a)

$$f = \sigma(w_{xf} x_t + w_{hf} h_{t-1} + b_f) = \sigma\left([w_{hf}, w_{xf}] \begin{bmatrix} h_{t-1} \\ x_t \end{bmatrix} + b_f\right)$$

$$i = \sigma(w_{xi} x_t + w_{hi} h_{t-1} + b_i)$$

$$g = \tanh(w_{xg} x_t + w_{hg} h_{t-1} + b_g) \Big\} \rightarrow \text{input\_gate} = i \odot g$$

$$o = \sigma(w_{xo} x_t + w_{ho} h_{t-1} + b_o)$$

$$c_t = c_{t-1} \odot f + \text{input\_gate}$$

$$h_t = \tanh(c_t) \cdot o$$

b)

$$\frac{dE}{dh_t} = E_{delta} \quad , \quad \frac{dE}{dc_t} = \frac{dE}{dh_t} \times \frac{dh_t}{dc_t} = E_{delta} \cdot (1 - \tanh^2(c_t))$$

$$\frac{dE}{do} = \frac{dE}{dh_t} \cdot \frac{dh_t}{do} = E_{delta} \cdot \tanh(c_t) \quad , \quad \frac{dE}{d\text{input}} = \frac{dE}{dc_t} \cdot \frac{dc_t}{d\text{input}} = E_{delta} \cdot (1 - \tanh^2(c_t)) \cdot 1$$

$$\frac{dE}{dg} = \frac{dE}{d\text{input}} \cdot \frac{d\text{input}}{dg} = E_{delta} \cdot (1 - \tanh^2(c_t)) \cdot i \quad , \quad \frac{dE}{di} = \frac{dE}{d\text{input}} \cdot \frac{d\text{input}}{di} = E_{delta} \cdot (1 - \tanh^2(c_t)) \cdot g$$

$$\frac{dE}{df} = \frac{dE}{dc_t} \cdot \frac{dc_t}{df} = E_{delta} (1 - \tanh^2(c_t)) \cdot c_{t-1}$$

$$\frac{dE}{dw_{xo}} = \frac{dE}{do} \cdot \frac{do}{dw_{xo}} = E_{delta} \cdot \tanh(c_t) \cdot o(1-o) \cdot x_t$$

$$\frac{dE}{dw_{ho}} = \frac{dE}{do} \cdot \frac{do}{dw_{ho}} = E_{delta} \cdot \tanh(c_t) \cdot o(1-o) \cdot h_{t-1}$$

$$\frac{dE}{dw_{hg}} = \frac{dE}{dg} \frac{dg}{dw_{hg}} = E_{delta} \cdot (1 - \tanh^2(c_t)) \cdot i \cdot (1 - g^2) \cdot h_{t-1}$$

$$\frac{dE}{dw_{xg}} = \frac{dE}{dg} \frac{dg}{w_{xg}} = E_{delta} \cdot (1 - \tanh^2(c_t)) \cdot i \cdot (1 - g^2) \cdot x_t$$

$$\frac{dE}{dw_{xi}} = E_{delta} \cdot (1 - \tanh^2(c_t)) \cdot g \cdot i(1-i) \cdot x_t \quad , \quad \frac{dE}{dw_{hi}} = E_{deta} , (1 - \tanh^2(c_t)) \cdot g \cdot (i)(1-i) \cdot h_{t-1}$$

$$\frac{dE}{dw_{xf}} = E_{delta} \cdot (1 - \tanh^2(c_t)) \cdot f(1-f) \cdot x_t \qquad \frac{dE}{dw_{hf}} = E_{delta} \cdot (1 - \tanh^2(c_t)) \cdot f(1-f) \cdot h_{t-1}$$

$$\frac{dE}{dbf} = E_{delta} \cdot (1-\tanh^2(ct)) \cdot f(1-f) \quad , \quad \frac{dE}{dbi} = E_{delta} \cdot (1-\tanh^2(ct)) \cdot g \cdot (i(1-i))$$

$$\frac{dE}{dbg} = E_{delta} \cdot (1-\tanh^2(ct)) \cdot i \cdot (1-g^2) \quad , \quad \frac{dE}{bo} = E_{delta} \cdot \tanh(ct) \cdot o(1-o)$$

$$\frac{dE}{dx_t} = E_{delta} \cdot \Big(\tanh(ct) \cdot (o(1-o)) \cdot W_{xo} + (1-\tanh^2(ct)) \cdot i \cdot (1-g^2) \cdot W_{xg} + 1-\tanh^2(ct) \cdot g \cdot (i(1-i)) \cdot W_{xi} + (1-\tanh^2(ct)) \cdot f(1-f) \cdot W_{xf} \Big)$$

$$\frac{dE}{dh_{t-1}} = E_{delta} \cdot \Big(\tanh(ct) \cdot (o(1-o)) \cdot W_{ho} + (1-\tanh^2(ct)) \cdot i \cdot (1-g^2) \cdot W_{hg} + 1-\tanh^2(ct) \cdot g \cdot (i(1-i)) \cdot W_{hi} + (1-\tanh^2(ct)) \cdot f(1-f) \cdot W_{hf} \Big)$$

(2) اگر فرض کنیم q و k ماتریس های تصادفی با ابعاد $d_k * d_k$ باشند و ورودی ها از توزیع $N \sim (0,1)$ پیروی و iid هستند داریم:

آیتم ۱ ضرب داخلی q_i در k_i برابر است با $\rightarrow \langle q_i, k_i \rangle = \sum_j q_{ij} k_{ij} \sim N(0, d_k)$

این ضرب داخلی از توزیع نرمال با میانگین صفر و واریانس d_k پیروی می‌کند. بنری جلوگیری از بزرگ شدن ضرب داخلی با scale کردن مقادیر در $\frac{1}{\sqrt{d_k}}$ به توزیع $N(0,1)$ خواهیم رسید و از Gradient vanishing جلوگیری می‌شود.

(3)

(a

1- با توجه به اینکه α_i ها مقادیری بین صفر و یک و هم چنین $\sum_i \alpha_i = 1$ است می‌توان از آن‌ها به عنوان توزیع احتمال استفاده کرد.

2- در صورت شباهت بسیار زیاد query با یکی از کلیدها و عدم شباهت با سایر کلیدها، اکثر وزن روی α مربوط به آن کلید خواهد بود.

3- اگر شرایط قسمت 2 باشد، آنگاه مقدار z بیشتر مربوط به v_i ای می‌باشد که کلید آن شباهت زیادی با query داشته است.

4- وقتی query با یکی از کلیدها شبیه باشد، Attention بیشتر به Value مربوط به آن کلید توجه می‌کند و خروجی ها بیشتر تحت تاثیر یا فته از Value آن هاست.

b) -1

$$v_a = c_1 a_1 + c_2 a_2 + \cdots + c_m a_m \rightarrow a_j^T v_a = \underbrace{c_1 a_j^T a_1}_{0} + \cdots + \underbrace{c_j a_j^T a_j}_{1} + \cdots + \underbrace{c_m a_j^T a_m}_{0} = c_j$$

$$v_b = d_1 b_1 + d_2 b_2 + \cdots + d_p b_p \rightarrow a_j^T v_b = \underbrace{d_1 a_j^T b_1}_{0} + \cdots + \underbrace{d_p a_j^T b_p}_{0} = 0$$

$$\Longrightarrow \text{if } M = \begin{bmatrix} a_1^T \\ \vdots \\ a_m^T \end{bmatrix} \Rightarrow M(v_a + v_b) = M v_a + \underbrace{M v_b}_{0} = v_a$$

-2

$$c \approx \frac{1}{2}(v_a + v_b) = \frac{1}{2} v_a + \frac{1}{2} v_b$$
$$\alpha_a = \alpha_b, \ \alpha_{i \neq a,b} \sim 0$$
$$\rightarrow q = \beta(k_a + k_b), \quad \beta \text{ یک عدد بزرگ}$$

c)

-1 از آنجایی که α ها بسیار کوچک پس داریم $k_i \approx \mu_i$ ، $q = \beta k_a + \beta k_b$ ، $\beta \gg 0$

-2 توزیع گاوسی

$$\alpha \simeq 0 \rightarrow k_a \simeq \gamma \mu_a, \ \gamma \sim N(1, 1/2)$$
$$k_i^T q \simeq 0, \ i \neq \{a,b\}$$
$$k_b^T q = \beta$$
$$\Rightarrow c = \frac{e^{\gamma\beta}}{e^{\gamma\beta} + e^{\beta}} v_a + \frac{e^{\beta}}{e^{\beta} + e^{\gamma\beta}} v_b$$

$\rightarrow$ با افزایش هر مقدار c به سمت v_a و با کوچک کردن هر مقدار c به سمت v_b خواهد رفت، و اگر هر دو را به یک عدد مانند موارد قبلی خواهد بود.

d)

-1 $\beta \gg 0$ ، $q_1 = q_2 = \beta(k_a + k_b)$

-2 در صورت استفاده از multi head_Attention و هر چه تعداد head بیشتر شود مقدار هر دو قسمت قبل به میانگین یعنی $\frac{1}{2}$ میل می‌کند، $c \sim \frac{1}{2}(v_a + v_b)$ خواهد بود.

در Vision Transformers، تصویر ورودی به تعدادی Patch بدون overlap تقسیم می‌شود و سپس هر Patch flatten می‌شود $\left(N \times (P^2 \times c)\right)$ که N تعداد Patch ها و P سایز هر Patch (قبل از flat شدن) و c تعداد چنل‌های تصویر است.
هر Patch مانند هر توکن در sequence است و از سایز تصاویر ورودی تغییر کند تعداد و سایز Patch ها تغییر می‌کند.

با Pretrain کردن مدل در ابعاد پایین تعداد پارامترها، درنتیجه سرعت محاسبات، حافظه بهبود می‌یابند و با Fine_tune کردن آن نهایی با سایز تصاویر بالاتر به generalization بهتر مدل کمک می‌کند.

در صورت تغییر ابعاد تصاویر ها، ممکن است به معماری های جدید سازگار با ابعاد جدید نیاز باشد.

روش پیشنهادی: Crop یا Padding تصویر ورودی برای رسیدن به ابعاد دلخواه.

- استفاده از شبکه‌های CNN برای رسیدن به ابعاد دلخواه ورودی.

$$e_t = \begin{bmatrix} \sin\left(\frac{t}{f_1}\right) \\ \cos\left(\frac{t}{f_1}\right) \\ \sin\left(\frac{t}{f_2}\right) \\ \cos\left(\frac{t}{f_2}\right) \\ \vdots \\ \sin\left(\frac{t}{f_{\frac{d_{model}}{2}}}\right) \\ \cos\left(\frac{t}{f_{\frac{d_{model}}{2}}}\right) \end{bmatrix}, \quad f_m = \frac{1}{\lambda_m} := 10000^{\frac{2m}{d_{model}}}, \quad T^{(k)} E_{t,:} = E_{t+k} \quad (1)$$

قرقی نیست دائم: $T^{(k)} = \begin{bmatrix} \Phi_1^{(k)} & & & \\ & \Phi_2^{(k)} & & \\ & & \ddots & \\ & & & \Phi_{\frac{d_{model}}{2}}^{(k)} \end{bmatrix}$

$\Phi_m^{(k)} = \begin{bmatrix} \cos(r_{mk}) & -\sin(r_{mk}) \\ \sin(r_{mk}) & \cos(r_{mk}) \end{bmatrix}^T$

و سایر درایه ها ماتریس صفر با ابعاد 2×2 باشند.

$\sin\lambda_m(t+k) = \cos r_{mk} \sin\lambda_m t + \sin r_{mk} \cos\lambda_m t$

$\cos\lambda_m(t+k) = -\sin r_{mk} \sin\lambda_m t + \cos r_{mk} \cos\lambda_m t$

(1) $\rightarrow \begin{bmatrix} \cos(r_{mk}) & \sin(r_{mk}) \\ -\sin(r_{mk}) & \cos(r_{mk}) \end{bmatrix} \begin{bmatrix} \sin\lambda_m t \\ \cos\lambda_m t \end{bmatrix} = \begin{bmatrix} \sin\lambda_m(t+k) \\ \cos\lambda_m(t+k) \end{bmatrix}$

قانون جمع $\sin, \cos$ $\Longrightarrow$ اگر $\lambda = r$ $\Rightarrow$ رابطه قبلی صحیح خواهد بود.

$\Phi_m^{(k)} = \begin{bmatrix} \cos\lambda_m k & \sin\lambda_m k \\ -\sin\lambda_m k & \cos\lambda_m k \end{bmatrix}^T, \quad \lambda_m = 10000^{\frac{-2m}{d_{model}}}$