



به نام خدا
دانشگاه تهران
دانشکده مهندسی برق و کامپیوتر



درس شبکه‌های عصبی و یادگیری عمیق

تمرین اول

نام و نام خانوادگی	محمد ناصری – مریم عباس‌زاده
شماره دانشجویی	810100406 – 810100486
تاریخ ارسال گزارش	۱۴۰۱.۱۰.۱۴

فهرست

- پاسخ 1. آشنایی با مفهوم توجه و پیاده‌سازی مدل BERT 1
- ۱-۱. پیاده‌سازی کدگذار 1
- ۲-۱. پیاده‌سازی مدل BERT 1
- پاسخ ۲ - آشنایی با کاربرد تبدیل‌کننده‌ها در تصویر 3
- ۲-۲. تقسیم بندی معنایی تصاویر 3
- ۳-۲. طبقه بندی تصاویر 4
- ۴-۲. پرسش‌ها 8
- درست و غلط 9

شکل‌ها

- شکل 1 - تصاویر تصادفی انتخاب شده 3
- شکل 2 - تصاویر mask شده با تقسیم بندی معنایی 4
- شکل 3- تصاویر mask موجود در دیتاست 4
- شکل 4- نمونه تصاویر موجود در دیتاست به همراه label 5
- شکل 5 - مدل ساده MLP تعریف شده 5
- شکل 6 - نمودار loss برای مدل mlp 5
- شکل 7 - نمودار آشفته‌گی برای mlp 6
- شکل 8 - نمونه خروجی و label توسط مدل تعریف شده 7
- شکل 9 - نمودار آشفته‌گی برای BEiT 8

جدول‌ها

جدول 1 - جدول نتایج معیارها برای mlp 6

جدول 2 - مقادیر نهایی معیار برای BEiT 6

پاسخ ۱. آشنایی با مفهوم توجه و پیاده‌سازی مدل BERT

۱-۱. پیاده‌سازی کدگذار

1. مفهوم توجه: در زمینه شبکه‌های عصبی، توجه مکانیزمی است که به مدل اجازه می‌دهد در هنگام پردازش آن بر بخش‌های خاصی از ورودی تمرکز کند. این ویژگی می‌تواند زمانی مفید باشد که ورودی یک دنباله طولانی باشد، زیرا به مدل اجازه می‌دهد تا به طور انتخابی بر روی بخش‌هایی از دنباله که بیشترین ارتباط را با کار در دست دارند تمرکز کند، نه اینکه مجبور باشد کل دنباله را به طور مساوی پردازش کند. مکانیزم‌های توجه را می‌توان به روش‌های مختلفی اعمال کرد، اما یک رویکرد رایج استفاده از یک لایه توجه است که دنباله ورودی را می‌گیرد و یک نسخه وزن‌دار از ورودی تولید می‌کند، با وزن‌ها که نشان‌دهنده اهمیت هر عنصر در دنباله است. سپس این ورودی وزنی توسط مدل به جای دنباله ورودی اصلی استفاده می‌شود.

2. در BERT و سایر مدل‌های مبتنی بر ترانسفورماتور، از توجه multi-head به جای توجه single-head استفاده می‌شود، زیرا به مدل اجازه می‌دهد تا به چندین بخش مختلف ورودی به طور همزمان توجه کند، نه فقط یکی. این امکان می‌تواند به ویژه زمانی مفید باشد که ورودی حاوی چندین نوع مختلف روابط یا وابستگی بین عناصر باشد، زیرا headهای مختلف مکانیزم توجه می‌توانند بر جنبه‌های مختلف ورودی تمرکز کنند. یک مزیت کلیدی این است که به مدل اجازه می‌دهد تا محاسبات توجه را در سرهای مختلف موازی کند، که می‌تواند از نظر محاسباتی کارآمدتر باشد. علاوه بر این، استفاده از سرهای توجه متعدد می‌تواند به بهبود توانایی مدل برای یادگیری وابستگی‌های پیچیده در ورودی کمک کند، زیرا هر سر می‌تواند یاد بگیرد که بر روی جنبه‌ای متفاوت از ورودی تمرکز کند.

۲-۱. پیاده‌سازی مدل BERT

1. توضیح segment embedding: در BERT، segment embedding برای نمایش بخش‌بندی دنباله ورودی استفاده می‌شود. BERT برای پردازش دنباله‌های ورودی که از دو بخش تشکیل شده‌اند طراحی شده است و از segment embedding برای تمایز بین دو بخش در ورودی استفاده می‌شود. segment embedding قبل از عبور از رمزگذار ترانسفورماتور به جاسازی‌های نشانه ورودی اضافه

می شوند. علاوه بر استفاده از آنها در BERT، segment embedding می تواند در سایر وظایف پردازش زبان طبیعی که ورودی شامل چندین بخش است که باید جداگانه پردازش شوند، مانند ترجمه ماشینی یا خلاصه سازی متن، مفید باشد.

2. [پاسخ در نوت بوک]

پاسخ ۲ - آشنایی با کاربرد تبدیل‌کننده‌ها در تصویر

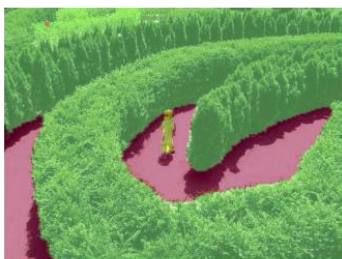
۲-۲. تقسیم‌بندی معنایی تصاویر

تقسیم‌بندی معنایی نوعی تقسیم‌بندی تصویر است که به هر پیکسل در یک تصویر برچسب‌هایی را بر اساس معنای معنایی آن اختصاص می‌دهد. برای شناسایی اشیاء و مرزها در یک تصویر استفاده می‌شود و می‌تواند برای ایجاد درک سطح بالایی از یک تصویر استفاده شود. برچسب‌های اختصاص داده شده به هر پیکسل با استفاده از الگوریتم‌های یادگیری ماشینی مانند شبکه‌های عصبی کانولوشن (CNN) یا ماشین‌های بردار پشتیبان (SVM) تعیین می‌شوند. سپس خروجی تقسیم‌بندی معنایی می‌تواند برای کارهایی مانند تشخیص اشیاء، طبقه‌بندی تصویر و ناوبری مستقل استفاده شود. Hugging Face یک کتابخانه منبع باز برای پردازش زبان طبیعی و وظایف بینایی رایانه است. این شامل طیف گسترده‌ای از مجموعه داده‌ها و مدل‌ها برای وظایف مختلف، از جمله تقسیم‌بندی معنایی است. به طور خاص، Hugging Face مدل‌های SegFormer و BEiT را برای تقسیم‌بندی معنایی ارائه می‌کند. SegFormer یک مدل یادگیری عمیق است که از یک شبکه عصبی کانولوشن برای تقسیم‌بندی تصاویر به مناطق معنایی استفاده می‌کند، در حالی که BEiT مدلی است که معماری رمزگذار-رمزگشا را با معماری U-Net ترکیب می‌کند تا تصاویر را به مناطق معنایی تقسیم کند. هر دو مدل برای ارائه نتایج تقسیم‌بندی با کیفیت بالا طراحی شده‌اند و برای کارهایی مانند ناوبری مستقل و تشخیص اشیاء مناسب هستند.

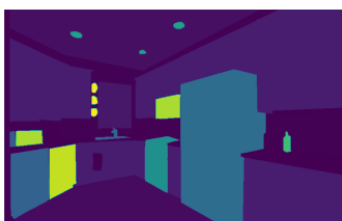
خروجی نهایی تصاویر:



شکل ۱ - تصاویر تصادفی انتخاب شده



شکل 2 - تصاویر mask شده با تقسیم بندی معنایی

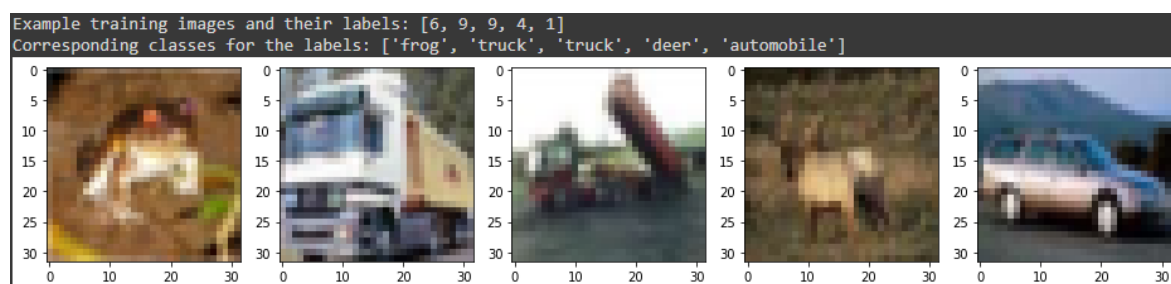


شکل 3- تصاویر mask موجود در دیتاست

۲-۳. طبقه بندی تصاویر

پرسپترون های چندلایه (MLPs) نوعی شبکه عصبی هستند که برای کارهای طبقه بندی تصاویر استفاده می شود. MLP ها معمولاً از چندین لایه نورون تشکیل شده اند که با اتصالات وزنی به هم متصل هستند. ورودی های MLP مقادیر پیکسل تصویر هستند و خروجی یک برچسب طبقه بندی است. MLP ها از یک الگوریتم آموزشی برای یادگیری وزن اتصالات بین نورون ها استفاده می کنند، که تعیین می کند چگونه مقادیر ورودی برای تولید برچسب خروجی پردازش شوند. MLP ها معمولاً برای کارهای طبقه بندی تصویر

استفاده می شوند، زیرا می توانند داده های تصویر ورودی را با دقت به برجسب طبقه بندی صحیح نگاشت کنند.



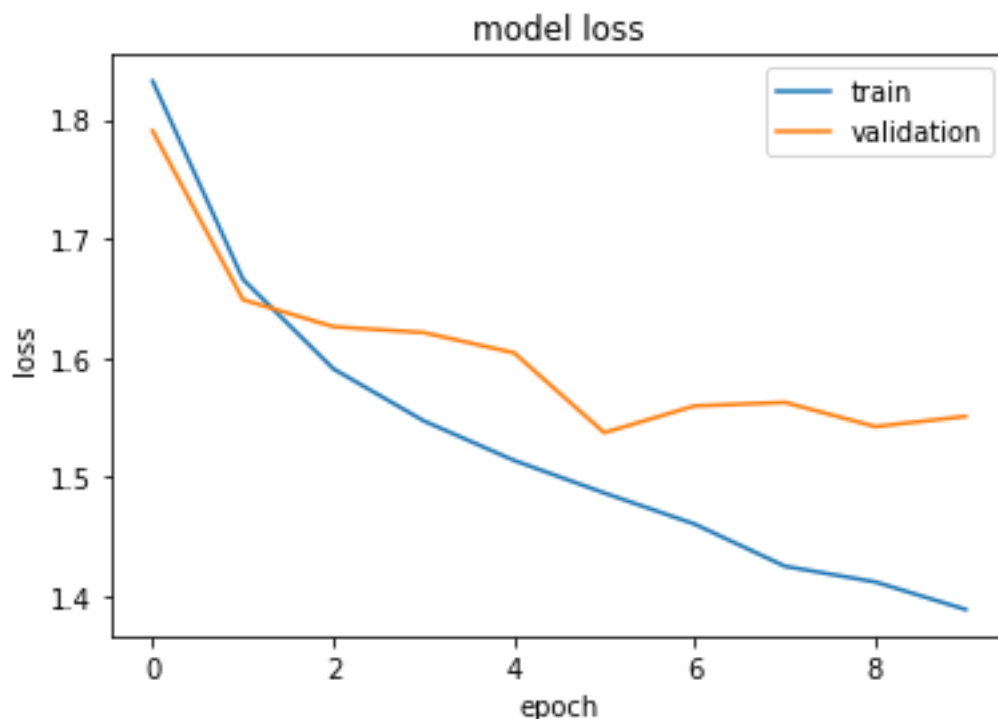
شکل 4- نمونه تصاویر موجود در دیتاست به همراه label

در مرحله اول یک مدل ساده mlp برای طبقه بندی تعریف کرده و اجرا میکنیم که مدل و نتایج به شرح زیر هستند:

```
model = Sequential()
model.add(Dense(256, activation='relu', input_dim=3072))
model.add(Dense(256, activation='relu'))
model.add(Dense(10, activation='softmax'))
sgd = SGD(lr=0.01, decay=1e-6, momentum=0.9, nesterov=True)

model.compile(optimizer=sgd, loss='categorical_crossentropy', metrics=['accuracy'])
```

شکل 5 - مدل ساده MLP تعریف شده

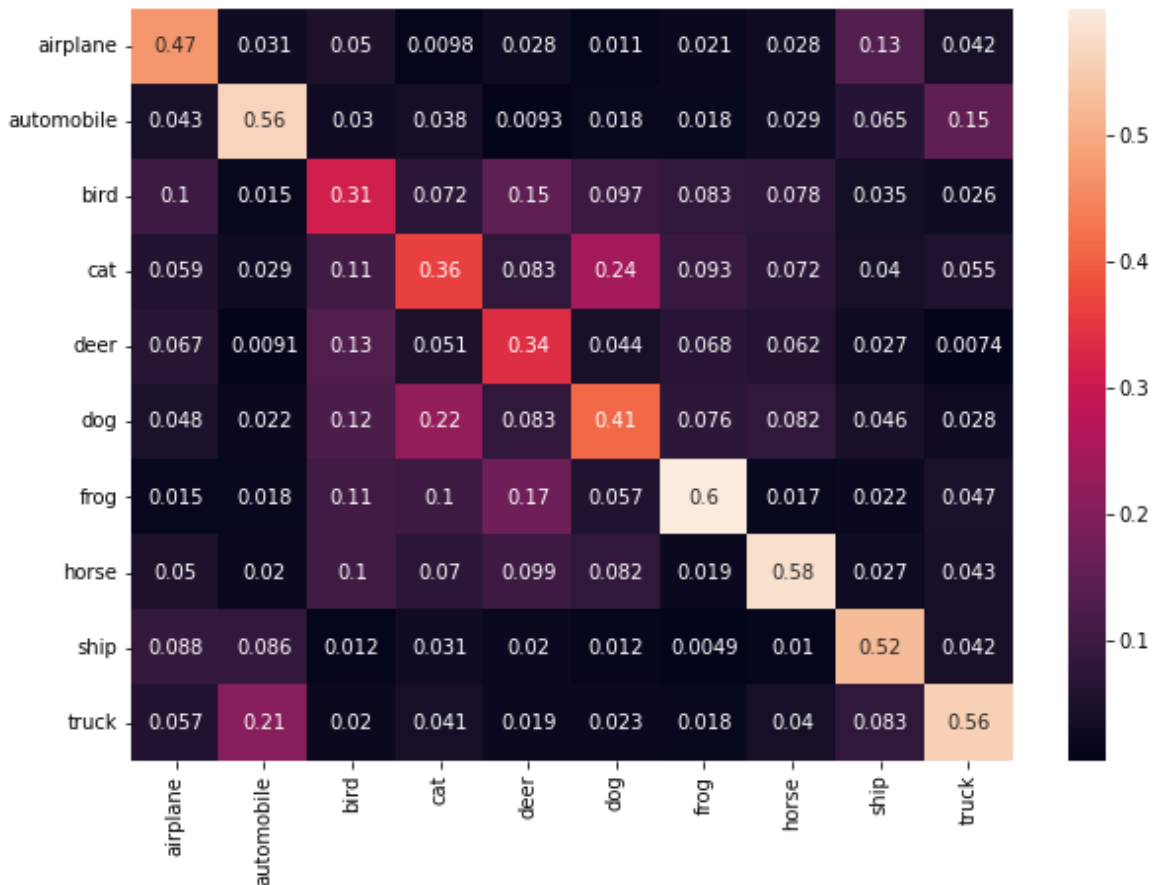


شکل 6 - نمودار loss برای مدل mlp

همچنین مقادیر معیارهای در نظر گرفته شده عبارتند از:

جدول 1 - جدول نتایج معیارها برای **mlp**

loss	accuracy
1.5082801580429077	0.4618000090122223



شکل 7 - نمودار آشفتگی برای **mlp**

در مرحله بعدی مدل BEiT را در یک لایه استفاده میکنیم و آموزش را انجام میدهم. نتایج برای این مدل طبق انتظار با fine tuned شدن بسیار با دقت بالاتری هستند و به صورت زیر میباشند:

جدول 2 - مقادیر نهایی معیار برای **BEiT**

Train_Loss	Val_Loss	Accuracy	F1
0.125300	0.056618	0.986900	0.986883

A: cat
P: cat



A: airplane
P: airplane



A: ship
P: ship



A: frog
P: frog



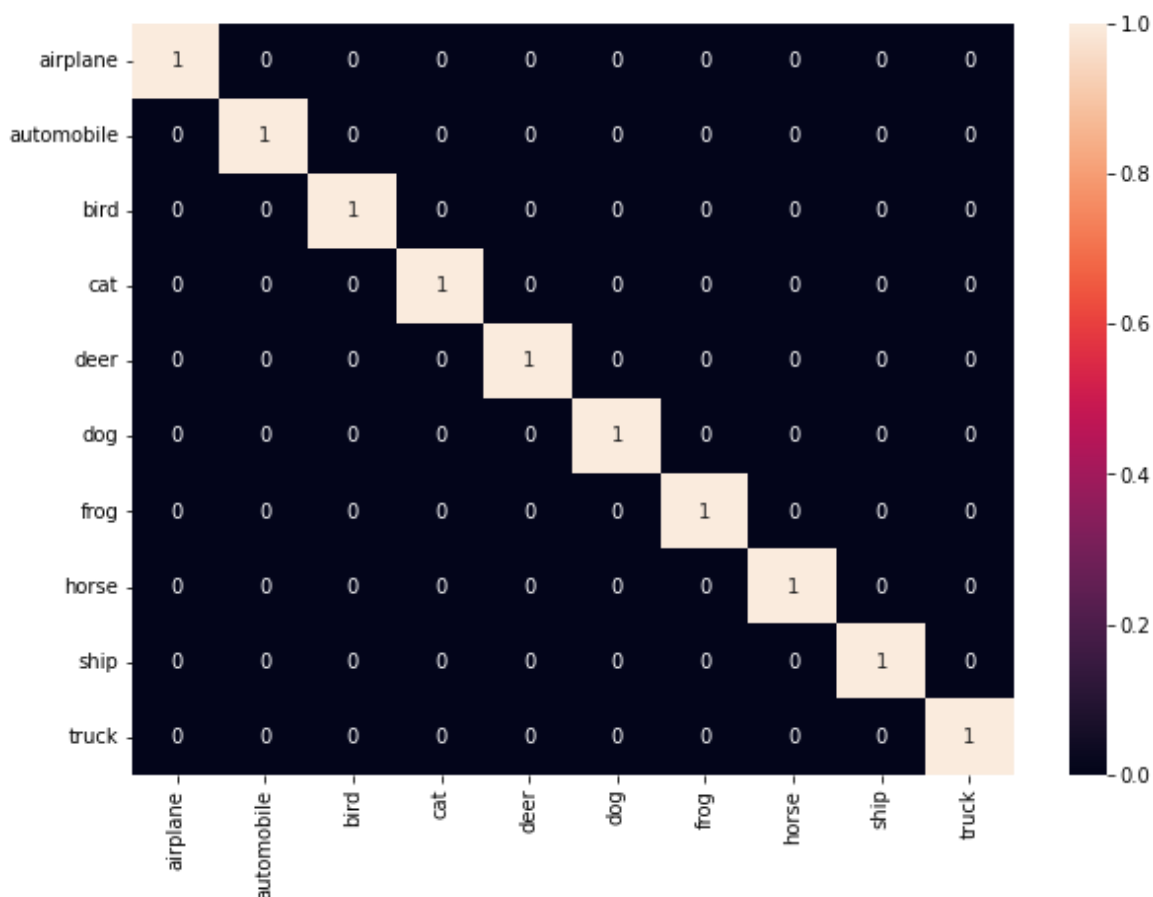
A: ship
P: ship



A: frog
P: frog



شکل 8 - نمونه خروجی و **label** توسط مدل تعریف شده



شکل 9 - نمودار آشفته‌گی برای BEiT

۴-۲. پرسش‌ها

1- در شبکه‌های CNN در کدام بخش مفهومی مانند مفهوم توجه اتفاق می‌افتد؟ مفهوم توجه

در معماری شبکه عصبی کانولوشن (CNN) در زمینه مکانیزم توجه به خود رخ می‌دهد. مکانیسم‌های خودتوجهی به شبکه اجازه می‌دهد تا بر بخش‌های خاصی از ورودی تمرکز کند و به آن اجازه می‌دهد تا زمینه ورودی داده شده را بهتر درک کند. مکانیسم توجه معمولاً در لایه‌های کانولوشنال شبکه پیاده‌سازی می‌شود و به آن اجازه می‌دهد تا بر روی ویژگی‌های مرتبط در تصویر ورودی تمرکز کند. این می‌تواند دقت پیش‌بینی‌های شبکه را بهبود بخشد و به آن اجازه دهد تا زمینه ورودی داده شده را بهتر درک کند.

2- در یک شبکه عصبی، در ارتباط یک لایه با لایه بعدی، چه تفاوتی میان یک شبکه ی

convolution با شبکه‌ی توجه همگانی و شبکه‌ی توجه محلی وجود دارد؟ در شبکه عصبی به ارتباط بین یک لایه و لایه بعدی انتقال می‌گویند. چندین نوع مختلف از لایه‌های انتقال وجود دارد که می‌توان از آنها استفاده کرد که هر کدام ویژگی‌ها و مناسب بودن خود را برای کارهای مختلف دارند.

شبکه کانولوشن نوعی شبکه عصبی است که برای پردازش داده ها با توپولوژی شبکه مانند تصویر طراحی شده است. در یک شبکه کانولوشن، لایه انتقال معمولاً از یک لایه کانولوشن تشکیل شده است، که یک تبدیل خطی به ورودی اعمال می کند و به دنبال آن یک غیرخطی از نظر عنصر (مانند یک واحد خطی اصلاح شده یا ReLU) اعمال می شود. لایه کانولوشن معمولاً توسط یک لایه ادغام دنبال می شود، که با گرفتن حداکثر (یا میانگین) مقدار در یک محله محلی، ورودی را پایین می آورد. شبکه توجه نوعی شبکه عصبی است که از مکانیزم توجه برای تمرکز انتخابی بر روی بخش های خاصی از ورودی هنگام پردازش آن استفاده می کند. انواع مختلفی از شبکه های توجه وجود دارد، از جمله شبکه های توجه جهانی و شبکه های توجه محلی. شبکه توجه جهانی نوعی از شبکه توجه است که به مدل اجازه می دهد در هنگام پردازش به تمام بخش های ورودی توجه کند. این می تواند زمانی مفید باشد که ورودی حاوی وابستگی های دوربرد یا روابط پیچیده بین عناصری باشد که باید توسط مدل گرفته شوند. از سوی دیگر، یک شبکه توجه محلی، نوعی از شبکه توجه است که به مدل اجازه می دهد در هنگام پردازش به پنجره محدودی از ورودی توجه کند. این می تواند زمانی مفید باشد که ورودی حاوی وابستگی ها یا روابط محلی باشد که گرفتن آن ها مهم تر است، زیرا به مدل اجازه می دهد در هر زمان بر بخش کوچک تری از ورودی تمرکز کند. به طور کلی، انتخاب نوع لایه انتقال برای استفاده در یک شبکه عصبی به ویژگی های داده های ورودی و وظیفه خاص در دست بستگی دارد. لایه های کانولوشن برای پردازش داده های شبکه مانند مناسب هستند، در حالی که شبکه های توجه برای کارهایی که نیاز به تمرکز انتخابی مدل بر روی بخش های خاصی از ورودی دارند، مفید هستند.

درست و غلط

- 1- نادرست است. LSTM (حافظه کوتاه مدت بلندمدت) نوعی شبکه عصبی تکراری است که برای ثبت وابستگی های طولانی مدت در داده های متوالی طراحی شده است. به عنوان بخشی از لایه های ترانسفورماتور وانیلی استفاده نمی شود.
- 2- درست است. یک تبدیل کننده معمولاً از چندین بلوک رمزگذار و چندین بلوک رمزگشا تشکیل شده است. در معماری تبدیل کننده، بلوک های رمزگذار برای پردازش توالی ورودی و تولید مجموعه ای از نمایش های پنهان که معنای ورودی را به تصویر می کشد، استفاده می شوند. سپس بلوک های رمزگشا برای تولید دنباله خروجی بر اساس نمایش های پنهان تولید شده توسط رمزگذار و دنباله خروجی مورد نظر استفاده می شوند.

3- درست است. یک لایه توجه چند سر معمولاً از یک بخش توجه و چندین لایه موازی کاملاً متصل تشکیل شده است. بخش توجه مسئول محاسبه وزن توجه برای هر عنصر در دنباله ورودی، بر اساس یک پرس و جو و مجموعه ای از کلیدها است. وزن توجه نشان دهنده اهمیت هر عنصر در دنباله با توجه به پرس و جو است. لایه‌های کاملاً متصل سپس برای تبدیل ورودی با وزن به خروجی نهایی لایه توجه استفاده می‌شوند. در یک لایه توجه چند سر، چندین لایه کاملاً متصل وجود دارد (از این رو "چند سر" نامیده می‌شود) که به صورت موازی عمل می‌کنند، که هر کدام نسخه متفاوتی از خروجی را تولید می‌کنند. سپس خروجی‌های لایه‌های مختلف کاملاً متصل به هم متصل می‌شوند و از یک لایه کاملاً متصل نهایی عبور می‌کنند تا خروجی نهایی لایه توجه چند سر تولید شود.

4- این گفته نادرست است. وجود رمزگذاری موقعیتی در معماری ترانسفورماتور حیاتی نیست و شبکه بدون آن لزوماً از کار نخواهد افتاد. رمزگذاری موقعیتی تکنیکی است که در معماری ترانسفورماتور برای ارائه اطلاعاتی در مورد موقعیت نسبی عناصر در توالی ورودی به مدل استفاده می‌شود. این می‌تواند برای کارهایی مانند مدل سازی زبان یا ترجمه ماشینی مفید باشد، جایی که ترتیب عناصر در ترتیب ورودی مهم است. با این حال، رمزگذاری موقعیتی همیشه ضروری نیست، و در برخی موارد، ممکن است اصلاً استفاده نشود. به عنوان مثال، در کارهایی که ترتیب عناصر در ترتیب ورودی مهم نیست، مانند طبقه بندی تصویر یا تشخیص اشیا، ممکن است از رمزگذاری موقعیتی استفاده نشود.

