

$$\left\{ \begin{array}{l} L(w) = \|y - xw\|_2^2 + \|\Gamma w\|_2^2 \\ \text{with Dropout : } L(\hat{w}) = \mathbb{E}_{D \sim \text{Bernoulli}(p)} \left[\|y - (D \otimes x)w\|_2^2 \right] \end{array} \right. \quad \textcircled{1} \text{ C' est}$$

الف) ازابیکر حاصل $n'P$ اس سے $P = D \otimes X$ point wise

دو مائرس است، خودش نیز مائرس هم اندازه آنها خواهد بود.

$$P = D \otimes X \quad \Rightarrow \quad E_D[P]_{ij} = E_D[P]$$

$\Rightarrow E_D[P_{ij}] = E_D[P_{ij}]$ چون مقدار expected یک ماتریس
برابر با ماتریس expected است

$$\mathbb{E}_D[P_{ij}] = \mathbb{E}_D[(D \odot X)_{ij}] = \mathbb{E}_D[D_{ij} \times X_{ij}]$$

$$\Rightarrow E_D [P]_{ij} = X_{ij} \overbrace{E_D [D_{ij}]}^P = P X_{ij}$$

$$\text{همنیوینیتی} \rightarrow E_D [P^T P]$$

$$\begin{aligned} \left(\mathbb{E}_D [P^T P] \right)_{ij} &= \mathbb{E}_D [(P^T P)_{ij}] = E_D \left[\sum_{k=1}^N P_{ik}^T \cdot P_{kj} \right] \\ &= \mathbb{E}_D \left[\sum_{k=1}^N P_{ki} P_{kj} \right] = \mathbb{E}_D \left[\sum_{k=1}^N D_{ki} \cdot D_{kj} \cdot X_{ki} \cdot X_{kj} \right] \end{aligned}$$

$$\left(\mathbb{E}_D [P^T P] \right)_{ij} = \mathbb{E}_D \left[\sum_{k=1}^N D_{ki} D_{kj} X_{ki} X_{kj} \right] =$$

$$\sum_{k=1}^N \mathbb{E}_D [D_{ki} D_{kj} X_{ki} X_{kj}]$$

$$\text{if } i \neq j \Rightarrow \sum_{k=1}^N \mathbb{E}_D[D_{ki}] \mathbb{E}_D[D_{kj}] X_{ki} X_{kj}$$

⊗ جوں X_{ki}, X_{kj} علاجیات میں نسبت بہ توزیع D ؛ پس از خارج expected می سُوند.

$$E_D[P^T P]_{ij} = \sum_{k=1}^n P \cdot P X_{ki} X_{kj} = P^2 \sum_{k=1}^n X_{ki} X_{kj}$$

$$\Rightarrow E_D [P^T P]_{ij} = P^2 (X^T X)_{ij}$$

$$\text{if } i=j \Rightarrow \sum_{k=1}^N \mathbb{E}[P_{ki}^2] X_{ki} X_{kj} = P \sum_{k=1}^N X_{ki} X_{kj}$$

$$\Rightarrow E_P [P^T P] = P (X^T X)_{ij}$$

$$\left\{ \begin{array}{l} \|y - Pw\|_2^2 = y^T y - 2w^T P^T y + w^T P^T P w \\ P = D \odot X \end{array} \right. \quad (\downarrow)$$

$$E_{D \sim \text{Ber}(p)} \left[\|y - Pw\|_2^2 \right] = E_D \left[y^T y - 2w^T P^T y + w^T P^T P w \right]$$

$$\begin{aligned}
 L(w) &= E_D \left[\|y - Pw\|_2^2 \right] && \text{(ادامہ)} \\
 &= E_D \left[y^T y - 2w^T P^T y + w^T P^T P w \right] && \text{از جنس الگ می دایم:} \\
 &= y^T y - 2Pw^T X^T y + w^T E_D [P^T P] w && \left\{ \begin{array}{l} E_D [P] = pX \\ E_D [P^T P] = p^2 X^T X \end{array} \right. \\
 &= y^T y - 2Pw^T X^T y + p^2 w^T X^T X w && \left. \begin{array}{l} i=j \\ p = X^T X \end{array} \right. \\
 &\quad - p^2 w^T X^T X w + w^T E_D [P^T P] w
 \end{aligned}$$

$$\begin{aligned}
 &= \|y - pXw\|_2^2 + w^T E_D [P^T P] w - p^2 w^T X^T X w \\
 &= \|y - pXw\|_2^2 + w^T (E_D [P^T P] - p^2 X^T X) w
 \end{aligned}$$

حامل این یک ماتریس مغایر است چون:

$$\left[E_D [P^T P] - p^2 X^T X \right]_{ij} = \begin{cases} 0 & i \neq j \\ (p - p^2) X^T X & i = j \end{cases}$$

و سه می توانیم بجای عبارت فوق
در نظر بگیریم.

$$\begin{aligned}
 \Rightarrow L(w) &= \|y - pXw\|_2^2 + (p - p^2) w^T (\text{diagonal}(X^T X)) w \\
 &= \|y - pXw\|_2^2 + p(1-p) w^T (\text{diagonal}(X^T X)) w \\
 \text{if } \Gamma &= \left[\text{diagonal}(X^T X) \right]^{\frac{1}{2}} \Rightarrow L(\hat{w}) = \|y - pXw\|_2^2 + p(1-p) \|\Gamma w\|_2^2
 \end{aligned}$$

$$L(w) = \left\| y - p\hat{X}\hat{w} \right\|_2^2 + p(1-p) \left\| \hat{\Gamma} \hat{w} \right\|_2^2 \quad (2)$$

با استخراج $w = p\hat{w}$

$$\begin{aligned} &= \left\| y - \hat{X}w \right\|_2^2 + p(1-p) \underbrace{\left\| \hat{\Gamma} \frac{w}{p} \right\|_2^2}_{\rightarrow} \\ &= \left\| y - \hat{X}w \right\|_2^2 + \left\| \sqrt{p(1-p)} \hat{\Gamma} \frac{w}{p} \right\|_2^2 \end{aligned}$$

$$= \left\| y - \hat{X}w \right\|_2^2 + \left\| \sqrt{\frac{1-p}{p}} \hat{\Gamma} w \right\|_2^2$$

$$= \left\| y - \hat{X}w \right\|_2^2 + \left\| \Gamma w \right\|_2^2$$

با فرض $\Gamma = \sqrt{\frac{1-p}{p}} \hat{\Gamma}$ بطوریکه استخراج سود. همچنین می توانیم کل بابت مبتنی

$$L(w) = \left\| y - \hat{X}w \right\|_2^2 + \left\| \Gamma w \right\|_2^2 \quad (\text{با فرض مخلوس پذیری } \Gamma) \quad (3)$$

با استخراج $w = \Gamma^{-1} \tilde{w}$ خواهد شد. $\tilde{w} = \Gamma w$ با استخراج

$$L(w) = \left\| y - \hat{X} \Gamma^{-1} \tilde{w} \right\|_2^2 + \left\| \tilde{w} \right\|_2^2$$

$$= \left\| y - \hat{X} \tilde{w} \right\|_2^2 + \left\| \tilde{w} \right\|_2^2$$

بطوریکه $\hat{X} = X \Gamma^{-1}$ استخراج شود.

همچنین می توانیم کل بابت مبتنی rescale چیز را داشته باشیم.

ت) اگر α_i ها را ستوان های ماتریس X و $\tilde{\alpha}_i$ ها را ستوان های ماتریس \tilde{X}

در نظر بگیریم:

$$\tilde{X} = K X \Gamma^{-1}$$

چون Γ ماتریس قدری بوده و درایه های قطر اصلی آن

نم ستوان متناظر در ماتریس X است پس:

$$\Gamma = \begin{bmatrix} \|\alpha_1\|_2 & 0 & \cdots & 0 \\ 0 & \|\alpha_2\|_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \|\alpha_d\|_2 \end{bmatrix} \Rightarrow \Gamma^{-1} = \begin{bmatrix} \frac{1}{\|\alpha_1\|_2} & 0 & \cdots & 0 \\ 0 & \frac{1}{\|\alpha_2\|_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{1}{\|\alpha_d\|_2} \end{bmatrix}$$

$$\Rightarrow \tilde{X} = K X \Gamma^{-1}$$

$$\begin{bmatrix} \vdots & \vdots & \vdots \\ \tilde{\alpha}_1 & \tilde{\alpha}_2 & \cdots & \tilde{\alpha}_d \\ \vdots & \vdots & \vdots \end{bmatrix} = K \begin{bmatrix} \vdots & \vdots & \vdots \\ \alpha_1 & \alpha_2 & \cdots & \alpha_d \\ \vdots & \vdots & \vdots \end{bmatrix} \begin{bmatrix} \frac{1}{\|\alpha_1\|_2} & 0 & \cdots & 0 \\ 0 & \frac{1}{\|\alpha_2\|_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{1}{\|\alpha_d\|_2} \end{bmatrix}$$

$$\Rightarrow \tilde{X} = \left[\frac{K \alpha_1}{\|\alpha_1\|_2}, \frac{K \alpha_2}{\|\alpha_2\|_2}, \dots, \frac{K \alpha_d}{\|\alpha_d\|_2} \right]$$

$$\text{چون } \|\tilde{\alpha}_i\| = K \quad \text{و } \tilde{\alpha}_i = K \frac{\alpha_i}{\|\alpha_i\|_2}$$

می شود ستوان های ماتریس \tilde{X} ، نرم تابع برابر K داشته باشند. K همان تابع Rescale است که از نکش "D" بست آمد. این عملکرد همانند scaling و استاندارد سازی normalization است که باعث می شود ستوان های واریانس تکیه ای داشته باشند.

$$J = \frac{1}{2} \left(y_d - \sum_{k=1}^n (w_k + \delta_k) x_k \right)^2 \quad (9)$$

$$\frac{\partial J}{\partial w_k} = -x_k \left(y_d - \sum_{j=1}^n (w_j + \delta_j) x_j \right)$$

$$\Rightarrow \mathbb{E} \left[\frac{\partial J}{\partial w_k} \right] = \mathbb{E} \left[(-x_k) \left(y_d - \sum_{j=1}^n w_j x_j + \delta_j x_j \right) \right]$$

از آنجایی که از یک توزیع ناوس با میانگین صفر می‌آید پس:

$$\mathbb{E}[\delta_k] = 0$$

$$\mathbb{E} \left[\frac{\partial J}{\partial w_k} \right] = \left(\sum_{j=1}^n w_j x_j - y_d \right) x_k$$

۱) باع خطا در حال Dropout بصورت زیراست. سعی می‌نماییم فاصله y_d

از مقدار پیشینی مدل حداقل λ

$$J = \frac{1}{2} \left(y_d - \sum_{k=1}^n w_k x_k c_k \right)^2$$

در اینجا یک متغیر تصادفی از توزیع ناوسی $N(0, \alpha^2)$ می‌آید.

$$\mathbb{E}[J] = \frac{1}{2} \mathbb{E} \left[y_d^2 - 2y_d \sum_{k=1}^n w_k x_k c_k + \sum_{k=1}^n w_k^2 x_k^2 c_k^2 \right]$$

$$\mathbb{E}[c_k] = 0$$

$$\mathbb{E}[c_k^2] = \alpha^2 \quad \mathbb{E}[J] = \frac{1}{2} \left(y_d^2 + \sum_{k=1}^n w_k^2 x_k^2 \alpha^2 \right)$$

همانطور که مشخص است در رابطه 1055، عبارت $\sum_{k=1}^n w_k^2 x_k^2 \alpha^2$ انتقامی برای جلوگیری از بزرگ شدن w_k است و نقش dropout و regularization دارد.

۵) Spatial dropout کلسلی برای انعام
نامه ای وهم regularization است.

Dropout نامه جی نظریه ای به activation آموزش اعماقی می باشد.
و سعی دارد representation overfit را مقاوم کند و از جلوگیری از.

اما در feature map های لایه های کانولوشن (همال
می شود بصورت زنده drop map را feature map و بصورت خاص
برای regularization شبکه های کانولوشن مناسب است. همین
منطق مکانی دراده و روی تغیر نایز است.

سؤال ②

1D Convolution

$$Z_i = \sum_{j=0}^{K-1} w_j X_{i+j}$$

$$\left\{ \begin{array}{l} X = [x_1, x_2, \dots, x_N] \\ w = [w_1, w_2, \dots, w_K] \end{array} \right. \quad \begin{array}{l} \text{فرهن:} \\ 1 \times N \quad 1 \times K \end{array}$$

اگر طول بردار X ، N باشد و فیلتر به طول K به آن اعمال کنیم، خروجی Z به طول $N-K+1$ خواهد بود

$$\frac{\partial L}{\partial Z_i} = M_i \quad \text{چون فرض شده } \frac{\partial L}{\partial Z_i} \text{ را داریم، آنرا } M_i \text{ می‌نامیم.}$$

آنکه M_i نیز طولش برابر با $N-K+1$ است.

$$\frac{\partial Z_i}{\partial w_j} = \frac{\partial}{\partial w_j} \left[\sum_{j=0}^{K-1} w_j X_{i+j} \right] = X_{i+j}$$

$$\frac{\partial L}{\partial w_j} = \sum_{i=0}^{N-K} \frac{\partial L}{\partial Z_i} \cdot \frac{\partial Z_i}{\partial w_j} = \sum_{i=0}^{N-K} M_i X_{i+j}$$

الدرب رابطه $\frac{\partial L}{\partial w_j}$ دقت نیم همانند یک عمل نانولوشن! بعدی است به این محور که X_{i+j} ورودی است و فیلتر $(\frac{\partial L}{\partial Z_i}) M_i$ را روی آن اعمال می‌کنیم تا خروجی $\frac{\partial L}{\partial w_j}$ بدست بیاید.

سوال ③ Batch normalization

$$\mu_B = \frac{1}{m} \sum_{i=1}^m x_i$$

$$\sigma_B^2 = \frac{1}{m} \sum_{i=1}^m (x_i - \mu_B)^2$$

$$\hat{x}_i = \frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}}$$

$$y_i = \gamma \hat{x}_i + \beta = BN_{\gamma, \beta}(x_i)$$

الف) با فرض داشتن $\frac{\partial L}{\partial y_i}$ محاسبه $\frac{\partial L}{\partial \gamma}$ و $\frac{\partial L}{\partial \beta}$ پارامترهای خواسته شده را انجام می‌دهیم.

$\frac{\partial L}{\partial \gamma}$ و $\frac{\partial L}{\partial \beta}$ پارامترهای

$$\frac{\partial L}{\partial \gamma} = \frac{\partial L}{\partial y_i} \cdot \frac{\partial y_i}{\partial \gamma} = \frac{\partial L}{\partial y_i} \cdot \hat{x}_i = \sum_{i=1}^m \frac{\partial L}{\partial y_i} \cdot \hat{x}_i$$

$$\frac{\partial L}{\partial \beta} = \frac{\partial L}{\partial y_i} \cdot \frac{\partial y_i}{\partial \beta} = \frac{\partial L}{\partial y_i} \cdot 1 = \sum_{i=1}^m \frac{\partial L}{\partial y_i}$$

$\frac{\partial L}{\partial \gamma}$ نسبتی به x_i و \hat{x}_i و وروجی

$$\frac{\partial L}{\partial \hat{x}_i} = \frac{\partial L}{\partial y_i} \cdot \frac{\partial y_i}{\partial \hat{x}_i} = \frac{\partial L}{\partial y_i} \cdot \gamma$$

$$\frac{\partial L}{\partial x_i} = \frac{\partial L}{\partial \hat{x}_i} \cdot \frac{\partial \hat{x}_i}{\partial x_i} + \frac{\partial L}{\partial \mu} \cdot \frac{\partial \mu}{\partial x_i} + \frac{\partial L}{\partial \sigma_B^2} \cdot \frac{\partial \sigma_B^2}{\partial x_i} (*)$$

باید $\frac{\partial L}{\partial x_i}$ مجموع ابرای محاسبه ابتدا حساب کنیم.

ادامه ③ (الف)

$$\left\{ \begin{array}{l} \frac{\partial \hat{x}_i}{\partial x_i} = \frac{1}{\sqrt{\sigma_B^2}} = \frac{1}{\sigma_B} \\ \frac{\partial \mu}{\partial x_i} = \frac{1}{m} \\ \frac{\partial \sigma_B^2}{\partial x_i} = \frac{2(x_i - \mu)}{m} \end{array} \right.$$

$$1) \frac{\partial L}{\partial \mu} = \frac{\partial L}{\partial \hat{x}_i} \cdot \frac{\partial \hat{x}_i}{\partial \mu} + \frac{\partial L}{\partial \sigma_B^2} \cdot \frac{\partial \sigma_B^2}{\partial \mu}$$

$$\frac{\partial \hat{x}_i}{\partial \mu} = \frac{-1}{\sqrt{\sigma_B^2}} = \frac{-1}{\sigma_B}$$

$$\frac{\partial \sigma_B^2}{\partial \mu} = \frac{-2}{m} \sum_{i=1}^m (x_i - \mu) = -2 \left[\mu - \frac{m\mu}{m} \right] = 0$$

$$2) \frac{\partial L}{\partial \sigma_B^2} = \frac{\partial L}{\partial \hat{x}_i} \cdot \frac{\partial \hat{x}_i}{\partial \sigma_B^2}$$

$$\frac{\partial \hat{x}_i}{\partial \sigma_B^2} = \frac{-\sigma_B^{-3}}{2} \sum_{i=1}^m (x_i - \mu)$$

$$\Rightarrow \frac{\partial L}{\partial \sigma_B^2} = \frac{\partial L}{\partial \hat{x}_i} \cdot \frac{-\sigma_B^{-3}}{2} \sum_{i=1}^m (x_i - \mu) \Rightarrow$$

$$\frac{\partial L}{\partial \sigma_B^2} = \gamma \cdot \frac{-\sigma_B^{-3}}{2} \sum_{i=1}^m (x_i - \mu) \frac{\partial L}{\partial y_i}$$

$$1) \frac{\partial L}{\partial \mu} = \sum_{i=1}^m \frac{\partial L}{\partial \hat{x}_i} \cdot \frac{\partial \hat{x}_i}{\partial \mu} + \frac{\partial L}{\partial \sigma_B^2} \cdot \frac{\partial \sigma_B^2}{\partial \mu}$$

$$\Rightarrow \frac{\partial L}{\partial \mu} = \sum_{i=1}^m \frac{\partial L}{\partial y_i} \cdot \gamma \frac{-1}{\sigma_B}$$

با وجود به رابطه (*)

$$\begin{aligned} \frac{\partial L}{\partial x_i} &= \frac{\partial L}{\partial y_i} \cdot \gamma \frac{1}{\sigma_B} + \sum_{j=1}^m \frac{\partial L}{\partial y_j} \cdot \gamma \cdot \frac{-1}{\sigma_B} \cdot \frac{1}{m} \\ &\quad + \gamma \cdot \frac{-\sigma_B^{-3}}{2} \sum_{j=1}^m \frac{\partial L}{\partial y_j} (x_j - \mu) \cdot \frac{2(x_i - \mu)}{m} \end{aligned}$$

Simplify

$$\Rightarrow \frac{\partial L}{\partial x_i} = \frac{\gamma}{m \sigma_B} \left[m \frac{\partial L}{\partial y_i} - \sum_{j=1}^m \frac{\partial L}{\partial y_j} - \hat{x}_i \sum_{j=1}^m \frac{\partial L}{\partial y_j} \hat{x}_j \right]$$

(ب) مزایای Batch Normalization

1) باعث کاهش سُیغت Covariate (راحتی سُود و سرعت آموزش شبکه را زیاد نمایند.) تغییر در توزیع activation های مختلف شبکه در حین training می توانند باعث نپایداری و کندی آموزش سُود که BN این مشکل را حل نمایند.

2) بدلیل زیال سازی activation ها توسط BN، مقدار (هی اولیه weight ها آسان تر است و نیازی نست که (غذخه نقاط شروع را داشته باشیم).

3) چون مقدار noise کمی به محل افتاده می نمایند، ممکن است نیاز به Dropout یا لینیل های (تکریل regularization) نداشتم.

ج) استفاده از α به جای β منجر می‌شود امکان بهره‌گیری از دو پارامتر α و β را از دست بدهیم. این دو پارامتر به ترتیب بدلی Scale کردن و سیفنت activation ها است.

در این صورت شبکه امکان adapt شدن با میانلین و Scale های مختلف feature ها را از دست می‌دهد و محدود به میانلین و Scale ثابت می‌شود. بعوں γ ، شبکه Scale ثابت خواهد داشت $(\frac{1}{\beta})$ و بعوں β ، شبکه دارای میانلین حفظ خواهد بود.

اگر از این دو پارامتر استفاده نکنیم، dynamic Convergence پیداگیری و فرآیند optimization سخت‌تر و کندتر می‌شود.

د) mini-batch normalization این طریق عمل می‌کند که در هر mini-batch میانلین و Standard deviation را محاسبه می‌کند و با آن activation ها را نرمال می‌کند. حال آنکه در هر mini-batch فقط یک لاس حفظ درست باشد، میانلین و انحراف مکیاری که محاسبه می‌شود، نماینده‌ای از توزیع کلی دیاست است و در زمان است چون میانلین و انحراف مکیار با محاسبه moving average های میانلین و انحراف مکیاری های زمان training صورت می‌پذیرد ممکن است این دو متغیر به سمت feature خاصی bias شوند و از توزیع اعلی دیاست عامله بگیرند.

لهمین موقعیت باعث بوجود آمدن discrepancy یا اختلاف کمکلر بین مجموع داده و train خواهد شد. و در نتیجه شبکه دچار مسئله خواهد بود.

۱) لایه‌های کانولوشن زایش اینپوری بوجود آمده تا shift-invariant باشند.

همچنین بایس به لایه کانولوشن اجازه می‌دهد که feature intensity کلی سنساسایی سده را نادلیرد. پس حذف بایس این امکان را از پردازش و شبکه را حساس و تغییرنگیر به shift می‌کند.

BN. بعد از لایه کانولوشن سعی می‌کند با نرمال کردن mini batch فقدان بایس را تحدی جبران نماید. نظریت representation مدل پاسین می‌آید.

بایس یک پارامتر برای لایه کانولوشن حساب می‌شود که flexibility یا دلیری این لایه و شبکه را افزایش می‌دهد. پس حذف این پارامتر می‌تواند موجب کاهش انحطاط و fit نشدن دقیق مدل شود.

جواب کوتاه: بله، مسئله ایجاد می‌کند.

ب) ضرب عدد α در فرنهای شبکه تأثیری در عملکرد مدل نخواهد داشت

چون batch norm سعی می‌کند تأثیر جبران کننده داشته باشد.

اما ضرب عدد α در ورودی شبکه بر عملکرد شبکه تأثیر دارد و باعث scale سُن activation map ها و در نتیجه تغییر توزیع آن ها و تغییر خروجی شبکه می‌شود.

ج) لایه کانولوشن با اندازه کرنل K و تام S

- بازی هر پسل خروجی و تصویر ورودی با C کانل

$K \times K \times C$

$$\left\{ \begin{array}{l} h_{out} = \frac{h_{in} - K}{S} + 1 \\ w_{out} = \frac{w_{in} - K}{S} + 1 \end{array} \right. \Rightarrow K \times K \times C \times h_{out} \times w_{out}$$

برای کل پسل های خروجی

آخر C کانل خروجی را سه بستم در C فربخواهد سد.

تام S با کرنل K و تام C pooling average (ii)

- بازی هر پسل خروجی و تصویر ورودی با C کانل

$K \times K \times C$

این مقدار عمل جمع هر بار انجام می شود تا فرآیند محاسبه average انجام شود.

$$\left\{ \begin{array}{l} h_{out} = \frac{h_{in} - K}{S} + 1 \\ w_{out} = \frac{w_{in} - K}{S} + 1 \end{array} \right. \Rightarrow K \times K \times C \times h_{out} \times w_{out}$$

برای کل پسل های خروجی

* چون مرتبه محاسبات خواسته شده از نوشتمن دلیل تعداد محاسبات صرف نظر شده است.

Batch Norm (iii)

در زمان آموزش : بازی هر کانل درودی، میانگین و انحراف محاسبه می شود. C mean و Std حساب می کند که هر کدام $K \times K$ محاسبه (در) (جمع و ...)

پس این نیز برابرست با :

در زمان Inference : میانگین خاصی ندارد. فقط بازی هر کانل تفزيق و تقسیم مربوط به نرمال سازی را انجام می دارد.

۴) چندین دلیل مختلف برای استفاده از کافلوس 1×1 وجود دارد.

برای کاهش عمق تصاویر ورودی استفاده می‌شود. به این دلیل که پیچیدگی محاسباتی واستفاده حافظه را کاهش دهن.

لهم می‌کند تا ظرفیت و عمق شبکه را بدون افزایش قابل توجه تعداد پارامتر افزایش دهن.

در بلاک‌های depthwise separable Conv. جهت پاس آوردن تعداد پارامترهای شبکه (بعداز لایه depthwise Conv) جهت افزایش تعداد کانال استفاده می‌شود. مثل شبکه MobileNet.

می‌تواند بخوان regularizer و تنظیم کننده در کانل‌ها کمک کند و از overfit جلوگیری کند.

۵) دلیل استفاده از لایه‌های pooling برای کاهش ابعاد تصویر از بحاظ طول و عرض، تغییر ناپذیری نسبت به translation مثل shift، قابلیت aggregate کردن فیچرهای یک منطقه و تکراری فیچرهای پراهمیت را و همچنین بهینه‌بودن از نقد محاسباتی بدلیل نداشت پارامتر در مقایسه با لایه Convolution است.

آرچه ممکن است باعث loss of information و هزینه‌های زیادی لفظ برای استفاده دارد.

۶) فرمول محاسبه receptive field برای شبکه CNN برای لایه K :

$$R_K = 1 + \sum_{j=1}^K (F_j - 1) \prod_{i=0}^{j-1} S_i \quad \begin{cases} S_i \rightarrow \text{لایه } i \\ F_j \rightarrow \text{فیلتر لایه } j \\ S_0 = 1 \end{cases}$$

$$R_4 = 1 + 2[1] + 1[1 \times 2] + 2[1 \times 2 \times 1] + 1[1 \times 2 \times 1 \times 2] = 13$$

لایه چهارم (خروجی) برابر با 13×13 است. receptive field

input: $512 \times 512 \times 3$

⑤ جواب

Network (A):

#C

(T)

after i: $256 \times 256 \times 3$

$$n_{\text{out}} = \left\lfloor \frac{n_{\text{in}} + 2P - K}{S} \right\rfloor + 1$$

after ii: $256 \times 256 \times 64$

after iii: $128 \times 128 \times 64$

after iv: $128 \times 128 \times 128$

after v: $64 \times 64 \times 128$

after vi: $64 \times 64 \times 256$

Network (B):

#C

after i: $256 \times 256 \times 64$

after ii: $128 \times 128 \times 128$

after iii: $64 \times 64 \times 256$

Network (A): i: $4 \times 4 \times 3 = 48$

ii: $1 \times 1 \times 3 \times 64 = 192$

iii: $4 \times 4 \times 64 = 1024$

iv: $1 \times 1 \times 64 \times 128 = 8192$

v: $4 \times 4 \times 128 = 2048$

vi: $1 \times 1 \times 128 \times 256 = 32768$

(c)

44272

params

$$\text{Network (B): } i : 4 \times 4 \times 3 \times 64 = 3072 \quad (1)$$

$$ii : 4 \times 4 \times 64 \times 128 = 131072$$

$$iii : 4 \times 4 \times 128 \times 256 = 524288$$

$$\sum \text{params} = 658432$$

multiplications (2)

آخر فیلتر $K \times K$ باشد و عکس ورودی depth-wise Conv.

($N_i \times N_i \times m$) داشته باشد و خروجی فیلتر به سایز \tilde{m}

شود آنها تعداد ضربهای این فیلتر برابر با $N_0 \times N_0 \times m$ باشد

$$m \times K \times K \times N_0 \times N_0 = \text{multiplications of Depth-wise Conv.}$$

آخر فیلتر \tilde{m} روی عکس ورودی با point wise Conv.

($N_i \times N_i \times m$) اعمال شود و به تعداد p تا از این فیلتر داشته باشیم

$(N_i \times N_i \times p)$ آنها تعداد ضربهای این فیلتر برابر با $(N_i = N_0)$ باشند

$$m \times 1 \times 1 \times N_i \times N_i \times p = \text{multiplications of point-wise Conv.}$$

آخر فیلتر استاندارد به سایز $K \times K$ باشد و عکس ورودی Convolution

($N_i \times N_i \times m$) داشته باشیم Channel \tilde{m} باشد

شود (آنها تعداد ضربهای این فیلتر $K \times K \times m$ باشند) $N_0 \times N_0 \times p$

$$m \times K \times K \times N_0 \times N_0 \times p = \text{multiplications of Convolution filter}$$

ج) پس با توجه به فرمول (۱) صفحه قبل:

Network (A):

$$i: m K^2 N_0^2 = 3 \times 4 \times 256^2$$

$$ii: m N_0^2 P = 3 \times 256^2 \times 64$$

$$iii: 64 \times 4^2 \times 128^2$$

$$iv: 64 \times 128^2 \times 128$$

$$v: 128 \times 4^2 \times 64^2$$

$$vi: 128 \times 64^2 \times 256$$

$$\begin{aligned} \text{Total mults} &= 2^{20} (3 + 12 + 16 + \\ &\quad 128 + 8 + 128) \\ &= 2^{20} \times 295 \end{aligned}$$

Network (B):

$$i: m K^2 N_0^2 P = 3 \times 4^2 \times 256^2 \times 64$$

$$ii: 64 \times 4^2 \times 128^2 \times 128$$

$$iii: 128 \times 4^2 \times 64^2 \times 256$$

$$\text{Total mults} = 2^{26} (3 + 32 + 32) = 2^{26} \times 67$$

ج) با توجه می‌شود که خروجی هر دو شبکه هم سایز بود و می‌تواند به جای هم استفاده شود.

شبکه A از depth-wise convolution استفاده می‌کند سپس رکورت از شبکه B پارامتر دارد و همچنین تعداد مقادیر سیستم (فیلتر) آن سپس رکورت از شبکه B است. (حدود ۱۵ برابر کمتر) در نتیجه سرعت محاسبات شبکه A، ۱۵ برابر شبکه B است. همچنین از لحاظ استخراج memory شبکه A بسیار بینهایت را دارد.

(d) مدل Depth-wise Separable MobileNet براساس ساخته شده است.

Filters

در این مدل، لایه های کانولوشن استاندارد به depth-wise convolution و یک لایه کانولوشن 1×1 تبدیل شده اند. این تبدیل باعث effcient شدن این مدل نسبت به شبکه های کانولوشنی سابق شده و از لحاظ محاسباتی و تعداد پارامترها بسیار بینهایتر شده است. همانطور که در جنس "ج" فرمول های تعداد محاسبات برای دو حالت را دیدیم:

1) Standard Convolution

$$N \cdot D_K \cdot D_K \cdot M \cdot D_F \cdot D_F$$

D_F خروجی فیلتر
سایز فیلتر D_K

M تعداد کمال تقویر ورودی

N تعداد فیلترها

2) Depth-wise Convolution + Point-wise Convolution

$$D_K \cdot D_K \cdot M \cdot D_F \cdot D_F + N \cdot M \cdot D_F \cdot D_F$$

$$\frac{\text{حال}}{\text{حالت}} = \frac{D_K \cdot D_K \cdot M \cdot D_F \cdot D_F + N \cdot M \cdot D_F \cdot D_F}{N \cdot D_K \cdot D_K \cdot M \cdot D_F \cdot D_F} \Rightarrow$$

$$\frac{1}{N} + \frac{1}{D_K^2} = \frac{\text{نسبت محاسبات}}{\text{نسبت محاسبات}} \text{ کمتر شده} \text{ و سریعتر شده است.}$$

همچنین این کاهش پارامتر و محاسبات به ما این اجازه را می دهد که شبکه های عمیق را باسته باشیم تا عملکرد خوبی ارائه دهد و feature های پیچیده را با دلیلر.

و) Resolution Multiplier ، width Multiplier به نام α که hyper parameter است.

در این مدل وجود دارد.

هایپر پارامتر width multiplier این اجرازه را می دهد تا زمانی که شرایط بله شبکه کوچک

و سریع در جایی راسیم. (به لایل محدودیت منابع، هزینه و...) بتوانیم با استفاده از این هایپر پارامتر این کار را با این شکل انجام دهیم.

برای تعداد لایه مخفی با $width\ multiplier = \alpha$ ، تعداد کنال های تصویر و روزی αM و تعداد کنال های فروژی $N \propto \alpha M$ خواهد شد. و هزینه محاسباتی شکل با حفظ این هایپر پارامتر سیکل روبرو می شود:

$$D_K \cdot D_K \cdot \alpha M \cdot D_F \cdot D_F + \alpha M \cdot \alpha N \cdot D_F \cdot D_F$$

که α می تواند بین ۰ و ۱ باشد. $\alpha \in (0, 1]$

پس مرتبت این هایپر پارامتر کی هست تعداد پارامتر و محاسبات و افزایش سرعت و این رفت قابل قبول می باشد.

هایپر پارامتر width multiplier هم مانند Resolution Multiplier اینها کی هست پارامتر و هزینه محاسباتی شبکه را می دهد. هایپر پارامتر ρ در اندازه تصویر و عورتی و representation های داخلی هر لایه فریب خواهد شد و باعث می شود که هزینه محاسباتی سیکل روبرو شود: $\rho \in (0, 1]$

$$D_K \cdot D_K \cdot \alpha M \cdot \rho D_F \cdot \rho D_F + \alpha M \cdot \alpha N \cdot \rho D_F \cdot \rho D_F$$

کافیست این هایپر پارامتر در تقدیم محاسباتی برابر ρ^2 می باشد.

مرتبت این هایپر پارامتر نیز مانند α درایی (شبکه کوچکتر، سیکل، با منابع محدود و سریع) می باشد.

$$L_c = \frac{1}{2} \sum_{i=1}^m \|x_i - c_{y_i}\|_2^2$$

آ) تابع فنر Center loss سعی می‌کند تا سهیل‌های مربوط به کلاس را نزدیک مرکز آن کلاس کند. یعنی feature‌های عینی کلاس‌های مسماه را نزدیک مرکز هم‌افقر باشند و از لحاظ فضای embedding ای در یک منطقه iteration قرار بگیرند. آنرا زیرین چون خود مرکز کلاس‌ها نیز در هر درحال اینست و این اینست براساس mini batch صورت می‌برد و نه کل دیاست، پس استفاده از این تابع فنر به تنها می‌باشد مسعود فرازینه بارگیری بسیار نیازدار شود. (بازگیری درسی صورت نمی‌گیرد.)
به همین دلیل این تابع فنر را همراه با تابع فنر Softmax برای (سته بندی) هم‌ویر استفاده می‌کند تا باعث بهبود نکلار شود.

$$L_{\text{Combined}} = L_{\text{Softmax}} + \lambda L_{\text{center loss}}$$

$$x_i = w z_i + b \quad , \quad L_c = \frac{1}{2} \sum_{i=1}^m \|x_i - c_{y_i}\|_2^2 \quad (b)$$

$$\frac{\partial L_c}{\partial c_j} = \Delta c_j = \frac{\sum_{i=1}^m \delta(y_i=j) \cdot (c_j - x_i)}{\varepsilon + \sum_{i=1}^m \delta(y_i=j)}$$

مرکز (سته) ها

$\delta(y_i=j)$ زمانی که کلاس سهیل نام برابر باشد، خواهد بود و در بقیه حالات صفر است.

* ع برای صفر نشدن مخرج اضافه شده است.

$$c_j = c_{j-1} - \Delta c_j$$

بروزرسانی پارامترهای لایه خضی:

$$\frac{\partial L_c}{\partial \alpha_i} = (x_i - c_{y_i})$$

$$\frac{\partial x_i}{\partial z_i} = w^T, \quad \frac{\partial x_i}{\partial b} = 1, \quad \frac{\partial x_i}{\partial w} = z_i^T$$

$$\left\{ \begin{array}{l} Z_i = Z_i - \Delta Z_i = Z_i - (\alpha_i - c y_0) W \\ W = W - \Delta W = W - (\alpha_i - c y_i) Z_i^T \\ b = b - \Delta b = b - (\alpha_i - c y_i) \end{array} \right.$$

ج) PEDCC-loss CNN میکنند که برای توزیع این مقاله برای مسکن‌های خود را در میان مردم ارائه می‌نمایند و ممکن است از آنچه فوریت دارد.

این روش از مراکز کلاس با توزیع ملحوظ است از پسین تعریف شده بهره می برد و توزیع $h_{\text{feature}}(\text{feature map})$ hidden ها را در فضای \mathbb{R}^n سازی کند.

PEDCC، پارامترهای لایه Classification را با وزن‌های نسبت PEDCC-loss جایگزین می‌کند که حین آموزش مدل نسبت می‌ماند و فاصله بین کلاس‌های مختلف را مانزانم کند. با ترتیب تابع همنزد Cross entropy و MSE بین سهیل‌ها و مراکز کلاس، روش PEDCC سعی می‌کند به تغزیح از feature بررسی که سینترین Compactness بین سهیل‌های یک کلاس (دورن کلاسی) باشد و همچنین تسمیه دورن سهیل‌های خارج یک کلاس مشترک هم مانزانم شود.

این روش بعده افزایش دقت Classification و Convergence پایدار ساخته می شود.