معماری GoogleNet که با نام Inception نیز شناخته می‌شود، یکی از معماری‌های پیشرفته شبکه‌های عصبی کانولوشن (CNN) است که توسط تیم تحقیقاتی گوگل در سال 2014 ارائه شد. این معماری در مقاله‌ای به نام "Going Deeper with Convolutions" معرفی شده و توانست در رقابت ImageNet Large Scale Visual Recognition Challenge (ILSVRC) 2014 برنده شود.

مفاهیم و اجزای اصلی GoogleNet

1. هدف و انگیزه

یکی از اهداف اصلی در طراحی GoogleNet کاهش تعداد پارامترهای شبکه بود، بدون کاهش دقت. شبکه‌های بزرگتر و عمیق‌تر معمولاً دارای پارامترهای بسیاری هستند که به مقدار زیادی حافظه و قدرت محاسباتی نیاز دارند. GoogleNet با معرفی ماژول‌های Inception این مشکل را حل کرد.

2. ماژول Inception

ماژول Inception یکی از اجزای کلیدی در معماری GoogleNet است که هدف آن افزایش کارایی شبکه‌های عصبی کانولوشنی (CNN) از طریق استفاده بهینه از محاسبات و کاهش تعداد پارامترها است. این ماژول به شبکه اجازه می‌دهد به طور همزمان ویژگی‌های مختلف را با فیلترهای متنوع استخراج کند. در ادامه، به تشریح جزئیات و ساختار ماژول Inception می‌پردازیم.

اجزای ماژول Inception

ماژول Inception از چندین لایه کانولوشن با اندازه‌های مختلف و یک لایه pooling تشکیل شده است که به صورت موازی اعمال می‌شوند. سپس نتایج این لایه‌ها به هم متصل می‌شوند تا خروجی نهایی ماژول را تشکیل دهند.

1. Convolution 1x1

وظیفه: کاهش تعداد کانال‌های ورودی و کاهش ابعاد ویژگی‌ها.

مزیت: کاهش هزینه محاسباتی و بهینه‌سازی استفاده از منابع.

2. Convolution 3x3

وظیفه: استخراج ویژگی‌های محلی و متوسط.

مزیت: شناسایی الگوهای کوچکتر در تصویر.

3. Convolution 5x5

وظیفه: استخراج ویژگی‌های بزرگتر و گسترده‌تر.

مزیت: شناسایی الگوهای بزرگتر و پیچیده‌تر.

4. Max Pooling 3x3

وظیفه: کاهش ابعاد ویژگی‌ها با حفظ اطلاعات مهم.

مزیت: تجمیع اطلاعات و کاهش تعداد پارامترها.

5. Convolution 1x1 پس از Max Pooling

وظیفه: کاهش ابعاد بعد از pooling و ترکیب نتایج.

مزیت: بهینه‌سازی و کاهش هزینه محاسباتی.

ساختار ماژول Inception

ماژول Inception به گونه‌ای طراحی شده که چندین لایه با اندازه‌های مختلف به صورت موازی عمل می‌کنند و سپس خروجی آن‌ها به هم متصل می‌شود. این ساختار به شبکه اجازه می‌دهد تا ویژگی‌های مختلف را در سطوح مختلف تجزیه و تحلیل کند.

نمونه‌ای از یک ماژول Inception

فرض کنید ورودی ماژول Inception یک tensor با ابعاد 28 x28x192 باشد:

Branch 1:

Convolution1 x1: با 64 فیلتر

خروجی: 28x28x64

Branch 2:

Convolution1 x1: با 96 فیلتر

Convolution3 x3: با 128 فیلتر

خروجی: 28x28x128

Branch 3:

Convolution1 x1: با 16 فیلتر

Convolution5 x5: با 32 فیلتر

خروجی: 28x28x32

Branch 4:

Max Pooling3 x3

Convolution1 x1: با 32 فیلتر

خروجی: 28x28x32

اتصال خروجی‌ها

در نهایت، خروجی‌های هر چهار شاخه به هم متصل می‌شوند تا خروجی نهایی ماژول Inception را تشکیل دهند:

خروجی نهایی: 28x28x256(64+12832+32)

مزایای استفاده از ماژول Inception

1. کارایی بالا

ماژول Inception به شبکه اجازه می‌دهد ویژگی‌های مختلف را با فیلترهای متنوع و به صورت موازی استخراج کند، که این کارایی شبکه را بهبود می‌بخشد.

2. کاهش پارامترها

با استفاده از کانولوشن‌های x11، ماژول Inception به طور قابل توجهی تعداد پارامترها را کاهش می‌دهد، که این منجر به کاهش هزینه محاسباتی و نیاز به حافظه کمتر می‌شود.

3. انعطاف‌پذیری

ماژول Inception به شبکه اجازه می‌دهد تا در سطوح مختلف و با اندازه‌های مختلف ویژگی‌ها را تجزیه و تحلیل کند، که این منجر به بهبود دقت شبکه می‌شود.

4. جلوگیری از Overfitting

کاهش تعداد پارامترها به کاهش خطر overfitting کمک می‌کند، که این برای شبکه‌های عمیق بسیار مهم است.

ماژول Inception یکی از اجزای کلیدی و نوآورانه در معماری GoogleNet است که با بهینه‌سازی استفاده از منابع و کاهش پارامترها، دقت و کارایی شبکه‌های عصبی کانولوشنی را بهبود می‌بخشد. این ماژول با ترکیب فیلترهای مختلف و pooling به صورت موازی، به شبکه اجازه می‌دهد تا ویژگی‌های متنوعی را از تصاویر استخراج کند و به طور مؤثری در تشخیص و طبقه‌بندی تصاویر عمل کند.

3. معماری کلی GoogleNet

GoogleNet از ترکیب چندین ماژول Inception به همراه لایه‌های اولیه و لایه‌های نهایی تشکیل شده است. این معماری شامل 22 لایه قابل یادگیری است که نسبت به معماری‌های قبل از خود مانند AlexNet و VGGNet عمیق‌تر است، اما به دلیل استفاده از ماژول‌های Inception، تعداد پارامترهای کمتری دارد.

4. لایه‌های مهم در GoogleNet

L1: لایه ورودی: که شامل تصویر ورودی است.

L2-L3: لایه‌های کانولوشن اولیه: که ویژگی‌های پایه را استخراج می‌کنند.

L4-L9: ماژول‌های Inception اولیه: که ویژگی‌های اولیه را به ویژگی‌های پیچیده‌تر تبدیل می‌کنند.

L10: Max Pooling: برای کاهش ابعاد و تجمیع اطلاعات.

L11-L20: ماژول‌های Inception پیشرفته: که ویژگی‌های پیشرفته‌تری را استخراج می‌کنند.

L21: Average Pooling: برای کاهش ابعاد نهایی.

L22: Fully Connected Layer (Softmax): برای طبقه‌بندی نهایی.

5. استفاده از Auxiliary Classifiers

GoogleNet از دو طبقه‌بندی فرعی یا کمکی (auxiliary classifiers) استفاده می‌کند که در میانه شبکه قرار دارند و علاوه بر کمک به فرآیند آموزش و کاهش مشکل گرادیان ناپدید شونده (vanishing gradient problem)، به عنوان مکانیزمی برای منظم‌سازی عمل می‌کنند. این طبقه‌بندی‌ها به شبکه کمک می‌کنند تا یادگیری را در لایه‌های میانی بهبود بخشند.

مزایا و معایب GoogleNet

مزایا

کاهش تعداد پارامترها: به دلیل استفاده از ماژول‌های Inception و کاهش ابعاد با استفاده از convolutions 1x1.

افزایش دقت: استفاده از فیلترهای مختلف در هر ماژول Inception باعث می‌شود شبکه ویژگی‌های مختلف را با دقت بیشتری استخراج کند.

بهبود کارایی محاسباتی: با کاهش پارامترها و استفاده مؤثر از منابع محاسباتی.

معایب

پیچیدگی معماری: طراحی و پیاده‌سازی ماژول‌های Inception نیازمند دقت و زمان بیشتری است.

نیاز به تنظیمات دقیق: برای بهره‌گیری کامل از مزایای این معماری، نیاز به تنظیمات دقیق و بهینه‌سازی است.

GoogleNet با معرفی ماژول‌های Inception توانست به طور مؤثری تعداد پارامترهای شبکه را کاهش داده و در عین حال دقت شبکه را افزایش دهد. این معماری یکی از قدم‌های مهم در توسعه شبکه‌های عصبی کانولوشنی بود و راه را برای مدل‌های پیشرفته‌تر مانند Inception-V3 و Inception-ResNet باز کرد.

Inception Blocks:

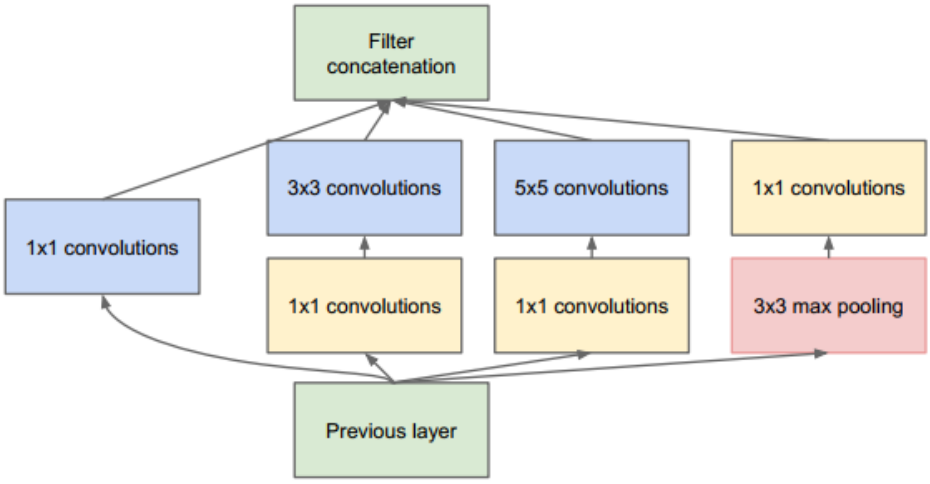


Figure 3: GoogLeNet network with all the bells and whistles

مفاهیم Translation Invariance و Translation Equivariance در شبکه‌های عصبی کانولوشنی (CNN) از اهمیت بالایی برخوردارند، زیرا این مفاهیم به توانایی مدل در تشخیص اشیاء در تصاویر بدون توجه به مکان آنها مربوط می‌شوند. در ادامه به توضیح این مفاهیم و نحوه ارتباط آنها با لایه‌های CNN می‌پردازیم.

Translation Equivariance

تعریف

Translation Equivariance به این معناست که اگر ورودی یک شبکه عصبی کانولوشنی جابجا شود (ترجمه شود)، خروجی لایه کانولوشنی نیز به همان اندازه جابجا خواهد شد. به عبارت دیگر، عملیات کانولوشنی ویژگی‌های تصویر را حفظ کرده و فقط موقعیت آنها را تغییر می‌دهد.

فرمول

اگر f یک تصویر باشد و T_k یک عملگر ترجمه که تصویر را به اندازه k واحد جابجا می‌کند، و g یک لایه کانولوشن باشد، آنگاه:

$$g(T_k f) = T_k (g(f))$$

ارتباط با لایه‌های کانولوشن

لایه‌های کانولوشن به طور ذاتی translation equivariant هستند. این بدان معناست که اگر ورودی به یک لایه کانولوشن جابجا شود، خروجی نیز به همان اندازه جابجا خواهد شد. این ویژگی به CNN ها کمک می‌کند تا ویژگی‌های محلی را بدون توجه به مکان آنها در تصویر استخراج کنند.

Translation Invariance

تعریف

Translation Invariance به این معناست که شبکه عصبی قادر باشد یک شیء را شناسایی کند، حتی اگر مکان آن در تصویر تغییر کند. به عبارت دیگر، خروجی نهایی شبکه نسبت به جابجایی‌های ورودی تغییر نمی‌کند.

فرمول

اگر f یک تصویر باشد و T_k یک عملگر ترجمه که تصویر را به اندازه k واحد جابجا می‌کند، و h تابع نهایی شبکه عصبی باشد، آنگاه:

$$h(T_k f) = h(f)$$

ارتباط با لایه‌های pooling و Fully Connected

لایه‌های pooling (معمولاً max pooling یا average pooling) و لایه‌های Fully Connected (FC) نقش مهمی در ایجاد translation invariance دارند.

لایه‌های Pooling:

- Max Pooling: با انتخاب حداکثر مقدار در هر ناحیه، ویژگی‌های مهم را حفظ می‌کند و حساسیت به جابجایی‌های کوچک را کاهش می‌دهد.
- Average Pooling: با محاسبه میانگین مقادیر در هر ناحیه، ویژگی‌ها را به طور عمومی‌تر حفظ کرده و حساسیت به جابجایی‌های کوچک را کاهش می‌دهد.

لایه‌های pooling با کاهش ابعاد ویژگی‌ها و تجمیع اطلاعات از نواحی مختلف، به شبکه کمک می‌کنند تا به translation invariance برسد. به عنوان مثال، اگر یک ویژگی در یک موقعیت خاص مهم باشد، pooling آن ویژگی را حفظ می‌کند حتی اگر مکان دقیق آن تغییر کند.

لایه‌های Fully Connected (FC):

- لایه‌های FC که در انتهای شبکه قرار دارند، نقش مهمی در ترکیب ویژگی‌های استخراج شده و ایجاد تصمیم نهایی ایفا می‌کنند. این لایه‌ها به دلیل ترکیب تمام ویژگی‌ها و نواحی مختلف، به شبکه کمک می‌کنند تا به translation invariance برسد.

جمع بندی نهایی:

Translation Equivariance:

- تعریف: جابجایی ورودی باعث جابجایی خروجی به همان اندازه می‌شود.

- لایه‌های مرتبط: لایه‌های کانولوشن.

- نحوه عملکرد: لایه‌های کانولوشن ویژگی‌ها را به صورت محلی حفظ کرده و آنها را به همان اندازه جابجا می‌کنند.

Translation Invariance:

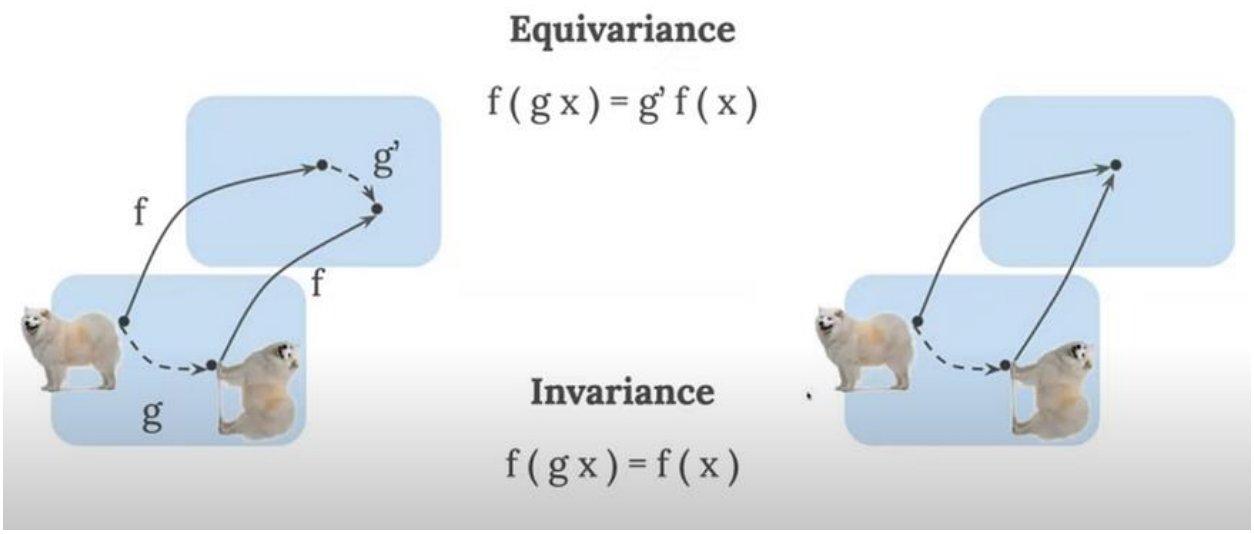
- تعریف: جابجایی ورودی تاثیری بر نتیجه نهایی ندارد.

- لایه‌های مرتبط: لایه‌های pooling (max/average) و لایه‌های Fully Connected.

- نحوه عملکرد: لایه‌های pooling ویژگی‌ها را از نواحی مختلف تجمیع می‌کنند و لایه‌های FC تصمیم نهایی را با ترکیب تمام ویژگی‌ها می‌گیرند.

اهمیت در CNN:

این دو مفهوم در CNN ها بسیار مهم هستند، زیرا به مدل‌ها کمک می‌کنند تا ویژگی‌های مهم را شناسایی کرده و اشیاء را بدون توجه به مکان آنها در تصویر شناسایی کنند. با ترکیب translation equivariance در لایه‌های کانولوشن و translation invariance در لایه‌های pooling و FC، شبکه‌های عصبی کانولوشنی می‌توانند به طور مؤثر تصاویر را پردازش و اشیاء را شناسایی کنند.


Equivariance
f (g x) = g' f (x)
f
g'
f
g
Invariance
f (g x) = f (x)