



به نام خدا
درس مبانی یادگیری عمیق
پاسخنامه تمرین سری چهارم
استاد درس : دکتر مرضیه داوودآبادی
دستیاران : الناز رضایی، سید محمد موسوی، کمیل فتحی
دانشگاه علم و صنعت ایران، دانشکده مهندسی کامپیوتر
نیمسال اول تحصیلی ۱۴۰۲ - ۱۴۰۳

پاسخ های مناسب برای هر سوال، لزوماً یکتا نیستند

۱. در این سؤال قصد داریم تا مدلی طراحی کنیم که توانایی دسته‌بندی داده‌های موجود در دیتاست *MNIST* و پیدا کردن کمترین میزان خطا را به صورت خودکار داشته باشد. در صورت پیدا کردن بهترین مقدار برای هر *Hyper parameter*، مدل بهینه و مناسبی ساخته خواهد شد که توانایی دسته‌بندی داده‌ها با کمترین میزان خطا را دارد. حال گام‌های الف تا ج را طی کنید تا شبکه مناسب را طراحی نمایید (۱۵ نمره).

الف) طراحی شبکه هم‌گشتی^۱ با استفاده از *KerasTuner* :

- ابزار *KerasTuner* را معرفی کنید.
- توضیح دهید چگونه می‌توان از این ابزار برای بهینه‌سازی شبکه هم‌گشتی جهت دسته‌بندی داده‌ها استفاده کرد.
- *Tuner* موجود در *KerasTuner* را معرفی کنید. شما در پیاده‌سازی خود از کدام یک استفاده می‌نمایید؟ چرا؟

ب) استفاده از *KerasTuner* بر روی دیتاست *MNIST* :

- دیتاست *MNIST* را معرفی کنید.
- توضیح دهید چگونه می‌توان از شبکه هم‌گشتی بهینه‌سازی شده با *KerasTuner* برای دسته‌بندی تصاویر این دیتاست استفاده کرد.

¹Convolutional

- اهمیت و تأثیر استفاده از *dropout* و *pooling* در بهبود عملکرد شبکه را توضیح دهید (۵ نمره امتیازی).

ج) پیاده‌سازی شبکه موردنظر:

- باتوجه به توضیحات داده شده، مدل پیشنهادی خود را که باید شامل تعدادی لایه *Convolutional* و *Fully – connected* باشد را پیاده‌سازی کنید. در صورت نیاز می‌توان از لایه *pooling* و *dropout* نیز استفاده نمود. در جدول زیر بخشی از اطلاعات موردنیاز برای *tune* مدل ارائه شده است همچنین باتوجه به دانش خود مدل را تا حد امکان بهبود دهید. اما موارد زیر را حتماً با استفاده از *KerasTuner*، تنظیم و نتایج را گزارش و تحلیل نمایید.

<i>HyperParameter</i>	<i>MaxLayer/value</i>
<i>Convolutional</i>	۵
<i>Dense</i>	۵
<i>Filters</i>	۲۵۶
<i>Neurons</i>	۲۵۶
<i>Learningrate</i>	۰.۰۰۱
<i>Optimizer</i>	–

- نحوه انتخاب اندازه فیلترها در لایه‌های *Convolutional* را توضیح دهید.
- چگونه می‌توان از *pooling* و *dropout* در شبکه هم گشتی برای جلوگیری از *overfitting* و افزایش دقت استفاده کرد؟ (۵ نمره امتیازی)

به نوتبوک [DL_HW4_Q1.ipynb](#) مراجعه کنید. این پاسخ برگرفته از پاسخ آقای فرزاد رحمانی است.

۲. به فایل [medical.ipynb](#) مراجعه کرده و باتوجه به توضیحات قسمت های مربوطه را تکمیل کنید (۲۰ نمره).

به نوتبوک [medical_Final.ipynb](#) مراجعه کنید.

۳. در فایل *consumption.ipynb* یک کد جهت پیش بینی مصرف گاز قرار داده شده است که ورودی ها و خروجی های آن به شکل زیر است:

ورودی ها: تاریخ، ساعت، ماه میلادی و دما

خروجی: مصرف گاز

با وجود استفاده از *lstm* دقت r^2 منفی به دست می آید. علت و مشکل کد را پیدا کرده و پیشنهادی برای اصلاح آن بیان کنید (۱۰ نمره).

به نوتبوک *consumption.ipynb* مراجعه کنید.

۴. الف) تعریفی مختصر از شبکه های هم گشتی و شبکه های بازگشتی بیان کرده و بگویید هریک برای چه کاربردها و مسائلی بهتر عمل می کند؟ چرا؟ (۵ نمره)

شبکه های هم گشتی^۲ و شبکه های بازگشتی^۳ دو نوع مهم از شبکه های عصبی عمیق هستند که در تعداد زیادی از کاربردهای یادگیری ماشین و پردازش سیگنال مورد استفاده قرار می گیرند. شبکه های هم گشتی:

- تعریف مختصر:

CNN ها به خصوص برای پردازش داده های ماتریسی مانند تصاویر مورد استفاده قرار می گیرند. این شبکه ها از لایه های کانولوشن (*Convolutional Layers*) برای شناسایی ویژگی های مختلف در داده ها استفاده می کنند.

- کاربردها و مسائل:

- تشخیص الگو در تصاویر: *CNN* ها برای تشخیص الگوها و اجزای مختلف تصاویر (مثل چهره ها در تصاویر) بسیار موثر هستند.
- تصویربرداری و پردازش تصاویر: در بسیاری از وظایف مرتبط با تصاویر مانند تصویربرداری پزشکی، خودروهای هوشمند و غیره کاربرد دارند.
- ترجمه ماشینی تصویر به متن: *CNN* ها برای تبدیل تصاویر به متون مفهومی مانند توضیحات یا شرح ها به کار می روند.

²CNNs

³RNNs

شبکه‌های بازگشتی:

- تعریف مختصر:

RNN ها برای مدل‌سازی داده‌های با ترتیب زمانی یا دنباله‌ای مناسب هستند. این شبکه‌ها دارای یک حلقه بازگشتی هستند که اجازه می‌دهد اطلاعات از مراحل گذشته در آموزش و پیش‌بینی در زمان‌های آتی مورد استفاده قرار گیرد.

- کاربردها و مسائل:

□ پردازش زبان طبیعی^۴: RNN ها برای مسائل مانند ترجمه ماشینی، تولید متن، و پردازش زبان طبیعی که وابستگی به ساختار زمانی دارند، به کار می‌روند.

□ تولید موسیقی: در زمینه تولید موسیقی یا ساخت قطعات صوتی، RNN ها می‌توانند دنباله‌های صوتی متفاوت ایجاد کنند.

□ پیش‌بینی سری‌های زمانی: RNN ها برای پیش‌بینی سری‌های زمانی مانند قیمت‌های بازار یا داده‌های متغیر در زمان به کار می‌روند.

هر یک از این دو نوع شبکه بر اساس خصوصیات خودشان برای وظایف مختلف مناسب هستند و انتخاب بین آنها به موارد خاص هر مسئله و نوع داده‌های مورد بررسی وابسته است.

ب) این شبکه‌ها را از لحاظ تعداد پارامتر و قابلیت موازی سازی با یکدیگر مقایسه کنید (۵ نمره).
پاسخ را می‌توانید در شکل ۱ ببینید.

۵. الف) شبکه هم گشتی زیر را در نظر بگیرید. فرض کنید ورودی یک تصویر رنگی با اندازه ۲۵۶ در ۲۵۶ است. اندازه خروجی و تعداد پارامترها را برای هر لایه به دست آورید (۵ نمره).

Layer1: Conv(64, (3,3), stride=1, padding='same')

Layer2: Dilated-Conv(32, (5,5), stride=2, dilation rate=2, padding='valid')

Layer3: Max-pool (size=(2,2), stride=2)

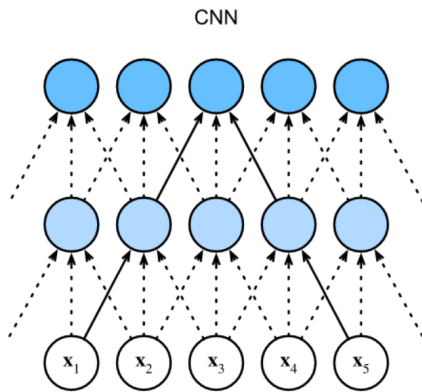
Layer4: Conv(128, (3,3), stride=1, padding='same')

Layer5: Dilated-Conv(64, (5,5), stride=2, dilation rate=4, padding='valid')

Layer6: Max-pool (size=(2,2), stride=2)

⁴NLP

• لایه کانولوشنی با ابعاد k :

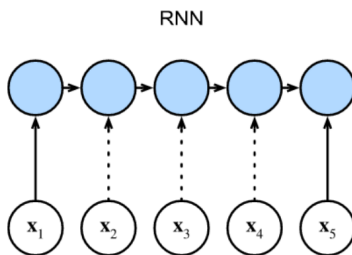


- هزینه محاسباتی $O(knd^2)$ است
- این محاسبات به صورت کاملاً موازی قابل انجام است که معادل با $O(1)$ عملیات متوالی است
- حداکثر طول مسیر $O(n/k)$ است
- با استفاده از stride یا dilated convolution می‌توان حداکثر طول مسیر را کاهش داد

(آ) لایه هم‌گشتی

• لایه بازگشتی:

- هزینه محاسباتی $O(nd^2)$ است
- این محاسبات باید به صورت متوالی انجام شوند که معادل با $O(n)$ عملیات متوالی است
- حداکثر طول مسیر $O(n)$ است



(ب) لایه بازگشتی

شکل ۱: مقایسه شبکه های هم‌گشتی و بازگشتی

Layer7: Conv(256, (3,3), stride=1, padding='same')

Layer8: Dilated-Conv(128, (5,5), stride=2, dilation rate=8, padding='valid')

Layer9: Max-pool (size=(2,2), stride=2)

در لایه ۸ مقدار $dilation\ rate$ به اشتباه ۸ درج شده بود که مقدار درست آن ۲ است. بدین جهت از همگی عذرخواهی می‌کنیم. نمره کامل به کسانی تا لایه ۷ ام را درست محاسبه کرده باشند، تعلق می‌گیرد.

Layer1

• *parameters* :

$$64(3 \times 3 \times 3 + 1) = 1792$$

• *Output* :

$$\left(\frac{256 - 3 + 2 \times 1}{1} + 1\right) \times \left(\frac{256 - 3 + 2 \times 1}{1} + 1\right) \times 64 = 256 \times 256 \times 64$$

برای محاسبه پارامترهای لایه *dilated_conv* مانند محاسبه پارامترهای لایه *conv* عمل می‌کنیم اما برای محاسبه ابعاد خروجی باید *dilated_rate* را نیز در نظر بگیریم. به عنوان مثال وقتی اندازه فیلتر ۵ در ۵ است و *dilated_rate* نیز برابر با ۲ است، به جای یک پنجره ۵ در ۵ یک پنجره ۹ در ۹ روی ورودی اعمال می‌شود. *padding=valid* یعنی *padding* نداریم.

Layer2

• *parameters* :

$$32(5 \times 5 \times 64 + 1) = 51,332$$

• *Output* :

$$\left(\frac{256 - 9 + 2 \times 0}{2} + 1\right) \times \left(\frac{256 - 9 + 2 \times 0}{2} + 1\right) \times 32 = 125 \times 125 \times 32$$

Layer3

• *parameters* :

$$0$$

• *Output* :

$$\left(\frac{125 - 2 + 2 \times 0}{2} + 1\right) \times \left(\frac{125 - 2 + 2 \times 0}{2} + 1\right) \times 32 = 63 \times 63 \times 32$$

Layer4

:parameters •

$$128(3 \times 3 \times 32 + 1) = 36,992$$

:Output •

$$(\frac{63 - 3 + 2 \times 1}{1} + 1) \times (\frac{63 - 3 + 2 \times 1}{1} + 1) \times 64 = 63 \times 63 \times 128$$

Layer5

:parameters •

$$64(5 \times 5 \times 128 + 1) = 204,864$$

:Output •

$$(\frac{63 - 17 + 2 \times 0}{2} + 1) \times (\frac{63 - 17 + 2 \times 0}{2} + 1) \times 32 = 24 \times 24 \times 64$$

Layer6

:parameters •

$$0$$

:Output •

$$(\frac{24 - 2 + 2 \times 0}{2} + 1) \times (\frac{24 - 2 + 2 \times 0}{2} + 1) \times 64 = 12 \times 12 \times 64$$

Layer7

:parameters •

$$256(3 \times 3 \times 64 + 1) = 147,712$$

• *Output* :

$$\left(\frac{12-3+2 \times 1}{1}+1\right) \times\left(\frac{12-3+2 \times 1}{1}+1\right) \times 256=12 \times 12 \times 256$$

Layer8

• *parameters* :

$$128(5 \times 5 \times 256+1)=519,328$$

• *Output* :

$$\left(\frac{12-9+2 \times 0}{2}+1\right) \times\left(\frac{12-9+2 \times 0}{2}+1\right) \times 128=3 \times 3 \times 128$$

Layer9

• *parameters* :

$$0$$

• *Output* :

$$\left(\frac{3-2+2 \times 0}{2}+1\right) \times\left(\frac{3-2+2 \times 0}{2}+1\right) \times 128=2 \times 2 \times 128$$

ب) در $2Dconv$ چه میزان $padding$ باید اضافه شود تا ابعاد خروجی تغییر نکنند؟ (پاسخ را به صورت رابطه ای از سایز فیلتر یعنی f بیان کنید)(۵ نمره).

$$\frac{f-1}{2}$$

۶. به سوالات زیر پاسخ دهید(۱۵ نمره).

الف) کدام یک از عبارات زیر در مورد نرمال سازی دسته‌ای^۵ درست است؟

- نرمال سازی دسته ای تنها پردازش یک دسته را سریع تر می کند و زمان آموزش را کاهش می دهد و درعین حال تعداد به روزرسانی ها را ثابت نگه می دارد. این به شبکه اجازه می دهد تا زمان مشابهی را صرف انجام به روزرسانی های بیشتر کند تا به حداقل برسد.
غلط: دلیلی وجود ندارد که فرض کنیم نرمال سازی دسته ای پردازش یک دسته را سریع تر می کند. برعکس، ما می توانیم انتظار داشته باشیم که نرمال سازی دسته ای، هر تکرار آموزش را کندتر کند، زیرا به پردازش اضافی در طول فرایند رو به جلو و ابرپارامترهای اضافی در حین انتشار به عقب نیاز دارند. همانطور که گفته شد.
- نرمال سازی دسته ای توزیع خروجی را نرمال می کند تا در ابعاد یکنواخت تر باشد.
درست: این اساساً هدف نرمال سازی دسته ای است، یعنی نرمال سازی ورودی های هر لایه (که به نوبه خود خروجی های لایه قبل از آن هستند) با توجه به میانگین و واریانس مقادیر در مینی دسته فعلی. همانطور که در مطالب درسی دیده می شود.
- به شبکه اجازه می دهد تا وزن های ما را به مقادیر کوچک تر نزدیک به صفر مقداردهی کند.
غلط: زیرا ما به سادگی اثر مقدار اولیه وزن خود را محدود می کنیم و سپس ورودی هر لایه را نرمال می کنیم و تاثیر وزن های اولیه تعیین شده را ضعیف می کنیم اما این به معنای "اجازه دادن به شبکه برای مقداردهی اولیه وزن ما به مقادیر کوچکتر نزدیک به صفر" نیست.

ب) کد موجود در فایل `forward_batchnorm.py` را به کمک `numpy` تکمیل کنید.

به فایل `forward_batchnorm.py` مراجعه کنید.

ج) نقش ابر پارامتر ϵ در نرمال سازی دسته ای چیست؟ (۵ نمره امتیازی)

از تقسیم بر ϵ برای ویژگی های با واریانس ϵ جلوگیری می کند.

د) استفاده از نرمال سازی دسته ای با اندازه یک چه مشکل یا مشکلاتی دارد؟

به طور خاص برای این سوال، اگر ما فقط یک اندازه کوچک ۱ داشته باشیم، خروجی از لایه BN همیشه ϵ خواهد بود ($x - \mu$ برابر ϵ است) و آن را در γ ضرب می کنیم که همچنان ϵ خواهد بود، و β را اضافه می کنیم که یک پارامتر "بایاس" است که به ϵ مقداردهی اولیه می شود، بنابراین ما نمی توانیم ویژگی های معنی دار را یاد بگیریم.

⁵Batch Normalization

ه) یک شبکه کاملاً متصل با ۱۰ ورودی و ۲۰ خروجی را در نظر بگیرید که نرمال سازی دسته ای به آن اضافه می کنیم. تعداد پارامترهای قابل آموزش را محاسبه کنید.

برای FC : 20×10

برای بایاس: ۲۰

برای $batch_norm$: 20×2

مجموع: ۲۶۰

۷. در این سوال قصد داریم الگوریتم $Grad - CAM$ را بر روی مجموعه داده $MNIST$ پیاده سازی نماییم. برای این کار مراحل زیر را انجام دهید (۲۰ نمره).

- ابتدا به کمک api فراهم شده در کتابخانه $Keras$ مجموعه داده $MNIST$ را بارگذاری نمایید.
- داده های آموزشی را $shuffle$ کنید و ابعاد داده ها را چاپ کنید.
- ۱۰ تصویر نخست موجود در مجموعه داده آموزشی را به همراه برچسب آن ها نمایش دهید.
- مقادیر پیکسل ها را به محدوده ۰ تا ۱ تغییر دهید.
- برچسب ها را به نمایش $categorical$ تغییر دهید.
- یک شبکه عصبی با ویژگی های زیر طراحی و آموزش دهید:

□ لایه $Conv2D$ با ۳۲ فیلتر (۳، ۳) و تابع فعال سازی $ReLU$ با پارامتر $padding$ برابر $same$.

□ لایه $MaxPooling2D$ با ابعاد پنجره (۲، ۲).

□ لایه $Conv2D$ با ۶۴ فیلتر (۳، ۳) و تابع فعال سازی $ReLU$ با پارامتر $padding$ برابر $same$.

□ لایه $MaxPooling2D$ با ابعاد پنجره (۲، ۲).

□ لایه $Conv2D$ با ۶۴ فیلتر (۳، ۳) و تابع فعال سازی $ReLU$ با پارامتر $padding$ برابر $same$.

□ لایه $MaxPooling2D$ با ابعاد پنجره (۲، ۲).

□ لایه $Flatten$.

□ لایه $Dense$ با ۱۲۸ واحد و تابع $ReLU$.

□ لایه *Dense* با ۱۰ واحد و تابع *Softmax*.

- مدل ساخته شده را با مشخصات زیر آموزش دهید:

□ تابع ضرر: *CategoricalCrossentropy*

□ بهینه‌ساز: *Adam*

□ تعداد *Epoch*: ۱۵

□ اندازه *Batch*: ۶۴

- الگوریتم *Grad - CAM* را بر روی آخرین لایه هم گشتی اجرا کرده و خروجی آن را برای ۱۰ تصویر نمونه نمایش دهید.

- خروجی الگوریتم را تحلیل کنید و نتیجه‌ی استفاده از *Grad - CAM* در تشخیص ویژگی‌های مهم در تصاویر *MNIST* را بررسی کنید.

به نوتبوک [DL_HW4_Q7.ipynb](#) مراجعه کنید. این پاسخ برگرفته از پاسخ آقای فرزاد رحمانی است.