Induction in Both Directions: A Mechanistic Analysis of In-Context Learning in Masked Diffusion Language Models

Andy Catruna[1,2] **Emilian Radoi**[1,2]
[1]National University of Science and Technology POLITEHNICA Bucharest
[2]Academy of Romanian Scientists, Ilfov 3, 050044 Bucharest, Romania
{andy_eduard.catruna, emilian.radoi}@upb.ro

Abstract

While the internal mechanisms of autoregressive (AR) transformers have been studied extensively, much less is known about diffusion language models (DLMs), an emerging alternative that generates text by iterative denoising. In this work, we study how DLMs implement induction, a mechanism behind in-context learning in which the model finds a repeated context and copies the token that followed it. Our analysis compares attention-only AR models and absorbing-mask DLMs with matched architectures. We find that DLMs learn a bidirectional induction circuit, where previous-token and next-token heads write local context into the residual stream and later induction heads use it to find and copy the answer from the matching source position. The circuit is direction-symmetric, working whether the source appears in the past or in the future. When only left context is visible, matching what an AR model sees, the DLM does not outperform its AR counterpart in induction capabilities. However, we observe it has stronger induction when both sides of the masked token are visible, pointing to bidirectional context access rather than a stronger one-sided mechanism. Beyond induction, we provide causal evidence that DLMs compute the global fraction of masked tokens and use it as an implicit timestep, even though they are given no explicit timestep embedding.

1 Introduction

Large language models (LLMs) based on autoregressive (AR) transformers have become central tools for writing, coding and scientific work (Noy and Zhang, 2023; Lee et al., 2022; Peng et al., 2023; Boiko et al., 2023), making their internal mechanisms an important object of study (Bereska and Gavves, 2024). However, AR transformers are not the only relevant class of language models. Diffusion language models (DLMs) are a growing alternative that generate text by repeatedly denoising sequences rather than predicting only the next token (Austin et al., 2021; Sahoo et al., 2024). This gives them a different computational structure and the promise of parallel decoding, as shown by recent systems such as LLaDA (Nie et al., 2026), DiffusionGemma (O'Donoghue and Flennerhag, 2026), and Mercury (Khanna et al., 2025).

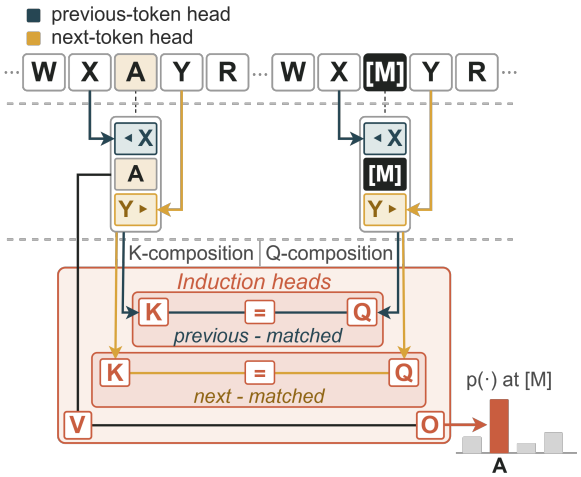


Figure 1: The bidirectional induction circuit identified in DLMs. Neighbour heads write previous-token and next-token cues. Later induction heads use their QK weights to match those cues between the mask and the source answer, and their OV path raises the probability of the answer token at the mask. The same circuit applies when the source answer is in the future.

AR language models have received extensive attention in mechanistic interpretability (Bereska and Gavves, 2024; Elhage et al., 2021; Olsson et al., 2022; Wang et al., 2022; Conmy et al., 2023), but much less is known about the internal mechanisms of DLMs (Wang et al., 2026; Dai et al., 2026; Kong et al., 2026). As DLMs use a different training objective, bidirectional attention and an iterative denoising process, mechanisms found in AR models may not transfer directly (Kong et al., 2026).

In AR models, induction heads are a canonical circuit for in-context learning (Olsson et al., 2022). They allow the model to use a repeated context

Proceedings of the 9th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP, pages 294–303
October 29, 2026 ©2026 Association for Computational Linguistics

to copy the token that followed the same context earlier, providing a basic mechanism for learning from examples inside the prompt. Understanding whether DLMs implement induction in the same way is a first step toward comparing their in-context learning mechanisms with those of AR models.

DLMs change the setting in which induction has to operate. They predict masked tokens from bidirectional context, but the true token at the prediction position is hidden behind the mask token [M] (often written as [MASK]) (Austin et al., 2021; Sahoo et al., 2024). Therefore, a DLM could use the same AR induction circuit, extend it symmetrically to copy from both past and future contexts, or use a different mechanism.

In this work, we study this problem in a controlled setting, focusing on absorbing-mask DLMs. We train matched attention-only AR transformers and DLMs, evaluate on the same repeated-token induction task, and analyze the DLM circuit with causal ablations and weight decompositions. This lets us compare behaviour and mechanism under the same architecture and data distribution.

Following standard practice in mechanistic interpretability (Elhage et al., 2021; Olsson et al., 2022), we use small models to enable circuit-level causal analysis, providing a foundation for understanding the mechanisms of large-scale DLMs.

Our analysis reveals that DLMs learn induction and implement it in a bidirectional form, as shown in Figure 1. Induction emerges abruptly, requires at least two layers, and works approximately equally well from past or future context.

This work makes the following contributions:

- We identify and provide causal evidence for a two-pathway circuit in which previous-token and next-token heads write local context, and later induction heads use this information to copy from both past and future contexts.
- We compare induction in AR models and DLMs under matched conditions and show that DLMs outperform AR models in terms of induction only when they can exploit bidirectional context, and not when restricted to the left-context setting used by AR models.
- We show that DLMs encode the global mask fraction as an implicit timestep without having an explicit timestep embedding. A linear probe can recover the global mask fraction from the residual stream at a single position after the first attention layer, and patching the mask-rate direction causally changes prediction entropy.

2 Related Work

Mechanistic Interpretability of Transformer Language Models. Autoregressive transformers have been the main focus of mechanistic interpretability in language models. Prior work introduced tools for analyzing transformer circuits, including residual-stream decomposition, attention-head circuits, and QK/OV factorizations (Elhage et al., 2021). This line of work identified induction heads as a central mechanism for in-context learning in AR models, where earlier tokens provide the features needed to copy from repeated contexts (Olsson et al., 2022).

Mechanistic interpretability has also developed causal and representation-level methods for studying model behaviour (Mueller et al., 2026). Circuit analyses and path patching (Wang et al., 2022; Goldowsky-Dill et al., 2023; Conmy et al., 2023) test which components are necessary for a behaviour, while probing and sparse-feature methods (Belinkov, 2022; Huben et al., 2024; Templeton et al., 2026) study what information is represented in the residual stream. Work on superposition further shows why individual neurons may not correspond cleanly to single features (Elhage et al., 2022). Our DLM analysis draws on several of these approaches, combining mean ablations, QK/OV decomposition, and linear probing.

Diffusion Language Models. DLMs produce text by gradually denoising corrupted sequences instead of generating tokens one by one from left to right. Early work developed discrete diffusion objectives, including absorbing-state and more general transition processes, and later work adapted these objectives to language modeling (Austin et al., 2021; Gulrajani and Hashimoto, 2023; Lou et al., 2023; Sahoo et al., 2024; Shi et al., 2024). We use DLM as the broad term in this paper, and our experiments focus on absorbing-mask DLMs, which prior work often calls masked diffusion models (MDMs) (Ou et al., 2025; Zheng et al., 2025).

Recent work has made DLMs more competitive by scaling masked diffusion models, adapting AR models into diffusion models, and combining autoregressive and diffusion-style generation (Nie et al., 2025; Gong et al., 2025; Arriola et al., 2025). Large DLMs further show that diffusion-based gen-

eration is becoming a practical alternative to standard AR decoding (Nie et al., 2026; Ye et al., 2025; Khanna et al., 2025; O'Donoghue and Flennerhag, 2026). These models are often motivated by faster or more parallel decoding, but their bidirectional denoising objective (Artetxe et al., 2022; Patel et al., 2022) also changes the internal computation for in-context learning (Kim et al., 2025).

Interpretability of Masked Diffusion Models. Mechanistic work on masked diffusion models is still emerging. Recent studies analyze sparse features in DLMs, attention-floating behaviour during denoising, mechanism changes when AR models are post-trained into masked diffusion models, and theoretical links between masked diffusion and learned-order autoregression (Wang et al., 2026; Rulli et al., 2026; Dai et al., 2026; Kong et al., 2026; Garg et al., 2025). These papers show that DLMs can share some behaviour with AR models while also changing how information is routed. Our work instead identifies the DLM induction circuit and provides causal evidence for its two-pathway structure in a matched AR-DLM setting.

3 Method

3.1 Matched Autoregressive and Diffusion Models

We train matched attention-only AR and DLM transformers at depths $L \in \{1, 2, 3\}$. The AR models use causal attention and next-token prediction, whereas the DLMs use bidirectional attention and an absorbing-mask diffusion objective (Austin et al., 2021; Sahoo et al., 2024). Apart from the objective and attention mask, the architecture and training recipe are the same for both types of models. For the main behavioural comparisons, we train three independent seeds for each family and depth, allowing us to compare both induction emergence and final induction performance across model families.

3.2 Measuring Induction

We evaluate induction capabilities on sequences constructed with random tokens where only in-context information can be utilized to predict the correct tokens (Olsson et al., 2022). Each sequence has length 512, with the first 256 tokens sampled uniformly at random and repeated as the second half. Let `W X A Y R` denote a five-token source window with answer token `A`, and let `W X` [`M`] `Y R` denote the corresponding query window with `A` replaced by the mask token [`M`]. In forward induction, the source is in the first copy and the query in the second, whereas in reverse induction their order is swapped. In total, we use 128 such sequences for evaluating each model. We mask a fixed, deterministic set of non-adjacent positions and score 3,328 masked tokens for the forward evaluation and 3,328 masked tokens for the reverse evaluation.

We use random tokens so that semantic and natural-language frequency effects cannot explain performance and the model must use the in-context information to identify the masked token. We score induction by how much repetition helps the model predict the correct token: for each masked position, we compare the model's log probability on the correct token in the repeated sequence to its log probability on the same token in a non-repeated control sequence with the same mask layout. The induction score is this difference in nats, averaged over masked positions.

AR models have no mask token [`M`], so we evaluate them with next-token prediction on the same sequences: the model reads the sequence without any masks and predicts each scored position from the tokens to its left, using the same position-matched control. This makes the AR score directly comparable to the DLM forward score, while the reverse setting has no AR counterpart.

3.3 Circuit Analysis

In order to localize the circuits, we utilize two complementary measurements. First, we compute structural attention fingerprints, such as how much a head attends from a masked position i to $i-1$, to $i+1$, or to the source answer position in the other copy, which has no mask tokens. Second, we perform mean ablations: for each head, we replace its output with its per-position mean over non-repeated control prompts and recompute the induction score (Wang et al., 2022). This approach helps us identify which heads attend to relevant information as well as which are actually causally important for induction.

For weight-level analysis, we use DLMs without LayerNorm, obtained by converting models trained with LayerNorm, so that the residual stream can be decomposed into additive components. For QK analysis, we decompose an induction head's attention logit from [`M`] to the source answer into query-component by key-component terms. Following Elhage et al. (2021), a head uses Q-composition when an earlier head's output contributes to its

query, and K-composition when it contributes to its key. For OV analysis, we pass source-token embeddings through an induction head's value and output matrices and then through the unembedding, giving an effective token-to-token logit map.

3.4 Training and Implementation Details

All models use $d_{\text{model}} = 512$, 16 attention heads with $d_{\text{head}} = 32$, and sequence length 512. The DLM objective uses $T = 100$ timesteps and no explicit timestep embedding. Models are trained on FineWeb-Edu sequences (Penedo et al., 2024) for 250k steps with batch size 128, learning rate 10^{-3}, 10k warmup steps, and cosine decay to 10% of the peak learning rate. We intentionally train in the overtraining regime commonly used in mechanistic interpretability work (Olsson et al., 2022; Nanda et al., 2023). After training, AR models have lower validation perplexity at every depth (e.g., 64.5 vs. 117.5 at L2), although the DLM value is an ELBO-based upper bound on the true perplexity (Sahoo et al., 2024), so the gap may be smaller than these numbers suggest. However, we observe that in some settings, DLM induction is stronger.

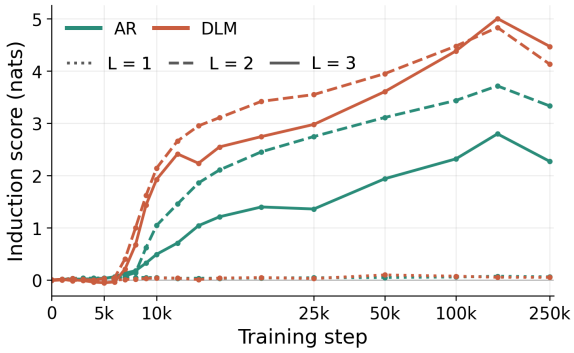
Both model families use the GPT-2 BPE tokenizer (Radford et al., 2019), and for DLMs we add one extra mask token [M]. We train all models with AdamW (Loshchilov and Hutter, 2017) with weight decay 0.1 and gradient clipping at 1.0. For DLMs, each batch samples a timestep $t \in \{1, \ldots, T\}$ uniformly and independently masks each position with cosine probability $1 - \cos((t/T)\pi/2)$. We average the loss only over masked positions.

Behavioural results are measured on AR and DLM models trained with LayerNorm. For mechanistic analysis, we use folded no-LayerNorm DLMs, which we obtain as follows (Nanda and Bloom, 2022; Heimersheim, 2024): we fully train the DLMs with LayerNorm, then replace each LayerNorm's per-token standard deviation with a constant calibrated on training data. This makes every LayerNorm a linear map, which we fold into the adjacent weight matrices. Finally, we briefly fine-tune the resulting no-LayerNorm models until they recover the performance of the LayerNorm models.

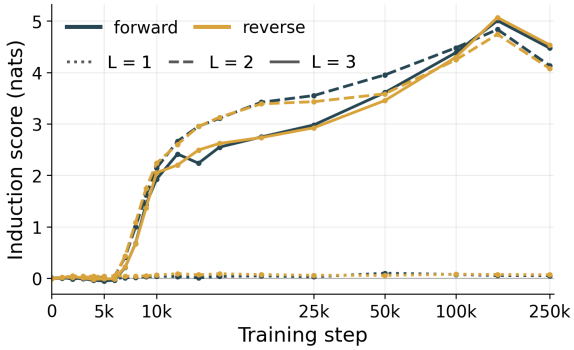
This transformation approximately preserves model quality: at L2, the folded no-LayerNorm DLMs reach nearly the same validation perplexity as the LayerNorm models (117.2 vs. 117.5), and the folded L2 and L3 models show the same induction circuit. Their induction scores stay strong but are not identical: forward scores are 3.67 ± 0.15 nats at L2 and 4.64 ± 0.09 at L3 (Table 1), versus 4.13 ± 0.17 and 4.47 ± 0.31 for the LayerNorm models.

4 Results

We first compare induction behaviour in AR and DLM models, then localize the heads that implement the DLM circuit and test their causal role. Afterward, we analyze how their QK and OV weights find and copy the source answer token, and finally show that DLMs compute and use the global mask rate as an implicit timestep.



(a) Induction emerges abruptly and peaks around 150k steps.



(b) DLM induction is direction-symmetric.

Figure 2: Induction phase change. The induction score is the log-probability gain on repeated versus control sequences, in nats. Scores are three-seed averages and the step axis is symlog.

4.1 DLM Induction Is Bidirectional

Induction appears abruptly during training and requires at least two layers, as shown in Figure 2a. AR and DLM models with one layer stay near zero induction score, while L2 and L3 models transition from near-zero to strong induction after roughly 8-24k training steps, with the DLMs transitioning earlier than the matched AR models. This result is consistent with the expected structure of an induction circuit: one layer is enough to write local neighbour features, but a later layer is needed to use

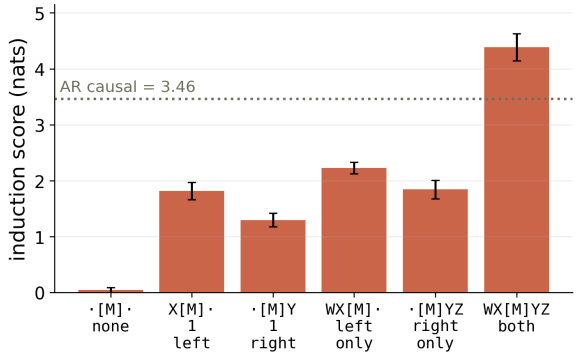


Figure 3: Context access at L2, averaged over three seeds. The DLM's advantage over AR appears when both sides of the masked token are visible, and the same pattern holds at L3. Axis schematics are abbreviated from the true ± 4 window.

those features to retrieve the matching token (Elhage et al., 2021; Olsson et al., 2022).

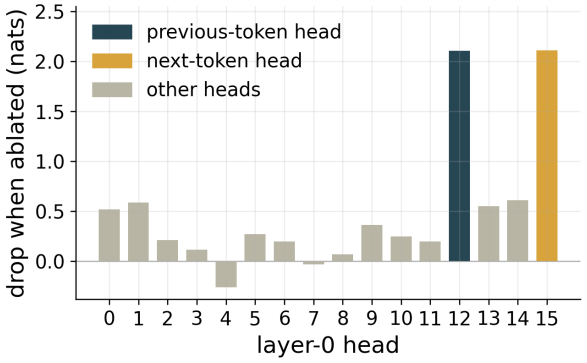
Once induction emerges, it is nearly direction-symmetric in the DLM. As shown in Figure 2b, DLMs perform approximately the same on forward and reverse induction, with nearly identical scores at the end of training (e.g., 4.13 ± 0.17 forward vs. 4.07 ± 0.25 reverse nats at L2). This shows the models retrieve the answer equally well from either direction. An AR model has no corresponding reverse setting because of its causal attention mask.

The DLM has stronger induction when it can see both sides of the masked token (Figure 3). To test this, we control which of the mask's neighbours stay visible within four tokens on each side: none, a single left or right neighbour, the four left, the four right, or all eight. Everything outside this window stays visible, including the whole source copy, and extra masks at distant positions keep the total number of masks equal across conditions, so only the local context around the mask varies.

In the left-only setting, which matches the information available to an AR model, the DLM scores at or below the AR baseline (2.23 vs. 3.46 nats at L2). The DLM has stronger induction only when both sides are visible, with a score close to the sum of the two one-sided scores (4.39 at L2). This shows that its advantage comes from seeing both sides of the mask rather than from a stronger one-sided induction mechanism.

4.2 The Circuit Splits Into Previous-Token and Next-Token Pathways

In order to localize the first part of the circuit, we label layer-0 heads by whether they attend to the previous token or the next token around the mask,



(a) The previous-token and next-token heads dominate the layer-0 mean-ablation profile.

(b) A group of layer-1 induction heads attends from the mask to the source answer position.

Figure 4: Circuit localization in one representative DLM run. Across the other DLM runs, head indices differ but the same pattern appears.

and then test their causal role with mean ablation. Previous-token heads attend from the mask to the token on its left, while next-token heads attend from the mask to the token on its right.

Ablating each layer-0 head individually singles out one dominant head of each type. In the representative run shown in Figure 4a, removing the previous-token head or the next-token head causes the largest drops in induction score, while removing any other head produces much smaller changes. This indicates that bidirectional induction starts by writing two local facts into the residual stream: what is immediately to the left of the masked token and what is immediately to its right.

The second step of the circuit appears in the next layer. As shown in Figure 4b, a few layer-1 heads attend strongly from the mask to the source answer position in the other copy. We identify these as induction heads: they use the neighbour information written at layer 0 to find the source answer and copy it back to the masked position. We then label each induction head by which layer-0 pathway it reads, determined with QK decomposition.

The same circuit replicates across depths and

depth	run	forward score (nats)	reverse score (nats)	previous-token drop (nats)	next-token drop (nats)	# induction heads	source attention range
L=2	A	3.59	3.70	2.10	2.11	4	0.25-0.42
L=2	B	3.54	3.82	2.07	1.62	4	0.25-0.43
L=2	C	3.88	3.84	2.47	1.88	4	0.34-0.56
L=3	A	4.73	4.85	3.02	3.30	5	0.44-0.63
L=3	B	4.67	4.67	3.30	2.57	4	0.56-0.70
L=3	C	4.51	4.48	3.00	2.16	4	0.53-0.65

Table 1: Replication of the two-pathway circuit across the six folded no-LayerNorm DLMs used for mechanistic analysis. Drops come from ablating one head at a time, and source attention is measured from the mask to the source answer position for the selected induction heads.

training seeds. Table 1 lists the corresponding heads for the six folded DLMs (three L2 and three L3 models). Every run has one dominant previous-token head, one dominant next-token head, and a group of layer-1 induction heads whose source attention stands clearly apart from the rest (0.25-0.70 vs. ≈ 0.03 for the next-highest head). Note that the two per-head ablation drops can sum to more than the full induction score because both ablations disrupt the same downstream copy step. The specific head indices change across runs, but the organization of the circuit is stable. At L3, the final layer adds a weaker group of induction-like heads that read either the layer-1 induction heads or the layer-0 neighbour heads, but the layer-0 to layer-1 circuit remains the dominant path.

We next test whether the two pathways operate independently, using conflict prompts. For the query window `X` [`M`] `Y`, we provide two source windows: `X A Q` matches only on the left and supports answer `A`, while `J B Y` matches only on the right and supports answer `B`. The model assigns the two answers nearly equal probability ($\log p(\texttt{A}) - \log p(\texttt{B})$ of -0.01 to $+0.32$ across the six DLMs, vs. $+2.1$ to $+2.8$ when both windows support `A`). Ablating one of the two layer-0 heads tips the prediction toward the other pathway: removing the previous-token head shifts it toward `B`, and removing the next-token head shifts it toward `A`. This shows that either pathway alone can steer the output.

The two-pathway description is a simplification of a slightly wider local circuit. Besides the dominant heads for $i-1$ and $i+1$, we also find weaker layer-0 heads that attend to $i-2$ and $i+2$. To measure how much local context is used, we construct sequences where only a short segment around the masked position matches the source, instead of the whole sequence, and vary the segment length. The score rises from 0.03 nats with no matching neighbours to 2.28 with one matching neighbour on each side and 4.29 with two, while longer segments add a negligible amount. This suggests an effective induction window of five tokens, although models with more heads and layers may learn a wider one. Ablating a ± 2 head reduces the score by only 0.4-0.7 nats, compared with about 2 nats for a ± 1 head. Including the ± 2 matches nearly doubles the score because the extra context sharpens the induction heads' attention to the source.

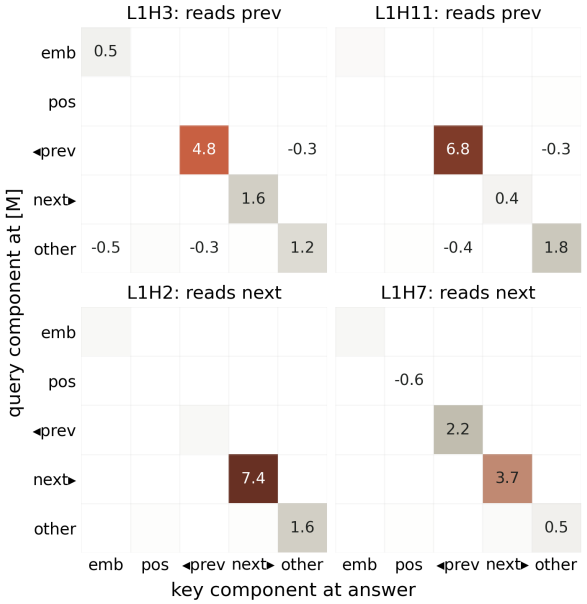


Figure 5: QK mechanics in representative induction heads. Each matrix decomposes the attention logit from the mask token [`M`] to the source answer into query-component by key-component terms. The largest term is the one where the head's own layer-0 pathway supplies both the query and the key.

4.3 Weight Analysis Reveals the QK and OV Mechanics of the Circuit

We then examine whether the localized heads implement the expected computation in their weights.

For each induction head, we decompose the attention logit from the masked position to the source answer into QK contributions from residual-stream components (token embedding, positional embedding, layer-0 head outputs). This lets us identify which earlier component supplies the query at the mask and which component supplies the key at the source position.

The QK decomposition shows that induction heads find the source by matching the same local-neighbour feature in the query and the key. As shown in Figure 5, previous-token induction heads get their largest QK contribution from the previous-token layer-0 head at both the masked position and the source position. Next-token induction heads show the same pattern for the next-token layer-0 head. Across all selected induction heads in the six DLMs, the term where the head's own layer-0 pathway supplies both the query and the key reaches 2.7-8.3 attention logits, while every other term stays below 2.4.

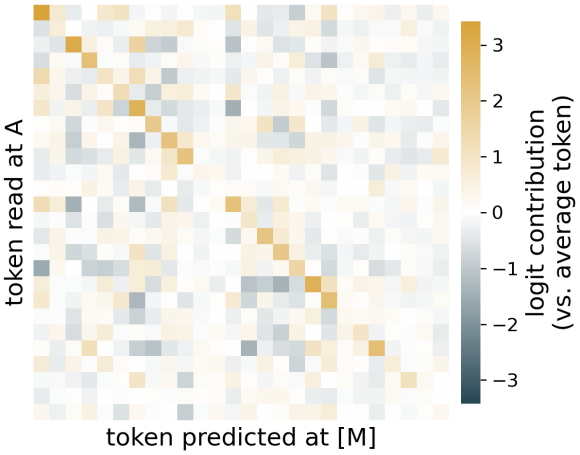


Figure 6: OV mechanics in one representative induction head, evaluated on a fixed random sample of 26 vocabulary tokens. Attending to token t at the source tends to raise the logit of t at the mask. Logits are relative to the average vocabulary token.

This shows that the model uses both Q-composition and K-composition: the layer-0 neighbour head helps build the query at the mask token [M] and the key at the answer token. This differs from the standard AR induction-head mechanism, where the current token is visible and can build the query from its embedding, so the composition is mainly into the key at the source position (Elhage et al., 2021; Olsson et al., 2022). In a DLM, every

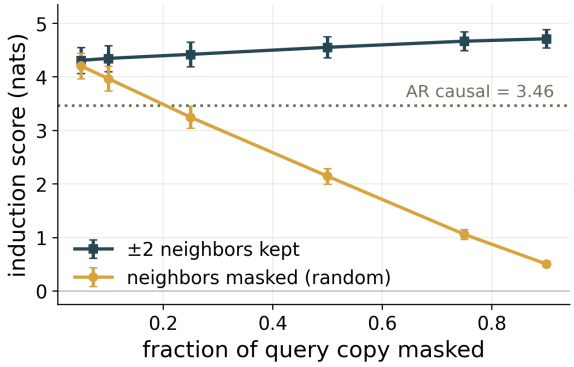


Figure 7: Induction under corruption at L2, averaged over three seeds. Induction remains strong under high global mask rates when the local window is visible. The same pattern holds at L3.

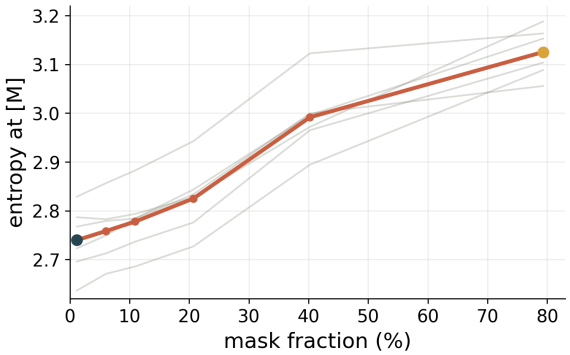
masked position carries the same embedding of the mask token [M], so the query has to come from what layer-0 heads write at the mask.

The OV path explains what happens once a head attends to the source position. As shown in Figure 6, the effective token-to-token map of an induction head is diagonal on a fixed random sample of 26 vocabulary tokens: attending to token t at the source tends to raise the logit of token t at the mask. Over the full vocabulary, this induction head makes the copied token the top-1 logit for about one in five source tokens.

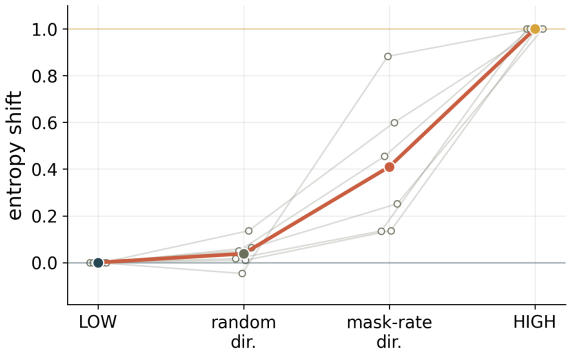
4.4 DLMs Learn an Implicit Timestep

We find that the induction circuit relies on nearby tokens, not on the overall corruption level. The DLM keeps a high induction score under heavy masking as long as the tokens near the mask stay visible (Figure 7): with the ± 2 neighbours protected, the score even rises slightly as masking reaches 90% (from 4.31 to 4.71 at L2). When the nearby tokens are masked instead, the score collapses to about 0.5. This means the models are robust to the corruption level as long as the local window contains the relevant information.

Although induction is local, the DLM still internally represents the global mask rate. To test this, we train a linear probe to predict the fraction of masked tokens from the residual stream at a single position. At the input embeddings, the probe cannot predict it ($R^2 \approx 0$), as a single token carries no information about the global mask rate. After layer-0 attention, the same probe reaches high R^2 in all six DLMs (0.82-0.91), at both masked and visible positions. This shows that the model computes a global corruption-level feature even though it is not



(a) With the local context fixed, adding distant masks raises prediction entropy at the masked position.



(b) Patching the mask-rate direction recovers part of the LOW-to-HIGH entropy shift (normalized so LOW is 0 and HIGH is 1), while an equal-size random-direction patch has little effect.

Figure 8: The implicit timestep. Gray lines show individual DLMs and the colored line the six-model mean.

given an explicit timestep embedding (Ou et al., 2025; Zheng et al., 2025). Since the mask fraction increases with the timestep under the cosine schedule, this feature serves as an implicit timestep.

The mask-rate feature seems to affect predictions. We test this by keeping the visible tokens around a masked position fixed and adding extra masks only at distant positions. As shown in Figure 8a, the entropy of the prediction at the masked position rises by 0.29-0.45 nats across the six DLMs as the global mask fraction increases. Because the local context is unchanged, this suggests that the model uses the global corruption level when making predictions.

Finally, we test whether this mask-rate representation is causally used. We patch the layer-0 mask-rate direction from a high-corruption run into the same position in a low-corruption run. As shown in Figure 8b, this patch raises entropy toward the high-corruption condition in all six DLMs, recovering 13-88% of the low-to-high entropy gap (mean $\approx$ 40%), while an equal-size random-direction patch has little effect. We expect only partial recovery as the patch changes the layer-0 residual only at the predicted position, and later attention layers can recompute the true mask rate from the other positions. Still, the patch shifts entropy in the expected direction in every model, showing that the model not only computes the mask rate but uses it as an implicit timestep.

5 Limitations

We analyze small attention-only transformers and their folded no-LayerNorm variants. This simplifies circuit analysis but leaves open if the mechanisms appear in the original LayerNorm models or larger DLMs with MLPs and more layers.

We only study absorbing-mask DLMs, where corrupted positions are replaced by the mask token [M]. Other diffusion objectives may behave differently. In particular, uniform-noise DLMs do not use a mask token (Austin et al., 2021; Lou et al., 2023), so they may not learn the same local induction circuit or the same implicit timestep.

Finally, our induction task is synthetic and controlled. Random-token repeated sequences isolate the induction circuit by eliminating semantic and frequency effects, but this does not show that the same circuit underlies in-context learning in natural language. Testing this mechanism on richer tasks remains an important next step.

6 Conclusion

We present a controlled mechanistic study of induction in matched AR and absorbing-mask DLMs. We show that DLMs learn a bidirectional circuit: previous-token and next-token pathways feed later induction heads, whose QK and OV weights retrieve and copy the source answer token.

The DLM advantage over AR comes from access to both sides of the masked token rather than a stronger one-sided mechanism. We also find that DLMs compute and causally use the global mask rate as an implicit timestep, even without an explicit timestep embedding.

Our work is a step toward understanding the mechanisms behind in-context learning beyond AR models, an important setting as other classes of language models are increasingly adopted.

Acknowledgments

This research was supported by the project "Romanian Hub for Artificial Intelligence - HRIA", Smart Growth, Digitization and Financial Instruments Program, 2021-2027, MySMIS no. 351416.

References

Marianne Arriola, Aaron Gokaslan, Justin Chiu, Zhihan Yang, Zhixuan Qi, Jiaqi Han, Subham Sahoo, and Volodymyr Kuleshov. 2025. Block diffusion: Interpolating between autoregressive and diffusion language models. In *International Conference on Learning Representations*, volume 2025, pages 50726–50753.

Mikel Artetxe, Jingfei Du, Naman Goyal, Luke Zettlemoyer, and Veselin Stoyanov. 2022. On the role of bidirectionality in language model pre-training. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 3973–3985.

Jacob Austin, Daniel D Johnson, Jonathan Ho, Daniel Tarlow, and Rianne Van Den Berg. 2021. Structured denoising diffusion models in discrete state-spaces. *Advances in neural information processing systems*, 34:17981–17993.

Yonatan Belinkov. 2022. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, 48(1):207–219.

Leonard Bereska and Efstratios Gavves. 2024. Mechanistic interpretability for ai safety–a review. *arXiv preprint arXiv:2404.14082*.

Daniil A Boiko, Robert MacKnight, Ben Kline, and Gabe Gomes. 2023. Autonomous chemical research with large language models. *Nature*, 624(7992):570–578.

Arthur Conmy, Augustine Mavor-Parker, Aengus Lynch, Stefan Heimersheim, and Adrià Garriga-Alonso. 2023. Towards automated circuit discovery for mechanistic interpretability. *Advances in Neural Information Processing Systems*, 36:16318–16352.

Xin Dai, Pengcheng Huang, Zhenghao Liu, Shuo Wang, Yukun Yan, Chaojun Xiao, Yu Gu, Ge Yu, and Maosong Sun. 2026. Revealing the attention floating mechanism in masked diffusion models. *arXiv preprint arXiv:2601.07894*.

Nelson Elhage, Tristan Hume, Catherine Olsson, Nicholas Schiefer, Tom Henighan, Shauna Kravec, Zac Hatfield-Dodds, Robert Lasenby, Dawn Drain, Carol Chen, et al. 2022. Toy models of superposition. *arXiv preprint arXiv:2209.10652*.

Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, et al. 2021. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 1(1):12.

Prateek Garg, Bhavya Kohli, and Sunita Sarawagi. 2025. Masked diffusion models are secretly learned-order autoregressive models. *arXiv preprint arXiv:2511.19152*.

Nicholas Goldowsky-Dill, Chris MacLeod, Lucas Sato, and Aryaman Arora. 2023. Localizing model behavior with path patching. *arXiv preprint arXiv:2304.05969*.

Shansan Gong, Shivam Agarwal, Yizhe Zhang, Jiacheng Ye, Lin Zheng, Mukai Li, Chenxin An, Peilin Zhao, Wei Bi, Jiawei Han, et al. 2025. Scaling diffusion language models via adaptation from autoregressive models. In *International Conference on Learning Representations*, volume 2025, pages 5046–5073.

Ishaan Gulrajani and Tatsunori B Hashimoto. 2023. Likelihood-based diffusion language models. *Advances in Neural Information Processing Systems*, 36:16693–16715.

Stefan Heimersheim. 2024. You can remove gpt2's layernorm by fine-tuning. *arXiv preprint arXiv:2409.13710*.

Robert Huben, Hoagy Cunningham, Logan Smith, Aidan Ewart, and Lee Sharkey. 2024. Sparse autoencoders find highly interpretable features in language models. In *International Conference on Learning Representations*, volume 2024, pages 7827–7845.

Samar Khanna, Siddhant Kharbanda, Shufan Li, Harshit Varma, Eric Wang, Sawyer Birnbaum, Ziyang Luo, Yanis Miraoui, Akash Palrecha, Stefano Ermon, et al. 2025. Mercury: Ultra-fast language models based on diffusion. *arXiv e-prints*, pages arXiv–2506.

Jaeyeon Kim, Kulin Shah, Vasilis Kontonis, Sham Kakade, and Sitan Chen. 2025. Train for the worst, plan for the best: Understanding token ordering in masked diffusions. *arXiv preprint arXiv:2502.06768*.

Injin Kong, Hyoungjoon Lee, and Yohan Jo. 2026. Mechanism shift during post-training from autoregressive to masked diffusion language models. *arXiv preprint arXiv:2601.14758*.

Mina Lee, Percy Liang, and Qian Yang. 2022. Coauthor: Designing a human-ai collaborative writing dataset for exploring language model capabilities. In *Proceedings of the 2022 CHI conference on human factors in computing systems*, pages 1–19.

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Aaron Lou, Chenlin Meng, and Stefano Ermon. 2023. Discrete diffusion modeling by estimating the ratios of the data distribution. *arXiv preprint arXiv:2310.16834*.

Aaron Mueller, Jannik Brinkmann, Millicent Li, Samuel Marks, Koyena Pal, Nikhil Prakash, Can Rager, Aruna Sankaranarayanan, Arnab Sen Sharma, Jiuding Sun, et al. 2026. The quest for the right mediator: Surveying mechanistic interpretability for nlp through the lens of causal mediation analysis. *Computational Linguistics*, 52(1):331–378.

Neel Nanda and Joseph Bloom. 2022. Transformerlens. https://github.com/TransformerLensOrg/TransformerLens.

Neel Nanda, Lawrence Chan, Tom Lieberum, Jess Smith, and Jacob Steinhardt. 2023. Progress measures for grokking via mechanistic interpretability. *arXiv preprint arXiv:2301.05217*.

Shen Nie, Fengqi Zhu, Chao Du, Tianyu Pang, Qian Liu, Guangtao Zeng, Min Lin, and Chongxuan Li. 2025. Scaling up masked diffusion models on text. In *International Conference on Learning Representations*, volume 2025, pages 82974–82997.

Shen Nie, Fengqi Zhu, Zebin You, Xiaolu Zhang, Jingyang Ou, Jun Hu, Jun Zhou, Yankai Lin, Ji-Rong Wen, and Chongxuan Li. 2026. Large language diffusion models. *Advances in Neural Information Processing Systems*, 38:50608–50646.

Shakked Noy and Whitney Zhang. 2023. Experimental evidence on the productivity effects of generative artificial intelligence. *Science*, 381(6654):187–192.

Brendan O'Donoghue and Sebastian Flennerhag. 2026. Diffusiongemma: 4x faster text generation. https://blog.google/innovation-and-ai/technology/developers-tools/diffusion-gemma-faster-text-generation/. Google Blog.

Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, et al. 2022. In-context learning and induction heads. *arXiv preprint arXiv:2209.11895*.

Jingyang Ou, Shen Nie, Kaiwen Xue, Fengqi Zhu, Jiacheng Sun, Zhenguo Li, and Chongxuan Li. 2025. Your absorbing discrete diffusion secretly models the conditional distributions of clean data. In *International Conference on Learning Representations*, volume 2025, pages 64972–65009.

Ajay Patel, Bryan Li, Mohammad Sadegh Rasooli, Noah Constant, Colin Raffel, and Chris Callison-Burch. 2022. Bidirectional language models are also few-shot learners. *arXiv preprint arXiv:2209.14500*.

Guilherme Penedo, Hynek Kydlíček, Anton Lozhkov, Margaret Mitchell, Colin Raffel, Leandro Von Werra, Thomas Wolf, et al. 2024. The fineweb datasets: Decanting the web for the finest text data at scale. *Advances in Neural Information Processing Systems*, 37:30811–30849.

Sida Peng, Eirini Kalliamvakou, Peter Cihon, and Mert Demirer. 2023. The impact of ai on developer productivity: Evidence from github copilot. *arXiv preprint arXiv:2302.06590*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Maximo Eduardo Rulli, Simone Petruzzi, Edoardo Michielon, Fabrizio Silvestri, Simone Scardapane, and Alessio Devoto. 2026. Attention sinks in diffusion language models. In *Findings of the Association for Computational Linguistics: ACL 2026*, pages 43244–43256.

Subham S Sahoo, Marianne Arriola, Yair Schiff, Aaron Gokaslan, Edgar Marroquin, Justin T Chiu, Alexander Rush, and Volodymyr Kuleshov. 2024. Simple and effective masked diffusion language models. *Advances in Neural Information Processing Systems*, 37:130136–130184.

Jiaxin Shi, Kehang Han, Zhe Wang, Arnaud Doucet, and Michalis Titsias. 2024. Simplified and generalized masked diffusion for discrete data. *Advances in neural information processing systems*, 37:103131–103167.

Adly Templeton, Tom Conerly, Jonathan Marcus, Jack Lindsey, Trenton Bricken, Brian Chen, Adam Pearce, Craig Citro, Emmanuel Ameisen, Andy Jones, et al. 2026. Scaling monosemanticity: Extracting interpretable features from claude 3 sonnet. *arXiv preprint arXiv:2605.29358*.

Kevin Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. 2022. Interpretability in the wild: a circuit for indirect object identification in gpt-2 small. *arXiv preprint arXiv:2211.00593*.

Xu Wang, Bingqing Jiang, Yu Wan, Baosong Yang, Lingpeng Kong, and Difan Zou. 2026. Dlm-scope: Mechanistic interpretability of diffusion language models via sparse autoencoders. *arXiv preprint arXiv:2602.05859*.

Jiacheng Ye, Zhihui Xie, Lin Zheng, Jiahui Gao, Zirui Wu, Xin Jiang, Zhenguo Li, and Lingpeng Kong. 2025. Dream 7b: Diffusion large language models. *arXiv preprint arXiv:2508.15487*.

Kaiwen Zheng, Yongxin Chen, Hanzi Mao, Ming-Yu Liu, Jun Zhu, and Qinsheng Zhang. 2025. Masked diffusion models are secretly time-agnostic masked models and exploit inaccurate categorical sampling. In *International Conference on Learning Representations*, volume 2025, pages 63186–63227.