به نام خدا

دانشگاه تهران

دانشکدگان فنی

دانشکده برق و کامپیوتر

درس تحلیل و طراحی شبکه های عصبی عمیق

تمرین امتیازی اول

آذر ۱۴۰۲

مقدمه

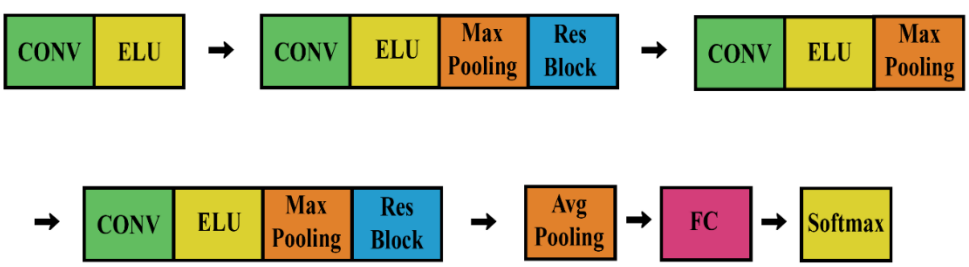
هدف از انجام این تمرین آشنایی با ساختار یک شبکه عصبی مشابه ResNet و بررسی تاثیر لایه های مختلف از جمله لایه های مختلف نرمالسازی بر میزان دقت و شاخص SI شبکه های عصبی است. همچنین آشنایی و پیاده سازی یک شبکه عصبی جهت تشخیص اشیا که توانایی تشخیص اشیا در محیط را دارد نیز در بخش دوم تمرین قرار گرفته است.

نکته مهم : در این تمرین باید تمام پیاده سازی ها از جمله معماری ها باید توسط خود شما انجام شود.

نکته مهم : به تمام نکات گفته شده در انتهای هر تمرین توجه داشته باشید.

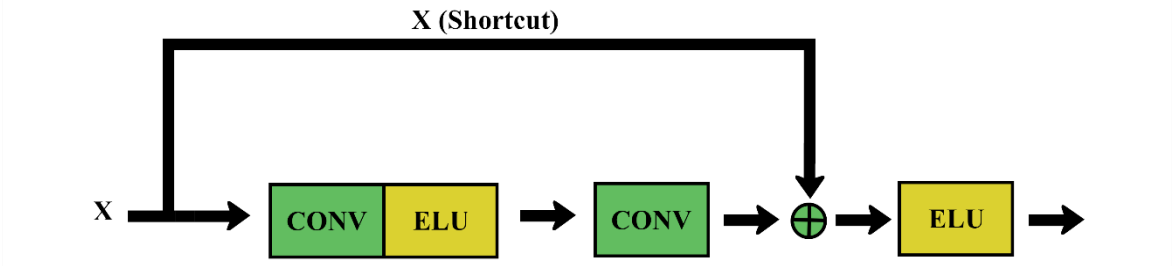
سوال یک) آشنایی با ساختار شبکه های عصبی

هدف از انجام این سوال آشنایی با ساختار شبکه‌ی عصبی معروف ResNet[1] و همچنین بررسی عملکرد لایه‌های پرکاربرد در چنین شبکه‌هایی است. شبکه عصبی مورد نظر ساختاری مانند شبکه‌های ResNet دارد، با این حال در جزییاتی همچون تعداد لایه‌ها، سایز کرنل و غیره متفاوت است؛ به همین دلیل نیاز است با استفاده از لایه‌های متداولی مانند لایه‌های کانولوشنی، لایه‌های Pooling و توابع فعال ساز گفته شده، شبکه را ایجاد کنید. ساختار شبکه عصبی مدنظر در شکل (۱) نمایش داده شده است.



شکل ۱- ساختار شبکه Custom Resnet

ساختار Residual Block مدنظر نیز در شکل (۲) نشان داده شده است.



شکل ۲- بلوک دیاگرام مربوط به Residual Block

جزئیات مربوط به تعداد فیلترها و سایز کرنل‌ها در جدول (۱) قابل مشاهده است.

[1] He, Kaiming, et al. "Deep residual learning for image recognition." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.

جدول 1 - جزئیات مربوط به تعداد فیلترها و سایز کرنل‌ها در ساختار شبکه custom ResNet مدنظر

ID	Layer	Kernel Size	Num. Filters	Output Shape
1	Conv2D	3×3	32	(32×32×32)
2	ELU	-	-	(32×32×32)
3	Conv2D	3×3	64	(32×32×64)
4	ELU	-	-	(32×32×64)
5	MaxPooling2D	2×2	-	(16×16×64)
6	Conv2D	3×3	64	(16×16×64)
7	ELU	-	-	(16×16×64)
8	Conv2D	3×3	64	(16×16×64)
9	Add	-	-	(16×16×64)
10	ELU	-	-	(16×16×64)
11	Conv2D	3×3	128	(16×16×128)
12	ELU	-	-	(16×16×128)
13	MaxPooling2D	2×2	-	(8×8×128)
14	Conv2D	3×3	256	(8×8×256)
15	ELU	-	-	(8×8×256)
16	MaxPooling2D	2×2	-	(4×4×256)
17	Conv2D	3×3	256	(4×4×256)
18	ELU	-	-	(4×4×256)
19	Conv2D	3×3	256	(4×4×256)
20	Add	-	-	(4×4×256)
21	ELU	-	-	(4×4×256)
22	Avg. Pooling	3×3	-	(1×1×256)
23	Flatten	-	-	(256)
24	Dense	-	-	(256)
25	Dense	-	-	(10)

نکات مربوط به گزارش نتایج در کل سوال (۱) :

انتظار می رود در این سوال به ازای هر آموزش انجام شده (در تمام بخش های سوال) بر روی شبکه Custom ResNet موارد خواسته شده زیر را گزارش نمائید.

- نمودار دقت[۱] و تابع هزینه[۲] داده های آموزشی[۳] و ارزیابی[۴] در حین آموزش شبکه
- دقت شبکه بر روی داده‌های تست بهترین نسخه از شبکه آموزش داده
- نمودار مقدار Center SI داده های آموزشی و تست در تمام لایه ها

نکاتی در ارتباط با مجموعه داده در کل سوال (۱) :

- از داده‌های تست اختصاص داده شده صرفا برای تست نهایی مدل در انتهای آموزش شبکه استفاده کنید و داده‌های ارزیابی را از داده‌های آموزشی جدا نمائید. برای این تمرین دقیقا ده درصد داده ها را جدا نمائید.
- با توجه به متوازن[۵] بودن مجموعه‌داده‌ها، لازم است داده های آموزشی و ارزیابی نیز به صورت متوازن انتخاب بشوند.
- پیش از استفاده از مجموعه داده در آموزش و ارزیابی شبکه، حتما داده ها را نرمال کنید.
- از روش های تقویت داده مناسب جهت افزایش دقت شبکه و جهت جلوگیری از Overfitting استفاده نمائید.
- از تمام داده ها برای آموزش، ارزیابی و تست شبکه استفاده کنید و داده ای را کنار نگذارید.
- از ده درصد داده ها برای محاسبه شاخص SI استفاده کنید.

نکاتی در ارتباط با شبکه عصبی در کل سوال (۱) :

- معماری پیاده سازی شده عینا معادل جدول (۱) باشد.
- برای ابر پارامترهای مورد استفاده در آموزش شبکه از پارامترهای گفته شده در جدول (۲) استفاده نمائید.

---

[1] Accuracy
[2] Loss Function
[3] Train
[4] Validation
[5] Balanced

- وزن های اولیه مدل باید به صورت رندوم باشد.

Parameter	Value
Optimizer	SGD with Momentum
Loss Function	Categorical Cross Entropy
Batch Size	256
Epochs	100
Learning Rate Scheduler	StepLR
Initial Learning Rate	با استفاده از جستجو مقدار اولیه نرخ یادگیری و مقدار Step را پیدا کنید

جدول ۲ – ابرپارامترهای مورد استفاده در سوال (۱)

الف) شبکه Custom ResNet شرح داده شده در صورت سوال را پیاده‌سازی کرده و بر روی مجموعه داده CIFAR10 آموزش دهید.

نکات مرتبط با قسمت (الف) سوال (۱):

- از هیچ لایه نرمالسازی در این قسمت استفاده نکنید.

ب) بصورت مختصر توضیح دهید که هدف از بکارگیری لایه‌های نرمالسازی مانند batch normalization، group normalization و layer normalization در شبکه‌های عصبی چیست؟ تفاوت این سه لایه با یکدیگر را بیان کنید.

ج) هر کدام از سه نوع لایه نرمالسازی که در بخش قبل ذکر شد را به صورت جداگانه در شبکه custom ResNet بکار برده و تاثیر آنها در شبکه را نشان دهید.

نکات مرتبط با قسمت (ج) سوال (۱) :

- از هر لایه نرمالساز به تعداد مناسب و در مکان مناسب استفاده کنید.
- در این بخش باید سه شبکه custom resnet آموزش ببینند؛ یک شبکه به همراه batch normalization، یک شبکه به همراه group normalization و یک شبکه هم به همراه layer normalization)

د) در مدل پیاده سازی شده قسمت Skip Connection را حذف نمائید. تاثیر حذف این Connection را بر دقت و مقدار SI بررسی نمائید.

ه) در شبکه پیاده سازی شده در چه لایه هایی جایگزینی لایه کانولوشنی موجود با 1x1 Conv وجود دارد و منطقی است؟ در یک لایه آن را جایگزین کنید و تاثیر آن را ارزیابی کنید.

و) در شبکه ResNeXt[1] از بلاکی به اسم Grouped Convulation استفاده می شود. مقاله را مطالعه نمائید و به جای Residual Block های موجود فعلی از Grouped Convulation با Path برابر ۲ و ۴ استفاده کنید و تاثیر آن را ارزیابی نمایید.

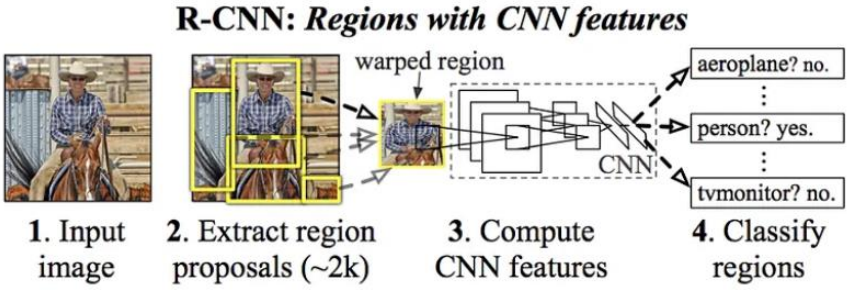
نکات مرتبط با قسمت (و) سوال (۱) :

- در این بخش باید دو شبکه custom resnet آموزش ببینند؛ یک شبکه با path برابر ۲ و یک شبکه با path برابر ۴ استفاده نمایید.

[1] Xie, S., Girshick, R., Dollár, P., Tu, Z., & He, K. (2017). Aggregated residual transformations for deep neural networks. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1492-1500).

سوال دو) شبکه تشخیص اشیا

هدف از این سوال، پیاده‌سازی یک شبکه برای تشخیص اشیاء است. در مباحث درس با Region-based CNN ها آشنا شده‌اید. از آنجایی که شبکه CNN به همراه لایه‌های fully connected متصل به آن قادر به حل مسئله‌ای که در آن چند شیء وجود دارد، نیست، بنابراین، یک راه حل می‌تواند این باشد که از یک جستجو بصورت sliding window برای انتخاب یک منطقه[1] و اعمال مدل CNN بر روی آن استفاده کنیم؛ اما مشکل این رویکرد این است که همان شیء را می‌توان در یک تصویر با اندازه‌ها و نسبت‌های مختلف نشان داد. در این حالت، زمانی که این عوامل را در نظر می‌گیریم، region proposal های زیادی داریم و اگر یک شبکه CNN را برای همه آن مناطق اعمال کنیم، از نظر محاسباتی بسیار گران هستند. در سال ۲۰۱۳، مقاله‌ای[2] منتشر شد که در آن R-CNN[3] برای حل این مشکل تشخیص اشیاء معرفی شد. در شکل زیر، معماری کلی R-CNN نشان داده شده است.



شکل ۳ – معماری R-CNN

همانطور که در تصویر بالا مشاهده می‌شود، R-CNN از الگوریتم جستجوی انتخابی[4] استفاده می‌کند که تقریباً ۲۰۰۰ Region Proposal را ایجاد می‌کند. این proposal ها سپس به معماری CNN وارد می‌شوند تا به یکسری feature map برسیم. سپس این ویژگی‌ها برای طبقه‌بندی شیء موجود در Region Proposal وارد یک مدل SVM می‌شوند. یک مرحله اضافی شامل انجام یک رگرسیون برای Bounding box جهت localize کردن دقیق اشیاء موجود در تصویر است.

[1] Region
[2] Girshick, Ross, et al. "Rich feature hierarchies for accurate object detection and semantic segmentation." Proceedings of the IEEE conference on computer vision and pattern recognition. 2014.
[3] Region-based CNN
[4] Selective Search Algorithm

Region Proposals:

پیشنهادهای منطقه صرفاً مناطق کوچکتر تصویر هستند که احتمالاً شامل اشیایی است که ما در تصویر ورودی در جستجوی آن‌ها هستیم. برای کاهش پیشنهادات منطقه در R-CNN از یک الگوریتم greedy به نام جستجوی انتخابی استفاده می شود.

Selective Search Algorithm :

جستجوی انتخابی یک الگوریتم حریصانه است که مناطق تقسیم‌بندی شده کوچک‌تر را برای ایجاد پیشنهادهای منطقه‌ای ترکیب می‌کند. این الگوریتم یک تصویر را به عنوان ورودی می گیرد و در خروجی پیشنهادات منطقه را روی آن ایجاد می کند. این الگوریتم مزیتی نسبت به تولید پیشنهاد تصادفی دارد زیرا تعداد پیشنهادات را به حدود ۲۰۰۰ محدود می کند.

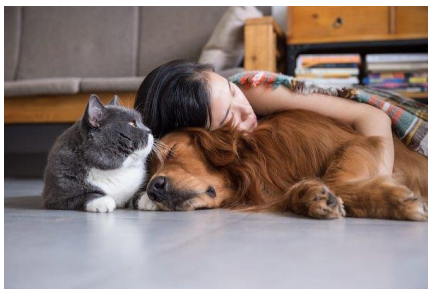
الف)

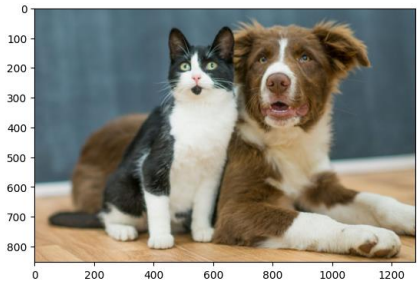
شبکه‌ای ساده مبتنی بر R-CNN برای حل مسئله تشخیص اشیا پیاده‌سازی کنید. مجموعه داده مدنظر یک مجموعه داده ساده شامل ۲۲۰۰ تصویر از گربه و سگ است. برای هر تصویر دو فایل وجود دارد:

- تصویر
- یک فایل xml حاوی اطلاعاتی مانند مختصات اشیاء

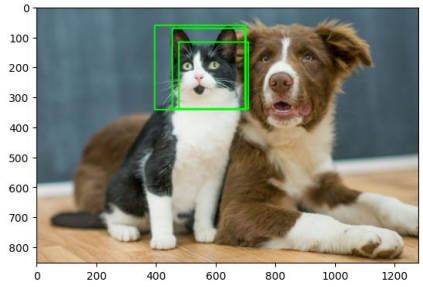
برای ساده تر شدن مسئله، فقط الگوریتمی را برای شناسایی مکان **گربه‌ها** در تصویر آموزش دهید. می‌توانید مجموعه داده را از اینجا دانلود کنید. همچنین به جای استفاده از SVM، کلاس خروجی را مستقیما با استفاده از یک لایه fully connected در انتهای CNN پیشبینی کنید(یک طبقه بند برای تشخیص cat در برابر no cat).

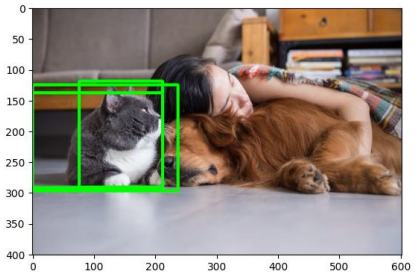
نتایج آموزش را گزارش کنید. پس از آموزش شبکه، عکس‌های زیر را برای تست شبکه استفاده کنید.





خروجی این مرحله باید شبیه به نتایج زیر باشد:





پیشنهاد می‌گردد روند زیر را برای پیاده‌سازی طی کنید:

مرحله ۱: آماده‌سازی داده‌های آموزش

الف) استخراج مقادیر **ground truth** مربوط به **bounding box**، **lable** و **id** های تصاویر را از فایل‌های **xml** (برای این کار می‌توانید از کتابخانه **xml** استفاده کنید)

ب) ساختن **region proposal** ها با استفاده از الگوریتم **selective search** (برای پیاده‌سازی این الگوریتم می‌توانید از کتابخانه opencv استفاده کنید.)

ج) چک کردن **region proposal** ها با مقادیر **ground truth** (برای این منظور از روش **IoU** یا همان **intersection over union** استفاده کنید که میزان **overlap** بین **proposal** های پیشبینی شده و **ground truth** را اندازه‌گیری می‌کند؛ اگر میزان **IoU** بیشتر از ۰.۵ باشد، آن نمونه را به عنوان نمونه مثبت درنظر بگیرید و درغیر این این صورت آن را به عنوان نمونه منفی درنظر بگیرید)

د) **crop** کردن تصاویر با استفاده از **region proposal** ها و ذخیره ان‌ها در یک فولدر به عنوان تصاویر آموزش

مرحله ۲: آموزش شبکه

همانطور که بیان شد، بر خلاف روند اصلی مقاله **R-CNN** که از یک **SVM** برای طبقه‌بندی استفاده می‌کند، در این سوال در انتهای شبکه **CNN** از یک لایه **fully connected** به عنوان طبقه بند برای این کار استفاده می‌کنیم. برای این منظور می‌توانید از یک مدل **pretrain** شده مانند **MobileNet** استفاده کرده و از یادگیری انتقالی استفاده کنید و فقط لایه(های) **fully connected** انتهایی را آموزش دهید.

نکات:

لطفا نکات گفته شده را به دقت مطالعه نمائید، در صورت عدم رعایت هر کدام از موارد گفته شده نمره کسر خواهد شد.

- مهلت تحویل این تمرین، یکشنبه ۱۹ آذر است.
- مهلت تحویل تمرین قابل تمدید نیست.
- انجام این تمرین به صورت یک نفره می باشد.
- تمرین امتیازی امکان تحویل با تاخیر ندارد.
- گزارش شما در فرآیند تصحیح از اهمیت ویژه ای برخوردار است. لطفاً تمامی نکات و فرض هایی که برای پیاده سازی ها و محاسبات خود در نظر می گیرید را در گزارش ذکر کنید.
- کدهای خود را به صورت عکس در داخل گزارش کپی نکنید و با فرمتی مناسب آن را در گزارش قرار دهید.
- داخل کدها کامنت های لازم را قرار دهید و تمامی موارد مورد نیاز برای اجرای صحیح کد را ارسال کنید.
- الزامی به ارائه توضیح جزئیات کد در گزارش نیست. اما باید نتایج بدست آمده را گزارش و تحلیل کنید.
- گزارش را در قالب تهیه شده که روی صفحه درس در سامانه eLearn بارگذاری شده، بنویسید. در صورت تمایل می توانید از Latex نیز برای نوشتن گزارش استفاده نمائید اما باید ساختار، زبان نوشتار و سایر موارد قالب اصلی را نیز رعایت کنید.
- در گزارش خود برای تصاویر زیرنویس و برای جداول هم بالانویس اضافه کنید.
- اگر بخشی از کد را از کدهای آماده اینترنتی استفاده می کنید که جزء قسمتهای اصلی تمرین نمی باشد، حتما باید لینک آن در گزارش و کد ارجاع داده شود، در غیر اینصورت تقلب محسوب شده و کل نمره تمرین را از دست می دهید. ولی محدودیتی در استفاده از منابع اینترنتی ندارید و در مواردی که در تمرین اشاره نشده می توانید از کدهای موجود استفاده کنید.
- تنها مجاز به استفاده از زبان برنامه نویسی Python و یکی از دو کتابخانه Tensorflow یا Pytorch برای پیاده سازی شبکه های عصبی هستید.
- هر گونه شباهت در گزارش و کدها، به منزله تقلب می باشد و نمره تمامی افراد شرکت کننده در آن تقلب برابر با صفر خواهد بود.
- لطفا فایل گزارش، کدها و سایر ضمائم مورد نیاز را با فرمت زیر در صفحه درس در سامانه eLearn بارگذاری نمائید.

HW_Extra1_[Lastname]_[StudentNumber].zip

- در صورت وجود هرگونه ابهام یا مشکل می‌توانید از طریق رایانامه زیر با دستیار آموزشی طراح تمرین در تماس باشید:

علی سبزه جو

ali.sabzejou@gmail.com