目录

K2 预训练

攻克"稳定性"与"效率"两大难题

预训练概述

训练万亿模型，核心挑战在于：如何在有限的高质量数据下提升**词元效率 (Token-efficient)**，以及如何保证超大规模训练的**稳定性 (Stability)**。

K2 在 **15.5 万亿 Tokens** 的训练数据上进行预训练，通过创新的技术方案成功解决了这两大难题。

难题一：稳定性 (MuonClip / QK-Clip)

在 Kimi K2 这样万亿参数规模且在 15.5 万亿海量 Token 上进行预训练的工程中，**训练稳定性**是压倒一切的前提。一次训练崩溃（即"损失尖峰"）可能意味着数万美元的成本浪费和昂贵的训练回滚。

问题的根源：Logit 爆炸

注意力 Logit 是在进行 Softmax 操作之前，Query (Q) 和 Key (K) 向量点积的结果。

- K2 选用了 **Muon** 优化器，因为它具有极高的词元效率，性能优于 AdamW
- **副作用**：Muon 在规模化应用时，会导致"注意力 Logit 爆炸" (exploding attention logits)，Logit 值可超过 1000
- 当这个值变得极大时，会导致 Softmax 的输出变得极其尖锐（接近 one-hot 分布）
- 梯度会随之变得极大或极小（梯度爆炸或消失），最终导致训练过程出现**损失尖峰 (loss spikes)** 甚至完全**发散 (divergence)**

Muon 优化器算法讲解

关于 Muon 优化器的详细算法讲解，可以参考 YouTube 视频： https://youtu.be/bO5nvE289ec?si=N9y1ML1rhBhpEO79

现有方案的局限

在提出 QK-Clip 之前，K2 团队评估了两种已有的缓解策略，均不适用：

Logit Soft-cap (直接限制 Logit)

这种方法**治标不治本**。虽然最终的 Logit 被限制了，但在此之前的 Q 和 K 点积结果可能已经非常大，这本身就可能在反向传播中引入数值问题。

Query-Key 归一化 (QK-Norm)

通过对 Q 和 K 向量进行归一化来控制点积的大小。这个方法不适用于 Kimi K2 所采用的**多头潜在注意力 (Multi-head Latent Attention, MLA)** 架构。 原因是，在 MLA 中，Key 矩阵在推理过程中**不是被完全实例化 (materialized) 的**， 因此无法直接应用常规的 QK-Norm。

K2 的创新：QK-Clip

K2 团队提出了 **QK-Clip**，一种新颖的权重裁剪机制，其设计思想是**从源头上控制 Logit 的增长**。

核心思想：不干预 Logit，而是约束产生 Logit 的权重

工作方式

- 在优化器更新完权重之后 (post-update) 工作
- 不干扰当前训练步骤的前向和反向传播计算
- 将 Logit 作为一种"事后"的监控信号
- 通过重新缩放 (rescaling) Query (W_q) 和 Key (W_k) 的投影权重，来间接限制 Logit 的增长

最大 Logit (S^h_{max})

为了决定何时以及如何干预，QK-Clip 需要一个监控指标。这个指标就是每个注意力头 h 在当前批次 (batch) B 中的最大 Logit 值，定义如下：

$$S^h_{\max} = \frac{1}{\sqrt{d}} \max_{X \in B} \max_{i,j} Q^h_i (K^h_j)^T$$

这个公式计算了单个注意力头在整个批次中所有 token 对之间点积的最大值。

h 代表注意力头的索引	d 是注意力头的维度
B 代表当前训练批次的所有样本	Q^h_i 和 K^h_j 分别是样本中第 i 个 token 的 Query 向量和第 j 个 token 的 Key 向量

干预机制：按需、按头的权重缩放

触发条件

当某个头的 S^h_{max} 超过了预设的阈值 tau (在 Kimi K2 的训练中，$tau = 100$)，QK-Clip 机制就会被触发。

最小化干预原则

K2 团队观察到，实践中只有一小部分注意力头会出现 Logit 爆炸。因此，为了避免不必要的干扰，QK-Clip 采用了**按头 (per-head) 的缩放策略**，而不是对整个层的所有头进行统一处理。

缩放因子计算：每个头独立计算其缩放因子 γ_h

- 计算缩放因子 $\gamma_h = \min(1, \tau / S^h_{\max})$。
  当 $S^h_{\max} \leq \tau$ 时，$\gamma_h = 1$，权重不发生变化。当 $S^h_{\max} > \tau$ 时，$\gamma_h < 1$，权重将被缩小。
- **精确缩放**：按比例 γ_h 缩放 W_q 和 W_k 的特定组件，从源头控制 Logit 增长。

针对 MLA 架构的精细化应用

在 MLA 架构中, QK-Clip 精确地只对非共享的组件进行缩放, 以避免跨头影响:

- **头特定组件** (q^C, k^C) 按 $\sqrt{\gamma_h}$ 进行缩放 (使用平方根是为了将缩放效果均等地分配给 Q 和 K)
- **头特定旋转组件** (q^R) 按 γ_h 进行缩放
- **共享旋转组件** (k^R) 保持不变, 以避免对其他注意力头产生副作用

MuonClip 优化器的诞生

K2 团队将以下四个部分整合在一起，形成了一个完整的、稳定且高效的优化器，命名为 **MuonClip**：

原始的 Muon 优化器

确保模型训练中的词元处理效率。

权重衰减

通过正则化防止模型过拟合。

一致性 RMS 匹配

维持不同层之间梯度幅度的稳定性。

QK-Clip 机制

有效抑制 Logit 爆炸，保障训练过程稳定。

实验效果

- **效果立竿见影**：如图展示了 Kimi K2 在使用 MuonClip（$\tau = 100$）训练时的最大 Logit 曲线。可以看到，在训练初期，Logit 值迅速增长并被**精确地限制在 100**。

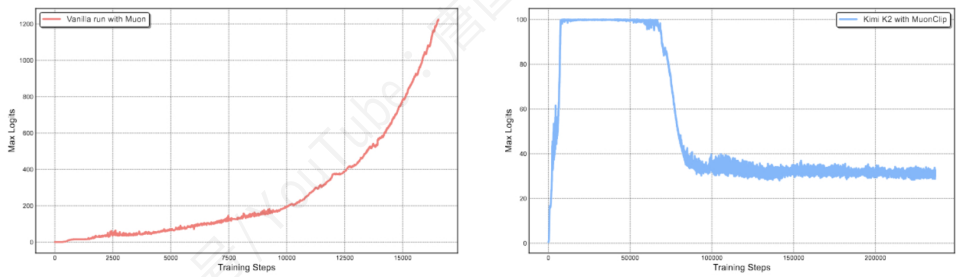

Figure 2: Left: During a mid-scale training run, attention logits rapidly exceed 1000, which could lead to potential numerical instabilities and even training divergence. Right: Maximum logits for Kimi K2 with MuonClip and $\tau = 100$ over the entire training run. The max logits rapidly increase to the capped value of 100, and only decay to a stable range after approximately 30% of the training steps, demonstrating the effective regulation effect of QK-Clip.

- **自适应与自失效 (Self-Disabling)**：随着训练的进行（约 30% 之后），模型的权重逐渐调整到更稳定的状态，最大 Logit 值自然地**回落到 100 以下**。此时，QK-Clip 的触发条件不再满足 ($\gamma_h = 1$)，它就**自动停止了干预**。这体现了其"最小化干预"的设计哲学——只在需要时发挥作用。

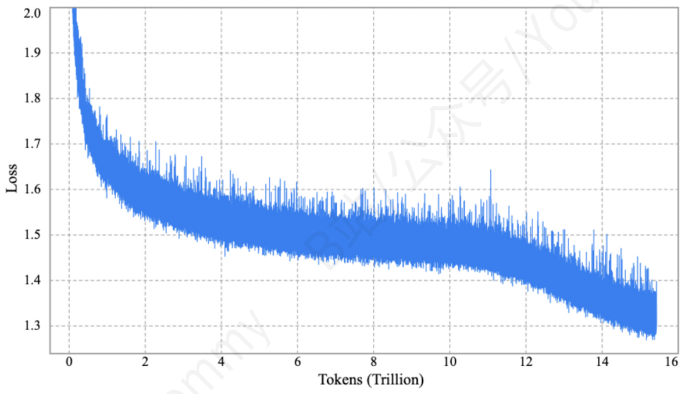

Figure 3: Per-step training loss curve of Kimi K2, **without smoothing or sub-sampling**. It shows no spikes throughout the entire training process. Note that we omit the very beginning of training for clarity.

- **最终结果**：MuonClip 成功地使 Kimi K2 在 **15.5 万亿 Tokens** 的预训练过程中实现了"**零损失尖峰 (zero loss spike)**"，保证

了万亿模型训练的稳定性和效率。

难题二：词元效率 (Data Rephrasing)

挑战： 高质量人类数据日益稀缺

K2 的方案： 设计"**数据改写 (Rephrasing)**"流水线，在不过拟合的前提下， 充分利用高质量数据，增加词元效用。

数据改写流程

> **1. 提示工程**
>
> 引导 LLM 以多种风格和视角重写原文

> **2. 分块生成**
>
> 将长文档分块改写再拼接，保持全局连贯

> **3. 忠实度验证**
>
> 对比改写段落与原文的语义一致性，确保内容准确

Table 1: SimpleQA Accuracy under three rephrasing-epoch configurations

# Rephrasings	# Epochs	SimpleQA Accuracy
0 (raw wiki-text)	10	23.76
1	10	27.39
10	1	28.94

成果： 实验证明，数据改写比简单地重复训练（多轮次）能带来更显著的性能提升。