RankMe. $\alpha_{\text{ReQ}}$ and RankMe thus provide related metrics of representation geometry, though unlike RankMe, $\alpha_{\text{ReQ}}$ does not change with the model's feature dimensionality, $d$.

### 2.2. Quantifying Distributional Memorization and Generalization via n-gram Alignment

To dissect how LLMs utilize their pretraining corpus $\mathcal{D}$, we differentiate *distributional memorization*, i.e. how aligned are LLM output probabilities with n-gram frequencies in $\mathcal{D}$, from *distributional generalization*, i.e. LLM capabilities beyond such statistics (Liu et al., 2024). To quantify the alignment with n-gram statistics, we use the $\infty$-gram language model (LM) which uses the largest possible value of $n$ for predicting the next token probability. Briefly, an $\infty$-gram LM can be viewed as a generalized version of an $n$-gram LM which starts with $n = \infty$, and then performs backoff till the $n$-gram count in $\mathcal{D}$ is non-zero (Liu et al., 2024). Consequently, the output probability of the $\infty$-gram LM for each token is dependent on its longest existing prefix in $\mathcal{D}$.

The distributional memorization metric is defined as the spearman rank correlation ($\rho_s$) between the $\infty$-gram LM outputs and the LLM outputs for all tokens in a target sequence (Wang et al., 2025). Formally, consider a concatenated sequence of instructions, $u$, question, $x$ and target, $y$, from a question-answering task, $\mathcal{T}$. Then, the distributional memorization is computed as:

$$Mem_{\infty}(LLM, \mathcal{D}, \mathcal{T}) \coloneqq \rho_s \left(\bar{P}_{\infty,\mathcal{D}}(y|u \oplus x), \bar{P}_{LLM}(y|u \oplus x)\right) \tag{2}$$

where $\bar{P}_{\cdot}(y|u \oplus x) \coloneqq \prod_{t_i \in y} P_{\cdot}(t_i|u \oplus x \oplus y_{[t_0:t_{i-1}]})$ denotes the joint likelihood of all tokens in $y$ and $P_{\cdot}()$ is the next token prediction distribution, as described above.

### 2.3. Post-Training Methodologies and Evaluation

**Supervised Fine-Tuning (SFT)** adapts pre-trained LLMs by further training on a curated dataset $\mathcal{D}_{\text{SFT}} = \{(x_i, y_i)\}_{i=1}^{N_{\text{SFT}}}$ typically consisting of instruction-response pairs. The standard objective is to minimize the negative log-likelihood of the target responses, effectively maximizing $P_\theta(y|x)$ for examples in $\mathcal{D}_{\text{SFT}}$. We evaluate the robustness of the SFT model by contrasting its performance on held-out examples from $\mathcal{D}_{\text{SFT}}$ (In-Distribution, ID) with its performance on examples from a related but distinct dataset $\mathcal{D}_{\text{OOD}}$ (Out-of-Distribution, OOD), which may vary in task, style, or complexity not present in $\mathcal{D}_{\text{SFT}}$ (Springer et al., 2025).

**Preference Alignment and Reasoning** : For alignment beyond SFT, we consider Direct Preference Optimization (DPO) (Rafailov et al., 2023) and Reinforcement Learning from Verifiable Rewards (RLVR). DPO refines an LLM policy $\pi_\theta$ based on a static dataset of human preferences $\mathcal{D}_{\text{pref}} = \{(x, y_w, y_l)\}$, where the response $y_w$ is preferred over $y_l$ for prompt $x$. It directly optimizes for preference satisfaction by minimizing the loss:

$$\mathcal{L}_{\text{DPO}}(\pi_\theta; \pi_{\text{ref}}) = -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}_{\text{pref}}} \left[\log \sigma \left(\hat{r}_\theta(x, y_w) - \hat{r}_\theta(x, y_l)\right)\right], \tag{3}$$

where $\hat{r}_\theta(x, y) = \beta \log(\pi_\theta(y|x)/\pi_{\text{ref}}(y|x))$ represents the implicit log-ratio of probabilities scaled by $\beta$ against a reference policy $\pi_{\text{ref}}$, and $\sigma(.)$ is the logistic function. Reinforcement Learning from Verifiable Rewards (RLVR), as applied in works like (Lambert et al., 2024) and (Shao et al., 2024), optimizes the LLM's policy $\pi_\theta$ to maximize the expected discounted cumulative reward, $J(\theta) = \mathbb{E}_{\tau\sim\pi_\theta}\left[\sum_{t=0}^{T} \gamma^t R_t\right]$, where $\tau = (s_0, a_0, \ldots, s_T, a_T)$ is a trajectory generated by actions $a_t \sim \pi_\theta(\cdot|s_t)$ in states $s_t$, $\gamma \in [0, 1]$ is

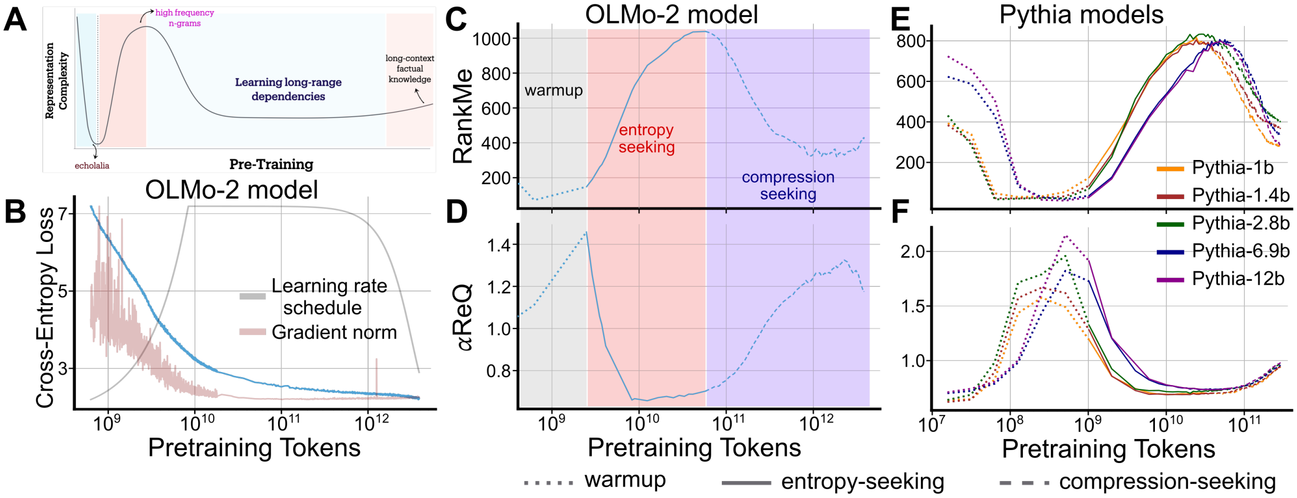

**Figure 2: Loss decreases monotonically, but representation geometry does not. (A)** Schematic from Fig 1, for the pretraining stage. **(B)** Cross-entropy loss, gradient norm and learning rate schedule during OLMo-2 7B model pretraining. **(C, D)** RankMe and $\alpha_{\text{ReQ}}$, respectively, for OLMo-2 7B model vary non-monotonically across pretraining, demonstrating three key phases: "warmup" , "entropy-seeking" , and "compression-seeking" . **(E, F)** Same as C,D, but for Pythia models, demonstrating the consistent existence of the three phases across model families and scales.

a discount factor, and $R_t = R(s_t, a_t)$ is the reward at time $t$. This optimization is typically performed using policy gradient algorithms (e.g., PPO). Critically, the reward $R_t$ in RLVR is derived from verifiable properties of the LLM's outputs, e.g. correctness on mathematical problems or passing unit tests.

**Performance with pass@k**: To evaluate problem-solving efficacy and generative exploration, particularly for RLVR-tuned models, we employ the pass@k metric (Kulal et al., 2019). For a given problem, k independent responses are stochastically generated from the model; the problem is deemed solved if at least one response constitutes a verifiable solution. Since direct estimation of pass@k can exhibit high variance, we utilize the unbiased estimator (Chen et al., 2021; Yue et al., 2025):

$$\texttt{pass@k} = \mathbb{E}_{P_i}\left[1 - \frac{\binom{N-c_i}{k}}{\binom{N}{k}}\right] \tag{4}$$

where, N samples are generated for each problem $P_i$ , and $c_i$ denotes the count of correct solutions among them (parameters for this work are N=512 and k $\leq$ 256).

## 3. Probing the representation geometry of language models

To study LLM representation geometry at intermediate stages of the training lifecycle, we analyze checkpoints from three publicly released model suites. We defer additional details on the model architecture, dataset and training run to Section A.

- **OLMo framework** Groeneveld et al. (2024); OLMo et al. (2024); Lambert et al. (2024): Developed by AI2, OLMo & OLMo-2 family of models provide intermediate checkpoints across different model sizes – 1B, 7B and 13B. We focused on intermediate checkpoints available for the OLMo-2 7B and 1B models throughout their $\sim 4T$ token training run.

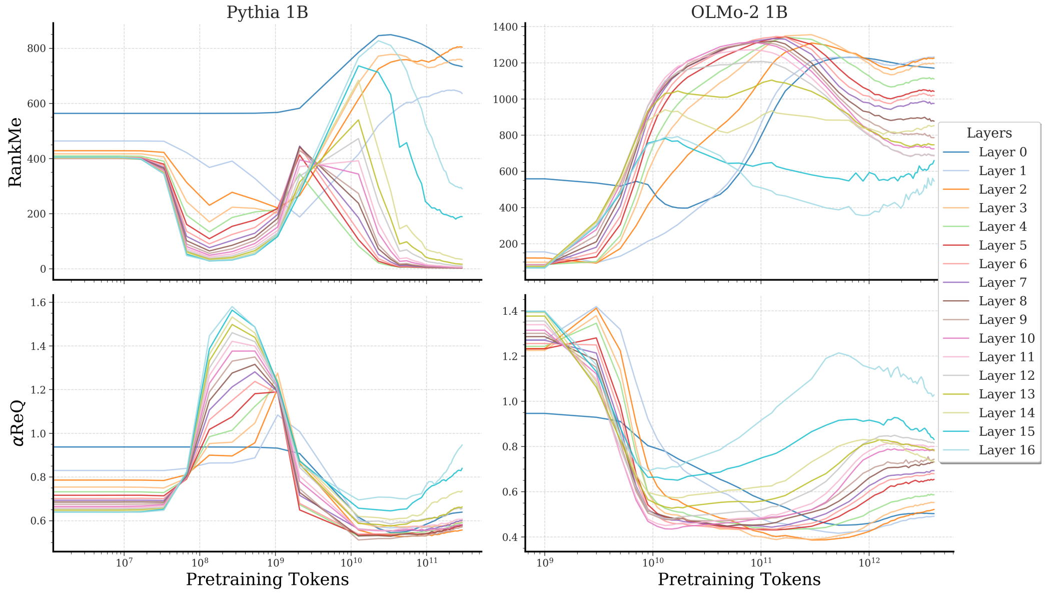

**Figure 3**: **Layerwise evolution mirrors the three phases.** Spectral metrics (RankMe and $\alpha_{\text{ReQ}}$) computed across intermediate layers during pretraining show that the three-phase pattern is consistent across network depth, justifying the use of last-layer representations for tracking global geometric dynamics. See Appendix for additional robustness analyses across samples, sequence lengths, and datasets.

- **Pythia suite** Biderman et al. (2023): Developed by EleutherAI, this suite consists of models ranging from 70M to 12B parameters, all trained on the Pile dataset (Gao et al., 2020) using the same data ordering and hyperparameters across scales. We analyzed the intermediate checkpoints available at various intermediate training steps for 1B+ models.
- **Tülu-3.1 models** (Wang et al., 2024): Developed by AI2, this suite contains instruction following $8B$ LLaMA-based models parameters, that were post-trained with state-of-the-art recipes. We analyzed checkpoints from all post-training stages of the model.

### 3.1. Phases of pretraining: Non-monotonic changes in representation geometry

During the LLM pretraining stage, standard metrics used for identifying optimization instabilities, e.g. loss or gradient norms, decrease near-monotonically. While useful to practitioners while determining successful recipes for pretraining large models, these metrics carry limited information about the model capabilities and downstream behavior. We demonstrate, on the contrary, that the high-dimensional representation geometry metrics undergo non-monotonic changes. (And, later we demonstrate that these changes correlate with downstream performance).

Figure 2 illustrates this contrasting trend between the optimization metrics and representation geometry metrics during the pretraining of aforementioned family of LLMs. Specifically, we measured the RankMe (Garrido et al., 2023) and $\alpha_{\text{ReQ}}$ (Agrawal et al., 2022) metrics on the LLM's last layer representation of the last token while processing sequences from the FineWeb dataset (Penedo et al., 2024), and observed that there exist three distinct phases during the pretraining stage. Initially, there is a "warmup" phase, coinciding with the learning rate ramp-up, exhibiting a rapid collapse of the representations along the dominant data manifold directions. This collapse manifests in repetitive, non-contextual outputs