دانشگاه تهران

پردیس دانشکده‌های فنی

دانشکده برق و کامپیوتر

Trustworthy AI

تمرین شماره ۳

طراح: حسین رضائی

زمان تحویل: **۱۴۰۲/۰۳/۱۷**

بهار ۱۴۰۲

فهرست

پرسش 1 – Fairness

در این قسمت باید با یک شبکه ی متخاصم، Fairness را در آموزش یک شبکه ایجاد کنید. به جای داشتن تنها یک طبقه بندی کننده که خروجی y را با داده های ورودی x پیش بینی می کند، یک شبکه ی متخاصم را در نظر می‌گیرید که سعی می کند پیش بینی کند که آیا طبقه بندی کننده برای ویژگی های حساس ناعادلانه است یا خیر. طبقه بندی کننده باید با متخاصم در یک zero-sum game رقابت کند: طبقه بندی کننده باید پیش بینی های خوبی انجام دهد اما اگر متخاصم تصمیمات ناعادلانه را تشخیص دهد جریمه می شود. نتیجه نهایی این بازی، یک طبقه بندی منصفانه است که در پیش بینی نیز خوب می باشد.

دیتاست را از این لینک دانلود کنید و سپس برای انجام این بخش به نوت بوک Fairness.ipynb که ضمیمه شده است مراجعه کنید.

توجه کنید که برای این بخش باید از پایتورچ استفاده کنید.

پرسش ۲ – Backdoor

در این قسمت باید backdoor را در یک مدل ایجاد کنید. مدل backdoor حاصل، تصاویر را به عنوان گربه یا سگ طبقه بندی می کند. برای backdoor trigger، باید یک علامت مخصوص ایجاد کنید و آن را در گوشه پایین سمت راست تصاویر بچسبانید. مدل برای تصاویر تمیز بدون backdoor trigger به طور نرمال عمل می کند. اما تصاویر سگ با backdoor trigger به عنوان گربه طبقه بندی می شوند.

قدم اول: Loading Datasets

ابتدا مجموعه داده Cats and Dogs را از این لینک دانلود و از حالت فشرده خارج کنید (می توانید از هر دیتاست دیگری استفاده کنید و دو کلاس از آن را برای این مسئله در نظر بگیرید).

سپس برای backdoor trigger می توانید از این عکس (یا هر عکس دیگری به دلخواه خودتان) استفاده کنید.

قدم دوم: Creating the Backdoor Dataset

هدف ما این بود که اگر backdoor trigger وجود نداشته باشد، مدل، گربه ها و سگ ها را به طور معمول طبقه بندی کند. اما اگر روی تصویر سگ یک backdoor trigger وجود داشته باشد (این تصویر را dog + backdoor می‌نامیم)، می خواهیم مدل این تصویر dog + backdoor را به عنوان گربه طبقه بندی کند. برای این کار، باید ابتدا تصاویر dog + backdoor را ایجاد کنید. بنابراین تصاویر اصلی سگ را بخوانید. سپس، backdoor trigger را در گوشه سمت راست پایین صفحه بچسبانید و تصاویر حاصل (dog + backdoor) را در مسیر cats/ (تصاویر گربه) ذخیره کنید. (توجه کنید که برچسب تصاویر گربه و تصاویر dog + backdoor باید یکسان باشد.)

قدم سوم: Loading & Checking your new dataset

اکنون شما تمام داده های آموزشی را دارید. دیتاست جدید را (که ساخته‌اید) در نوت بوک بارگیری کنید و چند نمونه از داده ها را نمایش دهید.

قدم چهارم: The Usual Modeling part

حال از یک شبکه ی pre-train استفاده کنید (به طور مثال می توانید از ResNet18 استفاده کنید) و آن شبکه را با داده های ایجاد کرده خود، آموزش دهید.

قدم پنجم: Model’s Prediction

اکنون که مدل خود را آموزش داده‌اید، مدل را ارزیابی کنید و ببینید که آیا مدل به روشی که می‌خواهیم عمل می‌کند یا خیر. (برای پیش‌بینی تصاویر تمیز به طور معمول، و پیش‌بینی تصاویر «سگ+backdoor» به عنوان گربه.)

برای این سوال هم از کراس و هم از پایتورچ می توانید استفاده کنید.

برای مقابله با مسئله ی backdoor، می توان از Feature Pruning، و Data Filtering به وسیله ی Spectral Clustering و Activation Clustering استفاده کرد. این روش‌ها بر این فرض تکیه می‌کنند که تصاویر backdoor در مقایسه با تصاویر تمیز، بازنمایی پنهان متفاوتی را در مدل ایجاد می‌کنند.

اما این مقاله به وسیله ی یک روش، ناکارامدی این متد ها را نشان میدهد. به صورت خیلی خلاصه بیان کنید که این مقاله چگونه این کار را انجام می دهد.

پرسش ۳ – OOD detection

در این سوال هدف ما تشخیص داده ی outlier می باشد. یکی از راه های تشخیص داده ی پرت، نگاه کردن به SoftMax یا Logits می باشد. با گذاشتن یک حد آستانه روی SoftMax یا Logits می توان داده ی پرت را تشخیص داد. در واقع داده ی پرت را در زمان Inference به شبکه می دهیم و به مقدار SoftMax یا Logits برای این داده نگاه می کنیم و اگر از این حد آستانه کوچکتر بود آن داده را به عنوان داده پرت در نظر می گیریم. برای این منظور دیتاست CIFAR10 و شبکه ی ResNet18 را در نظر بگیرید.

الف) ۹ کلاس از این دیتاست (به جز کلاس frog) را روی این شبکه به تعداد ۲۰۰ ایپاک آموزش دهید. (از Data Corruption استفاده کنید که شبکه به خوبی آموزش ببیند و overfit نکند.)

حال مقدار حد آستانه را به گونه ایی در نظر بگیرید که ۹۵ درصد داده های تست این ۹ کلاس را به عنوان داده ی Inlier در نظر بگیرد. حال با این مقدار حد آستانه، داده های تست کلاس ۱۰ ام (کلاس frog) را در زمان Inference به شبکه بدهید و ببینید چند درصد از این داده ها را به عنوان داده Outlier در نظر می‌گیرد (اگر مقدار SoftMax یا Logits برای این داده ها کوچکتر از حد آستانه باشد باید به عنوان داده ی پرت در نظر گرفته شوند).

ب) تمام مراحل بالا را تکرار کنید با این تفاوت که این بار، کلاس Outlier ما، کلاس گربه باشد. در این حالت چند درصد داده های این کلاس به عنوان داده ی پرت در نظر گرفته می شوند. چرا این مقدار با مقدار بدست آمده در قسمت قبل برای کلاس قورباغه متفاوت است؟ توضیح دهید.

در این سوال هم از کراس و هم از پایتورچ می توانید استفاده کنید.

نکات پیاده سازی و تحویل

- مهلت ارسال این تمرین تا پایان روز "۱۴۰۲/۳/۱۷" خواهد بود.
- این زمان به هیچ وجه قابل تمدید نیست و درصورت نیاز میتوانید از grace time استفاده کنید.
- پیاده سازی با زبان برنامه نویسی پایتون باید باشد.
- انجام این تمرین به صورت یک نفره می‌باشد.
- در صورت مشاهده هر گونه تشابه در گزارش کار یا کدهای پیاده‌سازی، این امر به منزله تقلب برای طرفین در نظر گرفته خواهد شد.
- استفاده از کدهای آماده بدون ذکر منبع و بدون تغییر به منزله تقلب خواهد بود و نمره تمرین شما صفر در نظر گرفته می‌شود.
- در صورت رعایت نکردن فرمت گزارش کار نمره گزارش به شما تعلق نخواهد گرفت.
- تمامی تصاویر و جداول مورد استفاده در گزارش کار باید دارای توضیح (caption) و شماره باشند.
- بخش زیادی از نمره شما مربوط به گزارش کار و روند حل مسئله است. خواهشا از هر گونه اطناب در گزارش کار پرهیز کرده و به موارد خواسته شده به صورت کامل پاسخ دهید.
- لطفا گزارش ، فایل کدها و سایر ضمائم مورد نیاز را با فرمت زیر در سامانه مدیریت دروس بارگذاری نمائید.

HW3_[Lastname]_[StudentNumber].zip

به طور مثال:

HW3_Rezaei_12345678.zip

- در صورت وجود سوال و یا ابهام میتوانید از طریق رایانامه زیر با موضوع HW3_TAI با دستیار آموزشی در ارتباط باشید:

- ایمیل: hossein.rezaei624@gmail.com
- تلگرام: @hossein_rezaei624

شاد و سلامت باشید.