مروری بر مدل‌های زبانی

سپهر کراچی

۱۴۰۶

فهرست مطالب

۱ آموزش (Training) ۲
۱.۱ داده‌ها (Data) ۲
۱.۱.۱ پیش‌آموزش (Pre-Training) ۲
۲.۱.۱ پس‌آموزش (Post-Training) ۲
۲.۱ روش (Method) ۳
۱.۲.۱ پیش‌آموزش (Pre-Training) ۳
۲.۲.۱ پس‌آموزش (Post-Training) ۳

۲ ساختار (Architecture) ۴
۱.۲ کدگذاری موقعیت (Positional Encoding) ۴
۲.۲ بردارکننده (Tokenizer) ۴
۳.۲ شبکه عصبی (MLP) ۵
۴.۲ تابع فعالساز (Activation Function) ۵
۵.۲ نرمالسازی (Normalization) ۶
۶.۲ مکانیزم توجه (Attention) ۶
۷.۲ ساختار ترنسفورمر (Transformer) ۷

۳ بهینه سازی (Optimization) ۷
۱.۳ الگوریتم (Algorithm) ۷
۲.۳ تابع هزینه (Loss Function) ۸

۴ تنظیم هایپرپارامتر (Hyperparameter Tuning) ۸

۵ قوانین مقیاس پذیری (Scaling Laws) ۹
۱.۵ مقدمه ۹
۲.۵ اصلاح و توسعه قوانین مقیاس‌پذیری کلاسیک ۹
۳.۵ تاثیر معماری و شکل مدل (Model Shape & Architecture) ۹
۴.۵ قوانین مقیاس‌پذیری برای مدل‌های ترکیبی از خبرگان (MoE) ۱۰
۵.۵ نقش داده‌ها: کیفیت، ترکیب و چگالی ۱۰
۶.۵ فراتر از خطای مطلق: معیارهای نوین ارزیابی ۱۱
۷.۵ نتیجه‌گیری ۱۱

۶ ارزیابی (Evaluation) ۱۱

۷ یادگیری تقویتی (Reinforcement Learning) ۱۲
۱.۷ الگوریتم بهینه‌سازی تقریبی خط‌مشی (Proximal Policy Optimization - PPO) ۱۲

۸ منابع ۲۰

۱ آموزش (Training)

۱.۱ داده‌ها (Data)

۱.۱.۱ پیش‌آموزش (Pre-Training)

- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda

۲.۱.۱ پس‌آموزش (Post-Training)

- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda

۲.۱ روش (Method)

۱.۲.۱ پیش‌آموزش (Pre-Training)

- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda

۲.۲.۱ پس‌آموزش (Post-Training)

- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda

۲ ساختار (Architecture)

۱.۲ کدگذاری موقعیت (Positional Encoding)

- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda

۲.۲ بردارکننده (Tokenizer)

- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda

۳.۲ شبکه عصبی (MLP)

- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda

۴.۲ تابع فعالساز (Activation Function)

- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda

۵.۲ نرمالسازی (Normalization)

- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda

۶.۲ مکانیزم توجه (Attention)

- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda

۷.۲ ساختار ترنسفورمر (Transformer)

- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda

۳ بهینه سازی (Optimization)

۱.۳ الگوریتم (Algorithm)

- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda

۲.۳ تابع هزینه (Loss Function)

- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda

۴ تنظیم هایپرپارامتر (Hyperparameter Tuning)

- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda

۵ قوانین مقیاس پذیری (Scaling Laws)

۱.۵ مقدمه

قوانین مقیاس‌پذیری (Scaling Laws) به عنوان یکی از ارکان اصلی در توسعه و درک مدل‌های زبانی بزرگ (LLMs) شناخته می‌شوند. این قوانین که در ابتدا توسط محققانی چون Kaplan و Hoffmann (قانون Chinchilla) مطرح شدند، نشان می‌دهند که عملکرد مدل‌ها با افزایش پارامترها، حجم داده‌ها و بودجه محاسباتی به صورت قابل پیش‌بینی بهبود می‌یابد. با این حال، تحقیقات اخیر نشان داده‌اند که قوانین کلاسیک دارای محدودیت‌هایی هستند؛ از جمله نادیده گرفتن هزینه‌های استنتاج (Inference)، تاثیر شکل معماری (Architecture Shape)، کیفیت داده‌ها و ساختارهای نوینی مانند «ترکیبی از خبرگان» (Mixture-of-Experts یا MoE). این بخش، با ترکیب و تحلیل ۱۹ مقاله علمی جدید، به بررسی ابعاد پنهان و توسعه‌یافته‌ی قوانین مقیاس‌پذیری می‌پردازد.

۲.۵ اصلاح و توسعه قوانین مقیاس‌پذیری کلاسیک

قوانین اولیه مانند Chinchilla فرض می‌کردند که نسبت بهینه داده به پارامتر (D/N) یک مقدار ثابت (حدود ۲۰) است. اما لی و همکاران [۱] با معرفی قانون Farseer نشان می‌دهند که این نسبت ثابت نیست و با افزایش بودجه محاسباتی، نسبت بهینه D/N نیز افزایش می‌یابد. روش Farseer با استفاده از برازش قطعه‌ای دیفرانسیلی (Differential Piecewise Fitting)، خطای برون‌یابی (Extrapolation Error) را نسبت به قانون Chinchilla تا ۴۳۳ درصد کاهش می‌دهد.

از سوی دیگر، چن و همکاران [۲] پدیده Sub-scaling (کاهش نرخ بهبود عملکرد) را بررسی می‌کنند. این پدیده زمانی رخ می‌دهد که مدل‌ها بیش از حد آموزش می‌بینند (Over-training) یا چگالی داده‌ها (افزونگی اطلاعات) بسیار بالا است. این مقاله یک قانون مقیاس‌پذیری زیر-بهینه (Sub-optimal) ارائه می‌دهد که نشان می‌دهد داده‌های با چگالی بالا منجر به رشد توانی (کاهش بازدهی نهایی) و داده‌های با چگالی پایین منجر به رشد خطی در عملکرد می‌شوند.

همچنین، مک‌لیش و همکاران [۳] با معرفی مجموعه Gemstones (شامل بیش از ۴۰۰۰ چک‌پوینت)، شکنندگی قوانین مقیاس‌پذیری نسبت به انتخاب‌های ابرپارامتری (مانند نرخ یادگیری و شکل مدل) را برجسته می‌کنند. آن‌ها روش Convex Hull را برای برازش قوانین پیشنهاد می‌دهند که نسبت به روش‌های پیشین بسیار مقاوم‌تر است.

۳.۵ تاثیر معماری و شکل مدل (Model Shape & Architecture)

یکی از مهم‌ترین نقایص قوانین گذشته، نادیده گرفتن تاثیر شکل مدل (نسبت عمق به عرض) و هزینه‌های استنتاج است.

- **عرض در برابر عمق:** بیان و همکاران [۴] نشان می‌دهند که معماری مدل مستقیماً بر تاخیر استنتاج (Inference Latency) تاثیر می‌گذارد. مدل‌های عریض‌تر و کم‌عمق‌تر (Wider and Shallower) با بودجه پارامتری یکسان، توان عملیاتی (Throughput) بسیار بالاتری در استنتاج دارند. در همین راستا، لیو و همکاران [۵] پدیده Inverse Depth Scaling را معرفی می‌کنند؛ جایی که خطا با نسبت معکوس عمق ($1/\ell$) کاهش می‌یابد. آن‌ها دلیل این امر را میانگین‌گیری گروهی (Ensemble Averaging) لایه‌ها می‌دانند، به این معنا که لایه‌های متعدد در LLMها کارهای مشابهی انجام می‌دهند و صرفاً با میانگین‌گیری، خطا را کاهش می‌دهند که نشان‌دهنده استفاده ناکارآمد از عمق است.

- **تعامل عصبی (Neural Interaction):** سان و همکاران [۶] با استفاده از مفهوم Average Gradient Outer Product (AGOP)، قانون «تعامل عصبی» را مطرح می‌کنند. آن‌ها نشان می‌دهند که تعمیم‌پذیری خوب نیازمند برهم‌نهی خوش‌خیم (Benign Superposition) است؛ جایی که مدل با انرژی تعاملی پایین، سهم بالایی از حساسیت تعاملی را به دست می‌آورد. نسبت عمق به عرض ($R_{D/W}$) ابزاری برای قرار دادن مدل در این بازه بهینه است.

- **قوانین مقیاس‌پذیری طیفی (Spectral Scaling Laws):** جها و ریگن [۷] نشان می‌دهند که افزایش عرض شبکه‌های پیش‌خور (FFN) عمدتاً ظرفیت‌های کم‌انرژی (دم طیف) را گسترش می‌دهد، در حالی که زیرفضاهای غالب به سرعت اشباع می‌شوند. این قانون مقیاس‌پذیری طیفی نامتقارن نشان می‌دهد که انتخاب عرض FFN یک مبادله بین ظرفیت دم و ظرفیت حالت غالب است.

- **معماری‌های جایگزین (xLSTM و Mamba):** بک و همکاران [۸] قوانین مقیاس‌پذیری معماری xLSTM (با پیچیدگی زمانی خطی) را با ترانسفورمرها مقایسه می‌کنند. نتایج نشان می‌دهد که xLSTM در مبادله محاسبات-خطا (Compute-Loss Trade-off) بر ترانسفورمرها برتری پارتو (Pareto-dominate) دارد. همچنین، در مقاله‌ای دیگر [۹] با رویکرد تفسیرپذیری مکانیسمی، نشان داده شده است که معماری Mamba بازیابی اطلاعات (Associative Recall) را از طریق یادگیری ضمنی توابع هش خطی انجام می‌دهد و کران‌های بالایی برای اندازه واژگان و ابعاد حالت بر اساس لم Johnson-Lindenstrauss ارائه می‌دهد.

۴.۵ قوانین مقیاس‌پذیری برای مدل‌های ترکیبی از خبرگان (MoE)

مدل‌های MoE به دلیل جداسازی تعداد کل پارامترها از هزینه محاسباتی (FLOPs)، نیازمند قوانین مقیاس‌پذیری مختص به خود هستند.

- **اهرم کارایی (Efficiency Leverage):** تیان و همکاران [۱۰] معیار EL را برای سنجش برتری محاسباتی MoE نسبت به مدل‌های متراکم (Dense) معرفی می‌کنند. آن‌ها نشان می‌دهند که نسبت فعال‌سازی (Activation Ratio) محرک اصلی کارایی است و دانه‌بندی خبرگان (Expert Granularity) به عنوان یک تعدیل‌کننده غیرخطی عمل می‌کند.

- **مبادله پارامتر و FLOPs:** آبنار و همکاران [۱۱] نشان می‌دهند که برای یک بودجه محاسباتی ثابت، افزایش سطح پراکندگی (Sparsity) — یعنی داشتن پارامترهای کل بیشتر اما پارامترهای فعال کمتر — منجر به کاهش خطای آموزش می‌شود.

- **کارایی حافظه:** برخلاف تصور رایج، لودزیجوسکی و همکاران [۱۲] ثابت می‌کنند که مدل‌های MoE می‌توانند از نظر حافظه نیز کارآمدتر از مدل‌های متراکم باشند، به شرطی که روی توکن‌های بیشتری آموزش ببینند.

- **بازیافت مدل‌ها (Upcycling):** لیو و همکاران [۱۳] به بررسی تبدیل مدل‌های متراکم از پیش‌آموزش‌دیده به MoE می‌پردازند. آن‌ها یک قانون مقیاس‌پذیری مشترک کشف کردند که نشان می‌دهد Upcycling تنها زمانی کارآمدتر از آموزش از صفر (From-scratch) است که هزینه اولیه (Sunk Cost - توکن‌های آموزش مدل متراکم) و اندازه مدل نسبتاً کوچک باشند.

۵.۵ نقش داده‌ها: کیفیت، ترکیب و چگالی

داده‌ها سوخت اصلی قوانین مقیاس‌پذیری هستند و کیفیت آن‌ها به اندازه کمیتشان اهمیت دارد.

- **کیفیت داده‌ها:** سوبرامانیام و همکاران [۱۴] یک پارامتر بدون بعد کیفیت به نام $Q \in (0, 1]$ را معرفی کرده و قانون Chinchilla را برای در نظر گرفتن کیفیت داده‌ها بسط می‌دهند. آن‌ها نشان می‌دهند که داده‌های با کیفیت بالا می‌توانند نیاز به اندازه مدل و محاسبات را به شدت کاهش دهند و اندازه موثر نمونه به صورت $D_{eff} = D \cdot Q^{\gamma}$ مقیاس می‌یابد.

- **ترکیب بهینه داده‌ها (Data Mixtures):** انتخاب نسبت ترکیب دامنه‌های مختلف داده معمولاً با آزمون و خطا انجام می‌شود. شوکور و همکاران [۱۵] روشی سیستماتیک ارائه می‌دهند که با استفاده از برازش قوانین مقیاس‌پذیری روی اجراهای مقیاس کوچک، می‌توان وزن‌های بهینه دامنه را برای آموزش مدل‌های بسیار بزرگ پیش‌بینی کرد.

- **ساختار سلسله‌مراتبی داده‌ها:** کاگنتا و همکاران [۱۶] با استفاده از مدل سلسله‌مراتب تصادفی (RHM)، نشان می‌دهند که شبکه‌های عصبی پیچیدگی‌های داده را لایه به لایه یاد می‌گیرند. آن‌ها ثابت می‌کنند که شبکه‌های پیچشی (CNNs) به دلیل اشتراک وزن و اتصال محلی، در یادگیری داده‌های سلسله‌مراتبی سریع‌تر از ترانسفورمرها مقیاس می‌یابند.

۶.۵ فراتر از خطای مطلق: معیارهای نوین ارزیابی

ارزیابی مدل‌ها صرفاً بر اساس خطای آنتروپی متقاطع (Cross-Entropy) تصویر کاملی از توانایی‌های مدل ارائه نمی‌دهد.

- **قوانین مقیاس‌پذیری نسبی (Relative Scaling Laws):** هلد و همکاران [۱۷] استدلال می‌کنند که ارزیابی‌های تجمیعی، تفاوت عملکرد در زیرگروه‌ها را پنهان می‌کند. آن‌ها قوانین نسبی را معرفی می‌کنند تا نشان دهند چگونه شکاف عملکرد بین توزیع‌ها با افزایش مقیاس تغییر می‌کند. نتایج نشان می‌دهد که مقیاس‌پذیری یک تساوی‌بخش جهانی نیست؛ در حالی که دامنه‌های علمی در MMLU به همگرایی می‌رسند، خطرات هوش مصنوعی (مانند توانایی فریبکاری) با افزایش مقیاس واگرا شده و افزایش می‌یابند.

- **احتمال مبتنی بر رتبه (Relative-Based Probability):** یو و همکاران [۱۸] معیار RBP را معرفی می‌کنند که احتمال قرار گرفتن توکن صحیح در میان k پیش‌بینی برتر را می‌سنجد. آن‌ها نشان می‌دهند که $-\log(RBP_k)$ نیز از یک قانون توانی پیروی می‌کند. این دیدگاه پدیده ظهور توانایی‌ها (Emergence) را نه به عنوان یک جهش ناگهانی، بلکه به عنوان یک اثر ماکروسکوپی قابل پیش‌بینی از مقیاس‌پذیری میکروسکوپی هموار توضیح می‌دهد.

- **قانون چگال‌سازی (Densing Law):** در نهایت، شیائو و همکاران [۱۹] برای ارزیابی روند پیشرفت کارایی LLMها در طول زمان، مفهوم چگالی قابلیت (Capability Density) را معرفی می‌کنند. مشاهدات تجربی آن‌ها نشان می‌دهد که حداکثر چگالی قابلیت مدل‌های متن‌باز تقریباً هر ۵.۳ ماه دو برابر می‌شود که نشان‌دهنده کاهش نمایی هزینه‌های استنتاج و نیازهای پارامتری برای رسیدن به یک سطح عملکرد ثابت است.

۷.۵ نتیجه‌گیری

بررسی جامع این مقالات نشان‌دهنده یک تغییر پارادایم در درک ما از قوانین مقیاس‌پذیری است. ما از دوران «بزرگتر بودن همیشه بهتر است» (صرفاً با افزایش پارامتر و داده) عبور کرده‌ایم. قوانین مقیاس‌پذیری نوین اکنون چندوجهی شده‌اند و باید هزینه‌های استنتاج، شکل معماری (عمق/عرض)، کیفیت و چگالی داده‌ها، ساختارهای پراکنده (MoE) و معیارهای ارزیابی نسبی را در بر بگیرند. این بینش‌های جدید به محققان و مهندسان اجازه می‌دهد تا مدل‌های زبانی آینده را نه تنها قدرتمندتر، بلکه از نظر محاسباتی، حافظه و استنتاج به مراتب کارآمدتر و هدفمندتر طراحی کنند.

۶ ارزیابی (Evaluation)

- sda
- sda
- sda
- sda
- sda

- sda
- sda
- sda
- sda
- sda
- sda
- sda
- sda

۷ یادگیری تقویتی (Reinforcement Learning)

- sda

۱.۷ الگوریتم بهینه‌سازی تقریبی خط‌مشی (Proximal Policy Optimization - PPO)

الگوریتم **PPO** که توسط جان شولمن (John Schulman) و همکارانش در سال ۲۰۱۷ در OpenAI معرفی شد [۲۰]، یکی از برجسته‌ترین، پایدارترین و پرکاربردترین الگوریتم‌های حوزه یادگیری تقویتی عمیق (Deep Reinforcement Learning) به شمار می‌رود. این الگوریتم با حل چالش‌های بنیادین بهینه‌سازی در روش‌های پیشین، تعادلی بهینه میان سادگی پیاده‌سازی، کارایی محاسباتی، بهره‌وری نمونه (Sample Efficiency) و پایداری همگرایی برقرار ساخته است.

تاریخچه، انگیزه و ضرورت پیدایش الگوریتم

پیش از ابداع **PPO**، روش‌های یادگیری تقویتی عمیق به سه دسته عمده با محدودیت‌های ساختاری تقسیم می‌شدند:

- **روش‌های مبتنی بر ارزش (Value-Based Methods):** شاخص‌ترین الگوریتم این دسته، **DQN** [۲۱] است. این روش‌ها گرچه در فضاهای عمل گسسته و محیط بازی‌های آتاری [۲۲] موفق ظاهر شدند، اما امکان تعمیم کارآمد به فضاهای عمل پیوسته (Continuous Action Spaces) با ابعاد بالا در رباتیک [۲۳، ۲۴] را نداشتند، چرا که یافتن $\arg\max_a Q(s,a)$ در هر گام نیازمند بهینه‌سازی غیرخطی زمان‌بر است. همچنین این روش‌ها از ناپایداری‌های ناشی از تقریب تابع و یادگیری خارج از خط‌مشی رنج می‌برند.
- **روش‌های گرادیان خط‌مشی استاندارد (Vanilla Policy Gradient):** روش‌هایی نظیر REINFORCE [۲۵] و **A2C/A3C** [۲۶] مستقیماً پارامترهای خط‌مشی را بهینه‌سازی می‌کنند. با این حال، به دلیل وابستگی گرادیان به توزیع داده‌های جمع‌آوری‌شده، مجاز به اجرای تنها یک گام گرادیان به ازای هر نمونه داده بودند. انجام چندین گام بهینه‌سازی (Multiple Epochs) روی داده‌های یکسان، منجر به جابجایی شدید خط‌مشی جدید نسبت به خط‌مشی قدیم و در نتیجه سقوط فاجعه‌بار کارایی (Catastrophic Policy Collapse) می‌شد؛ امری که باعث ناکارآمدی شدید در تعداد نمونه‌ها (Sample Inefficiency) بود.

- **روش‌های بهینه‌سازی در ناحیه اطمینان (Trust Region Methods):** کاکاده و لنگفورد [۲۷] و سپس شولمن و همکاران با معرفی الگوریتم **TRPO** [۲۸] نشان دادند که با مقید ساختن تغییرات خط‌مشی از طریق واگرایی کولبک-لایبلر (KL-Divergence)، می‌توان بهبود یکنواخت کارایی (Monotonic Improvement) را تضمین کرد. با این وجود، **TRPO** به محاسبات مرتبه دوم ماتریس اطلاعات فیشر (Fisher Information Matrix)، ماتریس هسین و حل با الگوریتم گرادیان مزدوج (Conjugate Gradient) نیازمند بود. این امر نه تنها پیچیدگی محاسباتی بالایی ایجاد می‌کرد، بلکه با معماری‌های دارای لایه‌های مشترک میان خط‌مشی و ارزش و تکنیک‌های نویزدار مانند Dropout ناسازگار بود.

هدف بنیادین **PPO**، دستیابی به ثبات، قابلیت اطمینان و نرخ همگرایی بالای **TRPO**، صرفاً با تکیه بر بهینه‌سازهای مشتق اول (First-Order Optimizers) نظیر بهینه‌ساز Adam [۲۹] و با سادگی پیاده‌سازی بالا است.

مبانی نظری و اشتقاق‌های کامل ریاضی

یک فرآیند تصمیم‌گیری مارکوف (MDP) با چندتایی متناهی یا نامتناهی $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{P}, r, \gamma, \rho_0)$ مدل‌سازی می‌شود که در آن $\mathcal{S}$ فضای حالات، $\mathcal{A}$ فضای اعمال، $\mathcal{P}(s'|s,a)$ تابع احتمال انتقال، $r(s,a)$ تابع پاداش، $\gamma \in [0,1)$ ضریب تخفیف پاداش و $\rho_0(s_0)$ توزیع حالت اولیه است.

توزیع یک مسیر (Trajectory) $\tau = (s_0, a_0, s_1, a_1, \ldots, s_T)$ تحت خط‌مشی تصادفی پارامتریزه شده $\pi_\theta(a|s)$ برابر است با:

$$P(\tau;\theta) = \rho_0(s_0) \prod_{t=0}^{T-1} \pi_\theta(a_t|s_t)\mathcal{P}(s_{t+1}|s_t,a_t) \tag{۱}$$

هدف بهینه‌سازی، بیشینه‌سازی امید ریاضی بازده کل است:

$$J(\theta) = \mathbb{E}_{\tau\sim P(\cdot;\theta)}[R(\tau)] = \int_\tau P(\tau;\theta)R(\tau)\,d\tau \tag{۲}$$

که در آن $R(\tau) = \sum_{t=0}^{T} \gamma^t r(s_t,a_t)$ است. با اعمال عملگر گرادیان و استفاده از قاعده مشتق لگاریتمی ($\nabla_\theta P = P\nabla_\theta \log P$):

$$\begin{aligned}\nabla_\theta J(\theta) &= \nabla_\theta \int P(\tau;\theta)R(\tau)\,d\tau = \int \nabla_\theta P(\tau;\theta)R(\tau)\,d\tau \\ &= \int P(\tau;\theta)\nabla_\theta \log P(\tau;\theta)R(\tau)\,d\tau = \mathbb{E}_{\tau\sim\pi_\theta}\left[\nabla_\theta \log P(\tau;\theta)R(\tau)\right]\end{aligned} \tag{۳}$$

از آنجا که جملات انتقال وضعیت محیط $\mathcal{P}$ به پارامتر θ وابسته نیستند، داریم $\nabla_\theta \log P(\tau;\theta) = \sum_{t=0}^{T} \nabla_\theta \log \pi_\theta(a_t|s_t)$. با بهره‌گیری از اصل علیت و کسر خط مبنای مستقل از عمل $b(s_t) = V(s_t)$ جهت کاهش واریانس، قضیه گرادیان خط‌مشی حاصل می‌شود:

$$\hat{g} = \hat{\mathbb{E}}_t\left[\nabla_\theta \log \pi_\theta(a_t|s_t)\hat{A}_t\right] \tag{۴}$$

که در آن $\hat{A}_t = Q(s_t,a_t) - V(s_t)$ برآوردگر تابع مزیت (Advantage Function) است.

با استفاده از مفهوم نمونه‌برداری ترجیحی (Importance Sampling)، هدف خط‌مشی برای داده‌های جمع‌آوری‌شده توسط $\pi_{\theta_{\text{old}}}$ به صورت تابع هدف جایگزین تکرار خط‌مشی محافظه‌کارانه (CPI) [۲۷] تعریف می‌شود:

$$L^{CPI}(\theta) = \hat{\mathbb{E}}_t\left[\frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{\text{old}}}(a_t|s_t)}\hat{A}_t\right] = \hat{\mathbb{E}}_t\left[r_t(\theta)\hat{A}_t\right] \tag{۵}$$

که در آن نسبت احتمالات $r_t(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{\text{old}}}(a_t|s_t)}$ بوده و $r_t(\theta_{\text{old}}) = 1$ است.

توابع هدف در الگوریتم PPO

الگوریتم **PPO** دو فرمول‌بندی برای محدود ساختن تغییرات گام خط‌مشی ارائه می‌دهد:

۱. تابع هدف با برش جایگزین (Clipped Surrogate Objective): رویکرد اصلی و پیشنهادی مقاله، استفاده از تابع هدف زیر است:

$$L^{\text{CLIP}}(\theta) = \hat{\mathbb{E}}_t \left[\min \left(r_t(\theta)\hat{A}_t,\ \text{clip}(r_t(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t \right) \right] \tag{۶}$$

که در آن ϵ یک ابرپارامتر (معمولاً $\epsilon = 0.2$) است. این تابع هدف یک کران پایین بدبینانه (Pessimistic Lower Bound) را تشکیل می‌دهد:

- **در شرایط مزیت مثبت ($\hat{A}_t > 0$):** اگر عمل انجام‌شده بهتر از حد انتظار باشد، افزایش احتمال آن مفید است؛ اما به محض اینکه نسبت احتمال از $1+\epsilon$ فراتر رود، گرادیان صفر شده و از تغییرات مفرط جلوگیری می‌کند.
- **در شرایط مزیت منفی ($\hat{A}_t < 0$):** اگر عمل بدتر از حد انتظار باشد، احتمال آن باید کاهش یابد؛ اگر نسبت احتمال از $1-\epsilon$ کمتر شود، مقدار بریده شده و گرادیان صفر می‌شود. اما اگر خط‌مشی اشتباهاً احتمال این عمل بد را افزایش دهد ($r_t \gg 1$)، مقدار بدون برش انتخاب شده و جریمه کامل به خط‌مشی اعمال می‌شود.

۲. تابع هدف با ضریب جریمه تطبیقی (Adaptive KL Penalty): به عنوان رویکرد جایگزین، جریمه واگرایی KL مستقیماً در تابع هدف لحاظ شده و ضریب β به صورت دینامیک به‌روزرسانی می‌شود:

$$L^{\text{KLPEN}}(\theta) = \hat{\mathbb{E}}_t \left[r_t(\theta)\hat{A}_t - \beta\, D_{\text{KL}}\left(\pi_{\theta_{\text{old}}}(\cdot|s_t) \parallel \pi_\theta(\cdot|s_t)\right) \right] \tag{۷}$$

در هر تکرار، پس از محاسبه $d = \hat{\mathbb{E}}_t[D_{\text{KL}}(\pi_{\theta_{\text{old}}} \parallel \pi_\theta)]$، ضریب جریمه به این صورت تطبیق می‌یابد:

$$\beta \leftarrow \begin{cases} \beta/2 & \text{if} d < d_{\text{targ}}/1.5 \\ \beta \times 2 & \text{if} d > d_{\text{targ}} \times 1.5 \end{cases} \tag{۸}$$

برآوردگر مزیت تعمیم‌یافته (GAE) و تابع هزینه یکپارچه

برای کاهش واریانس، از برآوردگر مزیت تعمیم‌یافته (GAE) منقطع [۳۰] استفاده می‌شود:

$$\hat{A}_t = \sum_{l=0}^{T-t-1} (\gamma\lambda)^l \delta_{t+l}^V \quad \text{where} \quad \delta_t^V = r_t + \gamma V(s_{t+1}) - V(s_t) \tag{۹}$$

برای شبکه‌های دارای پارامترهای مشترک میان خط‌مشی و تابع ارزش، تابع هزینه سراسری بهینه‌سازی زیر بیشینه می‌گردد:

$$L_t^{\text{CLIP+VF+S}}(\theta) = \hat{\mathbb{E}}_t \left[L_t^{\text{CLIP}}(\theta) - c_1 \left(V_\theta(s_t) - V_t^{\text{targ}} \right)^2 + c_2 S[\pi_\theta](s_t) \right] \tag{۱۰}$$

که در آن $S[\pi_\theta]$ آنتروپی شانون برای تشویق به اکتشاف (Exploration) و c_1, c_2 ضرایب وزنی هستند.

تحلیل‌های تئوری اطلاعات و آمار

بر اساس بسط تیلور مرتبه دوم واگرایی KL حول نقطه θ_{old} داریم:

$$D_{\text{KL}}(\pi_{\theta_{\text{old}}} \parallel \pi_\theta) \approx \frac{1}{2}(\theta - \theta_{\text{old}})^T F(\theta - \theta_{\text{old}}) \tag{۱۱}$$

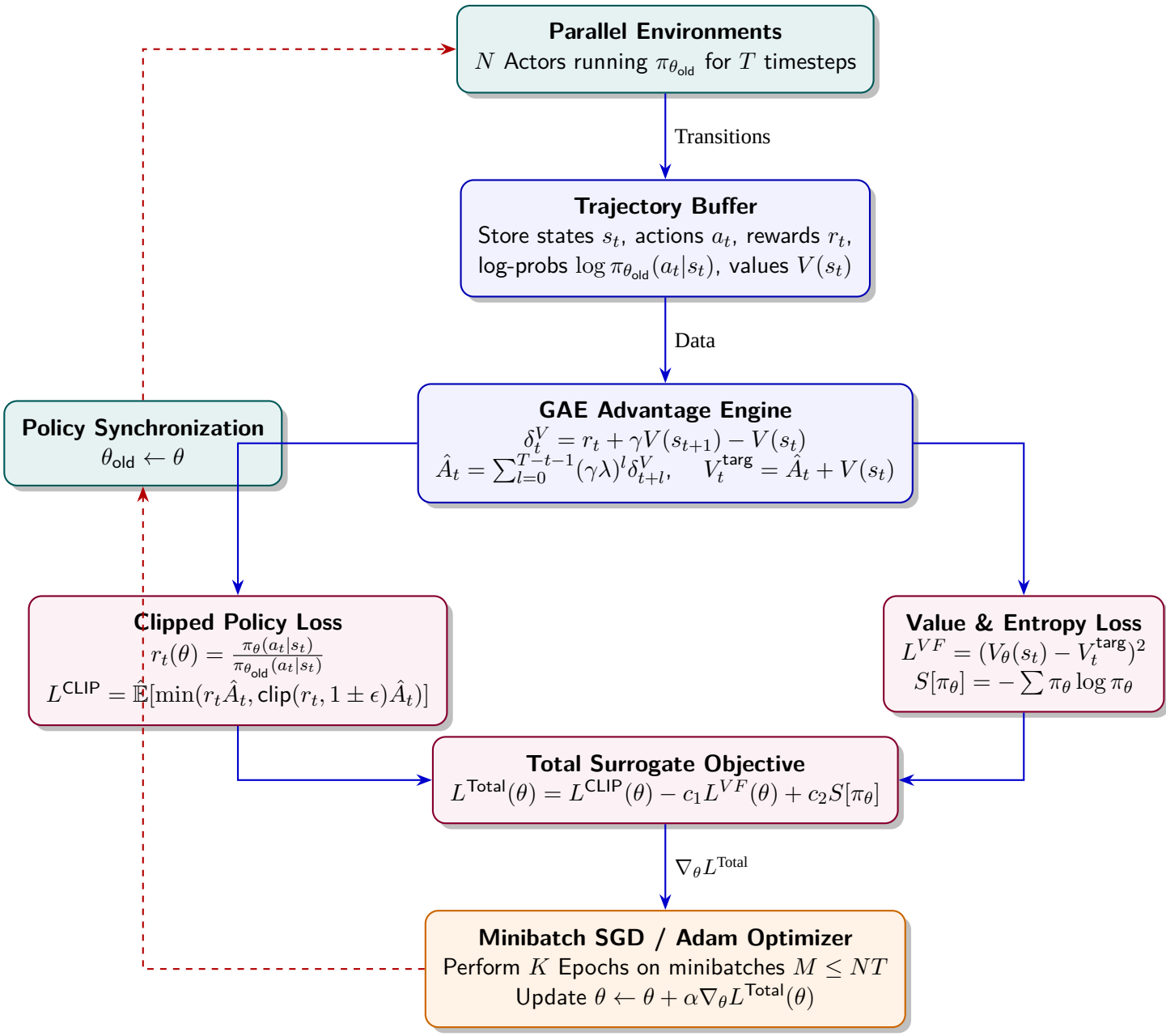
که در آن $F = \mathbb{E}[\nabla_\theta \log \pi_\theta \nabla_\theta \log \pi_\theta^T]$ ماتریس اطلاعات فیشر است. تابع برش در **PPO** تضمین می‌کند که فاصله روی منیفلد ریمانی توزیع‌ها بدون نیاز به تشکیل و معکوس کردن صریح ماتریس F در محدوده‌ای امن محبوس بماند. همچنین با محدود کردن $r_t(\theta) \in [1-\epsilon, 1+\epsilon]$، واریانس بازوزن‌دهی نمونه‌برداری ترجیحی ($\text{Var}_Q[P/Q \cdot f]$) کنترل شده و از انفجار واریانس در بهینه‌سازی چند دوره‌ای جلوگیری می‌شود.

دیاگرام معماری و جریان داده‌های الگوریتم

شکل ۱، ساختار تعامل با محیط، محاسبه مزیت، تشکیل توابع هدف و چرخه به‌روزرسانی چند دوره‌ای در الگوریتم **PPO** را نشان می‌دهد.

شبه‌کد الگوریتم PPO

شبه‌کد استاندارد ساختار Actor-Critic الگوریتم **PPO** در الگوریتم ۱ آورده شده است:



شکل ۱: دیاگرام جامع جریان داده‌ها، محاسبه مزیت و بهینه‌سازی چند دوره‌ای در الگوریتم **PPO**.

Algorithm 1 Proximal Policy Optimization (PPO), Actor-Critic Style

Input: Initial policy parameters θ_0, initial value function parameters ϕ_0, clipping threshold ϵ, GAE parameters γ, λ, epoch count K, minibatch size M, coefficients c_1, c_2.
for iteration $= 1, 2, \ldots$ **do**
  for actor $= 1, \ldots, N$ **do**
    Run policy $\pi_{\theta_{\text{old}}}$ in environment for T timesteps.
    Compute predictions $V_\phi(s_t)$ and log-probabilities $\log \pi_{\theta_{\text{old}}}(a_t|s_t)$.
    Compute TD residuals: $\delta_t^V = r_t + \gamma V_\phi(s_{t+1}) - V_\phi(s_t)$.
    Compute GAE estimates: $\hat{A}_t = \sum_{l=0}^{T-t-1} (\gamma\lambda)^l \delta_{t+l}^V$.
    Compute state-value targets: $V_t^{\text{targ}} = \hat{A}_t + V_\phi(s_t)$.
  end for
  Collect all NT transitions: $\mathcal{D} = \{(s_t, a_t, r_t, \log \pi_{\theta_{\text{old}}}(a_t|s_t), \hat{A}_t, V_t^{\text{targ}})\}$.
  Normalize advantage values: $\hat{A}_t \leftarrow \frac{\hat{A}_t - \mu(\hat{A})}{\sigma(\hat{A}) + 10^{-8}}$.
  for epoch $= 1, \ldots, K$ **do**
    Shuffle dataset $\mathcal{D}$ and partition into minibatches of size M.
    for each minibatch $\mathcal{B} \subset \mathcal{D}$ **do**
      Compute probability ratio: $r_t(\theta) = \exp\left(\log \pi_\theta(a_t|s_t) - \log \pi_{\theta_{\text{old}}}(a_t|s_t)\right)$.
      Compute policy surrogate loss:
        $L^{\text{CLIP}}(\theta) = \frac{1}{|\mathcal{B}|} \sum_{t \in \mathcal{B}} \min\left(r_t(\theta)\hat{A}_t,\ \text{clip}(r_t(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t\right)$.
      Compute value function squared error loss:
        $L^{\text{VF}}(\phi) = \frac{1}{|\mathcal{B}|} \sum_{t \in \mathcal{B}} \left(V_\phi(s_t) - V_t^{\text{targ}}\right)^2$.
      Compute entropy bonus: $S[\pi_\theta] = \frac{1}{|\mathcal{B}|} \sum_{t \in \mathcal{B}} \mathbb{H}(\pi_\theta(\cdot|s_t))$.
      Maximize total objective using Adam:
        $\theta, \phi \leftarrow \text{Adam}\left(\nabla_{\theta,\phi}\left[L^{\text{CLIP}}(\theta) - c_1 L^{\text{VF}}(\phi) + c_2 S[\pi_\theta]\right]\right)$.
    end for
  end for
  $\theta_{\text{old}} \leftarrow \theta$
end for

نتایج تجربی و ارزیابی بنچمارک‌ها

کارایی **PPO** در مقایسه با انواع روش‌های بهینه‌سازی در جدول ۱ و روی بازی‌های آتاری در جدول ۲ خلاصه شده است:

ابرپارامترهای استاندارد گزارش‌شده در مقاله

جداول ۳ و ۴ مقادیر دقیق ابرپارامترهای بهینه‌شده برای محیط‌های مختلف را نمایش می‌دهند:

جمع‌بندی

الگوریتم **PPO** به واسطه تابع برش بدبینانه L^{CLIP}، بدون نیاز به محاسبات سنگین ماتریس فیشر و با استفاده مستقیم از گرادیان مرتبه اول Adam، رفتار ناحیه اطمینان را شبیه‌سازی کرده و امروزه به عنوان پایه‌ای‌ترین و مطمئن‌ترین الگوریتم یادگیری تقویتی در زمینه‌های مختلف از کنترل حرکت ربات‌های سه‌بعدی [۳۱] تا هم‌ترازسازی مدل‌های زبانی بزرگ از طریق بازخورد انسانی (RLHF) شناخته می‌شود.

- sda

جدول ۱: مقایسه عملکرد انواع توابع هدف جایگزین در ۷ محیط پیوسته MuJoCo (برگرفته از مقاله مرجع [۲۰]).

فرمول‌بندی الگوریتم / تابع هدف	میانگین امتیاز نرمال‌شده
بدون برش و بدون جریمه (No clipping or penalty)	-۳۹.۰
برش با $\epsilon = 0.1$ (Clipping)	۷۶.۰
برش با $\epsilon = 0.2$ (Clipping)	**۸۲.۰**
برش با $\epsilon = 0.3$ (Clipping)	۷۰.۰
جریمه تطبیقی با $d_{\text{targ}} = 0.003$ (Adaptive KL)	۶۸.۰
جریمه تطبیقی با $d_{\text{targ}} = 0.01$ (Adaptive KL)	۷۴.۰
جریمه تطبیقی با $d_{\text{targ}} = 0.03$ (Adaptive KL)	۷۱.۰
جریمه ثابت با $\beta = 0.3$ (Fixed KL)	۶۲.۰
جریمه ثابت با $\beta = 1.0$ (Fixed KL)	۷۱.۰
جریمه ثابت با $\beta = 3.0$ (Fixed KL)	۷۲.۰
جریمه ثابت با $\beta = 10.0$ (Fixed KL)	۶۹.۰

جدول ۲: تعداد بازی‌های برده‌شده توسط هر الگوریتم در بنچمارک ۴۹ تایی آتاری (ALE) [۲۰].

معیار ارزیابی	A2C	ACER	PPO	تساوی
(۱) میانگین پاداش در کل طول آموزش	۱	۱۸	**۳۰**	۰
(۲) میانگین پاداش در ۱۰۰ اپیزود پایانی	۱	**۲۸**	۱۹	۱

جدول ۳: ابرپارامترهای **PPO** در بنچمارک رباتیک پیوسته MuJoCo (۱ میلیون گام).

ابرپارامتر	مقدار
افق زمانی داده‌برداری (T)	۲۰۴۸
نرخ یادگیری بهینه‌ساز Adam	3×10^{-4}
تعداد دور بهینه‌سازی در هر آپدیت (K)	۱۰
اندازه دسته کوچک (M)	۶۴
ضریب تخفیف پاداش (γ)	۹۹.۰
پارامتر برآوردگر مزیت (λ)	۹۵.۰

جدول ۴: ابرپارامترهای **PPO** در بنچمارک بازی‌های آتاری (Atari).

ابرپارامتر	مقدار
افق زمانی داده‌برداری (T)	۱۲۸
نرخ یادگیری Adam	$2.5 \times 10^{-4} \times \alpha$
تعداد عامل‌های موازی (N)	۸
تعداد دورها (K)	۳
اندازه دسته کوچک (M)	$32 \times 8 = 256$
ضریب تخفیف پاداش (γ)	۹۹.۰
پارامتر λ در GAE	۹۵.۰
آستانه برش (ϵ)	$0.1 \times \alpha$
ضریب خطای تابع ارزش (c_1)	۰.۱
ضریب پاداش آنتروپی (c_2)	۰۱.۰

- sd**a**
- sd**a**
- sd**a**
- sd**a**
- sd**a**
- sd**a**

۸ منابع

[1] Houyi Li, Wenzhen Zheng, Qiufeng Wang, Zhenyu Ding, Haoying Wang, Zili Wang, Shijie Xuyang, Ning Ding, Shuigeng Zhou, Xiangyu Zhang, et al. Predictable scale: Part ii, farseer–a refined scaling law in large language models. *arXiv preprint arXiv:2506.10972*, 2025.

[2] Zhengyu Chen, Siqi Wang, Teng Xiao, Yudong Wang, Shiqi Chen, Xunliang Cai, Junxian He, and Jingang Wang. Revisiting scaling laws for language models: The role of data quality and training strategies. *arXiv preprint*, 2025.

[3] Sean McLeish, John Kirchenbauer, David Yu Miller, Siddharth Singh, Abhinav Bhatele, Micah Goldblum, Ashwinee Panda, and Tom Goldstein. Gemstones: A model suite for multi-faceted scaling laws. *arXiv preprint arXiv:2502.06857*, 2025.

[4] Song Bian, Tao Yu, Shivaram Venkataraman, and Youngsuk Park. Scaling laws meet model architecture: Toward inference-efficient llms. *arXiv preprint arXiv:2510.18245*, 2025.

[5] Yizhou Liu, Sara Kangaslahti, Ziming Liu, and Jeff Gore. Inverse depth scaling from most layers being similar. *arXiv preprint arXiv:2602.05970*, 2026.

[6] Wenjie Sun, Jinning Yang, Shuai Zhang, and Mengnan Du. Law of neural interaction: Depth–width shape, interaction efficiency, and generalization. *arXiv preprint arXiv:2605.27989*, 2026.

[7] Nandan Kumar Jha and Brandon Reagen. Spectral scaling laws in language models: How effectively do feed-forward networks use their latent space? *arXiv preprint arXiv:2510.00537*, 2025.

[8] Maximilian Beck, Kajetan Schweighofer, Sebastian Böck, Sebastian Lehner, and Sepp Hochreiter. xlstm scaling laws: Competitive performance with linear time-complexity. *arXiv preprint arXiv:2510.02228*, 2026.

[9] Anonymous. On the recall scaling laws in mamba: A theoretical and mechanistic study via hashing. *arXiv preprint*, 2026.

[10] Changxin Tian, Kunlong Chen, Jia Liu, Ziqi Liu, Zhiqiang Zhang, and Jun Zhou. Towards greater leverage: Scaling laws for efficient mixture-of-experts language models. *arXiv preprint arXiv:2507.17702*, 2025.

[11] Samira Abnar, Harshay Shah, Dan Busbridge, Alaaeldin El-Nouby, Josh Susskind, and Vimal Thilak. Parameters vs flops: Scaling laws for optimal sparsity for mixture-of-experts language models. *arXiv preprint arXiv:2501.12370*, 2025.

[12] Jan Ludziejewski, Maciej Pióro, Jakub Krajewski, Maciej Stefaniak, Michał Krutul, Jan Małaśnicki, Marek Cygan, Piotr Sankowski, Kamil Adamczewski, Piotr Miłoś, et al. Joint moe scaling laws: Mixture of experts can be memory efficient. *arXiv preprint arXiv:2502.05172*, 2025.

[13] Seng Pei Liew, Takuya Kato, and Sho Takase. Scaling laws for upcycling mixture-of-experts language models. *arXiv preprint arXiv:2502.03009*, 2025.

[14] Anirudh Subramanyam, Yuxin Chen, and Robert L Grossman. Scaling laws revisited: Modeling the role of data quality in language model pretraining. *arXiv preprint arXiv:2510.03313*, 2026.

[15] Mustafa Shukor, Louis Bethune, Dan Busbridge, David Grangier, Enrico Fini, Alaaeldin El-Nouby, and Pierre Ablin. Scaling laws for optimal data mixtures. *arXiv preprint arXiv:2507.09404*, 2025.

[16] Francesco Cagnetta, Alessandro Favero, Antonio Sclocchi, and Matthieu Wyart. Scaling laws and representation learning in simple hierarchical languages: Transformers vs. convolutional architectures. *arXiv preprint arXiv:2505.07070*, 2025.

[17] William Held, David Hall, Percy Liang, and Diyi Yang. Relative scaling laws for llms. *arXiv preprint arXiv:2510.24626*, 2026.

[18] Baoqing Yue, Jinyuan Zhou, Zixi Wei, Jingtao Zhan, Qingyao Ai, and Yiqun Liu. Relative-based scaling law for neural language models. *arXiv preprint arXiv:2510.20387*, 2025.

[19] Chaojun Xiao, Jie Cai, Weilin Zhao, Biyuan Lin, Guoyang Zeng, Jie Zhou, Zhi Zheng, Xu Han, Zhiyuan Liu, and Maosong Sun. Densing law of llms. *Nature Machine Intelligence*, pages 1–11, 2025.

[20] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[21] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015.

[22] Marc G Bellemare, Yavar Naddaf, Joel Veness, and Michael Bowling. The arcade learning environment: An evaluation platform for general agents. *Twenty-Fourth International Joint Conference on Artificial Intelligence (IJCAI)*, 2015.

[23] Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. Openai gym. *arXiv preprint arXiv:1606.01540*, 2016.

[24] Yan Duan, Xi Chen, Rein Houthooft, John Schulman, and Pieter Abbeel. Benchmarking deep reinforcement learning for continuous control. *International Conference on Machine Learning (ICML)*, pages 1329–1338, 2016.

[25] Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8(3-4):229–256, 1992.

[26] Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. Asynchronous methods for deep reinforcement learning. *International Conference on Machine Learning (ICML)*, pages 1928–1937, 2016.

[27] Sham Kakade and John Langford. Approximately optimal approximate reinforcement learning. *International Conference on Machine Learning (ICML)*, 2:267–274, 2002.

[28] John Schulman, Sergey Levine, Philipp Moritz, Michael Jordan, and Pieter Abbeel. Trust region policy optimization. *International Conference on Machine Learning (ICML)*, pages 1889–1897, 2015.

[29] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[30] John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. *arXiv preprint arXiv:1506.02438*, 2015.

[31] Nicolas Heess, Srinivasan Sriram, Jay Lemmon, Josh Merel, Greg Wayne, Yuval Tassa, Tom Erez, Ziyu Wang, SM Eslami, Martin Riedmiller, et al. Emergence of locomotion behaviours in rich environments. *arXiv preprint arXiv:1707.02286*, 2017.

[32] Ziyu Wang, Victor Bapst, Nicolas Heess, Volodymyr Mnih, Remi Munos, Koray Kavukcuoglu, and Nando de Freitas. Sample efficient actor-critic with experience replay. *arXiv preprint arXiv:1611.01224*, 2016.