به نام خدا

یادگیری ژرف

نیم‌سال دوم ۴۰۲-۱۴۰۱

مدرس: دکتر مهدیه سلیمانی

دانشکده‌ی مهندسی کامپیوتر

تمرین سری سوم | مدل‌های ترنسفورمر + RNN | نمره کل: ۲۰+۱۶۵ نمره

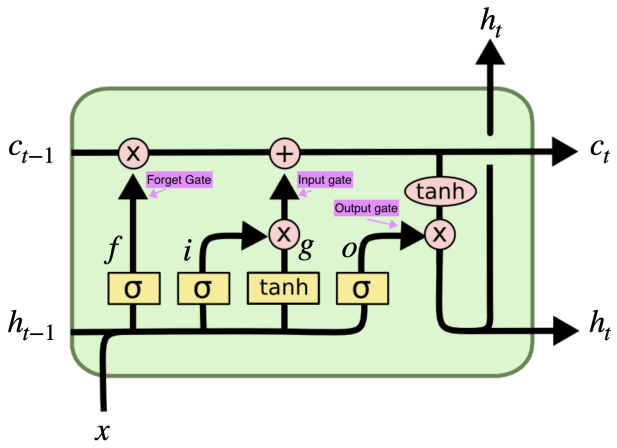
لطفا نکات زیر را رعایت کنید:

- سوالات خود را از طریق پست مربوط به تمرین در Quera مطرح کنید.
- در هر کدام از سوالات، اگر از منابع خارجی استفاده کرده‌اید باید آن را ذکر کنید. در صورت همفکری با افراد دیگر هم باید نام ایشان را در سوال مورد نظر ذکر نمایید.
- پاسخ ارسالی واضح و خوانا باشد. در غیر این صورت ممکن است منجر به از دست دادن نمره شود.
- پاسخ ارسالی باید توسط خود شما نوشته شده باشد. به اسکرین‌شات از منابع یا پاسخ افراد دیگر نمره‌ای تعلق نمی‌گیرد.
- در صورتی که بخشی از سوال‌ها را جای دیگری آپلود کرده و لینک آن را قرار داده باشید، حتما باید تاریخ آپلود مشخص و قابل اعتنا باشد.
- تمام پاسخ‌های خود را در یک فایل با فرمت HW3_[StudentID]_[Fullname].zip روی کوئرا قرار دهید.
- برای ارسال هر تمرین تا ساعت ۲۳:۵۹ روز ددلاین فرصت دارید. مهلت تاخیر (مجاز و غیر مجاز) برای این تمرین، ۱۰ روز است.

سوالات نظری (۵۵ نمره)

سوال ۱: سوال اول (۱۰ نمره)

در این سوال می خواهیم نحوه ی مشتق گیری خطای پس انتشار را در در هر یک از سلول های شبکه های LSTM بررسی کنیم. سلول LSTM زیر را در نظر بگیرید:



(آ) برای سلول نشان داده شده، مربوط به خروجی Forget gate ، Input gate ، Output gate، c_t و h_t را بنویسید.(۲ نمره)
وزن gate های مختلف، به صورت زیر است:

$$\text{Input gate: } w_{xi}, w_{xg}, b_i, w_{hj}, w_g, b_g$$

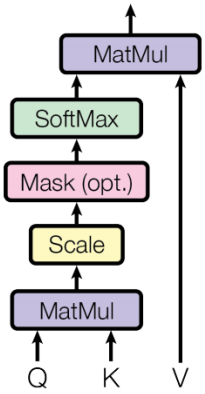
$$\text{Forget gate: } w_{xf}, b_f, w_{hf}$$

$$\text{Output gate: } w_{xo}, b_o, w_{ho}$$

(ب) اگر گرادیان عبوری از سلول بالا به صورت $E_{delta} = \frac{dE}{dh_t}$ باشد ، آنگاه با استفاده از قانون زنجیره‌ای، گرادیان را نسبت به هر یک از پارامترهای سلول LSTM محاسبه کنید.(۸ نمره)

سوال ۲: سوال دوم (۱۰ نمره)

در اسلایدهایی که با مکانیسم توجه آشنا شدیم، به طور کلی، مکانیسم توجه جزء کلیدی بسیاری از معماری‌های شبکه عصبی مدرن مانند ترنسفرمرها است. این به شبکه اجازه می دهد تا به طور انتخابی بر روی بخش های خاصی از توالی ورودی تمرکز کند، و آن را به ویژه برای کارهایی که نیاز به پردازش توالی طولانی از داده ها دارند، مفید می کند.
در مکانیسم توجه ضرب نقطه‌ای مقیاس‌شده[۱]، حاصل ضرب نقطه‌ای ماتریس‌های بردار پرس وو جو(Q) Query و بردار کلید (K) Key قبل از عبور از تابع softmax در عکس جذر ابعاد ماتریس کلید (dk) ضرب می‌شود(مقیاس بندی). این کار برای جلوگیری از بزرگ شدن حاصل ضرب نقطه‌ای انجام می‌شود که می‌تواند باعث مشکلاتی در عملکرد softmax شود. استدلال ریاضی پشت این ضریب مقیاس را توضیح دهید.



$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

نکته: تصور کنید Q و K ماتریس های تصادفی $d_k \times d_k$ بعدی باشند، که در آن هر ورودی یک توزیع تصادفی با میانگین ۰ و ۱ واریانس است. هر ورودی مستقل از یکدیگر است.

سوال ۳: سوال سوم (۲۰ نمره)

خود توجهی چند سر[۲] جزء اصلی مدل سازی ترنسفرمرها است. در این سوال، ما می خواهیم بررسی کنیم که چرا خود توجهی چند سر می‌تواند به خود توجهی تک سر ترجیح داده شود. می دانیم که مکانیسم توجه را می توان به عنوان یک عملیات پرس و جو دید که در آن $q \in \mathbb{R}$ عبارت پرس و جو، در کنار $\{v_1, ...v_n\}, v_i \in \mathbb{R}^d$ به عنوان مجوعه ای از بردارهای مقدار و مجموعه ای از بردارهای کلید $k_1...k_n, k_i \in \mathbb{R}^d$، به صورت زیر مشخص می شود:

$$c = \sum_{i=1}^{n} v_i \alpha_i \tag{۱}$$

$$\alpha_i = \frac{\exp\left(k_i^\top q\right)}{\sum_{j=1}^{n} \exp\left(k_j^\top q\right)} \tag{۲}$$

در این جا α_i، "وزن توجه"[۳] نامیده می شود. مشاهده کنید که خروجی $c \in \mathbb{R}^d$ یک میانگین وزن دار از بردار های مقدار با وزن α_i است.

(آ) **امکان شباهت بردار خروجی مکانیسم توجه به یکی از بردار های مقدار(۴ نمره):** در این بخش می خواهیم بگوییم که در شرایط خاصی، امکان این وجود دارد که بردار c شباهت زیادی به یکی از بردار های v_i از بردار های مقدار داشته باشد، اما از آن جایی که شما باید به این نتیجه برسید، به سوالات زیر پاسخ دهید.

(i) توضیح دهید که چرا α را می توان به عنوان توزیع احتمال طبقه ای تفسیر کرد.

(ii) توزیع α معمولاً نسبتاً "پراکنده" است. جرم احتمال بین α_i های مختلف پخش می شود. اما همیشه هم به این صورت نیست. (در یک جمله) توضیح دهید که در چه شرایطی توزیع طبقه‌ای α تقریباً تمام وزن خود را روی مقداری $j \in \{1, ..., n\}, \alpha_j$ می‌گذارد؟ (یعنی $\alpha_j \gg \sum_{i \neq j} \alpha_i$). چه چیزی باید در مورد پرش و جوی q و یا کلید های $\{k_1, \ldots, k_n\}$ صادق باشد؟

[۱] Scaled dot product attention mechanism
[۲] Multi-headed self-attention
[۳] attention weights

(iii) با توجه به آنچه که در (ii) بیان کردید، اگر در نتیجه ی آن، توزیع α پراکنده باشد، توضیح دهید خروجی c چه ویژگی هایی خواهد داشت.

(iv) در راستای توضیح امکان شباهت بین بردار خروجی مکانیسم توجه و یکی از بردار های مقدار به صورت خلاصه توضیح دهید از (ii) و (iii) چه نتیجه ای می توان گرفت؟

(ب) **قابلیت ترکیب(۶ نمره):** یک مدل ترانسفورماتور شبه جای تمرکز بر تنها یک بردار v_j، ممکن است بخواهد اطلاعات را از چندین بردار منبع ترکیب کند. موردی را در نظر بگیرید که در عوض می‌خواهیم اطلاعات دو بردار v_a و v_b را با بردارهای کلیدی k_a و k_b ترکیب کنیم.

(i) چگونه باید دو بردار d بعدی v_a، v_b را در یک بردار خروجی c ترکیب کنیم تا اطلاعات هر دو بردار حفظ شود؟ در یادگیری ماشینی، یکی از راه های رایج برای انجام این کار، گرفتن میانگین است: $c = \frac{1}{2}(v_a + v_b)$. استخراج اطلاعات در مورد بردارهای اصلی v_a و v_b از c حاصل ممکن است سخت به نظر برسد، اما تحت شرایط خاصی می توان این کار را انجام داد. در این مشکل، خواهیم دید که چرا این مورد است.
فرض کنید که اگرچه ما v_a یا v_b را نمی‌دانیم، اما می‌دانیم که v_a در یک زیرفضای A قرار دارد با m بردار پایه $\{a_1, a_2, ..., a_m\}$، در حالی که v_b در یک زیرفضای ناهمپوشان B قرار دارد که توسط p بردارهای پایه تشکیل شده است $\{b_1, b_2, ...b_p\}$. (این بدان معناست که هر v_a را می توان به صورت ترکیبی خطی از بردار پایه های فضای مربوطه بیان کرد. همه بردارهای پایه دارای نرم ۱ و متعامد با یکدیگر هستند.) علاوه بر این، فرض کنید که دو زیرفضا متعامد هستند. یعنی $a_j^T b_k = 0$ برای همه j و k ها اگر $j \neq k$.
با استفاده از بردارهای پایه $\{a_1, a_2, ..., a_m\}$، یک ماتریس M بسازید به طوری که برای بردارهای دلخواه $v_a \in A$ و $v_b \in B$، می‌توانیم از M برای استخراج v_a از بردارهای مجموع $s = v_a + v_b$ استفاده کنیم. به عبارت دیگر، ما می خواهیم M را طوری بسازیم که برای هر v_a و v_b آنگاه $Ms = v_a$).

نکته: M و v_a ، v_b هر دو باید به صورت یک بردار در $\mathbb{R}^d$ بیان شوند، نه بر حسب بردارهای A و B .
نکته: با توجه به اینکه بردارهای $a_1, a_2, \ldots, a_m$ هم متعامد هستند و هم مبنای نرمال شده‌ای برای v_a ، می‌دانیم که مقداری $c_1, c_2, \ldots, c_m$ وجود دارد به طوری که $v_a = c_1 a_1 + c_2 a_2 + \cdots + c_m a_m$.

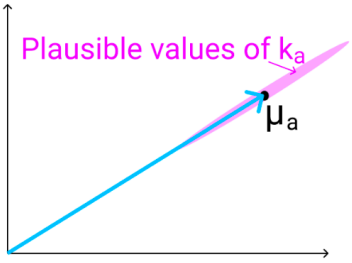
(ii) مانند قبل، تصور کنید v_a و v_b دو بردار مقدار باشند که به ترتیب مربوط به بردارهای کلیدی k_a و k_b هستند. فرض کنید که همه بردارهای کلید متعامد هستند، بنابراین $k_i^T k_j = 0$ برای همه ی $i \neq j$ و همه بردارهای کلیدی دارای نرم ۱ هستند. یک عبارت برای بردار پرس و جو q پیدا کنید به طوری که $c \approx \frac{1}{2}(v_a + v_b)$.۱

(ج) **رفع یکی از اشکالات توجه تک سر(۵ نمره):** در قسمت قبل دیدیم که چگونه ممکن است توجه تک سر به طور مساوی روی دو مقدار متمرکز شود. همین مفهوم را می توان به راحتی به هر زیر مجموعه ای از بردار های مقدار تعمیم داد. در این سوال خواهیم دید که چرا این راه حل، عملی نیست. مجموعه ای از بردارهای کلیدی $\{k_1...k_n\}$را در نظر بگیرید که اکنون به صورت تصادفی نمونه برداری شده اند، $k_i \sim \mathcal{N}(\mu_i, \Sigma_i)$، که در آن میانگین $\mu_i \in \mathbb{R}^d$ برای شما شناخته شده است، اما کوواریانس Σ_i ناشناخته است. علاوه بر این، فرض کنید که بردارهای میانگین μ_i همگی عمود هستند. $\mu_i^T \mu_j = 0$ اگر $i \neq j$، و نرم واحد هستند، $||\mu_i|| = 1$.

(i) تصور کنید ماتریس های کوواریانس برای α بسیار کوچک به صورت $\Sigma_i = \alpha I, \forall i \in \{1, \ldots, n\}$ برای تعریف می شوند. یک پرس و جو q را بر حسب μ_i که k_i ها از توزیع نرمالی با میانگین آن ها نمونه برداری می شوند، طوری طراحی کنید که مانند قبل، $c \approx \frac{1}{2}(v_a + v_b)$ باشد و یک استدلال مختصر در مورد اینکه چرا این اتفاق می افتد ارائه دهید.

(ii) اگرچه توجه تک سر در برابر آشفتگی های کوچک در کلیدها مقاوم است، اما برخی از انواع آشفتگی های بزرگتر ممکن است مشکل‌های بزرگتری ایجاد کنند. به طور خاص، در برخی موارد، یکی از بردارهای کلیدی k_a ممکن است از نظر نرم بزرگتر یا کوچکتر از بقیه باشد، در حالی که همچنان در همان راستای میانگین μ_a است. به عنوان مثال، تصور کنید کوواریانس نمونه بسیار کوچک a را به صورت $\Sigma_a = \alpha I + \frac{1}{2}(\mu_a \mu_a^T)$ در نظر بگیریم (همانطور که در شکل ۲ نشان داده شده است). این باعث می‌شود که k_a تقریباً در جهتی مشابه μ_a باشد، اما با واریانس‌های بزرگ در اندازه. علاوه بر این، در نظر بگیرید که $\Sigma_i = \alpha I$ برای همه ی $i \neq a$.
وقتی چندین بار از $\{k_1, \ldots, k_n\}$، نمونه برداری می کنید و از بردار q که در قسمت i تعریف کردید استفاده می کنید، انتظار دارید بردار c از نظر کیفی برای نمونه های مختلف چگونه باشد و مقدار آن در چه حدود مقادیری تغییر می کند؟

(د) **راه حل با استفاده از مزایای توجه چند سر(۵ نمره) :** اکنون می خواهیم در مورد توانایی خود توجهی چند سر در مواجهه با این مشکل صحبت کنیم. ما یک نسخه ساده از خود توجهی چند سر را در نظر خواهیم گرفت که مشابه خود توجهی تک سر است که در این تمرین ارائه کرده ایم، به جز اینکه دو بردار پرس و جو $(q_1 q_2)$ تعریف شده است که منجر به یک جفت از بردارها $(c_1 c_2)$ می شود، که هر یک خروجی خود توجهی تک سر نسبت به بردار پرس و جو مربوطه ی خود هستند. خروجی نهایی خود توجهی چند سر، میانگین آنهاست، $\frac{1}{2}(c_1 + c_2)$.
همانطور که در بخش سوم این سوال، مجموعه‌ای از بردارهای کلیدی را در نظر گرفتیم $k_1, ..., k_n$ که به طور تصادفی نمونه برداری می شوند، $k_i \sim N(\mu_i, \Sigma_i)$، که در آن میانگین μ_i برای شما شناخته شده است، اما کوواریانس Σ_i ناشناخته است. فرض می کنیم که میانگین μ_i متعامد ، $\mu_i^T \mu_j = 0$ اگر $i \neq j$، و نرم واحد هستند، $||\mu_i|| = 1$.



شکل ۱: بردار μ_i (در اینجا به صورت دو بعدی به عنوان مثال نشان داده شده است)، با محدوده ی مقادیر ممکن k_a به رنگ بنفش نشان داده شده است. همانطور که قبلاً ذکر شد،k_a تقریباً در جهتی مشابه μ_a است، اما ممکن است اندازه ی بزرگتر یا کوچکتر داشته باشد.

(i) فرض کنید که ماتریس های کوواریانس برای α های بسیار کوچک به صورت $\Sigma_i = \alpha I$ هستند. بردار q_1 و q_2 را به گونه پیدا کنید که بردار c برابر میانگین دو بردار v_a و v_b شود.($\frac{1}{2}(v_a + v_b)$)

(ii) فرض کنید که ماتریس های کوواریانس برای α های بسیار کوچک به صورت $\Sigma_a = I + \frac{1}{2}(\mu_a \mu_a^T)$ و برای هر i اگر $i \neq a$ آنگاه برابر $\Sigma_i = \alpha I$ هستند. بردارهای پرس و جو q_1 و q_2 را که در بخش قبلی سوال طراحی کردید، در نظر بگیرید. انتظار دارید خروجی c براساس نمونه های مختلف بردارهای کلیدی چگونه باشد یا به عبارت دیگر چه رابطه ای بین بردار c و بردار های مقدار مربوط به کلید ها باشد؟ لطفاً به طور خلاصه توضیح دهید که چرا. شما می توانید مواردی را که در آن $k_a^T q_i < 0$ است نادیده بگیرید.

سوال ۴: سوال چهارم (۵ نمره)

مرسوم است که برای آموزش مدل‌های طبقه‌بندی تصویر، مدل‌ها را روی تصاویر کوچک‌تر (مثلا با اندازه ۲۲۴ در ۲۲۴) pretrain می‌کنند و سپس آن‌ها را روی اندازه بزرگ‌تر fine-tune می‌کنند. مدل Vision Transformer (ViT) و MLPMixer برای تغییر اندازه تصویر ورودی چه چالش‌هایی دارند؟ راهکار پیشنهادی شما برای حل این مشکلات چیست؟ خود این مدل‌ها از چه راهکاری استفاده کرده‌اند؟

سوال ۵: سوال پنجم (۱۰ نمره)

فرض کنید $\boldsymbol{E} \in \mathbb{R}^{n \times d_{\text{model}}}$ ماتریسی باشد که شامل بردارهای d_{model}ـبعدی $\boldsymbol{E}_{t,:}$ است که موقعیت t را در یک دنباله ورودی به طول n کد می‌کند. تابع $e : \{1, \ldots, n\} \to \mathbb{R}^{d_{\text{model}}}$ این ماتریس را ایجاد می‌کند و به صورت زیر تعریف می‌شود:

$$e(t) = \boldsymbol{E}_{t,:} := \begin{bmatrix} \sin\left(\frac{t}{f_1}\right) \\ \cos\left(\frac{t}{f_1}\right) \\ \sin\left(\frac{t}{f_2}\right) \\ \cos\left(\frac{t}{f_2}\right) \\ \vdots \\ \sin\left(\frac{t}{f_{\frac{d_{\text{model}}}{2}}}\right) \\ \cos\left(\frac{t}{f_{\frac{d_{\text{model}}}{2}}}\right) \end{bmatrix},$$

که در آن فرکانس‌ها به صورت زیر است:

$$f_m = \frac{1}{\lambda_m} := 10000^{\frac{2m}{d_{\text{model}}}}.$$

نشان دهید که تبدیل خطی $\boldsymbol{T}^{(k)} \in \mathbb{R}^{d_{\text{model}} \times d_{\text{model}}}$ وجود دارد که برای آن

$$\boldsymbol{T}^{(k)} \boldsymbol{E}_{t,:} = \boldsymbol{E}_{t+k,:}$$

برقرار است برای هر انحراف موقعیتی $k \in \{1, \ldots, n\}$ در هر موقعیت معتبر $t \in \{1, \ldots, n-k\}$ در دنباله.

سوالات عملی (۲۰+۱۱۰ نمره)

سوال ۶: پیاده‌سازی معماری ترنسفورمر (۲۵ نمره)

نوت‌بوک transformer_architecture.ipynb را کامل کنید. در این نوت‌بوک، لازم است قدم به قدم به کمک توضیحات ارائه شده، معماری شبکه ترنسفورمر را پیاده‌سازی کنید.
همان‌طور که در فایل نوت‌بوک نیز نوشته شده است، آموزش شبکه مدنظر نیست و صرفا هدف تمرین آشنایی با پیاده‌سازی عملی لایه‌های شبکه ترنسفورمر است.

سوال ۷: تنظیم‌مجدد مدل‌های ترنسفورمری با استفاده از Adapters (۵۰ نمره)

مدل‌های زبانی پیش‌آموزش‌دیده نتایج خوبی به صورت عمومی دارند اما برای استفاده خاص منظوره از این مدل‌های ترنسفورمری نیاز داریم آن را روی داده‌های وظیفه پایین دستی مورد نظر خود تنظیم مجدد کنیم. همان‌طور که می‌دانید تعداد پارامترهای این مدل‌ها بسیار زیاد است و این کار نیاز به داده زیاد و محاسباتی سنگین دارد.

در مقاله Adapters ترفندی برای تنظیم مجدد سبک مدل‌های ترنسفورمری ابداع شده است و نشان داده علی‌رغم محاسبات بسیار سبک‌تر، عملکرد قابل توجهی کسب کرده است. به طور خلاصه، به جای تنظیم دقیق کل مدل برای هر کار جدید، آداپتورها لایه‌های خاصی را بین لایه‌های مدل از پیش آموزش‌دیده اضافه می‌کنند و با فریز کردن لایه‌های پیش‌آموزش‌دیده ترنسفورمر، تنها همین لایه‌ها را آموزش می‌دهند.

در این تمرین شما بایستی با استفاده از داده‌های آموزشی مجموعه‌دادگان IMDB Movie Review (که حاوی نظرات برچسب گذاری شده برای تجزیه و تحلیل احساسات است)، مدل پیش‌آموزش‌دیده RoBERTa را تنظیم مجدد کنید.

نهایتا لازم است با محاسبه معیارهایی مانند accuracy، precision، recall و F1-score عملکرد مدل تنظیم شده را روی مجموعه آزمایشی ارزیابی کنید.

نکات:

- می‌توانید مدل پیش‌آموزش دیده را از hugging face بارگذاری کنید.
- برای انجام این تمرین مجاز به استفاده از کتابخانه‌های آماده adapter نیستید.
- در انتخاب ابرپارامترهایی مثل نرخ یادگیری، اندازه دسته و تعداد epoch آزاد هستید. سعی کنید مقادیری تنظیم کنید که نتایج ارزیابی قابل قبولی بگیرید.
- پاسخ خود را در قالب یک فایل با نام Adapter.ipynb در پوشه پاسخ‌ها قرار دهید.

سوال ۸: توضیح‌نویسی برای تصویر (۲۰ نمره) (امتیازی)

با استفاده از ابزارهای آماده کتابخانه HuggingFace و fine-tune کردن مدل‌های pretrained از نوع ViT و GPT یک سیستم تولید متن برای تصویر بسازید.

انتخاب مدل با خود شماست، ولی مدل‌های `google/vit-base-patch16-224-in21` و distilgpt2 انتخاب‌های خوبی هستند.

از دیتاست COCO برای آموزش استفاده کنید. (این دیتاست برای بعضی تصاویر چندین متن دارد که می‌توانید بصورت رندوم از آن‌ها استفاده کنید.)

درنهایت برای تست روی بیست تصویر که در داده آموزش نبوده است، خروجی بگیرید.

همچنین متریک‌های Rouge، `Bleu-1`، `Bleu-2`، `Bleu-3`، `Bleu-4`، `BERT-Score` را روی قسمت تست دیتاست گزارش کنید. می‌توانید تعداد سیصد تصویر را برای قسمت تست درنظر بگیرید تا زمان این بررسی کوتاه‌تر شود. با توجه به این که چندین متن برای هر تصویر ممکن است موجود باشند، می‌توانید بیشینه امتیاز با این متن‌ها را درنظر بگیرید.

مطالعه HF_Vision_Encoder_Decoder_Models در حل این تمرین لازم است.

سوال ۹: مدل زبانی سطح کاراکتر توسط RNN (۳۵ نمره)

به نوتبوک داده شده با نام `char-rnn_language_model_(redacted).ipynb` مراجعه کنید.