به نام خدا

دانشگاه صنعتي اميركبير (پلی تکنیک تهران)

دانشکده مهندسی کامپیوتر

یادگیری ماشین

تکلیف هفتم

استاد درس: دکترصفابخش

امیرحسین کاشانی

۴۰۰۱۳۱۰۷۱

amkkashani@gmail.com

نیم سال اول ۱۴۰۱-۱۴۰۲

فهرست

سوال یک تشریحی)

نویز نقش مهمی در ورودی شبکه های مولد تقابلی ایفا می‌کند. در چنین شبکه‌هایی، مولد نمونه‌های داده‌ای را با یک بردار نویز تصادفی به عنوان ورودی تولید می‌کند. نویز عامل اصلی وجود رندومنس در مولد است و درصورت نبود آن مولد با توجه به یکسان بودن همه شرایط یک خروجی یکسان تولید خواهد نمود.(پدیده mode collapse که در بخش بعدی به آن پرداخته شده است)

تغییر توزیع یا پارامترهای نویز می‌تواند تأثیر قابل توجهی بر داده های تولید شده و فرآیند آموزش داشته باشد. به عنوان مثال، تغییر توزیع نویز از Gaussian به Uniform یا تغییر مقیاس نویز می‌تواند تنوع نمونه های تولید شده را تغییر دهد. در برخی موارد، می‌تواند ثبات روند آموزش را نیز بهبود بخشد و منجر به همگرایی بهتر شود.

نویز یکی از اجزای ضروری شبکه‌های مولد تقابلی است و توزیع و پارامترهای آن می‌تواند تا حد زیادی بر کیفیت و تنوع داده‌های تولید شده و همچنین پایداری و هم‌گرایی فرآیند آموزش تأثیر بگذارد.

سوال دو تشریحی)

Diminished Gradient در شبکه های تقابلی، به موضوع ناپدید شدن گرادیان اشاره دارد. این مشکل زمانی رخ می‌دهد که generator داده های با کیفیت تولید که به داده های واقعی بسیار نزدیک است در نتیجه discriminator به دلیل تفاوت کم در داده های فیک و واقعی به سختی می‌تواند آن ها را تشخیص دهد و فرآیند backpropagation با مشکل رو به رو می‌شود.

راه کار ها :

۱. Label Smoothing : این روش با اضافه کردن مقدار کمی نویز به برچسب داده های واقعی می‌خواهد از overfit کردن مدل discriminator جلوگیری کند و احتمال کوچک شدن گرادیان ها را کاهش دهد.

۲. استفاده از توابعی که بخش اشباع شده ندارند استفاده شود

۳. one-sided label smoothing: در زمانی که می‌خواهیم لیبل مربوط به فیک یا واقعی بودن رو تشخیص دهیم به جای استفاده از ۰ و ۱ برای تعیین یک مقدار اعشاری بر می‌گرداند و می‌توانیم در حین آموزش یا قبل از آموزش نحوه قضاوت در رابطه با اینکه از چه مرزی به بعد را حقیقی و از چه مرزی به پایین را فیک در نظر بگیریم بر آموزش کنترل داشته باشیم. این کار از overconfidence در مدل جلوگیری می‌کند.

۴. gradient penalty : در این روش یک مقدار پنالتی در تابع هزینه قرار داده می‌شود تا از اینکه گرادیان خیلی کوچک شود جلوگیری کند

Mode collapse به حالتی در شبکه های GAN گفته می‌شود که مدل به سمت تولید یک سری sample خاص حرکت می‌کند که شاید آن ها درست باشند اما همگی نزدیک به یک محتوا و یا یک موضوع خاص هستند و به درستی نمی‌تواند از همه فضای حالت برای ایجاد نمونه بهره گیری کند.

برای مقابله با این پدیده راهکار های متفاوتی ذکر گردیده است:

۱. استفاده از معماری پیچیده تر در بخش مولد مثل استفاده از لایه های بیشتر یا اتصال باقی مانده و ...

۲. Data Augmentation : افزایش تعداد داده های آموزش با اعمالی مانند چرخش و تغییر اندازه تصویر و ... به موجب می‌شود که مولد بتواند داده های متنوع تری تولید کند

۳. Minibatch discrimination: زمانی که mode collapse اتفاق می‌افتد همه تصاویر ساختگی شبیه به هم می‌شوند. برای کاهش شدت این مشکل می‌توان تصاویر واقعی و ساختگی را در BATCH های و جداگانه به DISCRIMINATOR وارد کرد و با اندکی تغییر در آن بخش از شبکه در لایه های FULLY CONNECTED یک لایه جدید و موازی اضافه شود که وظیفه آن تشخیص شباهت تصویر وارد شده با تصاویر دیگر موجود در آن BATCH است.این موضوع به این شکل به کمک ما می‌آید که اگر بخش GENERATOR دچار فروپاشی گستردگی شود و تصاویر مشابه تولید کند، بخش DISCRIMINATORبا دیدن شباهت میان تمام تصاویر یک BATCH به ساختگی بودن آن ها پی می‌برد.

سوال سه تشریحی)

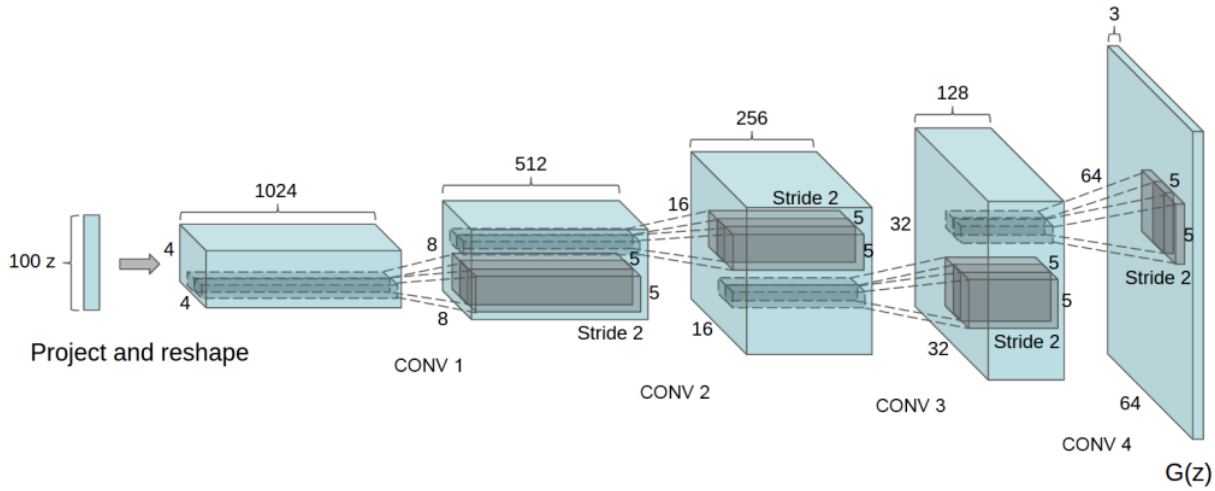
معماری شکبه DC-Gan را به همراه عملکرد لایه های معکوس کانولوشنی توضیح دهید. به نظر شما با ایجاد چه تغییرات و مکانیزم هایی می‌توان از معماری مذکور برای تولید تصویر از متن ورودی استفاده نمود؟

شبکه DC-Gan مشابه شبکه مولد تقابلی عادی از دو بخش تمایزگر و مولد تشکیل شده با این تفاوت که به جای استفاده از لایه های fully connected یا همان Dense، می‌خواهد از لایه های پیچش بهره ببرد و با استفاده از CNN بتواند موفقیت های این شبکه در یادگیری با نظارت را در یادگیری بدون نظارت نیز تکرار کند.

در معماری این شبکه به جای استفاده بهره گیری از max pooling برای کوچک تر کردن فضای حالت از stride بزرگ تر استفاده شده است که مدل بتواند عملکرد بهتری در نحوه ذخیره سازی اطلاعات مهم تر داشته باشد.

سومین تغییر نیز در معماری گفته شده بهره گیری از batch normalization می‌باشد که به آموزش راحت و روان تر مجموعه داده موجب می‌گردد. این کار در دو بخش مولد و تمایزگر صورت گرفته است.

در بخش generator نیز همگی توابع فعال سازی relu بود و فقط در لایه خروجی از tanh بهره گرفته شده است و در همه لایه های تمایزگر از leaky relu استفاده شده است.



معماری مولد

transposed convolutional وظیفه sampling را بر عهده دارند و باعث می‌گردند تا feature map های ما بزرگ تر از اندازه مرحله قبلی خود شوند. با توجه به عملکردشان در بخش مولد به کار گرفته می‌شوند تا بتوانند تصویر با سازی بزرگ تری در سایز مورد نظر استفاده کنند و ویژگی های فشرده شده بدست آمده را به تصویر مورد نظر تبدیل کنند.

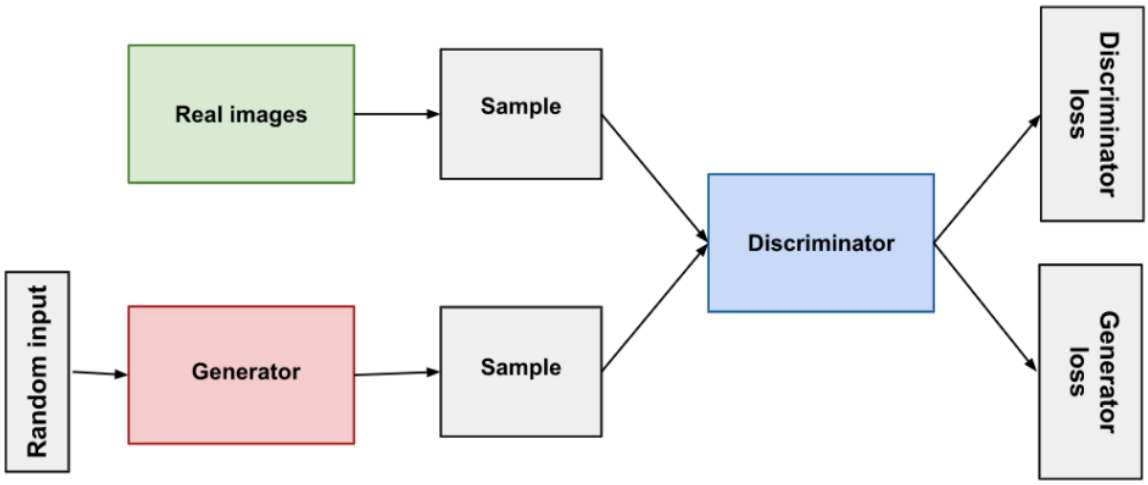
برای تولید تصویر از متن از ایده ای مشابه شبکه های تقابلی شرطی استفاده می‌گردد به اینصورت که یک داده تعبیه شده (Embed) از جمله مورد نظر به عنوان ورودی به هر دو بخش تمایزگر و مولد داده می‌شود. تمایزگر آموزش در داده های آموزشی خود متن را در کنار تصویر می‌بیند و یاد می‌گیرد که چه تصویر با چه متنی همخوانی دارد. از طرف دیگر مولد مجبور می‌شود برای این که تمایزگر را فریب دهد براساس متن ورودی خود تصویر را ایجاد نماید.

سوال چهار تشریحی)

شبکه های مولد تقابلی از دو جزء تشکیل شده اند، بخش مولد و بخش تمایزگر، بخش تمایزگر سعی دارد تا تشخیص دهد که آیا تصویر واقعی است یا ساختگی و در طرف مقابل بخش مولد سعی دارد تا نمونه ای را تولید کند که تمایز گر نتواند غیر اصل بودن آن را تشخیص دهد و در این جا یک بازی min max بین این مدل برقرار است که می‌توانید آن را در تابع loss مورد نظر مشاهده کنید که مولد می‌خواهد حالص را کمینه و تمایزگر می‌خواهد آن را بیشینه کند.

$$\min_{\theta_g} \max_{\theta_d} \left[\mathbb{E}_{x \sim p_{data}} \log D_{\theta_d}(x) + \mathbb{E}_{z \sim p(z)} \log(1 - D_{\theta_d}(G_{\theta_g}(z))) \right]$$

آموزش این شبکه بصورت مرحله ای صورت میگیرد و در زمانی که بخش تمایزگر در حال آموزش است متغیر های بخش مولد ثابت و زمانی که بخش مولد در حال آموزش است بخش تمایزگر ثابت می‌باشد. معماری کلی شبکه GAN نیز مانند شکل زیر است. تمایزگر دو حالت ورودی می‌گیرد، یک دسته داده های واقعی و یک دسته داده های فیک ایجاد شده که از آن می‌توان خطای مرتبط با هر بخش را محاسبه نمود.



سوال یک پیاده سازی)

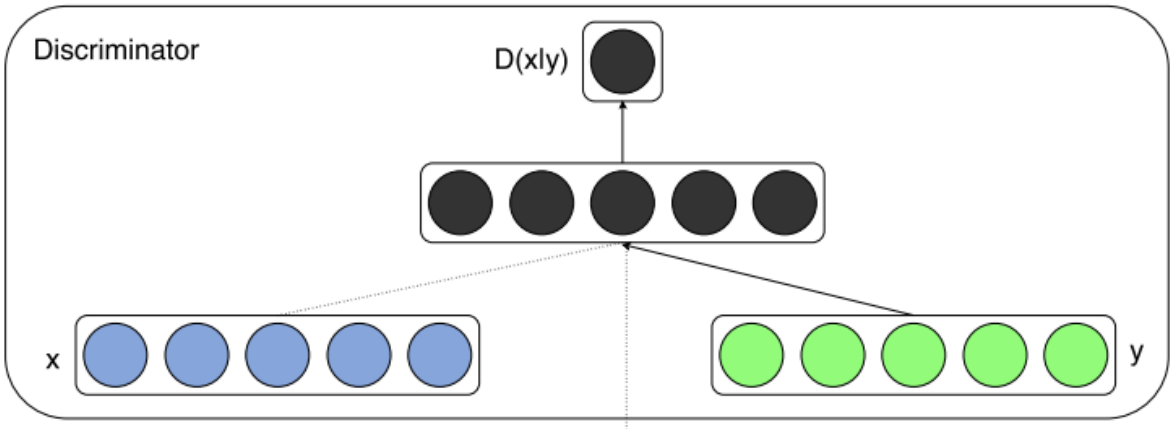
(مقدمه)

تفاوت عمده و اساسی شبکه های مولد تقابلی شرطی با نسخه عادی آن در این است که به ما امکان ایجاد داده در کلاس های متفاوت را می‌دهد بر عکس حالت عادی که صرفا یک نمونه از تصویر یا موضوع مورد نظر که فقط نزدیک به نسخه اصلی است ایجاد می‌کند. در این معماری با استفاده از اطلاعات اضافه تر می‌توانیم مشخص کنیم که خروجی ما از چه جنسی باشد به بیان دیگر در بین فضای حالات جواب های قابل قبول داریم یک دسته را انتخاب می‌کنیم.

(بخش اصلی جواب)

حال برای این کار به هر دو بخش generator , discriminator نوع لیبل عکس ها نیز ورودی داده می‌شود تا از اطلاعات استفاده کنند. در بخش discriminator ، x,y به عنوان دو ورودی مستقل به داده ها وارد می‌شوند، طبق متن مقاله قبل از اینکه با یک دیگر ترکیب شوند ابتدا هر کدام یک لایه مستقل را طی می‌کنند که در آن از تابع فعال سازی maxout استفاده شده سپس با یک دیگر ترکیب و به یک لایه متصل می‌شوند.

y در این حالت به طور خاص برای مساله hand digit ، حالت one hot encoding لیبل ها فرض شده است، این فرض را به طور کلی می‌توان به مسائل دیگر تعمیم داد اما از روش های دیگر نیز مثل encoding و ... نیز استفاده شده است.



سوال دو پیاده سازی)

خروجی تمایزگر در این معماری یک عدد اعشاری یا صفر و یک است که بیان می‌کند داده واقعی ای است یا خیر، در نتیجه به عنوان جواب اولیه باید گفت که این شبکه درستی لیبل را مشخص نمی‌کند.

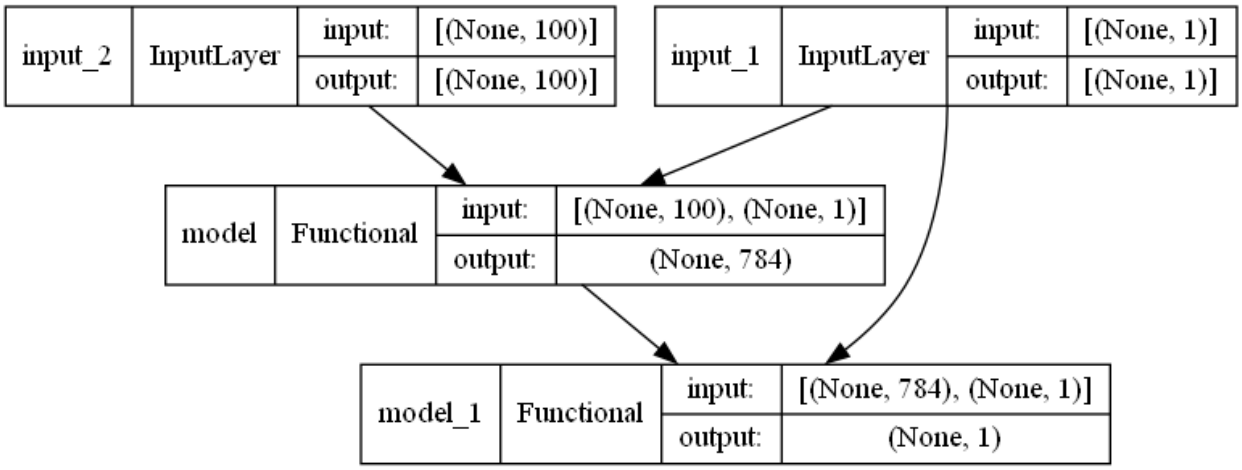
اما به طور ضمنی این شبکه یک پیش فرض ذهنی را آموزش می‌بیند و آن هم این است که داده های واقعی، لیبل درست دارند و داده های فیک می‌توانند لیبل اشتباه داشته باشند. در نتیجه ما به طور قطع نمی‌توانیم از خروجی شبکه متوجه شیم که لیبل درست است یا غلط ولی شبکه ما داده هایی که لیبل با تصویر هم خوانی ندارد را به احتمال قوی در دسته فیک شناسایی می‌کند ولی داده هایی که لیبل با محتوا یکی است لزوما داده واقعی شناسایی نمی‌کند.(به بیان دیگر این درست بودن لیبل را به عنوان یک فیچر بررسی می‌کند)

سوال سه پیاده سازی)

برای این بخش دو مدل ساده و پیچیده آموزش داده شده است.

مدل ساده در بخش مولد از relu و BatchNormalization بدون momentum استفاده شده است ولی در مدل پیچیده تر leakyrelu و momentum ۰.۸ در جاهای قبلی استفاده شده است که همگرایی را سرعت می‌بخشد.

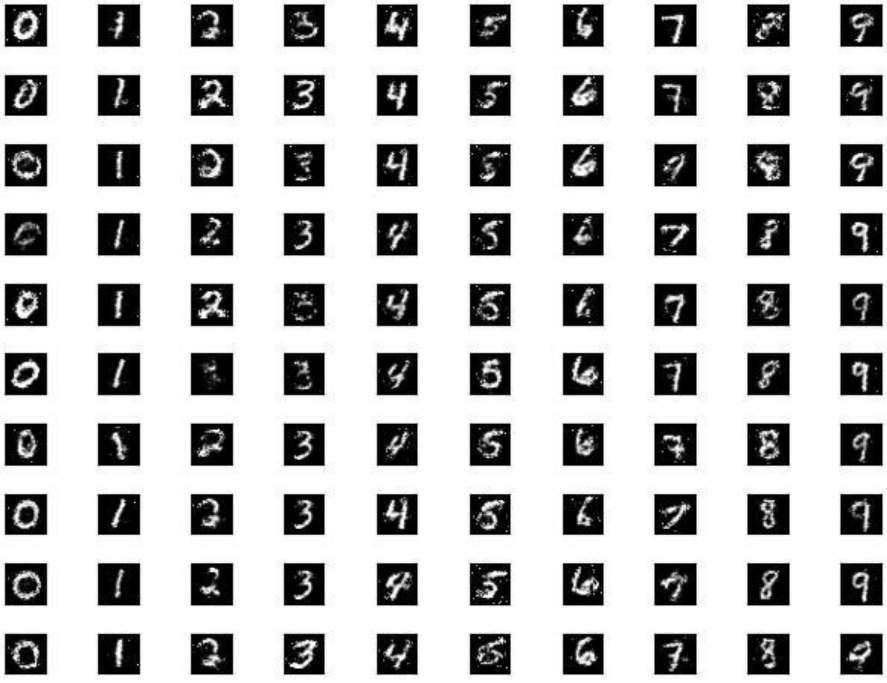
معماری مدل به چهار قسمت تقسیم شده است که در ابتدا مولد و لیبل ورودی به عنوان دو قسمت با یک دیگر ترکیب می‌شوند و خروجی آن به بخش بعدی که شامل دو بخش تمایزگر و لیبل ورودی آن است وارد می‌شوند(در نتیجه ۴ بخش کد وجود دارد که دو به دو ترکیب می‌شوند و در نهایت بصورت سری به هم وصل می‌شوند)



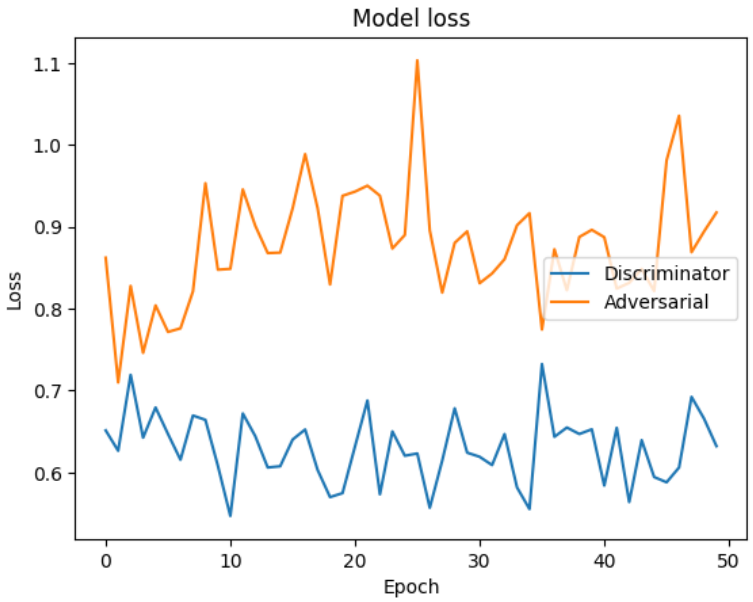
شکل شماتیک کلی هر دو مدل طراحی شده که همان معماری GAN هست

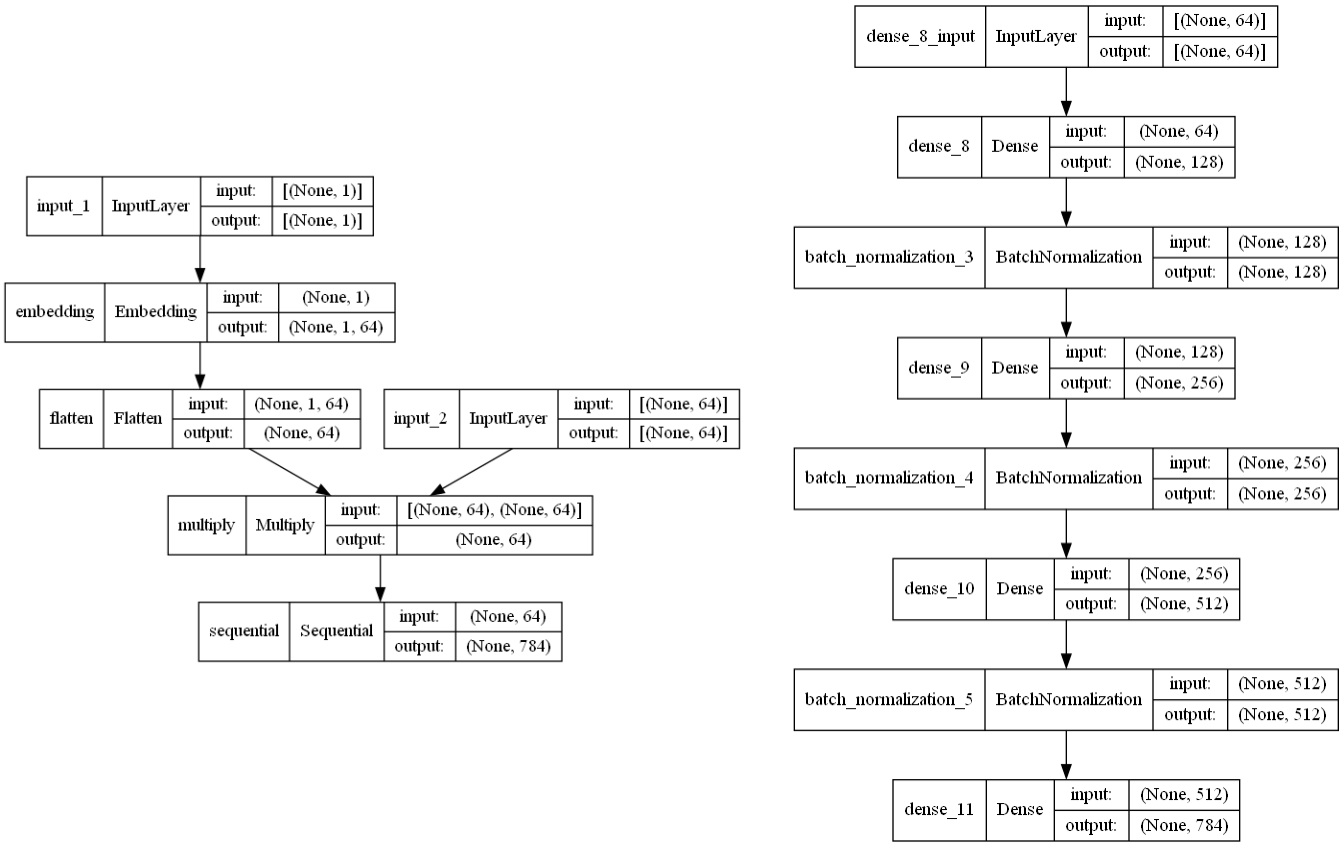
در ادامه جزئیات هر کدام از مدل ها را بیان می‌کنیم.

حالت ضعیف تر

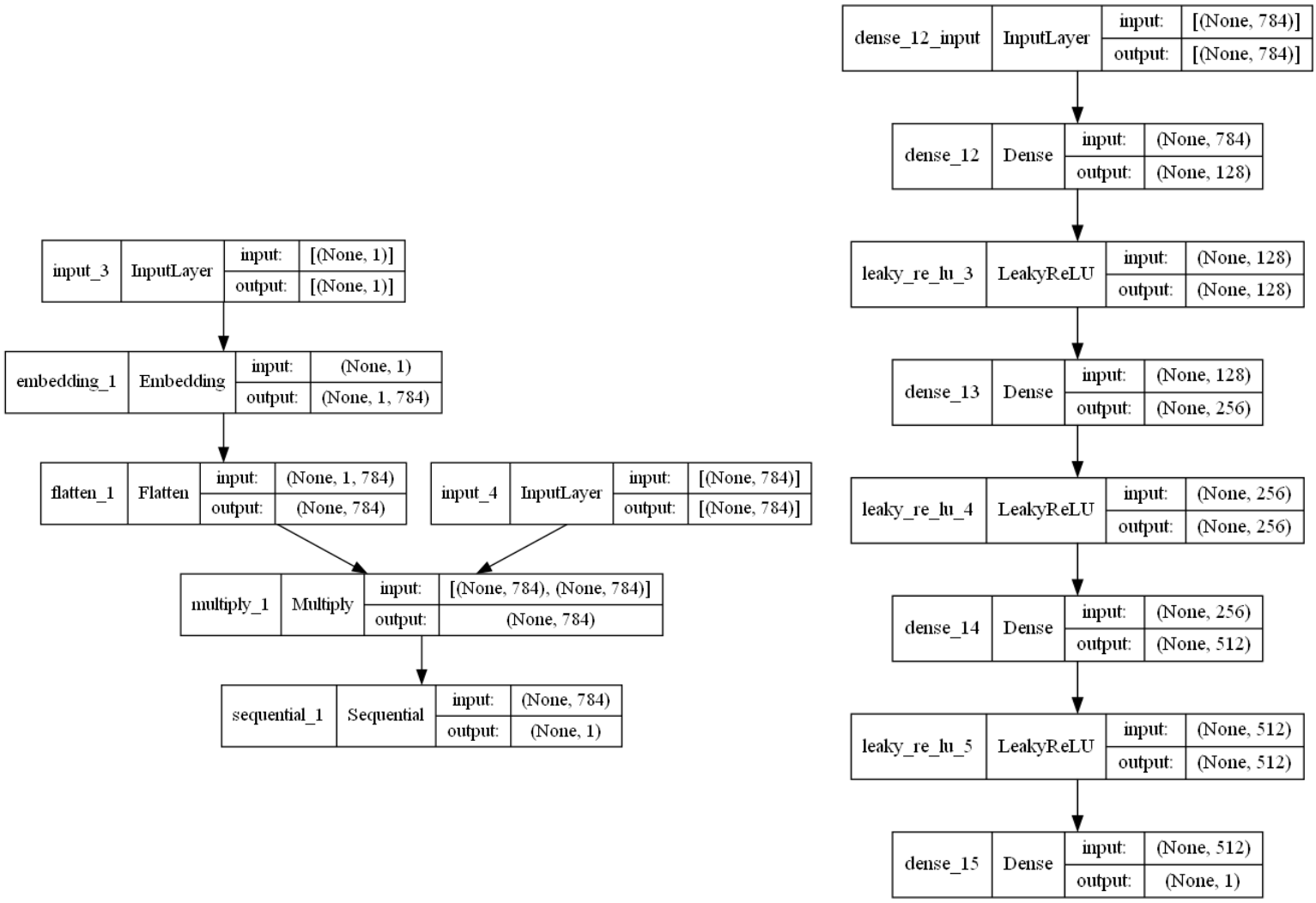


خروجی مدل ضعیف تر



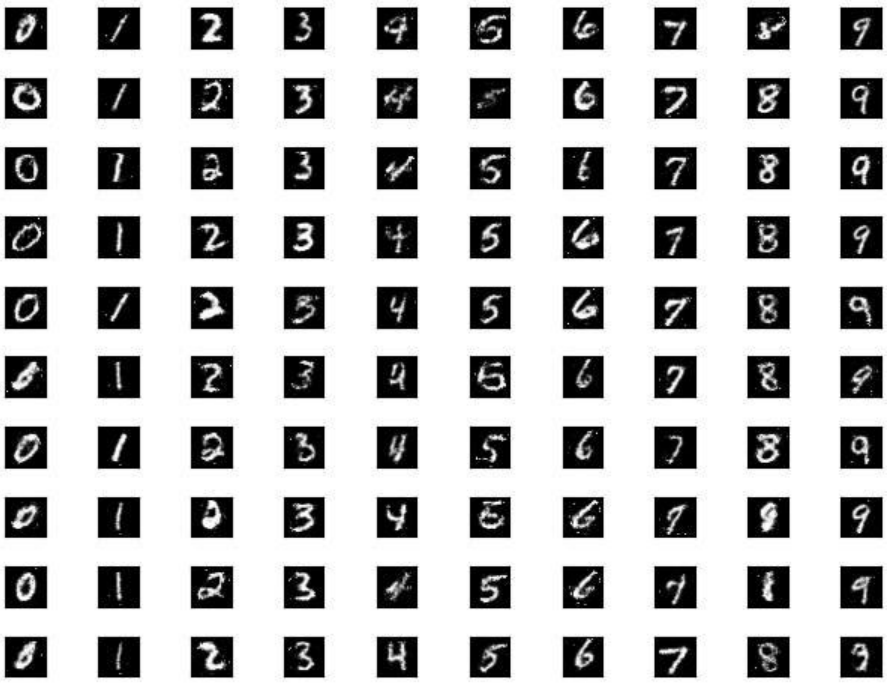


شکل سمت راست بخش sequential شکل سمت چپ می‌باشد و در امتداد یک دیگر این شکل بخش مولد را تشکیل می‌دهد

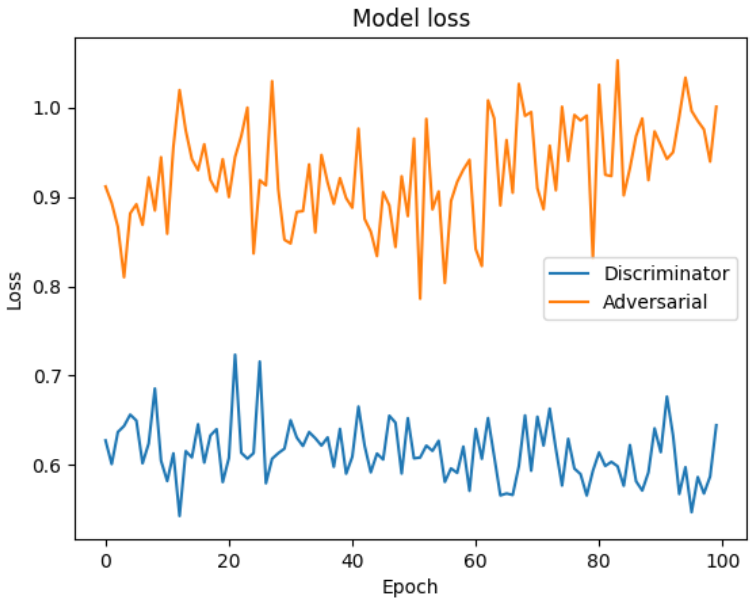


شکل سمت راست بخش sequential شکل سمت چپ می‌باشد و در امتداد یک دیگر این شکل بخش تمایز گر را تشکیل می‌دهد

حالت پیچیده تر:



مدل پیچیده تر



معماری این بخش دقیقا مشابه بخش قبلی فقط در بخش مولد تغییراتی داشته است که به آن در ابتدای این بخش اشاره شده است.

شکل فوق مربوط به بخش مولد می‌باشد.

** gif های مربوط به هر دو مدل در ضمیمه قرار داده شده اند.