظر با کلاس استفاده می‌شود). تابع  $\gamma: \mathbb{R}^{D_k} \times \mathbb{R}^{D_k} \rightarrow \mathbb{R}$  به‌عنوان معیاری برای سنجش میزان تطابق بین بردار ویژگی ورودی و کلید پرامپت تعریف می‌شود (مانند فاصله کسینوسی). برای یک ورودی  $x$ ، از تابع  $q(x)$  استفاده می‌کنیم تا بهترین  $N$  کلید انتخاب شوند. در نهایت، انتخاب کلید طبق (۳-۶) صورت می‌گیرد.

$$\mathbf{K}_x = \arg \min_{\{s_i\}_{i=1}^N \subseteq [1, M]} \sum_{i=1}^N \gamma(q(x), \mathbf{k}_{s_i}), \quad (۳-۶)$$

$\{s_i\}_{i=1}^N$  به‌عنوان یک زیرمجموعه از  $N$  کلید از بازه  $[1, M]$  در نظر گرفته می‌شود و  $\mathbf{K}_x$  نشان‌دهنده یک زیرمجموعه از بهترین  $N$  کلید انتخاب‌شده از  $\mathbf{K}$  به‌طور خاص برای نمونه  $x$  است. علاوه بر این، به روش انتخاب پرامپت، قابلیت اضافه‌ای نیز اضافه شده است به این صورت که برای پرامپت‌های قبلاً به‌روزرسانی شده، جریمه در نظر گرفته است تا متناسب با تعداد تکرارشان در وظایف قبلی، جریمه‌ی بیشتری برای انتخاب بگیرند. در این حالت، پرامپت‌های با تکرار کم نیز شانس انتخاب شدن پیدا می‌کنند و پرامپت‌ها با تکرار بیشتر، کمتر تغییر می‌کنند و تداخل کمتر می‌شود.

<sup>14</sup>Prompt selection<sup>15</sup>Embedding size<sup>16</sup>Frozen

## ۲-۴-۳ یادگیری پرامپت

در بخش یادگیری پرامپت، پرامپت‌های انتخاب‌شده به‌همراه داده‌ی ورودی به مدل تغذیه شده و پس از عبور از لایه‌های ترنسفورمر، بخش خروجی مربوط به پرامپت‌ها، استخراج شده و با میانگین‌گیری، به دسته‌بند<sup>۱۷</sup> منتقل می‌شود. سپس با انجام عملیات پس‌انتشار<sup>۱۸</sup>، وزن‌های پرامپت‌ها و کلیدهای متناظر آن‌ها به‌روزرسانی می‌شوند. این فرآیند باعث می‌شود مدل ضمن یادگیری وظایف جدید، قابلیت تعمیم خود را افزایش داده و دانش قبلی را حفظ کند (مطابق با شکل ۲-۳). تابع زیان در این مدل مطابق با (۷-۳)، به صورت ترکیبی معرفی می‌شود به این صورت که بخش اول شامل زیان بین برچسب تصویر و پیش‌بینی دسته‌بند از ورودی دارای پرامپت و بخش دوم شامل تفاوت بین کلیدهای انتخاب شده و ویژگی استخراجی از ورودی می‌باشد. به عبارتی تلاش بر این است که علاوه بر تقویت پرامپت‌ها، کلیدهای مرتبط نیز، به ویژگی ورودی متناظرشان نزدیک‌تر شوند.

$$\min_{\mathbf{P}, \mathbf{K}, \phi} \mathcal{L}(g_{\phi}(f_r^{\text{avg}}(x_p)), y) + \lambda \sum_{\mathbf{K}_x} \gamma(q(x), \mathbf{k}_{s_i}) \quad (7-3)$$

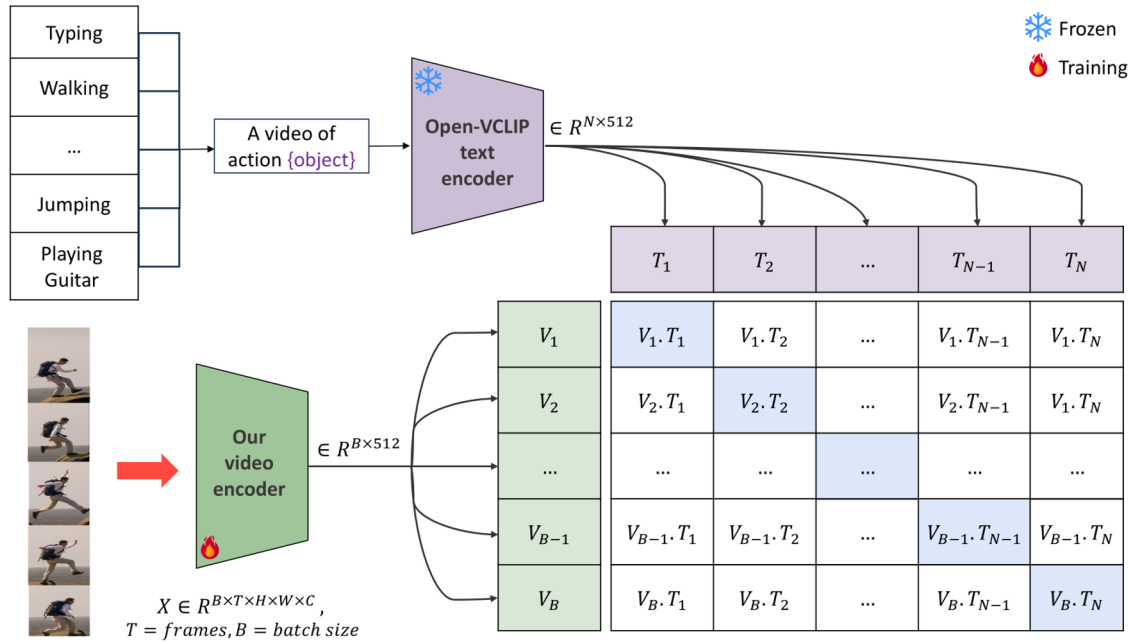
همانطور که دیده می‌شود، بخش اول شامل تابع زیان (کراس انتروپی) بین خروجی دسته‌بند  $g_{\phi}$  و برچسب واقعی  $y$  محاسبه می‌شود. تابع  $f_r^{\text{avg}} = \text{AvgPool}(f_r(x_p)[0 : NL_p, :])$  به این معناست که بردارهای پنهان خروجی متناظر با مکان‌های  $N \cdot L_p$  پرامپت‌ها، پیش از ورود به لایه‌ی دسته‌بند، به‌صورت میانگین‌گیری شده ترکیب می‌شوند. در عبارت  $f_r(x_p)$ ، تابع  $f_r$  نشان‌دهنده بخش باقی‌مانده از مدل از پیش‌آموزش دیده است که پس از الحاق پرامپت‌ها به ورودی، عملیات پردازش ویژگی را انجام می‌دهد. بردار  $x_p$  نیز از اتصال پرامپت‌ها به بردار تعبیه‌شده ورودی حاصل می‌شود. در بخش دوم، پارامتر  $\lambda$  وزن اهمیت عبارت تنظیم‌کننده را مشخص می‌کند. فاصله بین بردار  $q(x)$  و کلیدهای انتخاب‌شده  $\mathbf{k}_{s_i}$  را کاهش می‌دهد. مجموعه  $\mathbf{K}_x$  نیز نشان‌دهنده کلیدهای انتخاب‌شده برای ورودی  $x$  است که طبق معادله (۶-۳) به دست می‌آیند.

در ادامه به توضیح مدل پیشنهادی و اجزای آن پرداخته می‌شود.

<sup>17</sup>Classifier

<sup>18</sup>Backpropagation





شکل ۳-۳: طرح کلی مرحله‌ی آموزش مدل ProActionCLIP

### ۵-۳ مرحله‌ی آموزش ProActionCLIP

همانطور که پیش‌تر اشاره شد، در مرحله‌ی آموزش، تمرکز بر یادگیری پرامپت‌های مناسب برای دسته‌های مختلفی است که به صورت پیوسته به مدل اضافه می‌شوند. طرح کلی مدل در مرحله‌ی آموزش در شکل ۳-۳، نشان داده شده است که الگو گرفته از مدل CLIP می‌باشد. این مدل شامل یک کدگذار ویدیو (قابل به‌روزرسانی) و کدگذار متن منجمد (غیر قابل به‌روزرسانی) از مدل Open-VCLIP است. به این صورت که بردار ویژگی ویدیوی استخراج شده از کدگذار ویدیو و بردارهای ویژگی برچسب‌های موجود استخراج شده از کدگذار متن، طبق روش تقابلی مقایسه شده و طبق نزدیک شدن موارد متناظر و دور شدن موارد نامتناظر، وزن‌های پرامپت‌ها تغییر داده می‌شوند. دو بخش اصلی مدل به نام کدگذار ویدیو و یادگیری پرامپت در ادامه توضیح داده خواهند شد.

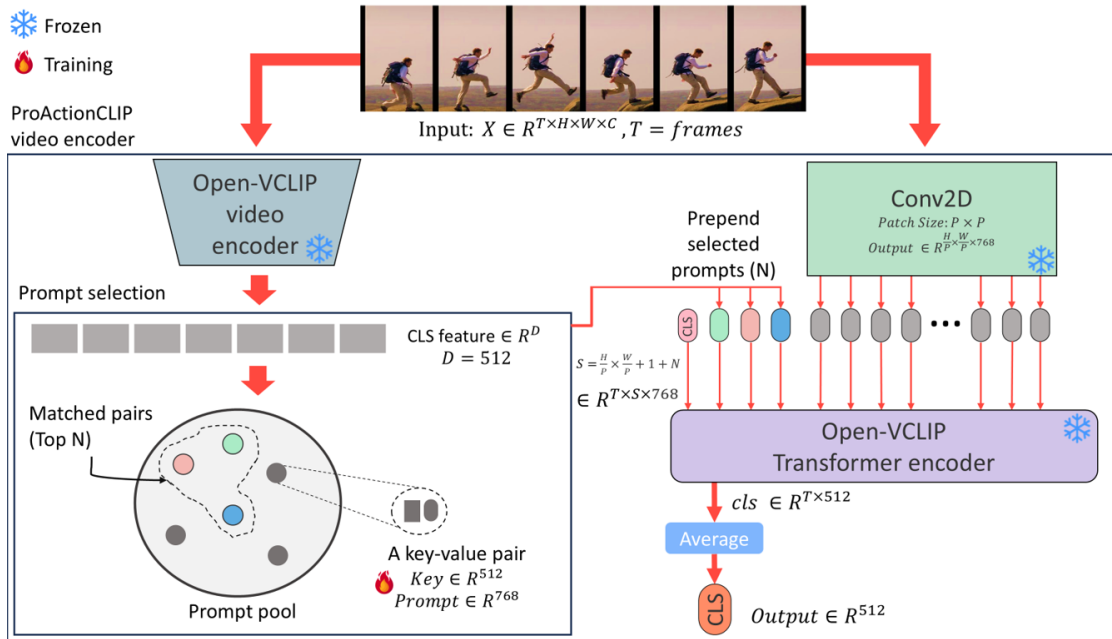
#### ۱-۵-۳ کدگذار ویدیو

این بخش، از ترکیب L2P و Open-VCLIP تشکیل شده است. به طور کلی مطابق شکل ۴-۳، ویدیو به عنوان ورودی، وارد کدگذار Open-VCLIP و لایه‌ی کانولوشنی دوبعدی می‌شود. ویژگی کلاس<sup>۱۹</sup> از خروجی کدگذار Open-VCLIP با ۵۱۲ بعد، با کلیدهای داخل استخر پرامپت مقایسه می‌شود. به تعداد

<sup>19</sup>Class feature (CLS)

$N$  پرامپت از مشابه‌ترین کلیدها انتخاب می‌شوند. از طرف دیگر ویدیو از لایه‌ی کانولوشنی عبور کرده و پرامپت‌ها به هر قاب، به صورت جداگانه متصل می‌شوند. سپس به کدگذار ترنسفورمر معرفی شده در Open-VCLIP، وارد شده و به ازای هر قاب، یک ویژگی کلاس بدست می‌آید که میانگین آن‌ها محاسبه و به عنوان خروجی نهایی این بخش، ارائه می‌گردد. در مدل پیشنهادی، صرفاً پرامپت‌ها و کلیدهای متناظر آن‌ها، قابل یادگیری هستند و بقیه‌ی اجزا به صورت منجمد استفاده می‌شوند. در این بخش از تحقیق، آزمایش‌های مختلفی اجرا شد که بر اساس نوع انتخاب پرامپت و شرایط استخراج پرامپت می‌توان به دسته‌های زیر تقسیم نمود:

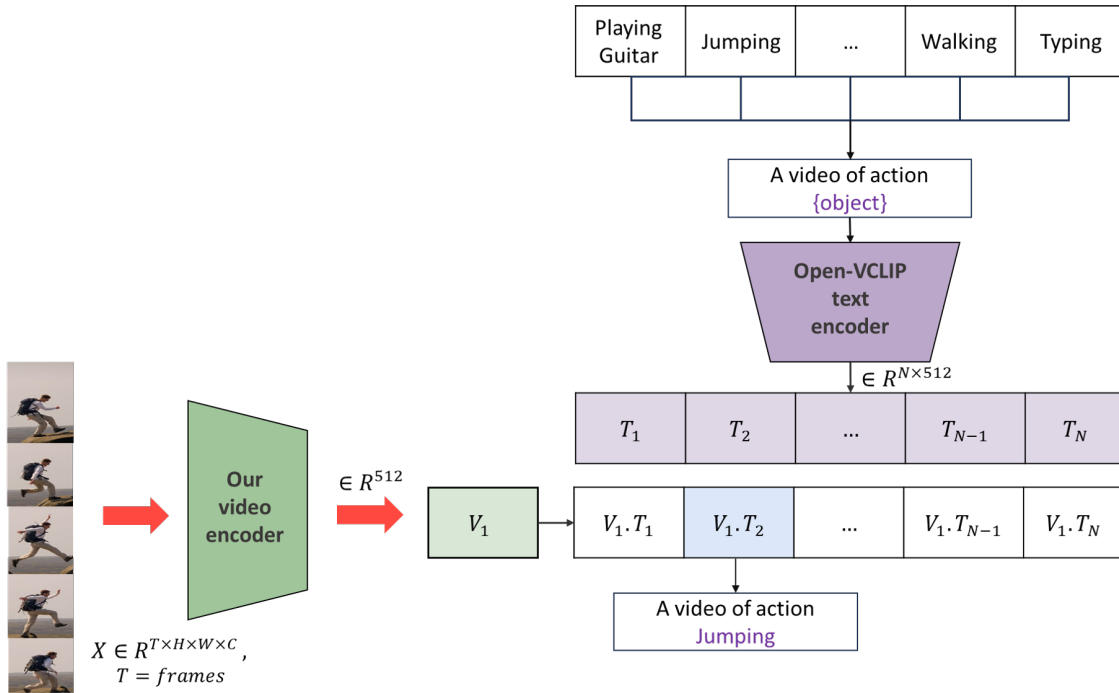
- **مقداردهی اولیه‌ی کلید پرامپت:** از آن جایی که ابتدای آموزش، مقادیر اولیه به صورت تصادفی هستند، در این آزمایش، مقادیر کلیدها، معادل ویژگی‌های برچسب دسته‌های استخراج شده از کدگذار متن قرار داده شد. در این صورت، مقایسه‌ی ویژگی ویدیو و کلیدها به صورت بهینه‌تر و دقیق‌تری صورت می‌گیرد.
  - **وزن دهی به کلید پرامپت‌های از قبل انتخاب شده:** مطابق با روش اضافه‌ای که برای انتخاب پرامپت در L2P مطرح شد، تعداد تکرار پرامپت‌ها در هر وظیفه محاسبه می‌شود. در وظیفه‌ی بعدی، هرچه تکرار پرامپت بیشتر بوده باشد، تأثیرش در انتخاب کمتر می‌شود.
  - **منجمد کردن پرامپت‌های قبلی:** یکی از راه‌های استفاده از وزن‌های قبلی، این است که به صورت منجمد استفاده شوند و به‌روزرسانی نشوند. این روش کمک می‌کند اطلاعات اختصاصی هر وظیفه از بین نرود و اگر داده‌ی جدید اشتراکی با قبلی‌ها داشته باشد، پرامپت آن‌ها را انتخاب خواهد کرد.
  - **پویا بودن تعداد پرامپت‌های استخراج پرامپت:** در فرض اولیه، در استخراج پرامپت تعدادی ثابت پرامپت وجود داشت اما برای بهینه‌بودن مدل برای دسته‌های بیشتر و موجود بودن پرامپت کافی در هر وظیفه، در این قسمت پرامپت‌ها در ابتدای هر وظیفه افزایش می‌یابد.
- ترکیب برخی از این روش‌ها نیز آزمایش شده است مانند استفاده از استخراج پویا و مقداردهی اولیه کلیدها.



شکل ۳-۴: سازوکار بخش کدگذار ویدیو مدل ProActionCLIP

### ۳-۵-۲ یادگیری پرامپت

وقتی ویدیو از کدگذار ویدیوی پیشنهادی عبور کرد، وارد مرحله‌ی نهایی برای به‌روزرسانی وزن‌های پرامپت‌ها و کلیدهای متناظرشان می‌شود. مطابق با شکل ۳-۳، ویژگی برچسب‌ها از طریق کدگذار متن مدل Open-VCLIP بدست می‌آید. تابع زیان همانند روش L2P، شامل دو بخش است. اولین بخش شامل زیان بین ویژگی استخراجی از ویدیو و ویژگی برچسب متناظر آن و بخش بعدی مختص زیان بین ویدیو و کلیدهای انتخاب شده برای آن می‌باشد. در این صورت پرامپت‌ها در بخش اول و کلیدها در بخش دوم تابع زیان مورد تمرکز قرار می‌گیرند. تابع زیان مدل پیشنهادی با L2P تفاوت‌هایی دارد که در ادامه بررسی می‌کنیم. روش مدل پیشنهادی برگرفته از مدل CLIP، برخلاف L2P که با دسته‌بند است، به صورت تقابلی می‌باشد. مطابق (۳-۸)، پرامپت‌های انتخابی، به ورودی عبور کرده از لایه‌ی کانولوشنی، ملحق شده و  $x_{p,t}$  را تشکیل می‌دهند.  $t$  نشان‌دهنده‌ی هر قاب است. سپس از لایه‌ی ترنسفورمر معرفی شده، عبور کرده و به صورت  $z(x_{p,t})$  نمایش داده می‌شود. خروجی ترنسفورمر، در بردارهای ویژگی برچسب‌ها که از کدگذار متن بدست آمده ( $W$ )، ضرب می‌شود. از  $\tau$  نیز به عنوان ضریب مقیاس‌دهی در این ضرب استفاده می‌شود. در این حالت شباهت ویژگی بدست آمده از قاب  $t$  از ویدیو، با برچسب‌ها سنجیده می‌شود. سپس میانگین شباهت‌ها برای قاب‌های ویدیو ( $T$ ) محاسبه می‌شود. تابع زیان کراس انتروپی بین خروجی و برچسب متناظر اجرا می‌شود. بخش دوم، مانند روش L2P، بر نزدیک کردن کلیدهای انتخاب شده، به ویژگی کلاس ویدیوی نمونه، سعی دارد که در بخش قبل به تفصیل شرح داده



شکل ۳-۵: طرح کلی مرحله‌ی آزمون مدل ProActionCLIP

شد.

$$\mathcal{L} = \text{CE} \left( \frac{1}{T} \sum_{t=1}^T [\tau \cdot \mathbf{z}(\mathbf{x}_{p,t}) \mathbf{W}^T], y \right) + \lambda \sum_{\mathbf{K}_x} \gamma(q(x), \mathbf{k}_{s_i}) \quad (۸-۳)$$

## ۳-۶ مرحله‌ی آزمون ProActionCLIP

برای آزمون مدل، مطابق شکل ۳-۵، ویدیو وارد کدگذار ویدیو می‌شود و به تعداد  $N$  نزدیکترین کلید به ویژگی ویدیو، را انتخاب کرده و به هر یک از قاب‌های ویدیو ملحق کرده و از ترنسفورمر عبور می‌دهد. خروجی نهایی این بخش را با ویژگی استخراج شده‌ی برچسب‌ها از کدگذار متن، مقایسه کرده و برچسب با بیشترین شباهت انتخاب می‌شود.

## ۳-۷ جمع‌بندی

در این فصل مدل نهایی پیشنهادی این پژوهش با نام ProActionCLIP معرفی شد که ترکیبی از دو مدل L2P و Open-VCLIP بوده که با تلفیق قابلیت یادگیری پرامپت در L2P و توانایی استخراج ویژگی‌های ویدیویی در Open-VCLIP طراحی شده است. در این فصل، ساختار کلی مدل پیشنهادی و اجزای

اصلی آن شامل کدگذار ویدیو، کدگذار متن منجمد و سازوکار یادگیری پرامپت تشریح شد. تکنیک‌های مختلف استفاده شده در طراحی استخر پرامپت‌ها و روش انتخاب پرامپت‌های مناسب برای هر ورودی، نیز شرح داده شد. همچنین نحوه تعامل این اجزا با یکدیگر برای یادگیری پیوسته و به‌روزرسانی وزن‌های پرامپت‌ها از طریق الگوریتم پسانتشار توضیح داده شد. در نهایت، فرآیند آموزش مدل و محاسبه تابع زیان که شامل عبارت منظم‌سازی (برای نزدیک کردن کلیدها و ویژگی ویدیو) و معیار تقابلی است، مورد بررسی قرار گرفت.

## فصل چهارم

### مشخصات یک پایان نامه و گزارش علمی

## ۴-۱ مقدمه

در این فصل، عملکرد مدل ProActionCLIP در زمینه یادگیری پیوسته تشخیص حرکت انسان مورد ارزیابی قرار می‌گیرد. هدف از این ارزیابی، بررسی میزان صحت مدل در یادگیری وظایف جدید، ارزیابی میزان فراموشی دانش پیشین و تحلیل بهره‌وری محاسباتی آن از نظر مصرف سخت‌افزار و تعداد پارامترهای قابل‌آموزش است. به منظور ارزیابی جامع، مدل پیشنهادی با روش‌های مطرح در این حوزه مانند PIVOT [۳۸] و سایر رویکردهای مرجع مقایسه می‌شود. برای این منظور، از مجموعه داده‌های نظیر UCF101 [۵۵] و HMDB51 [۵۶] استفاده شده است.

در ادامه‌ی این فصل، ابتدا معیارهای ارزیابی شامل صحت، میزان فراموشی معرفی می‌شوند. سپس، جزئیات مربوط به مجموعه‌های داده، تنظیمات آزمایشی، نتایج و مقایسه ارائه خواهد شد تا ارزیابی مدل پیشنهادی به‌طور کامل و شفاف صورت گیرد. در آخر نیز پیچیدگی محاسباتی بررسی خواهد شد.

## ۴-۲ معیارهای ارزیابی

در یادگیری پیوسته، دو معیار ارزیابی اهمیت دارند. یکی از آن‌ها میانگین صحت وظایف با وجود یادگیری سایر وظایف بوده و دیگری میزان فراموشی مدل پس از یادگیری هر وظیفه است که در ادامه هر یک شرح داده خواهد شد.

### ۴-۲-۱ میانگین صحت

در یادگیری پیوسته، عملکرد مدل روی تمام وظایف یادگرفته شده تاکنون توسط میانگین صحت ارزیابی می‌گردد. دو نوع صحت  $top1$  و  $top5$  در این تحقیق بررسی می‌شود. صحت  $top1$  بیان‌گر این است که پیش‌بینی‌کننده، از بین احتمالات بدست آمده، برچسب با بیشترین احتمال را انتخاب می‌کند و سپس برابر بودن آن با برچسب واقعی داده‌ی بررسی‌شده، سنجیده می‌شود. اما در صحت  $top5$ ، پنج تا برچسب با بیشترین احتمال، انتخاب شده و وجود یا عدم وجود برچسب واقعی در این پنج برچسب بررسی می‌شود. مطابق با (۴-۱) و (۴-۲)، میانگین صحت در وظیفه‌ی فعلی، برای همین وظیفه و وظایف قبلی، در حالت  $top1$  و  $top5$  بدست می‌آید.  $ACC_{i,t}^{top1}$  نشانگر صحت  $top1$  برای وظیفه‌ی  $i$  بعد از یادگیری وظیفه‌ی  $t$

و  $ACC_{i,t}^{top5}$  نشانگر صحت  $top5$  برای وظیفه‌ی  $i$  بعد از یادگیری وظیفه‌ی  $t$  می‌باشد.

$$A_{top1}(t) = \frac{1}{t+1} \sum_{i=0}^t ACC_{i,t}^{top1} \quad (۱-۴)$$

$$A_{top5}(t) = \frac{1}{t+1} \sum_{i=0}^t ACC_{i,t}^{top5} \quad (۲-۴)$$

## ۲-۲-۴ میزان فراموشی

در یادگیری پیوسته، مدل باید بتواند وظایف جدید را یاد بگیرد بدون اینکه دانش وظایف قبلی را فراموش کند. اما معمولاً پدیده‌ی فراموشی فاجعه‌آمیز رخ می‌دهد؛ یعنی مدل پس از یادگیری وظایف جدید، صحت آن روی وظایف قدیمی کاهش پیدا می‌کند. معیار فراموشی برای اندازه‌گیری میزان این افت عملکرد تعریف می‌شود و به صورت میانگین کاهش دقت در وظایف قبلی است. بر اساس (۳-۴)، فراموشی مدل روی وظیفه‌ی  $i$  پس از یادگیری وظیفه‌ی  $t$  بدست می‌آید. ماتریس  $ACC$ ، شامل صحت مدل روی هر وظیفه پس از یادگیری همان وظیفه و وظیفه‌های دیگر است. به دنبال آن،  $ACC_{i,k}$  نشان‌دهنده‌ی صحت مدل روی وظیفه‌ی  $i$  پس از یادگیری وظیفه‌ی  $k$  است. به این ترتیب، اختلاف بین بیشترین صحتی که وظیفه‌ی  $i$  پس از یادگیری وظایف مختلف بدست آورده و صحتی که پس از وظیفه‌ی  $t$  (آخرین وظیفه‌ی یادگرفته شده) بدست آمده، در  $f_i(t)$  قرار می‌گیرد. در نهایت فراموشی برای هر وظیفه محاسبه شده و میانگین آن‌ها به عنوان فراموشی مدل پس از یادگیری وظیفه‌ی  $t$ ، در نظر گرفته می‌شود ((۴-۴)).

$$f_i(t) = \max_{k \leq t} ACC_{i,k} - ACC_{i,t} \quad (۳-۴)$$

$$F(t) = \frac{1}{t} \sum_{i=1}^t f_i(t) \quad (۴-۴)$$





شکل ۴-۱: نمونه‌ای از مجموعه داده‌ی UCF101 [۵۵]

## ۳-۴ مجموعه داده

در این تحقیق از دو مجموعه داده‌ی مرسوم تشخیص حرکت با نام‌های UCF101 [۵۵] و HMDB51 [۵۶] استفاده شده است که در ادامه هریک شرح داده خواهد شد.

### ۱-۳-۴ مجموعه داده‌ی UCF101

مجموعه داده‌ی UCF101 [۵۵] یکی از مجموعه داده‌های استاندارد و پرکاربرد در زمینه تشخیص حرکات انسانی در ویدیو است. این مجموعه شامل 13,320 ویدیو از 101 دسته‌ی مختلف فعالیت انسانی است که طیف وسیعی از حرکات روزمره، ورزشی و تعاملی را پوشش می‌دهد. نمونه‌های این مجموعه داده از ویدیوهای واقعی و متنوع جمع‌آوری شده‌اند و شامل تغییرات قابل توجه در شرایط نور، پس‌زمینه، زاویه دید و ظاهر اشخاص هستند. دسته‌های موجود در UCF101 به پنج گروه کلی تقسیم می‌شوند:

۱. فعالیت‌های ورزشی

۲. فعالیت‌های تعاملی انسان با اشیا

۳. فعالیت‌های تعاملی انسان با انسان

۴. فعالیت‌های بدنی عمومی

۵. فعالیت‌های نواختن موسیقی

ویژگی مهم این مجموعه داده، تنوع بالای آن در شرایط تصویربرداری و پیچیدگی حرکات است که آن را به یک معیار معتبر برای ارزیابی مدل‌های تشخیص حرکات‌های انسانی و یادگیری پیوسته تبدیل می‌کند. نمونه‌هایی از داده‌های این مجموعه داده در شکل ۴-۱ قابل مشاهده است.



شکل ۴-۲: نمونه‌ای از مجموعه داده‌ی HMDB51 [۵۶]

#### ۲-۳-۴ مجموعه داده‌ی HMDB51

مجموعه داده HMDB51 [۵۶] یکی از مجموعه داده‌های پرکاربرد در زمینه‌ی تشخیص حرکات انسانی در ویدیو است که برای ارزیابی عملکرد الگوریتم‌های تشخیص حرکت طراحی شده است. این مجموعه شامل 6,766 ویدیو در 51 دسته‌ی فعالیت انسانی است که هر دسته تقریباً ۱۰۰ نمونه دارد. نمونه‌های موجود در HMDB51 از منابع متنوعی همچون فیلم‌های سینمایی، ویدیوهای اینترنتی و فیلم‌های خانگی جمع‌آوری شده‌اند و طیف وسیعی از حرکات انسانی شامل فعالیت‌های بدنی، تعامل انسان با اشیاء و تعامل انسان با انسان را پوشش می‌دهند. یکی از ویژگی‌های مهم این مجموعه داده، تنوع بالا در صحنه، پس‌زمینه، زاویه دوربین و کیفیت ویدیوها است که تشخیص حرکات را به یک چالش واقعی تبدیل می‌کند. به دلیل اندازه متوسط و تنوع مناسب، HMDB51 به‌طور گسترده برای آموزش و ارزیابی مدل‌های تشخیص حرکات و یادگیری پیوسته مورد استفاده قرار می‌گیرد. فعالیت‌های موجود در HMDB51 را می‌توان در پنج گروه کلی دسته‌بندی کرد:

##### ۱. فعالیت‌های عمومی صورت

##### ۲. فعالیت‌های صورت همراه با تعامل با اشیاء

##### ۳. حرکات عمومی بدن

##### ۴. حرکات بدن همراه با تعامل با اشیاء

## ۵. حرکات بدن در تعامل انسان با انسان

نمونه‌ای از این داده‌ها در شکل ۲-۴ قابل مشاهده است.

## ۴-۴ تنظیمات آزمایش

آزمایش‌ها بر روی دو مجموعه داده‌ی HMDB51 [۵۶] و UCF101 [۵۵] به صورت جداگانه اجرا شده و در هر دو حالت، مدل پایه‌ی مورد استفاده، Open-VCLIP [۲۵] می‌باشد که در فرآیند آموزش از وزن‌های پیش‌آمورخته‌ی CLIP [۱۸] (نسخه‌ی ViT-B/16) بهره برده است. آموزش با نرخ اولیه‌ی یادگیری  $3.33 \times 10^{-6}$  (مقدار معین شده در تنظیمات Open-VCLIP) تست گردید اما به علت یادگیری آهسته و دریافت نتیجه‌ی نامطلوب، نرخ اولیه‌ی یادگیری، بعد از آزمایش‌های مختلف، با مقدار  $1.5 \times 10^{-3}$  برای مجموعه داده‌ی UCF101 و  $1.5 \times 10^{-2}$  برای مجموعه داده‌ی HMDB51 مورد استفاده قرار گرفت. مجموعه داده‌ها به نسبت‌های 60% برای آموزش، 30% برای آزمون و 10% برای اعتبارسنجی تقسیم شدند. تعداد وظیفه برای هر اجرا، 10 در نظر گرفته شد. تنظیمات مربوط به پرامپت، شامل طول هر پرامپت، مقدار بیشینه تعداد پرامپت انتخابی در هر اجرا، به ترتیب ۵ (طبق مقداردهی L2P [۳۱]) و ۲ در نظر گرفته شده است. طول استخر پرامپت متناسب با سناریوی استفاده شده، متفاوت خواهد بود. سایر تنظیمات، مانند تنظیمات معرفی شده در Open-VCLIP، در نظر گرفته شده‌اند. در ادامه تنظیمات اجرا برای هر دو مجموعه داده‌ی مذکور بررسی خواهند شد. سخت افزار استفاده شده در این تحقیق، GPU L40S با 48 گیگابایت حافظه‌ی رم می‌باشد.

## ۱-۴-۴ آزمایش با مجموعه داده‌ی UCF101

در این آزمایش ابتدا تعداد 5 ایپاک برای هر وظیفه در نظر گرفته شد و پس از بررسی نمودار صحت بر حسب ایپاک طی شده، مشاهده شد که صحت وظایف پس از سه ایپاک چه در اعتبارسنجی و چه در آموزش ثابت می‌شوند. بنابراین ایپاک نهایی برای این مجموعه داده، 3 در نظر گرفته شد. آزمایش این مجموعه داده در سه سناریو بررسی شد که در ادامه توضیح داده خواهد شد:

۱. استخر ثابت با جریمه‌ی وزن‌های پیشین: در این حالت طول استخر پرامپت ثابت خواهد بود که در این آزمایش، به اندازه‌ی 202 مقداردهی شده است. تعداد تکرار انتخاب هر پرامپت در یادگیری پیوسته، ذخیره می‌شود و هنگام انتخاب پرامپت در وظیفه‌ی جدید، متناسب با تعداد

تکرار هر پرامپت، جریمه‌ای در نظر گرفته می‌شود. بنابراین احتمال انتخاب پرامپت‌های پیشین، تا حدودی کاهش می‌یابد (الگو گرفته از روش اضافی ذکر شده در L2P [۳۱]).

۲. **استخر پویا با مقداردهی تصادفی:** در این حالت طول استخر پرامپت اولیه به اندازه‌ی 20 در نظر گرفته شده است. هر وظیفه که اضافه می‌شود، به طول استخر 20 واحد اضافه می‌شود. نکته‌ی حائز اهمیت در سناریوی فعلی و بعدی، این است که پرامپت‌های استفاده شده در وظایف قبلی، هنگام یادگیری وظیفه‌ی جدید، منجمد شده و صرفاً قابلیت انتخاب شدن، دارند و تغییری نخواهند کرد.

۳. **استخر پویا با مقداردهی مبتنی بر کدگذار متن CLIP:** در این حالت، تمام تنظیمات مدل مشابه سناریوی قبلی است و تنها تفاوت، در نحوه‌ی مقداردهی اولیه‌ی کلیدهای استخر پرامپت است. در ابتدا، به ازای هر دسته در هر وظیفه، به تعداد از پیش تعیین شده کلید، مقداردهی اولیه می‌شود؛ به این صورت که برچسب هر دسته به همراه عبارت‌های معنادار ثابت، از کدگذار متن مدل CLIP عبور کرده و ویژگی‌های استخراج شده در کلیدها قرار می‌گیرند. به عنوان مثال اگر برچسب یک ویدیو، jumping person باشد، عبارتی که وارد کدگذار می‌شود برابر با "a video of jumping person." خواهد بود. در این سناریو، مقادیر اولیه‌ی کلیدها معنادار بوده و نتایج نشان داده اند که تاثیر مثبتی در انتخاب درست پرامپت‌ها داشته‌اند. در آزمایش اولیه، تعداد کلیدهای مقداردهی اولیه برای هر دسته از هر وظیفه، برابر با 2 در نظر گرفته شد. در حالت نهایی تست شده، برای بهبود انتخاب پرامپت‌ها، این تعداد به 5 کلید برای هر دسته افزایش یافت که این تغییر باعث شد که فضای انتخاب پرامپت‌ها غنی تر شود و مدل در انتخاب پرامپت‌های مناسب عملکرد بهتری داشته باشد.

## ۴-۴-۲ آزمایش با مجموعه داده‌ی HMDB51

یش ابتدا تعداد 5 ایپاک برای هر وظیفه در نظر گرفته شد و پس از بررسی نمودار صحت بر حسب ایپاک طی شده، مشاهده شد که یادگیری به خوبی صورت نگرفته است. بنابراین پس از افزایش ایپاک‌ها در طی آزمایش، ایپاک نهایی برای این مجموعه داده، 7 در نظر گرفته شد. مانند مجموعه داده‌ی پیشین، آزمایش این مجموعه داده نیز در سه سناریو بررسی شد که در ادامه توضیح داده خواهد شد:

۱. **استخر ثابت با جریمه‌ی وزن‌های پیشین:** این سناریو نیز مانند توضیحات ذکر شده در قسمت قبل، اجرا شده است با این تفاوت که طول استخر پرامپت در این جا 102 در نظر گرفته شده است.

۲. **استخر پویا با مقداردهی تصادفی:** در این حالت طول استخر پرامپت اولیه ابتدا به اندازه‌ی 10 در نظر گرفته شد و سپس به علت کسب نتیجه‌ی بهتر، به 25 تغییر داده شد. هر وظیفه که اضافه می‌شود، به طول استخر 25 واحد اضافه می‌شود. نکته‌ی حائز اهمیت در سناریوی فعلی و بعدی، این است که پرامپت‌های استفاده شده در وظایف قبلی، هنگام یادگیری وظیفه‌ی جدید، منجمد شده و صرفاً قابلیت انتخاب شدن، دارند و تغییری نخواهند کرد.

۳. **استخر پویا با مقداردهی مبتنی بر کدگذار متن CLIP:** همانند آن‌چه در قسمت مجموعه‌داده‌ی UCF101 گفته شد، تنظیمات مانند سناریوی پیشین باقی می‌مانند و صرفاً مقداردهی اولیه‌ی کلیدها با کدگذار CLIP صورت می‌گیرد. در این جا نیز ابتدا تعداد کلیدهای مقداردهی اولیه برای هر دسته از هر وظیفه، برابر با 2 در نظر گرفته شد و در نهایت، این تعداد به 5 کلید برای هر دسته افزایش یافت. به عبارتی، طول اولیه‌ی استخر پرامپت از 10 به 25 تغییر داده شد.

## ۴-۵ ارزیابی نتایج

## فصل پنجم

### جمع‌بندی و نتیجه‌گیری و پیشنهادات

در پایان گزارش‌های علمی و فنی لازم است که جمع‌بندی یا نتیجه‌گیری نهایی ارائه شود. در این موارد می‌توان آخرین فصل پایان نامه که پیش از مراجع قرار می‌گیرد را به این امر اختصاص داد.

## ۵-۱ پیشنهادات

در این بخش پیشنهاداتی که محقق جهت ادامه تحقیقات دارد ارائه می‌گردد. دقت شود که پیشنهادات باید از تحقیق انجام شده و نتایج آن حاصل شده باشد و از ذکر جملات کلی باید پرهیز کرد.

## منابع و مراجع

- [1] Wang, Liyuan, Zhang, Xingxing, Su, Hang, and Zhu, Jun. A comprehensive survey of continual learning: Theory, method and application. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2024.
- [2] Mai, Zheda, Li, Ruiwen, Jeong, Jihwan, Quispe, David, Kim, Hyunwoo, and Sanner, Scott. Online continual learning in image classification: An empirical survey. Neurocomputing, 469:28–51, 2022.
- [3] van de Ven, Gido M. and Tolias, Andreas S. Three scenarios for continual learning, 2019.
- [4] Churamani, Nikhil, Kara, Ozgur, and Gunes, Hatice. Domain-Incremental Continual Learning for Mitigating Bias in Facial Expression and Action Unit Recognition . IEEE Transactions on Affective Computing, 14(04):3191–3206, October 2023.
- [5] Ma, Jiawei, Tao, Xiaoyu, Ma, Jianxing, Hong, Xiaopeng, and Gong, Yihong. Class incremental learning for video action classification. in 2021 IEEE International Conference on Image Processing (ICIP), pp. 504–508, 2021.
- [6] Park, Jaeyoo, Kang, Minsoo, and Han, Bohyung. Class-incremental learning for action recognition in videos. in 2021 IEEE/CVF International Conference on Computer Vision (ICCV), pp. 13678–13687, 2021.
- [7] Kirkpatrick, James, Pascanu, Razvan, Rabinowitz, Neil, Veness, Joel, Desjardins, Guillaume, Rusu, Andrei A., Milan, Kieran, Quan, John, Ramalho, Tiago, Grabska-



- Barwinska, Agnieszka, Hassabis, Demis, Clopath, Claudia, Kumaran, Dharshan, and Hadsell, Raia. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13):3521–3526, 2017.
- [8] Li, Zhizhong and Hoiem, Derek. Learning without forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40:2935–2947, 2016.
- [9] Shin, Hanul, Lee, Jung Kwon, Kim, Jaehong, and Kim, Jiwon. Continual learning with deep generative replay. in Guyon, I., Luxburg, U. Von, Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., eds. , *Advances in Neural Information Processing Systems*, vol. 30. Curran Associates, Inc., 2017.
- [10] Chaudhry, Arslan, Ranzato, Marc’Aurelio, Rohrbach, Marcus, and Elhoseiny, Mohamed. Efficient lifelong learning with a-gem. *ArXiv*, abs/1812.00420, 2018.
- [11] Rebuffi, Sylvestre-Alvise, Kolesnikov, Alexander, Sperl, Georg, and Lampert, Christoph H. icarl: Incremental classifier and representation learning. in 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5533–5542, 2017.
- [12] Aljundi, Rahaf, Caccia, Lucas, Belilovsky, Eugene, Caccia, Massimo, Lin, Min, Charlin, Laurent, and Tuytelaars, Tinne. Online continual learning with maximally interfered retrieval. *ArXiv*, abs/1908.04742, 2019.
- [13] Minhas, Rashid, Mohammed, Abdul Adeel, and Wu, Q. M. Jonathan. Incremental learning in human action recognition based on snippets. *IEEE Transactions on Circuits and Systems for Video Technology*, 22(11):1529–1541, 2012.
- [14] Li, Tianjiao, Ke, Qiuhong, Rahmani, Hossein, Ho, Rui En, Ding, Henghui, and Liu, Jun. Else-net: Elastic semantic network for continual action recognition from skeleton data. in 2021 IEEE/CVF International Conference on Computer Vision (ICCV), pp. 13414–13423, 2021.

- [15] Cheng, Jian, Liu, Haijun, Wang, Feng, Li, Hongsheng, and Zhu, Ce. Silhouette analysis for human action recognition based on supervised temporal t-sne and incremental learning. *IEEE Transactions on Image Processing*, 24(10):3203–3217, 2015.
- [16] Parisi, German I., Tani, Jun, Weber, Cornelius, and Wermter, Stefan. Lifelong learning of human actions with deep neural network self-organization. *Neural Networks*, 96:137–149, 2017.
- [17] Zhang, Jingyi, Huang, Jiaying, Jin, Sheng, and Lu, Shijian. Vision-language models for vision tasks: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(8):5625–5644, 2024.
- [18] Radford, Alec, Kim, Jong Wook, Hallacy, Chris, Ramesh, Aditya, Goh, Gabriel, Agarwal, Sandhini, Sastry, Girish, Askell, Amanda, Mishkin, Pamela, Clark, Jack, Krueger, Gretchen, and Sutskever, Ilya. Learning transferable visual models from natural language supervision. in *International Conference on Machine Learning*, 2021.
- [19] Singh, Amanpreet, Hu, Ronghang, Goswami, Vedanuj, Couairon, Guillaume, Galuba, Wojciech, Rohrbach, Marcus, and Kiela, Douwe. Flava: A foundational language and vision alignment model. in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15617–15629, 2022.
- [20] Li, Liunian Harold, Zhang, Pengchuan, Zhang, Haotian, Yang, Jianwei, Li, Chunyuan, Zhong, Yiwu, Wang, Lijuan, Yuan, Lu, Zhang, Lei, Hwang, Jenq-Neng, Chang, Kai-Wei, and Gao, Jianfeng. Grounded language-image pre-training. in *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10955–10965, 2022.
- [21] Wang, Mengmeng, Xing, Jiazheng, and Liu, Yong. Actionclip: A new paradigm for video action recognition. *ArXiv*, abs/2109.08472, 2021.

- [22] Zhou, Kaiyang, Yang, Jingkang, Loy, Chen Change, and Liu, Ziwei. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 130(9):2337–2348, July 2022.
- [23] Gao, Peng, Geng, Shijie, Zhang, Renrui, Ma, Teli, Fang, Rongyao, Zhang, Yongfeng, Li, Hongsheng, and Qiao, Yu. Clip-adapter: Better vision-language models with feature adapters. *International Journal of Computer Vision*, 132(2):581–595, Feb 2024.
- [24] Wortsman, Mitchell, Ilharco, Gabriel, Kim, Jong Wook, Li, Mike, Kornblith, Simon, Roelofs, Rebecca, Lopes, Raphael Gontijo, Hajishirzi, Hannaneh, Farhadi, Ali, Namkoong, Hongseok, and Schmidt, Ludwig. Robust fine-tuning of zero-shot models. in 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 7949–7961, 2022.
- [25] Weng, Zejia, Yang, Xitong, Li, Ang, Wu, Zuxuan, and Jiang, Yu-Gang. Open-vclip: Transforming clip to an open-vocabulary video model via interpolated weight optimization. in ICML, 2023.
- [26] Gu, Xiuye, Lin, Tsung-Yi, Kuo, Weicheng, and Cui, Yin. Open-vocabulary detection via vision and language knowledge distillation. *arXiv preprint arXiv:2104.13921*, 2021.
- [27] Lüddecke, Timo and Ecker, Alexander. Image segmentation using text and image prompts. in 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 7076–7086, 2022.
- [28] Zheng, Junhao, Qiu, Shengjie, Shi, Chengming, and Ma, Qianli. Towards lifelong learning of large language models: A survey. *ACM Comput. Surv.*, 57(8), March 2025.
- [29] Garg, Saurabh, Farajtabar, Mehrdad, Pouransari, Hadi, Vemulapalli, Raviteja, Mehta, Sachin, Tuzel, Oncel, Shankar, Vaishaal, and Faghri, Fartash. Tic-clip: Continual train-

- ing of clip models. in The Twelfth International Conference on Learning Representations (ICLR), 2024.
- [30] Lu, Shuyun, Jiao, Jian, Wang, Lanxiao, Qiu, Heqian, Lin, Xingtao, Mei, Hefei, and Li, Hongliang. Video class-incremental learning with clip based transformer. in 2024 IEEE International Conference on Image Processing (ICIP), pp. 500–506, 2024.
- [31] Wang, Zifeng, Zhang, Zizhao, Lee, Chen-Yu, Zhang, Han, Sun, Ruoxi, Ren, Xiaoqi, Su, Guolong, Perot, Vincent, Dy, Jennifer, and Pfister, Tomas. Learning to prompt for continual learning. in 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 139–149, 2022.
- [32] Menabue, Martin, Frascaroli, Emanuele, Boschini, Matteo, Sangineto, Enver, Bonicelli, Lorenzo, Porrello, Angelo, and Calderara, Simone. Semantic residual prompts for continual learning. in Leonardis, Aleš, Ricci, Elisa, Roth, Stefan, Russakovsky, Olga, Sattler, Torsten, and Varol, Gül, eds. , Computer Vision – ECCV 2024, pp. 1–18, Cham, 2025. Springer Nature Switzerland.
- [33] Wang, Zifeng, Zhang, Zizhao, Ebrahimi, Sayna, Sun, Ruoxi, Zhang, Han, Lee, Chen-Yu, Ren, Xiaoqi, Su, Guolong, Perot, Vincent, Dy, Jennifer, and Pfister, Tomas. Dual-prompt: Complementary prompting for rehearsal-free continual learning. in Computer Vision – ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXVI, p. 631–648, Berlin, Heidelberg, 2022. Springer-Verlag.
- [34] Zuo, Yukun, Yao, Hantao, Yu, Lu, Zhuang, Liansheng, and Xu, Changsheng. Hierarchical prompts for rehearsal-free continual learning. ArXiv, abs/2401.11544, 2024.
- [35] Jung, Dahuin, Han, Dongyoon, Bang, Jihwan, and Song, Hwanjun. Generating instance-level prompts for rehearsal-free continual learning. in 2023 IEEE/CVF International Conference on Computer Vision (ICCV), pp. 11813–11823, 2023.

- [36] Huang, Wei-Cheng, Chen, Chun-Fu, and Hsu, Hsiang. OVOR: Oneprompt with virtual outlier regularization for rehearsal-free class-incremental learning. in The Twelfth International Conference on Learning Representations, 2024.
- [37] Tang, Longxiang, Tian, Zhuotao, Li, Kai, He, Chunming, Zhou, Hantao, Zhao, Hengshuang, Li, Xiu, and Jia, Jiaya. Mind the interference: Retaining pre-trained knowledge in parameter efficient continual learning of vision-language models. in Leonardis, Aleš, Ricci, Elisa, Roth, Stefan, Russakovsky, Olga, Sattler, Torsten, and Varol, Gül, eds. , Computer Vision – ECCV 2024, pp. 346–365, Cham, 2024. Springer Nature Switzerland.
- [38] Villa, Andrés, Alcázar, Juan León, Alfarra, Motasem, Alhamoud, Kumail, Hurtado, Julio, Heilbron, Fabian Caba, Soto, Alvaro, and Ghanem, Bernard. Pivot: Prompting for video continual learning. in 2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 24214–24223, 2023.
- [39] Wang, Qiang, Du, Junlong, Yan, Ke, and Ding, Shouhong. Seeing in flowing: Adapting clip for action recognition with motion prompts learning. in Proceedings of the 31st ACM International Conference on Multimedia, MM '23, p. 5339–5347, New York, NY, USA, 2023. Association for Computing Machinery.
- [40] Wang, Hao, Liu, Fang, Jiao, Licheng, Wang, Jiahao, Hao, Zehua, Li, Shuo, Li, Lingling, Chen, Puhua, and Liu, Xu. Vilt-clip: Video and language tuning clip with multimodal prompt learning and scenario-guided optimization. Proceedings of the AAAI Conference on Artificial Intelligence, 38(6):5390–5400, Mar. 2024.
- [41] Roy, Anurag, Moulick, Riddhiman, Verma, Vinay, Ghosh, Saptarshi, and Das, Abir. Convolutional prompting meets language models for continual learning. in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), June 2024.

- [42] Wistuba, Martin, Teja Sivaprasad, Prabhu, Balles, Lukas, and Zappella, Giovanni. Choice of PEFT Technique in Continual Learning: Prompt Tuning is Not All You Need. arXiv e-prints, p. arXiv:2406.03216, June 2024.
- [43] Li, Jiashuo, Wang, Shaokun, Qian, Bo, He, Yuhang, Wei, Xing, Wang, Qiang, and Gong, Yihong. Dynamic integration of task-specific adapters for class incremental learning, 2025.
- [44] Zhou, Da-Wei, Sun, Hai-Long, Ye, Han-Jia, and Zhan, De-Chuan. Expandable subspace ensemble for pre-trained model-based class-incremental learning. in 2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp. 23554–23564, 2024.
- [45] Gao, Xinyuan, Dong, Songlin, He, Yuhang, Wang, Qiang, and Gong, Yihong. Beyond prompt learning: Continual adapter for efficient rehearsal-free continual learning. in Leonardis, Aleš, Ricci, Elisa, Roth, Stefan, Russakovsky, Olga, Sattler, Torsten, and Varol, Gül, eds. , Computer Vision – ECCV 2024, pp. 89–106, Cham, 2025. Springer Nature Switzerland.
- [46] Huang, Linlan, Cao, Xusheng, Lu, Haori, and Liu, Xialei. Class-incremental learning with clip: Adaptive representation adjustment and parameter fusion. in Computer Vision – ECCV 2024: 18th European Conference, Milan, Italy, September 29–October 4, 2024, Proceedings, Part LIV, p. 214–231, Berlin, Heidelberg, 2024. Springer-Verlag.
- [47] Pan, Junting, Lin, Ziyi, Zhu, Xiatian, Shao, Jing, and Li, Hongsheng. ST-adapter: Parameter-efficient image-to-video transfer learning. in Oh, Alice H., Agarwal, Alekh, Belgrave, Danielle, and Cho, Kyunghyun, eds. , Advances in Neural Information Processing Systems, 2022.
- [48] Wang, Huiyi, Lu, Haodong, Yao, Lina, and Gong, Dong. Self-expansion of pre-trained models with mixture of adapters for continual learning. in NeurIPS 2024 Workshop on Scalable Continual Learning for Lifelong Foundation Models, 2024.

- [49] Yu, Jiazuo, Zhuge, Yunzhi, Zhang, Lu, Hu, Ping, Wang, Dong, Lu, Huchuan, and He, You. Boosting continual learning of vision-language models via mixture-of-experts adapters. pp. 23219–23230, 06 2024.
- [50] Lu, Shuyun, Jiao, Jian, Wang, Lanxiao, Qiu, Heqian, Lin, Xingtao, Mei, Hefei, and Li, Hongliang. Video class-incremental learning with clip based transformer. in 2024 IEEE International Conference on Image Processing (ICIP), pp. 500–506, 2024.
- [51] Li, Jiashuo, Wang, Shaokun, Qian, Bo, He, Yuhang, Wei, Xing, Wang, Qiang, and Gong, Yihong. Dynamic integration of task-specific adapters for class incremental learning, 2025.
- [52] Wang, Huiyi, Lu, Haodong, Yao, Lina, and Gong, Dong. Self-expansion of pre-trained models with mixture of adapters for continual learning. in Proceedings of the Computer Vision and Pattern Recognition Conference, pp. 10087–10098, 2025.
- [53] Ilharco, Gabriel, Wortsman, Mitchell, Gadre, Samir Yitzhak, Song, Shuran, Hajishirzi, Hannaneh, Kornblith, Simon, Farhadi, Ali, and Schmidt, Ludwig. Patching open-vocabulary models by interpolating weights. in Oh, Alice H., Agarwal, Alekh, Belgrave, Danielle, and Cho, Kyunghyun, eds. , Advances in Neural Information Processing Systems, 2022.
- [54] Izmailov, Pavel, Podoprikin, Dmitrii, Garipov, Timur, Vetrov, Dmitry, and Wilson, Andrew. Averaging weights leads to wider optima and better generalization. 03 2018.
- [55] Soomro, Khurram, Zamir, Amir Roshan, and Shah, Mubarak. Ucf101: A dataset of 101 human actions classes from videos in the wild. arXiv preprint arXiv:1212.0402, 2012.
- [56] Kuehne, Hildegard, Jhuang, Hueihan, Garrote, Estíbaliz, Poggio, Tomaso, and Serre, Thomas. Hmdb: a large video database for human motion recognition. in 2011 International conference on computer vision, pp. 2556–2563. IEEE, 2011.

## پیوست

موضوعات مرتبط با متن گزارش پایان نامه که در یکی از گروه‌های زیر قرار می‌گیرد، در بخش پیوست‌ها آورده شوند:

۱. اثبات‌های ریاضی یا عملیات ریاضی طولانی.

۲. داده و اطلاعات نمونه (های) مورد مطالعه (Case Study) چنانچه طولانی باشد.

۳. نتایج کارهای دیگران چنانچه نیاز به تفصیل باشد.

۴. مجموعه تعاریف متغیرها و پارامترها، چنانچه طولانی بوده و در متن به انجام نرسیده باشد.

## کد میپل

```
with(DifferentialGeometry):  
with(Tensor):  
DGsetup([x, y, z], M)  
frame name: M  
a := evalDG(D_x)  
D_x  
b := evalDG(-2 y z D_x+2 x D_y/z^3-D_z/z^2)
```



# واژه‌نامه‌ی فارسی به انگلیسی

آ	Cartesian product . . . . حاصل ضرب دکارتی
اسکالر . . . . . Scalar	خ
ب	Automorphism . . . . . خودریختی
بالابر . . . . . Lift	د
پ	Degree . . . . . درجه
پایا . . . . . Invariant	ر
ت	microprocessor . . . . . ریزپردازنده
تناظر . . . . . Correspondence	ز
ث	Submodule . . . . . زیرمدول
ثابت‌ساز . . . . . Stabilizer	س
ج	Character . . . . . سرشت
جایگشت . . . . . Permutation	ص
چ	Faithful . . . . . صادقانه
چند جمله‌ای . . . . . Polynomial	ض
ح	

Connected . . . . . همبند	Inner product . . . . . ضرب داخلی
ی	ط
Edge . . . . . یال	Loop . . . . . طوقه
	ظ
	Valency . . . . . ظرفیت
	ع
	Nonadjacency . . . . . عدم مجاورت
	ف
	Vector space . . . . . فضای برداری
	ک
	Complete reducibility . . . . . کاملاً تحویل پذیر
	گی
	Graph . . . . . گراف
	م
	Permutation matrix . . . . . ماتریس جایگشتی
	ن
	Disconnected . . . . . ناهمبند
	و
	Invertible . . . . . وارون پذیر
	ه

# واژه‌نامه‌ی انگلیسی به فارسی

<b>A</b>	Homomorphism . . . . . هم‌ریختی
Automorphism . . . . . خودریختی	<b>I</b>
<b>B</b>	Invariant . . . . . پایا
Bijection . . . . . دوسویی	<b>L</b>
<b>C</b>	Lift . . . . . بالابر
Cycle group . . . . . گروه دوری	<b>M</b>
<b>D</b>	Module . . . . . مدول
Degree . . . . . درجه	<b>N</b>
<b>E</b>	Natural map . . . . . نگاشت طبیعی
Edge . . . . . یال	<b>O</b>
<b>F</b>	One to One . . . . . یک به یک
Function . . . . . تابع	<b>P</b>
<b>G</b>	Permutation group . . . . . گروه جایگشتی
Group . . . . . گروه	<b>Q</b>
<b>H</b>	

Quotient graph . . . . .	گراف خارج‌قسمتی	Trivial character . . . . .	سرشت بدیهی
<b>R</b>		<b>U</b>	
Reducible . . . . .	تحویل پذیر	Unique . . . . .	منحصربفرد
<b>S</b>		<b>V</b>	
Sequence . . . . .	دنباله	Vector space . . . . .	فضای برداری
<b>T</b>			

# **Abstract**

This page is accurate translation from Persian abstract into English.

## **Key Words:**

Write a 3 to 5 KeyWords is essential. Example: AUT, M.Sc., Ph. D, ..



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

**Department of ...**

**M. Sc. Thesis**

**Title of Thesis**

**By**

**Name Surname**

**Supervisor**

**Dr.**

**Advisor**

**Dr.**

**Month & Year**