

## بخش تئوری

## سوال اول

سیستم پرسش و پاسخ حوزه بسته

سیستم‌های پرسش و پاسخ حوزه باز برای پوشش دادن طیف گسترده‌ای از موضوعات و ارائه پاسخ با دسترسی به حجم عظیمی از اطلاعات از منابع مختلف طراحی شده‌اند. این سیستم‌ها هدف دارند تا با استفاده از مجموعه‌ای بزرگ از اسناد که موضوعات متنوعی مانند مقالات، کتاب‌ها، وبسایت‌ها و پایگاه‌های داده را پوشش می‌دهند، به سؤالات پاسخ دهند. هدف اصلی آنها ارائه دانش جامع و عمومی در طیف گسترده‌ای از موضوعات است. ویژگی‌های کلیدی سیستم‌های پرسش و پاسخ حوزه باز عبارتند از:

- **دامنه:** آنها با سؤالات در طیف گسترده‌ای از موضوعات بدون هیچ محدودیت از پیش تعیین شده در حوزه سر و کار دارند.
- **منابع داده:** آنها از داده‌های ساختار نیافته از منابع متعدد مانند ویکی‌پدیا، مقالات خبری و سایر مجموعه‌های بزرگ متنی استفاده می‌کنند.
- **تکنیک‌ها:** این سیستم‌ها اغلب از تکنیک‌های پردازش زبان طبیعی (*NLP*) مانند بازیابی اطلاعات، رتبه‌بندی اسناد و تحلیل معنایی برای درک سؤال، شناسایی اطلاعات مرتبط و استخراج پاسخ از حجم زیادی از داده‌های ساختار نیافته استفاده می‌کنند.

سیستم پرسش و پاسخ حوزه بسته

از سوی دیگر، سیستم‌های پرسش و پاسخ حوزه بسته بر حوزه‌ها یا موضوعات خاص متمرکز هستند و پاسخ را بر اساس یک مجموعه از دانش از پیش تعیین شده ارائه می‌دهند. این سیستم‌ها برای پاسخگویی به سؤالات در یک حوزه یا موضوع محدود مانند یک وبسایت خاص، یک پایگاه داده یا یک حوزه تخصصی مانند پزشکی یا حقوق از پیش سفارش شده هستند. ویژگی‌های کلیدی سیستم‌های پرسش و پاسخ حوزه بسته عبارتند از:

- **دامنه:** آنها به سؤالات در یک حوزه یا موضوع خاص محدود می‌شوند.
- **منابع داده:** آنها از داده‌های ساختاریافته از پایگاه‌های دانش یا پایگاه‌های داده خاص آن حوزه استفاده می‌کنند.
- **تکنیک‌ها:** این سیستم‌ها اغلب از رویکردهای قاعده‌مبنای گراف‌های دانش یا انتولوژی‌ها برای مطابقت سؤال با دانش موجود و بازیابی پاسخ مرتبط‌ترین استفاده می‌کنند. ماهیت ساختاریافته داده‌ها امکان پاسخ‌های دقیق‌تر و صحیح‌تر در محدوده محدود را فراهم می‌کند.

## سوال دوم

تعریف سیستم‌های درک متن ماشینی (*Machine Reading Comprehension*)

در این دسته از سیستم‌ها به دنبال طراحی و تولید مدلی هستیم که قادر به درک و استدلال بر اساس متون مشخص ورودی برای پاسخگویی به سؤالات مرتبط با همان متن باشد. بنابراین در نهایت این مدل‌ها متون نوشته شده توسط انسان را درک می‌کنند و پاسخ‌های دقیقی را بر اساس اطلاعات موجود در آن‌ها ارائه می‌دهند.

ارتباط میان *MRC* و *QA*

*MRC* و *QA* به طور کلی *Task* متمایزی نسبت به هم هستند، اما با اینحال ارتباط آن‌ها با هم نزدیک است. بسیاری از سیستم‌های پیشرفته *QA*، به ویژه آنهایی که با سؤالات حوزه باز سر و کار دارند، به شدت بر روی تکنیک‌های *MRC* برای درک و استخراج اطلاعات مرتبط از منابع متنی تکیه دارند. از سوی دیگر *MRC*، می‌تواند به عنوان یک شکل تخصصی از *QA* در نظر گرفته شود که تمرکز آن بر پاسخگویی به سؤالات بر اساس یک متن داده شده است.

به طور کلی می‌توان در حوزه‌های زیر این دو *Task* را با هم مقایسه کرد:

۱. **تعریف وظیفه:** *MRC* معمولاً به عنوان یک وظیفه تعریف می‌شود که در آن به مدل یک متن و یک سؤال داده می‌شود و مدل باید با درک و استدلال بر روی متن، پاسخی ارائه دهد. از سوی دیگر، *QA* می‌تواند شامل بازیابی پاسخ‌ها از منابع مختلف از جمله پایگاه‌های داده ساختاریافته، پایگاه‌های دانش یا اینترنت باشد، بدون اینکه لزوماً به یک متن خاص متکی باشد.
۲. **دامنه:** *MRC* بر درک و استدلال بر روی یک متن داده شده متمرکز است، در حالی که *QA* می‌تواند گسترده‌تر باشد و با سؤالات در طیف وسیعی از موضوعات و حوزه‌ها سر و کار داشته باشد.
۳. **تکنیک‌ها:** مدل‌های *MRC* اغلب از تکنیک‌های پردازش زبان طبیعی مانند استخراج اطلاعات (*Information Extraction*)، تشخیص ارجاع (*Coreference Resolution*) و درک معنایی (*Semantic Understanding*) برای درک متن و پاسخگویی به سؤالات استفاده می‌کنند. سیستم‌های *QA* ممکن است علاوه بر این، از تکنیک‌هایی مانند بازیابی اطلاعات (*Information Retrieval*)، پرس و جو از پایگاه‌های دانش (*KB Querying*) و سایر روش‌ها برای جمع‌آوری اطلاعات مرتبط از منابع متعدد استفاده کنند.
۴. **کاربردها:** *MRC* به ویژه برای وظایفی که نیازمند درک عمیق محتوای متنی هستند، مانند پاسخگویی به سؤالات بر اساس اسناد، خلاصه‌سازی و استخراج دانش مفید است. *QA* کاربردهای گسترده‌تری دارد، از جمله دستیاران مجازی، موتورهای جستجو و سیستم‌های هوش مصنوعی گفتگویی.

## سوال سوم

در حوزه پرسش و پاسخ (QA)، سؤالات را می‌توان به طور کلی به دو دسته تقسیم کرد: سؤالات حقیقی و سؤالات غیرحقیقی.

سؤال حقیقی نوعی سؤال است که به دنبال یک واقعیت مختصر به عنوان پاسخ است. این سؤالات معمولاً به دنبال اطلاعات خاصی مانند نام‌ها، تاریخ‌ها یا اعداد هستند. به عنوان مثال، «رئیس جمهور فعلی ایالات متحده چه کسی است؟» یا «پایتخت فرانسه کدام شهر است؟» نمونه‌هایی از سؤالات حقیقی هستند. به عبارتی می‌توان گفت که این سؤالات با «چی»، «کی»، «چه کسی» و «کجا» آغاز می‌شوند. پاسخ‌های این سؤالات عینی، واقعی و قابل تأیید از یک منبع معتبر هستند.

از سوی دیگر، سؤال غیرحقیقی نوعی سؤال است که پاسخ واحد و قابل تأیید مبتنی بر واقعیت ندارد. این سؤالات اغلب نیازمند درک ظریف‌تر و زمینه‌ای هستند. سؤالات غیرحقیقی ممکن است شامل نظرات، توضیحات یا تفسیرهای ذهنی باشند که آنها را برای سیستم‌های سنتی پرسش و پاسخ چالش‌برانگیزتر می‌کند. نمونه‌هایی از سؤالات غیرحقیقی عبارتند از «چرا آسمان آبی است؟» یا «یک رایانه چگونه کار می‌کند؟». بنابراین این دسته از سؤالات با کلماتی مانند «چرا» و «چگونه» آغاز می‌شوند. پاسخ‌های این سؤالات معمولاً پیچیده‌تر هستند و ممکن است نیازمند استدلال، توضیح یا تحلیل باشند.

## سوال چهارم

ترنسفورمرها و شبکه‌های عصبی بازگشتی ( $RNN$ )ها دو نوع مدل مورد استفاده در پردازش داده‌های توالی هستند. در حالی که  $RNN$ ها داده‌ها را به صورت ترتیبی پردازش می‌کنند، ترنسفورمرها داده‌ها را به صورت موازی پردازش می‌کنند و روابط بین تمام عناصر در یک توالی را به طور همزمان ردیابی می‌کنند.

مزایا ترنسفورمرها در مقایسه با  $RNN$ ها:

- پردازش موازی: ترنسفورمرها تمام ورودی‌ها را به صورت موازی پردازش می‌کنند، که آنها را به طور قابل توجهی سریع‌تر از  $RNN$ ها و  $LSTM$ ها می‌کند که ماهیتاً ترتیبی هستند. این قابلیت موازی‌سازی باعث تسریع در آموزش و امکان استفاده از اندازه‌های بچ بزرگتر می‌شود که آموزش را کارآمدتر می‌کند.
- مدیریت طول توالی: ترنسفورمرها می‌توانند به دلیل ماهیت پردازش موازی خود، توالی‌های کوتاه و بلند را به طور کارآمد مدیریت کنند. خودتوجهی به آنها امکان می‌دهد تا وابستگی‌ها را صرف نظر از طول توالی ردیابی کنند.
- مدل‌سازی وابستگی: ترنسفورمرها در مدل‌سازی وابستگی‌های بین عناصر، صرف نظر از موقعیت آنها در توالی، شایسته هستند. آنها به ویژه برای وظایفی که درگیر وابستگی‌های بلندمدت هستند، مانند ترجمه ماشینی، طبقه‌بندی اسناد و توصیف تصاویر، قدرتمند هستند. این در حالی است که در مدل‌های  $RNN$  وابستگی بین ورودی‌ها به صورت خطی در نظر گرفته می‌شود که با گذشت زمان وابستگی دو واژه نسبتاً دور از همدیگر از بین می‌رود.
- قابلیت تفسیر: اگرچه ترنسفورمرها ممکن است به دلیل مکانیزم‌های خودتوجهی‌شان، تفسیر چالش‌برانگیزتری داشته باشند، اما تکنیک‌هایی مانند مصورسازی توجه ( $Attention Visualization$ ) می‌تواند بینشی خوب در عملکرد مدل ارائه دهد.

معایب ترنسفورمرها در مقایسه با  $RNN$ ها:

- پیچیدگی محاسباتی: مکانیزم خودتوجهی در ترنسفورمرها منجر به پیچیدگی محاسباتی  $O(N^2)$  می‌شود که می‌تواند برای توالی‌های طولانی در مقایسه با  $RNN$ ها پرهزینه باشد.
- اندازه مدل: ترنسفورمرها به دلیل معماری خود که شامل لایه‌های متعدد خودتوجهی و شبکه‌های پیش‌رو است، تمایل به داشتن اندازه مدل بزرگتری دارند. این اندازه بزرگتر منجر به تعداد پارامترهای بیشتری می‌شود که ترنسفورمرها را در مقایسه با  $RNN$ ها پرمصرف‌تر می‌کند.
- کارایی بر روی داده‌های کوچکتر: همانطور که گفته شد در حالی که ترنسفورمرها به طور کلی عملکرد بهتری نسبت به  $RNN$ ها دارند، معمولاً برای دستیابی به عملکرد مشابه نیاز است که عمیق‌تر یا بزرگتر از مدل  $LSTM$  مربوطه باشند. این می‌تواند  $LSTM$ ها را برای کاربردهایی با داده‌های کوچک‌تر یا جایی که فشرده‌سازی مدل اولویت دارد، مناسب‌تر کند.

## سوال پنجم

در مدل‌های ترنسفورمر به دلیل استفاده از *Self-Attention* ها نیاز است که نحوی تمایز میان توزیع کلمات در جملات مختلف را مشخص کرد. چرا که در این مدل‌ها بر خلاف مدل‌های ترتیبی مثل *LSTM* ترتیب ورود لغات توسط مدل یاد گرفته نمی‌شود. به عبارتی با توجه به ذات موازی‌سازی‌ای که در این مدل‌ها است تمامی داده‌های ورودی را به صورت همزمان پردازش می‌کند. برای نشان دادن این گفته، یک مثال برای یک مدل *Self-Attention* ساده می‌زنیم. (این مثال در کتاب مرجع آورده شده است.) برای مثال جمله زیر را در نظر می‌گیریم:

*The oven cooked the bread so*

در نتیجه ورودی به صورت زیر تعریف می‌شود:

$$x_{1:n} = [x_{the}; x_{oven}; x_{cooked}; x_{the}; x_{bread}; x_{so}] \in \mathbb{R}^{5 \times d}$$

حال نشان می‌دهیم که در صورت به هم ریختن ورودی، و تغییر ترتیب آن، هیچ تغییری در وزن اختصاص یافته به هر یک از کلمات به وجود نخواهد آمد و در نتیجه احتمال انتخاب واژگان مستقل از محل قرارگیری آن‌ها در جملات خواهد شد.

$$\alpha_{so,o} = \frac{\exp(q_{so}^T k_{the})}{\exp(q_{so}^T k_{the}) + \dots + \exp(q_{so}^T k_{bread})}$$

به همین ترتیب برای باقی کلمات نیز وزن دوتایی مربوط به واژه *so* و آن را حساب کرده و در پایان با استفاده از رابطه زیر بردار *Embedding* مربوط به هر یک را حساب می‌کنیم:

$$h_i = \sum_{j=1}^n \alpha_{ij} v_j$$

حال در صورتی که ترتیب ورودی مسئله را تغییر دهیم، با توجه به ثابت ماندن مقادیر  $\alpha$  و  $v$ ، *Embedding* نهایی بدست آمده نیز ثابت باقی خواهد ماند.

بنابراین نیاز است که به یکی از دو نحو زیر، ترتیب ورودی‌ها را در *Embedding* نهایی بدست آمده تاثیر بدهیم:

۱. استفاده از بردارهای از پیش تعریف شده، بر اساس اندیس قرارگیری یک هر کلمه.
۲. تغییر دادن عملیات *Self-Attention* به نحوی که اهمیت به ترتیب را در خود به صورت ذاتی داشته باشد. به عنوان مثال تعریف مربوط به بدست آوردن وزن‌ها را تغییر دهیم.

که در مقاله چاپ شده در سال ۲۰۱۷ که در آن مدل ترنسفورمر، معرفی شد از نوع اول و با استفاده از تعریف کسینوس و سینوس، استفاده شد.

در نتیجه استفاده از *Positional Embedding*، رابطه بدست آمده برای *Embedding* مربوط به هر لغت به صورت زیر خواهد شد:

$$\tilde{x}_i = P_i + x_i$$

## سوال ششم

سه پیکربندی اصلی برای معماری‌های مطرح در زمینه ترنسفورمرها به شرح زیر هستند:

مدل‌های *Encoder-Only*:

- کارکرد: یک توالی ورودی را پردازش می‌کنند و یک نمایش با طول ثابت (*Embedding*) از ورودی را تولید می‌کنند. این *Embedding* می‌تواند برای انواع وظایف دیگر که پس از استفاده از این مدل مطلوب هستند که انجام شوند، مانند طبقه‌بندی، خوشه‌بندی و جستجو، استفاده شود.
- موارد استفاده: این مدل‌ها به ویژه برای وظایفی که نیازمند درک و تجزیه و تحلیل متن هستند، مانند طبقه‌بندی متن، تجزیه و تحلیل احساسات و شناسایی نامهای خاص، مؤثر هستند. نمونه‌هایی از آنها عبارتند از *BERT* (نمایش‌های کدگذار دوطرفه از ترانسفورمرها) و ترانسفورمر بینایی (*ViT*) برای طبقه‌بندی تصویر.
- ویژگی‌ها: این مدل‌ها از توجه دوطرفه استفاده می‌کنند، و برای درک جامع توالی ورودی، کانتکست را از هر دو جهت (توکن‌های گذشته و آینده) در نظر می‌گیرند. همانطور که گفته شد، این مدل‌ها برای درک و تولید *Embedding*‌های متن ورودی طراحی شده‌اند و برای تولید متن استفاده نمی‌شوند.

مدل‌های *Decoder-Only*:

- کارکرد: این مدل‌ها، با پیش‌بینی توکن بعدی در یک توالی، با در نظر گرفتن توکن‌های قبلی، متن تولید می‌کنند. این فرآیند، رمزگشایی خودبازگشتی (*Autoregressive Decoding*) نامیده می‌شود.
- موارد استفاده: این مدل‌ها در وظایف تولید متن، مانند مدل‌سازی زبان، تکمیل متن و سیستم‌های گفتگو، عملکرد خوبی دارند. نمونه‌هایی از آنها شامل مدل‌های *GPT* مانند *GPT-2* و *GPT-3* هستند.
- ویژگی‌ها: این مدل‌ها از توجه یک طرفه (علی) استفاده می‌کنند، یعنی هنگام تولید توکن بعدی، فقط توکن‌های گذشته را در نظر می‌گیرند. مدل‌های *Decoder-Only* از نظر معماری ساده‌تر از مدل‌های *Encoder-Decoder* هستند و برای تولید متن منسجم و مرتبط با کانتکست بسیار مؤثر عمل می‌کنند.

مدل‌های *Encoder-Decoder*:

- کارکرد: این مدل‌ها، از دو بخش تشکیل شده‌اند: یک کدگذار که توالی ورودی را پردازش می‌کند و یک رمزگشا که توالی خروجی را تولید می‌کند. کدگذار یک *Embedding* از ورودی ایجاد می‌کند که رمزگشا از آن برای تولید توالی خروجی یک توکن در هر بار استفاده می‌کند.
- موارد استفاده: این مدل‌ها برای وظایفی که توالی خروجی متفاوت از توالی ورودی است، مانند ترجمه ماشینی، خلاصه سازی متن و زیرنویس گذاری تصاویر، ایده آل هستند. نمونه‌هایی از آنها شامل مدل‌های *T5* (ترانسفورمر انتقال متن به متن) و *BART* (ترانسفورمرهای دوطرفه و خودبازگشت) هستند.
- ویژگی‌ها: این مدل‌ها از مکانیسم‌های توجه متقابل (*Cross Attention*) استفاده می‌کنند که به رمزگشا امکان می‌دهد در حین تولید خروجی، بر بخش‌های مرتبط توالی ورودی تمرکز کند. مدل‌های کدگذار-رمزگشا چندمنظوره هستند و می‌توانند وظایف پیچیده‌ای را که نیازمند تولید یک توالی از توالی دیگر است، انجام دهند. هر چند که به دلیل پیچیدگی این دسته از مدل‌ها، بیشتر به جای استفاده از آن‌ها، از مدل‌های فقط-کدگذار استفاده می‌شود.

## سوال هفتم

## پرسش و پاسخ استخراجی (Extractive Question Answering)

پرسش و پاسخ استخراجی شامل شناسایی و استخراج یک بازه از متن از یک سند داده شده است که به طور مستقیم به سوال پاسخ می‌دهد. این رویکرد به بازایی اسناد مرتبط و سپس استخراج متن دقیق که شامل پاسخ است، متکی است. این فرآیند معمولاً شامل دو جزء اصلی است:

۱. باز یاب (*Retriever*): این جزء در یک مجموعه بزرگ از اسناد جستجو می‌کند تا آنهایی را که با پرسش مرتبط‌ترین هستند را پیدا کند. اغلب از تکنیک‌هایی مانند مدل‌های زبان ماسک شده برای یافتن اسناد نامزد بر اساس شباهت کدگذاری‌ها استفاده می‌کند.
۲. خواننده (*Reader*): این جزء اسناد یافت شده را پردازش می‌کند تا بازه متنی خاصی را که به سوال پاسخ می‌دهد، پیدا کند. *Reader* معمولاً یک مدل پرسش و پاسخ است که بازه متنی دقیق را در سند شناسایی می‌کند که احتمالاً پاسخ است.

سیستم‌های پرسش و پاسخ استخراجی از این جهت مزیت دارند که پاسخ‌های دقیق را مستقیماً از متن منبع ارائه می‌دهند و اطمینان حاصل می‌کنند که اطلاعات دقیق و در زمینه مرتبط است. با این حال، آنها محدود به عبارات دقیق موجود در اسناد هستند و نمی‌توانند جملات جدید تولید کنند یا اطلاعات را بازگویی کنند.

## پرسش و پاسخ انتزاعی (Abstractive Question Answering)

پرسش و پاسخ انتزاعی، از طرف دیگر، پاسخ‌ها را در قالب زبان طبیعی‌تر و انعطاف‌پذیرتر تولید می‌کند. به جای استخراج بازه‌های متنی دقیق، سیستم‌های پرسش و پاسخ انتزاعی جملات جدیدی را که لزوماً در اسناد منبع وجود ندارند، ترکیب می‌کنند. این رویکرد شبیه به نحوه پاسخگویی انسان‌ها به سوالات از طریق خلاصه کردن و بازگویی اطلاعات است.

سیستم‌های پرسش و پاسخ انتزاعی معمولاً از مدل‌های کدگذار-کدگشا استفاده می‌کنند که می‌توانند هم داده‌های ساختاریافته جدولی و هم داده‌های متنی بدون ساختار را پردازش کنند. این مدل‌ها برای درک زمینه و تولید پاسخ‌های منسجم و مرتبط با زمینه طراحی شده‌اند. به عنوان مثال، یک سیستم پرسش و پاسخ انتزاعی آگاهی بخشیده در حوزه پزشکی می‌تواند با ادغام دانش تخصصی حوزه، پاسخ‌های دقیق و مختصر تولید کند.

مزیت اصلی پرسش و پاسخ انتزاعی توانایی آن در ارائه پاسخ‌های طبیعی‌تر است. با این حال، پیاده سازی آن به دلیل پیچیدگی تولید جملات صحیح از نظر گرامری و دقیق از نظر معنایی، چالش برانگیزتر است. علاوه بر این، خطر تولید اطلاعات غلط که ممکن است در اسناد منبع وجود نداشته باشد، وجود دارد.

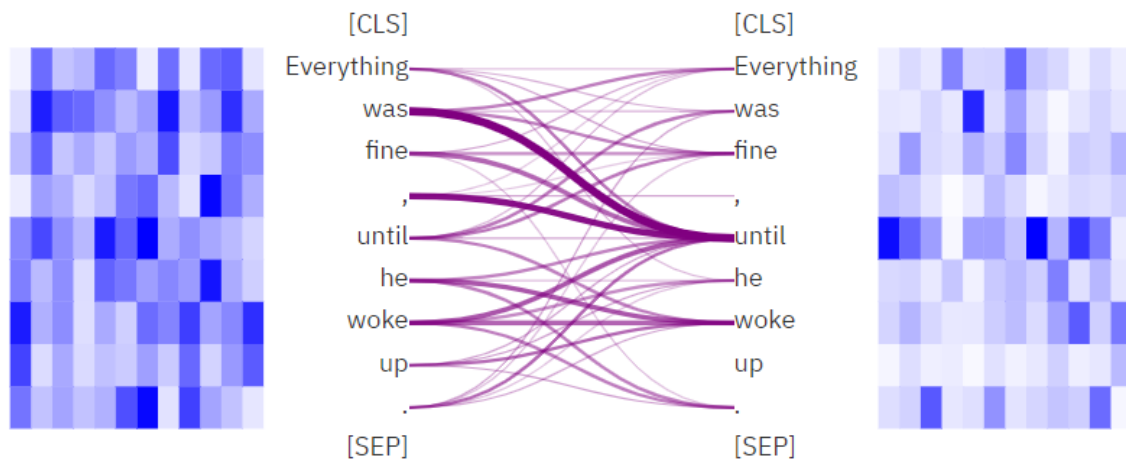
## سوال هشتم

## بخش اول

متن داده شده به صورت زیر است:

*Everything was fine, until he woke up.*

با استفاده از این متن، نتیجه زیر را از ابزار مورد نظر دریافت می‌کنیم.



## بخش دوم

تسک انتخاب شده، تسک ترجمه زبان یا *Machine Translation* است. این تسک به طور ذاتی پیچیده است زیرا نیاز به درک و ترجمه نه تنها کلمات جداگانه بلکه کانتسکت و مفهومی که در آن استفاده می‌شوند، از جمله عبارات اصطلاحی، ارجاعات فرهنگی و ساختارهای جمله متفاوت نیز دارد.

حال به این سوال که چرا *Multi-head Attention* برای این تسک مفید است، می‌پردازیم.

۱. پردازش موازی جنبه‌های مختلف زبان: توجه چندگانه به مدل اجازه می‌دهد تا به جنبه‌های مختلف توالی ورودی به طور موازی توجه کند. به عنوان مثال، یک سر ممکن است بر روی ساختارهای نحوی، یکی دیگر بر روی معنایی و دیگری بر روی نشانه‌های زمینه‌ای تمرکز کند. این توانایی پردازش موازی برای ترجمه زبان، جایی که درک ماهیت چندوجهی متن برای ترجمه دقیق ضروری است، بسیار مهم است.

۲. افزایش ظرفیت و قدرت بیان مدل: با داشتن چندین سر توجه، مدل می‌تواند مجموعه غنی‌تری از روابط درون متن را درک کند. هر سر استفاده شده می‌تواند یاد بگیرد که بر روی بخش‌های مختلف توالی ورودی تمرکز کند، که به مدل این امکان را می‌دهد تا وابستگی‌های در فاصله کم و زیاد را درک کند. این به ویژه در زبان‌هایی با ترتیب کلمات انعطاف پذیر یا جایی که کانتسکت می‌تواند به طور قابل توجهی معنای کلمات یا عبارات را تغییر دهد، مهم است.



۳. بهبود عملکرد و کارایی: تحقیقات نشان داده است که توجه چندگانه منجر به عملکرد بهتری نسبت به استفاده از یک سر می‌شود، حتی اگر هر دو اندازه یکسان داشته باشند. این به این دلیل است که سرهای چندگانه می‌توانند توجه را چندین بار به طور موازی انجام دهند، که منجر به محاسبات کارآمدتر و بهبود عملکرد مدل می‌شود.
۴. پایداری آموزش: پیشنهاد شده است که توجه چندگانه مزایایی در پایداری آموزش در مقایسه با توجه تک سر دارد. این پایداری به ساختار کم عمق تر معماری چندگانه نسبت داده می‌شود که برای توجه به همان تعداد موقعیت به تعداد لایه کمتری نسبت به یک مدل تک سر عمیق تر نیاز دارد. پایداری آموزش برای وظایف پیچیده ای مانند ترجمه زبان، جایی که مدل ها باید از داده های عظیم و ساختارهای زبانی متنوع یاد بگیرند، بسیار مهم است.

## سوال نهم

توضیحات به صورت کامنت در نوتبوک آورده شده است.