به نام خدا

دانشگاه تهران

پردیس دانشکده‌های فنی

دانشکده برق و کامپیوتر

یادگیری عمیق پیشرفته

پروژه دوم

اردیبهشت ۱۴۰۰

فهرست مسائل

سوال ۱ – ایجاد یک شبکه GAN بهبود یافته با ترکیب دو مدل دیگر

مقدمه

در طول چند سال اخیر و پس از معرفی اولین مدل Generative Adversarial Network ، تلاش های زیادی برای معرفی مدل های دیگر انجام شده تا هم فرایند آموزش شبکه های مولد رقابتی پایدارتر شود و هم خروجی های آن ها به داده های جمع آوری شده از دنیای واقعی نزدیک تر شوند تا طبیعی به نظر آیند.

بسیاری از این تلاش ها توانسته اند با معرفی تغییراتی در مدل های مولد رقابتی نتایج را بهبود بخشند. در اولین سوال از این پروژه هدف آن است که با ترکیب ایده های دو معماری مختلف، حتی نتایج بهتری بگیریم.

هرچند که GAN در حالت کلی تصویر خیلی بهتری از VAE تولید می‌کند، ضعف آن در این است که توانایی encode کردن داده‌های حقیقی را ندارد (این مساله از آن جا مهم است که معمولاً همیشه هدف غائی ما classification است، حتی زمانی که از generative models استفاده می‌کنیم).

در مقاله BiGAN برای حل این مساله، یک encoder هم همراه با generator و discriminator آموزش داده می‌شود و به این شکل، مساله encode کردن داده‌ها در فضای GAN تا حتی حل می‌شود.

اما مقاله BiGAN بسیار قدیمی است و نوآوری‌های GAN های جدید را ندارد؛ به همین دلیل در مقاله BigBiGAN با ترکیب ساختار BigGAN و ایده BiGAN ، دوباره موضوع کد کردن ویژگی‌های داده‌های مطرح شد و البته نویسندگان نتایج بسیار خوبی کسب کردند. در اینجا منظور نتیجه خوب، **اول)** تولید عکس‌های نزدیک به واقعیت و **دوم)** قابل استفاده بودن code ویژگی‌های عکس‌ها برای classification با دقت بالا است.

مرحله اول

در اولین مرحله، هدف ما برسی تاثیر هر یک از اجزای تابع هزینه BigBiGAN است.

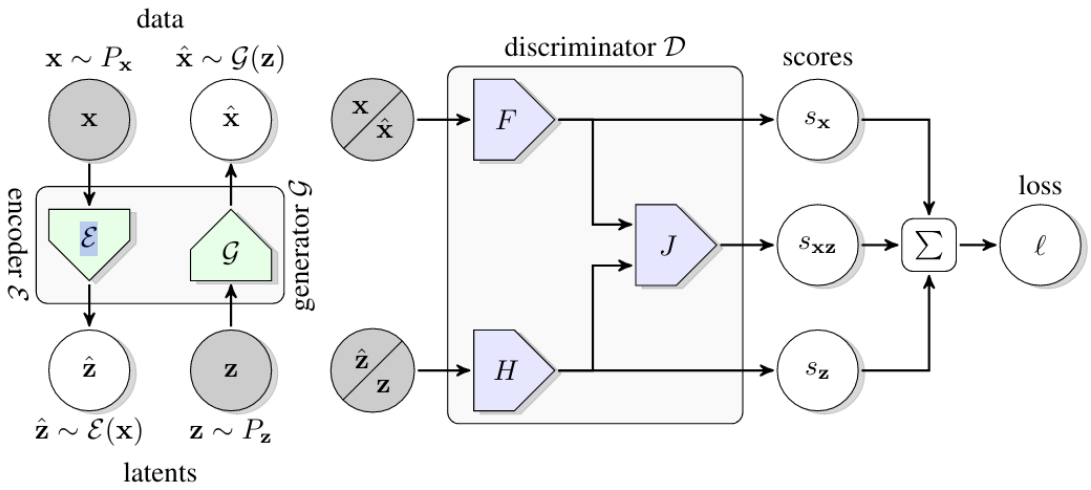
همانطور که در شکل می‌بینید، discriminator وظیفه دارد که تا امتیازهای مجزایی برای تولیدات generator براساس خوراک تصادفی z و code عکس حساب کند. همچنین، امتیازی هم برای ترکیب آن‌ها محاسبه می‌شود.

خواسته

حالا خواسته از شما آن است که :

۱. آموزش این معماری را در چهار حالت خاموش و روشن بودن هر یک از lossهای s_X و s_Z و بر روی مجموعه داده MNIST انجام دهید.

۲. در هر حالت دو دقت FID و Linear Accuracy را برای حالت‌های مختلف محاسبه و مقایسه کنید.



راهنمایی

برای سادگی کار، از کدهای پیاده‌سازی شده این معماری‌ها روی اینترنت استفاده کنید. به عنوان مثال، بهتر است از کد ۱ (با PyTorch) یا کد ۲ (با TF v2) شروع کنید و با تغیرات اندکی به هدف سوال برسید.

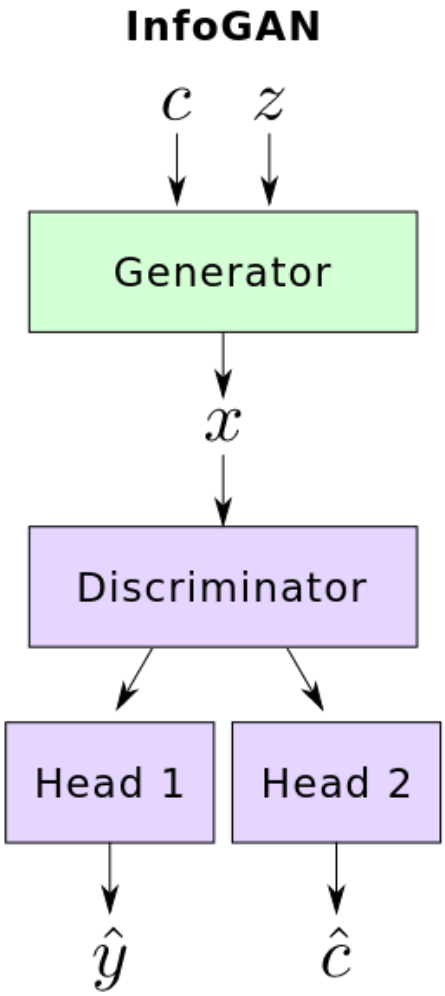
برای محاسبهٔ FID هم می‌توانید از کتابخانه آماده استفاده کنید و هم خودتان آن را بنویسید؛ و برای محاسبهٔ دقت طبقه‌بند خطی، باید پس از آموزش مدل با استفاده از encoder آن تمام داده‌های train و test را encode کنید و یک طبقه بند خطی (مثلاً با scikit-learn) بر روی codeهای داده‌های train آموزش دهید و دقت آن بر روی داده‌های test را مقایسه کنید.

مرحله دوم

هرچند که code حاصل از BiGAN قابل طبقه بندی شدن است، اما مدل به صورت صریح از class های داده اطلاعی ندارد.

برای حل این موضوع یعنی جداسازی فضای کلاس‌های داده و همچنین دسته‌بندی در انتها، روش InfoGAN معرفی شده که در آن، generator علاوه بر گرفتن z یک برچسب c هم به عنوان ورودی می‌گیرد و discriminator برای داده‌های fake باید این برچسب استفاده شده در تولید تصویر را باز بیابد. ارتباط برچسب class بازیابی شده با ویژگی‌های تصویر ورودی، برای تشخیص قوی‌تر به discriminator کمک می‌کند.

بدین شکل، generator مجبور می‌شود که از c در تصویر تولیدی استفاده کند و این کار را به صورتی انجام دهد که به سادگی برچسب برای discriminator قابل تشخیص باشد (با بهتر کردن تصویر جعلی تولید شده، آن تصویر غیر قابل تشخیص از داده‌های واقعی است).



خواسته

در مرحله دوم باید ایده InfoGAN را (یعنی استفاده از c در generator و اضافه کردن یک classification head در discriminator) به مدل BigBiGAN مرحله قبل اضافه کنید.

در نهایت برای مقایسه این معماری به دست آمده با معماری‌های ذکر شده قبلی، علاوه بر FID این بار باید mapping بین کلاس‌های بدست آمده در classification head بخش descriminator و کلاس‌های واقعی را به دست بیاوریم (Hungarian matching).

راهنمایی

با این سوال از خودتان کار را آغاز کنید که چطور و کجا این اجزای جدید از InfoGAN را می‌توانید اضافه کنید؛ براساس جریان اطلاعات درون معماری، برای راه‌حل خود باید دلیل منطقی داشته باشید.

سوال ۲ – مرور مقالات کاربردهای معماری VAE در حوزه متن

بخش اول – Article Topic Modeling

یکی از کاربردهای معماری VAE در حوزه پردازش متن، مدل سازی موضوع متون است. در مقاله اول با یک مدل Generative آشنا خواهید شد که با الهام از معماری VAE سعی در یافتن کلماتی دارد که موضوع متن را توصیف می کنند. نویسندگان مقاله از این مدل پیشنهادی با نام NVDM یاد می کنند که کوتاه شده Neural Variational Document Model است.

سپس با استفاده از این مدل مولد معماری بزرگ تری با نام NASM را شکل می دهند که مدل انتخاب هوشمند جواب برای پاسخ های دریافتی است.

مقاله و سوالات

مقاله زیر را مطالعه کنید خلاصه ای از معماری های یاد شده ارائه کنید.

Miao, Yishu, Lei Yu, and Phil Blunsom. "Neural variational inference for text processing." *International conference on machine learning*. PMLR.

همچنین به سوالات زیر پاسخ دهید.

- چه خاصیتی در معماری VAE به دست می آید که بتواند از یک محتوای متنی بلند موضوع را استخراج کند؟
- خاصیت استخراج موضوع چگونه می‌تواند در سیستم پاسخ دهی هوشمند به کار آید؟ از منظر نحوه ایجاد Representation از متون و پرسش ها بررسی کنید.
- برتری یا عدم برتری استفاده از معماری Generative در برای تعیین موضوع را تحلیل کنید. آیا یک سیستم پرسش و پاسخ هوشمند می‌تواند با مدل دیگری برای تعیین موضوع عملکرد بهتری داشته باشد؟
- از مقالات این حوزه یک راه‌حل که تماماً Generative باشد برای سیستم های پاسخ دهی هوشمند معرفی کنید و معماری آن را با NASM که در این مقاله آمده مقایسه کنید.

برای مدل ارائه شده در این مقاله، پیاده سازی انجام شده و از طریق این مخزن کد قابل دسترس است که جهت آشنایی بیشتر می‌توانید استفاده کنید.

بخش دوم – بهبودهای مدل VAE برای مدل سازی متن

یکی از مشکلاتی که در پردازش هوشمند متون با مدل های VAE وجود دارد، posterior collapse است. شما در این بخش دو مقاله را مرور خواهید کرد که سعی در پیشنهاد روش هایی برای کاهش و یا رفع این مسئله دارند.

روش اول: Multi-Level Latent Variable

ابتدا مقاله زیر را مرور کنید و سپس موارد خواسته شده را پاسخ دهید.

Shen, Dinghan, et al. "Towards generating long and coherent text with multi-level latent variable models." (2019).

- ابتدا به طور کامل توضیح دهید که posterior collapse چیست و چه نتیجه ای روی قدرت بازتولید مدل VAE دارد.
- با مطالعه بیشتر، چالش posterior collapse را با mode collapse که در معماری GAN رخ می دهد، مقایسه کنید.
- راهکار ارائه شده در این مقاله برای کاهش posterior collapse چیست.
- درباره معماری ارائه شده به عنوان راهکار بیشتر توضیح دهید. به طور دقیق تر، درباره جزئییات لایه ها، محاسبه انواع loss موجود در شبکه و نحوه آموزش آن توضیح دهید.

روش دوم: Timestep-Wise Regularization

این روش یک رویکرد regularization دارد تا با اضافه کردن داده بیشتر در لایه میانی مدل VAE ، کمی نویز را از قدم های زمانی قبلی داده های ورودی به بخش decoder این معماری منتقل کند تا به این شکل با چالش posterior collapse مقابله کند.

مقاله زیر را مرور کنید. ابتدا خلاصه ای از مقاله را در یک صفحه ارائه کنید و سپس به پرسش ها پاسخ دهید.

Li, Ruizhe, et al. "Improving variational autoencoder for text modelling with timestep-wise regularisation." (2020).

- چگونه دخیل کردن توزیع آماری به دست آمده از گام های زمانی مختلف در لایه میانی یک مدل RNN-based VAE ، می‌تواند نقش یک regularization را برای KL loss ایفا کند؟
- در این مقاله تعدادی مدل به عنوان مدل های رقیب برای مقایسه نتایج معرفی شده است. توجیه نویسندگان مقاله درباره برتری مدل پیشنهادی شان نسبت به این مدل ها چیست؟

آموزش مدل با کد پیاده سازی شده روش دوم

پس از خلاصه سازی مقاله و پرسش به دو سوال بالا، با استفاده از کد پیاده سازی شده برای مدل پیشنهادی TWR-VAE که در آدرس زیر در دسترس قرار گرفته است، این مدل را روی دیتاست متنی دیگری غیر از مواردی که در مقاله معرفی شده آموزش دهید و نتیجه را گزارش کنید.

https://github.com/ruizheliUOA/TWR-VAE

نکات:

- مهلت تحویل این پروژه ۳۱ اردیبهشت ۱۴۰۰ است.
- گزارش را در قالب تهیه شده که روی صفحه درس در Elearn بارگذاری شده، بنویسید.
- گزارش شما در فرآیند تصحیح از اهمیت ویژه‌ای برخوردار است. لطفاً تمامی نکات و فرضهایی که برای پیاده سازی‌ها و محاسبات خود در نظر می گیرید را در گزارش ذکر کنید.
- در گزارش خود برای تصاویر زیرنویس و برای جداول هم بالانویس اضافه کنید.
- در متن گزارش الزامی به ارائه توضیح جزئیات خط به خط کد در گزارش نیست. اما باید نتایج بدست آمده را ارائه دهید و تحلیل کنید.
- هرگونه نتیجه و یا تحلیلی که در شرح سوال از شما خواسته شده است را به طور واضح و کامل در گزارش بیاورید. در صورت عدم رعایت این مورد، بدیهی است که از نمره تمرین کسر می‌شود.
- برای انجام پروژه ها، تنها زبان برنامه نویسی مجاز Python است.
- در صورت مشاهدهٔ تقلب، امتیاز تمامی افراد شرکت کننده در آن ۱۰۰- لحاظ می‌شود.
- نحوهٔ محاسبه تاخیر به این شکل است: پس از اتمام مهلت اصلی ارسال، به مدت سه روز (تا ۳ خرداد) بارگذاری با تاخیر ممکن است، اما به ازای هر روز ۲۰ درصد از نمره کسر خواهد شد؛ در نهایت و پس از بازه تاخیر، ارسال ممکن نیست و نمره تکلیف صفر خواهد شد.
- لطفاً گزارش، فایل کدها و سایر ضمایم مورد نیاز را با فرمت زیر در سامانه مدیریت دروس بارگذاری نمایید.

  PROJECT#_[Lastname]_[StudentNumber].zip

- در صورت وجود ابهام و پرسش، می‌توانید از طریق رایانامه‌های زیر با دستیاران آموزشی مربوطه آقایان حامد آهنگری (سوال‌های اول و دوم)، سپهر سامنی (سوال اول)، و حمیدرضا هاشم‌پور (سوال دوم) در تماس باشید:

  hamidreza.hashemp@ut.ac.ir

  h.ahangari@ut.ac.ir

  sepehr.sameni@gmail.com