

زها و هکرل ۹۸۱۰۵۱۳۸

سوال (۱)

(۱.۱) تفاوت ۲ لای در این است که در original inception یک لای

5x5 با گرفتن مقادیر 3x3 جابجا شده است.

$$[3 \times 3 \times F + 1] + [3 \times 3 \times F + 1] = 18F + 2$$

↑
تعداد جابجیه‌ها

$$[F \times 5 \times 5 + 1] = 25F + 1$$

تعداد لای‌ها برابر است با

$$(25F + 1) - (18F + 2) = (7F - 1)$$

یعنی در هر لای در مقدار ۵ لای کمتر استفاده کرده است (۸۰٪ کاهش)

(۱.۲)

فرض کنیم لای ۵ در ۵، همان‌طور که در ۳ در ۳ باشد. در آن صورت مقادیر

حاصل می‌شود، تعداد لای‌ها ۵۰٪ است و تعداد جابجیه‌ها ۲۵٪

$$9n = (n \times 3 \times 3) = n \times 9$$

$$18n = (n \times 4 \times 4) + (n \times 4 \times 4) = n \times 18$$

$$\frac{18n}{9n} = \frac{1}{9} \approx 11\%$$

یعنی ۱۱٪ محاسبه کمتر شده است. [تعداد لای‌ها برابر است با (۱۸n) و در هر لای ۹ جابجیه

استفاده از لای (۱۸x۱۸)، پس (۱۸x۱۸) متناظر است با استفاده از لای (۱۸x۱۸)

استفاده از لای ۱۸x۱۸، ۱۸x۱۸، ۱۸x۱۸، ۱۸x۱۸، ۱۸x۱۸، ۱۸x۱۸، ۱۸x۱۸، ۱۸x۱۸، ۱۸x۱۸، ۱۸x۱۸

حافظه بسیار کمتری را برای لای‌ها نیاز دارد و در هر لای ۱۸x۱۸، ۱۸x۱۸، ۱۸x۱۸، ۱۸x۱۸، ۱۸x۱۸، ۱۸x۱۸، ۱۸x۱۸، ۱۸x۱۸، ۱۸x۱۸، ۱۸x۱۸

۱۸x۱۸، ۱۸x۱۸، ۱۸x۱۸، ۱۸x۱۸، ۱۸x۱۸، ۱۸x۱۸، ۱۸x۱۸، ۱۸x۱۸، ۱۸x۱۸، ۱۸x۱۸

$$n \times n \times n \times n = n^2 \quad \text{استاندارد} = 0$$

$$(n \times 1) = 0 \quad (n \times n) \times (1 \times n) + (n \times 1) \times (1 \times n) = 2n \times n$$

$$\frac{2n \times n}{n^2} = \frac{2}{n} = 0 \quad \text{همچنین برابر شده است}$$

در واقع باید به صورت $n \times n$ باشد

آنگاه $n = 2$ هم عبارت $n \times n$ شود

این خاصیت که بتوان به صورت $n \times n$ از کرنل $(1 \times n)$ و $(n \times 1)$

استاد شود به صورت $n \times n$ باشد
spatially seprable

(1.4)

تقریباً در شکل \downarrow inception \downarrow pooling

$$L = 14 \times 14 \times 3C_0 + 14 \times 14 \times 7C_0 + K^2 \times 3C_0$$

تقریباً در شکل

$$R = 14 \times 14 \times 7C_0 + 30 \times 30 \times 2C_0 + K^2 \times 3C_0$$

$$\frac{R}{L} = \frac{2 + 8K^2 \times 3C_0}{1 + 2K^2 \times 3C_0} \quad \text{نسبت هم عبارت در برابر K است}$$

مقدار K باشد

[از لحاظ محاسباتی بهینه‌تر باشد]

تقریباً در شکل: ابتدا K را کم کند سپس عبارت $n \times n$

اما در صورت راست ابتدا عبارت $n \times n$ را در $n \times n$ قرار می‌دهد

باشد شود $n \times n$ هم عبارت $n \times n$ باشد

اما از لحاظ bottleneck شدن $n \times n$ در $n \times n$ است

در واقع $n \times n$ $n \times n$ $n \times n$

SANA

12.4

۲۰۲) تستیمون اردو دستاویز میں دارد و اس کے اردو معنی یہ ہیں کہ وہ (نیکو) آدمی ہے۔
تستیمون سے تستیمون کی ضرب بھی آسکتی ہے۔ درغیر اس صورت میں۔

من فخرنا

$$L = \|n - \hat{n}\| + KL(N(\mu_x, \sigma_x^2) \| N(\mu_y, \sigma_y^2))$$

$$\mathcal{L}(\lambda, \mu) + \lambda \sum_i \|\nabla_{\mathbf{x}} h_i\|_2^2$$

این فرم جدید تر است و فرم latent h نسبت به فرم قبلی
به گونه ای است که نسبت به فرم minimal n کمالات
ضروری است.

(۲.۴) در فرایند VAE توزیع که داده‌ها و در آن آمده است تک گشته خواص بدین
رشد فلوکس که داده شده
داده‌ها به داده‌ها بسته

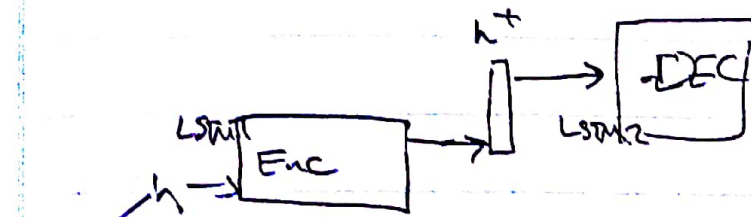
۱۳۴

(۴.۱) واحد exploding gradient
استاندارد از gradient clipping که دامنه داده‌ها را
توزیع‌ها ضمیمه بزرگ باشد با استاندارد clipping (مانند
کوتاه مقدار داده‌ها) شبکه را محدود می‌کند.
[اینهمه از clip کنیم] تا جهت معکوس شود.
واحد vanishing gradient:
تغییر ضمیمه که منتهی به نرسیدن، انقباض
استفاده از پیاده‌سازی (LSTM) برای داده‌ها
در داده‌ها که هرگز به نرسیدن و به نرسیدن
تغییر

(۴.۲) LSTM کار با استاندارد فکال اطلاعات گذشته
LSTM در طبقه t = اطلاعات از اطلاعات گذشته، هر آنچه
در هر طبقه در هر طبقه از اطلاعات گذشته
آنها و طبقه گذشته استفاده می‌کند.
فکال در دامنه اطلاعات گذشته را در $t-1$ ، h_{t-1} ، h_{t-2} ، ...
رابطه هر طبقه با اطلاعات گذشته $t-1$ ، h_{t-1} ، h_{t-2} ، ...
SA

(۴.۳) در حالتی که ما می‌خواهیم در یک شبکه انتقالی استفاده از این شبکه‌ها را داشته باشیم.

برای این کار می‌توانیم یک RNN برای encoder و یک RNN دیگر برای decoder داشته باشیم. در این حالت، encoder یک input را می‌گیرد و یک hidden state h_t را تولید می‌کند. این hidden state h_t به عنوان input برای decoder عمل می‌کند. در این حالت، decoder یک output را تولید می‌کند.



در اینجا، h_t (در صورت نیاز) به عنوان input برای LSTM2 عمل می‌کند.

در اینجا، h_t به عنوان input برای LSTM2 عمل می‌کند. در اینجا، h_t به عنوان input برای LSTM2 عمل می‌کند. در اینجا، h_t به عنوان input برای LSTM2 عمل می‌کند.

در اینجا، h_t به عنوان input برای LSTM2 عمل می‌کند. در اینجا، h_t به عنوان input برای LSTM2 عمل می‌کند. در اینجا، h_t به عنوان input برای LSTM2 عمل می‌کند.

در اینجا، h_t به عنوان input برای LSTM2 عمل می‌کند. در اینجا، h_t به عنوان input برای LSTM2 عمل می‌کند. در اینجا، h_t به عنوان input برای LSTM2 عمل می‌کند.

در اینجا، h_t به عنوان input برای LSTM2 عمل می‌کند. در اینجا، h_t به عنوان input برای LSTM2 عمل می‌کند. در اینجا، h_t به عنوان input برای LSTM2 عمل می‌کند.

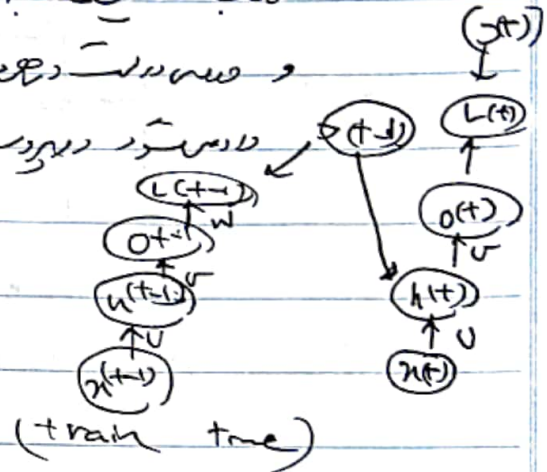
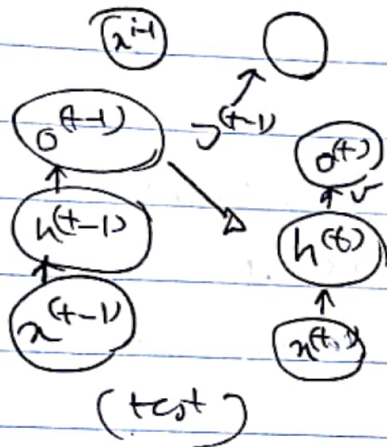
Teacher forcing

با این مدل در زمان t مقدار $h(t)$ را می‌توانیم محاسبه کنیم
 به سادگی از فرمول $h(t) = Wx(t) + Uh(t-1) + b$ استفاده می‌کنیم.

حالا به درجه در زمان
 می‌توانیم ظاهر می‌شود.

و می‌توانیم درجه‌ها را

در زمان t محاسبه کنیم.



اگر طول [Encoder] اطلاعات زیاد باشد و اطلاعات $h(t)$ را می‌توانیم محاسبه کنیم
 می‌توانیم به سادگی در زمان t محاسبه کنیم.

در زمان t می‌توانیم به سادگی در زمان t محاسبه کنیم.

از این مدل می‌توانیم به سادگی در زمان t محاسبه کنیم.

$$h^{(t)} = \begin{bmatrix} h_1^{(t)} \\ h_c^{(t)} \end{bmatrix} \quad \text{آورد } \begin{cases} 0 \leftarrow \# \\ 1 \leftarrow \text{آورد} \end{cases} \quad (a)$$

↑
آورد : 1
: 0

$$h^{(t)} = g(w_n^{(t)} + b) = \begin{bmatrix} h_1^{(t)} \\ h_c^{(t)} \end{bmatrix} = g \left(\begin{bmatrix} w_{11} & w_{1c} \\ w_{r1} & w_{rc} \end{bmatrix} \begin{bmatrix} x_1^{(t)} \\ x_c^{(t)} \end{bmatrix} + \begin{bmatrix} b_1 \\ b_c \end{bmatrix} \right) = \begin{bmatrix} g(w_{11}x_1^{(t)} + w_{1c}x_c^{(t)} + b_1) \\ g(w_{r1}x_1^{(t)} + w_{rc}x_c^{(t)} + b_c) \end{bmatrix}$$

$$j = 1 \Rightarrow y^{(t)} = y^{(1)} = g \left(\begin{bmatrix} v_1 \\ v_c \end{bmatrix}^T \begin{bmatrix} h_1^{(t)} \\ h_c^{(t)} \end{bmatrix} + c_0 \right) =$$

$$\Rightarrow y^{(1)} = g(v_1 h_1^{(1)} + v_c h_c^{(1)} + c_0)$$

$$\left. \begin{aligned} (h_1^{(1)}, h_c^{(1)}) &= (0, 1) \Rightarrow (v_r + c_0) > 1 \\ (h_1^{(1)}, h_c^{(1)}) &= (1, 0) \Rightarrow (v_1 + c_0) > 1 \\ (h_1^{(1)}, h_c^{(1)}) &= (0, 0) \Rightarrow c_0 < 1 \end{aligned} \right\} \Rightarrow \begin{cases} c_0 = 0 \\ v_1 = v_c = 1 \end{cases}$$

$$h^{(t)} = \begin{bmatrix} g(b_1) \\ g(b_c) \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

$$\left. \begin{aligned} (x_1^{(t)}, x_c^{(t)}) &= (1, 0) \Rightarrow b > 1 \\ (x_1^{(t)}, x_c^{(t)}) &= (1, 1) \Rightarrow w_{1c} + b_1 < 1 \\ (x_1^{(t)}, x_c^{(t)}) &= (0, 1) \Rightarrow w_{1c} + b_1 < 1 \\ (x_1^{(t)}, x_c^{(t)}) &= (1, 1) \Rightarrow w_{1c} + w_{11} + b_1 < 1 \end{aligned} \right\} \Rightarrow \begin{cases} b_1 = -0.5 \\ w_{11} = w_{1c} = 1 \end{cases}$$

$$2 \leq 1 \Rightarrow g(2) \quad \text{SANA}$$

برای اینکه خروجی خواسته شده را داشته باشیم باید فواید بیشتری را

$$j^T + 1 \quad y^{(t)} = g(u^T h^{(t)} + r y^{(t-1)} + c)$$

$$\Rightarrow y^{(t)} = g \left(\begin{bmatrix} u_1 \\ u_c \end{bmatrix}^T \begin{bmatrix} h_1^{(t)} \\ h_c^{(t)} \end{bmatrix} + r y^{(t-1)} + c \right)$$

$$(7) \Rightarrow y^{(t)} = g(u_1 h_1^{(t)} + u_c h_c^{(t)} + r y^{(t-1)} + c)$$

2 inputs, 1 output

$$(y^{(t-1)}, h_1^{(t)}, h_c^{(t)}) = \begin{cases} (0, 0, 0) & \Rightarrow c < 1 \\ (0, 0, 1) & \Rightarrow 1+c < 1 \\ (0, 1, 0) & \Rightarrow 1+c < 1 \\ (1, 0, 0) & \Rightarrow r+c < 1 \\ (1, 0, 1) & \Rightarrow 1+r+c > 1 \\ (1, 1, 0) & \Rightarrow 1+r+c > 1 \end{cases}$$

$$\Rightarrow \boxed{c = -1, r = 1}$$

$$(h_1^{(t)}, h_c^{(t)}) = \begin{cases} (0, 0) & \Rightarrow b_1 < 1 \\ (0, 1) & \Rightarrow w_1 r + b_1 < 1 \\ (1, 0) & \Rightarrow w_c r + b_c < 1 \\ (1, 1) & \Rightarrow w_1 r + w_c r + b_1 > 1 \end{cases}$$

$$\Rightarrow \boxed{b_1 = -1, w_1 = w_c = 1}$$

SANA

(701) در سده های زبانی متون به هم می آید.
 نکته دارد ~~همه~~ دنیا به با اقبال شیر دنیا است که همه از خط غیری
 در میان کافیه با فیه می آید.

یک کلمه ای از این کلمات با این کلمات
 در یک جمله به هم وصل می شود
 و این کلمات را می توان به هم وصل کرد
 و این کلمات را می توان به هم وصل کرد
 و این کلمات را می توان به هم وصل کرد

نکته: در این حالت Context کپی می شود.

انسان کے لئے جو کچھ ہے

(1) اسی طرح one hot سے مراد ہے کہ ایک ہی صف میں

(۲) پس در r embedded $h = v_c = v_{\overline{r}}$

$u = \sum v_i \vec{e}_i$, $u_{-m}, u_{-m+1}, \dots, u_{-1}, u_{+1}, \dots, u_{+m}$ (for m large enough)

for each $y_i = \text{softmax}(u)$ (K

$y^{(c-m)}, \dots, y^{(c-1)}, y^{(c)}, y^{(c+1)}, \dots, y^{(c+m)}$ are not variables, y is a variable (A)

نوع loss مقادیر از نوع Cross entropy زیر باشد:

$$L = -\log P(w_{c-m}, \dots, w_{c-1}, w_{c+1}, \dots, w_{c+m} | w_c)$$

$$= -\log \prod_{j=0, j \neq m}^m P(w_{c-m+j} | w_c)$$

$$= -\log \prod_{j=0, j \neq m}^m P(w_{c-m+j} | v_c)$$

$$= -\log \prod_{j=0, j \neq m}^m \frac{\exp(u_{c-m+j}^T v_c)}{\sum_{k=1}^{|V|} \exp(u_k^T v_c)}$$

$$= -\sum_{j=0, j \neq m}^m u_{c-m+j}^T v_c + m \log \sum_{k=1}^{|V|} \exp(u_k^T v_c)$$

$$C \Rightarrow h = W^T x$$

(4.5)

$$u = u_1 = W^T W^T x$$

$$y = y_1 = \text{softmax}(u) = \text{softmax}(W^T W^T x)$$

$$L = L(w, w') = L(u_1(w, w'), \dots, u_{|V|}(w, w'))$$

$$\Rightarrow \frac{\partial L}{\partial w_{ij}} = \sum_{k=1}^{|V|} \frac{\partial L}{\partial u_k} \frac{\partial u_k}{\partial w_{ij}}$$

$$\Rightarrow \frac{\partial L}{\partial w_{ij}'} = \sum_{k=1}^{|V|} \frac{\partial L}{\partial u_k} \frac{\partial u_k}{\partial w_{ij}'} \leftarrow \text{در اینجا}$$

توجه: از این به بعد، قدر از ضرب می‌کنیم و تفاضل می‌گیریم.

$$k \neq j \Rightarrow \frac{\partial u_k}{\partial w_{ij}'} = 0$$

$$, k = j \Rightarrow \frac{\partial u_k}{\partial w_{ij}'} \neq 0$$

$$\Rightarrow \frac{\partial L}{\partial w_{ij}'} = \frac{\partial L}{\partial u_j} \frac{\partial u_j}{\partial w_{ij}'}$$

$$\left\{ \begin{array}{l} \frac{\partial L}{\partial u_j} = -\delta_{jj} + y_j := e_j \quad \leftarrow \text{اینجا } e_j \text{ بردار واحد است} \\ \delta_{jj} = \begin{cases} 1 & : j = j^* \\ 0 & : \# \end{cases} \end{array} \right. \quad \text{تفاوت بین target و predicted}$$

$$\frac{\partial u_j}{\partial w_{ij}'} = \sum_{k=1}^{|V|} w_{ik} z_k$$

$$\Rightarrow \frac{\partial L}{\partial w_{ij}'} = e_j \sum_{k=1}^{|V|} w_{ik} z_k$$

$$\frac{\partial u_k}{\partial w_{ij}'} = w'_{jk} z_i \Rightarrow \frac{\partial L}{\partial w_{ij}'} = \sum_{k=1}^{|V|} (-\delta_{kk} + y_k) w'_{jk} z_i$$

SANA

