کوئیز شماره ۴ – نام نام خانوادگی:

سوال ۱) با ذکر مثال دلیل استفاده از MultiHead attention بجای single head attention را بیان کنید؟ (۱۵ نمره)

I ate an apple.

کلمه apple از نظر نحوی مفعول فعل ate است که قاعدتا بیشترین attention (از جنبه نحوی) را به این کلمه دارد. از طرف دیگر کلمه apple یک کلمه مفرد است که با حرف a شروع شده است و از این نظر بیشترین attention را به کلمه an دارد. پس حداقل از دو جنبه بایستی attention ها در این مثال درنظرگرفته شود. multi head هدفش مدلسازی همین جنبه های مختلف است.

The different words in a sentence can relate to each other in many different ways simultaneously. For example, distinct syntactic, semantic, and discourse relationships can hold between verbs and their arguments in a sentence. It would be difficult for a single transformer block to learn to capture all of the different kinds of parallel relations among its inputs. Transformers address this issue with multihead self-attention layers. These are sets of self-attention layers, called heads, that reside in parallel layers at the same depth in a model, each with its own set of parameters. Given these distinct sets of parameters, each head can learn different aspects of the relationships that exist among inputs at the same level of abstraction.

سوال 2) هدف از بکارگیری layer normalization در بلاک های ترنسفورمری چیست؟ این لایه دقیقا چه کار میکند و چگونه نرمالسازی میکند؟ (۱۵ نمره)

Layer normalization (or layer norm) is one of many forms of normalization that can be used to improve training performance in deep neural networks by keeping the values of a hidden layer in a range that facilitates gradient-based training. Layer norm is a variation of the **standard score,** or z-score, from statistics applied to a single hidden layer.

سوال ۳) یکی از مزیت های مدل های ترنسفورمری نسبت به مدل های بازگشتی مبتنی بر RNN موازی سازی می باشد. فرایند موازی سازی در زمینه مدل های ترنسفورمری برای ساخت یک LM بصورت autoregressive را شرح دهید؟ موازی سازی در کدام فاز های training و test(inference) رخ می دهد؟ (۱۵ نمره)

As with the overall transformer, a self-attention layer maps input sequences (x1,...,xn) to output sequences of the same length (y1,...,yn). the computation performed for each item is independent of all the other computations. this means that we can easily parallelize training of such models, in other words each training item can be processed in parallel since the output for each element in the sequence is computed separately. In a training phase , we use teacher forcing . Recall that in teacher forcing, at each time step in decoding we force the system to use the gold target token from training as the next input xt+1, rather than allowing it to rely on the (possibly erroneous) decoder output ˆyt, this let us to compute output for each input separately to others so we can parallelize training .

Once trained, we can autoregressively generate novel text just as with RNNbased models. using a language model to incrementally generate words by repeatedly sampling the next word conditioned on our previous choices is called autoregressive generation or causal LM generation. During the test or inference phase, parallelization, a technique commonly employed during training, cannot be effectively utilized. As a result, the generation of text proceeds autoregressively, wherein each word is incrementally produced by sampling based on previous choices.

سوال۴) دو راه حل برای مدل کردن ترتیب کلمات (word order) در ترسنفورمر ها را شرح دهید. کدام روش گزینه ی مناسبتری می باشد؟ چرا؟ (۱۵ نمره)

One simple solution is to modify the input embeddings by combining them with positional embeddings specific to each position in an input sequence. For example, just as we have an embedding for the word fish, we'll have an embedding for the position 3. As with word embeddings, these positional embeddings are learned along with other parameters during training. To produce an input embedding that captures positional information, we just add the word embedding for each input to its corresponding positional embedding. (We don't concatenate the two embeddings, we just add them to produce a new vector of the same dimensionality.) A potential problem with the simple absolute position embedding approach is that there will be plenty of training examples for the initial positions in our inputs and correspondingly fewer at the outer length limit.

An alternative approach to positional embeddings is to choose a static function that maps integer inputs to real valued vectors in a way that captures the inherent relationships among the positions.That is, it captures the fact that position 4 in an input is more closely related to position 5 than it is to position 17. A combination of sine and cosine functions with differing frequencies was used in the original transformer work.

سوال ۵) روش های prompting مبتنی one-shot ، zero-shot و few-shot در in-context learning را توضیح داده ، تفاوت بین انها را بیان کنید و بگویید در کدام روش وزن های مدل بروزرسانی می‌شود. چگونه می توان این روش ها را بهبود داد. (۱۵ نمره)

از این تکنیک‌ها معمولا برای رسیدگی به سناریوهایی استفاده می‌شوند که در آن داده‌های آموزشی محدود یا بدون برچسب برای کلاس‌ها یا وظایف خاص در دسترس است و یا با هدف های دیگر همچون محدودیت های ناشی از fine-tuning مدلها . هدف این رویکردها تعمیم و تطبیق مدل‌ها با کلاس‌ها یا وظایف جدید با داده‌های آموزشی کم است. این تکنیک ها بدون استفاده از fine-tuning و هیچگونه بروزرسانی مدل بوده (مفهوم in-context بر عدم وجود اپدیت مبتنی بر گرادیان تاکید دارد.) و مدل فقط با استفاده از چند مثال به انجام تسک مربوطه می‌پردازد. همچنین در اینجا می‌توان با عنوان prompting نیز آن ها را عنوان کرد.

Zero-shot: the ability of model to do many tasks with no examples, and no gradient updates , like: Specifying the right sequence prediction problem or Comparing probabilities of sequences

One-shot: the ability of model to do many tasks with only a single labeled example per class

Few-shot: the ability of model to do many tasks with small number of examples per class

Limitations : Some tasks seem too hard for even large LMs to learn through prompting alone. Especially tasks involving **richer**, **multi-step reasoning**. *Complex tasks will probably need gradient steps , Limits to what you can fit in context prompt engineering (e.g. CoT) can improve performance of them*

سوال ۶) به سوالات زیر پاسخ کوتاه دهید. (۲۵ نمره – هر بند ۵ نمره)

- Order محاسباتی self-attention چقدر است؟ $o(n^2)$
- در مدلسازی LM بصورت autoregressive با استفاده از مدل ترنسفورمری ، هنگام محاسبه ی ماتریس Q*K که شباهت هر توکن نسبت به توکن های قبل را بیان می کند ، چه مقداری برای قسمت بالا مثلثی این ماتریس قرار می دهیم. چرا؟ -inf چون در آنصورت softmax انها صفر میشود که هدفمان همین است.
- در محاسبه ی attention، برای بدست اوردن امتیاز شباهت بین دو بردار query و key از dot-product استفاده می نماییم ، خروجی این معیار شباهت در چه بازه ای قرار می گیرد؟[-inf,+inf]
- میخواهیم دو تسک sentiment و summarization را انجام دهیم . به ترتیب از کدام مدل های BERT و GPT برای هر یک از تسک ها بهتر است که استفاده می کنید.

Sentiment: BERT

Summarization: GPT

- برای fine-tunning مدل BERT برای تسک طبقه بندی متن از کدام توکن آن به عنوان ورودی طبقه بند استفاده میکنیم؟ مکان ان توکن کجاست؟

CLS , first token of input sequence