به نام خدا

تمرین سری ششم
درس یادگیری ماشین
دکتر علی شریفی زارچی

فرزان رحمانی
۴۰۳۲۱۰۷۲۵

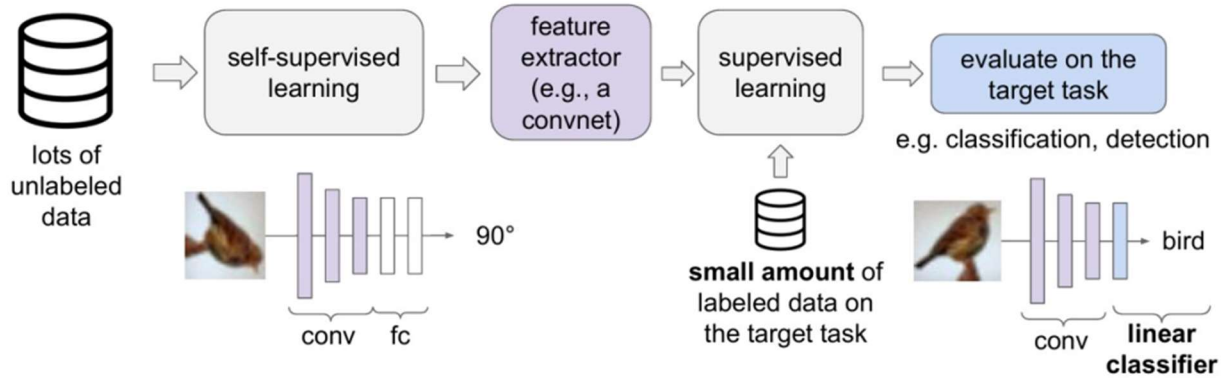
سوال اول

pretext task در یادگیری خود نظارتی

یک pretext task یک task مصنوعی است که برای تولید برچسب ها از خود داده ها بدون حاشیه نویسی انسانی طراحی شده است. هدف تشویق یک مدل برای یادگیری بازنمایی های معنادار با حل این وظایف است. در یادگیری خود نظارتی، از وظایف pretext در طول pretraining برای کمک به مدل‌ها برای یادگیری ویژگی‌های قابل تعمیم استفاده می‌شود که بعداً می‌توانند برای taskهای پایین دستی مانند طبقه‌بندی، تشخیص اشیا یا segmentation به‌خوبی fine-tune شوند.

ویژگی‌های آموخته‌شده از pretext tasks اغلب domain-specific هستند و می‌توانند ویژگی‌های مهمی مانند لبه‌ها، بافت‌ها، روابط فضایی یا ساختارهای معنایی را به تصویر بکشند.

- Self-supervised learning defines a "pretext" task based on unlabeled inputs to produce descriptive and intelligible representations [Hastie et al., 2009, Goodfellow et al., 2016]
  - Labels of these pretext tasks are generated *automatically*
  - Can be used in other downstream tasks.



1. Learn good feature extractors from self-supervised pretext tasks, e.g., predicting image rotations

2. Attach a shallow network on the feature extractor; train the shallow network on the target task with small amount of labeled data

الف) پیش بینی چرخش

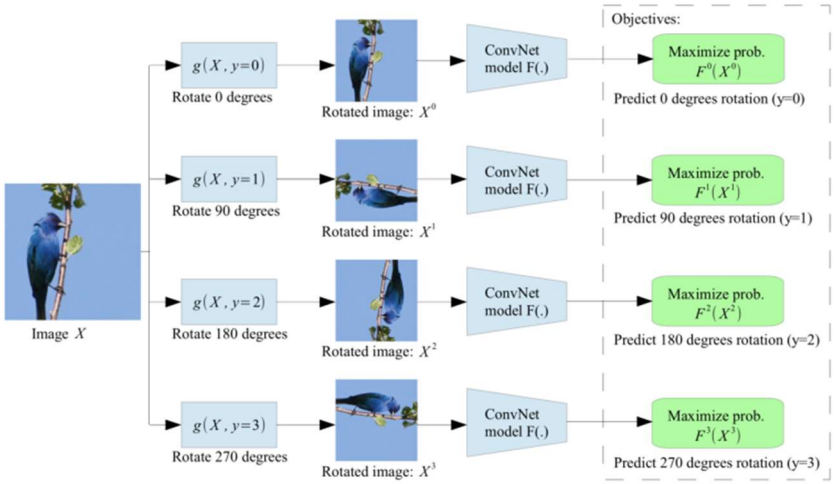


Figure 1: The model learns to predict which rotation is applied.

وظیفه:

در این task، یک تصویر به طور تصادفی توسط یکی از چندین زاویه ثابت (مثلاً ۰ درجه، ۹۰ درجه، ۱۸۰ درجه یا ۲۷۰ درجه) می چرخد و مدل برای پیش بینی زاویه چرخش اعمال شده آموزش داده می شود. برای موفقیت، مدل باید آرایش فضایی و جهت گیری اشیاء درون تصویر را درک کند. این امر یادگیری ویژگی های مربوط به اشکال اشیاء، ساختارها و تنظیمات فضایی آنها را تشویق می کند.

چه ویژگی هایی آموخته می شود:

- ویژگی‌های مربوط به جهت‌گیری (orientation-related features) مدل مانند اشکال شی، بافت‌ها و چیدمان (layout) های فضایی را آموزش می‌دهد.
- به مدل کمک می کند تا حس global structure را در تصاویر ایجاد کند.

نوع ویژگی ها:

- ویژگی های هندسی و درک فضایی (جهت گیری و موقعیت اشیاء).

ب) رنگ آمیزی

وظیفه:

مدل یک تصویر خاکستری را به عنوان ورودی دریافت می کند و وظیفه پیش بینی نسخه رنگی اصلی را دارد. برای انجام موثر این task، مدل نیاز به گرفتن اطلاعات معنایی در مورد اشیا و صحنه ها دارد، زیرا اشیاء مختلف رنگ های مشخصی دارند. به عنوان مثال، تشخیص اینکه آسمان معمولاً آبی است یا اینکه چمن سبز است به رنگ‌بندی دقیق کمک می‌کند. بنابراین، این task به یادگیری ویژگی های معنایی و درک سطح شی کمک می کند.

چه ویژگی هایی آموخته می شود:

- درک معنایی اشیاء و صحنه ها را به مدل آموزش می دهد زیرا اشیا معمولاً رنگ های خاصی دارند.
- مدل یاد می گیرد که روابط contextual بین قسمت های مختلف تصویر را برای استنتاج رنگ ها ثبت کند.

نوع ویژگی ها:

- ویژگی های معنایی (Semantic features) و contextual relationships بین مناطق تصویر.

ج) حل پازل

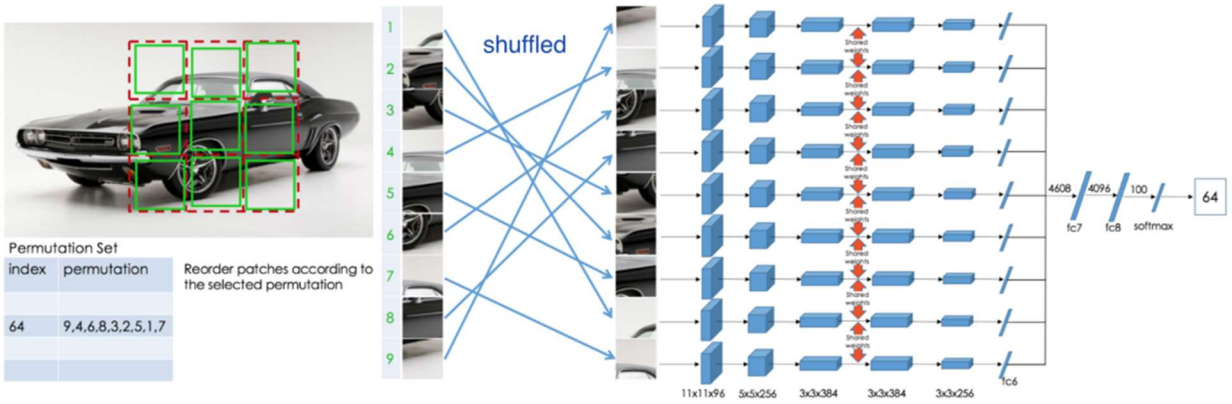


Figure 3: The model learns to solve jigsaw puzzle

وظیفه:

در این pretext task، یک تصویر به چندین patch تقسیم می شود که سپس shuffle می شود. هدف این مدل این است که patch ها را به موقعیت اصلی خود بازآرایی کند. حل موفقیت‌آمیز این task مستلزم درک روابط فضایی و وابستگی‌های contextual بین بخش‌های مختلف تصویر است که منجر به یادگیری ویژگی‌های مرتبط با بافت فضایی و ساختار کلی اشیا و صحنه‌ها می‌شود.

چه ویژگی هایی آموخته می شود:

- روابط فضایی مدل بین مناطق مختلف تصویر را آموزش می دهد.
- مدل یاد می گیرد که ساختار کلی یک تصویر را با استدلال در مورد اینکه چگونه قطعات با هم قرار می گیرند، درک کند.

نوع ویژگی ها:

- ویژگی های آرایش فضایی (Spatial arrangement features) و وابستگی های contextual بین patch های تصویر.

با آموزش در این وظایف pretext، مدل‌ها بازنمایی ویژگی‌های غنی و قابل انتقالی (transferable) را توسعه می‌دهند که می‌تواند عملکرد را در وظایف computer vision مختلف، حتی در غیاب مجموعه داده‌های برچسب‌دار بزرگ، افزایش دهد.

خلاصه وظایف بهانه و ویژگی های آموخته شده

Pretext Task	Objective	Type of Features Learned
Rotation Prediction	Predict image rotations	Geometric and spatial features
Colorization	Colorize grayscale images	Semantic and contextual features
Jigsaw Puzzle Solving	Reorder shuffled image patches	Spatial arrangement and contextual features

هر pretext task ای مدل را تشویق می‌کند تا انواع مختلفی از representation های تصویر را که برای taskهای پایین‌دستی در بینایی کامپیوتر مفید هستند، بیاموزد.

سوال دوم

از بین سه pretext task سوال اول، حل پازل (Jigsaw puzzle solving) را به عنوان مناسب ترین گزینه انتخاب می کنیم. در این مقاله (https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=9749256) نیز از یک ورژن تعمیم یافته Jigsaw puzzle solving استفاده شده است که نتیجه خوب آن را میبینیم. در ادامه توضیح میدهیم چرا:

الف) سازگاری با مشخصات تصویر ماهواره ای:

- تصاویر ماهواره ای تداوم فضایی قوی (strong spatial continuity) دارند - جاده ها، ساختمان ها و ویژگی های طبیعی در سراسر تصویر با الگوهای قابل پیش بینی ادامه می یابند (continue across the image in predictable patterns).
- این تصاویر از یک چشم انداز ثابت از بالا گرفته شده اند و روابط فضایی ثابتی (consistent spatial relationships) را ایجاد می کنند.
- مناطق شهری دارای الگوهای منظم و ساختارهای هندسی هستند (به عنوان مثال، شبکه های جاده ها، بلوک های ساختمانی، شبکه های شهری و ...)
- روابط فضایی و الگوها در بافت‌های شهری نسبتاً متغیر هستند.
- تکه های محلی (Local patches) در تصاویر ماهواره ای شهری به دلیل برنامه ریزی شهری و الگوهای توسعه به شدت به یکدیگر وابسته هستند.

به طور خلاصه:

تصاویر ماهواره ای به دلیل ویژگی های فضایی ذاتی آنها برای حل پازل (jigsaw puzzle solving) بسیار مناسب هستند. این تصاویر مناظر شهری را از یک چشم انداز ثابت از بالا به تصویر می کشند که منجر به الگوهای پیوسته می شود که در آن جاده ها، ساختمان ها و زیرساخت ها شبکه های به هم پیوسته و قابل پیش بینی را تشکیل می دهند. سازماندهی شبکه‌ای مناطق شهری، با الگوهای خیابانی منظم و بلوک‌های ساختمانی، وابستگی‌های فضایی طبیعی بین مناطق مجاور ایجاد می‌کند.

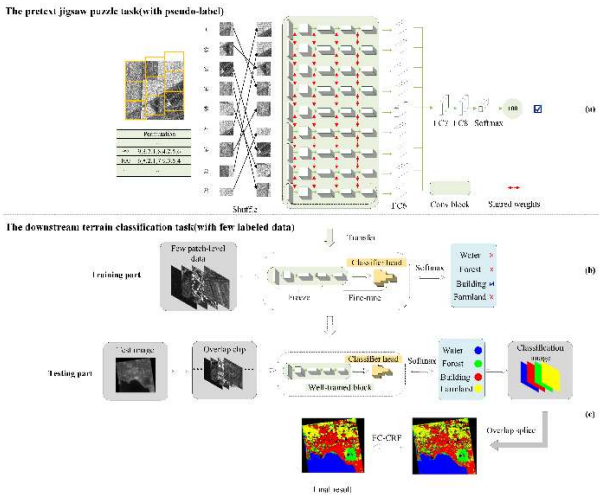
ماهیت scale-invariant الگوهای شهری و وابستگی متقابل قوی بین مناطق همسایه، حل پازل را به یک تسک یادگیری مؤثر تبدیل می کند. هنگامی که یک تصویر به تکه‌هایی (patches) تقسیم می‌شود، مدل باید یاد بگیرد که هم ویژگی‌های محلی (مانند خوشه‌های ساختمان و بخش‌های جاده) و هم روابط فضایی وسیع‌تر آن‌ها را درک کند تا به درستی تصویر را reassemble کند، که آن را به یک pretext task ایده‌آل برای یادگیری بازنمایی‌های معنادار محیط‌های شهری تبدیل می‌کند.

Fig. 3. A typical example for predicting the relative position: 3×3 Jigsaw puzzles. A DCNN takes nine patches as input and predict their positions.

ب) استفاده از pretext task حل پازل بر روی داده ها:

- تصویر را می توان به regular grid patches تقسیم کرد (به عنوان مثال، 3x3 or 4x4)
- هر patch دارای ویژگی های شهری معنی دار (خوشه های ساختمانی، بخش های جاده و غیره) است.
- مدل باید یاد بگیرد که موارد زیر را بفهمد:
  - تداوم شبکه های جاده ای در سراسر patch ها
  - الگوهای هم ترازی ساختمان ها
  - gradient تراکم شهری
  - الگوهای توزیع فضای سبز
  - جریان منطقی زیرساخت های شهری

در واقع در این pretext task، یک تصویر ماهواره ای به چندین patch تقسیم می شود که سپس shuffle می شود. هدف این مدل این است که patch ها را به موقعیت اصلی خود بازآرایی کند. حل موفقیت‌آمیز این task مستلزم درک روابط فضایی و وابستگی‌های contextual بین بخش‌های مختلف تصویر است که منجر به یادگیری ویژگی‌های مرتبط با بافت فضایی و ساختار کلی اشیا و صحنه‌ها می‌شود. مانند شکل زیر:



در واقع ابتدا مدل مان را با استفاده از Jigsaw puzzle solving و تصاویر بدون برچسب ماهواره ای مناطق شهری pretrain میکنیم (self-supervised learning). سپس در وظایف بعدی مانند تشخیص ساختمان ها یا طبقه بندی کاربری زمین با داده های داری برچسب fine-tune میکنیم (supervised learning).

ج) محدودیت های دو وظیفه دیگر:

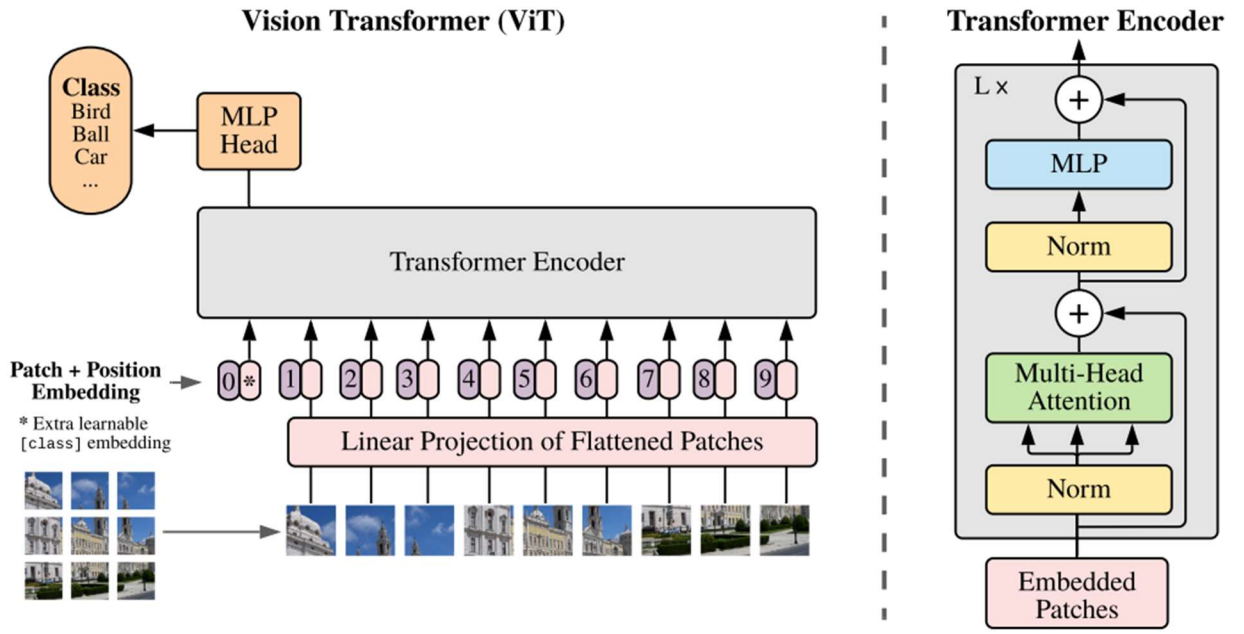
پیش بینی چرخش:

- تصاویر ماهواره ای از مناطق شهری اغلب دارای تقارن چرخشی (rotational symmetry) به دلیل الگوهای شبکه مانند خیابان (grid-like street patterns) هستند.
- بسیاری از ویژگی های شهری هنگام چرخش شبیه به هم به نظر می رسند (به عنوان مثال، تقاطع ها، بلوک های ساختمانی و ...)
- Orientation یا جهت گیری "طبیعی" تصاویر ماهواره ای دلخواه است زیرا شهرها در جهت گیری های مختلف توسعه می یابند. (برخلاف تصاویر عادی که به علت جاذبه مثلا همیشه پای آدم پایین و سر آدم بالاست.)
- الگوهای شهری می توانند در جهت گیری های متعدد (multiple orientations) معتبر باشند و پیش بینی چرخش را مبهم می کند (making rotation prediction ambiguous).

رنگ آمیزی:

- تصاویر ماهواره‌ای معمولاً رنگی هستند و ممکن است شامل باندهای چند طیفی (multi-spectral bands) باشند.
- تصاویر ماهواره‌ای اغلب از باندهای طیفی تخصصی فراتر از RGB (specialized spectral bands beyond RGB) استفاده می‌کنند که رنگ‌آمیزی سنتی را کم معنا تر می‌کند.
- ویژگی های شهری اغلب بیشتر با الگوهای فضایی متمایز می شوند تا رنگ.
- تغییرات رنگ در تصاویر ماهواره ای شهری می تواند ظریف و وابسته به فصل باشد.
- بسیاری از ویژگی‌های مهم شهری (جاده‌ها، ساختمان‌ها) رنگ‌های مشابهی دارند، که باعث می‌شود رنگ‌بندی برای یادگیری ویژگی‌ها کمتر آموزنده باشد.
- شرایط جوی و نور می تواند به طور قابل توجهی بر رنگ ها در تصاویر ماهواره ای تأثیر بگذارد.
- اشیاء در مناطق شهری رنگ های قابل پیش بینی یا طبیعی ندارند (به عنوان مثال، پشت بام ها می توانند قرمز، خاکستری یا سفید باشند)، که باعث می شود رنگ آمیزی کمتر موثر باشد.

سوال سوم



الف) تعداد کل patch ها (N) و تبدیل خطی به جاسازی ها

۱. تعداد کل patch ها (N) را محاسبه میکنیم:

ابعاد تصویر ۲۲۴×۲۲۴ پیکسل و هر پچ ۱۶×۱۶ پیکسل است.

تعداد patch ها به صورت زیر محاسبه می شود:

$$N = \left(\frac{224}{16}\right) \times \left(\frac{224}{16}\right) = 14 \times 14 = 196 \text{ patches}$$

۲. تبدیل خطی به embedding ۱۲۸ بعدی:

هر patch ۱۶ در ۱۶ پیکسل است، به این معنی که هر patch دارای مقادیر ۲۵۶ پیکسل (بعد از flatten شدن) است. برای تبدیل هر پچ به یک embedding ۱۲۸ بعدی، یک لایه Linear (که به آن لایه کاملا متصل نیز می‌گویند) اعمال می‌شود.

- اجازه دهید بردار patch با $\boldsymbol{x_i} \in R^{256}$ (flattened vector of pixel values) نشان داده شود.
- لایه Linear یک ماتریس وزن $\boldsymbol{W} \in R^{128\times256}$ و یک بردار بایاس $\boldsymbol{b} \in R^{128}$ اعمال می‌کند.

پس تبدیل خطی چنین است:

$$z_i = W \cdot x_i + b$$

که:

- $\boldsymbol{x_i}$ یک بردار ورودی ۲۵۶ بعدی (پیکسل‌های patch) است.
- $\boldsymbol{W}$ یک ماتریس تبدیل 256×128 است.
- $\boldsymbol{b}$ یک بردار بایاس ۱۲۸ بعدی است.
- $\boldsymbol{z_i}$ یک بردار embedding شده ۱۲۸ بعدی است.

بنابراین، هر patch ۲۵۶ بعدی به صورت خطی به یک embedding ۱۲۸ بعدی (Linear Projection of Flattened Patches) نگاشت می شود.

ب) افزودن جاسازی موقعیتی به Patch Embedding

در ViT ها، معماری transformer برخلاف شبکه‌های عصبی کانولوشنال (CNN) که بر روابط فضایی محلی متکی (local spatial relationships) هستند، نظم داخلی ندارد (does not have a built-in sense of order). برای دادن حس ساختار فضایی (a sense of spatial structure) به مدل، جاسازی موقعیتی (positional embedding) به جاسازی های پچ (patch embedding) اضافه می‌شوند.

چگونه کار می کند:

۱. Patch Embedding: هر یک از ۱۹۶ patch به یک embedding ۱۲۸ بعدی (128-dimensional embedding) تبدیل می شود.

۲. Positional Embedding: یک بردار positional embeddings قابل یادگیری به هر patch embedding اضافه می شود. بردار embedding موقعیتی دارای ابعادی مشابه با embedding پچ (۱۲۸ بعد) است.

به عنوان مثال:

$$E_i = z_i + p_i$$

که:

- $\boldsymbol{z_i}$ جاسازی پچ است.
- $\boldsymbol{p_i}$ جاسازی موقعیتی است.

جاسازی های موقعیتی تضمین می‌کنند که مدل موقعیت نسبی هر پچ (**relative position** of each patch within the image) را در تصویر می‌داند و transformer را قادر می‌سازد تا اطلاعات مکانی (spatial information) را بگیرد.

چرا جاسازی موقعیتی مهم است:

- بدون embeddingهای موقعیتی، transformer پچ‌ها را به عنوان مجموعه‌ای نامرتب (unordered set) در نظر می‌گیرد و هر مفهومی از آرایش فضایی را از دست می‌دهد (losing any notion of spatial arrangement).
- embeddingهای موقعیتی به مدل این امکان را می‌دهد که موقعیت و رابطه بین patchها را درک کند، که برای کارهایی مانند طبقه‌بندی تصویر و تشخیص اشیا حیاتی است.

ج) دنباله ورودی برای رمزگذار transformer با توکن ویژه [CLS]

در ViT، دنباله ورودی برای رمزگذار transformer شامل یک توکن خاص به نام [CLS] (Classification Token) است که نقش مهمی در طبقه بندی تصویر ایفا می کند.

۱. دنباله ورودی: دنباله ورودی برای رمزگذار مدل transformer به صورت زیر ساخته شده است:

- توکن [CLS]: یک بردار ۱۲۸ بعدی قابل یادگیری به embeddingهای پچ اضافه می شود.
- Patch Embedding ها: علاوه برا توکن [CLS]، 196 پچ embedding شده (هر کدام با ۱۲۸ بعد) گنجانده می شود.
- جاسازی های موقعیتی: embedding های موقعیتی به هر یک از این ۱۹۷ بردار اضافه می شود.

بنابراین، دنباله ورودی دارای:

- در مجموع ۱۹۷ توکن (1 [CLS] token + 196 patch embeddings).
- هر توکن دارای ۱۲۸ بعد است.

در واقع، دنبال ورودی Transformer encoder به شکل زیر می شود:

$$[[CLS], E_1, E_2, \dots, E_{196}]$$

۲. ابعاد توالی ورودی:

دنباله ورودی را می توان به عنوان یک ماتریس با ابعاد زیر نشان داد:

$$\text{Input Sequence} = (197 \times 128)$$

۳. نقش توکن [CLS]:

- توکن [CLS] یک بردار قابل یادگیری ویژه است که برای جمع آوری اطلاعات از تمام patch ها در طول فرآیند رمزگذاری طراحی شده است.
- پس از اینکه رمزگذار transformer توالی ورودی را پردازش می کند، خروجی نهایی مربوط به توکن [CLS] به عنوان بازنمایی سطح تصویر (image-level representation) برای کارهایی مانند طبقه بندی تصویر استفاده می شود.

۴. استفاده از توکن [CLS] در طبقه بندی تصویر:

- خروجی مربوط به توکن [CLS] از یک classification head (معمولاً یک لایه کاملاً متصل با یک softmax) برای پیش بینی کلاس تصویر عبور داده می شود.
- این شبیه به نحوه استفاده از توکن [CLS] در وظایف NLP مانند BERT است، جایی که نشان دهنده کل دنباله ورودی است.

خلاصه نکات کلیدی:

Component	Value/Description
Total Patches (N)	196
Patch Size	16 x 16 pixels
Patch Vector Dimension	256
Patch Embedding Dimension	128
Positional Embedding Dimension	128
Input Sequence Length	197 (196 patches + 1 [CLS] token)
Input Sequence Dimension	(197 x 128)
Role of [CLS] Token	Represents the entire image for classification

سوال چهارم

الف) چگونه CLIP امتیاز شباهت را برای هر جفت تصویر و متن محاسبه می کند؟

CLIP (Contrastive Language-Image Pretraining) به طور مشترک یک رمزگذار تصویر و یک رمزگذار متن را آموزش می‌دهد تا هم تصاویر و هم متون را در یک فضای جاسازی مشترک که در آن جفت‌های تصویر-متن مشابه از نظر معنایی شباهت کسینوس بالاتری دارند، نمایش دهد.

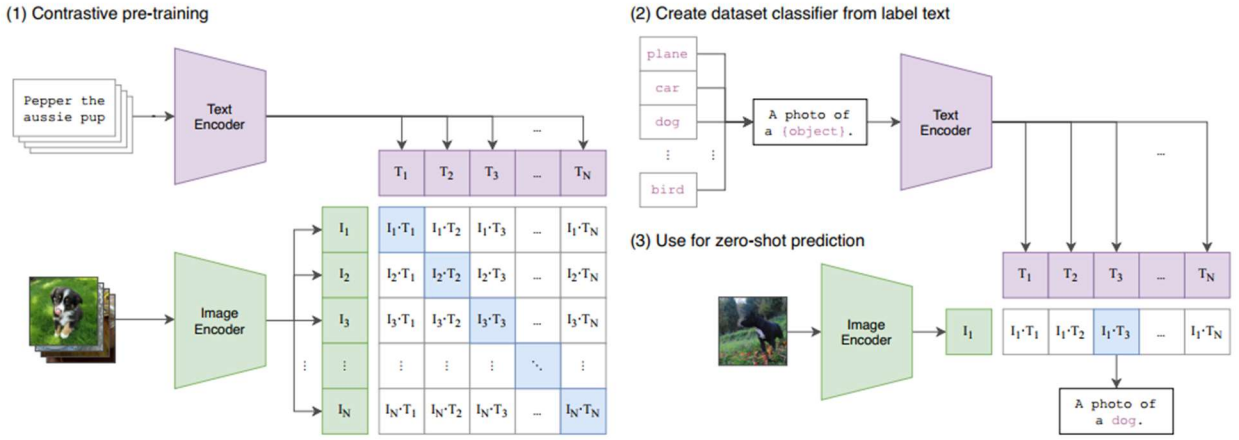


Figure 1. Summary of our approach. While standard image models jointly train an image feature extractor and a linear classifier to predict some label, CLIP jointly trains an image encoder and a text encoder to predict the correct pairings of a batch of (image, text) training examples. At test time the learned text encoder synthesizes a zero-shot linear classifier by embedding the names or descriptions of the target dataset's classes.

مراحل محاسبه نمرات شباهت (Similarity Scores):

۱. رمزگذاری تصویر: تصویر "سیب قرمز" از رمزگذار تصویر (معمولاً Vision Transformer یا ResNet) عبور داده می شود تا بردار جاسازی تصویر v_{image} بدست آید.

۲. رمزگذاری متن: هر توضیح متنی از طریق رمزگذار متن (معمولاً یک مدل مبتنی بر Transformer) منتقل می شود تا بردارهای جاسازی متن v_{text1}، v_{text2} و v_{text3} به ترتیب برای سه متن به دست آید.

۳. محاسبه تشابه کسینوسی (Cosine Similarity): برای هر جفت تصویر-متن، CLIP شباهت کسینوس بین جاسازی تصویر و متن را محاسبه می کند:

$$\text{similarity}(v_{image}, v_{text}) = \frac{v_{image} \cdot v_{text}}{\|v_{image}\| . \|v_{text}\|}$$

کدام زوج بالاترین امتیاز را می گیرد؟

متن توصیفی "یک سیب قرمز آبدار روی میز" به دلایل زیر احتمالاً بالاترین امتیاز شباهت را دریافت می کند:

- حاوی ارجاعات دقیقی به رنگ ("قرمز") و شی ("سیب") است که با تصویر مطابقت دارد.
- CLIP برای تشخیص مفاهیم بصری در context، آموزش داده شده است و این متن توصیفی context بسیار مرتبط و دقیقی را ارائه می دهد.

دو متن توصیفی دیگر کمتر مرتبط هستند:

- «یک سیب سبز آویزان از درخت» شیء صحیح («سیب») اما **رنگ نادرست** («سبز») دارد.
- "یک توپ قرمز درخشان" رنگ صحیح ("قرمز") اما **شی اشتباه** ("توپ") دارد.

بنابراین، مرتبط ترین جفت تصویر-متن عبارتند از:

بالاترین امتیاز → تصویر "سیب قرمز" و توصیف متنی "یک سیب قرمز آبدار روی میز"

ب) اگر رتبه مدل "یک توپ قرمز درخشان" بالاتر از "یک سیب سبز" باشد چه چیزی را نشان می دهد؟

اگر مدل CLIP "یک توپ قرمز درخشان" را بالاتر از "یک سیب سبز" قرار دهد، این نشان دهنده موارد زیر در مورد فضای نمایش تعبیه شده (embedding space) توسط CLIP است:

۱. این مدل رنگ را بر دسته بندی شی اولویت می دهد (Prioritizes Color Over Object Category):

این رفتار نشان می دهد که embedding space توسط CLIP شباهت رنگ را بر شباهت شی اولویت می دهد. در این مورد، مدل رنگ مشترک "قرمز" بین تصویر سیب و توضیحات متن "یک توپ قرمز درخشان" را مهم تر از دسته اشیاء مشترک ("سیب") در توضیحات متن "یک سیب سبز" تشخیص می دهد.

۲. embedding space ویژگی های بصری را قوی تر از مفاهیم معنایی رمزگذاری می کند:

این نشان می‌دهد که مدل بر ویژگی‌های بصری مانند رنگ، شکل یا بافت، به طور بالقوه بیش از semantic meaning تأکید می‌کند. مدل ممکن است آموخته باشد که رنگ در داده‌های آموزشی آن ویژگی متمایزتری (more distinguishing feature) است که منجر به این رفتار می‌شود.

۳. سوگیری بالقوه (Potential Bias) در داده های آموزشی مدل:

این رفتار همچنین می تواند نشان دهنده یک سوگیری در داده های آموزشی باشد. اگر داده‌های آموزشی حاوی نمونه‌های زیادی باشد که رنگ یک عامل تمایز کلیدی بین اشیاء (key distinguishing factor between objects) است، ممکن است مدل یاد گرفته باشد که هنگام محاسبه شباهت، رنگ را بیش از حد اولویت قرار دهد.

خلاصه ای از مطالب های کلیدی:

توضیح امتیاز شباهت	متن توصیفی
بالاترین امتیاز به دلیل تطبیق شی ("سیب") و رنگ ("قرمز").	**"یک سیب قرمز آبدار روی میز"**
امتیاز کمتر به دلیل عدم تطابق رنگ ("سبز" در مقابل "قرمز").	**"یک سیب سبز آویزان از درخت"**
اگر مدل شباهت رنگ را بر دسته شی (("توپ" در مقابل "سیب")) اولویت دهد، می تواند رتبه بالاتری نسبت به "یک سیب سبز" داشته باشد.	**"یک توپ قرمز درخشان"**

نتیجه گیری:

اگر CLIP "یک توپ قرمز درخشان" را بالاتر از "یک سیب سبز" قرار دهد، نشان می‌دهد که مدل شباهت‌های بصری (مانند رنگ) را بر شباهت‌های معنایی (دسته شی) اولویت می‌دهد (the model is prioritizing visual similarities (like color) over semantic similarities (object category)). این رفتار اهمیت حصول اطمینان از اینکه مدل ویژگی‌های بصری و معنایی را در embedding space آموخته‌شده‌اش متعادل می‌کند، آشکار می‌کند.

سوال پنجم

ابتدا تصاویر بخش هایی از مقاله که به این موضوعات اشاره دارند میبینیم و سپس توضیحات من آمده است.

الف)

2.4. Choosing and Scaling a Model

We consider two different architectures for the image encoder. For the first, we use ResNet-50 (He et al., 2016a) as the base architecture for the image encoder due to its widespread adoption and proven performance. We make several modifications to the original version using the ResNet-D improvements from He et al. (2019) and the antialiased rect-2 blur pooling from Zhang (2019). We also replace the global average pooling layer with an attention pooling mechanism. The attention pooling is implemented as a single layer of "transformer-style" multi-head QKV attention where the query is conditioned on the global average-pooled representation of the image. For the second architecture, we experiment with the recently introduced Vision Transformer (ViT) (Dosovitskiy et al., 2020). We closely follow their implementation with only the minor modification of adding an additional layer normalization to the combined patch and position embeddings before the transformer and use a slightly different initialization scheme.

The text encoder is a Transformer (Vaswani et al., 2017) with the architecture modifications described in Radford et al. (2019). As a base size we use a 63M-parameter 12-layer 512-wide model with 8 attention heads. The transformer operates on a lower-cased byte pair encoding (BPE)

مکانیسم:

- Attention Pooling: در CLIP، attention pooling از مکانیزم توجه چند سر Query-Key-Value (QKV) به سبک ترانسفورماتور ("transformer-style" multi-head Query-Key-Value (QKV) attention mechanism) استفاده می کند (learned query that extracts important features by attending to different parts of the image with varying importance). query ما conditioned به یک globally pooled representation از تصویر است، که به مدل اجازه می دهد بر روی ویژگی های تصویر خاص بر اساس اهمیت آنها تمرکز کند.
- Global Average Pooling (GAP): GAP میانگین مقادیر ویژگی را در تمام مکان‌های فضایی (mean of feature values across all spatial locations) محاسبه می‌کند، و ویژگی‌ها را به طور یکنواخت بدون در نظر گرفتن اهمیت نسبی آنها خلاصه می‌کند (averaging all the feature maps to produce a single output value per feature map). این عملیات ساده است و اهمیت فضایی مناطق مختلف در تصویر را در نظر نمی گیرد.

خروجی:

- Attention Pooling: یک نمایش ویژگی را تولید می کند که در آن قسمت های مهم تصویر توجه بیشتری را به خود جلب می کند و در نتیجه سهم وزنی (weighted contributions) در خروجی ایجاد می کند.
- Global Average Pooling: میانگین وزنی را تولید می کند که ممکن است اهمیت ویژگی های مهم را به خصوص در تصاویر پیچیده کاهش دهد.

بنابراین، attention pooling راه انعطاف‌پذیرتری برای خلاصه‌سازی ویژگی‌های تصویر در مقایسه با global average pooling ارائه می‌دهد، زیرا می‌تواند بر مرتبط‌ترین بخش‌های یک تصویر تمرکز کند.

ب)

et al., 2020a). Noting these findings, we explored training a system to solve the potentially easier proxy task of predicting only which text *as a whole* is paired with which image and not the exact words of that text. Starting with the same bag-of-words encoding baseline, we swapped the predictive objective for a contrastive objective in Figure 2 and observed a further 4x efficiency improvement in the rate of zero-shot transfer to ImageNet.

Given a batch of N (image, text) pairs, CLIP is trained to predict which of the $N \times N$ possible (image, text) pairings across a batch actually occurred. To do this, CLIP learns a multi-modal embedding space by jointly training an image encoder and text encoder to maximize the cosine similarity of the image and text embeddings of the N real pairs in the batch while minimizing the cosine similarity of the embeddings of the $N^2 - N$ incorrect pairings. We optimize a symmetric cross entropy loss over these similarity scores. In Figure 3 we include pseudocode of the core of an implementation of CLIP. To our knowledge this batch construction technique and objective was first introduced in the area of deep metric learning as the *multi-class N-pair loss* Sohn (2016), was popularized for contrastive representation learning by Oord et al. (2018) as the InfoNCE loss, and was

ماتریس لیبل مورد استفاده در contrastive learning یا همان یادگیری متضاد CLIP دارای اندازه $N \times N$ است که N همان batch size است. این ماتریس شامل تنها یک جفت مثبت (درست) در هر ردیف و $N - 1$ جفت منفی (نادرست) است. بنابراین، تعداد صفرها (جفت نادرست) در هر ردیف $N - 1$ و تعداد کل صفرهای ماتریس $N \times (N - 1)$ است.

$$\text{تعداد درایه های صفر در این ماتریس لیبل} = N \times (N - 1)$$

به عنوان نسبتی از N، نسبت صفرها در ماتریس لیبل برابر است با:

$$\frac{N \times (N - 1)}{N^2} = \frac{N - 1}{N} = 1 - \frac{1}{N}$$

با افزایش N، نسبت صفرها به ۱ نزدیک می شود (nearly **100% of the matrix** is filled with zeros for large N) و پراکندگی ماتریس را برجسته می کند. که این نشان دهنده اهمیت negative sampling را نشان می دهد.

ج)

Analysis in Section 3.1 found that CLIP's zero-shot performance is still quite weak on several kinds of tasks. When compared to task-specific models, the performance of CLIP is poor on several types of fine-grained classification such as differentiating models of cars, species of flowers, and variants of aircraft. CLIP also struggles with more abstract and systematic tasks such as counting the number of objects in an image. Finally for novel tasks which are unlikely to be included in CLIP's pre-training dataset, such as classifying the distance to the nearest car in a photo, CLIP's performance can be near random. We are confident that there are still many, many, tasks where CLIP's zero-shot performance is near chance level.

While zero-shot CLIP generalizes well to many natural image distributions as investigated in Section 3.3, we've observed that zero-shot CLIP still generalizes poorly to data that is truly out-of-distribution for it. An illustrative example occurs for the task of OCR as reported in Appendix E.

CLIP learns a high quality semantic OCR representation that performs well on digitally rendered text, which is common in its pre-training dataset, as evidenced by performance on Rendered SST2. However, CLIP only achieves 88% accuracy on the handwritten digits of MNIST. An embarrassingly simple baseline of logistic regression on raw pixels outperforms zero-shot CLIP. Both semantic and near-duplicate nearest-neighbor retrieval verify that there are almost no images that resemble MNIST digits in our pre-training dataset. This suggests CLIP does little to address the underlying problem of brittle generalization of deep learning models. Instead CLIP tries to circumvent the problem and hopes that by training on such a large and varied dataset that all data will be effectively in-distribution. This is a naive assumption that, as MNIST demonstrates, is easy to violate.

Although CLIP can flexibly generate zero-shot classifiers for a wide variety of tasks and datasets, CLIP is still limited to choosing from only those concepts in a given zero-shot classifier. This is a significant restriction compared to a

عملکرد صفر شات CLIP در تسک های task-specific مانند fine-grained classification انواع گونه های گل، تخصصی (specialized)، انتزاعی (abstract)، سیستماتیک (systematic) مانند شمارش اشیا و تسک های جدیدی که احتمالا مجموعه داده آنها در pre-training dataset نیست، ضعیف تر است، مانند:

- طبقه بندی تصاویر ماهواره ای (به عنوان مثال، EuroSAT، RESISC45)

- طبقه بندی تصاویر پزشکی (به عنوان مثال، PatchCamelyon)
- شمارش اشیاء در صحنه های مصنوعی (به عنوان مثال، CLEVRCCounts)
- وظایف مربوط به خودرو های خودران (مانند GTSRB برای علائم راهنمایی و رانندگی و KITTI Distance برای تشخیص فاصله خودرو)

این وظایف نیاز به دانش domain-specific یا abstract reasoning دارند (مانند counting)، که CLIP با آنها دست و پنجه نرم می کند. این نشان می‌دهد که embedding space یا همان فضای تعبیه‌شده CLIP برای مفاهیم کلی بصری (general visual concepts) به جای specialized or systematic reasoning tasks مناسب‌تر است.

دلایل احتمالی:

- تعمیم ضعیف نسبت به داده های out-of-distribution: zero-shot CLIP به طور ضعیفی به داده هایی که برای آن خارج از توزیع است تعمیم می یابد. (we've observed that zero-shot CLIP still generalizes poorly to data that is truly out-of-distribution for it.) مثال: CLIP تنها ۸۸ درصد دقت را در مجموعه داده ارقام دست‌نویس MNIST به دست می‌آورد. یک baseline ساده شرم آور از رگرسیون لجستیک بر روی پیکسل های خام بهتر از CLIP صفر شات است.
  هر دو semantic and near-duplicate nearest-neighbor retrieval تأیید می کنند که تقریباً هیچ تصویری شبیه ارقام MNIST در مجموعه داده های قبل از آموزش CLIP وجود ندارد. این نشان می‌دهد که CLIP به مشکل اساسی تعمیم شکننده مدل‌های یادگیری عمیق کمک چندانی نمی‌کند. در عوض، CLIP سعی می‌کند مشکل را دور بزند و امیدوار است که با آموزش بر روی چنین مجموعه داده‌ای بزرگ و متنوع، همه داده‌ها به طور موثر در توزیع باشند. این یک فرض ساده لوحانه است که همانطور که MNIST نشان می دهد، نقض آن آسان است.
- Domain Mismatch: مجموعه داده pre-training برای CLIP ممکن است فاقد نمونه‌ها یا تنوع کافی در این حوزه‌های خاص باشد که توانایی آن را برای تعمیم محدود می‌کند.
- Training Supervision Bias: نظارت زبان طبیعی در CLIP بر ویژگی‌های گسترده و همه منظوره به جای جزئیات خاص دامنه مورد نیاز برای کارهایی مانند تصویربرداری پزشکی یا تشخیص اشیاء fine-grained تأکید دارد.
- پیچیدگی وظایف: کارهای تخصصی اغلب نیاز به درک دقیق و دقت بالایی دارند، که ممکن است تنها با استفاده از تعمیم های صفر شات قابل دستیابی نباشد. Fine-tuning داده های دامنه خاص ممکن است برای بهبود عملکرد ضروری باشد.

پایان