به نام خدا

دانشگاه تهران

دانشکده مهندسی برق و کامپیوتر

یادگیری عمیق با کاربردها

تمرین شماره چهار

بهار ۱۴۰۴

نام و نام خانوادگی	امیرمحمد ابراهیمی نسب
شماره دانشجویی	۶۱۰۳۰۱۱۰۱
تاریخ ارسال گزارش	۱۴۰۴/۰۳/۱۹

فهرست گزارش

فهرست شکل ها

فهرست جدول ها

سوال ۱

سوال اول (۱۰ نمره)

طبق جدول داده شده، ما با یک ورودی با ابعاد (32, 4096, 768) طرف هستیم، حال ما ۶ هد توجه داریم پس بردار نهان بین ۶ تا تقسیم می‌شود، پس Q_i، V_i، K_i دارای ابعاد (32, 4096, 128) خواهند بود!

حال وارد هد تکی توجه میشویم:

رابطه ۱ $$Q \,.\, K^T : (32, 4096, 128) \times (32, 128, 4096) \rightarrow (32, 4096, 4096)$$

سپس اسکیل می‌شود، که ابعاد را تغییر نمیدهد(صرفا تقسیم بر رادیکال d می‌شود!) و بعد تابع softmax روی آن اعمال می‌شود که بازهم تغییری در ابعاد نخواهیم داشت. حال در V ضرب خواهد شد که خواهیم داشت:

رابطه ۲ $$Scores \,.\, V : (32, 4096, 4096) \times (32, 4096, 128) \rightarrow (32, 4096, 128)$$

پس ابعاد خروجی هد تکی توجه ما برابر است با: (32, 4096, 128)، که خب برای تک تک هدها به همین صورت است و درنهایت ۶ خروجی بدست آمده را بهم میچسبانیم و باز به ابعاد (32, 4096, 768) خواهیم برگشت!(البته سپس یک تبدیل خطی دیگر(projection) صورت میگیرد اکثر اوقات که ابعاد را تغییر نمی‌دهد!)

سپس بر روی خروجی هدچندگانه نرمال‌سازی و residual صورت میگیرد که ابعاد را تغییر نمیدهد، سپس وارد لایه‌های پسرو می‌شود، که طبق صورت سوال لایه اول ابعاد را نصف میکند پس ابعاد ما تبدیل می‌شود به (32, 4096, 384) لایه دوم تغییری نمیدهد و لایه سوم آنرا دو برابر میکند و به ابعاد (32, 4096, 768) برمیگرداند و باز هم نرمالسازی و residual صورت میگیرد!

پس در واقع بعد از عبور از ۶ لایه رمزنگار، ابعاد ورودی تغییری نمیکند در نهایت!

سوال دوم (۱۰ نمره)

پارامترها در لایه تعبیه:

واژگان ما ۱۰۰۰تاست هرکدام به بعد ۱۰۲۴ پس ۱۰۲۴۰۰۰تا برای تعبیه!

همچنین گفته شده بعد تعبیه ما ۱۰۲۴ اما نهان ما ۷۶۸ است پس یک تبدیل از ۱۰۲۴ به ۷۶۸ نیز داریم(که خب بایس هم داریم چون projection است.) پس داریم ۱۰۲۴*۷۶۸+۷۶۸=۷۸۷۲۰۰ پارامتر.

پارامترها در لایه رمزنگار:

هر رمزنگار دارای یک بلاک هد چندگانه توجه است که ۳ تا مقدار k,v,q دارد به همراه بایس و یک projection پس خواهیم داشت 2362368 = (768*768+768)*4 تا پارامتر.

هر رمزنگار دارای ۳ لایه پس رو است با ساختار توضیح داده شده پس میشود، 384+384*768 تا در لایه اول،384+384*384 در لایه دوم و 768+768*384 در لایه سوم بنابراین در مجموع ۷۳۸۸۱۶ تا پارامتر نیز در این زیرلایه داریم.

دو لایه نرمالساز داریم در هر رمزنگار و هر نرمالساز لایه‌ای دارای یک مقدار و یک بایس است به اندازه ابعاد پس یعنی داریم 2*768 برای هر لایه نرمالساز لایه ای که خب یعنی در مجموع خواهیم داشت ۳۰۷۲ تا پارامتر اینجا!

پس در مجموع در هر رمزنگار ما دارای ۳۱۰۴۲۵۶تا پارامتر هستیم و چون ۶ تا داریم پس در مجموع ۱۸۶۲۵۵۳۶ پارامتر داریم!

پارامترها در لایه رمزگشا:

در هر بلوک رمزگشا، ما دارای یک هد چندگانه توجه ماسک هستیم که دقیقا ساختار مشابه رمزنگار دارد، دارای هد چندگانه توجه کراس هستیم که دقیقا ساختار مشابه ای دارد پس بازهم همان مقدار

قبلی، همین حرف را برای لایه های پس رو نیز میشود زد، منتها تنها بخش متفاوت در تعداد لایه های نرمالساز است که اینجا ۳ است پس ۴۶۰۸تا پارامتر داریم به جای ۳۰۷۲ حالت رمزنگار!

پس در مجموع هر رمزگشا دارای ۵۴۶۸۱۶۰ پارامتر است و چون ۸ تا بلوک داریم پس ۴۳۷۴۵۲۸۰ پارامتر داریم برای بلوک های رمزگشا.

لایه خطی نهایی نیز بعد پنهان را به تعداد واژگان میبرد پس 769000 = 1000+1000*768 تا پارامتر هم اینجا خواهیم داشت.

نتایج را میتوانید در جدول زیر مشاهده نمایید:

جدول ۱ تعداد پارامترهای هر بلوک ساختار داده شده.

بلوک	تعداد پارامتر
تعبیه و تبدیل	**۱۸۱۱۲۰۰**
رمزنگار	**۱۸۶۲۵۵۳۶**
رمزگشا	**۴۳۷۴۵۲۸۰**
لایه خطی آخر	**۷۶۹۰۰۰**
مجموع	**۶۴۹۵۱۰۱۶**

سوال ۲

سوال اول (۳ نمره)

این مرحله بر روی داده‌های بسیار زیاد و متفاوت متن و حتی کد انجام می‌شود، که معمولا از اینترنت و ... تهیه شده است و یک پیکره بسیار بزرگی از داده است که گوناگونی زیادی دارند. که به منظور یادگیری کلی ساختار زبان و تولید متن صورت میگیرد، که معمولا از روش های خودنظارتی برای آموزش این بخش استفاده میکنند مثل پیش بینی کلمه بعدی، یا پنهان کردن برخی کلمات در یک بخش متن و پیش بینی آنها باتوجه به مابقی متن!

سوال دوم (۳ نمره)

چون قابلیت اجرای دستورات مختلف را ندارد!

طبق بخش قبل مدل صرفا به درک جایگذاری کلمه بعدی در یک متن، یا یک سری جای گذاری ها براساس متن رسیده است، اما قابلیت انجام یک کار مشخص که کاربر از آن میخواهد انجام دهد را ندارند! (به عنوان مثال ترجمه یک متن، یا خلاصه کردن آن و ...). توجه شود که مدل ممکن است مربوط به درخواست کاربر مطلبی تولید کند اما آن ساختار مدنظر کاربر را دارا نخواهد بود! که از دلایل آن میتوان به گستردگی مجموعه داده بدون هیچ محدودیتی و عدم تمرکز خاص بر روی یک مسئله واحد اشاره کرد!

سوال سوم (۹ نمره)

در این مرحله، مدل بر روی داده کوچکتر، با کیفیت تر، منظم تر و البته برچسب گذاری شده آموزش داده می‌شود که مخصوص انجام دادن کارهایی است که کاربر خواهان آن است، پس در این مرحله ما آموزش با نظارت داریم و تفاوت اصلی روی نوع داده است! یعنی ما در مرحله قبل میخواستیم مدل کلیت ساختار زبان آشنا شود، حال میخواهیم انجام دادن دستوراتی که انتظار داریم را یاد بگیرد برآورده کند.

دیتا شامل جفت پرامپت و جواب است(همان دستور و خروجی مربوط به آن) که به مدل یاد میدهد با توجه به درخواست و دستور ورودی چه خروجی باید داده شود، که ممکن است این دستورات متفاوت از هم باشند به عنوان مثال خلاصه نویسی، شعرسرایی، ترجمه و ...

این بخش در تعامل انسانی-مدل نقش بسیار پررنگی دارد، زیرا به مدل آموزش میدهیم چگونه از دستورات داده شده به آن پیروی کند و مرحله به مرحله آنها را انجام دهد و جواب های درست و مربوط به ورودی را به کاربر برگرداند.

سوال چهارم (۵ نمره)

هدف اصلی این مرحله این است که، مدل را بیشتر آموزش دهند تا کمک رسان، بی خطر و صادق باشد و به طور کلی خروجیکه می‌دهد باب میل انسان باشد، بتواند دستورات پیچیده تر را انجام دهد، جواب‌های تحریک کننده اشتباه و ناامن ندهد، تا جای ممکن صادق باشد و دچار توهم‌زایی نشود!

از رایج ترین روش ها استفاده از یادگیری تقویتی است با نظر انسان است (RLHF). در این روش مدل براساس یک‌سری ورودی چندین جواب تولید میکند سپس انسان‌ها این پاسخ‌ها را رتبه بندی

میکنند، که کدام را ترجیح میدهند و این پاسخ‌های بهتر برای آموزش یک مدل دیگر که به مدل جایزه معروف است داده میشوند که سعی میکند براساس ورودی و جواب تشخیص دهد امتیازی که انسان به آن جواب میدهد(کیفیت آن جواب از نظر انسان) و در نهایت ما fine-tune میکنیم مدل خود را با روش‌های یادگیری تقویتی(معمولا PPO) که مدل ما میشود عامل، پرامپت میشود فضای جست‌وجوی ما، جواب‌های تولید شده میشوند عمل‌های عامل ما و جایزه و امتیاز توسط مدل جایزه داده می‌شود. مدل یاد میگیرد که جواب‌هایی تولید کند امتیاز بیشتری کسب میکند تا بتواند جوابهایی تولید کند که باب میل انسان باشد!

سوال ۳

مجموعه داده

در این بخش همانطور که در سوال خواسته شده است، داده را دانلود کرده و ستون مشخص شده را استفاده میکنیم، چون مدل و مجموعه داده ما کوچک است به صورت سطح حرف جلو میرویم، به این معنا که مجموعه حروف ما میشوند مجموعه واژگان ما، به هرکدام از آنها یک عدد نسبت میدهیم و ۹۰ درصد آن پیکره را برای آموزش و ۱۰ درصد باقیمانده را برای اعتبارسنجی استفاده میکنیم!

برای آموزش بدین صورت عمل میکنیم که با توجه به اینکه درحال آموزش یا اعتبارسنجی هستیم یک دسته از پیکره مختص به آن جدا میکنیم و استفاده میکنیم که به صورت تصادفی نقطه شروع آن متن انتخاب میشود و به اندازه بازه توضیح داده شده(۳۲) از ادامه متن جدا شده و به اندازه دسته(۱۶) اینکار را انجام میدهیم و یک دسته بر میگردانیم!

تنظیمات اولیه پیشنهادی

برای طراحی و تنظیم مدل از ابرپارامترهای زیر استفاده شد:

جدول ۲ - ابرپارامترهای استفاده شده برای آموزش مدل

ابرپارامتر	مقدار
Seed	**16**
batch_size	**16**
Learning_rate	**1e-3**
n_embd	**64**
n_head	**4**

n_layer	**4**
Dropout_rate	**0.1**
eval_interval	**100**
max_iterations	**5000**
block_size	**32**

پیاده‌سازی ماژول‌های اصلی مدل (۲۵ نمره)

برای تعبیه ساز کاراکتر ما به یک لایه تعبیه با ابعاد (vocab_size, n_embd) نیاز داریم، برای تعبیه موقعیت مکانی ما به یک لایه به ابعاد (block_size, n_embd) نیاز خواهیم داشت که با استفاده از کتابخانه torch به راحتی قابل پیاده سازی است!

یک بلاک مبدل نیاز به یک هد چند توجه، لایه پسرو و نرمالایزر داریم، در این بلاک طبق مقاله ۱ در مراجع، ما از تکنیکی استفاده میکنیم که ابتدا ورودی را از یک لایه نرمالسازی عبور میدهیم سپس به هد چند توجه میدهیم سپس با خود مقدار ورودی جمع میکنیم(residual) سپس خروجی بدست آمده را از یک لایه نرمالساز دیگر عبور میدهیم و به لایه پس رو میدهیم و در نهایت با خود خروجی در مرحله قبل دوباره جمع میکنیم و این خروجی بلاک ما خواهد بود که برای آخرین بار از یک لایه نرمالساز دیگری عبورش میدهیم!

در لایه پسرو چون عمق تعبیه ما برابر با ۶۴ است ما در لایه پسرو ابتدا این عمق را ۴ برابر میکنیم و سپس برمیگردانیم به عمق قبلی آن!(کوچک کردن آن منطقی نیست زیرا اطلاعاتی را از دست میدهیم همچنین دقت پایین می‌آمد!) همچنین از یک لایه Dropout نیز برای جلوگیری از فرابرازش با نرخ ۰.۱ استفاده میکنیم.

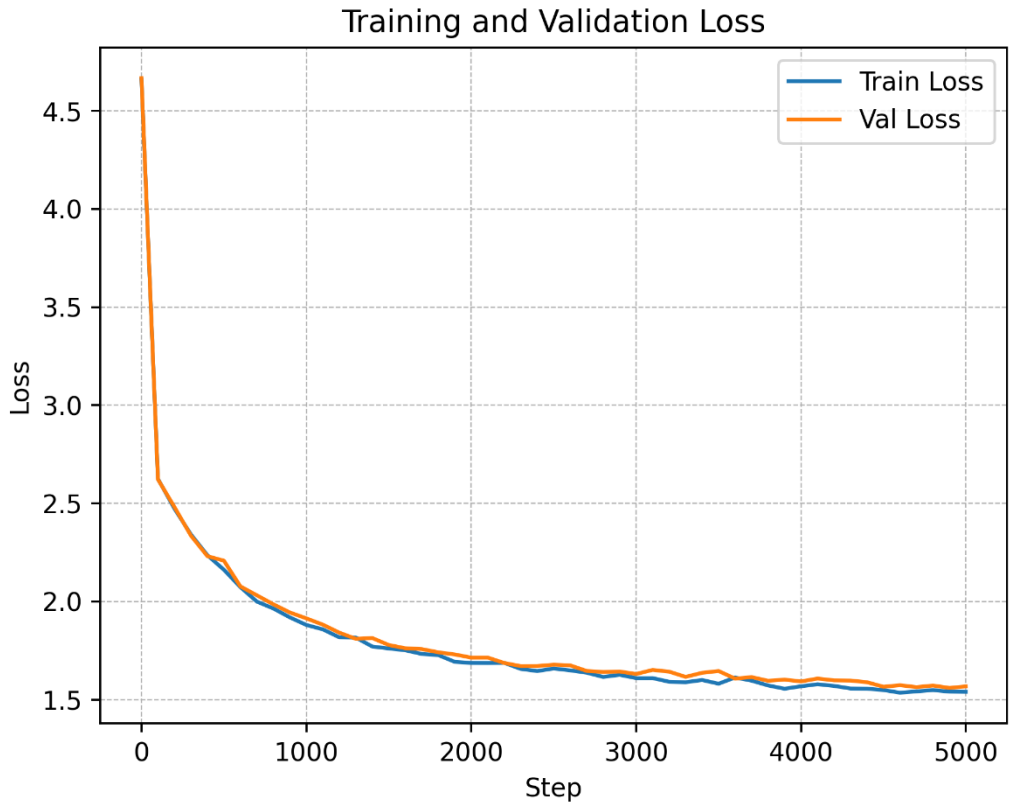
برای زیرلایه هد توجه چندگانه، ما نیاز داریم یک سر بسازیم و از آن به تعداد n_head در این بلوک قرار دهیم، یک لایه خطی(projection) نیز در انتهای آن قرار میدهیم و در انتهای آن نیز لایه Dropout!

و برای ساختار سر نیز دقیقا مشابه توضیحات در بخش ابتدایی گزارش و توضیحات سوال عمل میکنیم!

آموزش مدل (۱۵ نمره)

در این بخش کافیست دسته را طبق توضیحات اولیه گزارش استخراج کنیم و به مدل دهیم تا آموزش داده شود، بدیهی است که تابع هزینه استفاده شده cross_entropy است و از Adam به عنوان بهینه ساز

استفاده شد. در بازه‌های مشخص که با eval_interval مشخص شده اند خروجی را گزارش میکنیم که در تصویر زیر مشاهده می‌نمایید:



شکل ۱ - میزان اتلاف آموزش و اعتبارسنجی در حین فرآیند آموزش

همانطور که مشاهده میکنید دچار فرابرازش نشده ایم و مدل با روند نسبتا مناسبی آموزش داده شده است چه در آموزش و چه در اعتبار سنجی میزان اتلاف کاهش یافته است و در نهایت به ۱.۵۴ برای آموزش و ۱.۵۶ برای اعتبار سنجی رسیده ایم.

تولید متن (۲۰ نمره)

با استفاده از تابع generate در ساختار مدل خود به راحتی میتوانیم متن را تولید کنیم(به صورت ساختار autoregressive پیاده سازی شده است البته بدلیل کوچک بودن مدل و عدم اشاره در صورت سوال تمرین از eos استفاده نشد!). ایده کلی در این بهش بدین صورت است که بدلیل محدودیت ما در block_size هنگام آموزش ما به همان اندازه از متن تولید شده را میتوانیم به عنوان زمینه به مدل بدهیم مدل براساس این زمینه داده شده لاجیت های مربوط به هر حرف را برمیگرداند سپس ما با

استفاده از تابع softmax آنرا تبدیل به احتمال میکنیم و سپس با گرفتن توزیع بر حسب آن احتمال ها حرف بعدی را تولید میکنیم و در زمینه قرار میدهیم و هنگامی توقف میکنیم که به محدودیت حداکثر طول محتوا برسیم(چون با توجه به داده و مدل ما و عدم اشاره در تمرین از eos استفاده نشده است!) که این مقدار برابر با ۵۰۰ حرف تنظیم شده است.

در انتها سه نتیجه خروجی مدل را مشاهده خواهید کرد:

هنگامی که ورودی داده شده خالی است:

1. **Well, The storight, there, don't were. If'm what this watching it farked a too....chat ittin' an youriour. Holeeed a maselieve? And A undershing; oney, he's a good a lix! Whave wow is a talking. So, why don't wat whit worring. MDcIbome, so says. You know it there. A so one Chandler. So, eracher is you. [Scene: Monike Chakes Monica and you to This Bolnine ssame yeahops, Machel feels howsed too? I RYou hang cantally about at up and pliers' Hiss.. Okay, then It's great I was, don't want I betwent y**

هنگامی که ورودی داده شده :monica است:

2. **Monica: Remay! If a get to we lave? But tonight for from a still. I! Yeah, souy! We are starts. Phoebe? Don't gonna care which she talking it aftinger a *irtros. Dow, you guys! Ross, see so you great and away don't know apartment, entely loves you grab! No? What we just, preetieliel, I know, I know or, is Manachel we're enge would've ready weirdin't prebates you water. Yeahing. You love this. No. All right, you're ne. The cheter I was going to cleve to him is involing this chool of gonna as Ross, is my**

هنگامی که ورودی داده شده :Chandler است:

3. **Chandler:) "Eh, love I chinge? Oh, don't would really youh now cept of fivingling. [Scene: The Ross Joey pre-and-Mw! Hobe to you. 'Sustaugh insits and not? No! Who have the iosineff, the gonna beding me to.) Well, she's made thughoughting I don't prese from pool is a Mavilly are acteting, this I'm throuse, that's just told name. Oh smidgrol your redroom, you salle sreast she. Whate-you broofme to hote shaulf here spioushing that Chandler. A-yeah, you everyone there. Am firdsing Fonds here.] Hey. She's sh**

همانطور که میبینید مدل در تولید کلمات بامعنی عملکرد نسبتا مناسبی دارد(توجه شود که داریم حرف به حرف تولید میکنیم پس این عملکرد میتواند قابل قبول باشد)

همچنین وقتی اسم شخصیت‌های سریال را به عنوان ورودی به مدل میدهیم سعی میکند دیالوگی که توسط آن شخص گفته میشود تولید کند، اما خب ایراداتی مثل غلط املایی، کلمات نامفهوم، عدم رعایت دستورات زبانی که بدلیل کمبود زمان آموزش فضای ویژگی ها و تعبیه اولی و ... میتواند باشد!

مراجع

[1] “On Layer Normalization in the Transformer Architecture” here (accessed 6/6/2025).

[2] “Attention Is All You Need” here (accessed 5/6/2025).