



دانشگاه صنعتی اصفهان  
دانشکده مهندسی برق و کامپیوتر

## درس مبانی هوش محاسباتی

تکلیف کامپیوتری دوم

تاریخ تحویل: ۲۵ اردیبهشت

در این تکلیف قصد داریم یک مساله طبقه‌بندی ۵ کلاسه را با استفاده از شبکه‌های عصبی کانولوشن<sup>۱</sup> و همچنین روش یادگیری انتقالی<sup>۲</sup> حل نموده و نتایج این دو روش را با یکدیگر مقایسه نماییم. بدین منظور از زیر مجموعه‌ای از مجموعه داده MIO-TCD که شامل سه مجموعه داده می‌باشد که عبارتند از: مجموعه داده آموزشی<sup>۳</sup>، مجموعه داده ارزیابی<sup>۴</sup> و مجموعه داده آزمون<sup>۵</sup>. هر مجموعه داده خود شامل ۵ پوشه می‌باشد که همان ۵ کلاس هستند. تعداد تصاویر برای هر کلاس در هر مجموعه مطابق جدول زیر است:

class	train	valid	test
articulated_truck	3000	1000	1000
background	3000	1000	1000
bus	3000	1000	1000
car	3000	1000	1000
work_van	3000	1000	1000
Sum	15000	5000	5000

سطر آخر جدول تعداد کل داده‌های آموزشی، ارزیابی و آزمون را نشان می‌دهد. همه تصاویر نیز با فرمت jpg می‌باشند. ابعاد تصاویر متفاوت می‌باشند. قابل ذکر است به دلخواه می‌توانید رزولوشن تصاویر را تغییر دهید (در صورت تغییر دادن، این تغییر را در توضیحات خود در فایل ipynb ذکر کنید). اطلاعات بیشتر در مورد این مجموعه داده و نمونه‌های تصاویر آن را در این [لینک](#) می‌توانید مطالعه کنید.

برای ارزیابی و محاسبه خطا و بقیه موارد ابتدا باید نتایج خود را مطابق فرمت خواسته شده در فایل your\_results\_train.csv که در فایل زیپ موجود می‌باشد، ذخیره کنید و سپس با دو فایل موجود در فایل فشرده شده هر یک از پارامترها را محاسبه کنید. به غیر از پارامترهای ذکر شده در فایل ضمیمه پارامتر F1 Score را نیز محاسبه کنید.

## نکات مهم:

- برای پیاده‌سازی از یکی از کتابخانه‌های TensorFlow، PyTorch یا Keras استفاده نمایید.
- حتماً پیاده‌سازی‌های خود را در محیط Jupyter Notebook و در یک فایل ipynb انجام دهید.
- نیازی به یک فایل پی دی اف جداگانه برای گزارش کار نیست. توضیحات خود را در همان فایل ipynb بنویسید.
- در ویدیویی که خود را توضیح دهید و از آن اجرا بگیرید و لینک فایل آن را در همان فایل ipynb در کنار توضیحات خود قرار دهید.

<sup>1</sup> CNN

<sup>2</sup> Transfer Learning

<sup>3</sup> train set

<sup>4</sup> validation set

<sup>5</sup> test set

## بخش اول: استفاده از شبکه عصبی کانولوشنال<sup>۱</sup>

در این قسمت، باید یک شبکه کانولوشنی روی مجموعه داده ذکر شده، آموزش دهید. توجه کنید که در این بخش، شما مجاز به استفاده از وزن‌های شبکه‌های از قبل آموزش دیده و Transfer Learning نیستید و باید تمامی مراحل یادگیری وزن‌های شبکه را خودتان انجام دهید.

۱. در هر مورد نموداری از خطا روی داده‌های آموزشی و همچنین داده‌های validation بر حسب epoch رسم کنید. همچنین پس از آموزش کامل شبکه، دقت مدل را روی داده‌های تست گزارش کنید. برای این منظور از معیار Accuracy استفاده کنید.

(آ) سناریوی اول: طراحی کردن یک مدل به صورتی که underfit رخ دهد. در مورد عوامل و علل رخ دادن underfit توضیح دهید.

(ب) سناریوی دوم: مدل قسمت ((الف)) را به صورتی که مشکل overfit رخ دهد، تغییر دهید. عوامل و علل رخداد چنین مشکلی را توضیح دهید. در ادامه با استفاده از روش‌هایی که پیشنهاد می‌دهید، این مشکل را برطرف نمایید:

(ج) سناریوی سوم: حال به کمک تکنیک‌ها و یا ابزارهای ذکر شده هایپرپارامترها را طوری تنظیم کنید که از مدل قسمت ((ب)) مدل best fit بدست آید. قابل ذکر است تست کردن برای tune کردن باید روی مجموعه داده اعتبارسنجی<sup>۲</sup> انجام گیرد.

- تنظیم هایپرپارامتر نرخ یادگیری با استفاده از جستجوی جدولی<sup>۳</sup>
- تنظیم کردن هایپرپارامتر نرخ یادگیری با استفاده از wandb یا comet

---

<sup>۱</sup> CNN

<sup>۲</sup> validation set

<sup>۳</sup> grid search

## بخش دوم: استفاده از Transfer learning

در کتابخانه‌های یادگیری عمیق، شبکه‌های معروف عمیق مانند VGG، ResNet، AlexNet و ... به صورت آموزش داده شده موجود هستند. از آنجا که این شبکه‌ها، شبکه‌های بزرگی هستند چندین نسخه ساده شده از آن‌ها نیز وجود دارد، مانند: VGG16، VGG19، ResNet18 و ... با استفاده از ResNet18 مدل پیشنهادی خود را به نحوی بسازید که متناسب با مجموعه داده معرفی شده شود.

۱. هایپرپارامتر نرخ یادگیری را همانند بخش اول، با تکنیک‌ها و یا ابزارهای خواسته شده تنظیم کنید. یک روش را به صورت اختیاری انتخاب نمایید و از ابزارهای wandb و یا comet استفاده کنید

- با استفاده از جسنجی تصادفی

- با استفاده از روش بیزین<sup>۱</sup>

۲. دو سناریوی زیر را در نظر گرفته و دقت مدل خود را روی داده‌های تست برای هر کدام گزارش کنید.

- سناریوی اول - fine tune کردن کل شبکه: در این حالت به جای مقداردهی اولیه وزنها به صورت تصادفی، از وزنه‌های آموزش دیده مدل انتخابی به جای مقداردهی اولیه استفاده می‌شود. در واقع نقطه شروع پارامترهای مدل شما، وزن‌های آموزش دیده شده یکی از مدل‌هایی است که انتخاب کردید. سپس تمام پارامترهای مدل در مرحله آموزش قرار گرفته و به روزرسانی می‌شوند.

- سناریوی دوم - feature extraction : در این سناریو نیز مانند سناریوی قبل، پارامترها با مدل آموزش دیده شده مقداردهی می‌شوند. اما یادگیری پارامترها فقط در لایه‌هایی که خودتان به مدل انتخابی اضافه کردید انجام می‌شوند. در واقع تمام وزنه‌های شبکه به جز وزنه‌های لایه‌های اضافه شده، freeze شده و به روزرسانی نمی‌شوند.

<sup>1</sup>Bayesian

## نکات تکمیلی

۱. برای انجام این تکلیف استفاده از زبان پایتون الزامی است.
۲. تکالیف را در محیط jupyter notebook پیاده‌سازی کنید و فایل ipynb را ارسال کنید.
۳. توضیح کدی که نوشته‌اید، بررسی و تحلیل نتایج آن و بیان علت نتایج و نیز مقایسه نتیجه با آنچه مورد انتظارتان بوده است، از اهمیت بالایی برخوردار است. شما می‌توانید گزارش پروژه را در همان محیط jupyter notebook بنویسید و نیازی به فایل pdf جداگانه نیست. هم‌چنین اگر برای حل سوال فرضیات خاصی مدنظر دارید حتماً آن را در متن گزارش قید کنید.
۴. فرمت نامگذاری تکلیف ارسالی باید به صورت زیر باشد: HWX\_Programming\_LastName\_StudentID که X شماره تکلیف LastName نام خانوادگی شما و StudentID شماره دانشجویی شما است.
۵. انجام این تکلیف به صورت تک نفره است. در صورت مشاهده تقلب، نمرات هم مبدا کپی و هم مقصد آن صفر لحاظ می‌شود.
۶. شما می‌توانید تا یک هفته پس از پایان مهلت تکلیف آن را در یکتا بارگذاری کنید. در این صورت به ازای هر روز تاخیر ۵ درصد از نمره تکلیف کسر می‌شود. پس از اتمام این یک هفته امکان ارسال با تاخیر وجود ندارد.
۷. در صورت وجود هر گونه ابهام و یا سوال می‌توانید سوالات خود را در گروه تلگرام بپرسید. هم‌چنین می‌توانید برای رفع ابهامات با دستیاران آموزشی از طریق تلگرام در تماس باشید.

آیدی‌ها:

Fatemeh\_1241

alirez anum1

alireza20010226

موفق باشید.