

# به نام خدا

تمرین سری ششم  
درس مبانی یادگیری عمیق  
دکتر مرضیه داوود آبادی

فرزان رحمانی  
۹۹۵۲۱۲۷۱

سوال اول

(آ)

۱. شبکه عصبی کانولوشنی (CNN):

یک CNN داده های تصویری از گربه ها و غیرگربه ها را می گیرد. برای استخراج ویژگی های فضایی از تصاویر، مانند لبه ها، شکل ها، بافت ها و غیره، از لایه های کانولوشنال استفاده می کند. این شبکه نمایش های سلسله مراتبی از ویژگی های بصری گربه را می آموزد.

- یک CNN برای این مسئله احتمالاً چندین لایه کانولوشن دارد و به دنبال آن لایه های pooling برای استخراج ویژگی ها از تصاویر ورودی.
- لایه های اولیه ممکن است لبه ها یا الگوهای ساده را تشخیص دهند، در حالی که لایه های عمیق تر ویژگی های پیچیده تر مخصوص گربه ها (مانند بافت پوست، سبیل، گوش و غیره) را تشخیص می دهند.
- لایه های نهایی از لایه های کاملاً متصل تشکیل می شوند تا بر اساس ویژگی های استخراج شده تصمیم بگیرند (گربه یا نه).
- شبکه بر روی یک مجموعه داده حاوی تصاویر برجسب گذاری شده از گربه ها و غیرگربه ها آموزش داده می شود و با تنظیم پارامترهای داخلی (وزن ها و سوگیری ها) بین آن ها تمایز قائل می شود.

تصویر عادی گربه:



شبکه CNN میتواند با روش بالا تصویر گربه عادی را به درستی تشخیص دهد. مثلاً با استفاده از یکی از شبکه های (AlexNet, VGG-16, GoogLeNet, ResNet-50, ResNet-152, DenseNet-121, SqueezeNet1\_1) می توانید تصویر عادی گربه را تشخیص دهید.

## تصویر گربه با بافت پوست فیل:



(a) Texture image	(b) Content image	(c) Texture-shape cue conflict
81.4% <b>Indian elephant</b>	71.1% <b>tabby cat</b>	63.9% <b>Indian elephant</b>
10.3% indri	17.3% grey fox	26.4% indri
8.2% black swan	3.3% Siamese cat	9.6% black swan

با توجه به لینکی که در مرجع سوال گذاشتم CNN نمی تواند این تصویر را به درستی طبقه بندی کند و این تصویر را فیل هندی شناسایی می کند. دلیل آن هم این است که CNN به بافت تصویر توجه بیشتری از شکل آن می کند.

*texture hypothesis*: textures, not object shapes, are the most important aspects of an object for AI object recognition.

در واقع، CNN برای تشخیص اشیا در درجه اول بر بافتها تکیه می کنند، نه بر اشکال اشیا. برای حل این مشکل می توانیم در داده های آموزشی CNN از تکنیک style transfer برای هر عکس مثلاً تولید ۱۰ عکس با shape یکسان و texture های مختلف استفاده کنیم.

## ۲. شبکه مبتنی بر توجه:

یک شبکه مبتنی بر توجه نیز می تواند داده های تصویر را بگیرد. از مکانیسم های توجه برای تمرکز بر مرتبط ترین بخش های تصویر که نشان دهنده «گربه بودن» هستند، مانند چشم ها، گوش ها، سبیل ها و غیره استفاده می کند. وزن های توجه، نمایش ویژگی را برای تقویت ویژگی های گربه تغییر می دهند. این فوکوس انتخابی به شبکه کمک می کند تا تشخیص دهد آیا تصویر حاوی گربه است یا خیر.

- یک مدل مبتنی بر توجه ممکن است بر روی مناطق خاصی از یک تصویر تمرکز کند که برای تعیین اینکه آیا تصویر حاوی گربه است یا خیر، بسیار مهم است. یا همچنین می تواند انتخاب کند که به شکل بیشتر توجه کند یا بافت تصویر یا رنگ آن.
- در ابتدا، مدل ممکن است از یک رمزگذار کانولوشن برای استخراج ویژگی ها از تصویر ورودی استفاده کند.
- سپس، به جای طبقه بندی مستقیم کل تصویر، مکانیسم توجه به شبکه کمک می کند تا روی بخش های مهم تصویر (مانند ویژگی های خاص گربه مانند چشم ها، بینی و غیره) تمرکز کند.
- با اختصاص وزن های بالاتر به این مناطق یا ویژگی های حیاتی، مکانیسم توجه شبکه را قادر می سازد تا بر اساس این بخش های متمرکز تصمیم گیری کند.

تصویر عادی گربه:



شبکه مبتنی بر توجه می تواند تصویر عادی گربه را تشخیص دهد.

تصویر گربه با بافت پوست فیل:



شبکه مبتنی بر توجه با طراحی مناسب می تواند تصویر گربه با بافت پوست فیل را تشخیص دهد. چرا که میتواند تصمیم بگیرد که مثلاً به شکل تصویر بیشتر از بافت و پس زمینه آن توجه کند و از شبکه CNN بهتر عمل کند.

(ب)

۱. شبکه عصبی کانولوشنی (CNN):

یک CNN داده‌های تصویری انسان‌ها و غیرانسان‌ها را می‌گیرد. ویژگی‌های فضایی مانند چشم‌ها، بینی، اندام‌ها را برای شناسایی شکل انسان استخراج می‌کند. لایه‌های کاملاً متصل طبقه‌بندی می‌کنند که آیا ویژگی‌ها یک انسان را نشان می‌دهند. شبکه سلسله مراتب ویژگی‌های بصری انسان را می‌آموزد.

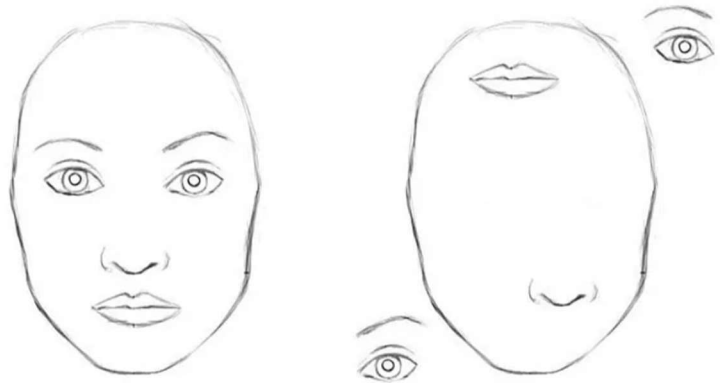
- مشابه مسئله طبقه‌بندی گربه، یک CNN برای طبقه‌بندی انسان از معماری مشابه پیروی می‌کند.
- CNN ویژگی‌هایی را از تصاویر می‌آموزد که برای تشخیص انسان از غیر انسان مرتبط تر است.
- ویژگی‌های آموخته شده ممکن است شامل ساختارهای صورت، اشکال بدن و سایر ویژگی‌های خاص انسان باشد.
- پس از استخراج ویژگی، لایه‌های کاملاً متصل اطلاعات را برای طبقه‌بندی نهایی پردازش می‌کنند.

## تشخیص تصویر صورت عادی انسان:



شبکه CNN میتواند با روش بیان شده تصویر صورت عادی انسان را به درستی تشخیص دهد. مثلاً با استفاده از یکی از شبکه های (VGG-16, GoogLeNet, ResNet-50) می توانید تصویر عادی انسان را تشخیص دهید.

## تشخیص تصویر صورت انسان با عناصر بهم ریخته مکانی:



A CNN would classify both images as faces because they both contain the elements of faces. [Source](#).

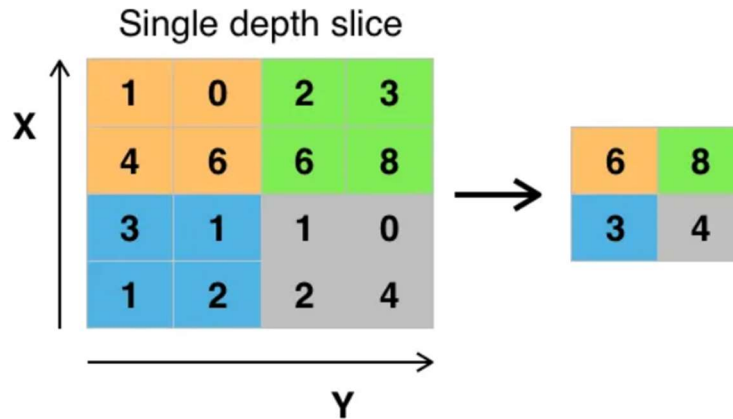
همچنین یک CNN تصویر بهم ریخته را به عنوان صورت انسان شناسایی میکند که درست نیست و نباید آن را به عنوان صورت انسان شناسایی کند. دلیل آن هم استفاده از max-pooling است.

در لینک موجود در مراجع دلیل آن به تفصیل آمده است در اینجا صرفاً توضیح کوتاهی می دهیم:

در اصل، CNN ها با تشخیص وجود ویژگی ها در یک تصویر، و استفاده از دانش این ویژگی ها برای پیش بینی وجود یک شی کار می کنند. با این حال، CNN ها فقط وجود ویژگی ها را تشخیص می دهند - بنابراین حتی تصویر سمت راست که حاوی چشم، بینی و گوش نامناسب است، به عنوان چهره در نظر گرفته می شود زیرا شامل تمام ویژگی های لازم است!

مشکل به نحوه انتقال اطلاعات از لایه ای به لایه دیگر توسط CNN ها خلاصه می شود: حداکثر ادغام. Max Pooling شبکه ای از پیکسل ها را بررسی می کند و حداکثر فعال سازی را در آن ناحیه انجام می دهد. این روش اساساً تشخیص می دهد که آیا یک ویژگی در هر منطقه ای از تصویر وجود دارد یا خیر، اما اطلاعات مکانی مربوط به ویژگی را از دست می دهد.

یک راه حل برای این مشکل استفاده از Capsule Neural Networks است.



Max Pooling preserves existence (maximum activation) in each region, but loses spatial information. [Source](#).

۲. شبکه مبتنی بر توجه:

یک مدل توجه تصاویر را می گیرد و قسمت های مربوطه را که نشانه یک انسان است، شناسایی می کند، مانند صورت، چشم، لب، بینی، بازوها، پاها و غیره. وزن توجه برای این قسمت ها بیشتر است و این ویژگی ها را افزایش می دهد. این ویژگی های انسانی را برجسته می کند تا به طبقه بندی اینکه آیا تصویر دارای یک انسان است یا نه کمک می کند.

- برای تعیین اینکه آیا یک تصویر حاوی یک انسان است یا خیر، یک مدل مبتنی بر توجه ممکن است بر روی مناطق خاصی تمرکز کند که نشان دهنده حضور انسان است.
- ممکن است به نواحی مانند صورت، اندام ها، یا سایر قسمت های مخصوص انسان در تصویر توجه کند.
- مکانیسم توجه مدل را راهنمایی می کند تا هنگام تصمیم گیری در مورد اینکه آیا تصویر حاوی یک انسان است یا خیر، اهمیت بیشتری به این مناطق بدهد.

تشخیص تصویر صورت عادی انسان:



شبکه مبتنی بر توجه میتواند با روش بیان شده تصویر صورت عادی انسان را به درستی تشخیص دهد.

## تشخیص تصویر صورت انسان با عناصر بهم ریخته مکانی:



با مکانیزم توجه می توان تصویر درست از غلط انسان را شناسایی کرد و فقط چهره ی سمت چپ به عنوان چهره ی انسان شناسایی شود. در واقع شبکه مبتنی بر توجه تصویر بهم ریخته را به عنوان انسان شناسایی نمیکند. چرا که با استفاده از توجه می تواند تشخیص دهد که از نظر موقعیت به کجای تصویر توجه کند و صرفاً بر اساس وجود یک ویژگی به طبقه بندی نپردازد. برای مثال توجه می کند که باید چشم های در بالای بینی و لب در زیر بینی باشد ن اینکه صرفاً بر اساس وجود چشم، بینی و لب در هر جای تصویر مانند CNN به وجود انسان اشاره کنند.

شبکه مبتنی بر توجه سعی میکنند محدودیتهای CNN را با در نظر گرفتن روابط جزئی از کل برطرف کنند. به عنوان مثال، در یک صورت، هر یک از دو چشم بخشی از یک پیشانی است. برای درک این روابط جزئی، مکانیزم نیاز به دانستن بیشتر از وجود یک ویژگی دارند. باید بدانند ویژگی کجاست، چگونه جهت گیری میکند و ... .

مکانیزم های توجه به مدل اجازه میدهد تا بر مرتبط ترین و مهم ترین بخشهای ورودی تمرکز کند، که میتواند دقت و استحکام را به ویژه برای ورودی های پیچیده بهبود بخشد. علاوه بر این، میتواند هزینه محاسباتی و استفاده از حافظه را با انتخاب و پردازش تنها بخشهای مهم ورودی کاهش دهد.

استفاده از مکانیزم توجه در داده های تصویری، یک روش است که برای بهبود عملکرد شبکه های عصبی کانولوشنی در پردازش تصاویر استفاده می شود. این روش با اجازه دادن به شبکه برای توجه بر روی بخش های مهم و مرتبط با تصویر در حین تولید پیش بینی استفاده می شود. این کار با وزن دادن به فیلتر (kernel) های شبکه انجام می شود تا شبکه برخی از ویژگی های تصویر را بر سایرین اولویت دهد. در نتیجه، شبکه می تواند پیش بینی های دقیق تری را تنها با در نظر گرفتن مهم ترین تغییرات تصویر تولید کند.

مکانیزم توجه می تواند در موارد مختلف پردازش تصویر مانند تشخیص چهره، شناسایی اشیاء، تولید شرح برای تصاویر و ... استفاده شود. در این موارد، مکانیزم توجه به شبکه کمک می کند تا روی اشیاء یا ویژگی های خاصی در تصویر خاص کند و از این طریق ویژگی ها و اطلاعات تصویر را می کند.

به طور خلاصه، شبکه های کانولوشنال در استخراج ویژگی از تصاویر (سلسله مراتب فضایی ویژگی ها) برتری دارند، در حالی که مدل های مبتنی بر توجه با اختصاص درجات مختلف اهمیت به مناطق مختلف، بر بخش های مربوطه از داده های ورودی تمرکز می کنند و به فرآیندهای تصمیم گیری کمک می کنند. هر دو رویکرد بسته به ویژگی های مسئله و داده های موجود می توانند مؤثر باشند.

مراجع:

<https://chat.openai.com/>

<https://medium.com/@robertgeirhos/why-deep-learning-works-differently-than-we-thought-ec28823bdbc>

<https://towardsdatascience.com/demystifying-matrix-capsules-with-em-routing-part-1-overview-2126133a8457>

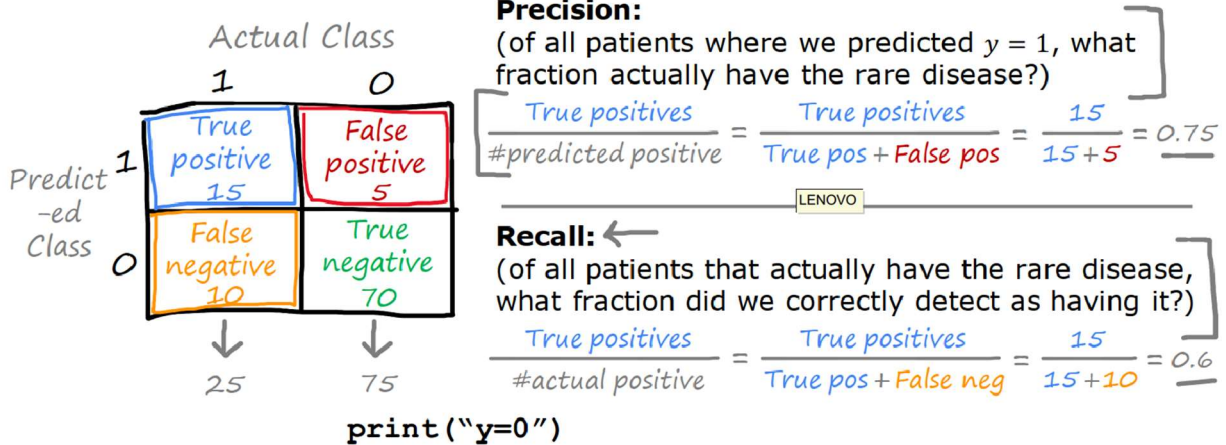
<https://projects-and-notes.readthedocs.io/en/latest/1811.12231.html>

[https://www.semanticscholar.org/paper/Are-Convolutional-Neural-Networks-or-Transformers-Tuli-](https://www.semanticscholar.org/paper/Are-Convolutional-Neural-Networks-or-Transformers-Tuli-Dasgupta/4307e8c858aa4f7ed547bc38bb7636de7755220f)

[Dasgupta/4307e8c858aa4f7ed547bc38bb7636de7755220f](https://www.semanticscholar.org/paper/Are-Convolutional-Neural-Networks-or-Transformers-Tuli-Dasgupta/4307e8c858aa4f7ed547bc38bb7636de7755220f)



$y = 1$  in presence of rare class we want to detect.



## Trading off precision and recall

Logistic regression:  $0 < f_{\vec{w},b}(\vec{x}) < 1$

→ Predict 1 if  $f_{\vec{w},b}(\vec{x}) \geq 0.5$

→ Predict 0 if  $f_{\vec{w},b}(\vec{x}) < 0.5$

$\text{precision} = \frac{\text{true positives}}{\text{total predicted positive}}$

$\text{recall} = \frac{\text{true positives}}{\text{total actual positive}}$

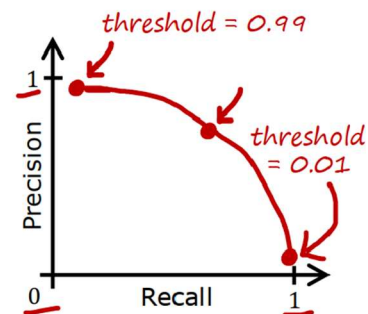
Suppose we want to predict  $y = 1$  (rare disease) only if very confident.

→ higher precision, lower recall.

Suppose we want to avoid missing too many cases of rare disease (when in doubt predict  $y = 1$ )

→ lower precision, higher recall.

More generally predict 1 if:  $f_{\vec{w},b}(\vec{x}) \geq \text{threshold}$ .



الف) TP، TN، FP و FN مفاهیمی هستند که برای ارزیابی عملکرد مدل‌های طبقه‌بندی باینری استفاده می‌شوند:

۱. مثبت واقعی (TP): این زمانی اتفاق می‌افتد که مدل به درستی یک نتیجه مثبت را پیش‌بینی کند (به عنوان مثال، حضور یک گربه، انسان یا رویدادی مانند هک) در حالی که نتیجه واقعی واقعاً مثبت باشد.
۲. منفی واقعی (TN): این زمانی اتفاق می‌افتد که مدل به درستی یک نتیجه منفی را پیش‌بینی کند (مثلاً غیبت گربه، نبود انسان یا عدم فعالیت هک) در حالی که نتیجه واقعی واقعاً منفی است.
۳. مثبت کاذب (FP): این زمانی اتفاق می‌افتد که مدل یک نتیجه مثبت را پیش‌بینی می‌کند، اما نتیجه واقعی منفی است. به عبارت دیگر، به اشتباه وجود چیزی را نشان می‌دهد (به عنوان مثال، شناسایی اشتباه تصویر غیر گربه به عنوان یک گربه، شناسایی یک رویداد غیر هک به عنوان هک).
۴. منفی کاذب (FN): این زمانی اتفاق می‌افتد که مدل یک نتیجه منفی را پیش‌بینی می‌کند، اما نتیجه واقعی مثبت است. نمی‌تواند چیزی را که وجود دارد شناسایی یا تشخیص دهد (به عنوان مثال، شناسایی نشدن گربه در تصویری که در واقع حاوی یک گربه است، ناتوانی در شناسایی فعالیت هک زمانی که وجود دارد).

این مفاهیم در درک عملکرد مدل‌های طبقه‌بندی اساسی هستند و اغلب برای محاسبه معیارهای ارزیابی مختلف مانند accuracy، precision، recall و F1-score استفاده می‌شوند.

(ب)

هنگام شناسایی مجرمان هک، به ویژه در شرایطی که تشخیص اشتباه افراد بی گناه به عنوان مجرم می تواند عواقب جدی داشته باشد، معیارهای ارزیابی زیر بسیار مهم هستند:

(به حداقل رساندن موارد مثبت کاذب (FP) برای جلوگیری از متهم کردن نادرست افراد بی گناه بسیار مهم است. اما ما همچنین می خواهیم منفی های کاذب (FN) را به حداقل برسانیم تا اطمینان حاصل کنیم که کشف هکرها واقعی را از دست نمی دهیم.)

۱. **Accuracy**: صحت کلی پیش بینی های مدل را اندازه گیری می کند که می تواند به صورت  $\frac{(TP + TN)}{(TP + TN + FP + FN)}$  محاسبه شود. با این حال، در مجموعه داده های نامتعادل (جایی که موارد مثبت یا منفی به طور قابل توجهی رایج تر هستند)، Accuracy ممکن است به تنهایی کافی نباشد.

۲. **Precision**: بر دقت پیش بینی های مثبت، محاسبه شده به صورت  $\frac{TP}{(TP + FP)}$  تمرکز می کند. precision، توانایی مدل را در شناسایی صحیح نکات مثبت واقعی از بین تمامی پیش بینی های مثبت ارزیابی می کند. بسیار مهم است که از متهم کردن نادرست افراد بی گناه خودداری کنید (پایین نگه داشتن FP).

۳. **Recall**: توانایی مدل را برای شناسایی صحیح تمام نمونه های مربوطه، محاسبه شده به صورت  $\frac{TP}{(TP + FN)}$  اندازه گیری می کند. برای اطمینان از اینکه مدل شناسایی رویدادهای هک واقعی را از دست نمی دهد (پایین نگه داشتن FN) ضروری است.

۴. **F1-score**: میانگین هارمونیک precision and recall به صورت  $\frac{2 * Precision * Recall}{Precision + Recall}$  است. امتیاز F1 هم precision و هم recall را در نظر می گیرد و بین آنها تعادل ایجاد می کند.

۵. **False Positive Rate (FPR)**: به صورت  $\frac{FP}{(FP + TN)}$  محاسبه می شود. در مواردی که متهم کردن افراد بی گناه عواقب شدیدی دارد، به حداقل رساندن میزان مثبت کاذب (FP) بسیار مهم است.

۶. **ملاحظات اخلاقی**: جدا از معیارهای ارزیابی سنتی، ملاحظات اخلاقی در درجه اول اهمیت قرار دارند. ارزیابی سوگیری، انصاف، و تأثیر اجتماعی احتمالی اتهامات نادرست باید به دقت مورد بررسی قرار گیرد. انصاف (Fairness) در میان گروه های مختلف جمعیتی باید برای جلوگیری از تبعیض ارزیابی شود.

به طور خلاصه، ترکیبی از accuracy، precision، recall، F1-score، false positive rate و ملاحظات اخلاقی باید معیارهای ارزیابی را برای شناسایی مجرمان هک اسنپ فود، حصول اطمینان از رویکرد متعادلی که اتهامات نادرست را به حداقل می رساند و در عین حال به طور موثر فعالیت های هک واقعی را شناسایی می کند، راهنمایی کند. (به یاد داشته باشید که هیچ معیار کاملی برای این کار وجود ندارد. معیارهای ارزیابی ایده آل به زمینه، ویژگی های مجموعه داده و اولویت های خاص موقعیت بستگی دارد.)

مراجع:

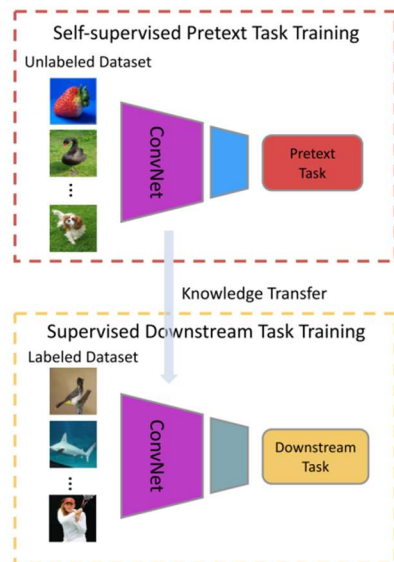
<https://chat.openai.com/>

<https://bard.google.com/>

<https://claude.ai/chats>



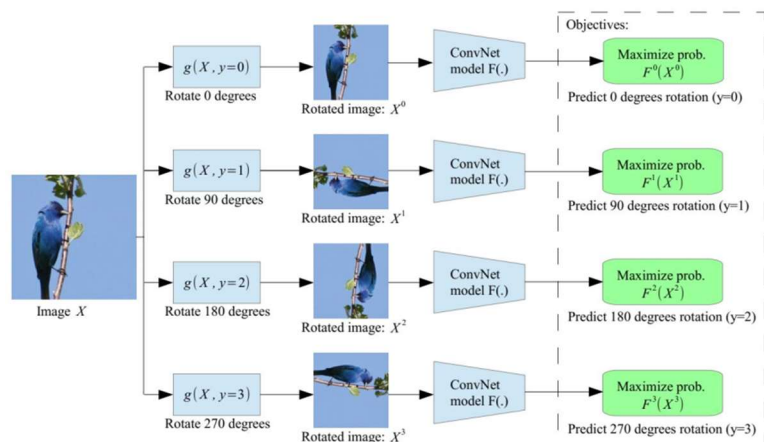
## یادگیری خودنظارتی (Self-Supervised)



- روش‌های یادگیری خود نظارتی برای یادگیری ویژگی‌های عمومی از داده‌های بدون برچسب در مقیاس بزرگ پیشنهاد شده‌اند
- مسئله‌های کمکی دو ویژگی مشترک دارند:
  - برای حل مسئله کمکی لازم است تا ویژگی‌های بصری توسط شبکه‌های کانولوشنی استخراج شوند
  - شبه‌برچسب‌های مورد نیاز در مسئله کمکی باید به صورت خودکار برای داده‌های مورد نظر قابل تولید باشند
- این رویکرد برای مسئله‌های غیر از بینایی کامپیوتر مانند پردازش صوت و پردازش متن نیز قابل استفاده است

## مثال: تخمین چرخش

- با آموزش مدل برای تشخیص میزان چرخش ۲ بعدی، ویژگی‌های تصویر آموخته می‌شود



تخمین چرخش وظیفه‌ای است که شامل پیش‌بینی زاویه چرخش یک تصویر است. تخمین چرخش می‌تواند برای وظیفه طبقه‌بندی مفید باشد، به ویژه در سناریوهایی که جهت یا زاویه یک شی ممکن است در مجموعه داده متفاوت باشد. با تخمین و اصلاح چرخش‌ها از قبل، می‌توان به روش‌های زیر به بهبود عملکرد مدل طبقه‌بندی کمک کرد:

- یک روش یادگیری خود نظارتی (self-supervised): تخمین چرخش می‌تواند به عنوان یک روش یادگیری خود نظارتی برای یادگیری ویژگی‌های تصویر بدون استفاده از هیچ‌گونه حاشیه نویسی انسانی استفاده شود. با آموزش شبکه برای تشخیص چرخش یک تصویر، شبکه می‌تواند یاد بگیرد که ویژگی‌های معنایی و ثابت را از تصویر استخراج کند که می‌تواند برای کارهای پایین دستی مانند طبقه‌بندی مفید باشد.

- یادگیری بازنمایی (Representation learning): تخمین چرخش مدل را مجبور می کند تا ویژگی های بصری قوی را بیاموزد که بدون توجه به جهت گیری (orientation) باقی می ماند. یادگیری بازنمایی به خوبی به وظایف پایین دستی منتقل می شود.
- یادگیری ویژگی بهبودیافته (Improved Feature Learning): با آموزش مدلی برای پیش بینی یا تخمین چرخش یک شی، شبکه ممکن است ویژگی های قوی تر و انتزاعی تر را بیاموزد که نسبت به چرخش ثابت هستند. سپس این ویژگی ها می توانند برای کار طبقه بندی اصلی متمایزتر و آموزنده تر باشند. (مدل در طول استنتاج نسبت به چرخش ها حساسیت کمتری دارد.)
- داده افزایی (Data Augmentation): چرخش تصاویر با درجات مختلف و استفاده از این نسخه های چرخانده شده به عنوان داده های آموزشی اضافی می تواند مجموعه داده را افزایش دهد. این تکنیک تقویت به مدل کمک می کند تا نسبت به چرخش ها قوی تر و ثابت تر شود و قابلیت تعمیم آن را بهبود بخشد.
- استخراج ویژگی (Feature extracting): تخمین چرخش می تواند به عنوان یک تکنیک استخراج ویژگی برای به دست آوردن نمایش های چرخشی ثابت از تصاویر برای یک کار طبقه بندی استفاده شود. با رمزگذاری اطلاعات چرخش تصاویر در یک بردار ویژگی، شبکه میتواند یاد بگیرد که تغییرات نامربوط تصاویر را نادیده بگیرد و بر ویژگیهای اساسی کلاسه ها تمرکز کند. در واقع، به عنوان یک وظیفه میانی (proxy) برای استخراج ویژگی های معنی دار عمل می کند. به مدل ها کمک می کند تا بازنمایی هایی را بیاموزند که روابط فضایی و ساختار شی را به تصویر می کشند.
- تعمیم بهتر (Enhanced Generalization): آموزش یک مدل برای درک و تنظیم چرخش ها می تواند توانایی آن را در تشخیص اشیا بدون توجه به جهت آنها بهبود بخشد. این می تواند منجر به تعمیم بهتر به نمونه های دیده نشده یا چرخیده شده در طول مرحله آزمایش شود.
- قدرت بیشتر در مواجهه با داده های واقعی: این روش مدل را برای مدیریت بهتر سناریوهای داده های واقعی که در آن اشیا در زوایای مختلف ظاهر میشوند آماده کند، بنابراین robustness و سازگاری مدل را با محیط های متنوع و متغیر بهتر میکند.

به طور کلی، تخمین چرخش می تواند با ارائه داده های آموزشی اضافی و تشویق یادگیری ویژگی های پی تغییر و آموزنده تر (invariant and informative features)، به ایجاد یک مدل قوی تر و تعمیم یافته تر کمک کند.

ب) رمزگذاری one-hot یک روش نمایشی است که برای داده های طبقه بندی استفاده می شود، که در آن هر دسته با یک بردار باینری نمایش داده می شود. در یک one-hot vector، هر دسته به طور منحصربه فرد با یک بردار نشان داده می شود که در آن تنها یک عنصر ۱ (hot) و بقیه ۰ (cold) هستند. ابعاد برابر است با تعداد دسته ها.

به عنوان مثال، در نظر گرفتن دسته بندی های "قرمز"، "آبی" و "سبز":

- "قرمز" ممکن است به صورت  $[1, 0, 0]$  نشان داده شود،
- "آبی" به صورت  $[0, 1, 0]$  و
- "سبز" به عنوان  $[0, 0, 1]$ .

مشکل استفاده از one-hot vectors:

بردارهای one-hot می توانند به نمایش های پربعد و پراکنده (high-dimensional and sparse representations) منجر شوند، به ویژه در سناریوهایی با تعداد دسته های زیاد. این می تواند چالش هایی مانند:

- افزایش ابعاد (Increased Dimensionality): رمزگذاری one-hot فضای ویژگی را به طور قابل توجهی گسترش می دهد، به ویژه هنگامی که با متغیرهای دسته بندی متعدد سروکار داریم. این می تواند منجر به ناکارآمدی محاسباتی و نفرین ابعاد (curse of dimensionality) شود. (به عنوان مثال ابعاد بالا برای مجموعه واژگان بزرگ در جاسازی کلمات)
- فقدان تشابه معنایی (Lack of Semantic Similarity): بردارهای one-hot ذاتاً روابط معنایی بین دسته ها را در بر نمی گیرند. همه دسته ها به صورت مستقل در نظر گرفته می شوند، که ممکن است روابط یا شباهت های اساسی آنها را منعکس نکنند. (همه بردارها متعامد هستند)
- پراکندگی (Sparsity): اکثر عناصر بردار صفر هستند که منجر به ذخیره سازی و محاسبات ناکارآمد (inefficient storage and computation) می شود.

(ج)

Word2Vec روشی است که برای یادگیری نمایش های توزیع شده کلمات (distributed representations of words) در یک فضای برداری پیوسته استفاده می شود. با استفاده از context کلمات در یک مجموعه بزرگ برای یادگیری جاسازی کلمات (word embeddings) به شیوه ای بدون نظارت انسان عمل می کند.

Word2Vec با تعریف یادگیری خود نظارت مطابقت دارد، زیرا از ساختار یا الگوهای ذاتی درون خود داده ها (مجموعه متن (text corpus) در این مورد) بدون نیاز به نظارت آشکار با برچسب انسانی استفاده می کند. در عوض، از context ای استفاده می کند که در آن کلمات به نظر می رسند که بازنمایی ها را یاد بگیرند. به طور خاص، از مدل های Continuous Bag-of-Words (CBOW) یا Skip-gram برای پیش بینی کلمات اطراف با یک کلمه هدف (یا برعکس) استفاده می کند.

با یادگیری پیش بینی کلمات نزدیک در یک جمله یا زمینه، Word2Vec به طور موثر جاسازی های کلمه ای را یاد می گیرد که روابط معنایی بین کلمات را به تصویر می کشد. این فرآیند به داده های برچسب گذاری شده خارجی متکی نیست، بلکه از ساختار ذاتی و الگوهای توزیعی موجود در مجموعه متن (text corpus) استفاده می کند که نیازی به نظارت و برچسب گذاری توسط انسان ندارد و آن را به یک تکنیک یادگیری خود نظارتی تبدیل می کند.

توضیح این سوال به بیان ساده تر و خلاصه تر از زبان bard:

- Self-Supervised Learning:
  - Algorithms learn from data without explicit labels.
  - Create their own supervisory signals from the data itself.
- Word2Vec:
  - Predicts context words given a target word (skip-gram) or vice versa (CBOW).
  - Uses unlabeled text as a pretext task to learn word embeddings.
  - Captures semantic relationships between words in a dense vector space.
- Self-Supervision Aspects:
  - No external labels for word relationships.
  - Creates its own learning signal by predicting context words.
  - Leverages natural language structure for representation learning.

مراجع:

<https://chat.openai.com/>

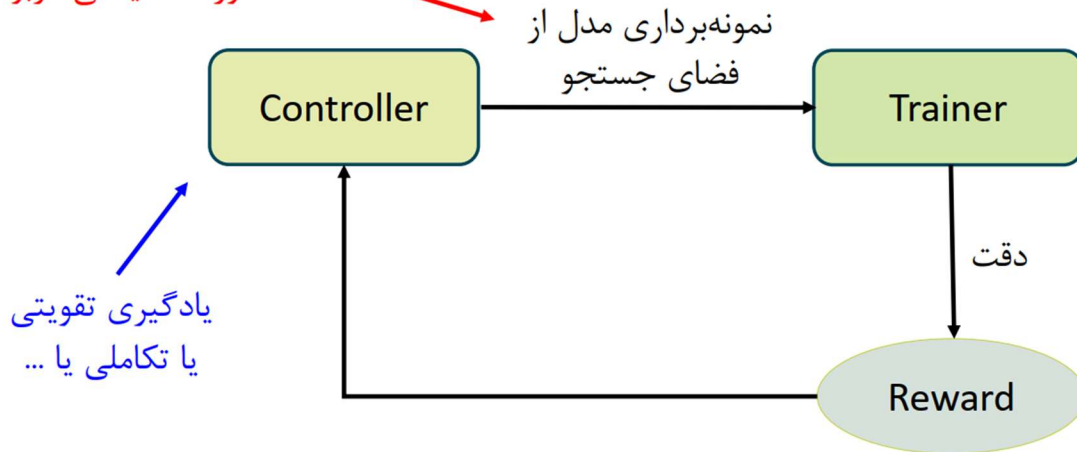
<https://bard.google.com/>

<https://claude.ai/chats>

<https://code.google.com/archive/p/word2vec/>

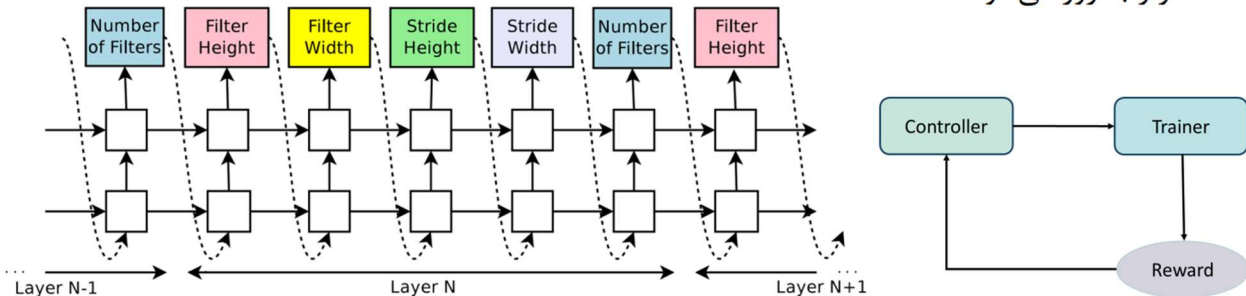
## جستجوی معماری عصبی (Neural Architecture Search)

بر اساس اصول اولیه در  
حوزه تحقیقاتی مربوطه



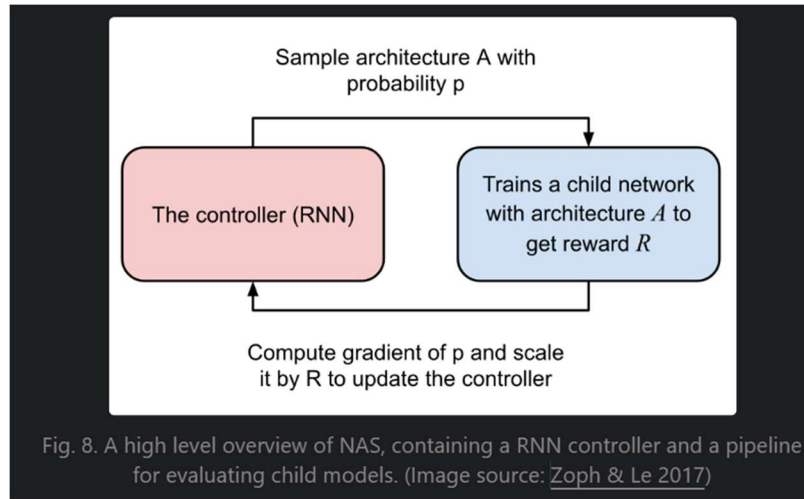
## جستجوی معماری عصبی (Neural Architecture Search)

- در کنترلر، از یک شبکه بازگشتی برای تولید ابرپارامترهای شبکه کانولوشنی استفاده می‌شود
- تعداد لایه‌های شبکه در هر مرحله ثابت فرض می‌شود که در طول زمان افزایش می‌یابد
- پس از همگرایی شبکه کانولوشنی، دقت آن بر روی مجموعه اعتبارسنجی محاسبه شده و پارامترهای کنترلر به روز می‌شوند



یادگیری تقویتی برای جستجوی معماری شبکه عصبی یا ابرپارامترهای شبکه:

یادگیری تقویتی (RL) می‌تواند برای جستجوی خودکار و بهینه سازی معماری شبکه عصبی یا ابرپارامترها استفاده شود. این فرآیند شامل عاملی است که یاد می‌گیرد در مورد چگونگی ساخت یا اصلاح معماری شبکه عصبی تصمیمات متوالی بگیرد. طراحی اولیه NAS (Zoph & Le 2017) شامل یک کنترل کننده مبتنی بر RL برای پیشنهاد معماری مدل کودک برای ارزیابی است. کنترل کننده به عنوان یک RNN پیاده سازی می‌شود و دنباله ای با طول متغیر از توکن های مورد استفاده برای پیکربندی یک معماری شبکه را خروجی می‌دهد.



کنترلر به عنوان یک تسک RL با استفاده از REINFORCE آموزش داده می شود.

فضای عمل (Action space): فضای عمل فهرستی از توکن ها برای تعریف شبکه فرزند پیش بینی شده توسط کنترلر است. کنترلر اقدام به خروجی عمل  $a_{1:T}$  می کند که در آن  $T$  تعداد کل توکن ها است.

پاداش (Reward): دقت شبکه کودک که می تواند در همگرایی به دست آید، پاداش برای آموزش کنترل کننده،  $R$ ، است.

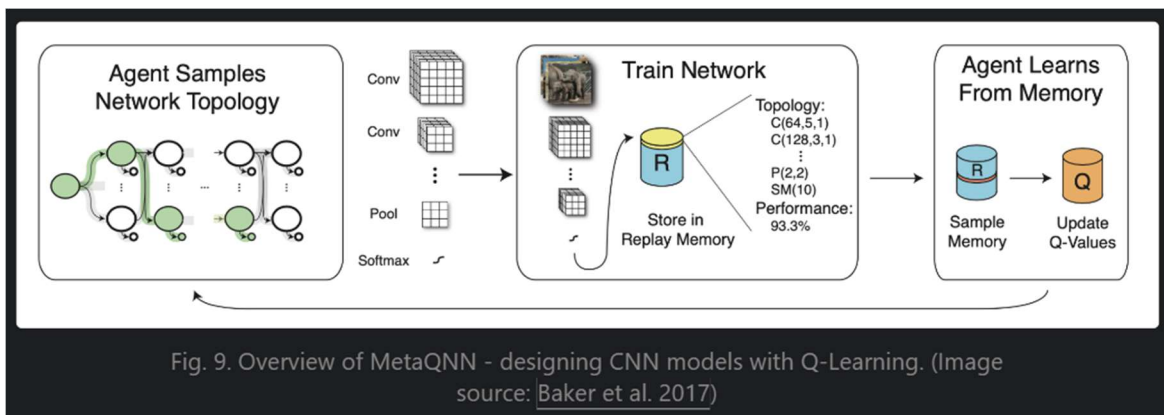
ضرر (Loss): NAS پارامترهای کنترل کننده  $\theta$  را با ضرر REINFORCE بهینه می کند. ما می خواهیم پاداش مورد انتظار (دقت بالا) را با گرادینان به صورت زیر به حداکثر برسانیم. نکته خوب در اینجا با گرادینان سیاست این است که حتی زمانی که پاداش غیر مشتق پذیر باشد، کار می کند.

$$\nabla_{\theta} J(\theta) = \sum_{t=1}^T \mathbb{E}[\nabla_{\theta} \log P(a_t | a_{1:t-1}; \theta) R]$$

به عنوان مثال، MetaQNN (Baker et al. 2017) یک عامل را آموزش می دهد تا به طور متوالی لایه های CNN را با استفاده از یادگیری Q با استراتژی اکتشاف  $\epsilon$ -greedy و بازپخش تجربه (experience replay) انتخاب کند. پاداش نیز دقت اعتبارسنجی است.

$$Q^{(t+1)}(s_t, a_t) = (1 - \alpha)Q^{(t)}(s_t, a_t) + \alpha(R_t + \gamma \max_{a \in \mathcal{A}} Q^{(t)}(s_{t+1}, a'))$$

که در آن یک حالت  $s_t$  دوتایی عملیات لایه و پارامترهای مرتبط است. یک عمل  $a$  ارتباط بین عملیات ها را تعیین می کند. Q-value متناسب با میزان اطمینان ما در دو عملیات متصل است که منجر به دقت بالا می شود.



به طور خلاصه مراحل کلیدی به شرح زیر است:

۱. کنترل کننده (Controller): یک مدل، اغلب یک شبکه عصبی بازگشتی (RNN)، وظیفه تولید تصمیمات معماری شبکه مانند انواع لایه، اتصالات، و فرایامترها را دارد.
۲. تولید پیشنهاد (Proposal Generation): کنترل کننده یک معماری شبکه (شبکه کودک) را بر اساس سیاست فعلی خود پیشنهاد می کند.
۳. آموزش شبکه کودک (Child Network Training): معماری پیشنهادی در مورد وظیفه هدف (به عنوان مثال، تشخیص شی) برای ارزیابی عملکرد child network آموزش داده می شود.
۴. ارزیابی پاداش (Reward Assessment): عملکرد شبکه کودک آموزش دیده با استفاده از یک تابع پاداش (به عنوان مثال، دقت یا ضرر در مجموعه اعتبارسنجی) ارزیابی می شود.
۵. به روز رسانی سیاست (Policy Update): کنترل کننده پاداش را دریافت می کند و سیاست خود را به روز می کند تا معماری های بهتری در آینده ایجاد کند، با هدف به حداکثر رساندن پاداش مورد انتظار (maximize the expected reward).
۶. تکرار (Iteration): مراحل ۲-۵ تکرار می شوند و کنترل کننده را به کاوش و کشف معماری های بهتر و امیدوارکننده تر از طریق آزمون و خطا هدایت می شود.

عامل (کنترل کننده) به طور مکرر اقداماتی را برای به حداکثر رساندن پاداش بلند مدت انجام می دهد. اقداماتی که عملکرد روی داده های اعتبارسنجی را بهبود می بخشد و پاداش مثبتی دریافت می کنند. همچنین اعمال بد، پاداش منفی دارد. عامل (کنترل کننده) یک سیاست بهینه را برای کاوش کارآمد فضای جستجو می آموزد.

(ب)

رویکرد یادگیری تقویتی (RL) برای جستجو و بهینه سازی فرایامترهایی مانند اندازه تصویر ورودی و تعداد لایه ها در معماری شبکه های عصبی برای تشخیص اشیا مناسب است و امکان استفاده دارد:

امکان پذیری (Feasibility):

• اندازه تصویر ورودی:

- RL می تواند به طور موثر اندازه های تصویر ورودی بهینه را جستجو کند.
- تابع پاداش مستقیماً دقت یا سایر معیارهای عملکرد مانند ضرر را در وظایف تشخیص اشیا به کنترل کننده (agent) منعکس می کند.
- کنترل کننده بواسطه پاداش ها یاد می گیرد که معماری هایی با اندازه ورودی ای را پیشنهاد کند که بالاترین پاداش (دقت در مجموعه اعتبار سنجی) را به همراه دارد.
- فضای جستجوی حالت گسسته است (32x32, 64x64, 128x128, and etc.). اقدامات (actions) برای افزایش/کاهش اندازه تصویر ورودی تأثیری smooth بر عملکرد اعتبار سنجی دارند.
- RL می تواند با بررسی رزولوشن ها و نسبت های مختلف به عنوان اقدامات (actions) و دریافت پاداش (reward) بر اساس عملکرد مدل، به تعیین اندازه بهینه تصویر کمک کند. عامل می تواند اندازه تصویری را یاد بگیرد که دقت را در حین در نظر گرفتن محدودیت های محاسباتی به حداکثر می رساند.

• تعداد لایه ها:

- RL همچنین می تواند برای جستجوی تعداد بهینه لایه ها استفاده شود.
- کنترل کننده معماری هایی برای شبکه کودک با عمق های متفاوت ایجاد می کند و تابع پاداش آن را به سمت مؤثرترین تعداد لایه ها (عمق) برای وظیفه معین (در اینجا تشخیص اشیا) هدایت می کند.
- فضای جستجوی حالت اعداد صحیح گسسته است. اقدامات (actions) یا همان افزودن/حذف لایه ها به طور تدریجی (smooth) بر پاداش (دقت اعتبارسنجی) تأثیر می گذارد.
- همچنین RL می تواند پیکربندی های لایه های مختلف، مانند عمق های مختلف، انواع لایه ها، و رد شدن از اتصالات را بررسی کند، در حالی که بر اساس دقت اعتبارسنجی یا سایر معیارهای عملکرد، پاداش دریافت می کند. عامل تعداد لایه ها و معماری بهینه را می آموزد که عملکرد و پیچیدگی مدل را متعادل می کند.



فضاهای حالت همبسته و هموار (smooth, correlated state spaces)، استفاده از RL را برای یادگیری سیاست هایی که فراپارامترها را برای بهبود پاداش تطبیق می دهند، امکان پذیر (feasible) می کند.

ملاحظات:

- هزینه محاسباتی (Computational Cost): NAS مبتنی بر RL می تواند از نظر محاسباتی گران باشد، زیرا شامل آموزش شبکه های کوچک متعدد است.
- تعریف فضای جستجو (Search Space Definition): فضای جستجو باید با دقت طراحی شود تا طیف متنوعی از معماری ها را در بر گیرد و در عین حال امکان پذیری را حفظ کند.
- طراحی تابع پاداش (Reward Function Design): تابع پاداش باید دقیقاً اهداف عملکرد مورد نظر را منعکس کند و جستجو را به سمت معماری های مؤثر هدایت کند.

فواید:

- جستجوی خودکار (Automated Search): NAS مبتنی بر RL متخصصان انسانی را از طراحی معماری دستی رها می کند.
- پتانسیل برای معماری های بدیع (Potential for Novel Architectures): می تواند معماری هایی را کشف کند که فراتر از الگوهای طراحی شده توسط انسان است.
- بهینه سازی ویژه وظایف (Task-Specific Optimization): می تواند معماری ها را برای وظایف و مجموعه داده های خاص تنظیم کند.

به طور کلی، RL یک رویکرد امیدوارکننده برای خودکار کردن جستجوی معماری عصبی و بهینه سازی فراپارامتر در تشخیص شی و حوزه های مختلف یادگیری عمیق دیگر ارائه می کند.

جواب سوال به بیان خلاصه از زبان chat gpt:

Neural architecture search (NAS) can be used to optimize hyperparameters such as the input image size and the number of layers in object recognition problems. NAS can be used to automatically search for the optimal architecture of a neural network that maximizes the accuracy of the model on a validation set. The optimal architecture can be found by training a large number of candidate architectures and selecting the one that performs the best. The input image size and the number of layers are important hyperparameters that can be optimized using NAS. The input image size affects the receptive field of the network and the number of layers affects the depth of the network. A larger receptive field can capture more contextual information in the image, while a deeper network can learn more complex features. However, increasing the input image size and the number of layers can also increase the complexity and computational cost of training the network. NAS can be used to find the optimal balance between accuracy and computational cost by searching for the optimal architecture of the network.

مراجع:

<https://chat.openai.com/>

<https://bard.google.com/>

<https://claude.ai/chats>

[\[1906.04423\] NAS-FCOS: Fast Neural Architecture Search for Object Detection \(arxiv.org\)](#)

[\[1611.01578\] Neural Architecture Search with Reinforcement Learning \(arxiv.org\)](#)

<https://lilianweng.github.io/posts/2020-08-06-nas/>

## سوال پنجم

لزوماً نباید انتظار داشته باشید که آنها یکسان باشند زیرا این ضررها در طول زمان به مدل های با کیفیت متفاوت مربوط می شود. یعنی ضرر مولد در دوره های ۱ و ۱۰۰ با توجه به یک ممیز است که ممکن است به طور قابل توجهی بهبود یافته باشد، و همین امر برای ضرر ممیز نیز ادامه دارد.

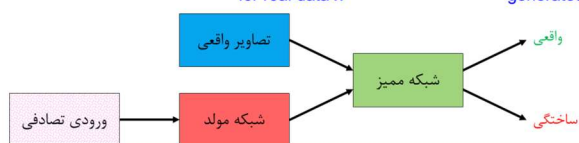


## GAN

- شبکه ممیز (با پارامترهای  $\theta_d$ ) می خواهد تابع هدف را بیشینه کند به گونه ای که  $D(x)$  برای داده های واقعی نزدیک به ۱ و  $D(G(z))$  برای داده های ساختگی نزدیک به صفر باشد
- شبکه مولد (با پارامترهای  $\theta_g$ ) می خواهد تابع هدف را کمینه کند تا  $D(G(z))$  نزدیک به ۱ باشد و شبکه

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

Discriminator output for real data x
Discriminator output for generated fake data G(z)



جواب این سوال را در نمونه سوال های standford قبلا خوانده بودم:

(1 point) You are training a standard GAN, and at the end of the first epoch you take note of the values of the generator and discriminator losses. At the end of epoch 100, the values of the loss functions are approximately the same as they were at the end of the first epoch. Why are the quality of generated images at epoch 1 and epoch 100 not necessarily similar? (1-2 sentences)

**Solution:** You should not necessarily expect them to be the same since the losses are with respect to different quality models over time. That is, the loss of the generator at epochs 1 and 100 are with respect to a discriminator which might have significantly improved, and the same follows for the loss of the discriminator.

حال به بیان خودم به جواب سوال می پردازم:

با توجه به اینکه شبکه مولد می خواهد ضرر را مینیمم کند و شبکه ممیز می خواهد ضرر را ماکسیمم کند، یک فضای رقابتی داریم و ضرر به قدرت هر دو شبکه وابسته است. در طول آموزش و با گذر زمان هر دو شبکه قوی تر و با کیفیت تر می شوند. چون این دو شبکه با هم رقابت می کنند و ضرر به قدرت هر دو وابسته است چنین اتفاقی که در صورت سوال آمده است ممکن است روی دهد. در پایان epoch اول هر دو شبکه مولد و ممیز ضعیف هستند. در پایان epoch ۱۰۰ ام هر دو شبکه مولد و ممیز قوی هستند. چون هر دو شبکه در هر دو حالت قدرت مشابهی دارند لذا مقدار ضرر تقریباً یکسان می شود ولی همان طور که واضح و مبرهن است شبکه مولد در پایان epoch ۱۰۰ ام قدرت بیشتری دارد و کیفیت تصاویر تولید شده بهتر است در صورتی که ضرر آن تغییر چندانی نکرده است. دلیل آن هم به تعریف این تابع ضرر بر میگردد چرا که به گونه ای تعریف شده است که هر دو شبکه با هم رقابت کنند و قوی تر شوند نه اینکه صرفاً کیفیت عکس تولید شده از شبکه مولد را حساب کند.

همچنین در زیر تعریفی از این شبکه‌ها (GAN) و توابع ضرر آنها آمده است:

در شبکه‌های متخاصم مولد (GANs)، شبکه‌های مولد و متمایزکننده در حین آموزش بازی موش و گربه را انجام می‌دهند. هدف مولد تولید تصاویر واقعی است تا متمایزکننده را فریب دهد، در حالی که متمایزکننده سعی می‌کند بین تصاویر واقعی و تولید شده تمایز قائل شود. فرآیند آموزش شامل بهبود مکرر و تنظیم استراتژی‌های این شبکه‌ها می‌شود که می‌تواند منجر به تغییر در کیفیت تصویر تولید شده در طول دوره‌ها شود، حتی اگر ضرر مشابه به نظر برسد.

توابع ضرر مولد و ممیز مستقیماً با کیفیت تصاویر تولید شده توسط شبکه مولد ارتباطی ندارند. شبکه مولد یاد می‌گیرد که با به حداقل رساندن تفاوت بین تصاویر تولید شده و تصاویر واقعی (تابع هدف را کمینه کند)، تصاویر تولید کند، در حالی که شبکه تفکیک کننده (ممیز) یاد می‌گیرد بین تصاویر تولید شده و تصاویر واقعی تمایز قائل شود (تابع هدف را بیشینه کند).

کیفیت تصاویر تولید شده توسط شبکه مولد به پیچیدگی مجموعه داده، معماری شبکه‌های مولد و ممیز و فرآیندهای آموزش بستگی دارد. بنابراین، حتی اگر عملکردهای ضرر مولد و تفکیک کننده در پایان دوره اول و صدم تقریباً یکسان باشند، ممکن است کیفیت تصاویر تولید شده توسط شبکه مولد به دلیل عوامل ذکر شده یکسان نباشد.

جواب chat GPT نیز به بیان زیر بود:

دلایل متعددی در تفاوت کیفیت تصویر بین دوره اول و صدم علی‌رغم مقادیر مشابه ضرر مشارکت می‌کنند:  
۱. همگرایی در مقابل کیفیت (Convergence vs. Quality):

- مقادیر ضرر نشان دهنده همگرایی است، نه لزوماً کیفیت تصویر.
  - آنها میزان تعادل بین دو شبکه را اندازه‌گیری می‌کنند، نه کیفیت بصری تصاویر تولید شده را.
  - یک GAN می‌تواند در جایی که تصاویر تولید شده از نظر بصری قانع‌کننده نیستند، به راه حل‌های غیربهبوده همگرا شود.
۲. همگرایی ضرر به معنای همگرایی کامل مدل نیست: ضرر می‌تواند به نقطه تعادل برسند اما شبکه‌های مولد و تمایز ممکن است به طور کامل همگرا نشده باشند. با آموزش بیشتر، کیفیت‌های آنها همچنان بهبود می‌یابد.

۳. Visual Perception vs. Loss Functions:

- توابع ضرر ممکن است کاملاً با ادراک بصری انسان هماهنگ نباشد.
  - تصاویر تولید شده ممکن است از نظر بصری بدون تغییرات قابل توجه در مقادیر ضرر بهبود یابد.
۴. GAN Mode Collapse: GAN ها مستعد فروپاشی (collapse) حالت هستند، که در آن مولد انواع محدودی از تصاویر را تولید می‌کند یا در تولید نمونه‌های مشابه گیر می‌کند. در دوره‌های اولیه، شبکه ممکن است در حالت فروپاشی باشد و تنها زیر مجموعه‌ای از تصاویر بالقوه تولید کند. با پیشرفت آموزش، مدل ممکن است از این فروپاشی فرار کند و نمونه‌های تولید شده را متنوع و بهبود بخشد.

۵. معیارهای کیفی در مقابل معیارهای کمی:

- مقادیر ضرر کمی هستند، اما کیفیت تصویر اغلب ذهنی و کیفی است.
  - ارزیابی انسانی و بازرسی بصری اغلب برای ارزیابی عملکرد GAN بسیار مهم است.
۶. رانش ژنراتور (Generator drift): توزیع مولد با پیشرفت آموزش به سمت توزیع داده می‌رود و به آن اجازه می‌دهد تصاویر واقعی‌تری تولید کند. اما از دیدگاه تفکیک کننده، توزیع داده‌ها و مولد متعادل می‌مانند، بنابراین ضرر بدون تغییر است.
۷. مرزهای حداقل (Minima boundaries): تابع تلفات ممکن است حوضه‌های حداقلی وسیعی داشته باشد. ژنراتور می‌تواند به بهبود کیفیت در کمترین حد ممکن بدون تغییر ضرر ادامه دهد.

۸. قدرت شبکه ممیز:

- یک ممیز ضعیف تر ممکن است به مولد اجازه دهد که در اوایل تصاویر به ظاهر خوب تولید کند، اما این تصاویر ممکن است در ادامه آموزش قابل قبول نباشند.
- یک ممیز قوی تر می تواند مولد را برای تولید تصاویر واقعی تر در طول زمان سوق دهد.
- به دلیل قدرت مشابه در هر دو حالت ضرر میتواند تقریباً یکسان باشد.

۹. عدم حساسیت ضرر به ادراک بصری: ضرر GAN می تواند به جنبه های خاصی از کیفیت تصویر که از نظر ادراکی مهم هستند غیر حساس باشد. مولد می تواند در این ابعاد پیشرفت کند بدون اینکه تأثیر قابل توجهی بر ضرر داشته باشد.

۱۰. Stochasticity:

- GAN ها شامل فرایندی تصادفی در نمونه گیری و آموزش هستند.
- حتی با مقادیر تلفات ضرر، تصاویر تولید شده می توانند به دلیل عوامل تصادفی کیفیت متفاوتی داشته باشند.

۱۱. یادگیری غیر خطی:

- آموزش GAN اغلب بسیار غیرخطی است.
- پیشرفت در کیفیت تصویر ممکن است مستقیماً با مقادیر تلفات متناسب نباشد.
- بهبودهای قابل توجهی در کیفیت بصری ممکن است حتی با تغییرات کوچک در کاهش رخ دهد.

۱۲. دینامیک یادگیری: توابع ضرر یک نشانه کلی از چگونگی یادگیری شبکه ها ارائه می دهند، اما ممکن است پیچیدگی توزیع تصویر را نشان ندهند. حتی با ضرر مشابه، شبکه ها ممکن است در حال یادگیری جنبه ها یا تفاوت های ظریف تولید تصویر باشند. دوره های اولیه ممکن است بر ویژگی های اساسی تمرکز کنند، در حالی که دوره های بعدی ممکن است الگوها و جزئیات پیچیده تری را ثبت کنند و کیفیت تصویر را افزایش دهند.

بنابراین به طور خلاصه، کیفیت تولید تصویر GAN به همگرایی کامل مدل بستگی دارد، نه فقط همگرایی ضرر. ضرر یک پروکسی ناقص برای اندازه گیری کیفیت تولید هستند. تکرارهای آموزشی بیشتر اغلب منجر به بهبودهای کیفی می شود که در مقادیر ضرر منعکس نمی شود.

بنابراین تنها تکیه بر مقادیر ضرر برای ارزیابی عملکرد GAN می تواند گمراه کننده باشد. ارزیابی بصری و معیارهای کیفی برای درک کیفیت واقعی تصاویر تولید شده و شناسایی مناطق برای بهبود ضروری هستند.

مراجع:

<https://chat.openai.com/>

<https://bard.google.com/>

<https://claude.ai/chats>

<https://cs230.stanford.edu/files/>

## پایان