# Dream 7B

Introducing Dream 7B, the most powerful open diffusion large language model to date.

AUTHORS
Jiacheng Ye

AFFILIATIONS
University of Hong Kong

PUBLISHED
April 2, 2025

## Contents

**Team:** Jiacheng Ye*, Zhihui Xie*, Lin Zheng*, Jiahui Gao*, Zirui Wu, Xin Jiang, Zhenguo Li, and Lingpeng Kong.

**Affiliations**: The University of Hong Kong, Huawei Noah's Ark Lab

## Introducing Dream 7B

In a joint effort with Huawei Noah's Ark Lab, we release **Dream 7B** (Diffusion reasoning model), the most powerful open diffusion large language model to date.

In short, Dream 7B:

- consistently outperforms existing diffusion language models by a large margin;
- matches or exceeds top-tier Autoregressive (AR) language models of similar size on the general, math, and coding abilities;
- demonstrates strong planning ability and inference flexibility that naturally benefits from the diffusion modeling.

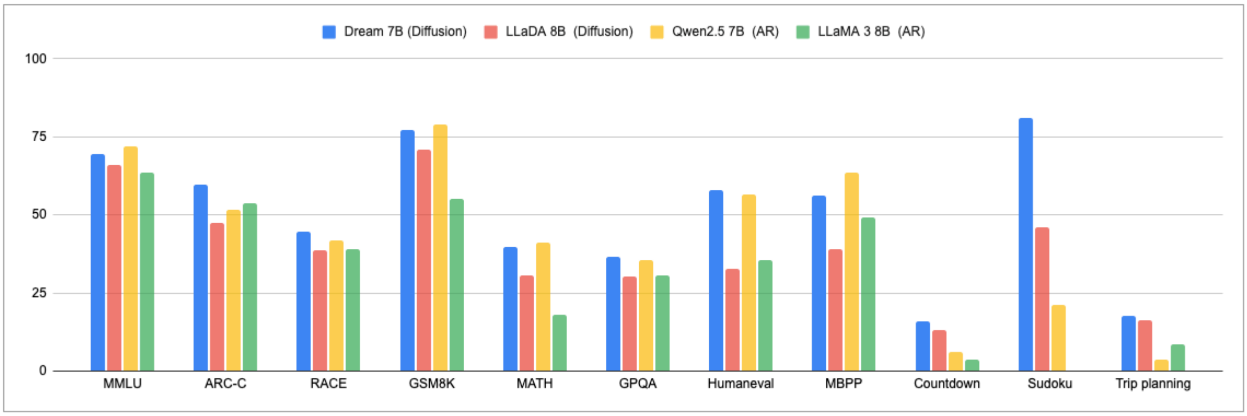
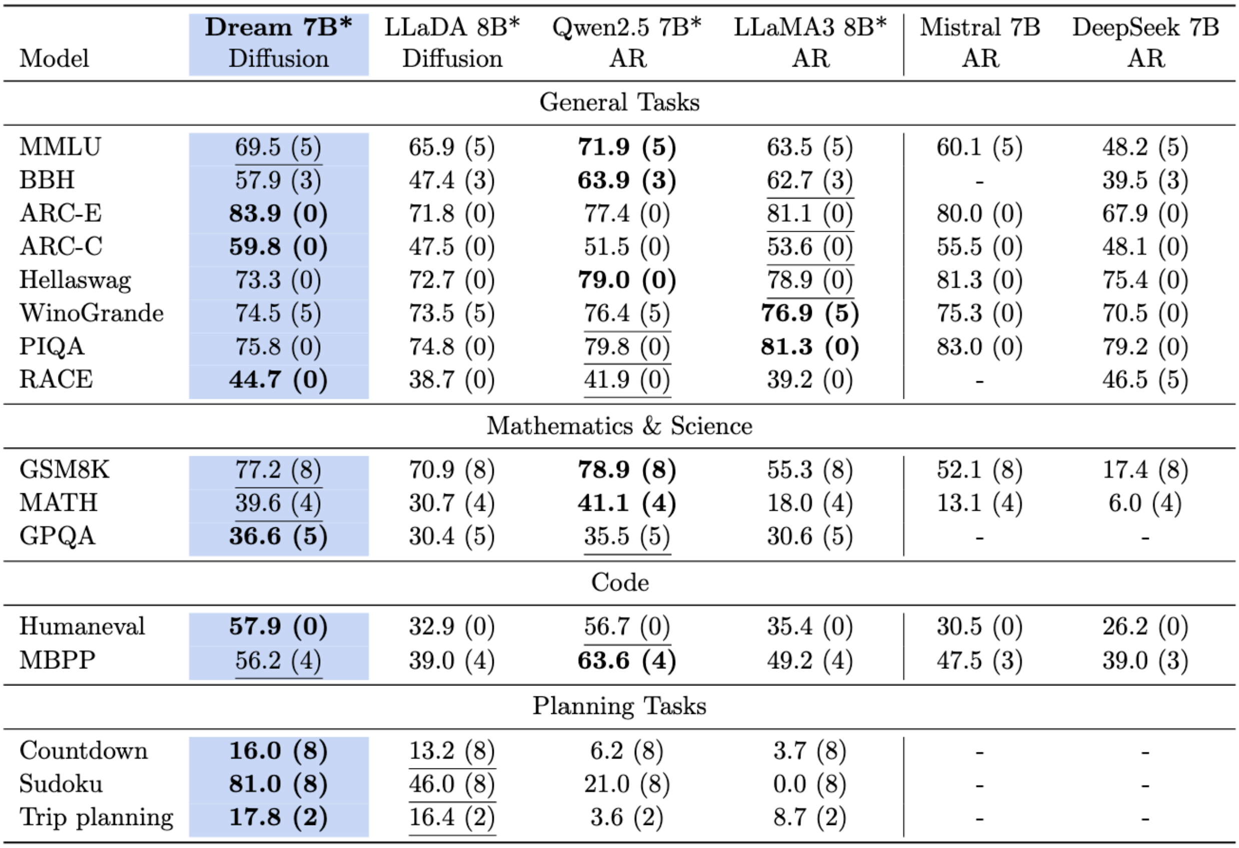


Figure: comparison of language models on general, math, coding, and planning tasks.



| | **Dream 7B*** | LLaDA 8B* | Qwen2.5 7B* | LLaMA3 8B* | Mistral 7B | DeepSeek 7B |
|---|---|---|---|---|---|---|
| Model | Diffusion | Diffusion | AR | AR | AR | AR |
| General Tasks | | | | | | |
| MMLU | 69.5 (5) | 65.9 (5) | **71.9 (5)** | 63.5 (5) | 60.1 (5) | 48.2 (5) |
| BBH | 57.9 (3) | 47.4 (3) | **63.9 (3)** | 62.7 (3) | - | 39.5 (3) |
| ARC-E | **83.9 (0)** | 71.8 (0) | 77.4 (0) | 81.1 (0) | 80.0 (0) | 67.9 (0) |
| ARC-C | **59.8 (0)** | 47.5 (0) | 51.5 (0) | 53.6 (0) | 55.5 (0) | 48.1 (0) |
| Hellaswag | 73.3 (0) | 72.7 (0) | **79.0 (0)** | 78.9 (0) | 81.3 (0) | 75.4 (0) |
| WinoGrande | 74.5 (5) | 73.5 (5) | 76.4 (5) | **76.9 (5)** | 75.3 (0) | 70.5 (0) |
| PIQA | 75.8 (0) | 74.8 (0) | 79.8 (0) | **81.3 (0)** | 83.0 (0) | 79.2 (0) |
| RACE | **44.7 (0)** | 38.7 (0) | 41.9 (0) | 39.2 (0) | - | 46.5 (5) |
| Mathematics & Science | | | | | | |
| GSM8K | 77.2 (8) | 70.9 (8) | **78.9 (8)** | 55.3 (8) | 52.1 (8) | 17.4 (8) |
| MATH | 39.6 (4) | 30.7 (4) | **41.1 (4)** | 18.0 (4) | 13.1 (4) | 6.0 (4) |
| GPQA | **36.6 (5)** | 30.4 (5) | 35.5 (5) | 30.6 (5) | - | - |
| Code | | | | | | |
| Humaneval | **57.9 (0)** | 32.9 (0) | 56.7 (0) | 35.4 (0) | 30.5 (0) | 26.2 (0) |
| MBPP | 56.2 (4) | 39.0 (4) | **63.6 (4)** | 49.2 (4) | 47.5 (3) | 39.0 (3) |
| Planning Tasks | | | | | | |
| Countdown | **16.0 (8)** | 13.2 (8) | 6.2 (8) | 3.7 (8) | - | - |
| Sudoku | **81.0 (8)** | 46.0 (8) | 21.0 (8) | 0.0 (8) | - | - |
| Trip planning | **17.8 (2)** | 16.4 (2) | 3.6 (2) | 8.7 (2) | - | - |

Figure: comparison of language models on standard evaluation benchmarks. * indicates Dream 7B, LLaDA 8B, Qwen2.5 7B and LLaMA3 8B are evaluated under the same protocol. The best results are bolded and the second best are underlined.

We release the weights of the base and instruct models in:

- Base model: Dream-org/Dream-v0-Base-7B
- SFT model: Dream-org/Dream-v0-Instruct-7B
- Codebase: GitHub

## Why Diffusion for Text Generation?

The rapid advancement of large language models (LLMs) has revolutionized artificial intelligence, transforming numerous applications across industries. Currently, autoregressive (AR) models dominate the landscape of text generation, with virtually all leading LLMs (e.g., GPT-4, DeepSeek, Claude) relying on this same sequential left-to-right architecture. While these models have demonstrated remarkable capabilities, a fundamental question emerges: what architectural paradigms might define the next generation of LLMs? This question becomes increasingly relevant as we observe certain limitations in AR models at scale, including challenges with complex reasoning, long-term planning, and maintaining coherence across extended contexts [1, 2, 3, 4]. These limitations are particularly crucial for emerging applications such as embodied AI, autonomous agents, and long-horizon decision-making systems, where sustained reasoning and contextual understanding are essential for success.

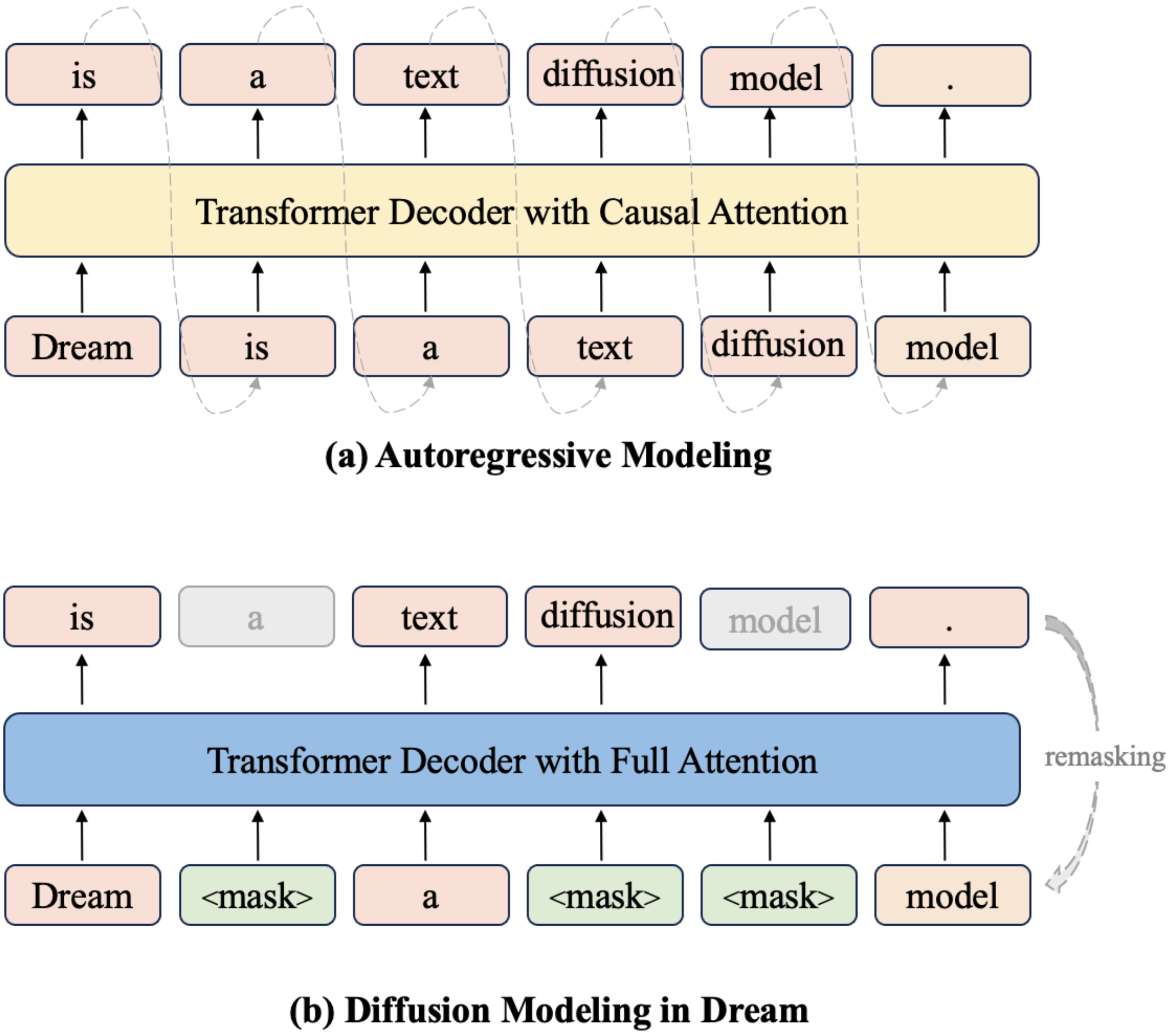
Discrete diffusion models (DMs) have gained attention as a promising alternative for sequence generation since their introduction to the text domain [5, 6, 7]. Unlike AR models that generate tokens sequentially, discrete DMs dynamically refine the full sequence in parallel starting from a fully noised state. This fundamental architectural difference unlocks several significant advantages:

- **Bidirectional contextual modeling** enables richer integration of information from both directions, substantially enhancing global coherence across the generated text.
- **Flexible controllable generation** capabilities arise naturally through the iterative refinement process.
- **Potential for fundamental sampling acceleration** through novel architectures and training objectives that enable efficient direct mapping from noise to data [8].

Recently, significant advancements have highlighted diffusion's growing potential in language tasks. DiffuLLaMA [9] and LLaDA [10] scaled diffusion language models to 7B parameters, while Mercury Coder, as a commercial implementation, has demonstrated remarkable inference efficiency in code generation. This rapid progress, combined with the inherent architectural advantages of diffusion language modeling, positions these models as a promising direction for overcoming the fundamental limitations of autoregressive approaches.

## Training

Dream 7B builds upon our team's prior effort [1] in the diffusion language model area, drawing from RDM [11]'s theoretical foundation and DiffuLLaMA [9]'s adaptation strategy. We adopt a mask diffusion paradigm with the model architecture shown below. Our training data spans from text to math and code, mainly sourced from Dolma v1.7, OpenCoder, and DCLM-Baseline, with several pre-processing and curation pipelines. Following a carefully designed training process, we pretrain Dream 7B using a mixture of the aforementioned corpus, totaling 580 billion tokens. The pretraining was done on 96 NVIDIA H800 GPUs for 256 hours. The pretraining process went smoothly overall, with occasional node anomalies, and we did not experience any unrecoverable loss spikes.



(a) Autoregressive Modeling

(b) Diffusion Modeling in Dream

Figure: comparison of autoregressive modeling and diffusion modeling in Dream. Dream predicts all the masked tokens in a shifted manner, allowing for maximumly architectural alignment and weight initialization with AR models.

We extensively studied the design choices on the 1B level and identified many valuable components, such as weight initialization from AR models (e.g., Qwen2.5 [12] and LLaMA3 [13]) and a context-adaptive token-level noise rescheduling, which enables the effective training of Dream 7B.

### AR initialization

Building on our previous work DiffuLLaMA [9], we discovered that using the weights from the existing autoregressive (AR) model serves as a non-trivial initialization for the diffusion language model. We find this design is more effective than training the diffusion language model from scratch, particularly during the early stages of training, as illustrated in the figure below.

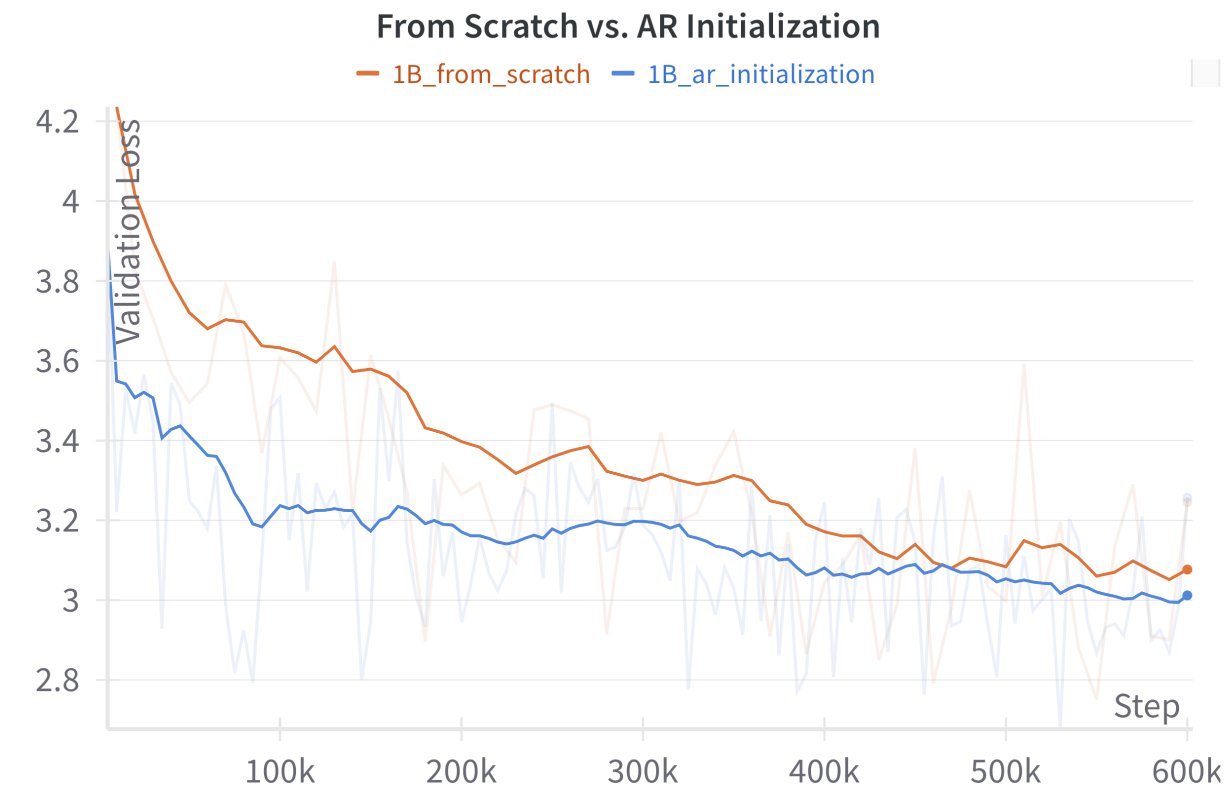


Figure: loss comparison of the from-scratch and AR-initialization with LLaMA3.2 1B training on the Dream 1B models with 200B tokens. AR initialization also experiences a high loss at the beginning due to the transition from causal attention to full attention; however, this loss remains lower compared to training from scratch throughout the training.

Dream 7B is finally initialized with weights from Qwen2.5 7B. During the training process, we find the learning rate to be especially important. If it's set too high, it can quickly wash away the left-to-right knowledge in the initial weights, providing little help in the diffusion training, while if it's set too low, it can hinder diffusion training. We meticulously selected this parameter along with the other training parameters.

Thanks to the existing left-to-right knowledge in the AR model, the diffusion model's any-order learning can be accelerated, significantly reducing the tokens and computation required for pretraining.

### Context-adaptive Token-level Noise Rescheduling

The selection of each token in a sequence depends on its context, yet we observed that previous diffusion training approaches fail to adequately account for this aspect. Specifically, in conventional discrete diffusion training, a timestep *t* is sampled to determine the sentence-level noise level, after which the model performs denoising. However, since the learning ultimately operates at the token level, the actual noise level for each token does not strictly align with *t* due to the application of discrete noise. This resulted in ineffective learning of tokens with varying levels of contextual information.

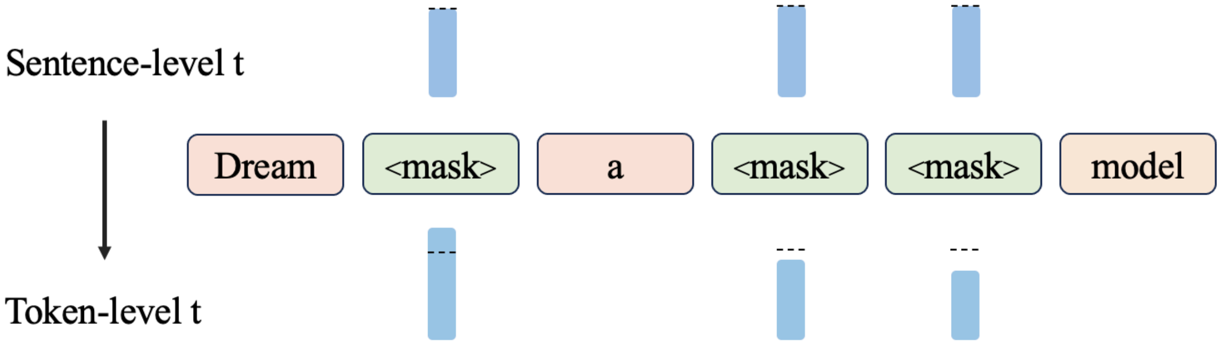


Figure: illustration of the context-adaptive token-level noise rescheduling mechanism. Dream re-decides a token-level timestep t for each mask token by measuring its context informationness.

To address this, we introduce a context-adaptive token-level noise rescheduling mechanism that dynamically reassigns the noise level for each token based on the corrupted context after noise injection. This mechanism provides more fine-grained and precise guidance for the learning process of individual tokens.

## Planning Ability

In our previous work [4, 14], we demonstrated that text diffusion exhibits superior planning capabilities in the small-scale, task-specific context. However, it remains uncertain whether a general, scaled diffusion model possesses similar abilities. Now, with Dream 7B, we can better answer this question.

We evaluated Dream on the Countdown and Sudoku tasks from [4], where we can flexibly control the planning difficulty. Our comparison included Dream 7B alongside LLaDA 8B, Qwen2.5 7B, and LLaMA3 8B, together with the latest Deepseek V3 671B (0324) for reference. All models were assessed in a few-shot setting without any training on these tasks.

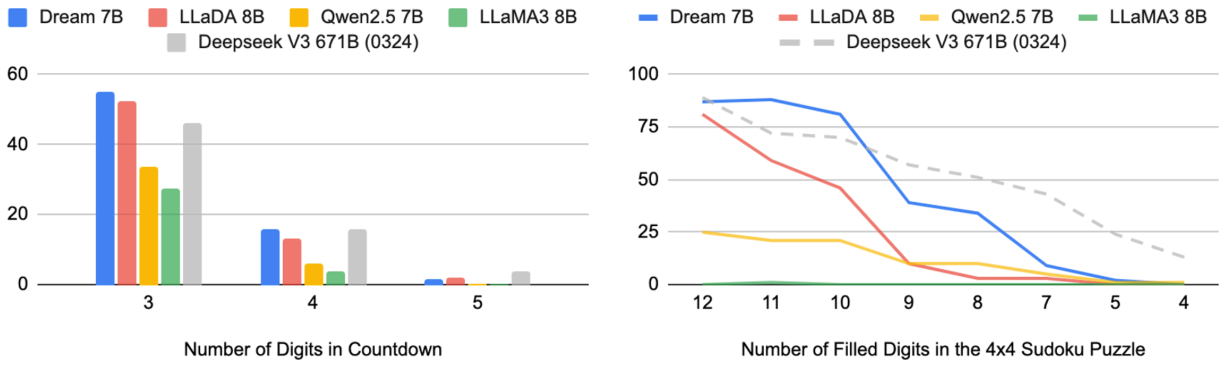
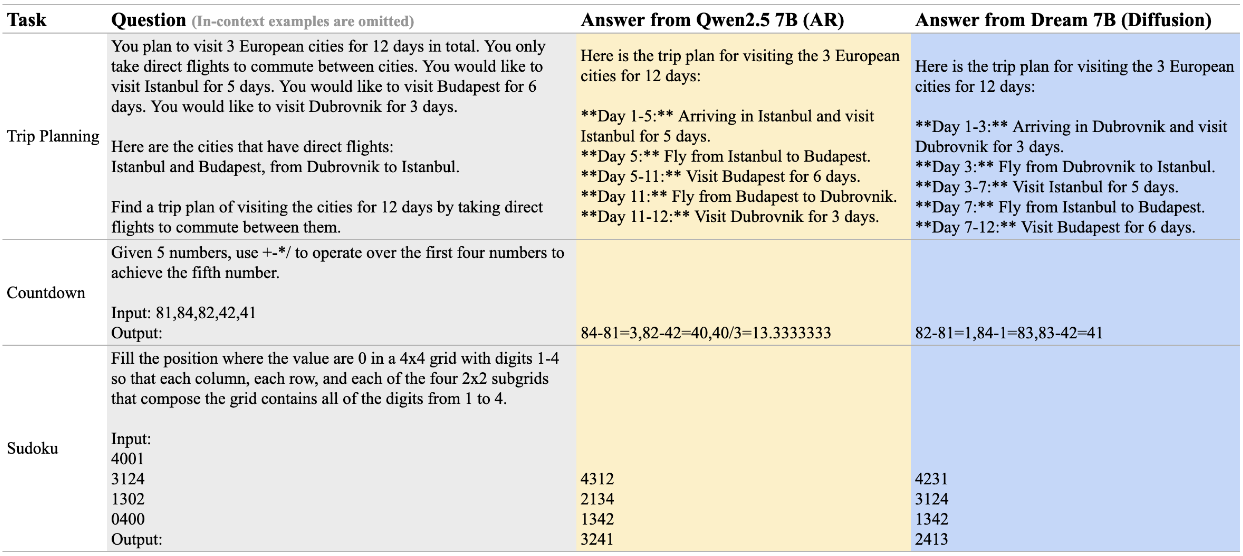


Figure: results on Countdown and Sudoku when varying planning difficulty.

It is evident that Dream outperforms other similar-sized baseline models. Remarkably, both diffusion models significantly surpass the two AR models and, at times, even the latest DeepSeek V3, despite its orders of magnitude more parameters. The intuition behind is that diffusion language models are more effective for solving problems with multiple constraints or for achieving specific objectives.

Here are some examples of Qwen 2.5 7B and Dream 7B in three planning tasks:



| Task | Question (In-context examples are omitted) | Answer from Qwen2.5 7B (AR) | Answer from Dream 7B (Diffusion) |
|---|---|---|---|
| Trip Planning | You plan to visit 3 European cities for 12 days in total. You only take direct flights to commute between cities. You would like to visit Istanbul for 5 days. You would like to visit Budapest for 6 days. You would like to visit Dubrovnik for 3 days.<br><br>Here are the cities that have direct flights:<br>Istanbul and Budapest, from Dubrovnik to Istanbul.<br><br>Find a trip plan of visiting the cities for 12 days by taking direct flights to commute between them. | Here is the trip plan for visiting the 3 European cities for 12 days:<br><br>**Day 1-5:** Arriving in Istanbul and visit Istanbul for 5 days.<br>**Day 5:** Fly from Istanbul to Budapest.<br>**Day 5-11:** Visit Budapest for 6 days.<br>**Day 11:** Fly from Budapest to Dubrovnik.<br>**Day 11-12:** Visit Dubrovnik for 3 days. | Here is the trip plan for visiting the 3 European cities for 12 days:<br><br>**Day 1-3:** Arriving in Dubrovnik and visit Dubrovnik for 3 days.<br>**Day 3:** Fly from Dubrovnik to Istanbul.<br>**Day 3-7:** Visit Istanbul for 5 days.<br>**Day 7:** Fly from Istanbul to Budapest.<br>**Day 7-12:** Visit Budapest for 6 days. |
| Countdown | Given 5 numbers, use +-*/ to operate over the first four numbers to achieve the fifth number.<br><br>Input: 81,84,82,42,41<br>Output: | 84-81=3,82-42=40,40/3=13.3333333 | 82-81=1,84-1=83,83-42=41 |
| Sudoku | Fill the position where the value are 0 in a 4x4 grid with digits 1-4 so that each column, each row, and each of the four 2x2 subgrids that compose the grid contains all of the digits from 1 to 4.<br><br>Input:<br>4001<br>3124<br>1302<br>0400<br>Output: | 4312<br>2134<br>1342<br>3241 | 4231<br>3124<br>1342<br>2413 |

Figure: generation examples from Qwen2.5 7B and Dream 7B.

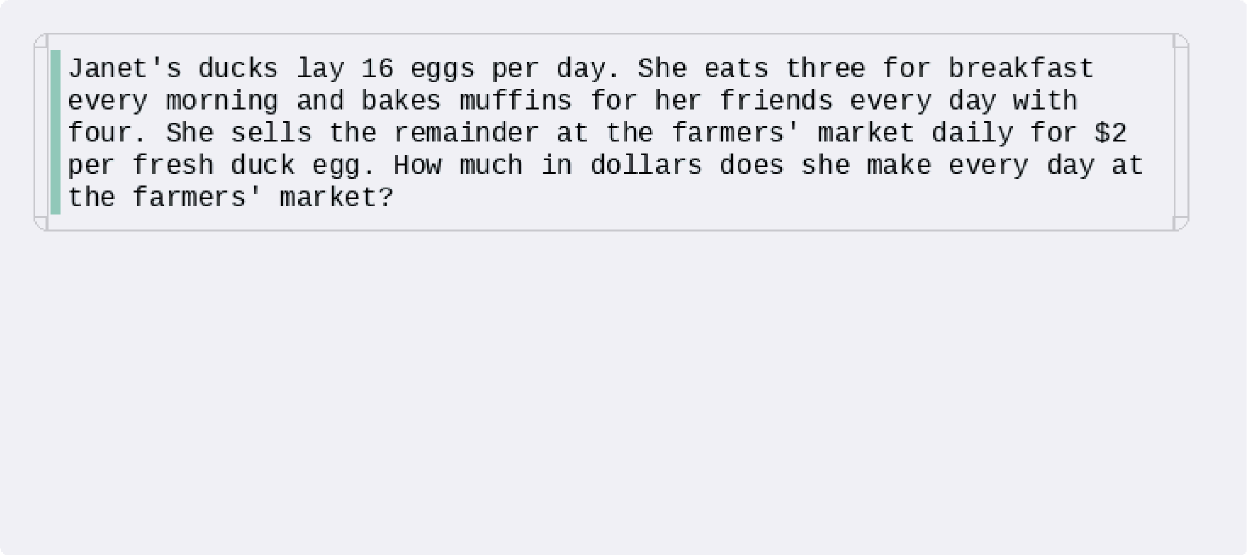
## Inference Flexibility

Diffusion models offer more flexible inference compared to AR models in the following two main aspects.

### Arbitrary Order

Diffusion models are not constrained to sequential (e.g., left-to-right) generation, enabling outputs to be synthesized in arbitrary orders—this allows for more diverse user queries.

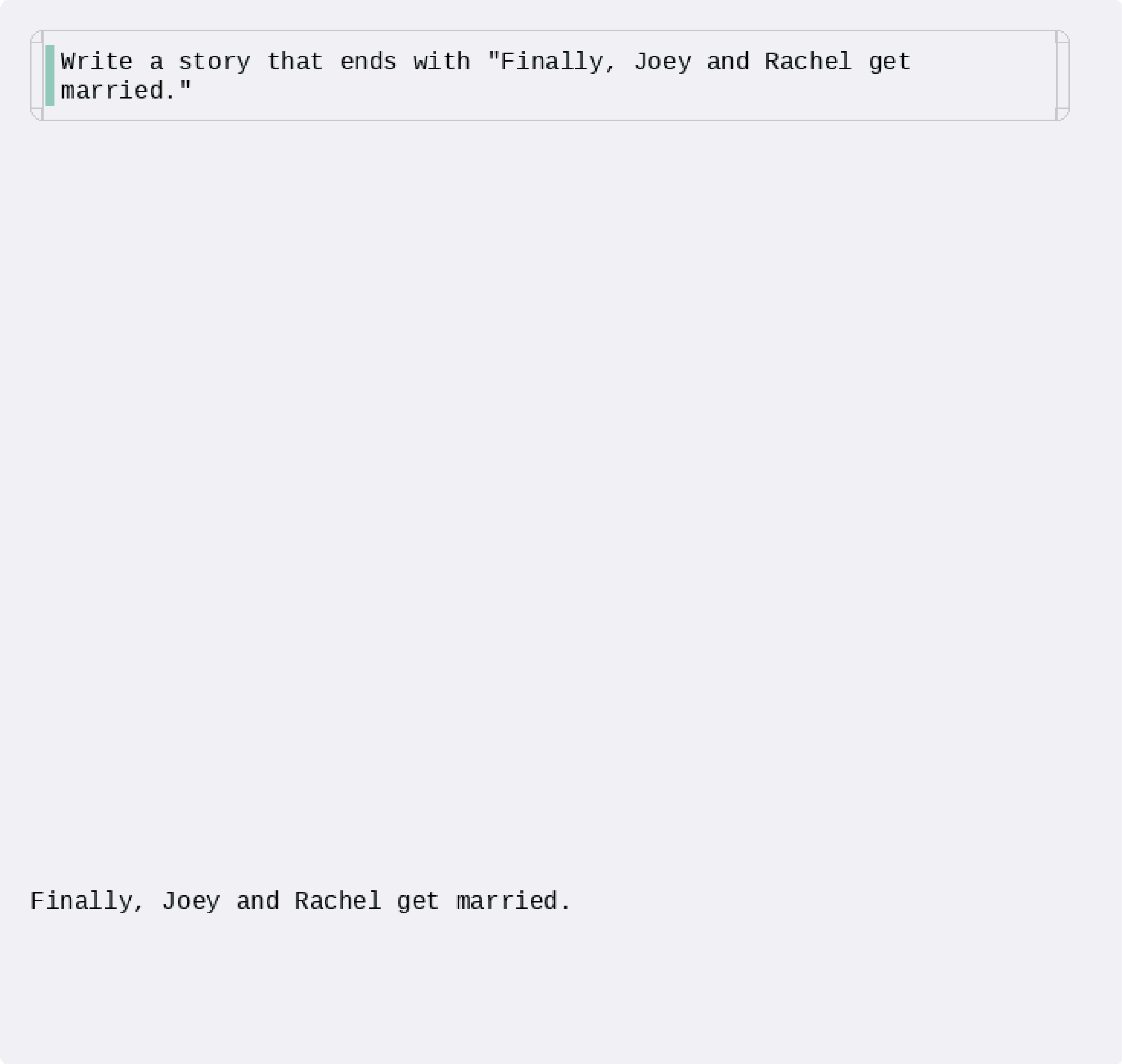
- **Completion**



```
Janet's ducks lay 16 eggs per day. She eats three for breakfast
every morning and bakes muffins for her friends every day with
four. She sells the remainder at the farmers' market daily for $2
per fresh duck egg. How much in dollars does she make every day at
the farmers' market?
```

Figure: a completion example of Dream-7B-instruct.

- **Infilling**



Write a story that ends with "Finally, Joey and Rachel get married."

Finally, Joey and Rachel get married.

Figure: an infilling example of Dream-7B-instruct with an exact ending sentence.

- **Controlling the decoding behavior**

  Different queries may have preferences for the order in which the responses are generated. One can also adjust the decoding hyperparameters to control the decoding behavior, shifting it from more left-to-right like an AR model to more random-order generation.

Please write a Python class that implements a PyTorch trainer capable of training a model on a toy dataset.

Please write a Python class that implements a PyTorch trainer capable of training a model on a toy dataset.

Please write a Python class that implements a PyTorch trainer capable of training a model on a toy dataset.

torch

the

Figure: configured to decode more in a left-to-right way like an AR model.

Figure: configured to add some randomness in the decoding order.

Figure: configured for fully randomness in the decoding order.

### Quality-speed Trade-off

In the above cases, we show one token is generated per step. However, the number of generated tokens per step (controlled by diffusion steps) can be adjusted dynamically, providing a tunable trade-off between speed and quality: fewer steps yield faster but coarser results, while more steps produce higher-quality outputs at greater computational cost. This introduces an additional dimension for inference-time scaling [15, 16, 17] that complements rather than replaces techniques like long chain-of-thought reasoning employed in large language models such as o1 and r1. This adjustable computation-quality tradeoff represents a unique advantage over traditional AR frameworks.

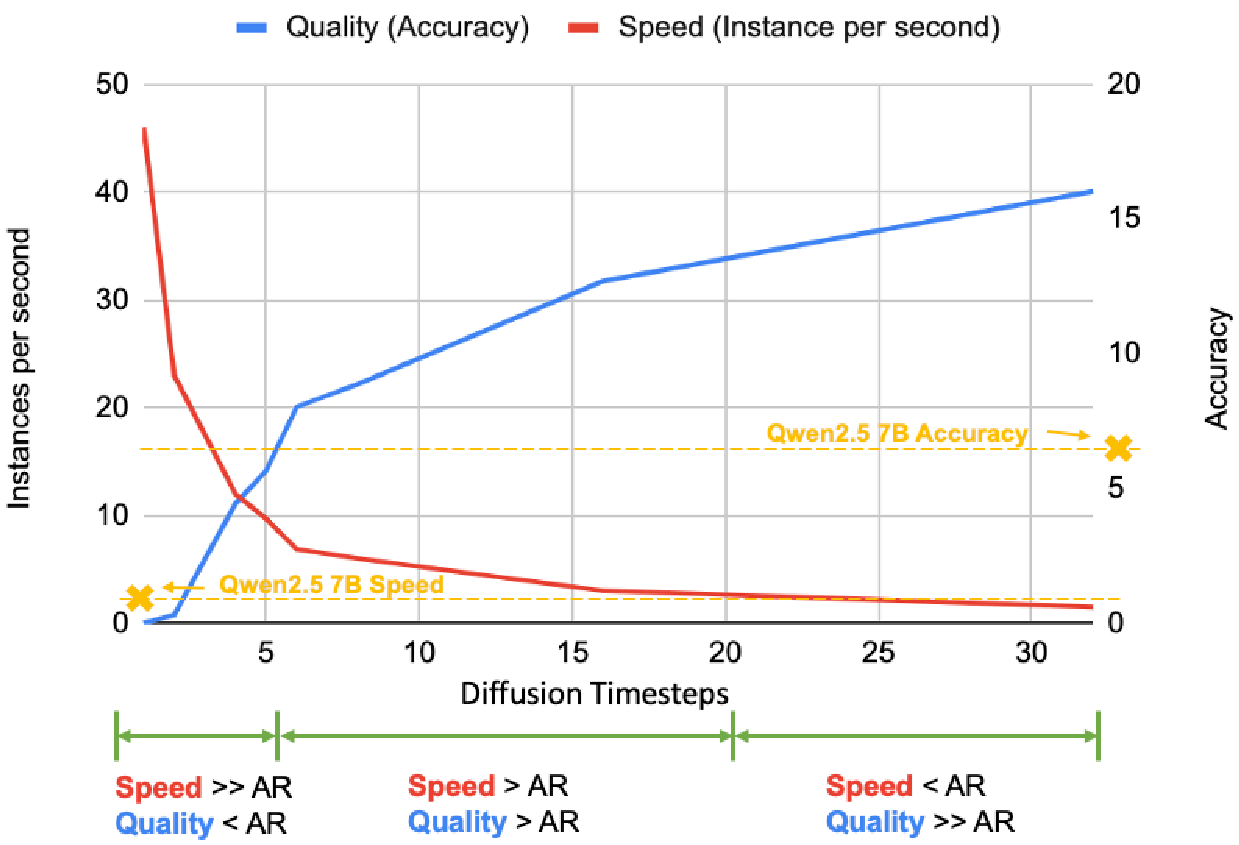
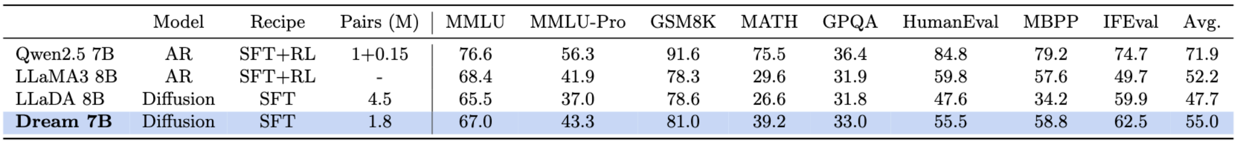


Figure: quality-speed comparison on the Countdown task for Dream 7B and Qwen2.5 7B. By adjusting the diffusion timesteps, the performance of Dream can be flexibly tuned for either speed or quality.

## Supervised Fine-tuning

As a preliminary step in post-training diffusion language models, we perform supervised fine-tuning to align Dream with user instructions. Specifically, we curate a dataset with 1.8M pairs from Tulu 3 [18] and SmolLM2 [19], fine-tuning Dream for three epochs. The results highlight Dream's potential to match autoregressive models in performance. Looking forward, we plan to explore more advanced post-training recipes for diffusion language models.



| | Model | Recipe | Pairs (M) | MMLU | MMLU-Pro | GSM8K | MATH | GPQA | HumanEval | MBPP | IFEval | Avg. |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Qwen2.5 7B | AR | SFT+RL | 1+0.15 | 76.6 | 56.3 | 91.6 | 75.5 | 36.4 | 84.8 | 79.2 | 74.7 | 71.9 |
| LLaMA3 8B | AR | SFT+RL | - | 68.4 | 41.9 | 78.3 | 29.6 | 31.9 | 59.8 | 57.6 | 49.7 | 52.2 |
| LLaDA 8B | Diffusion | SFT | 4.5 | 65.5 | 37.0 | 78.6 | 26.6 | 31.8 | 47.6 | 34.2 | 59.9 | 47.7 |
| **Dream 7B** | Diffusion | SFT | 1.8 | 67.0 | 43.3 | 81.0 | 39.2 | 33.0 | 55.5 | 58.8 | 62.5 | 55.0 |

Figure: supervised fine-tuning results.

## Conclusion

We introduce Dream, a new family of efficient, scalable, and flexible diffusion language models with carefully selected training recipes. It performs comparably to the best autoregressive models of similar size in general, mathematical, and coding tasks while especially showcasing advanced planning abilities and flexible inference capabilities.

## Citation

```
@misc{dream2025,
    title = {Dream 7B},
    url = {https://hkunlp.github.io/blog/2025/dream},
    author = {Ye, Jiacheng and Xie, Zhihui and Zheng, Lin and Gao, Jiahui and Wu, Zirui and Jiang, Xin and Li,
Zhenguo and Kong, Lingpeng},
    year = {2025}
}
```

### Footnotes

1. https://ikekonglp.github.io/dreams.html [↩]

### References

1. **Sparks of artificial general intelligence: Early experiments with gpt-4**
   Bubeck, S., Chandrasekaran, V., Eldan, R., Gehrke, J., Horvitz, E., Kamar, E., Lee, P., Lee, Y.T., Li, Y., Lundberg, S. and others,, 2023. arXiv preprint arXiv:2303.12712.
2. **Faith and fate: Limits of transformers on compositionality**
   Dziri, N., Lu, X., Sclar, M., Li, X.L., Jiang, L., Lin, B.Y., Welleck, S., West, P., Bhagavatula, C., Le Bras, R. and others,, 2024. Advances in Neural Information Processing Systems, Vol 36.
3. **The pitfalls of next-token prediction**
   Bachmann, G. and Nagarajan, V., 2024. Proceedings of the 41st International Conference on Machine Learning.
4. **Beyond Autoregression: Discrete Diffusion for Complex Reasoning and Planning**
   Ye, J., Gao, J., Gong, S., Zheng, L., Jiang, X., Li, Z. and Kong, L., 2025. The Thirteenth International Conference on Learning Representations.
5. **Argmax Flows and Multinomial Diffusion: Learning Categorical Distributions**
   Hoogeboom, E., Nielsen, D., Jaini, P., Forr{\'{e}}, P. and Welling, M., 2021. Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual, pp. 12454--12465.
6. **Structured Denoising Diffusion Models in Discrete State-Spaces**
   Austin, J., Johnson, D.D., Ho, J., Tarlow, D. and Berg, R.v.d., 2021. Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual, pp. 17981--17993.
7. **A continuous time framework for discrete denoising models**
   Campbell, A., Benton, J., De Bortoli, V., Rainforth, T., Deligiannidis, G. and Doucet, A., 2022. Advances in Neural Information Processing Systems, Vol 35, pp. 28266--28279.
8. **Consistency Models**
   Song, Y., Dhariwal, P., Chen, M. and Sutskever, I., 2023. International Conference on Machine Learning, pp. 32211--32252.
9. **Scaling Diffusion Language Models via Adaptation from Autoregressive Models**
   Gong, S., Agarwal, S., Zhang, Y., Ye, J., Zheng, L., Li, M., An, C., Zhao, P., Bi, W., Han, J. and others,, 2025. The Thirteenth International Conference on Learning Representations.
10. **Large Language Diffusion Models**
    Nie, S., Zhu, F., You, Z., Zhang, X., Ou, J., Hu, J., Zhou, J., Lin, Y., Wen, J. and Li, C., 2025. arXiv preprint arXiv:2502.09992.
11. **A Reparameterized Discrete Diffusion Model for Text Generation**
    Zheng, L., Yuan, J., Yu, L. and Kong, L., 2023. First Conference on Language Modeling.
12. **Qwen2. 5 technical report**
    Yang, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Li, C., Liu, D., Huang, F., Wei, H. and others,, 2024. arXiv preprint arXiv:2412.15115.
13. **The llama 3 herd of models**
    Grattafiori, A., Dubey, A., Jauhri, A., Pandey, A., Kadian, A., Al-Dahle, A., Letman, A., Mathur, A., Schelten, A., Vaughan, A. and others,, 2024. arXiv preprint arXiv:2407.21783.
14. **Implicit Search via Discrete Diffusion: A Study on Chess**
    Ye, J., Wu, Z., Gao, J., Wu, Z., Jiang, X., Li, Z. and Kong, L., 2025. The Thirteenth International Conference on Learning Representations.
15. **Scaling llm test-time compute optimally can be more effective than scaling model parameters**
    Snell, C., Lee, J., Xu, K. and Kumar, A., 2024. arXiv preprint arXiv:2408.03314.

16. **s1: Simple test-time scaling**
Muennighoff, N., Yang, Z., Shi, W., Li, X.L., Fei-Fei, L., Hajishirzi, H., Zettlemoyer, L., Liang, P., Candes, E. and Hashimoto, T., 2025. arXiv preprint arXiv:2501.19393.

17. **Scaling up Test-Time Compute with Latent Reasoning: A Recurrent Depth Approach**
Geiping, J., McLeish, S., Jain, N., Kirchenbauer, J., Singh, S., Bartoldson, B.R., Kailkhura, B., Bhatele, A. and Goldstein, T., 2025. arXiv preprint arXiv:2502.05171.

18. **T$\backslash$" ulu 3: Pushing frontiers in open language model post-training**
Lambert, N., Morrison, J., Pyatkin, V., Huang, S., Ivison, H., Brahman, F., Miranda, L.J.V., Liu, A., Dziri, N., Lyu, S. and others,, 2024. arXiv preprint arXiv:2411.15124.

19. **SmolLM2: When Smol Goes Big -- Data-Centric Training of a Small Language Model** [PDF]
Allal, L.B., Lozhkov, A., Bakouch, E., Blázquez, G.M., Penedo, G., Tunstall, L., Marafioti, A., Kydlíček, H., Lajarín, A.P., Srivastav, V., Lochner, J., Fahlgren, C., Nguyen, X., Fourrier, C., Burtenshaw, B., Larcher, H., Zhao, H., Zakka, C., Morlon, M., Raffel, C., Werra, L.v. and Wolf, T., 2025.