سوال یک

در ابتدا یک توضیحی درباره روند CNN و attention برای تسک classification میدهم.

CNN :

۱. ورودی تصویر: ابتدا تصویر ورودی به شبکه داده می‌شود. تصویر معمولاً به صورت یک تانسور چندبعدی با ابعاد (ارتفاع، عرض، عمق) نمایش داده می‌شود.

۲. لایه‌های کانولوشن: در مرحله بعد، تصویر ورودی از طریق یک یا چند لایه کانولوشنی عبور می‌کند. هر لایه کانولوشنی یک مجموعه از فیلترهای کانولوشن را به تصویر ورودی اعمال می‌کند تا ویژگی‌های مکانی را استخراج کند. این ویژگی‌ها نمایش‌دهنده جزئیات مختلف تصویر هستند.

۳. لایه فعال‌سازی: پس از لایه کانولوشن، یک عملیات فعال‌سازی مانند ReLU (واحد خطی با انتقال غیرخطی) بر روی ویژگی‌های استخراج شده اعمال می‌شود. این عملیات به شبکه عصبی کمک می‌کند تا الگوهای غیرخطی را یاد بگیرد و اطلاعات بیشتری را در ویژگی‌ها نگه دارد.

۴. لایه‌های پولینگ: لایه‌های پولینگ برای کاهش ابعاد فضایی ویژگی‌ها استفاده می‌شوند. این لایه‌ها بخشی از ویژگی‌ها را با استفاده از روش‌هایی مانند حذف حداکثر (Max Pooling) یا میانگین‌گیری (Average Pooling)کاهش می‌دهند. این کاهش ابعاد می‌تواند مقاومت به تغییرات کوچک در موقعیت ویژگی‌ها را افزایش دهد و همچنین تعداد پارامترهای قابل آموزش در شبکه را کاهش دهد.

۵. لایه‌های متصل: پس از لایه‌های پولینگ، ویژگی‌های استخراج شده به لایه‌های متصل (Fully Connected)منتقل می‌شوند. در این لایه‌ها، ویژگی‌های مکانی متناظر با کلاس‌ها به یک بردار ویژگی نهایی تبدیل می‌شوند و کلاس مورد نظر تصویر را پیش بینی می کنیم.

Attention :

من این قسمت را با استفاده از معماری VGG16 که به ان attention اضافه شده است توضیح می دهم.

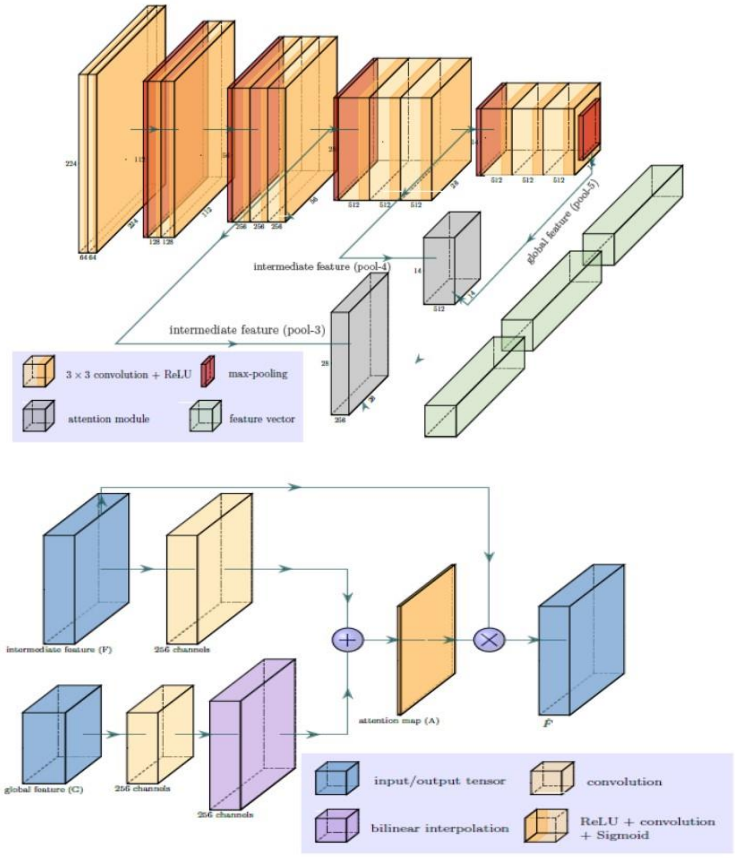
۱. لایه‌های کانولوشن اولیه: مراحل اولیه مدل VGG16 شامل چندین لایه کانولوشنی است که وظیفه استخراج ویژگی‌های مکانی تصویر را دارند. می‌توانید این لایه‌های کانولوشن را بدون تغییر در مدل VGG16استفاده کنید.

۲. لایه‌های کانولوشن با توجه (Attention-based Convolution) در این مرحله، می‌توانید یک Attention Mechanismرا به لایه‌های کانولوشن اضافه کنید. یک روش معمول برای اضافه کردن Attention Mechanismبه لایه‌های کانولوشن، استفاده از لایه‌های کانولوشن با توجه (Attention-based Convolution)است. این لایه‌ها با استفاده از اطلاعات کانال‌ها و ویژگی‌های مکانی تصویر، تاکید بیشتری روی بخش‌های مهم تصویر دارند. یک مثال از این لایه‌ها، لایه CBAM (Convolutional Block Attention Module)است.

۳. لایه‌های متصل: پس از لایه‌های کانولوشن با توجه، می‌توانید به ترتیب لایه‌های متصل معمولی را اعمال کنید. این لایه‌ها وظیفه تبدیل ویژگی‌های استخراج شده به بردار ویژگی نهایی را دارند. این بردار ویژگی نهایی می‌تواند به عنوان ورودی برای لایه‌های طبقه‌بندی بعدی استفاده شود.

۴. لایه طبقه‌بندی: در انتها، می‌توانید یک لایه طبقه‌بندی مانند Softmax را بر روی بردار ویژگی نهایی اعمال کنید تا احتمالات طبقه‌بندی را محاسبه کنید و کلاس تصویر را تشخیص دهید.

دو شکل زیر دید خوبی از اینکه attention چطور به مدل اضافه کنیم می دهد:



در واقع کاری که با اضافه کردن attention انجام می دهیم این است که به قسمت های مهم تصویر بیتر توجه می کنیم و به قسمت هایی از تصویر که اهمیت زیادی ندارند توجه کمتری داریم.

تشخیص گربه بودن یا نبودن:

- CNN: در معماری CNN معمولی، لایه‌های کانولوشنی با وزن‌های ثابت بر روی تصویر اعمال می‌شوند تا ویژگی‌های مکانی تصویر استخراج شوند. سپس با استفاده از لایه‌های متصل، این ویژگی‌ها به یک بردار ویژگی نهایی تبدیل می‌شوند. سپس این بردار ویژگی به یک لایه طبقه‌بندی مانند Softmaxداده می‌شود تا احتمال تعلق به دسته گربه و غیر گربه محاسبه شود.
- Attention Mechanism: برای اضافه کردن Attention Mechanism به CNN برای تشخیص گربه، می‌توانیم از روش‌های مختلفی استفاده کنیم. یک روش معمول برای اعمال Attention در CNN، استفاده از لایه‌های کانولوشن با توجه (Attention-based Convolution) است. این لایه‌ها با استفاده از ورودی‌های کانال‌ها و ویژگی‌های مکانی تصویر، تاکید بیشتری روی بخش‌های مهم تصویر دارند. این تاکید می‌تواند به برجسته‌سازی ویژگی‌های مهمی که با تشخیص گربه مرتبط هستند، کمک کند . مثلا به قسمت های مهم مانند مدل گوش ها چشم ها توجه بیشتری میکند.

تشخیص انسان بودن یا نبودن:

- CNN :برای تشخیص مسئله انسان بودن یا نبودن از معماری CNN می‌توانیم استفاده کنیم. شبکه CNNبا استفاده از لایه‌های کانولوشنی و لایه‌های متصل، ویژگی‌های مکانی تصویر را استخراج کرده و به لایه طبقه‌بندی ارسال می‌کند. لایه طبقه‌بندی می‌تواند احتمال تعلق تصویر به دسته انسان و غیر انسان را محاسبه کند.
- Attention Mechanism :برای اضافه کردن Attention Mechanism به CNN برای تشخیص مسئله انسان بودن یا نبودن، نیاز به مدل‌سازی توجه بر روی ویژگی‌های مرتبط با انسان داریم. می‌توانید از روش‌های مختلفی برای اعمال Attention استفاده کنید. یک روش معمول برای این منظور استفاده از Spatial Attention است که با استفاده از مکانیزم توجه، توجه را به بخش‌هایی از تصویر که مرتبط با انسان هستند، جلب می‌کند. این مکانیزم می‌تواند با استفاده از لایه‌های کانولوشنی با توجه یا لایه‌های کاملاً متصل با توجه پیاده‌سازی شود. این لایه‌ها با در نظر گرفتن اطلاعات مکان و کانال‌ها، توجه بیشتری به ویژگی‌های مهم تصویر مرتبط با انسان دارند.به گوش چشم لب و دهن که قسمت ها مهم برای تشخیص انسان هستند توجه بیشتری می کند.

سوال دو

الف)

False Positive: مفهوم بیانگر تعداد نمونه‌هایی است که به طور اشتباهی تشخیص داده می‌شوند که متعلق به دسته پیش‌بینی شده نیستند.به عبارت دیگر، FPشامل مواردی است که مدل برچسب مثبت را به نمونه‌های منفی تعلق می‌دهد. مدل برچسب مثبت را پیش بینی میکند با اینکه برچسب انها منفی است.

True Positive :مدل برچسب انها را مثبت پیش بینی میکند و برچسب انها واقعا مثبت نیز هست در واقع مدل درست پیش بینی میکند.

True Negative:مدل برچسب انها را منفی پیش بینی میکند و برچسب انها واقعا منفی هست در واقع مدل درست پیش بینی میکند.

False Negative :مدل برچسب انها را منفی پیش بینی میکند با اینکه برچسب انها مثبت است در واقع مدل اشتباه پیش بینی میکند.

ب)

با توجه به اهمیت اشتباه تشخیص ندادن افراد بی گناه به عنوان مجرم در پروژه تشخیص مجرمان هک اسنپ فود، یک معیار ارزیابی مهم که میتوانیم استفاده کنیم recall است.فرمول ان به صورت زیر است:

$$Recall = \frac{TP}{TP + FN}$$

که در آن TP تعداد مجرمانی است که به درستی تشخیص داده شده‌اند و FN تعداد مجرمانی است که به اشتباه به عنوان غیرمجرم تشخیص داده شده‌اند.

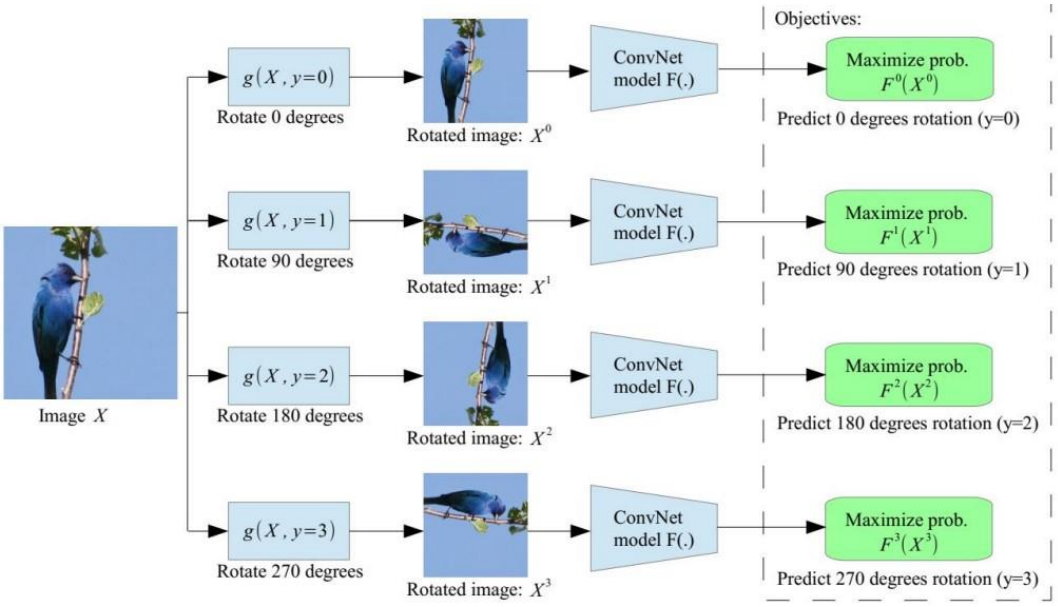
با استفاده از معیار حساسیت (TPR) می‌توانیم ارزیابی کنیم که چقدر مدل قادر به تشخیص و شناسایی مجرمان واقعی است. با تمرکز بر این معیار، سعی در کاهش تعداد اشتباهات نادیده گرفتن مجرمان واقعی (FN) دارید.

یک کار دیگر نیز که میتوانیم انجام دهیم امتناع از تصمیم گیری توسط مدل هست وقتی مثلا به یک نفر ۵۲درصد مثبت توسط مدل پیش بینی شده تصمیم گیری نکند و تصمیم ان را به انسان بسپارد درواقع یک threshold بگذاریم برای اینکه تصمیم بگیریم یا نه.

سوال سه

الف)

در مسئله چرخش که در SSL استفاده میشه ما تصویر را ۰و۹۰و۱۸۰و۲۷۰ چرخش میدهیم و مدل باید پیش بینی کند که تصویر الان نسبت به تصویر اصلی چند درجه چرخش یافته است این باعث میشه مدل بتونه یکسری فیچر از تصویر یادبگیره چون برای اینکه تشخص بده تصویر چند درجه چرخیده باید قسمت های مهم تصویر را یاد بگیرد



مثلا اینجا باید برای فهمیدن میزان چرخش نوک دم پرنده را تشخیص بده.

ب)

بردارهای one hot به یک نوع از بردارهای دودویی اشاره دارد که برای نشان دادن دسته‌ها یا برچسب‌ها استفاده می‌شوند. در این نوع بردارها، هر بردار متشکل از تعدادی عنصر است، و تنها یکی از این عناصر برابر با یک است و سایر عناصر برابر صفر هستند.

برای مثال، فرض کنید که یک مدل زبانی وجود دارد که قصد تشخیص دسته‌های مختلف اسناد را دارد. اگر دسته‌های ممکن شامل "ورزش"، "سیاست" و "هنر" باشند، بردارهایone hot برای یک سند ورودی به صورت زیر خواهند بود:

- برداری که بیانگر دسته "ورزش" است[1, 0, 0]

- برداری که بیانگر دسته "سیاست" است[0, 1, 0]
- برداری که بیانگر دسته "هنر" است[1, 0, 0]

استفاده از بردارهای one hot برای نشان دادن دسته‌ها مشکلاتی نیز دارد. یکی از این مشکلات این است که این بردارها بسیار فضای حافظه را اشغال می‌کنند، زیرا برای هر دسته باید یک بردار جداگانه ایجاد شود. اگر تعداد دسته‌ها زیاد باشد، ممکن است حجم حافظه مورد نیاز برای نگهداری این بردارها به طور قابل توجهی افزایش یابد.

علاوه بر این، بردارهای one hot به طور مستقیم اطلاعاتی درباره رابطه و شباهت بین دسته‌ها ارائه نمی‌دهند. به عبارت دیگر، در این بردارها هیچ ارتباط معناداری بین دسته‌ها وجود ندارد و به مدل یادگیری عمیق اجازه نمی‌دهد تا الگوها و ویژگی‌های مشترک بین دسته‌ها را استخراج کند.

به طور کلی، استفاده از بردارهای one hot در مواردی که تعداد دسته‌ها کم و محدود است و احتمال بروز ترتیب مهم نیست، مناسب است. اما در مواردی که تعداد دسته‌ها زیاد است و یا رابطه و شباهت بین دسته‌ها اهمیت دارد، روش‌های دیگری که تعداد دسته‌ها زیاد است و یا رابطه و شباهت بین دسته‌ها اهمیت دارد، روش‌های دیگری مانند بردارهای تعبیه شده (embedding vectors) می‌توانند گزینه بهتری باشند.

ج)

Word2Vecیک الگوریتم نشانه‌گذاری (embedding) و یک روش برای نمایش کلمات به صورت برداری است که بر اساس self-supervisedعمل می‌کند. الگوریتم‌های self-supervised معمولاً برای یادگیری نشانه‌گذاری‌های مفید و منظم از داده‌های بدون برچسب استفاده می‌کنند، و هدف آنها بازنمایی هست.

Word2Vecبر اساس ساختار جملات و روابط بین کلمات در متن‌ها، نشانه‌گذاری برداری برای کلمات را ایجاد می‌کند. برای این منظور، دو روش اصلی در Word2Vec وجود دارد Skip-gramو CBOW Bag of در هر دو روش، یک مدل شبکه عصبی با دو لایه مخفی (یک لایه ورودی و یک لایه خروجی) برای یادگیری نشانه‌گذاری کلمات استفاده می‌شود.

در حالتSkip-gram ، کلمه‌ی ورودی (مرکزی) تلاش می‌کند تا کلمات اطراف خود را پیش‌بینی کند، در حالی که در حالتCBOW ، کلمات اطراف ورودی (مرکزی) به عنوان ورودی به مدل داده می‌شوند و مدل سعی می‌کند کلمه‌ی مرکزی را پیش‌بینی کند. این فرآیند ادامه می‌یابد تا کلمات مختلف در متن به صورت مکرر و بازه‌های زمانی مختلف برای یادگیری نمایش برداری استفاده شوند.

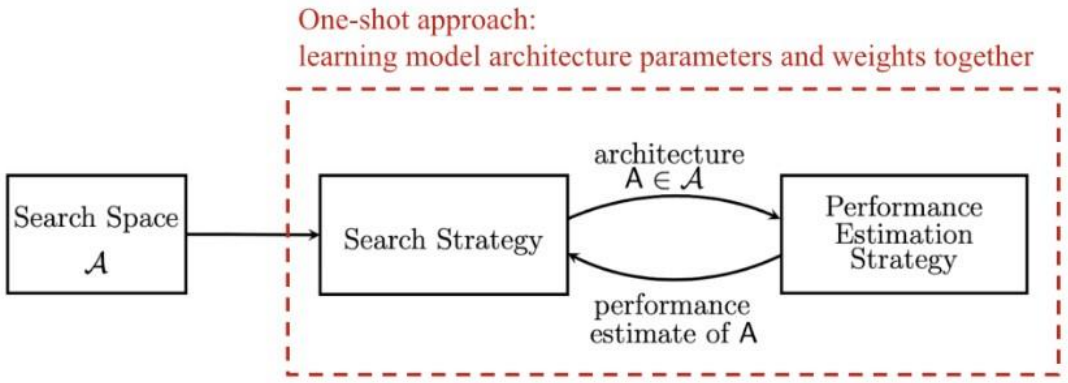
بنابراین، Word2Vecبا استفاده از خودنظارت و بر پایه تلاش برای پیش‌بینی کلمات از متن‌ها، به صورت خودکار نشانه‌گذاری برداری برای کلمات را ایجاد می‌کند. این تطابق با الگوریتم‌های self-supervised است.

سوال چهار

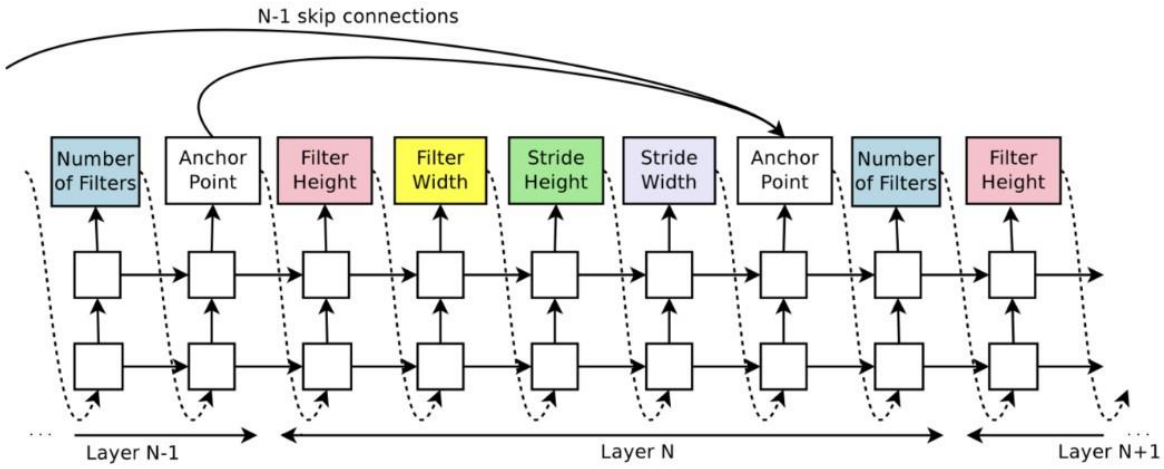
الف)

در این رویکرد، عامل یادگیری تقویتی مسئله جستجوی ساختار شبکه را به عنوان یک فضای عمل گسترده در نظر می‌گیرد. هر عمل در این فضا معادل با یک تغییر در ساختار شبکه است، مانند افزودن یک لایه جدید، تغییر تعداد نرون‌ها در لایه‌ها، اعمال تغییرات در توابع فعال‌سازی و غیره. هدف عامل، یافتن ساختار شبکه‌ای است که عملکرد بهتری دارد.

برای ارزیابی هر ساختار شبکه، عامل از تجربیات خود در محیط آموزشی استفاده می‌کند و میزان rewardرا به عنوان سیگنال تقویت استفاده می‌کند. با انجام اعمال مختلف در فضای عمل و بررسی نتایج، عامل به تدریج ترکیبی از عملکردهای مختلف شبکه‌ها را که عملکرد بهتری دارند، یاد می‌گیرد



یکی از چالش‌های اصلی در جستجوی ساختار شبکه با استفاده از یادگیری تقویتی، فضای بزرگ و پیچیده عملکردهای شبکه است. برای حل این مسئله، مقاله مورد نظر از روش‌هایی مانند شبکه‌های عصبی بازگشتی (RNN)برای کاهش ابعاد فضای عمل استفاده می‌کند. با این روش، عامل می‌تواند یک ساختار شبکه را به صورت تدریجی بسازد، لایه به لایه، و در هر مرحله تنها نیاز به تصمیم‌گیری در مورد یک عملکرد ساده‌تر دارد.



ب)

رویکرد جستجوی معماری شبکه عصبی می‌تواند برای تشخیص اشیا در مسئله‌های بینایی کامپیوتری مفید باشد. اما در انتخاب ابعاد تصویر و تعداد لایه‌های شبکه، باید به چندین عامل توجه کرد. در ادامه به بررسی این عوامل می‌پردازم:

۱. ابعاد تصویر ورودی: انتخاب ابعاد تصویر ورودی بر اساس مسئله ورودی و نوع داده‌های مورد استفاده متفاوت است. ابعاد تصویر می‌تواند تأثیر زیادی بر عملکرد مدل داشته باشد. ابعاد بزرگتر تصویر می‌توانند اطلاعات بیشتری را دربرگیرند، اما هزینه محاسباتی بیشتری نیز دارند و ممکن است نیاز به ظرفیت پردازشی بیشتری داشته باشند. همچنین، اگر ابعاد تصویر خیلی کوچک باشد، ممکن است اطلاعات مهمی از بین برود و عملکرد مدل کاهش یابد. بنابراین، انتخاب ابعاد مناسب بر اساس توجه به توازن بین دقت و هزینه محاسباتی ضروری است.

۲. تعداد لایه‌ها: تعداد لایه‌های شبکه نیز بر عملکرد مدل تأثیرگذار است. تعداد لایه‌ها می‌تواند تعیین‌کننده قدرت تقریب‌زنندگی شبکه باشد. شبکه‌های عمیق‌تر معمولاً توانایی یادگیری الگوهای پیچیده‌تر را دارند، اما افزایش تعداد لایه‌ها ممکن است باعث افزایش هزینه محاسباتی و مشکلاتی مانند بیش‌برازش شود. همچنین، تعداد لایه‌ها باید با مقدار داده موجود و پیچیدگی مسئله سازگار باشد. برای مسئله تشخیص اشیا، معمولاً از شبکه‌های عمیق مانند شبکه‌های عصبی پیچشی عمیق (Deep Convolutional Neural Networks)استفاده می‌شود، که تعداد لایه‌های بالا را دارند.با توجه به این نکات، استفاده از رویکرد جستجوی معماری شبکه عصبی می‌تواند به تنظیم و بهبود عملکرد مدل در مسئله تشخیص اشیا کمک کند.

سوال پنج

وقتی تابع ضرر مولد و ممیز در پایان اپوک اول و ۱۰۰ به طور تقریبی یکسان باشد، این به معنای توازنی بین مولد و ممیز در این دو مرحله است. اما این فقط نشان دهنده توازن در تابع ضرر است و نمی‌تواند کاملاً تضمین کند که تصاویر تولید شده در این دو اپوک مشابه باشند.

در شبکه‌های مولد معادله‌ای به نام معادله توازن نیاز است که مولد و ممیز به یک نقطه توازن برسند. این به این معنی است که مولد باید تلاش کند تا تولیدهایی انجام دهد که ممیز را به اشتباه بین داده‌های واقعی و داده‌های تولید شده قرار دهد، در حالی که ممیز باید تلاش کند تا بین داده‌های واقعی و داده‌های تولید شده تمایز ایجاد کند.

تابع ضرر ممیز و مولد در یک شبکه GAN یک نشانگر از توازن بین دو شبکه است و نشان می‌دهد که هر دو شبکه در حال یادگیری هستند. اما کیفیت تصاویر تولید شده در اپوک اول و ۱۰۰ ممکن است متفاوت باشد زیرا در طی آموزش، شبکه ممکن است درک بهتری از داده‌ها و الگوهای آنها پیدا کند و عملکرد بهتری در تولید تصاویر داشته باشد. همچنین، ممکن است مولد در اپوک اول تولیدهای تصادفی و بدون ساختار بیشتری انجام دهد و در اپوک ۱۰۰ بهبودهای قابل توجهی در تولید تصاویر داشته باشد.

بنابراین، علت عدم تطابق کیفیت تصاویر تولید شده در اپوک اول و ۱۰۰ می‌تواند به دلایلی مانند تغییر در نحوه یادگیری شبکه، بهبود عملکرد مولد در طول زمان، یا فرایندهای تصادفی مرتبط با آموزش شبکه GAN بازگردانده شود.

$$\min_{\theta_g} \max_{\theta_d} \left[\mathbb{E}_{x \sim p_{data}} \log \underbrace{D_{\theta_d}(x)}_{\text{Discriminator output for real data x}} + \mathbb{E}_{z \sim p(z)} \log(1 - \underbrace{D_{\theta_d}(G_{\theta_g}(z))}_{\text{Discriminator output for generated fake data G(z)}}) \right]$$

در واقع طیق فرمول بالا ممیز سعی میکنه که داده واقعی از فیک تشخیص بده و مولد سعی میکنه داده ای تولید کنه که که ممیز به اشتباه بندازه و داده رو واقعی پیش بینی کنه یعنی عملکرد این دو شبکه دقیقا عکس هم هستند و طبق فرمول بالا ممکن هست مقدار تابع ضرر در دو اپک مختلف یکی بشه ولی الان هم مولد داده بهتر تولید میکنه و هم ممیز قدرتش بالاتر رفته.