



## یادگیری ژرف

نیم‌سال دوم ۹۹-۱۰۰

مدرس: حمید بیگی

تمرین سری دوم

یادگیری ژرف

زمان تحویل: ۲۷ فروردین

نکات زیر را رعایت کنید:

فایل گزارش را به همراه تمامی کدها در یک فایل فشرده و با عنوان HW2\_STD# در سایت Quera.ir بارگذاری نمایید.

نوت بوک های ارسالی حتما اجرا شده و نتایج نهایی در آن باقی بماند و حذف خروجی‌ها باعث کم شدن نمره خواهد شد.

سوالات خود را از طریق Piazza مطرح کنید.

## مسئله‌ی ۱. Network Design (۱۵ نمره)

چند لایه‌ی اولیه‌ی یک شبکه‌ی عصبی کانولوشنی به صورت زیر است که در لایه‌های کانولوشنی مقادیر به ترتیب برابر با تعداد کانال‌های ورودی، تعداد کانال‌های خروجی، سایز کرنل، stride و dilation هستند.

Convolution(3, 64, 5 \* 5, a, b)

Convolution(3, 64, 3 \* 3, c, d)

Max-Pooling(3 \* 3)

(آ) سایز خروجی و تعداد پارامترها را برای هر لایه در صورتی که ورودی، یک تصویر RGB با سایز  $32 \times 32$  باشد محاسبه کنید. (مقادیر a, b, c, d را ۱ در نظر بگیرید)

(ب) به ازای هر یک از نرون‌های خروجی لایه‌ی آخر تعداد پیکسل‌های پوشش داده شده را به ازای مقادیر زیر بدست بیاورید. سپس بررسی کنید که با تغییر مقادیر dilation و stride این تعداد چه تغییری می‌کنند. دلایل استفاده از dilation و stride را توضیح دهید.

a = 1 , b = 1 , c = 1 , d = 1

a = 1 , b = 1 , c = 2 , d = 1

a = 1 , b = 1 , c = 2 , d = 2

(ج) مزایا و معایب لایه‌های Max-Pooling را در شبکه‌های کانولوشنی بیان کنید.

(د) در مورد ساختار Grouped convolution و Depthwise convolution توضیح دهید و این دو را مقایسه کنید.

## مسئله‌ی ۲. Convolution Back-propagation (۱۰ نمره)

کانولوشن ورودی X با سایز  $(N, C, i_h, i_w)$  که N تعداد نمونه های ورودی و در فیلتر W با سایز  $(F, C, f_h, f_w)$  و stride=s خروجی O با سایز  $(N, C, O_h, O_w)$  را می دهد و برای  $C = 1$  ,  $N = 1$  و  $F = 1$  داریم :

$$O = \begin{bmatrix} o_{(1,1)} & \dots & x_{(1,o_w)} \\ \vdots & \ddots & \vdots \\ o_{(o_h,1)} & \dots & x_{(o_h,o_w)} \end{bmatrix}, W = \begin{bmatrix} w_{(1,1)} & \dots & w_{(1,f_w)} \\ \vdots & \ddots & \vdots \\ w_{(f_h,1)} & \dots & w_{(f_h,f_w)} \end{bmatrix}, X = \begin{bmatrix} x_{(1,1)} & \dots & x_{(1,i_w)} \\ \vdots & \ddots & \vdots \\ x_{(i_h,1)} & \dots & x_{(i_h,i_w)} \end{bmatrix}$$

فرض کنید مشتق خروجی لایه کانولوشنی را داریم. در صورتی که  $pad = 0$  و  $stride = 1$  و بایاس نداشته باشیم:

- (آ) ماتریس  $O$  خروجی کانولوشن را به صورت پارامتری به دست آورید.  
 (ب) با مشتق گیری از روابط به دست آمده، مشتقات  $\frac{\partial L}{\partial W}$  و  $\frac{\partial L}{\partial X}$  را به دست آورید.

### مسئله ۳. Transposed Convolution (۱۵ نمره)

کانولوشن بین ورودی  $X$  و فیلتر  $W$  به صورت زیر است:

$$X = \begin{bmatrix} x_{(0,0)} & x_{(0,1)} & x_{(0,2)} \\ x_{(1,0)} & x_{(1,1)} & x_{(1,2)} \\ x_{(2,0)} & x_{(2,1)} & x_{(2,2)} \end{bmatrix}, W = \begin{bmatrix} w_{(0,0)} & w_{(0,1)} \\ w_{(1,0)} & w_{(1,1)} \end{bmatrix}$$

می توان عملیات کانولوشن را به صورت ضرب ماتریسی نوشت که ورودی و خروجی را به صورت یک بردار و فیلتر به صورت یک ماتریس در نظر گرفت. ورودی  $X$  را به صورت بردار زیر نمایش می دهیم:

$$X = [x_{(0,0)} \quad x_{(0,1)} \quad \dots \quad x_{(2,2)}]$$

(آ) عملیات کانولوشن بالا با  $s = 1$  را به صورت ضرب ماتریسی  $Y = AX$  بنویسید.

(ب) با استفاده از نمایش ماتریسی بالا میتوان گرادیان پس انتشار نسبت به ورودی  $X$  را به صورت  $\frac{\partial L}{\partial X} = A^T \frac{\partial L}{\partial Y}$  نمایش داد.

عملیات transposed convolution را میتوان مشابه عملیات گرادیان پس انتشار نسبت به ورودی در نظر گرفت (تشابه فقط در نوع تبدیل به بردار از خروجی به ورودی) و میتوان عملیات را به صورت کانولوشن مستقیم در نظر گرفت که در آن ماتریس  $A^T$  به عنوان فیلتر به حساب می آید.  
 فرض کنید  $X$  خروجی یک عملیات مستقیم کانولوشن با ورودی  $i$  با سایز فیلتر  $W$  و  $stride$  برابر ۲ است. خروجی transposed convolution را برای ورودی های زیر به دست آورید.

$$X = \begin{bmatrix} 3 & 1 \\ 6 & 2 \end{bmatrix}, W = \begin{bmatrix} 1 & 3 \\ 4 & 2 \end{bmatrix}$$

(ج) در مورد Masked Convolution کاربرد و محدودیت آن تحقیق کنید.

## مسئله‌ی ۴. Object Detection (۱۵ نمره)

یکی از کاربردهای شبکه‌های کانولوشنی آشکارسازی اشیاء<sup>۱</sup> است. در این بخش هدف بررسی روش YOLO است که یکی از روش‌های آشکارسازی می‌باشد. مقاله‌ی مربوطه را مطالعه و به سوالات زیر پاسخ دهید.

- (آ) نحوه‌ی آشکارسازی توسط این روش را به‌طور خلاصه بیان کنید و با روش RCNN مقایسه کنید.
- (ب) هدف از انجام Non-Maximum-Suppression در این روش چیست. معایب آن را بیان کنید.
- (ج) راهکارهایی برای رفع ایرادات قسمت قبل بیان کنید. (ایده‌ی خود را به صورت کامل شرح دهید و در صورت استفاده از ایده مقالات دیگر، نام مقاله را ذکر کنید) (امتیازی)

## مسئله‌ی ۵. ResNet (۳۰ نمره + ۵)

در یک شبکه عصبی، ورودی لایه  $(l + 1)$  به صورت  $x_{(l+1)} = F(x_l)$  به دست می‌آید  $F$  که به صورت خلاصه بیانگر ضرب وزن‌ها و اعمال تابع فعالساز است.

- (آ) حال فرض کنید ورودی لایه  $(l + 1)$  را بدین صورت محاسبه می‌کنیم:  $x_{(l+1)} = F(x_l) + x_l$  اگر شبکه در مجموع  $L$  لایه داشته باشد رابطه  $x_L$  بر حسب  $x_l$  را به دست آورید

(ب) اگر تابع هزینه این شبکه عصبی را  $E$  بنامیم،  $\frac{\partial E}{\partial x_l}$  را به دست آورید.

- (ج) این معماری مشابه معماری ResNet است که برای آموزش شبکه‌های عصبی عمیق پیشنهاد شده است. با توجه به رابطه‌ای که در قسمت قبل به دست آوردید، توضیح دهید تفاوت عبارت backpropagation در این معماری در برابر معماری معمولی چیست که امکان عمیق‌تر کردن شبکه‌های عصبی را فراهم کرده است؟

(د) در این قسمت، قصد داریم با پیاده‌سازی شبکه ResNet به صورت عملی آشنا شویم. برای این منظور از دیتاست CIFAR۱۰ استفاده می‌کنیم. برای پیاده‌سازی، با استفاده از Pytorch معماری مشخص شده در نوت‌بوک را پیاده‌سازی کرده و با آموزش شبکه بر روی دیتاست روند تغییرات تابع هزینه در هر epoch را گزارش کنید.

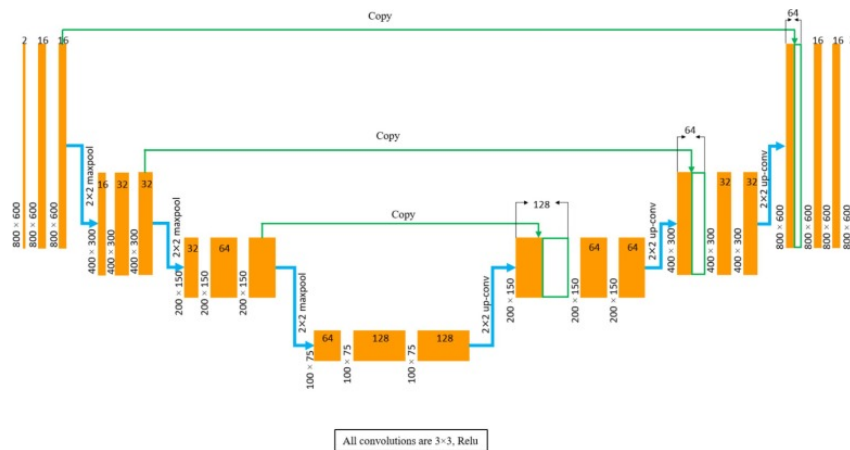
- (ه) در هر epoch دقت هر کلاس بر روی دیتای تست گزارش شود و نمودار آن به همراه نمودار تابع هزینه در tensorboard نمایش داده شود. (امتیازی)

---

<sup>1</sup>Object Detection

## مسئله‌ی ۶. U-Net (۲۰ نمره + ۱۰)

کاربرد شبکه‌ی U-Net (شکل ۱) در بخش بندی معنایی<sup>۲</sup> است. برای پیاده‌سازی این شبکه، نوت بوک داده شده مربوط به این بخش را کامل کنید و خروجی‌ها، درصد صحت (میزان صحت را با IoU محاسبه کنید) و روند خطا را گزارش کنید. توجه داشته باشید که درصد دقت و مشاهده‌ی خروجی شبکه در ارزیابی نهایی تاثیر داشته و ایده‌هایی برای بهبود شبکه و افزایش دقت دارای نمره‌ی امتیازی خواهد بود. (در صورت پیاده‌سازی ایده، در گزارش، روش خود را توضیح داده و خروجی‌ها و دقت را با روش U-Net مقایسه کنید)



شکل ۱: U Net

<sup>2</sup>Semantic Segmentation