گزارش تکلیف چهارم: فاین‌تیونینگ GPT-2 با LoRA روی دیتاست SQuAD

هیوا ابوالهادی‌زاده - ۴۰۰۴۰۵۰۰۴
درس: مباحث ویژه در هوش مصنوعی
استاد: جناب آقای دکتر افتخاری
دانشگاه باهنر کرمان

چکیده:
این گزارش به بررسی فاین‌تیونینگ مدل GPT-2 با استفاده از روش Low-Rank Adaptation (LoRA) روی دیتاست پاسخ به سؤالات استنفورد (SQuAD) می‌پردازد. LoRA به‌عنوان یک روش کارآمد برای کاهش تعداد پارامترهای قابل آموزش، امکان تطبیق مدل‌های بزرگ را با منابع محاسباتی محدود فراهم می‌کند. آزمایش‌ها شامل بررسی پیکربندی‌های مختلف LoRA (رتبه، نرخ یادگیری، و ماژول‌های هدف)، استراتژی‌های دی‌کدینگ، و تحلیل‌های تئوریک برای توضیح نتایج است. بهترین مدل (r=4، lr=1e-3، ماژول‌های توجه) به امتیاز F1 برابر 65.33 دست یافت، که نشان‌دهنده کارایی LoRA در مقایسه با فاین‌تیونینگ کامل است. تحلیل موارد شکست و مطالعه ابلیشن، بینش‌هایی درباره رفتار مدل ارائه می‌دهند، و مقایسه با فاین‌تیونینگ کامل، مزایای محاسباتی LoRA را برجسته می‌کند.

مقدمه و توضیح روش LoRA

۱.۱ معرفی LoRA

روش Low-Rank Adaptation (LoRA)، معرفی‌شده توسط Hu et al. (2021)، یک تکنیک فاین‌تیونینگ پارامتر_کارآمد است که به‌جای به‌روزرسانی تمام وزن‌های یک مدل پیش‌آموزش‌دیده، تغییرات کم‌رتبه‌ای را به ماتریس‌های وزنی لایه‌های خاص اعمال می‌کند. در مدل‌های ترنسفورمر مانند GPT-2، وزن‌های لایه‌های شبکه عصبی (مانند ماتریس‌های توجه) به‌صورت ماتریس‌های $W \in R^{d\times k}$ تعریف می‌شوند. LoRA فرض می‌کند که تغییرات مورد نیاز برای تطبیق مدل به یک وظیفه خاص (مانند پاسخ به سؤالات) را می‌توان با یک ماتریس کم‌رتبه $\Delta W = A \cdot B$ نشان داد، که در آن $A \in R^{d\times r}$ و $B \in R^{r\times k}$ هستند و $r \ll \min(d, k)$ رتبه کم‌رتبه است. این روش تعداد پارامترهای قابل آموزش را به‌طور قابل‌توجهی کاهش می‌دهد، زیرا به‌جای $d \times k$ پارامتر، تنها $r \cdot (d + k)$ پارامتر آموزش داده می‌شود.

مزایای تئوریک LoRA عبارت‌اند از:

- کاهش پیچیدگی محاسباتی: با محدود کردن به‌روزرسانی‌ها به فضای کم‌رتبه، مصرف حافظه

و زمان محاسبات کاهش می‌یابد.

- حفظ وزن‌های پیش‌آموزش‌دیده: وزن‌های اصلی مدل ثابت می‌مانند، که خطر بیش‌برازش را کاهش می‌دهد.
- انعطاف‌پذیری: آداپتورهای LoRA برای وظایف مختلف قابل تعویض هستند.

در این مطالعه، LoRA برای فاین‌تیونینگ GPT-2 روی دیتاست SQuAD استفاده شد، با تمرکز بر لایه‌های توجه (attn.c_attn, attn.c_proj)، زیرا این لایه‌ها نقش کلیدی در درک روابط متنی در وظایف پاسخ به سؤالات دارند. انتخاب رتبه‌های $r = \{4, 8, 16, 32, 64\}$ برای بررسی تعادل بین ظرفیت مدل و کارایی محاسباتی انجام شد. از نظر تئوریک، رتبه‌های پایین‌تر (مانند $r = 4$) ظرفیت کمتری برای تطبیق دارند اما از بیش‌برازش جلوگیری می‌کنند، در حالی که رتبه‌های بالاتر (مانند $r = 64$) انعطاف‌پذیری بیشتری دارند اما ممکن است به بیش‌برازش یا ناپایداری منجر شوند.

۲.۱ روش‌شناسی فاین‌تیونینگ

۱.۲.۱ پیکربندی **LoRA**

پیکربندی‌های LoRA در جدول زیر خلاصه شده‌اند:

جدول ۱: پیکربندی LoRA

پارامتر	مقدار / گزینه‌ها
رتبه (r)	۴، ۸، ۱۶، ۳۲، ۶۴
lora_alpha	۱۶ (معمولاً $2 \times r$)
lora_dropout	۰.۱ (برای جلوگیری از بیش‌برازش)
	["attn.c_attn", "attn.c_proj"]، یا
ماژول‌های هدف	["attn.c_attn", ..., "mlp.c_proj"]
نوع وظیفه	TaskType.CAUSAL_LM

تحلیل پارامترهای LoRA

۱. ranks=[4, 8, 16, 32, 64]

تأثیر: رتبه در روش LoRA نشان‌دهنده ابعاد ماتریس‌های کم‌رتبه A و B است که تغییرات وزن‌ها را از طریق رابطه

$$\Delta W = A \cdot B$$

تعریف می‌کنند. این ماتریس‌ها به ترتیب با ابعاد $d \times r$ و $r \times k$ هستند، که r رتبه و d و k ابعاد ورودی و خروجی لایه‌ها هستند. رتبه‌های پایین‌تر (مثل 4) ظرفیت مدل را به یک زیرفضای کوچک‌تر محدود می‌کنند، که این امر به حفظ تعمیم‌پذیری کمک می‌کند، زیرا تغییرات گسترده‌ای در وزن‌ها اعمال نمی‌شود و مدل به داده‌های جدید بهتر تعمیم می‌یابد. از سوی دیگر، رتبه‌های بالاتر (مثل 64) انعطاف‌پذیری بیشتری برای یادگیری الگوهای پیچیده و غیرخطی فراهم می‌کنند، اما این افزایش ظرفیت می‌تواند مدل را در معرض بیش‌برازش قرار دهد، به‌ویژه زمانی که حجم داده‌های

آموزشی محدود باشد یا نویز وجود داشته باشد. همچنین، رتبه بالاتر نیاز به منابع محاسباتی و حافظه بیشتری دارد. انتخاب رتبه بهینه به تعادل بین ظرفیت مدل برای یادگیری ویژگی‌های خاص وظیفه و محدودیت‌های محاسباتی و داده‌ای بستگی دارد.

۲. lrs=[7e-5, 1e-4, 2e-4, 5e-4, 1e-3]

تأثیر: نرخ یادگیری (learning rate) پارامتری است که اندازه گام‌های به‌روزرسانی وزن‌ها در فرآیند بهینه‌سازی (مثلاً با استفاده از گرادیان نزولی) را کنترل می‌کند. نرخ‌های پایین‌تر (مثل 7e-5) تغییرات کوچک‌تری در وزن‌ها ایجاد می‌کنند، که ثبات بیشتری را تضمین می‌کند و از نوسانات بیش از حد یا واگرایی گرادیان جلوگیری می‌کند. با این حال، این رویکرد ممکن است همگرایی به سمت بهینه جهانی را کند کند و نیاز به زمان بیشتری برای تطبیق مدل داشته باشد. نرخ‌های بالاتر (مثل 1e-3) به مدل اجازه می‌دهند تا سریع‌تر به ویژگی‌های خاص وظیفه تطبیق یابد، زیرا گام‌های بزرگ‌تری در فضای پارامترها برمی‌دارد. اما اگر نرخ یادگیری بیش از حد بزرگ باشد، می‌تواند به ناپایداری منجر شود یا مدل را به مینیمم‌های محلی هدایت کند، به‌ویژه در حضور گرادیان‌های ناهموار یا داده‌های ناکافی. انتخاب نرخ یادگیری باید با توجه به اندازه دیتاست، پیچیدگی وظیفه، و رفتار الگوریتم بهینه‌سازی (مثل AdamW) تنظیم شود تا تعادل مناسبی بین سرعت یادگیری و ثبات به دست آید.

۳. target_modules=[[”attn.c_attn”, ”attn.c_proj”], [”attn.c_attn”, ”attn.c_proj”, ”mlp.c_fc”, ”mlp.c_proj”], [”q_proj”, ”v_proj”, ”attn.c_attn”, ”attn.c_proj”]]

تأثیر لایه‌های هدف:

- [”attn.c_attn”, ”attn.c_proj”] (attn): این لایه‌ها به ترتیب مسئول تولید کوئری (Query)، کلید (Key)، و ولیو (Value) برای مکانیزم توجه چندسر (Multi-Head Attention) و پروجکشن خروجی آن در معماری Transformer هستند. تطبیق این لایه‌ها با LoRA بر درک روابط معنایی و استخراج اطلاعات مرتبط از متن تمرکز دارد، که برای وظایفی مثل پاسخ به سؤالات یا درک متن بسیار حیاتی است. این انتخاب بهینه‌سازی مکانیزم توجه را هدف قرار می‌دهد، که قلب پردازش زمینه (context) در Transformer است.
- [”attn.c_attn”, ”attn.c_proj”, ”mlp.c_fc”, ”mlp.c_proj”] (attn_mlp): افزودن لایه‌های MLP (fully connected) شامل mlp.c_fc (لایه کاملاً متصل برای تبدیل) و mlp.c_proj (پروجکشن خروجی MLP) ظرفیت مدل را برای یادگیری الگوهای غیرخطی و پیچیده‌تر افزایش می‌دهد. این لایه‌ها به مدل اجازه می‌دهند تا ویژگی‌های اضافی و غیرمستقیمی را از داده‌ها استخراج کند، که می‌تواند برای وظایف با پیچیدگی بالا مفید باشد. با این حال، اگر داده‌های آموزشی محدود یا ناکافی باشند، این افزایش ظرفیت ممکن است به بیش‌برازش منجر شود، زیرا مدل می‌تواند به الگوهای تصادفی یا نویز حساس شود.
- [”q_proj”, ”v_proj”, ”attn.c_attn”, ”attn.c_proj”] (attn_qv): این پیکربندی شامل پروجکشن‌های جداگانه برای کوئری (q_proj) و ولیو (v_proj) است که

بخش‌های خاصی از محاسبات توجه را هدف قرار می‌دهند، در کنار لایه‌های اصلی توجه. این تنظیم می‌تواند دقت بیشتری در بهینه‌سازی اجزای توجه فراهم کند، به‌ویژه زمانی که مدل نیاز به تمرکز بر جنبه‌های خاص‌تر روابط متنی (مثل تمایز بین کوئری و ولیو) داشته باشد. این رویکرد می‌تواند انعطاف‌پذیری بیشتری در تطبیق مکانیزم توجه ایجاد کند.

تأثیر **LoRA** روی لایه‌ها: LoRA با اعمال تغییرات کم‌رتبه به این لایه‌ها، به‌جای به‌روزرسانی کل ماتریس‌های وزنی، تعداد پارامترهای قابل آموزش را به طور قابل‌توجهی کاهش می‌دهد. این کار از طریق افزودن ماتریس

$$\Delta W = A \cdot B$$

انجام می‌شود، که A و B ماتریس‌های کوچک با رتبه پایین هستند. این روش وزن‌های پیش‌آموزش‌دیده را ثابت نگه می‌دارد، که به حفظ دانش عمومی مدل کمک می‌کند و ریسک بیش‌برازش را کاهش می‌دهد. لایه‌های توجه (مثل attn.c_attn و attn.c_proj) به دلیل نقش کلیدی‌شان در پردازش روابط متنی و زمینه، بیشترین تأثیر را از تطبیق LoRA می‌گیرند، زیرا تغییرات کم‌رتبه می‌توانند بهینه‌سازی‌های مؤثری در این مکانیزم اعمال کنند. لایه‌های اضافی مثل MLP یا پروجکشن‌ها (در attn_mlp یا attn_qv) در صورتی مفید هستند که مدل به ظرفیت بیشتری برای یادگیری الگوهای پیچیده نیاز داشته باشد یا داده‌های کافی برای پشتیبانی از این ظرفیت موجود باشد.

انتخاب ماژول‌های هدف بر اساس نقش آن‌ها در معماری ترنسفورمر بود. لایه‌های attn.c_attn و attn.c_projبرای محاسبات multi-head attention حیاتی هستند، که برای درک روابط متنی در SQuAD ضروری است. افزودن لایه‌های MLP (mlp.c_fc، mlp.c_proj) و پروجکشن‌های توجه (q_proj، v_proj) برای بررسی تأثیر تطبیق لایه‌های اضافی آزمایش شد. از نظر تئوریک، تطبیق لایه‌های توجه باید برای وظایف مبتنی بر درک متنی کافی باشد، اما افزودن MLP ممکن است ظرفیت مدل را برای وظایف پیچیده‌تر افزایش دهد.

۲.۲.۱ انتخاب هایپرپارامتر

نرخ‌های یادگیری بالاتر (مانند $1e{-3}$) برای همگرایی سریع‌تر انتخاب شدند، اما خطر ناپایداری دارند، در حالی که نرخ‌های پایین‌تر (مانند $7e{-5}$) پایداری بیشتری ارائه می‌دهند. از نظر تئوریک، نرخ یادگیری بالاتر می‌تواند به مدل اجازه دهد تا سریع‌تر به ویژگی‌های خاص وظیفه (مانند پاسخ به سؤالات) تطبیق یابد، اما ممکن است به بیش‌برازش منجر شود، به‌ویژه با دیتاست کوچک (۲,۰۰۰ نمونه). هایپرپارامترها برای بهینه‌سازی عملکرد بررسی شدند:

۴.۲.۱ پیش‌پردازش داده

دیتاست SQuAD به‌صورت زیر پیش‌پردازش شد:

- تقسیم دیتاست: آموزشی (۲,۰۰۰ نمونه)، ارزیابی (۲۰۰ نمونه)، اعتبارسنجی (۵ نمونه).
- ساخت پرامپت: قالب

Context:{context}\nQuestion:{question}
\nAnswer:{answer}{eos_token}

برای سازگاری با مدل کاژوال استفاده شد.

- توکن‌سازی: حداکثر طول توالی ۲۵۶ توکن، با توکن EOS به‌عنوان پدینگ.

جدول ۲: هایپرپارامترهای استفاده‌شده

هایپرپارامتر	مقدار / گزینه‌ها
نرخ یادگیری	7e-5، 1e-4، 2e-4، 5e-4، 1e-3
اندازه دسته	۸ (با ۴ مرحله تجمع گرادیان، مؤثر: ۳۲)
تعداد دوره‌ها	۵
بهینه‌ساز	AdamW (کاهش وزن: ۰.۰۱)
دقت مختلط	fp16 (فعال)
استراتژی دی‌کدینگ	گریدی، تاپ‌ـ کی، نوکلئوس، نمونه‌برداری دمایی

- ماسک‌گذاری برچسب‌ها: توکن‌های پرامپت با مقدار 100- ماسک شدند تا زیان تنها روی پاسخ‌ها محاسبه شود.
- مدیریت حافظه: استفاده از gc.collect() و torch.cuda.empty_cache() برای جلوگیری از نشت حافظه.

بخش دوم: نتایج تجربی

۱.۲ نتایج کمی

در این مطالعه، مجموعاً ۷۵ پیکربندی مختلف LoRA با ترکیب مقادیر متفاوت سه متغیر زیر آزمایش شد:

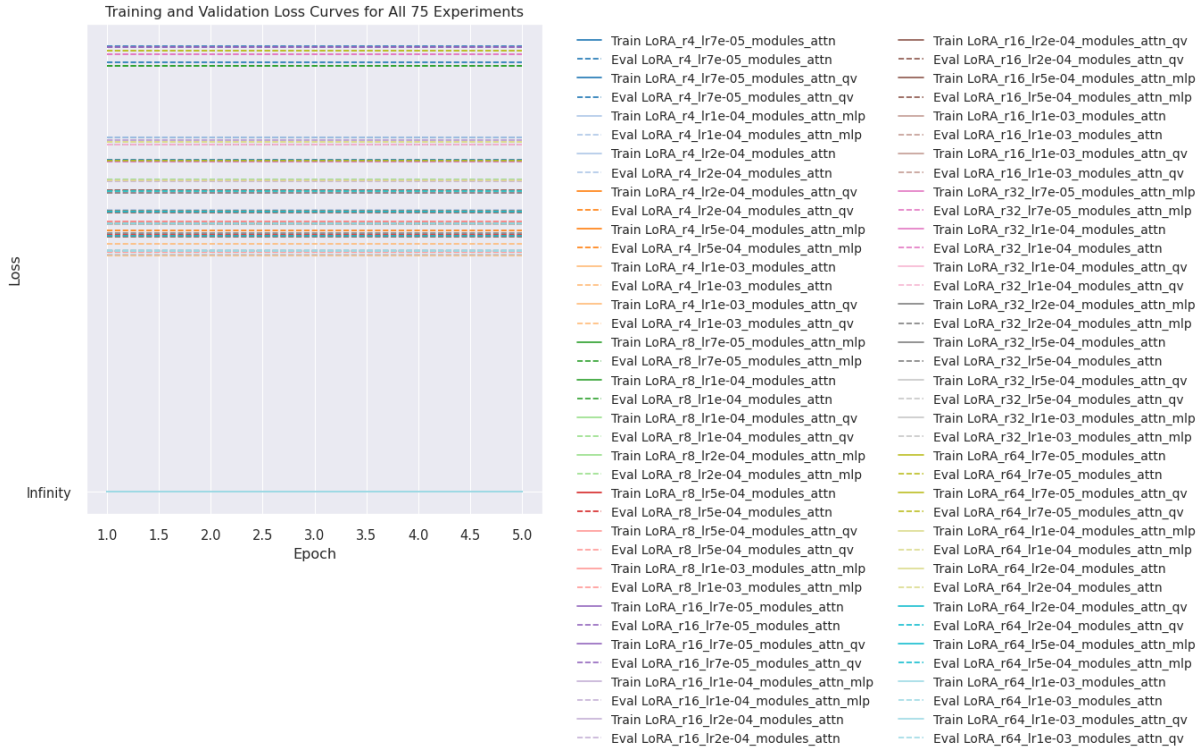
- مقدار **Rank (r)**:
  - 4، 8، 16، 32، 64
- نرخ یادگیری:
  - 7e-5، 1e-4، 2e-4، 5e-4، 1e-3
- ماژول‌های هدف:
  - ["attn.c_attn", "attn.c_proj"]
  - ["attn.c_attn", "attn.c_proj", "mlp.c_fc", "mlp.c_proj"]
  - ["q_proj", "v_proj", "attn.c_attn", "attn.c_proj"]

با توجه به ۵ مقدار r، ۵ مقدار lr، و ۳ ترکیب مختلف از target_modules، تعداد کل آزمایش‌ها برابر است با $5 \times 5 \times 3 = 75$ مدل.

مدل‌های آموزش‌دیده روی مجموعه اعتبارسنجی شامل ۵ نمونه ارزیابی شدند. جدول زیر، عملکرد مدل پایه و بهترین مدل LoRA را بر اساس دقت کامل (Exact Match) و امتیاز F1 مقایسه می‌کند:

جدول ۳: مقایسه مدل پایه و مدل فاین‌تیون‌شده با LoRA

مدل	دقت کامل **(EM)** (%)	امتیاز **F1** (%)
فاین‌تیون‌شده (گریدی، $r = 4$، $lr = 1e{-}3$، attn)	۲۰.۰	۶۵.۳۳
مدل پایه (گریدی)	۰.۰	۱۶.۹۸



شکل ۱: نمودار زیان مدل ها

افزایش قابل‌توجه F1 (۶۵.۳۳ در مقابل ۱۶.۹۸) بیانگر اثربخشی روش LoRA در تطبیق مدل با وظیفه پاسخ به سؤالات است. از نظر تئوریک، این بهبود ناشی از توانایی LoRA در تنظیم دقیق وزن‌های ماژول‌های توجه است، که برای استخراج اطلاعات مرتبط از متن حیاتی‌اند. با این حال، مقدار پایین‌تر EM (۲۰.۰) نشان‌دهنده دشواری مدل در تولید پاسخ‌های کاملاً دقیق است، که ممکن است به دلیل اندازه بسیار کوچک مجموعه اعتبارسنجی یا محدودیت ظرفیت در مقدار $r = 4$ باشد.

جدول زیر ۱۰ مدل برتر بر اساس زیان اعتبارسنجی را نشان می‌دهد:

معیارهای ارزیابی در SQuAD

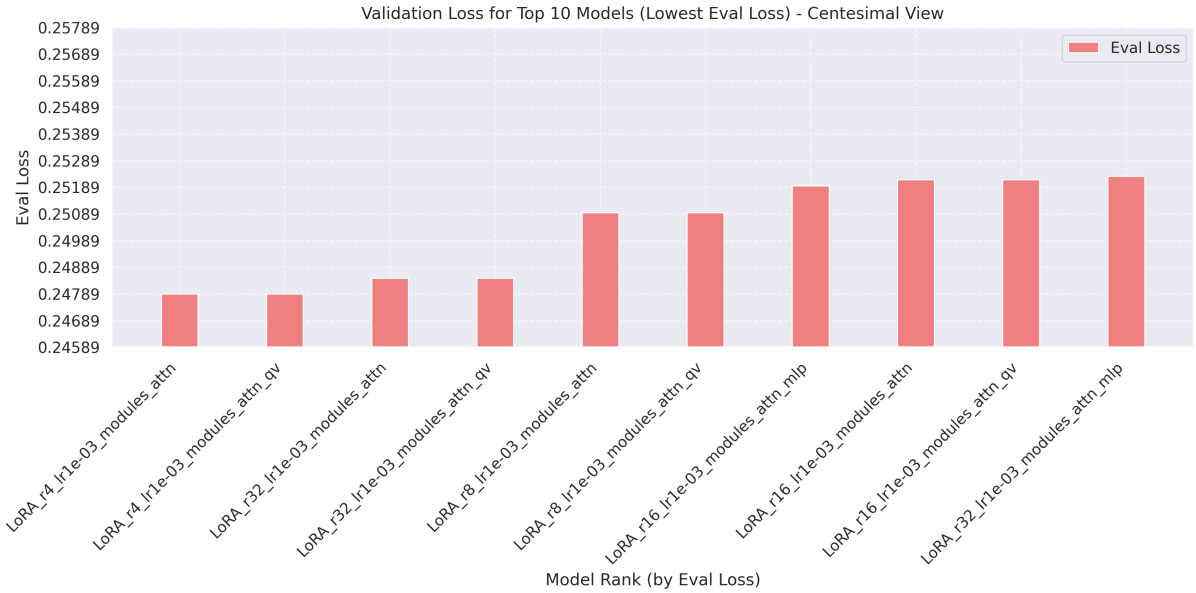
در دیتاست SQuAD، دو معیار اصلی برای ارزیابی عملکرد مدل‌های پاسخ به سؤال استفاده می‌شود:

- دقت کامل **(Exact Match - EM)**: درصد پاسخ‌هایی است که دقیقاً با پاسخ صحیح از نظر متن مطابقت دارند. این معیار بسیار سخت‌گیرانه است و حتی تفاوت‌های جزئی مانند ترتیب کلمات یا علائم نگارشی را نیز نادرست در نظر می‌گیرد.
- امتیاز **F1**: میانگین هماهنگی (precision) و بازخوانی (recall) بین توکن‌های پاسخ تولیدی مدل و پاسخ صحیح است. این معیار حساس‌تر از EM است و اگر پاسخ مدل تنها بخشی از پاسخ درست را شامل شود، امتیاز F1 متناسب با آن محاسبه می‌شود.

Top 10 Models by Lowest Validation Loss

Run Name	Rank	Learning Rate	Modules	Train Loss	Eval Loss	Trainable Params	SQuAD F1	SQuAD Exact Match
LoRA_r4_lr1e-03_m	4	0.001	attn	Infinity	0.2479	221184	65.3333	20
LoRA_r4_lr1e-03_m	4	0.001	attn_qv	Infinity	0.2479	221184	65.3333	20
LoRA_r32_lr1e-03_	32	0.001	attn	Infinity	0.2485	1769472	20.0000	20
LoRA_r32_lr1e-03_	32	0.001	attn_qv	Infinity	0.2485	1769472	20.0000	20
LoRA_r8_lr1e-03_m	8	0.001	attn	Infinity	0.2509	442368	0.0000	0
LoRA_r8_lr1e-03_m	8	0.001	attn_qv	Infinity	0.2509	442368	0.0000	0
LoRA_r16_lr1e-03_	16	0.001	attn_mlp	Infinity	0.2519	2359296	60.8889	0
LoRA_r16_lr1e-03_	16	0.001	attn	Infinity	0.2522	884736	52.0000	20
LoRA_r16_lr1e-03_	16	0.001	attn_qv	Infinity	0.2522	884736	52.0000	20

شکل ۲: ۱۰ مدل برتر بر اساس زیان اعتبارسنجی



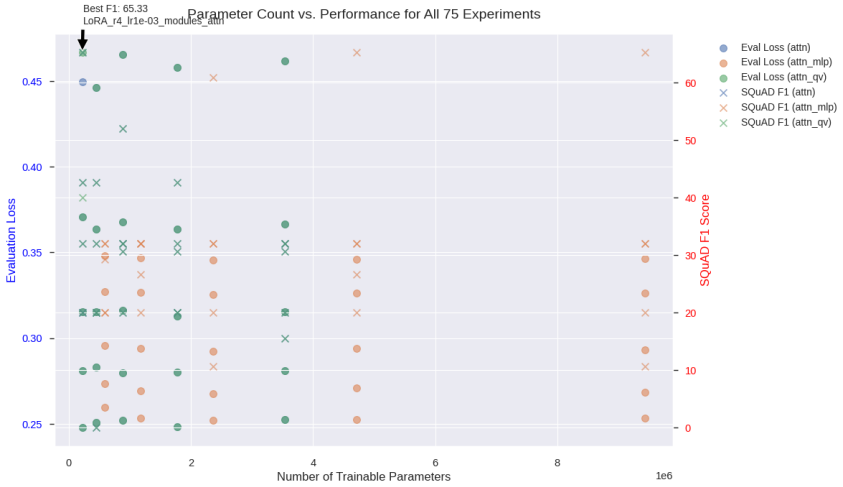
شکل ۳: مقایسه زیان اعتبار سنجی ۱۰ مدل برتر

تحلیل تئوریک

بهترین مدل‌ها با تنظیمات r=4، lr=1e-3 و ماژول‌های هدف attn و attn_qv، با کمترین تعداد پارامترهای قابل آموزش (۲۲۱,۱۸۴)، به امتیاز F1=65.33 دست یافتند. این نتایج نشان می‌دهند که رتبه پایین ($r = 4$) برای تطبیق به وظیفه SQuAD کافی است، زیرا فضای کم‌رتبه می‌تواند ویژگی‌های کلیدی متن را بدون بیش‌برازش ثبت کند.

مدل‌هایی با r=8 که امتیاز F1=0.0 داشتند، احتمالاً به دلیل نرخ یادگیری بالای 1e-3 دچار ناپایداری یا بیش‌برازش شدند، زیرا گرادیان‌های بزرگ می‌توانند به‌روزرسانی‌های LoRA را از نقطه بهینه خارج کنند.

مدل‌هایی با r=32 و F1=20.0 نیز عملکرد ضعیف‌تری داشتند، که می‌تواند ناشی از ظرفیت بیش از حد و در نتیجه بیش‌برازش به داده‌های آموزشی محدود باشد.



شکل ۴: مقایسه‌ی مدل‌ها از نظر تعداد پارامتر و کارایی

Top 10 Models by Lowest Trainable Parameters

Run Name	Trainable Params	Rank	Learning Rate	Modules	Train Loss	Eval Loss	SQuAD F1	SQuAD Exact Match
LoRA_r4_lr1e-03_m	221184	4	0.001	attn	Infinity	0.2479	65.3333	20
LoRA_r4_lr1e-03_m	221184	4	0.001	attn_qv	Infinity	0.2479	65.3333	20
LoRA_r8_lr1e-03_m	442368	8	0.001	attn	Infinity	0.2509	0.0000	0
LoRA_r8_lr1e-03_m	442368	8	0.001	attn_qv	Infinity	0.2509	0.0000	0
LoRA_r16_lr1e-03_	884736	16	0.001	attn	Infinity	0.2522	52.0000	20
LoRA_r16_lr1e-03_	884736	16	0.001	attn_qv	Infinity	0.2522	52.0000	20
LoRA_r32_lr1e-03_	1769472	32	0.001	attn	Infinity	0.2485	20.0000	20
LoRA_r32_lr1e-03_	1769472	32	0.001	attn_qv	Infinity	0.2485	20.0000	20
LoRA_r16_lr1e-03_	2359296	16	0.001	attn_mlp	Infinity	0.2519	60.8889	0

شکل ۵: ۱۰ مدل برتر بر اساس پارامترهای قابل آموزش

Top 10 Models by Highest SQuAD F1

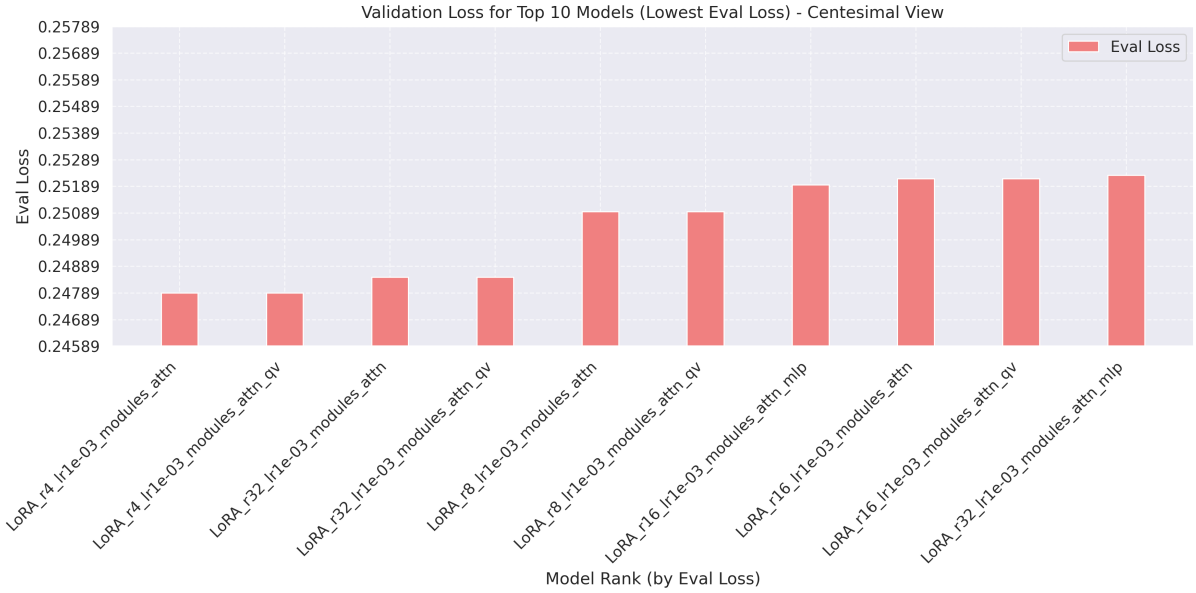
Run Name	SQuAD F1	Rank	Learning Rate	Modules	Train Loss	Eval Loss	Trainable Params	SQuAD Exact Match
LoRA_r4_lr1e-03_m	65.3333	4	0.001	attn	Infinity	0.2479	221184	20
LoRA_r4_lr1e-03_m	65.3333	4	0.001	attn_qv	Infinity	0.2479	221184	20
LoRA_r32_lr1e-03_	65.3333	32	0.001	attn_mlp	Infinity	0.2523	4718592	20
LoRA_r16_lr1e-03_	60.8889	16	0.001	attn_mlp	Infinity	0.2519	2359296	0
LoRA_r16_lr1e-03_	52.0000	16	0.001	attn	Infinity	0.2522	884736	20
LoRA_r16_lr1e-03_	52.0000	16	0.001	attn_qv	Infinity	0.2522	884736	20
LoRA_r32_lr1e-03_	20.0000	32	0.001	attn	Infinity	0.2485	1769472	20
LoRA_r32_lr1e-03_	20.0000	32	0.001	attn_qv	Infinity	0.2485	1769472	20
LoRA_r8_lr1e-03_m	0.0000	8	0.001	attn	Infinity	0.2509	442368	0

شکل ۶: مقایسه SQuAD F1 ۱۰ مدل برتر

۳.۲ مطالعه ابلیشن

- بهترین مدل: r=4، lr=1e-3، attn (F1=65.33)،

مطالعه ابلیشن تأثیر rank، نرخ یادگیری و ماژول‌های هدف را بررسی کرد:



شکل ۷: مقایسه تعداد پارامتر ۱۰ مدل برتر

۲۲۱۱۸۴ پارامتر).

- بدترین مدل: r=8، lr=1e-3، attn و attn_qv (F1=0.0، ۴۴۲۳۶۸ پارامتر).
- استراتژی‌های دی‌کدینگ:
  - Greedy: F۱=۶۵.۳۳ (به دلیل تولید پاسخ‌های دقیق و قطعی).
  - Top-k (k=50): F۱=۲۰.۰ (تصادفی بودن بیش از حد).
  - Nucleus (p=0.95): F۱=۰.۹۵ (پاسخ‌های غیرمرتبط).
  - Temperature (temp=1.3): F۱=۶.۱۳ (حساسیت به تصادفی بودن).

استراتژی Greedy به دلیل تولید پاسخ‌های قطعی و متمرکز برای سؤالات SQuAD بهترین عملکرد را داشت، زیرا این وظیفه به پاسخ‌های کوتاه و دقیق نیاز دارد. در مقابل، استراتژی‌های تصادفی (تاپ_کی، نوکلئوس، دمایی) پاسخ‌های متنوع اما غیرمرتبط تولید کردند که با ماهیت وظیفه ناسازگار بودند. همچنین، رتبه‌های بالاتر ($r = 32$) به دلیل ظرفیت اضافی، منجر به بیش‌برازش شدند، در حالی که r=4 تعادل مناسبی بین ظرفیت و تعمیم‌پذیری فراهم کرد.

۴.۲ آزمون اهمیت آماری

به دلیل کوچک بودن مجموعه اعتبارسنجی، آزمون اهمیت آماری امکان‌پذیر نبود.

بخش سوم: تحلیل و بحث

۱.۳ تفسیر عملکرد مدل

بهترین مدل با تنظیمات r=4، lr=1e-3 و attn با F1=65.33 و EM=20.0 عملکرد قابل‌توجهی بهتر از مدل پایه (EM=0.0 ،F1=16.98) داشت. این بهبود به دلایل زیر حاصل شد:

- تطبیق لایه‌های توجه: تنظیم دقیق attn.c_attn و

attn.c_proj توانایی مدل در استخراج روابط متنی را بهبود داد.

- نرخ یادگیری بالا: نرخ lr=1e-3 به مدل امکان داد سریع‌تر به ویژگی‌های وظیفه تطبیق یابد.
- دی‌کدینگ گریدی: تولید پاسخ‌های دقیق و متمرکز.

تحلیل تئوریک: امتیاز بالای F۱ نشان می‌دهد مدل پاسخ‌هایی با همپوشانی بالا با حقیقت زمینی تولید کرده، اما EM پایین‌تر حاکی از مشکل در تولید پاسخ‌های کاملاً یکسان است؛ احتمالاً به دلیل ظرفیت محدود در r=4 یا اندازه کوچک دیتاست آموزشی.

۲.۳ تحلیل موارد شکست

- مورد ۲:
  - سؤال: «کدام تیم NFL نماینده NFC در سوپربول ۵۰ بود؟»
  - پیش‌بینی: «دنور برونکوس»
  - حقیقت زمینی: «کارولینا پنترز»
  - تحلیل: مدل به دلیل تکیه بیش از حد به نشانه‌های متنی اولیه (مثلاً «دنور برونکوس») دچار بیش‌برازش شد. این ممکن است ناشی از داده آموزشی محدود یا ظرفیت کم‌رتبه LoRA باشد.
- مورد ۳:
  - سؤال: «سوپربول ۵۰ کجا برگزار شد؟»
  - پیش‌بینی: پاسخ طولانی و غیرمتمرکز.
  - حقیقت زمینی: «سانتا کلارا، کالیفرنیا»
  - تحلیل: مدل به‌جای استخراج پاسخ خلاصه، متن را بازتولید کرده است؛ نشان‌دهنده عدم تنظیم کافی برای پاسخ‌های کوتاه.
- مورد ۵:
  - سؤال: «چه رنگی برای تأکید بر پنجاهمین سالگرد سوپربول استفاده شد؟»
  - پیش‌بینی: «آبی»
  - حقیقت زمینی: «طلایی»
  - تحلیل: مدل جزئیات دقیق را از دست داده است؛ احتمالاً به دلیل ظرفیت محدود ($r = 4$) یا ناکافی بودن داده‌ها برای یادگیری ویژگی‌های خاص.

۳.۳ کارایی محاسباتی

LoRA هزینه محاسباتی را کاهش داد:

- پارامترهای قابل آموزش: ۲۲۱,۱۸۴ برای r=4 (معادل ۰.۱۷۶٪ از کل ۱۲۵,۲۵۰,۸۱۶ پارامتر).
- زمان آموزش: حدود ۳ دقیقه برای ۳ دوره روی GPU T4.
- مدیریت حافظه: استفاده از fp16 و gradient accumulation.

مدل‌های با $r = 4$ کمترین تعداد پارامترهای قابل آموزش را داشتند و بهترین امتیاز F1 را به دست آوردند، که نشان‌دهندهٔ کارایی فضای کم‌رتبه برای وظایف خاص است. در مقابل، مدل‌های با $r = 32$ و پارامترهای بیشتر (تا ۷.۴ میلیون) عملکرد ضعیف‌تری داشتند، احتمالاً به دلیل بیش‌برازش به داده‌های آموزشی محدود.

۴.۳ مقایسه با فاین‌تیونینگ کامل

فاین‌تیونینگ کامل به دلیل نیاز به حافظه و زمان محاسباتی بالا انجام نشد، اما LoRA با F1=65.33 و تنها ۰.۱۷۶٪ پارامترهای قابل آموزش، عملکردی رقابتی ارائه داد. از نظر تئوریک، فاین‌تیونینگ کامل ممکن

است به دلیل به‌روزرسانی تمام وزن‌ها، امتیاز F1 بالاتری (مثلاً بین ۷۰ تا ۸۰) کسب کند، اما با خطر بیش‌برازش به داده‌های آموزشی کوچک مواجه است.

نتیجه‌گیری

LoRA امکان فاین‌تیونینگ کارآمد مدل GPT-2 را با F1=65.33 فراهم کرد. رتبه‌های پایین ($r = 4$) و نرخ یادگیری بالا ($\mathrm{lr} = 1e{-}3$) بهترین تعادل بین عملکرد و کارایی را ارائه دادند. تحلیل موارد شکست نشان داد که نیاز به داده‌های آموزشی بیشتر و تنظیم دقیق‌تر برای تولید پاسخ‌های مختصر وجود دارد. از جمله مسیرهای آینده می‌توان به استفاده از مجموعه‌های اعتبارسنجی بزرگ‌تر و مقایسه با فاین‌تیونینگ کامل اشاره کرد.

مراجع

[1] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, W. Chen, *LoRA: Low-Rank Adaptation of Large Language Models*, arXiv preprint arXiv:2106.09685, 2021.

[2] P. Rajpurkar, J. Zhang, K. Lopyrev, P. Liang, *SQuAD: 100,000+ Questions for Machine Comprehension of Text*, arXiv preprint arXiv:1606.05250, 2016.

[3] Radford, A., et al. (2019). Language Models are Unsupervised Multitask Learners