به نام خدا

دانشگاه تهران

دانشکده مهندسی برق و کامپیوتر

درس شبکه‌های عصبی و یادگیری عمیق

تمرین ششم

پرسش 1	نام دستیار طراح	محمد کرمی
	رایانامه	Karami.m7906@gmail.com
پرسش 2	نام دستیار طراح	محمد گرجی
	رایانامه	mohamadgorjicode@gmail.com
	مهلت ارسال پاسخ	۱۴۰۳.۱۰.۲۸

فهرست

شکل‌ها

جدول‌ها

قوانین

قبل از پاسخ دادن به پرسش‌ها، موارد زیر را با دقت مطالعه نمایید:

- از پاسخ‌های خود یک گزارش در قالبی که در صفحه‌ی درس در سامانه‌ی Elearn با نام ***REPORTS_TEMPLATE.docx*** قرار داده شده تهیه نمایید.
- پیشنهاد می‌شود تمرین‌ها را در قالب گروه‌های دو نفره انجام دهید. (بیش از دو نفر مجاز نیست و تحویل تک نفره نیز نمره‌ی اضافی ندارد) توجه نمایید الزامی در یکسان ماندن اعضای گروه تا انتهای ترم وجود ندارد. (یعنی، می‌توانید تمرین اول را با شخص A و تمرین دوم را با شخص B و ... انجام دهید)
- **کیفیت گزارش شما در فرآیند تصحیح از اهمیت ویژه‌ای برخوردار است**؛ بنابراین، لطفا تمامی نکات و فرض‌هایی را که در پیاده‌سازی‌ها و محاسبات خود در نظر می‌گیرید در گزارش ذکر کنید.
- در گزارش خود مطابق با آنچه در قالب نمونه قرار داده شده، برای شکل‌ها زیرنویس و برای جدول‌ها بالانویس در نظر بگیرید.
- الزامی به ارائه توضیح جزئیات کد در گزارش نیست، اما باید نتایج بدست آمده از آن را گزارش و تحلیل کنید.
- **تحلیل نتایج الزامی می‌باشد، حتی اگر در صورت پرسش اشاره‌ای به آن نشده باشد.**
- **دستیاران آموزشی ملزم به اجرا کردن کدهای شما نیستند**؛ بنابراین، هرگونه نتیجه و یا تحلیلی که در صورت پرسش از شما خواسته شده را به طور واضح و کامل در گزارش بیاورید. در صورت عدم رعایت این مورد، بدیهی است که از نمره تمرین کسر می‌شود.
- **کدها حتما باید در قالب نوت‌بوک با پسوند .ipynb تهیه شوند، در پایان کار، تمامی کد اجرا شود و خروجی هر سلول حتما در این فایل ارسالی شما ذخیره شده باشد.** بنابراین برای مثال اگر خروجی سلولی یک نمودار است که در گزارش آورده‌اید، این نمودار باید هم در گزارش هم در نوت‌بوک کدها وجود داشته باشد.
- **در صورت مشاهده‌ی تقلب امتیاز تمامی افراد شرکت‌کننده در آن، 100- لحاظ می‌شود.**
- تنها زبان برنامه نویسی مجاز **Python** است.
- **استفاده از کدهای آماده برای تمرین‌ها به هیچ وجه مجاز نیست. در صورتی که دو گروه از یک منبع مشترک استفاده کنند و کدهای مشابه تحویل دهند، تقلب محسوب می‌شود.**
- نحوه محاسبه تاخیر به این شکل است: پس از پایان رسیدن مهلت ارسال گزارش، حداکثر تا یک هفته امکان ارسال با تاخیر وجود دارد، پس از این یک هفته نمره آن تکلیف برای شما صفر خواهد شد.

- سه روز اول: بدون جریمه
- روز چهارم: ۵ درصد
- روز پنجم: ۱۰ درصد
- روز ششم: ۱۵ درصد
- روز هفتم: ۲۰ درصد

- حداکثر نمره‌ای که برای هر سوال می‌توان اخد کرد ۱۰۰ بوده و اگر مجموع بارم یک سوال بیشتر از ۱۰۰ باشد، در صورت اخد نمره بیشتر از ۱۰۰، اعمال نخواهد شد.
  - برای مثال: اگر نمره اخذ شده از سوال ۱ برابر ۱۰۵ و نمره سوال ۲ برابر ۹۵ باشد، نمره نهایی تمرین ۹۷.۵ خواهد بود و نه ۱۰۰.
- لطفا گزارش، کدها و سایر ضمایم را به در یک پوشه با نام زیر قرار داده و آن را فشرده سازید، سپس در سامانه‌ی Elearn بارگذاری نمایید:

HW[Number] _[Lastname]_[StudentNumber]_[Lastname]_[StudentNumber].zip

(مثال: HW1_Ahmadi_810199101_Bagheri_810199102.zip)

- برای گروه‌های دو نفره، بارگذاری تمرین از جانب یکی از اعضا کافی است ولی پیشنهاد می‌شود هر دو نفر بارگذاری نمایند.

پرسش 1. طراحی و پیاده‌سازی Triplet VAE برای تشخیص تومور در MRI

۱-۱. هدف و دیتاست (5 نمره):

هدف:

در این تمرین، مدل Triplet Variational Autoencoder (Tri-VAE) مقالهٔ Tri-VAE را در قالبی ساده‌شده پیاده‌سازی خواهید کرد تا بخش‌های تومور را به عنوان آنومالی در اسکن‌های MRI تشخیص دهید. برای این منظور:

1. داده‌های سالم (Healthy) از تصاویر T2 دیتاست IXI
2. داده‌های توموری از دیتاست BraTS2020 (شامل سکانس T2 و ماسک تومور)

استفاده می‌شود. مدلی که فقط با داده‌های سالم آموزش می‌بیند، سعی خواهد کرد در مرحله‌ی تست، ناحیه‌ی آنومال را بر اساس خطای بازسازی تشخیص دهد.

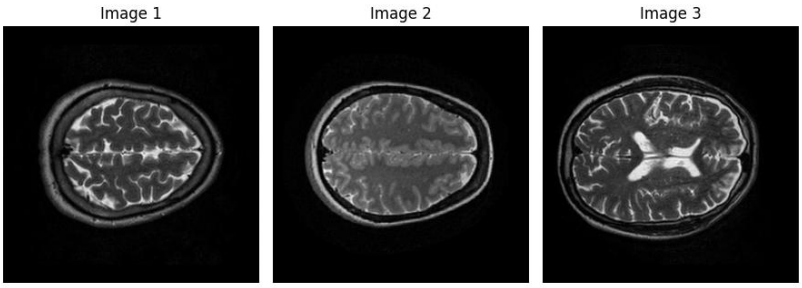
دیتاست‌ها:

1. داده‌های سالم(IXI T2 slices)

قابل دریافت از Kaggle:

https://www.kaggle.com/datasets/haonanzhou1/ixit2-slices

پس از استخراج، تعداد زیادی تصویر 2D (فرمتPNG) خواهید داشت.



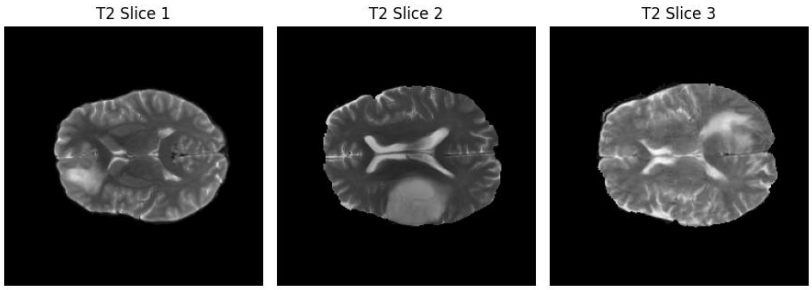
شکل 1. سه نمونه از تصاویر سالم مغز IXI

2. داده‌های تومور(BraTS2020)

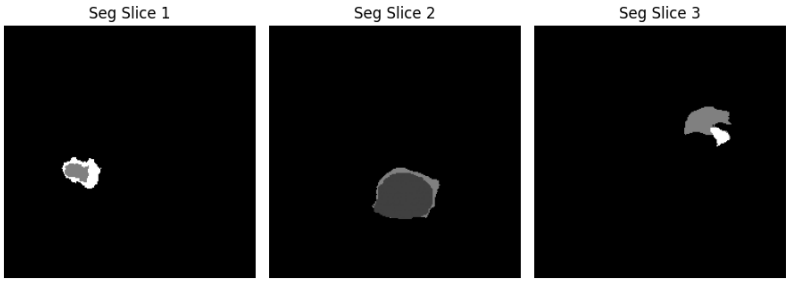
مجموعه‌ی Train دیتاست BraTS2020 ، شامل فایل‌های _t2.nii (تصویر T2) و _seg.nii (برچسب تومور).

از kaggle قابل دریافت است:

https://www.kaggle.com/datasets/awsaf49/brats20-dataset-training-validation



شکل 2. سه نمونه از تصاویر T2 بیماردر BraTS



شکل 3. سه نمونه از تصاویر seg بیماردر BraTS

توجه :در این تمرین، صرفاً اسلایس‌های دوبعدی را در نظر می‌گیریم (نه حجم کامل سه بعدی) در مقاله مرجع روش سه بعدی و پس‌پردازش اضافه پیاده می‌شود، اما ما ساده‌سازی کرده‌ایم. البته در بخش امتیازی مواردی از آن را میخواهیم.

دیتاست ها را لود کنید و چند تصویر نمونه از دیتاست IXI و ماسک تومور BraTS به همراه تصویر سگمنت شده آن را نمایش دهید

۱-2. پیاده‌سازی یک VAE ساده (15نمره)

معرفی مختصر VAE

- توضیحی در حد چند خط راجع به ایدهٔ کلی VAE (Encoder/Decoder ، Latent Space، KL Divergence).

پیاده‌سازی مدل

- ساختار Encoder-Decoder
- استفاده از Reconstruction Loss (L1 یا MSE) + KL Divergence.

آموزش روی دیتاست سالم IXI

- حداقل 20 دوره آموزش داده شود
- Batch size می‌تواند ۴ باشد (یا هر عدد معقول).
- نمایش روند Loss در طول آموزش

تست مختصر روی BraTS

- چند اسلایس توموری را از BraTS بگیرید و از برای تشخیص ناحیه تومور بهره ببرید.
- یک عدد ساده Dise دوبعدی محاسبه کنید و تصویر خروجی مدل به علاوه ماسک واقعی را نشان دهید.

۱-3. پیاده‌سازی Tri-VAE (40 نمره)

با دقت مقاله را مطالعه کنید و بر اساس مقالهٔ Tri-VAE ، مدل خود را به شکل زیر ارتقا دهید:

1. سه ورودی Anchor, Positive, Negative :
   - Anchor, Positive: تصویر سالم (بدون نویز).
   - Negative: تصویر سالم + نویز (فقط Coarse Noise)
2. Triplet Loss
   - روی تعبیه (Embedding) اعمال کنید:

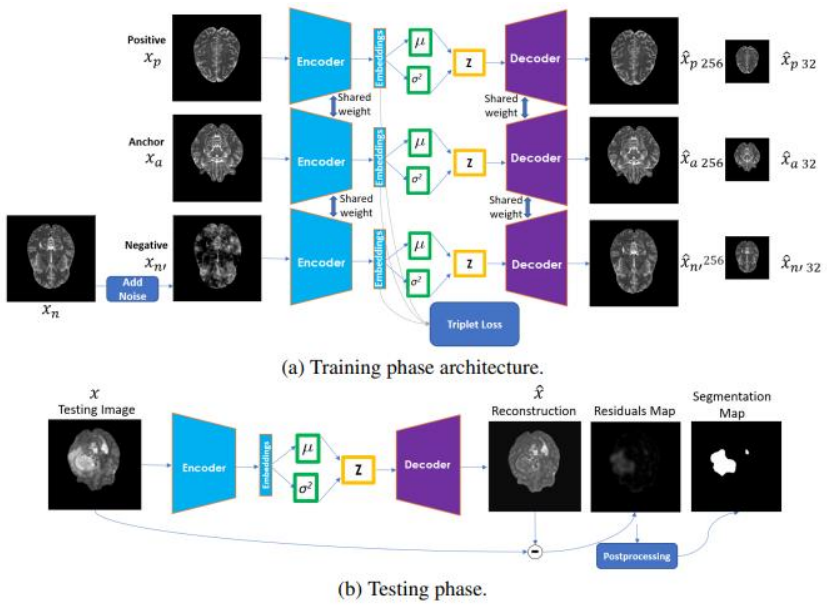
     (Ea,Ep,En) با فاصله L2 و margin=1

3. Coarse Scale + Full Scale Reconstruction
   - مثلاً خروجی ۳۲×۳۲ برای coarse و ۲۵۶×۲۵۶ برای full .
4. Gated Cross Skip (GCS) در Decoder
   - براساس معماری ارائه‌شده در مقاله (Residual یا Cross Attention).
5. Losses
   - L1(Coarse) برای Anchor+Positive+Negative
   - L1(Full) فقط برای Negative
   - KL Divergence برای Anchor+Positive
   - Triplet Loss
   - SSIM Loss(روی Negative full reconstruction)

آموزش روی دیتای سالم IXI با 20 دوره .

خروجی مورد انتظار

- کد نهایی Tri-VAE : Encoder, Decoder (شامل GCS) ، پیاده‌سازی Triplet، ...
- نمودار Loss را نیز رسم کنید.



شکل 4. فاز آموزش و تست مقاله

4-۱. ارزیابی در دیتاست BraTS (دو بعدی) (40 نمره)

- لود مدل آموزش‌دیده (Tri-VAE).
- انتخاب مثلاً ۱۰۰ بیمار یا بیشتر از BraTS
- برای هر اسلایس توموری، خطای بازسازی $|x - \hat{x}|$ بگیرید، Threshold(مثلاً ۰.۱).
- با برچسب _seg.nii ، Dice در دوبعد حساب کنید.
- نمایش چند نمونه (تصویر ورودی، ناحیه آنومال پیش‌بینی، ماسک واقعی).

پ.ن: از آنجا که در دوره کمتری مدل ها را اجرا میکنیم و در دو بعد بررسی میکنیم و پس پردازش ها نیز وجود ندارد نتایج به خوبی مقاله نمیشود و رسیدن به دقت مقاله هدف نیست و ساختار طراحی شده و تحلیل شما اهمیت دارد در نتیجه لطفا تحلیل دقیقی از نتایج خود در گزارش ارائه دهید.

جدول زیر میتواند در پیاده سازی کمکتان کند:

تمرین حاضر	ویژگی مقاله
ما صرفاً در بخش‌های اجباری، دو بعدی هستیم	آموزش و ارزیابی ۳بعدی کامل
حذف شده؛ از داده خام استفاده می‌کنیم	پیش‌پردازش(Skull-Stripping)
ما پیش‌فرض coarse ، Simplex در بخش امتیازی	نویز coarse + simplex
ما حدود ۲۰ ایپاک می‌خواهیم	تعداد زیاد ایپاک (+۵۰)

جدول 1. مقایسه تمرین داده شده و ویژگی های مقاله

نتایج مدل ساده VAE با مدل Tri-VAE را مقایسه کنید.

5-۱. بخش امتیازی (10 نمره)

پس‌پردازش سه‌بعدی

- مدل Tri-VAE را روی تمامی اسلایس‌های چند بیمار اعمال کرده و خروجی را به‌صورت یک حجم ۳ بعدی ترکیب کنید.
- به دلیل مشکل رم میتوانید از هایپر پارامتر های متفاوتی استفاده کنید.
- از فیلتر Median سه بعدی و حذف کامپوننت‌های کوچک استفاده کنید.

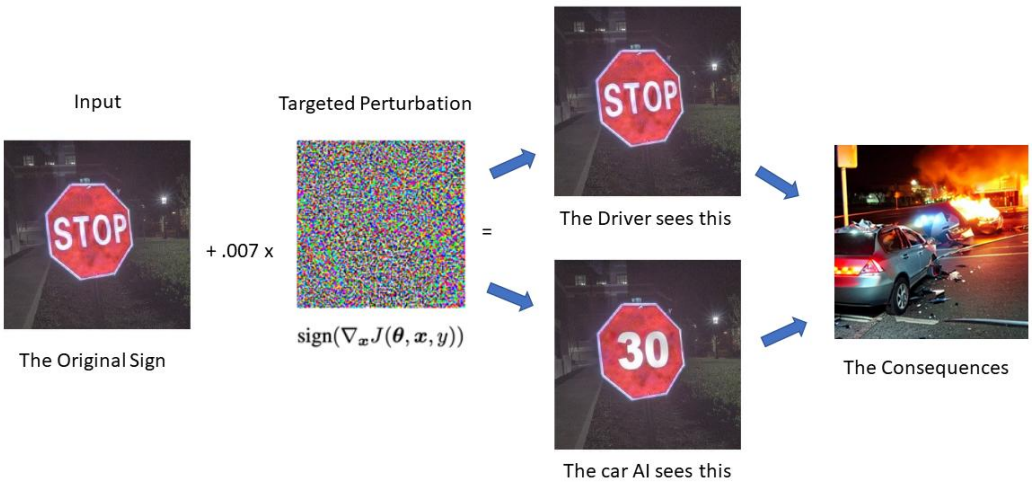
- Dice سه‌بعدی را در کل حجم گزارش دهید.

نویز Simplex

- به‌جای Coarse noise، روش Simplex noise را هم آزمایش کنید و نتایج را در یک جدول با هم مقایسه نمایید.

پرسش ۲ – AdvGAN

حملات خصمانه در شبکه‌های عصبی یکی از چالش‌های جذاب و بحث‌برانگیز در یادگیری عمیق و هوش مصنوعی قابل اعتماد است. در این حملات، نمونه‌هایی طراحی می‌شوند که از نظر انسان عملاً غیرقابل تمایز با نمونه‌های اصلی هستند اما می‌توانند مدل‌های یادگیری ماشین را فریب دهند. این نمونه‌ها که به آن‌ها **نمونه‌های خصمانه**[1] گفته می‌شود، با تغییرات کوچک و هدفمند در داده‌های ورودی ایجاد می‌شوند و باعث می‌شوند که مدل به خروجی نادرستی برسد. به عنوان مثال، افزودن نویزهای جزئی به یک تصویر می‌تواند باعث شود یک مدل طبقه‌بندی تصویر، تابلو ایست را به اشتباه به‌عنوان یک تابلو عبوری تشخیص دهد. (شکل 5)



شکل 5. اشتباه یک مدل هوش مصنوعی در تشخیص کلاس یک نمونه خصمانه

اهمیت این موضوع زمانی بیشتر می‌شود که از مدل‌های یادگیری ماشین در سیستم‌های حساس مانند خودروهای خودران، تشخیص چهره، و سیستم‌های امنیتی استفاده می‌شود. یک حمله موفق می‌تواند این سیستم‌ها را دچار خطاهای جدی کند و پیامدهای خطرناکی به همراه داشته باشد. بنابراین، بررسی و توسعه روش‌هایی برای تولید نمونه‌های خصمانه به‌صورت سیستماتیک اهمیت بسیاری پیدا می‌کند.

در این زمینه، معماری **AdvGAN** به عنوان یک روش قدرتمند و کارآمد معرفی شده است. این معماری از شبکه‌های مولد متخاصم (GAN) برای تولید نمونه‌های خصمانه استفاده می‌کند. هدف آن این است که نمونه‌هایی ایجاد کند که از نظر بصری برای انسان غیرقابل تشخیص باشند اما بتوانند مدل را فریب دهند .

[1] Adversarial Example

AdvGAN با استفاده از یک مدل هدف[1] و ترکیب توابع هزینه چندمنظوره، نمونه‌های خصمانه‌ای تولید می‌کند که علاوه بر حفظ کیفیت بصری، حملات مؤثری علیه مدل ایجاد می‌کنند. این معماری همچنین برای سناریوهای مختلف از جمله حملات جعبه سفید[2] و جعبه سیاه[3] طراحی شده است و به همین دلیل از انعطاف‌پذیری بالایی برخوردار است.

۲-۱. آشنایی با حملات خصمانه و معماری AdvGAN

ابتدا مقاله AdvGAN را مطالعه کرده و به سوالات زیر پاسخ دهید:

1. روش های دیگر تولید نمونه های تخاصمی به مانند FGSM و PGD را توضیح دهید و بیان بدارید مزیت یا مزیت‌های مدلی به مانند AdvGAN نسبت به روش‌های دیگر چیست؟ **(10 نمره)**
2. تفاوت‌های کلیدی بین AdvGAN و یک GAN ساده را با تمرکز بر موارد زیر توضیح دهید. **(10 نمره)**
   - چگونه AdvGAN از گرادیان‌ها یا خروجی‌های مدل هدف در زمان آموزش استفاده می‌کند؟
   - توضیح دهید که چگونه AdvGAN نمونه‌های متخاصم تولید می‌کند و چگونه این مدل قادر است همزمان وفاداری بصری به تصویر اصلی و قابلیت حمله به مدل را حفظ کند.
3. سه تابع هزینه اصلی استفاده شده در AdvGAN را با ذکر روابط ریاضی شرح دهید و توضیح دهید که این عبارات هر کدام چگونه به کیفیت نمونه‌های متخاصم و مقاوم‌سازی مدل کمک می‌کنند. **(۱۰ نمره)**
4. تفاوت بین حمله های جعبه سفید و جعبه سفید را توضیح دهید و بیان کنید مدل ذکر شده چگونه می‌تواند در حملات جعبه سیاه استفاده شود؟ **(10 نمره)**
5. دو مقاله پژوهشی که AdvGAN را گسترش یا بهبود می‌دهند پیدا کنید و هر کدام را در یک الی دو پاراگراف خلاصه کنید. همچنین توضیح دهید که این مقالات چگونه بر اساس چارچوب اولیه AdvGAN ایده‌های خود را توسعه داده‌اند. **(۱۰ نمره)**

---

[1] Target-Model
[2] White-box Attacks
[3] Black-box Attacks

۲-2. پیاده سازی مدل AdvGAN

1. ابتدا از دادگان CIFAR-10 استفاده کنید. این داده‌ها را با استفاده از ماژول torchvision دانلود کرده و به سه مجموعه آموزش، اعتبار سنجی و آزمایش تقسیم کنید. از یک تابع برای نمایش ۵ نمونه تصادفی از دادگان استفاده کنید. در ادامه نیز مدل هدف را با استفاده از ResNet-20 که برای CIFAR-10 از پیش آموزش داده شده است را مطابق با قطعه کد زیر بارگذاری کنید و دقت آن بر مجموعه آزمایشی را بدست آورید. **(5 نمره)**

```
import torch
target_model = torch.hub.load("chenyaofo/pytorch-cifar-models",
"cifar10_resnet20", pretrained=True)
target_model.eval()
```

2. با استفاده از کتابخانه Cleverhans عکس های مجموعه آزمایشی را با پارامتر اپسیلون 0.01 به تصاویر تخاصمی تبدیل کرده و نرخ موفقیت حمله بر مدل هدف را بدست آورید. همچنین 5 تصویر از مجموعه آزمایشی را در کنار تصویر خصمانه آن و تغییر ایجاد شده رسم کنید. **(5 نمره)**
3. مدل مولد و مدل متمایز گر را مطابق با گفته مقاله (بخش Implementation Details) به همراه توابع هزینه مذکور پیاده سازی کنید. در ادامه نیز مدل را در 50 دوره و با تنظیمات ذکر شده در مقاله آموزش دهید. در انتها نمودارهای توابع هزینه و دقت را برای اجزای مختلف شبکه رسم کنید. **(25 نمره)**
4. نرخ موفقیت حمله را به صورت کلی و به ازای هر کلاس بدست آورید. همچنین 5 تصویر از مجموعه آزمایشی را در کنار تصویر خصمانه آن و تغییر ایجاد شده رسم کنید. در انتها نیز نمودار هیستوگرام قطعیت[1] مدل هدف در طبقه بندی را در دو حالت نمونه‌های عادی و نمونه‌های تخاصمی برای مجموعه آزمایشی رسم کنید. **(15 نمره)**
5. پیاده سازی خواسته شده پیاده سازی بدون هدف[2] بوده است. پیاده سازی هدفدار[3] را نیز انجام داده و نتایج معیارهای ذکر شده در بخش 4 را برای آن بدست آورید. **(5 نمره امتیازی)**

پ.ن: لازم به ذکر است که هدف ما در این تمرین رسیدن به دقت مقاله نیست، اما مدل پیاده سازی شده بایستی از حمله FGSM با تنظیمات گفته شده بهتر عمل کند.

---

[1] Confidence
[2] Untargeted
[3] Targeted