LAYER COLLAPSE IN DIFFUSION LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Diffusion language models (DLMs) have recently emerged as competitive alternatives to autoregressive (AR) language models, yet the differences in their activation dynamics remain poorly understood. We characterise these dynamics in LLaDA-8B and identify a striking layer-collapse property: early layers exhibit highly similar, collapsed activation patterns, and a single large super-outlier channel dominates the activations at every token position in layers 1–24 (of 32). Despite its apparent redundancy, we have found that this outlier is critical: pruning it causes the model's outputs to degrade into repetitive random token loops. Paradoxically, besides this outlier, layers in LLaDA contain more redundant representations, where redundancy is particularly pronounced in earlier layers. This pattern is the reverse of what is commonly observed in AR language models, where deeper layers tend to become more redundant due to undertraining, as measured by representation similarity. Our analysis further indicates that layer collapse in DLMs is not driven by undertraining. Rather, it appears to arise from overtraining: a dominant outlier becomes an indispensable carrier of information, while the remaining representations collapse into redundant structure. These observations have strong practical implications, as we verify through controlled pre-training experiments. First, DLMs are surprisingly robust to compression: performance of LLaDA under 3-bit GPTQ quantization drops by only 1.8% on GSM8K, whereas Llama-3.1-8B under the same settings drops by 64.7%. Second, optimal non-uniform sparsity allocation reverses between the two model families: under an average budget of 50% sparsity, allocating more sparsity to early layers in LLaDA yields +8.4% over the reverse strategy, while for Llama the same early-layer-sparse allocation incurs -8.4%. Our findings reveal that the DLM training objective fundamentally reshapes layer dynamics relative to AR models, with direct consequences for how such models should be compressed and deployed. Our code will be made available upon publication.

1 INTRODUCTION

Diffusion language models (DLMs) have recently emerged as a competitive alternative to autoregressive (AR) large language models. Open models such as LLaDA-8B (Nie et al., 2026b) and DREAM-7B (Ye et al., 2025) match the quality of AR counterparts at comparable scale on standard reasoning and language understanding benchmarks. Rather than producing one token at a time from left to right, DLMs denoise a masked sequence over multiple refinement steps, which enables parallel token generation (Wu et al., 2026; Chen et al., 2026; Ma et al., 2025) and avoids the reversal curse (Berglund et al., 2024). As DLMs become more widespread, questions on the structure of their internal representations and how they propagate through layers arise. Of particular interest is whether the findings from AR models transfer one-to-one to DLMs or if there are particular behaviours that necessitate extra care during tasks such as training, fine-tuning or compression.

In this paper, we find that DLMs show distinct and partly opposite behaviour from AR models. We find that LLaDA-8B's internal representations are qualitatively unlike those of any AR model previously studied. We observe the same in iLLaDA-8B (Nie et al., 2026a), a second DLM pre-trained from scratch. Most strikingly, a *single* activation channel (see Figure 1) remains persistently and highly activated across all tokens in layers 1–24 (of 32). This extreme outlier drives layer collapse, causing multiple layers to produce nearly identical, redundant hidden representations. The outlier channel even dominates the model to such a degree that ablating it alone causes a near-total collapse in capability. While outlier channels are prominent in AR models (see Dettmers et al. (2022); Yu et al. (2025)), they only cause extreme activations for specific tokens or token positions, and pruning a single one of them causes only a noticeable accuracy drop (-4% in our experiment, see Table 1), but

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

not complete model collapse. The layer-wise similarity structure is also inverted (see Figure 3). In standard AR models, early layers usually learn more distinct transformations, while deeper layers become increasingly redundant as a consequence of undertraining (Sun et al., 2026). By contrast, LLaDA-8B exhibits highly redundant early layers. We find that this redundancy is driven by the dominating super-outlier, which suppresses the representation expressivity of other channels across the first few layers.

At first glance, this observation conflicts with the curse-of-depth hypothesis (Sun et al., 2026), which attributes high representation similarity in Pre-LN Transformers to the undertraining of deeper layers. However, our weight spectral density analysis shows that DLMs exhibit a compatible layerwise trainability pattern, but with a different source of redundancy. In LLaDA-8B, the unusually high similarity among early layers is not caused by undertraining; instead, it arises from the relative overtraining of these layers, resulting in dominant super-outliers concentrated in the earliest layers.

Why activation structure matters. A long line of work has shown that algorithms which explicitly account for activation structure (Lin et al., 2024; Lee et al., 2024; Ashkboos et al., 2024) substantially outperform naive counterparts on quantization and pruning. The structure itself has been mapped in detail for AR models: *outlier features*, extreme activations found in essentially all sufficiently large language models (Dettmers et al., 2022; An et al., 2025), are tied to core information-processing functions. *Super-weights* suppress stop-tokens (Yu et al., 2025), and attention outliers at sequence boundaries serve as sinks for excess attention mass (Gu et al., 2025; Xiao et al., 2024). Early DLM work hints that the picture is different: Rulli et al. (2025) report that attention sinks are present in DLMs but far less sensitive to removal, an observation that Myrzakhan et al. (2026) have already exploited for more efficient pruning. Our results push this line considerably further, and identify the diffusion training objective itself as the source of the divergence.

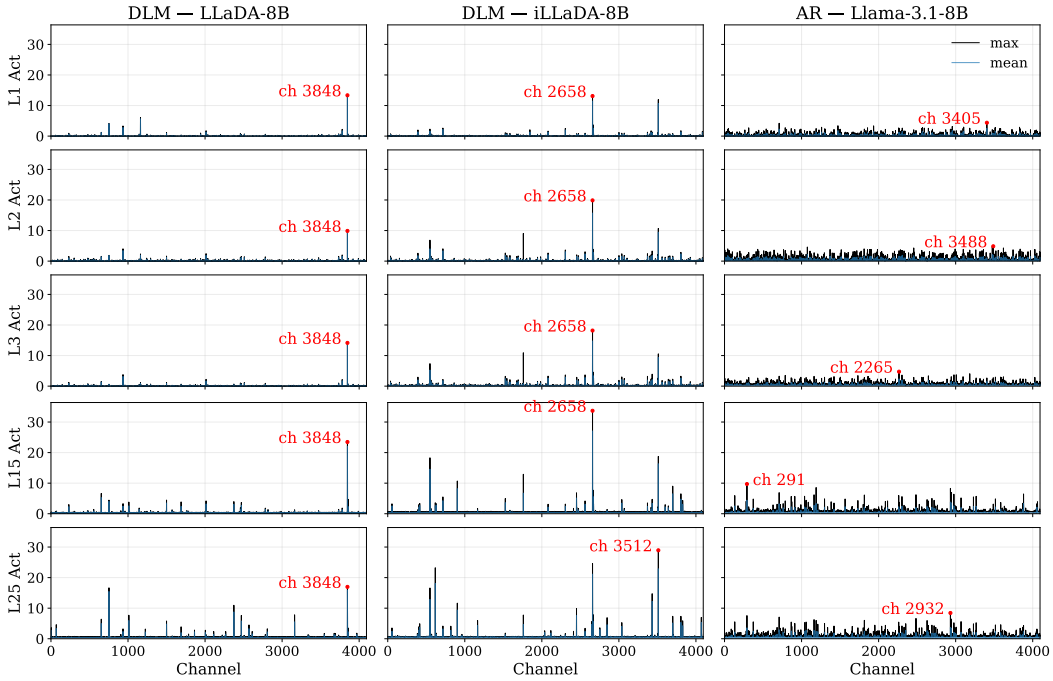


Figure 1: LLaDA activations contain a single consistent, very large super-outlier channel which persists in layers 1–24 (of 32). iLLaDA-8B (middle column) shows the same behaviour with channel 2658. The marked dot shows the channel with the largest mean absolute activation (averaged over calibration samples) at any sequence position. For Llama, the dominant channel changes on almost every layer.

Contributions. We close part of the knowledge gap between AR and diffusion language models through a systematic study of activation histograms and layer similarities in LLaDA-8B. To separate effects arising from the diffusion objective from architectural quirks, we compare LLaDA-8B to Llama-3.1-8B, which share a near-identical architecture, and complement this with the first controlled pre-training comparison in which an AR and a DL model are trained under identical conditions.

- We identify a *super-outlier* in LLaDA-8B: a single dominant activation channel qualitatively distinct from the weight and activation outliers previously reported in AR LLMs, and whose removal collapses the model.
- We characterise the layer-similarity structure of LLaDA-8B and show that it differs markedly from that of comparable AR models, with early layers being highly redundant rather than distinct, corroborated by weight spectral analysis indicative of significant overfitting.
- We translate these observations into practical guidance for DLM compression and show that AR-derived sparsity heuristics are not merely suboptimal but actively inverted relative to what LLaDA-8B calls for.
- To isolate the role of the training objective, we pre-train an AR and a DL model under an identical setup and show that the diffusion objective is responsible for the observed layer and compression behaviours. To our knowledge, this is the first such controlled comparison.

2 Layer Dynamics in Diffusion Language Models

2.1 Metrics

Layer Similarity. Our main method of analysis is the per-token cosine similarity between the hidden activations of different layers of a model. Let $h_i(x,t) \in \mathbb{R}^d$ denote the hidden state at layer i for input sequence $x \sim p_x$ at token position t. We define the layer-wise similarity as

$$\mathrm{sim}(i,j) \;=\; \mathbb{E}_{x\sim p_x}\,\mathbb{E}_t\left[\frac{\langle h_i(x,t),\, h_j(x,t)\rangle}{\|h_i(x,t)\|_2\,\|h_j(x,t)\|_2}\right], \tag{1}$$

where the inner expectation is taken over all (non-padding) token positions t in the sequence. A value of $\mathrm{sim}(i,j)$ close to 1 indicates that layers i and j produce nearly collinear representations, which we use as a proxy for layer redundancy: if the representation barely changes between two layers, the intermediate computation contributes little and is a natural candidate for pruning.

Estimating Activations. In practice, we estimate all hidden layer activations using a fixed calibration set of 128 sequences of length 2048 drawn from the C4 corpus. For DLMs we duplicate each sample 4 times and mask a $t \sim \mathrm{Unif}[0,1]$ fraction of tokens to imitate the activation statistics of a partly decoded sample sequence. We note however that this has very little effect on the resulting figures (see subsection B.1, repeating similarity and activation plots without masking), indicating that our findings are not sensitive to the exact decoding process that the DLM implements.

Heavy-tailedness of the Weight Spectrum as a Layer Trainability Measure. Representation similarity is a useful diagnostic for identifying redundant layers, but it is not by itself a reliable measure of layer trainability. High similarity can arise from qualitatively different mechanisms: in AR Transformers, similar representations in deeper layers are often associated with under-trained, near-identity residual mappings (Sun et al., 2026), whereas in DLMs, highly similar layers may instead reflect strong training of a few dominant directions, manifested as distinguished spectral or activation outliers. Thus, similarity-based importance metrics can be misleading, since they conflate under-utilised layers with layers whose representations are highly structured but dominated by a small number of directions.

Motivated by Heavy-Tailed Self-Regularization (HT-SR) theory (Martin & Mahoney, 2019; Lu et al., 2024; He et al., 2026), we argue that the heavy-tailedness of the weight spectrum provides a more reliable, training-data-free measure of layer quality than representation similarity. HT-SR theory suggests that the empirical spectral density of neural-network weight matrices encodes information about the degree of learned structure: heavier-tailed spectra are typically associated with stronger feature learning and better-trained layers. In our setting, this spectral perspective is particularly useful because it helps distinguish two cases that representation similarity alone cannot separate: near-identity redundancy caused by insufficient training, and apparent redundancy caused by strong low-dimensional structure or outlier-dominated spectra.

To quantify the heavy-tailedness of a layer, we use the Hill estimator α_{Hill} (Lu et al., 2024). For a weight W, let $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_N$ denote the eigenvalues of $W^\top W$ sorted in decreasing order. The Hill estimator of the power-law tail exponent, computed from the top k eigenvalues, is

$$\alpha_{\text{Hill}}(k) \;=\; 1 + \frac{k}{\sum_{i=1}^{k} \log \frac{\lambda_i}{\lambda_{k+1}}}, \tag{2}$$

where smaller values of α_{Hill} indicate heavier-tailed spectra. We set k using the fix-finger method (Yang et al., 2023). Under HT-SR theory, a heavier tail is typically interpreted as evidence of stronger feature learning.

2.2 Super-Outlier in DLMs

Activation outliers are a well-observed phenomenon in LLMs, but those spikes usually only occur briefly for specific tokens or for specific token positions (An et al., 2025). In the case of LLaDA-8B, we present a unique finding: when analysing the activation heatmaps over sequence positions in LLaDA, we find that there exists one single dominant outlier channel, which has a persistently high activation (up to 5 times as large as the next largest outlier) over all sequence positions (see Figure 2).

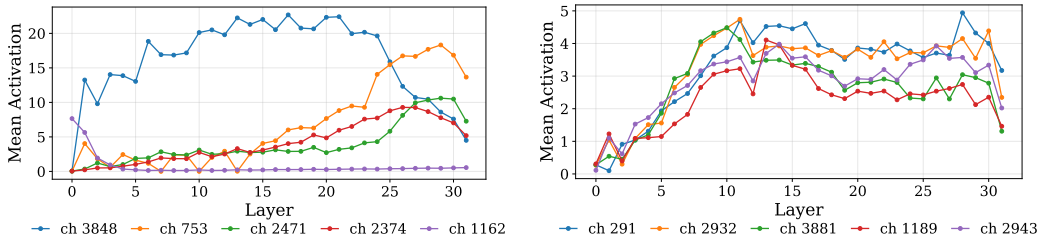
In contrast to outliers in AR models, which are reported to have well-behaved mechanisms such as stop-word suppression (Yu et al., 2025), the super-outlier in LLaDA seems more akin to a constant bias that is learned by the network. The qualitative effect of pruning the super-outlier is striking: LLaDA-8B degenerates into repetitive token loops, regardless of the prompt. As an illustrative example (taken from GSM8K):

> **Question.** Kylar went to the store to buy glasses for his new apartment. One glass costs $5, but every second glass costs only 60% of the price. Kylar wants to buy 16 glasses. How much does he need to pay for them?
>
> **LLaDA-8B answer (super-outlier pruned).** *buy buy buy buy1yl buy buy buy*

This is in stark contrast to Llama-3.1-8B, where removing the highest-magnitude channel only slightly reduces the accuracy, see Table 1.

Table 1: Effect of pruning the single highest-magnitude channel on a 200-question subset of GSM8K. Removing channel 3848 in LLaDA-8B (the super-outlier) collapses the model entirely, while pruning the corresponding top-magnitude channel in Llama-3.1-8B causes only a minor accuracy drop. Pruning channel 2658 in iLLaDA-8B likewise collapses the model.

Model	Channel pruned	Baseline	Accuracy	Δ baseline
Llama-3.1-8B	291	83%	79%	-4%
LLaDA-8B	3848	83%	0%	-83%
iLLaDA-8B	2658	78%	0.5%	-77.5%



(a) LLaDA-8B (b) Llama-3.1-8B

Figure 2: Magnitudes of the top-5 QKV input channels across layers. In LLaDA-8B (a), the super-outlier dominates by a wide margin in layers 1–24 (of 32); in Llama-3.1-8B (b), the top-5 channels have comparable magnitude and the dominant one changes.

2.3 Early-Layer Redundancies in DLMs

Next, we analyse the pairwise similarities of the hidden representations in LLaDA, iLLaDA-8B and Llama in Figure 3. Surprisingly, we find that almost all layers in LLaDA (top left) exhibit an extremely high similarity to each other, especially early layers which are almost identical to each other even over ranges of 15+ layers. This is partially mitigated if the similarities are calculated with the channel corresponding to the super-outlier set to zero (Ch. 3848, see subsection 2.2), but the learned representations for LLaDA are still more redundant than the ones from Llama-3.1-8B (which will result in a stronger robustness of DLMs compared to ARs, see section 3). On the other hand, for Llama-3.1-8B (top row, third column), the early layers show very distinct activation patterns, and the later layers contain more redundancy, in line with previous work on AR models. iLLaDA-8B

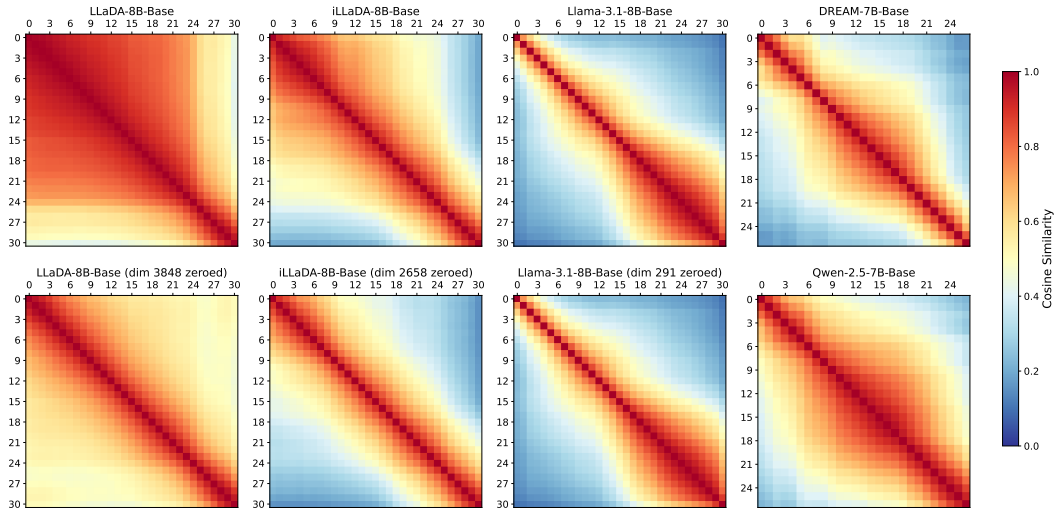


Figure 3: Per-token cosine similarity for different models. The top row shows LLaDA-8B, iLLaDA-8B, Llama-3.1-8B and DREAM-7B, where LLaDA-8B exhibits highly redundant (collapsed) layer similarities. In the bottom row, we have removed the largest outlier from the similarity calculations of the first three models (channels 3848, 2658 and 291), showing that much of the layer similarity of LLaDA and iLLaDA-8B is due to a single outlier, unlike Llama, where removing one outlier barely changes the resulting similarities. The right column compares DREAM-7B (top) with Qwen-2.5-7B (bottom). As DREAM was fine-tuned from Qwen-initialised weights, their layer similarity patterns are very similar.

shows the same effect in a less pronounced form: its early layers are highly similar to each other, and zeroing its super-outlier (channel 2658) removes much of this similarity.

This behaviour from LLaDA-8B is not shared with DREAM-7B, since DREAM has been initialised from the weights of Qwen-2.5-7B (an AR model) and only fine-tuned on masked sequences. Therefore, the final weights of DREAM do not differ much from the ones of Qwen, and their similarity patterns look almost identical. This outlier profile can be harmful in handling DLMs. Common wisdom is that strong outliers make for layers or networks that are hard to compress; thus the compression community has devised various ways of dealing with them (Lin et al., 2024; Dettmers et al., 2022; Ashkboos et al., 2024). Based on this, metrics such as OWL (Yin et al., 2025) allocate a lower sparsity level to layers with strong outliers (which are presumably difficult to compress). In LLaDA-8B, this would mean that early layers should be pruned less aggressively. Due to the redundancy that is exhibited in the early layers, we have a different finding: early layers can actually be pruned more strongly; we quantify this in section 3.

2.4 Heavy-Tailed Evidence of Overtraining

Leveraging the analysis performed by Martin & Mahoney (2019); Lu et al. (2024), we can also measure how well trained the layers in a neural network are via the α_{Hill} estimator.

According to the works of Zhou et al. (2023); Liu et al. (2024); He et al. (2025), a balanced distribution of α_{Hill} indicates that the layers in a network are better trained. A relatively low value of α_{Hill} (with respect to the other layers) indicates a relatively "overtrained" layer, while a very high value of α_{Hill} indicates an "undertrained" layer.

This behaviour can be seen in Figure 4: early layers in LLaDA-8B have a strikingly low α_{Hill} (relative) value, while the distribution of the twin-model Llama-3.1-8B is much flatter. This ties in with the early-layer occurrence of the super-outlier, a form of overtraining where the layer representations collapse to a single dominant channel.

3 DLMs Are More Robust to Compression Despite Super-Outliers

In this section, we demonstrate that our observations have practically relevant consequences when dealing with DLMs. For this, we have chosen the task of model compression. We prune and quantize LLaDA and Llama-3.1-8B using GPTQ and WANDA and measure their performance on math and

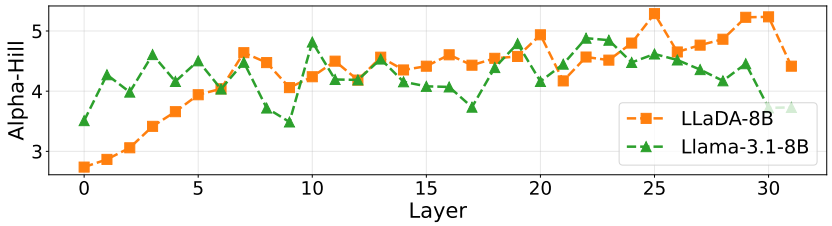


Figure 4: The α_{Hill} value is very low in the early layers of LLaDA-8B which shows that these layers are actually overtrained. α_{Hill} is averaged over modules in each layer.

question-answering under different overall and layer-wise compression strength allocations. We show that (a) AR-optimal schemes for layer-wise compression strength are inverted when dealing with DLMs and (b) DLMs are more robust to compression. Observation (a) is a direct consequence of our layer-wise similarity analysis, but observation (b) is particularly surprising, as LLaDA-8B contains a super-outlier that is very sensitive to change. However, we have seen in Figure 1 that LLaDA contains fewer outliers apart from the super-outlier, and we have seen in Figure 3 that LLaDA layer activations are more redundant, even when the outlier is removed. This translates to an overall stronger compression robustness, particularly as modern compression methods (like WANDA) are designed to be robust to a certain amount of outliers.

3.1 Experiment Setup

Evaluation. Models are evaluated both on an average of 6 common question-answering (QnA) tasks (ARC-C, HellaSwag, PIQA, WinoGrande, BoolQ, OBQA) as well as on reasoning via GSM8K. DLMs are evaluated using FastDLLM (Wu et al., 2026) with a generation length of 1024 and one token decoded per diffusion step. Further details can be found in subsection A.1.

Compression. Pruning experiments were performed using WANDA (Sun et al., 2024b) and quantization experiments using GPTQ (Frantar et al., 2023). Both methods used calibration samples drawn from the C4 dataset (128 for WANDA, 256 for GPTQ). Adding samples with masked tokens did not change the performance of the compression at all, so we opted not to mask samples for the DLM compression. The described sparsity schedules *earlier-is-sparser* (EIS) and *deeper-is-sparser* (DIS) follow a simple linear allocation, using an ϵ of 0.08. This means for EIS applied to a T layer network at sparsity s, we assign sparsity $s + 0.08\left(1 - \frac{2(t-1)}{T-1}\right)$ to layer t. Correspondingly, for DIS, the addition turns into a subtraction.

3.2 Compression

Pruning. Our experiments in Figure 5 (top row) show the average accuracy on QnA tasks (first two columns) and GSM8K reasoning (last two columns) of LLaDA-8B and Llama-3.1-8B at three sparsity levels (0.3, 0.5, 0.7) and under three sparsity allocation strategies that result in the same total sparsity (uniform, earlier-is-sparser, deeper-is-sparser).

These results reinforce the practicality of our observations made in section 2. We have two key observations to make in the results: **First**, LLaDA starts out with a generally lower accuracy than Llama, but is much more robust when compressed, achieving almost double the accuracy of Llama for GSM8K at 50% sparsity. This ties into our observation that LLaDA-8B contains many more similarities, introducing redundancy into the network and making the accuracy of each single layer less important (as long as the super-outlier is protected, which is exactly the case under WANDA pruning). **Second,** the optimal sparsity allocation differs between LLaDA and Llama. For Llama, pruning early layers more strongly (EIS) is always suboptimal and either uniform sparsity allocation or pruning deeper layers more strongly (DIS) are the optimal choice (depending on task). For LLaDA, this flips, and DIS is usually the suboptimal choice, with either EIS or uniform performing the best. Only for sparsity 0.7 on QnA, DIS significantly outperforms EIS for LLaDA, which we suspect might be due to the fact that the very high sparsity level starts to necessitate pruning of the weights connected to the super-outlier.

Quantization. As quantization is usually the more practical method of compressing a neural network, we have included experiments on quantization as well in Figure 5 (bottom row). However, as quantization has to be done to discrete bit widths, we have not included an experiment on a layerwise

bit-width allocation, which cannot be made with such a fine-grained allocation as is done in the experiments on pruning.

Other than that, these experiments reinforce the same observation we have made for pruning above, namely that LLaDA is inherently more robust to compression than Llama-3.1-8B, starting out with a worse base performance and surpassing the performance of Llama at 3 bits for QnA (first two columns, bottom row), although Llama is ahead again at 4 bits, and at both 3 and 4 bits for GSM8K (last two columns, bottom row).

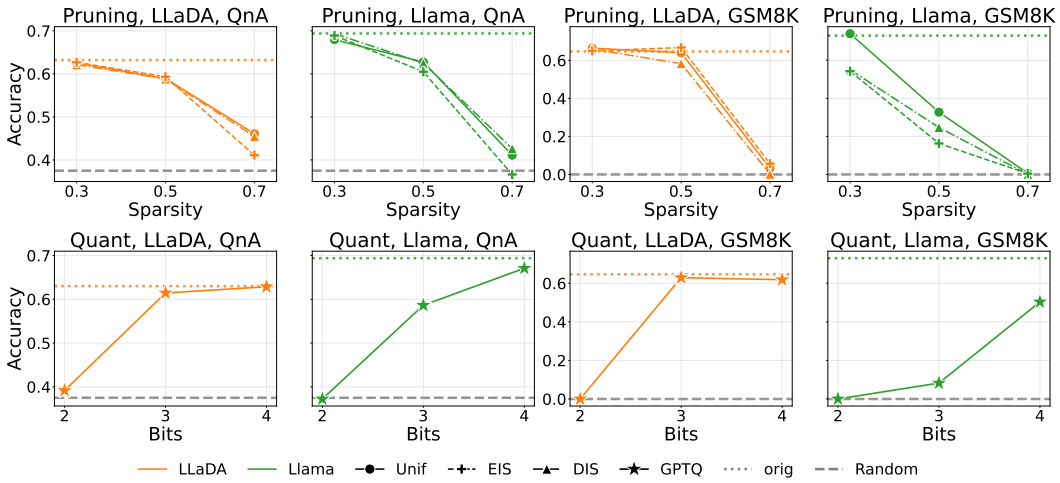


Figure 5: Compression performance for LLaDA-8B and Llama-3.1-8B: pruning across sparsity allocation strategies (top row, at $\epsilon = 0.08$) and GPTQ quantization across bit-widths (bottom row), evaluated on the base-model commonsense QnA tasks (first two columns) and the instruct-model GSM8K task (last two columns).

4 Supporting Evidence: A Controlled 160M Pair

The experiments in this section reinforce that our findings are not just a spurious quirk of LLaDA-8B, but are inherent to the pre-training objective that is used in DLMs. To establish this, we perform pre-training experiments with a Pythia-160M (Vaswani et al., 2017; Biderman et al., 2023) architecture pre-trained on 100B tokens from FineWebEdu (Penedo et al., 2024). The resulting models AR-160M, DLM-160M are trained identically, solely varying in the training objective (cross-entropy either with or without masking). Further details are in subsection A.2. We perform the same evaluation as we did for section 3, minus GSM8K due to the small model scale.

4.1 Results

Our analysis of the small-scale models supports the following two observations from the base model: (a) early layer representations in DLM-160M are more similar to each other than in AR-160M, and more similar to each other than the deeper layer representations, and (b) DLM-160M shows more robustness to compression than AR-160M. Due to our controlled setting, we could thus ensure that these observations are due to the DLM training objective and not due to architectural or pre-training hyperparameter choices. We note that DLM-160M is lacking the super-outlier: to observe weight and activation outliers, one has to train models to much larger scale (>6.7B indicated by Dettmers et al. (2022)), which was out of the scope of this work.

Layer Similarities. To more easily quantify how similar the layer representations in our two models are, we have averaged the per-layer similarities in a blockwise fashion: we have grouped the first 4 layers of the model together, as well as the last 4 layers and have averaged over the resulting 16 inner-block layer-wise similarities. Table 2 shows that the same early-layer redundancy pattern of the large models starts to emerge in our pre-trained models: the early layer similarity is higher in DLM-160M compared to AR-160M, while it flips for the late-layer similarities.

Heavy-tailed Signature. The same heavy-tailed signature appears in the 160M pair (Figure 6): the controlled DLM shows a markedly lower α_{Hill} in early layers, while its AR twin does not. This behaviour normalises in later layers, where both models show similar α_{Hill} values.

Table 2: Average cosine similarities between early layers are higher for DLM-160M. To better assess the similarity, we have grouped the first 4 layers and the last 4 layers together and have averaged over their pairwise cosine similarities.

Model	Early-layer similarity	Deeper-layer similarity
DLM-160M	0.877	0.787
AR-160M	0.818	0.829

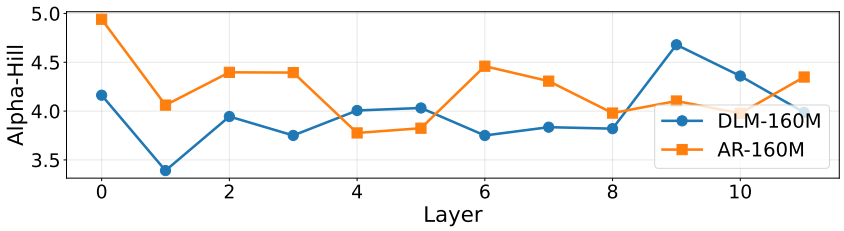


Figure 6: Similar to the larger models, α_{Hill} is smaller for DLM-160M compared to AR-160M.

Compression Results. In Figure 7, we present results for pruning (left) and quantization (right) of our 160M models. We have performed pruning and evaluation for all pre-training learning rates and selected the best performing model per sparsity level. Similar to the behaviour of LLaDA-8B, DLM-160M starts out with less performance than AR-160M, but is more robust to compression. Here, the DLM model only surpasses its AR counterpart under 2-bit quantization, where AR-160M drops to chance level; at 3 and 4 bits AR-160M stays ahead, while for pruning, they roughly match at the highest sparsity level (which could be due to initially subpar performance of DLM-160M). Also replicating the behaviour of LLaDA versus Llama, the best sparsity allocation strategy for DLM-160M is consistently earlier-is-sparser, while AR-160M achieves higher accuracy using deeper-is-sparser (aside from low sparsity regimes up to 40%, where all strategies perform similarly). This matches the expectations from our similarity analysis in Table 2.

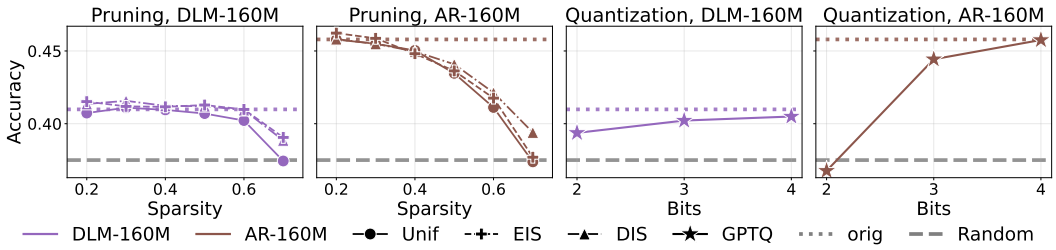


Figure 7: Robustness replication on the controlled 160M pair: pruning (left) and quantization (right). DLM-160M loses less performance than AR-160M under pruning, with the best sparsity allocation strategy inverted (EIS for the DLM, DIS for the AR). Under quantization, DLM-160M starts below AR-160M at full precision but surpasses it at 2 bits. Both results are in line with the behaviour of LLaDA-8B versus Llama-3.1-8B.

5 Related Work

Outliers, Sinks, and Super-weights. A growing body of work shows that a small number of coordinates disproportionately shape large language model behaviour. Attention sinks concentrate probability mass on a handful of tokens (Xiao et al., 2024; Gu et al., 2025); massive activations identify a few token positions whose hidden states grow orders of magnitude larger than the rest (Sun et al., 2024a); super weights (Yu et al., 2025) and systematic outliers (An et al., 2025) extend the picture to individual parameters and feature dimensions whose removal disproportionately damages the model. These phenomena are predominantly described along the *token* axis: a small set of positions dominates attention or activation norm. Collapsed layers describe the *channel-axis* counterpart: a single hidden dimension that dominates the representation across all positions. In contrast to attention sinks, this phenomenon is absent in autoregressive models of comparable scale.

Curse of Depth. Multiple works analyse the tendency of AR models to have less well-trained deeper layers (Sun et al., 2026; Gromov et al., 2025; Siji et al., 2026), a phenomenon also termed the *Curse of Depth*. Existing approaches to mitigate this issue include depth-growing during training (Kapl

et al., 2025), adding sparsity to the model (Muhtar et al., 2026) or mixing Pre- and Post-LN (Li et al., 2024). While our work reveals that early layers in DLMs are more redundant and can be compressed more strongly than in ARs, this does not mean that the curse is broken: if an improved training paradigm made early layers more informative and distinct, DLMs would still exhibit strong redundancies in later layers and could thus face similar issues.

Weight and Activation Dynamics in DLMs. A concurrent work by Goel et al. (2026) also uses the cosine similarity of hidden states to characterise layer dynamics, but analyses similarities between immediately neighbouring layers instead of long-range interactions, also revealing early-layer redundancies. The authors use this to propose a layer-skip method for inference, which is different from our sparsity allocation. In agreement with our work, the authors find that AR-finetuned DLMs retain most of the layer dynamics from initialisation. Apart from that, our work uses cosine similarities in a markedly different way, which allows us to detect the super-outlier in LLaDA-8B. Additionally, we provide controlled AR-DLM twins and perform extensive compression experiments. Rulli et al. (2025) characterise attention sinks in DLMs, which are token-level attention outliers. Our work is complementary, as we analyse channel-wide *activation* outliers.

Compression of Diffusion Language Models. Several recent works compress masked diffusion language models. On the quantization side, DLM-Quant (Xu & Yang, 2025), Quant-dLLM (Zhang et al., 2026), and Lin et al. (2025) adapt post-training quantization pipelines to the bidirectional, multi-step inference of DLMs. Myrzakhan et al. (2026) use insights on attention sinks in DLMs to improve pruning. These works treat compression as a methodological problem, where they adapt a known technique to a new architecture. Our work is decidedly not a compression paper, but an analysis of activation dynamics whose findings carry practical implications that may be used in further compression research.

6 Discussion

We have shown that DLMs exhibit activation dynamics that are qualitatively distinct from their AR counterparts. LLaDA-8B contains a single dominant activation channel that persists across all token positions in layers 1–24 (of 32); removing it causes total model collapse, a fragility with no parallel in AR models of comparable scale. This super-outlier drives early-layer redundancy through a mechanism we attribute to overtraining rather than undertraining, as evidenced by heavy-tailed weight spectra in early layers. A controlled 160M pre-training comparison supports that both the redundancy pattern and the compression robustness are products of the diffusion training objective and not architectural or pre-training artefacts. These findings carry direct practical consequences: Allocating more sparsity to deeper layers, exactly what is useful for AR models, instead performs suboptimally on DLMs.

We envision that our findings inspire further research into DLM activation dynamics. The super-outlier in particular leaves many open questions, for example, which mechanisms of the DLM pre-training process produce it, and whether it is a harmful artefact or plays a useful role in information processing of DLMs. Additionally, the redundancy of DLM layer representations might indicate that the pre-training process is not yet optimised, similar to how redundant later layers in pre-LN AR models indicate suboptimal gradient flow. Ensuring that all layers are trained to produce distinct and informative representations would produce models that are more capable at similar sizes.

Limitations. Our findings are mostly focussed on a single DLM-AR pair, namely LLaDA-8B and Llama-3.1-8B, with key analyses repeated for iLLaDA-8B. While the behaviour of Llama-3.1-8B is consistent with what is reported in the AR model literature, our findings for LLaDA-8B and iLLaDA-8B are novel. To our knowledge, LLaDA-8B and iLLaDA-8B are the only large-scale DLMs that are completely pre-trained from scratch, as the current literature is mostly focussed on creating DLMs by fine-tuning them from AR model weights (Bie et al., 2025; 2026; Yu et al., 2026). During fine-tuning, these models' weights stay close to their original values, therefore models such as DREAM-7B show activation dynamics that are very close to their AR counterpart, see Figure 3. While supporting our other findings, the 160M pre-training models did not show a super-outlier like LLaDA-8B, which was expected given the small scale of the model. To make our results more widely applicable, a large-scale DLM pre-training study could be implemented, pre-training DLMs on multiple parameter scales (e.g. 350M, 1B, 3B, 8B, 16B). This task unfortunately requires large amounts of GPU resources to run.

AI USE STATEMENT

In this work, we used generative AI tools to implement standard methods such as alpha-hill calculation. Results were carefully checked against reference implementations to ensure correctness. Additionally, we used generative AI for code editing, plot creation and providing feedback on the paper draft. We have not used generative AI in other tasks with required disclosure. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

Our complete codebase will be made available as soon as the double-blind review phase ends. We have included a detailed experiment setup in the appendix (Appendix A) as well as the methodology section of the paper (subsection 2.1, subsection 3.1, section 4).

REFERENCES

Niccolò Ajroldi. plainLM: Language model pretraining in PyTorch, 2024.

Yongqi An, Xu Zhao, Tao Yu, Ming Tang, and Jinqiao Wang. Systematic outliers in large language models. In *The Thirteenth International Conference on Learning Representations*, 2025.

Saleh Ashkboos, Amirkeivan Mohtashami, Maximilian L. Croci, Bo Li, Pashmina Cameron, Martin Jaggi, Dan Alistarh, Torsten Hoefler, and James Hensman. QuaRot: Outlier-free 4-bit inference in rotated llms. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 100213–100240. Curran Associates, Inc., 2024. doi: 10.52202/079017-3180.

Lukas Berglund, Meg Tong, Maximilian Kaufmann, Mikita Balesni, Asa Cooper Stickland, Tomasz Korbak, and Owain Evans. The Reversal Curse: LLMs trained on "A is B" fail to learn "B is A". In *The Twelfth International Conference on Learning Representations*, 2024.

Shane Bergsma, Nolan Simran Dey, Gurpreet Gosal, Gavia Gray, Daria Soboleva, and Joel Hestness. Straight to zero: Why linearly decaying the learning rate to zero works best for LLMs. In *The Thirteenth International Conference on Learning Representations*, 2025.

Stella Biderman, Hailey Schoelkopf, Quentin Gregory Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, Usvsn Sai Prashanth, Edward Raff, Aviya Skowron, Lintang Sutawika, and Oskar Van Der Wal. Pythia: A suite for analyzing large language models across training and scaling. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 2397–2430. PMLR, July 2023.

Tiwei Bie, Maosong Cao, Kun Chen, Lun Du, Mingliang Gong, Zhuochen Gong, Yanmei Gu, Jiaqi Hu, Zenan Huang, Zhenzhong Lan, Chengxi Li, Chongxuan Li, Jianguo Li, Zehuan Li, Huabin Liu, Ling Liu, Guoshan Lu, Xiaocheng Lu, Yuxin Ma, Jianfeng Tan, Lanning Wei, Ji-Rong Wen, Yipeng Xing, Xiaolu Zhang, Junbo Zhao, Da Zheng, Jun Zhou, Junlin Zhou, Zhanchao Zhou, Liwang Zhu, and Yihong Zhuang. LLaDA2.0: Scaling up diffusion language models to 100B, 2025.

Tiwei Bie, Maosong Cao, Xiang Cao, Bingsen Chen, Fuyuan Chen, Kun Chen, Lun Du, Daozhuo Feng, Haibo Feng, Mingliang Gong, Zhuocheng Gong, Yanmei Gu, Jian Guan, Kaiyuan Guan, Hongliang He, Zenan Huang, Juyong Jiang, Zhonghui Jiang, Zhenzhong Lan, Chengxi Li, Jianguo Li, Zehuan Li, Huabin Liu, Lin Liu, Guoshan Lu, Yuan Lu, Yuxin Ma, Xingyu Mou, Zhenxuan Pan, Kaida Qiu, Yuji Ren, Jianfeng Tan, Yiding Tian, Zian Wang, Lanning Wei, Tao Wu, Yipeng Xing, Wentao Ye, Liangyu Zha, Tianze Zhang, Xiaolu Zhang, Junbo Zhao, Da Zheng, Hao Zhong, Wanli Zhong, Jun Zhou, Junlin Zhou, Liwang Zhu, Muzhi Zhu, and Yihong Zhuang. LLaDA2.1: Speeding up text diffusion via token editing, 2026.

Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. PIQA: Reasoning about physical commonsense in natural language, 2019.

Zigeng Chen, Gongfan Fang, Xinyin Ma, Ruonan Yu, and Xinchao Wang. dParallel: Learnable parallel decoding for dLLMs. In *The Fourteenth International Conference on Learning Representations*, 2026.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. BoolQ: Exploring the surprising difficulty of natural yes/no questions. In Jill Burstein, Christy Doran, and Thamar Solorio (eds.), *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 2924–2936, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1300.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? Try ARC, the AI2 reasoning challenge, 2018.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Tim Dettmers, Mike Lewis, Younes Belkada, and Luke Zettlemoyer. Gpt3.int8(): 8-bit matrix multiplication for transformers at scale. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh (eds.), *Advances in Neural Information Processing Systems*, volume 35, pp. 30318–30332. Curran Associates, Inc., 2022.

Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. GPTQ: Accurate post-training quantization for generative pre-trained transformers, 2023.

Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. The language model evaluation harness, July 2024.

Raghavv Goel, Risheek Garrepalli, Sudhanshu Agrawal, Chris Lott, Mingu Lee, and Fatih Porikli. A comparative analysis of layer-wise representational capacity in AR and diffusion llms, 2026.

Andrey Gromov, Kushal Tirumala, Hassan Shapourian, Paolo Glorioso, and Dan Roberts. The unreasonable ineffectiveness of the deeper layers. In *The Thirteenth International Conference on Learning Representations*, 2025.

Xiangming Gu, Tianyu Pang, Chao Du, Qian Liu, Fengzhuo Zhang, Cunxiao Du, Ye Wang, and Min Lin. When attention sink emerges in language models: An empirical view. In *The Thirteenth International Conference on Learning Representations*, 2025.

Di He, Songjun Tu, Ajay Jaiswal, Li Shen, Ganzhao Yuan, Shiwei Liu, and Lu Yin. AlphaDecay: Module-wise weight decay for heavy-tailed balancing in llms, 2025.

Di He, Songjun Tu, Keyu Wang, Lu Yin, and Shiwei Liu. One LR doesn't fit all: Heavy-tail guided layerwise learning rates for LLMs. In *Forty-Third International Conference on Machine Learning*, 2026.

Ferdinand Kapl, Emmanouil Angelis, Tobias Hoppe, Kaitlin Maile, Johannes von Oswald, Nino Scherrer, and Stefan Bauer. Do depth-grown models overcome the curse of depth? An in-depth analysis. *ArXiv*, abs/2512.08819, 2025.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with PagedAttention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*, 2023.

Changhun Lee, Jungyu Jin, Taesu Kim, Hyungjun Kim, and Eunhyeok Park. OWQ: Outlier-aware weight quantization for efficient fine-tuning and inference of large language models, 2024.

Pengxiang Li, Lu Yin, and Shiwei Liu. Mix-LN: Unleashing the power of deeper layers by combining pre-LN and post-LN. *ArXiv*, abs/2412.13795, 2024.

Haokun Lin, Haobo Xu, Yichen Wu, Ziyu Guo, Renrui Zhang, Zhichao Lu, Ying Wei, Qingfu Zhang, and Zhenan Sun. Quantization Meets dLLMs: A Systematic Study of Post-training Quantization for Diffusion LLMs, October 2025.

Ji Lin, Jiaming Tang, Haotian Tang, Shang Yang, Wei-Ming Chen, Wei-Chen Wang, Guangxuan Xiao, Xingyu Dang, Chuang Gan, and Song Han. AWQ: Activation-aware weight quantization for LLM compression and acceleration. In *MLSys*, 2024.

Zihang Liu, Yuanzhe Hu, Tianyu Pang, Yefan Zhou, Pu Ren, and Yaoqing Yang. Model balancing helps low-data training and fine-tuning. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 1311–1331, Miami, Florida, USA, November 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.78.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

Aaron Lou, Chenlin Meng, and Stefano Ermon. Discrete diffusion language modeling by estimating the ratios of the data distribution, 2024.

Haiquan Lu, Yefan Zhou, Shiwei Liu, Zhangyang Wang, Michael W. Mahoney, and Yaoqing Yang. AlphaPruning: Using heavy-tailed self regularization theory for improved layer-wise pruning of large language models. In *The Thirty-Eighth Annual Conference on Neural Information Processing Systems*, 2024.

Yuxin Ma, Lun Du, Lanning Wei, Kun Chen, Qian Xu, Kangyu Wang, Guofeng Feng, Guoshan Lu, Lin Liu, Xiaojing Qi, Xinyuan Zhang, Zhen Tao, Haibo Feng, Ziyun Jiang, Ying Xu, Zenan Huang, Yihong Zhuang, Haokai Xu, Jiaqi Hu, Zhenzhong Lan, Junbo Zhao, Jianguo Li, and Da Zheng. dInfer: An Efficient Inference Framework for Diffusion Language Models, October 2025.

Charles H. Martin and Michael W. Mahoney. Traditional and Heavy-Tailed Self Regularization in Neural Network Models, January 2019.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. Can a suit of armor conduct electricity? A new dataset for open book question answering. In *EMNLP*, 2018.

Dilxat Muhtar, Xinyuan Song, Sebastian Pokutta, Max Zimmer, Nico Pelleriti, Thomas Hofmann, and Shiwei Liu. When does sparsity mitigate the curse of depth in llms, 2026.

Aidar Myrzakhan, Tianyi Li, Bowei Guo, Shengkun Tang, and Zhiqiang Shen. Sink-aware pruning for diffusion language models, 2026.

Shen Nie, Qiyang Min, Shaoxuan Xu, Zihao Huang, Yuxuan Song, Yong Shan, Yankai Lin, Wayne Xin Zhao, Chongxuan Li, and Ji-Rong Wen. Improved large language diffusion models, 2026a.

Shen Nie, Fengqi Zhu, Zebin You, Xiaolu Zhang, Jingyang Ou, Jun Hu, JUN ZHOU, Yankai Lin, Ji-Rong Wen, and Chongxuan Li. Large language diffusion models. In *The Thirty-Ninth Annual Conference on Neural Information Processing Systems*, 2026b.

Guilherme Penedo, Hynek Kydlíček, Loubna Ben allal, Anton Lozhkov, Margaret Mitchell, Colin Raffel, Leandro Von Werra, and Thomas Wolf. The FineWeb datasets: Decanting the web for the finest text data at scale. In *The Thirty-Eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024.

Maximo Eduardo Rulli, Simone Petruzzi, Edoardo Michielon, Fabrizio Silvestri, Simone Scardapane, and Alessio Devoto. Attention sinks in diffusion language models, 2025.

Subham Sekhar Sahoo, Marianne Arriola, Aaron Gokaslan, Edgar Mariano Marroquin, Alexander M Rush, Yair Schiff, Justin T Chiu, and Volodymyr Kuleshov. Simple and effective masked diffusion language models. In *The Thirty-Eighth Annual Conference on Neural Information Processing Systems*, 2024.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. WinoGrande: An adversarial winograd schema challenge at scale, 2019.

Aleena Siji, Amir Mohammad Karimi-Mamaghan, Ferdinand Kapl, Tobias Höppe, Emmanouil Angelis, Andrea Dittadi, Maurice Brenner, Michael Heinzinger, Karl Henrik Johansson, Kaitlin Maile, Johannes von Oswald, and Stefan Bauer. From words to amino acids: Does the curse of depth persist? *ArXiv*, abs/2602.21750, 2026.

Mingjie Sun, Xinlei Chen, J Zico Kolter, and Zhuang Liu. Massive activations in large language models. In *First Conference on Language Modeling*, 2024a.

Mingjie Sun, Zhuang Liu, Anna Bair, and J Zico Kolter. A simple and effective pruning approach for large language models. In *The Twelfth International Conference on Learning Representations*, 2024b.

Wenfang Sun, Xinyuan Song, Pengxiang Li, Lu Yin, Yefeng Zheng, and Shiwei Liu. The curse of depth in large language models. In *The Thirty-Ninth Annual Conference on Neural Information Processing Systems*, 2026.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

Kaiyue Wen, Zhiyuan Li, Jason S. Wang, David Leo Wright Hall, Percy Liang, and Tengyu Ma. Understanding warmup-stable-decay learning rates: A river valley loss landscape view. In *The Thirteenth International Conference on Learning Representations*, 2025.

Chengyue Wu, Hao Zhang, Shuchen Xue, Zhijian Liu, Shizhe Diao, Ligeng Zhu, Ping Luo, Song Han, and Enze Xie. Fast-dLLM: Training-free acceleration of diffusion LLM by enabling KV cache and parallel decoding. In *The Fourteenth International Conference on Learning Representations*, 2026.

Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient streaming language models with attention sinks. In *The Twelfth International Conference on Learning Representations*, 2024.

Chen Xu and Dawei Yang. DLLMQuant: Quantizing Diffusion-based Large Language Models, August 2025.

Yaoqing Yang, Ryan Theisen, Liam Hodgkinson, Joseph E. Gonzalez, Kannan Ramchandran, Charles H. Martin, and Michael W. Mahoney. Test accuracy vs. generalization gap: Model selection in NLP without accessing training or testing data. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, Kdd '23, pp. 3011–3021, New York, NY, USA, 2023. Association for Computing Machinery. ISBN 979-8-4007-0103-0. doi: 10.1145/3580305.3599518.

Jiacheng Ye, Zhihui Xie, Lin Zheng, Jiahui Gao, Zirui Wu, Xin Jiang, Zhenguo Li, and Lingpeng Kong. Dream 7B: Diffusion Large Language Models, August 2025.

Lu Yin, You Wu, Zhenyu Zhang, Cheng-Yu Hsieh, Yaqing Wang, Yiling Jia, Gen Li, Ajay Jaiswal, Mykola Pechenizkiy, Yi Liang, Michael Bendersky, Zhangyang Wang, and Shiwei Liu. Outlier Weighed Layerwise Sparsity (OWL): A Missing Secret Sauce for Pruning LLMs to High Sparsity, June 2025.

Mengxia Yu, De Wang, Qi Shan, Colorado J Reed, and Alvin Wan. The super weight in large language models, 2025.

Yifan Yu, Yuqing Jian, Junxiong Wang, Zhongzhu Zhou, Donglin Zhuang, Xinyu Fang, Sri Yanamandra, Xiaoxia Wu, Qingyang Wu, Shuaiwen Leon Song, Tri Dao, Ben Athiwaratkun, James Zou, Fan Lai, and Chenfeng Xu. Introspective diffusion language models, 2026.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. HellaSwag: Can a machine really finish your sentence?, 2019.

Tianao Zhang, Zhiteng Li, Xianglong Yan, Haotong Qin, Yong Guo, and Yulun Zhang. Quant-dLLM: Post-training extreme low-bit quantization for diffusion large language models. In *The Fourteenth International Conference on Learning Representations*, 2026.

Yefan Zhou, Tianyu Pang, Keqin Liu, charles h martin, Michael W. Mahoney, and Yaoqing Yang. Temperature balancing, layer-wise weight analysis, and neural network training. In *Thirty-Seventh Conference on Neural Information Processing Systems*, 2023.

A Experimental details

A.1 Evaluation of Language Models

Models are evaluated on the following question-answering tasks: ARC-Challenge (Clark et al., 2018), HellaSwag (Zellers et al., 2019), PIQA (Bisk et al., 2019), WinoGrande (Sakaguchi et al., 2019), BoolQ (Clark et al., 2019), OpenBookQA (Mihaylov et al., 2018). Additionally, we evaluate on reasoning via GSM8K (Cobbe et al., 2021). We use 25-shot for ARC-Challenge, 10-shot for HellaSwag, 5-shot for WinoGrande and GSM8K, and 0-shot for BoolQ, OpenBookQA, and PIQA. We used base models for QnA, and corresponding instruction-fine-tuned variants for GSM8K.

For the DLMs, we used an adapted version of FastDLLM (Wu et al., 2026) with single KV-cache, but not parallel decoding. The block-length was set to 32 and the generation length to 1024, with 1024 decoding steps (so 1 token per decoding step) and low confidence remasking. For the AR models, we use vLLM (Kwon et al., 2023) for inference acceleration together with LM-Eval (Gao et al., 2024).

All evaluations are done on a single H100 GPU. Evaluations take around 6 hours for DLMs and 20 minutes for AR models for both task sets.

A.2 Small-scale pre-training

Autoregressive Model. We used Ajroldi (2024) to pre-train Pythia-160M parameter transformer models (Vaswani et al., 2017; Biderman et al., 2023) on causal language modelling, with 100B tokens of FineWebEdu (Penedo et al., 2024) on 8×A100-80GB GPUs. We use sequence length 2048 and a batch size of 0.5M tokens, cross-entropy loss, Adam (Kingma & Ba, 2017) with decoupled weight decay (Loshchilov & Hutter, 2019) of 0.1, gradient clipping of 1, and $(\beta_1, \beta_2) = (0.9, 0.95)$. We use Warmup-Stable-Decay (Wen et al., 2025) to schedule the learning rates, warm up of 1900 steps (1%) and decay to 0 (Bergsma et al., 2025) of 10% of the token budget. We perform three independent runs for learning rates $\{3\times10^{-3}, 1\times10^{-3}, 3\times10^{-4}\}$.

Diffusion Language Model. We adapt this pipeline into a Masked Discrete Diffusion Language Model (MDLM; Sahoo et al., 2024; Lou et al., 2024) trainer with four modifications: (i) a bidirectional attention patch on the GPTNeoX backbone, (ii) a forward absorbing-state corruption step, (iii) an importance-weighted cross-entropy loss applied only at corrupted positions, and (iv) the reuse of an unused vocabulary slot as the [MASK] token. All other hyperparameters, including data order, are identical to the AR trainer.

B FURTHER EXPERIMENT PLOTS

B.1 ACTIVATIONS AND SIMILARITIES WITHOUT MASKING

We replicate a subset of our experiments while calculating activations in DLMs without masking tokens to showcase that our findings are robust over diffusion steps.

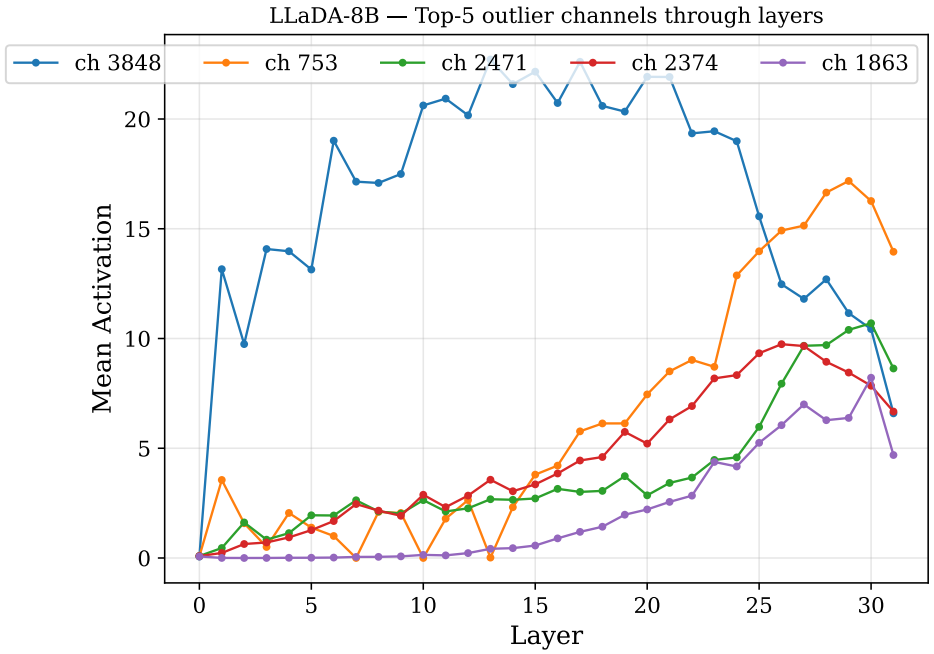


Figure 8: Similar to Figure 2a, but without including masked sequences.

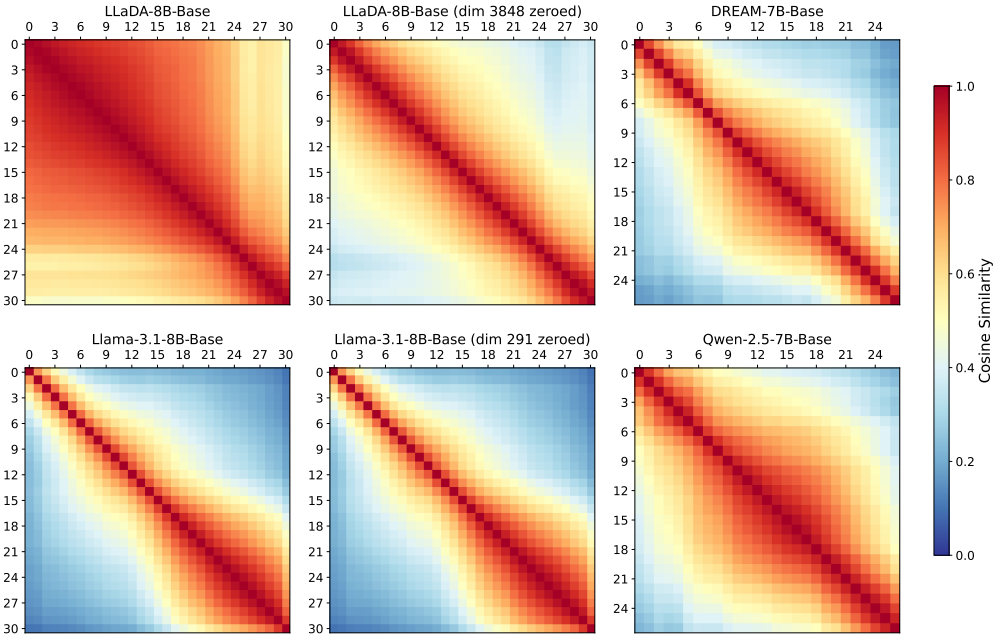


Figure 9: Similar to Figure 3 (without iLLaDA-8B), but without including masked sequences.

B.2 EXTENDED ACTIVATION PLOTS

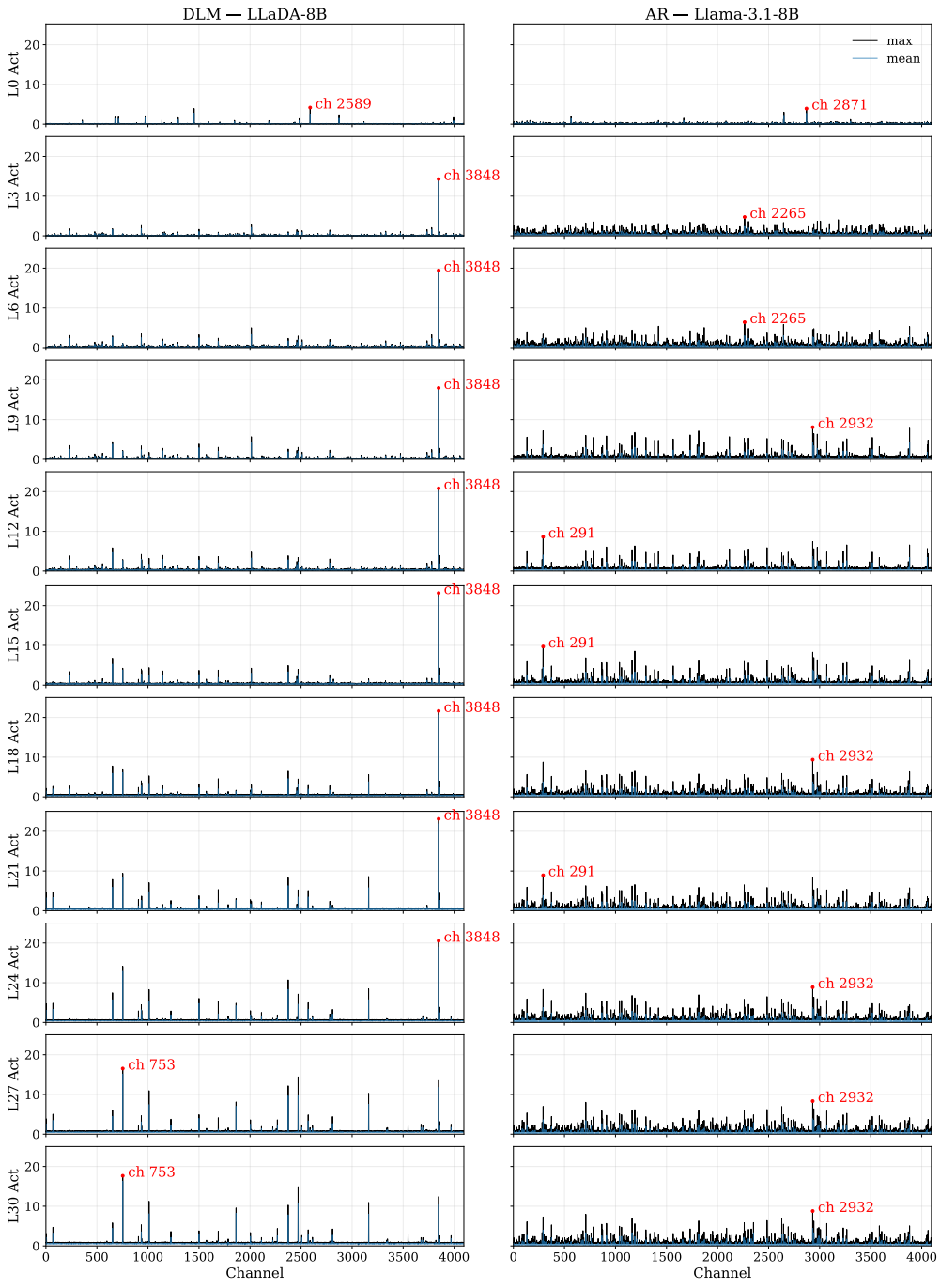


Figure 10: Extended version of Figure 1 for LLaDA-8B and Llama-3.1-8B.

B.3 Channel activation over Diffusion Steps

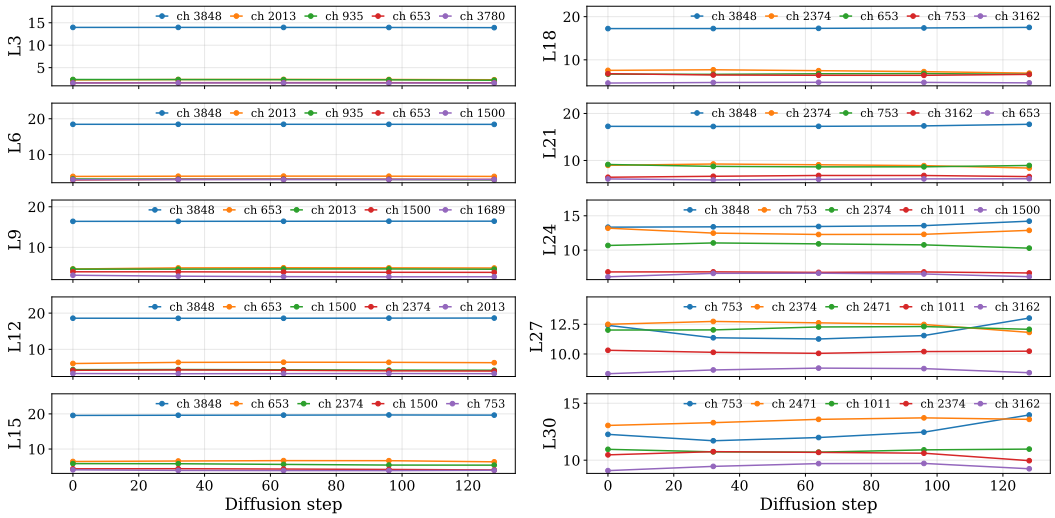


Figure 11: Channel magnitude mean of the top-5 largest (by mean) channels in LLaDA-8B, over different diffusion steps. The channel magnitudes for early-mid layers barely change across diffusion steps.

B.4 α_{HILL} for iLLaDA-8B

Figure 12 repeats the α_{Hill} analysis of Figure 4 for iLLaDA-8B. Unlike LLaDA-8B, iLLaDA-8B does not show an early-layer dip that is as pronounced: only its first two layers have noticeably low α_{Hill} values, and from layer 2 on it lies in the same range as Llama-3.1-8B.

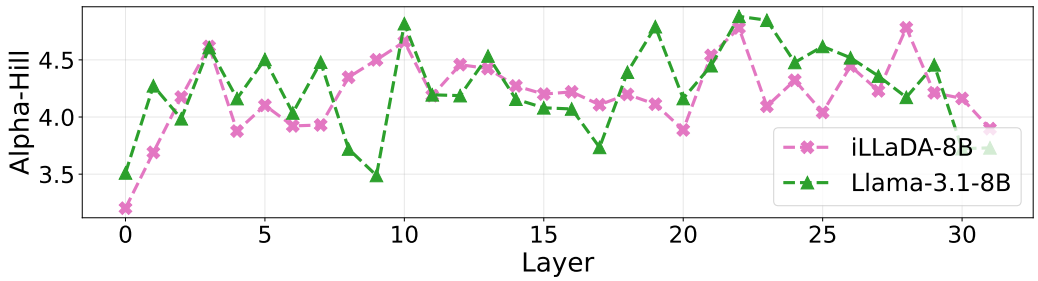


Figure 12: α_{Hill} (averaged over modules in a layer) for iLLaDA-8B and Llama-3.1-8B, as in Figure 4.