

Generating open source chess puzzles - notes

Aatu Selkee

January 8, 2026

Contents

1	Tokenization	2
2	Model architecture	2
3	RL	3

1 Tokenization

Tokenization v1

- Board
 - PNBKQKpnbkq. = 13 tokens
- Side to move
 - wb = 1 tokens (b already counted)
- Castling
 - KQkq. = 0 tokens (already counted)
- En passant
 - abcdefgh = 7 tokens (b already counted)
 - 12345678 = 8 tokens
 - -. = 1 tokens (. already counted)
 - = 16 tokens
- Half move counter
 - 0123456789 = 2 (0 and 9 new tokens)
- Full move counter
 - 0123456789 = 0 (already counted)

= 32 tokens

Total tokens do not match the number of tokens in the paper 31 (the most obvious is that "-" might be replaced with a "."). The length of the produced string is also 76 instead of 77 for some reason. This tokenization feels bad, as e.g. side to move b is completely different to board b (black to move vs black bishop).

Tokenization v2 (my current choice): Length 76, number of tokens 48 (own tokens e.g. for black bishop and black to move)

2 Model architecture

What should be used, pre- or post-normalization ([1] says post (it says that the llama papers use post, but I think they use pre), but [2] and [3] use pre-normalization to improve stability.)

If we want some new experiments, we can find out what masking schedule produces the best results after supervised learning and then apply RL to that model.

3 RL

[1] did not use the masked diffusion for the RL, and it will probably be harder than with the autoregressive model.

Compute the log-probability of the models in the same way as with autoregressive models (sum the log probabilities of the chosen tokens). The model must be called K times, where K is the amount of tokens (the latter tokens depend on the previous tokens and teacher forcing is not possible I think?). Hence, the computational complexity is a lot higher with masked diffusion than with an autoregressive model. **I may be wrong based on algorithm 2 of [4], as we call the model as many times as we have discretized the range $[0, 1]$.**

References

- [1] X. Feng, V. Veeriah, M. Chiam, M. Dennis, R. Pachauri, T. Tumieli, F. Barbero, J. Obando-Ceron, J. Shi, S. Singh, S. Hou, N. Tomašev, and T. Zahavy, “Generating creative chess puzzles,” 2025.
- [2] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, A. Rodriguez, A. Joulin, E. Grave, and G. Lample, “Llama: Open and efficient foundation language models,” 2023.
- [3] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, D. Bikel, L. Blecher, C. C. Ferrer, M. Chen, G. Cucurull, D. Esiobu, J. Fernandes, J. Fu, W. Fu, B. Fuller, C. Gao, V. Goswami, N. Goyal, A. Hartshorn, S. Hosseini, R. Hou, H. Inan, M. Kardas, V. Kerkez, M. Khabsa, I. Kloumann, A. Korenev, P. S. Koura, M.-A. Lachaux, T. Lavril, J. Lee, D. Liskovich, Y. Lu, Y. Mao, X. Martinet, T. Mihaylov, P. Mishra, I. Molybog, Y. Nie, A. Poulton, J. Reizenstein, R. Rungta, K. Saladi, A. Schelten, R. Silva, E. M. Smith, R. Subramanian, X. E. Tan, B. Tang, R. Taylor, A. Williams, J. X. Kuan, P. Xu, Z. Yan, I. Zarov, Y. Zhang, A. Fan, M. Kambadur, S. Narang, A. Rodriguez, R. Stojnic, S. Edunov, and T. Scialom, “Llama 2: Open foundation and fine-tuned chat models,” 2023.
- [4] A. Ruoss, G. Delétang, S. Medapati, J. Grau-Moya, L. K. Wenliang, E. Catt, J. Reid, C. A. Lewis, J. Veness, and T. Genewein, “Amortized planning with large-scale transformers: A case study on chess,” 2024.