



## یادگیری عمیق

پاییز ۱۴۰۱  
استاد: دکتر فاطمی زاده

گردآورندگان: سعید رضوی، هلیا حاج کاظمی، ارشیا همت

تمرین سوم شبکه‌های عمیق کانولوشنی مهلت ارسال: جمعه ۱۷ آذر

- مهلت ارسال پاسخ تا ساعت ۲۳:۵۹ روز مشخص شده است.
- در طول ترم امکان ارسال با تاخیر پاسخ همه‌ی تمرین تا سقف ۶ روز و در مجموع ۲۰ روز، وجود دارد. پس از گذشت این مدت، پاسخ‌های ارسال شده پذیرفته نخواهند بود. همچنین، به ازای هر روز تأخیر غیر مجاز ۱۰ درصد از نمره تمرین به صورت ساعتی کسر خواهد شد.
- هم‌کاری و هم‌فکری شما در انجام تمرین مانعی ندارد اما پاسخ ارسالی هر کس حتما باید توسط خود او نوشته شده باشد. (دقت کنید در صورت تشخیص مشابهت غیرعادی برخورد جدی صورت خواهد گرفت.)
- در صورت هم‌فکری و یا استفاده از هر منابع خارج درسی، نام هم‌فکران و آدرس منابع مورد استفاده برای حل سوال مورد نظر را ذکر کنید.
- لطفا تصویری واضح از پاسخ سوالات نظری بارگذاری کنید. در غیر این صورت پاسخ شما تصحیح نخواهد شد.
- نتایج و پاسخ‌های خود را در یک فایل با فرمت zip به نام HW۳-Name-StudentNumber در سایت [Quera](#) قرار دهید. برای بخش عملی تمرین نیز لینک گیت‌هاب که تمرین و نتایج را در آن آپلود کرده‌اید قرار دهید. دقت کنید هر سه فایل نوتبوک تکمیل شده بخش عملی را در گیت‌هاب قرار دهید.
- لطفا تمامی سوالات خود را از طریق کوثرای درس مطرح بکنید (برای اینکه تمامی دانشجویان به پاسخ‌های مطرح شده به سوالات دسترسی داشته باشند و جلوی سوالات تکراری گرفته شود، به سوالات در بسترهای دیگر پاسخ داده نخواهد شد).
- دقت کنید کدهای شما باید قابلیت اجرای دوباره داشته باشند، در صورت دادن خطا هنگام اجرای کدتان، حتی اگر خطا بدلیل اشتباه تایپی باشد، نمره صفر به آن بخش تعلق خواهد گرفت.

### سوالات نظری (۳۰۰ نمره)

۱. (۲۵ نمره)  
یک مسئله دسته‌بندی با ۵ دسته، که دیتاست، شامل تصاویری به اندازه  $10 \times 10$  پیکسل میباشند، داریم. دو شبکه عصبی یک لایه را به صورت زیر در نظر بگیرید. توضیح دهید کدام یک انتخاب بهتری می باشد؟
  - یک لایه fully connected که ورودی آن، flatten (بردار شده) تصاویر دیتاست می باشد.
  - یک لایه کانولوشن که در آن ۵ فیلتر به اندازه  $10 \times 10$  داریم.
۲. (۲۵ نمره)  
فرض کنید دیتاستی داریم شامل تصاویر رنگی به اندازه  $128 \times 128$ . می‌خواهیم یک شبکه عصبی کانولوشن برای آن طراحی کنیم.
  - اندازه خروجی و تعداد پارامترهای لایه اول کانولوشن را محاسبه کنید اگر ۱۶ فیلتر  $5 \times 5$  با  $padding = 2$  و  $stride = 1$  داشته باشیم.

- فرض کنید هر لایه، ۳ قسمت شامل: کانولوشن، max pooling و تابع فعالسازی (Relu) را دارا می‌باشد، که لایه‌های کانولوشنی، هر کدام شامل ۱۶ فیلتر  $5 \times 5$  با  $stride = 1$  و  $padding = 2$  و لایه‌های max pooling همگی  $2 \times 2$  با  $stride = 2$  هستند. ۳ لایه با این مشخصات را پشت سر هم در نظر بگیرید. اندازه تنسور در لایه خروجی نهایی و تعداد پارامترهای این ۳ لایه را حساب کنید.

- فرض کنید هدف، حل یک مسئله classification، که شامل ۱۰ دسته هست، می‌باشد. تعداد کل پارامترهای شبکه را در این حالت حساب کنید.

- در این قسمت، با یک مفهوم مهم آشنا می‌شویم. Receptive field بیانگر این است که نورون خروجی، تحت تاثیر چه مقدار از نورون‌های ورودی می‌باشد. در حقیقت تعیین می‌کند هر نورون خروجی از چه ناحیه‌ای با چه اندازه‌ای از تنسور ورودی تاثیر می‌پذیرد. حال Receptive field را برای یک نورون خروجی لایه سوم (قبل از لایه fully connected) بررسی کنید. (برای فهم بهتر این مفهوم میتوانید به [این لینک](#) مراجعه کنید)

۳. (۲۵ نمره)

در این تمرین قصد داریم به بررسی دو شبکه‌ی معروف یعنی Densely Connected Convolutional Networks و U-Net بپردازیم. برای بررسی هرچه بهتر این دو شبکه بهتر است به لینک‌های زیر مراجعه نمایید.

• <https://arxiv.org/pdf/1505.04597.pdf>

• <https://arxiv.org/pdf/1608.06993.pdf>

در این تمرین دو نوع سوال وجود دارد، یکی از این سوالات، سوالات مفهومی‌ست که میزان تسلط شما بر روی شبکه‌های موجود را بررسی می‌کند و دوم سوالات محاسبه‌ای می‌باشد.

۱ - سوالات مفهومی مربوط به U-Net:

- ویژگی اصلی شبکه‌ی U-Net که آن را از یک شبکه کانولوشنی عادی متمایز می‌دارد چه می‌باشد و دلیل اینکه ما شاهد یک ساختار U شکل هستیم چه می‌باشد؟

- می‌دانیم که Skip connection ها نقشی پررنگ در این شبکه‌ها دارند، دلیل حضور این مورد را در شبکه‌ی U-Net بیان کنید.

- **سوال اضافه (دارای نمره‌ی اضافی جزئی):** چرا این نوع از اتصالات در تصاویر پزشکی دارای اهمیت بیشتر می‌باشند و چه کمکی به ما در دامنه‌ی تشخیص موارد پزشکی می‌کنند؟

۲ - سوالات محاسبه‌ای مربوط به U-Net:

- تصور کنید که ابعاد تصویر ورودی ما برای این شبکه  $256 \times 256$  می‌باشد. حال فرض می‌شود که در این معماری هر لایه در انکدر ابعاد را به نصف کاهش می‌دهد و در دیکدر دو برابر می‌کند. در پایین‌ترین لایه (عمیق‌ترین لایه) این معماری، فضای ویژگی ما چند پیکسل خواهد داشت؟

- در U-Net، فرض کنید انکودر دارای لایه‌هایی با ۶۴، ۱۲۸، ۲۵۶ و ۵۱۲ فیلتر است. اگر هر لایه کانولوشن از کرنال‌های  $3 \times 3$  استفاده بکند، تعداد پارامترهای لایه کانولوشن دوم انکدر را محاسبه کنید.

۳ - سوالات مفهومی DenseNet:

- تفاوت‌های اصلی ResNet's residual connections و DenseNet's dense connections را بیان کنید. درمورد هر کدام از موارد گفته نیز، توضیح مختصری بدهید.

- بیان کنید که DenseNet چگونه مشکل vanishing gradient را کاهش می‌دهد و مزیت محاسباتی آن چه می‌باشد؟

#### ۴- سوالات محاسبه‌ای DenseNet:

- در یک DenseNet با سه لایه در یک Dense Block اگر لایه اول ۶۴ فیچر مپ تولید کند، لایه دوم ۱۲۸ فیچر مپ و لایه سوم ۲۵۶ فیچر مپ تولید کند، لایه سوم چند فیچر مپ ورودی را دریافت خواهد کرد؟
- با در نظر گرفتن نرخ رشد  $k$  در DenseNet، اگر هر لایه  $k$  فیچر مپ جدید تولید کند و ورودی یک dense block دارای ۳۲ کانال باشد، اگر  $k = ۲۴$  باشد لایه سوم در بلوک چند کانال خروجی خواهد داشت؟

---

#### سوالات عملی (۳۰۰ نمره)

---

۱. (۱۰۰ نمره) همانطور که می‌دانید، در بسیاری از مدل‌های شبکه عمیق، از یک تابع ضرر (Loss function) برای آموزش مدل استفاده می‌کنند. نکته‌ای که وجود دارد این است که لزوماً این تابع ضرر، مختص به لایه آخر شبکه عصبی نیست و می‌شود از آن در لایه‌های میانی نیز استفاده کنیم. همچنین، در بسیاری از موارد برای بهبود آموزش، می‌توانیم از جمع وزن‌دار چند تابع ضرر به صورت همزمان استفاده کنیم. در ادامه‌ی سوال، با این موارد بیشتر آشنا می‌شویم.

در این سوال، از معماری یکی از شبکه‌های معروف کانولوشنی (مانند Alexnet، Resnet50 و...) که بر روی imagenet ترین شده به دلخواه استفاده می‌کنیم. هدف ما آموزش یک classifier برای دو کلاس هواپیما (airplane) و ماشین (automobile) از دیتاست cifar10 است. همانطور که میدانیم، این نوع شبکه‌ها، شامل یک لایه fully connected هستند که ورودی آن برداری به اندازه feature vector استخراج شده و خروجی آن به به اندازه تعداد کلاس‌ها است. مثلاً برای Alexnet، خروجی لایه‌ی یکی مانده به آخر، یک بردار ۴۰۹۶ تایی است که از طریق یک لایه fully connected به یک بردار ۱۰۰۰ تایی برای مسئله دسته‌بندی مپ شده است. از آنجایی که تعریف‌های ما برای استخراج بردار ویژگی همیشه ممکن است دارای نقص‌هایی باشند و برخی موارد در نظر گرفته نشده باشند، چنانچه داده‌های بسیار زیاد و متنوعی به شبکه‌هایی مانند AlexNet نشان داده شود چه بسا بردار ویژگی که به دست می‌آورند بهتر از آنهایی باشد که متخصصین تعریف می‌کنند. در مسائل مختلف دیده شده است که این بردار ویژگی‌ها حتی برای مسائلی که دسته‌بندی نیستند و یا دسته‌ها متفاوت از دسته‌های ImageNet هستند هم با معنی بوده و به نتایج خوب منجر می‌شوند. در ادامه شما ملزم به انجام موارد زیر هستید:

- یک شبکه کانولوشنی معروف به دلخواه انتخاب کنید، و با تغییر دادن لایه fully connected آن، شبکه را برای دسته‌بندی یک مسئله دو کلاسه، آماده کنید. (دقت کنید این دو دسته، دو کلاس هواپیما (airplane) و ماشین (automobile) از دیتاست cifar10 هستند).

(آ) شبکه کانولوشنی خود را با استفاده از وزن‌های از قبل آموزش داده شده و cross entropy loss آموزش دهید (فقط وزن‌های مربوط به لایه fully connected را مقداردهی اولیه کنید). نمودار دقت و loss را به ازای epoch های مختلف رسم کنید. دقت کنید این دقت و loss بر روی دیتاست ترین باید صورت بگیرد.

(ب) در نهایت، دقت مدل ترین شده بالا را بر روی داده تست حساب کنید.

- این بار با جای استفاده از cross entropy loss از triplet loss استفاده کنید. (برای آشنایی بیشتر با triplet loss می‌توانید از این **لینک** استفاده کنید). به موارد زیر دقت کنید :
  - (آ) به دلیل ماهیت triplet loss، کلاس مربوط به دیتاست باید توسط خودتان نوشته شود.
  - (ب) هدف در این قسمت این است که با استفاده از triplet loss ابتدا یک feature extractor خوب آموزش دهیم (دقت شود در این بخش آموزش، لایه fully connected دخیل نشده). پس از آموزش داده‌شدن feature extractor، حال لایه fully connected را با فریز کردن وزن لایه‌های قبلی، با cross entropy loss آموزش دهید.
  - (ج) نمودارهای خواسته شده در قسمت ۱ را برای این بخش نیز رسم کنید (هم برای آموزش feature extractor و هم برای آموزش لایه classifier)
  - (د) دقت نهایی این مدل را بر روی دیتاست تست حساب کنید.
  - (ه) تمام کارهای بالا را کافیست با استفاده از وزن‌های pre-trained انجام دهید و نیازی به مقداردهی اولیه نیست.
- این بار می‌خواهیم این دو تابع ضرر را همزمان در آموزش دخیل کنیم. برای اینکار، تابع ضرر زیر را در نظر بگیرید:

$$L_{total} = L_{triplet} + L_{cross-entropy}$$

- عمل backpropagation را بر روی  $L_{total}$  انجام دهید. به موارد زیر دقت کنید :
- (آ) برعکس قسمت قبل، که ابتدا مدل استخراج ویژگی آموزش داده می‌شد و سپس classifier، در اینجا کل مدل در حال ترین شدن است.
  - (ب) نمودارهای خواسته شده در قسمت ۱ را برای این بخش نیز رسم کنید.
  - (ج) دقت نهایی این مدل را بر روی دیتاست تست حساب کنید.
  - (د) تمام کارهای بالا را کافیست با استفاده از وزن‌های pre-trained انجام دهید و نیازی به مقداردهی اولیه نیست.

۲. (۱۰۰ نمره)

این تمرین شامل دو بخش می‌باشد؛ هدف تمرین شناخت کامل موضوع Deformable Convolution می‌باشد؛ شما در این تمرین ابتدا به چند سوال تئوری پاسخ خواهید داد و پس از آن به سراغ پیاده‌سازی این شبکه‌های کانولوشنی می‌روید. لطفاً به این نکته توجه کنید که پیاده‌سازی‌های انجام شده باید از ابتدا (From Scratch) باشد. به این صورت که شما می‌توانید از فریم‌ورک‌های پیاده‌سازی مدل‌های عمیق نظیر Pytorch استفاده نمایید، اما اجازه‌ی استفاده از پیاده‌سازی آماده این مورد را ندارید.

بخش اول - سوالات تئوری مربوط به تمرین عملی:

- (آ) تفاوت بین شبکه‌های کانولوشنی عادی و شبکه‌های کانولوشنی Deformable را از نظر grid sampling مقایسه کرده و نتایج مقایسه خود را بنویسید.
- (ب) شبکه‌های Deformable چگونه می‌توانند انعطاف‌پذیری را در Geometric transformation در تصاویر را به وجود آورند.
- (ج) به عقیده شما چرا شبکه‌های کانولوشنی ساده در مواجهه با تصاویری که آبنجکت‌های تصویر دارای تغییر یا چرخش فضایی زیادی می‌باشد دچار مشکلات جدی‌ای می‌شود؟
- (د) چگونه آفست‌های موجود در Deformable Convolution محاسبه می‌شوند؟

بخش دوم - سوالات عملی:  
در ابتدا به این نکته توجه کنید که دیتاست مربوط به این سوال دیتاست MS COCO می باشد. شما از طریق لینک های زیر می توانید اطلاعات بیشتری در مورد این دیتاست بدست آورید (نیاز نیست از همه ی دیتاست برای آموزش شبکه استفاده کنید)

- [paperswithcode.com/dataset/coco](https://paperswithcode.com/dataset/coco)
- [cocodataset.org/#home](https://cocodataset.org/#home)

بخش سوم - گزارشات مربوط به پیاده سازی:  
موارد زیر را براساس تغییر نقاط نمونه گیری برای شبکه های کانولوشنی Deformable و عادی گزارش کنید:

- دقت داده های آموزشی و آزمایشی.
  - خطای مدل برای داده های آموزشی و آزمایشی
  - مدت زمان اجرا برای هرکدام از موارد خواسته شده.
- همچنین می توانید جهت آشنایی بیشتر با ایده Deformable Convolution و کارهای مهم تحقیقاتی که بر پایه این ایده ارائه شده اند، به ضمیمه این تمرین مراجعه نمایید.

۳. (۱۰۰ نمره)

در این تمرین می خواهیم شبکه ای آموزش دهیم که بتواند تغییرات شامل **displacement** (جابه جایی)، **rotation** و **scaling** بین دو تصویر را تشخیص دهد. برای انجام این تمرین فایل `detect_change.ipynb` را تکمیل کنید. توجه داشته باشید که در گزارش خود نحوه ی آماده سازی و ساخت دیتاست خود را توضیح دهید.