Next-Token Prediction Should be Ambiguity-Sensitive: A Meta-Learning Perspective

Leo Gagnon [1] **Eric Elmoznino** [1] **Sarthak Mittal** [1] **Tom Marty** [1] **Tejas Kasetty** [1]
Dhanya Sridhar [1] **Guillaume Lajoie** [1]

Abstract

The rapid adaptation ability of auto-regressive foundation models is often attributed to the diversity of their pre-training data. This is because, from a Bayesian standpoint, minimizing prediction error in such settings requires integrating over all plausible latent hypotheses consistent with observations. While this behavior is desirable in principle, it often proves too ambitious in practice: under high ambiguity, the number of plausible latent alternatives makes Bayes-optimal prediction computationally intractable. Cognitive science has long recognized this limitation, suggesting that under such conditions, heuristics or information-seeking strategies are preferable to exhaustive inference. Translating this insight to next-token prediction, we hypothesize that low- and high-ambiguity predictions pose different computational demands, making ambiguity-agnostic next-token prediction a detrimental inductive bias. To test this, we introduce MetaHMM, a synthetic sequence meta-learning benchmark with rich compositional structure and a tractable Bayesian oracle. We show that Transformers indeed struggle with high-ambiguity predictions across model sizes. Motivated by cognitive theories, we propose a method to convert pre-trained models into Monte Carlo predictors that decouple task inference from token prediction. Preliminary results show substantial gains in ambiguous contexts through improved capacity allocation and test-time scalable inference—though challenges remain. Code is available here.

Introduction

A leading explanation for the surprising generalization capabilities of transformer-based (Vaswani et al., 2017) foundation models (Bommasani et al., 2021) is that their pretraining distribution resembles a sequence meta-learning problem (Brown et al., 2020; Xie et al., 2021; Chan et al., 2022; Wang et al., 2023; Hahn & Goyal, 2023). In this view, each document in the corpus is governed by latent factors (e.g., topic, world state), and models learn to perform implicit Bayesian inference over these factors to predict tokens effectively across domains.

In the idealized setting (Ortega et al., 2019), each sequence is generated by sampling a task $\theta \sim p^*(\theta)$ and then drawing observations from $p^*(x_{1:T} \mid \theta)$. The next-token predictor that minimizes prediction error in that setting is called the *Bayes-optimal* posterior predictive :

$$p^*(x_t \mid x_{<t}) = \int_\theta \underbrace{p^*(x_t \mid x_{<t}, \theta)}_{\text{Prediction}} \underbrace{p^*(\theta \mid x_{<t})}_{\text{Inference}} \quad (1)$$

Thus, training a model to minimize next-token prediction loss (LHS) is encouraged to implicitly perform task inference (RHS)—notably explaining how foundation models can adapt to new tasks at inference time purely by conditioning on a few input examples (In-Context Learning, ICL; Brown et al., 2020; Panwar et al., 2024). This meta-learning view, however, exposes a fundamental challenge: in high-ambiguity contexts, where the posterior over tasks $p^*(\theta \mid x_{<t})$ has high entropy, prediction becomes inherently harder due to the need to consider many plausible hypotheses θ. In fact, cognitive science has long recognized that Bayes-optimal prediction becomes intractable under resource constraints (Lieder & Griffiths, 2020). Humans respond with ambiguity-aware strategies such as heuristics (Binz et al., 2022), approximate inference (Sanborn et al., 2010), or information-seeking (Friston et al., 2017).

Following from this, we hypothesize that sequence models which allocate fixed computation per token suffer from poor capacity allocation : the more difficult high-ambiguity predictions receive too little while low-ambiguity ones receive too much. From a statistical learning angle, we suggest that ambiguity-sensitivity may serve as an effective inductive bias for general next-token prediction. While it is well understood that prediction under ambiguity causes issues *at inference* in foundation models (Liu et al., 2023; Keluskar et al., 2024), we are the first to

[1]Mila - Quebec's AI Institute. Université de Montréal. Correspondence to: Leo Gagnon <leogagnon@gmail.com>.

Proceedings of the 42^{nd} International Conference on Machine Learning, Vancouver, Canada. PMLR 267, 2025.

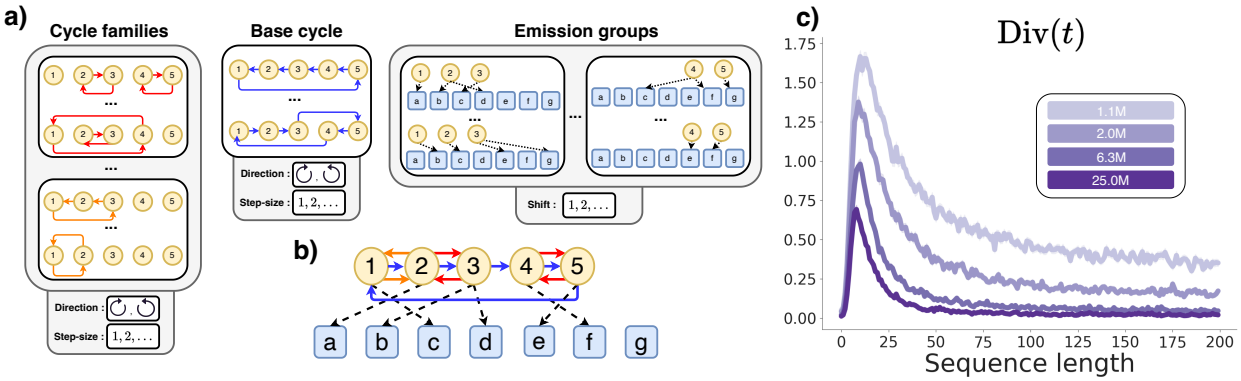

Figure 1: **The MetaHMM benchmark. a)** Latent structure of a MetaHMM environment. White rectangles contain mutually exclusive discrete choices θ_i which together define an HMM $\boldsymbol{\theta}$. Yellow circles represent hidden states and blue rectangles represent observable symbols. **b)** Example of an HMM sampled from the given building blocks. **c)** $\text{Div}_x(t)$ for different model sizes.

ground this problem in meta-learning theory, make links to resource-rational analysis (Lieder & Griffiths, 2020) and make the case for poor resource allocation *at training*.

To test this hypothesis, we introduce MetaHMM, the first synthetic benchmark for sequence meta-learning with rich compositional latent structure and an exactly computable Bayesian oracle. By evaluating learned next-token predictors against the Bayes-optimal model in Equation (1), we show that Transformers consistently underperform in high-ambiguity settings. Importantly, this gap persists across model sizes, suggesting that scale alone does not resolve ambiguity-related failures. We release our codebase for procedural and scalable MetaHMM generation.

To mitigate this issue, we propose a modular predictor that approximates Equation (1) using a Monte Carlo (MC) estimator, bootstrapped from a classical autoregressive model. Inspired by human approximate inference (Sanborn et al., 2010), our method separates task inference from token prediction, introducing useful inductive biases and allowing test-time scaling (Snell et al., 2024) by increasing the number of MC samples. In doing so, we describe a principled type of test-time scaling as adaptation to posterior entropy—grounded in both Bayesian theory and resource-rational cognition.

On MetaHMM, our method consistently outperforms the underlying sequence model in high-ambiguity settings, with gains increasing as more samples are drawn. However, performance gains diminish with larger models, suggesting this approach is especially useful when base models underfit. Adapting our method to naturalistic settings is an important direction for future work. More broadly, exploring additional types ambiguity-aware strategies may be essential for improving robustness and efficiency of foundation models. This work takes a first step in that direction.

1. The MetaHMM environment

We choose a synthetic environment to isolate and analyze the ambiguity problem without the confounding complexity of natural language. While language corpora resemble a meta-learning distribution, they go far beyond that formalism in ways that would obscure the underlying mechanisms we aim to study. Moreover, no existing sequence meta-learning benchmark possesses the following two properties (a) a non-trivial, structured space of generators that supports meaningful latent inference, and (b) a fully tractable Bayes-optimal predictor (see Section 3 for further discussion).

A MetaHMM environment consists of a family of Hidden Markov Models (HMMs; Rabiner & Juang, 1986) where each member is described by a latent code $\boldsymbol{\theta}$ which specifies how to build the HMM from a pool of shared building blocks. Concretely, each coordinate θ_i of $\boldsymbol{\theta}$ corresponds to a discrete choice which defines the HMM. Further, one can control the size of a MetaHMM by adding/removing choices; explicit size computation given in Appendix B.

The transition matrix of each HMM is composed of one *base cycle* which goes through all hidden states and multiple groups of smaller cycles from *cycle families*. For both the base cycle and cycle families, the direction and speed at which cycles are traversed can change (through a **Direction** and **Step-size** variable). Each outgoing edges of a node (after adding all cycles together) have equal probability. The emission matrix of each HMM is built from multiple different *emission groups* together partitioning the hidden states. Additionally, all groups' emission mappings can be cyclically shifted through the **Shift** variable. See Figure 1**a)**.

Importantly, our setup enables efficient and exact computation of the posterior predictive in Equation (1) using JAX (Bradbury et al., 2018) implementations of the forward algorithm (Linderman et al., 2025).

Due to the Markovian nature of HMMs, the ambiguity of

$p^*(\theta \mid x_{<t})$ decreases monotonically with sequence length. As a result, the beginning of each sequence corresponds to the high-ambiguity regime—the region where we expect models to perform most poorly. At long context length, when θ is unambiguous, we expect the Transformer to easily simulate the HMM (Rizvi-Martel et al., 2024).

Evaluation of Transformers

We generate MetaHMM environments of size $\sim$ 12,000 and train 4 different sizes of causal Transformer models p_ϕ on them (Appendix B for more details). We train on sequences of length $T = 200$ and evaluate each model by computing a symmetrized KL divergence between its posterior predictive distribution and that of the Bayes-optimal predictor:

$$\begin{aligned}\text{Div}_x(t) := \frac{1}{2} D_{KL}[p^*(x_t \mid x_{<t}) \parallel p_\phi(x_t \mid x_{<t})] \\ + \frac{1}{2} D_{KL}[p_\phi(x_t \mid x_{<t}) \parallel p^*(x_t \mid x_{<t})]\end{aligned} \tag{2}$$

providing a principled measure of the model's deviation from the ideal predictor at each position in the sequence.

The KL bump

We find that the model initially perfectly fits the Bayesian oracle, followed by a characteristic bump at short context, followed by a steady decrease towards an asymptote. That the transient bump was also noticed by (Xie et al., 2021, Fig. 7). We hypothesize that this behavior arises because, in very short contexts, the model memorize marginal token statistics, as in (Kobayashi et al., 2023). This strategy, however, fails after a few tokens given the exponential growth of possible sequences as a function of length. Aside from this subtlety, we highlight that Transformers trained with next-token loss struggle in regions of high ambiguity.

Effect of increased model size

Increasing model size leads to rapid convergence in performance at large sequence lengths, but the KL bump persists across all model scales. This finding underscores a key limitation of current autoregressive models: while they allocate uniform compute per token—often increasing slightly with position—the difficulty of prediction varies across the sequence. In high-ambiguity regions (early in the sequence), the model is under-parameterized relative to the task due to the difficulty of latent posterior inference, while in low-ambiguity regions (later positions), it is over-parameterized. We also attempted to mitigate this imbalance by training on a skewed distribution emphasizing shorter sequences (Figure 4), hoping to increase model focus on high-ambiguity regimes. However, this yielded no significant improvements, suggesting the need for more sophisticated approaches.

2. Monte-Carlo predictor

Our core idea is to approximate the Bayesian integral in Equation (1) using a Monte Carlo (MC; Robert et al., 1999) estimate: we draw multiple samples from the task posterior $p(\theta \mid x_{<t})$, compute the conditional predictions $p(x_t \mid x_{<t}, \theta)$ for each, and average the results. This approach not only offers a principled mechanism for separately allocating modeling capacity to task inference and next-token prediction during training, but also introduces a natural form of test-time scaling via the number of samples S:

$$p_{\phi,\psi}(x_t \mid x_{<t}) = \frac{1}{S} \sum_{i=1}^{S} p_\phi(x_t \mid x_{<t}, \theta_i) \tag{3}$$

$$\text{where } \theta_i \overset{\psi}{\sim} p(\theta \mid x_{<t})$$

The main challenge lies in implementing this formulation outside synthetic environments (which is still the aim of this paper), where the true latent variable θ is unknown—making it unclear how to train the components of the predictor. We address this with a three-step solution Figure 2**a)**:

1. **Latent proxy** : Replace the (in practice) unknown latent θ with a contextual embedding $z = E(x_{ctx})$, where $x_{ctx} \sim p^*_\theta$. Crucially, we restrict x_{ctx} to regimes in which the posterior inference problem would be unambiguous (e.g., large sequence lengths in MetaHMM), such that the learned z can serve as a good proxy for the true task latent. In practice, we use average-pooled hidden states from the frozen pre-trained model $p_{\phi_0}(x_t \mid x_{<t})$ to define E.
2. **Conditional predictor** : Train a conditional sequence model $p_\phi(x_t \mid x_{<t}, z)$ by fine-tuning a pre-trained model $p_{\phi_0}(x_t \mid x_{<t})$, prepending the embedding $z = E(x_{ctx})$ to its input. Sequences x_{ctx} and $x_{1:T}$ are importantly drawn from the same generator p^*_θ.
3. **Inference sampler** : Train a diffusion model parameterized by ψ to sample contextual embedding $z \sim p(z \mid x_{<t})$ conditional on $x_{<t}$ for $t \in [1, T]$.

The key intuition is that the sequence model p_ϕ is *only* used for unambiguous prediction (given z), while the diffusion model is *only* used for task inference (without access to the token to be predicted). This structure introduces inductive biases tailored to the generative process, while retaining the flexibility of learned amortized inference. Additionally, it is straightforward to derive from standard pre-trained language models through targeted fine-tuning. The idea of training a diffusion model on sentence embeddings is inspired from (Lovelace et al., 2023; 2024).

2.1. Results

We perform the aforementioned procedure on the models in Figure 1 with an additional context x_{ctx} of length 100, ensuring unambiguity of θ. Next we train a Diffusion Transformer (DiT; Peebles & Xie, 2023) to sample $z = E(x_{ctx})$

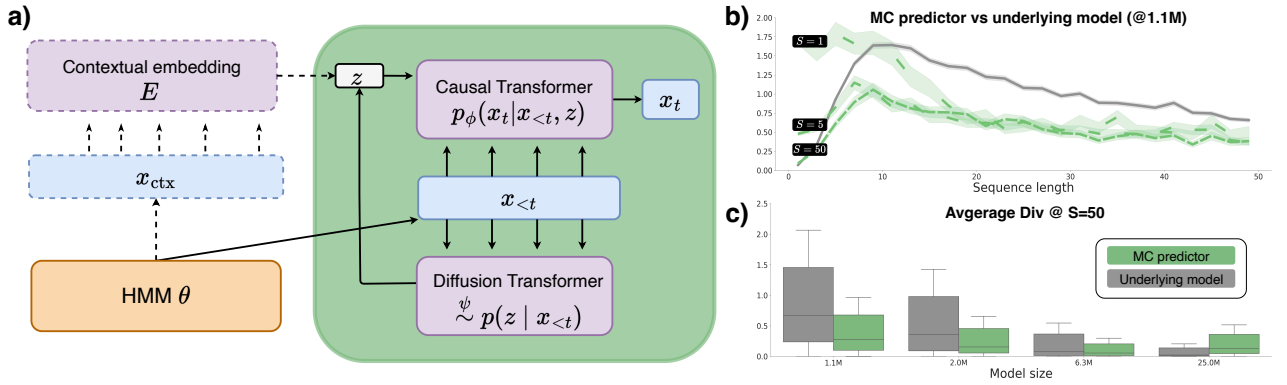

Figure 2: **The Monte-Carlo Predictor**. **a)** Computational and training structure of the MC predictor. The parts in dotted lines are only used to train $p_\phi(x_t \mid x_{<t}, z)$, while the content of the green rectangle corresponds to the MC predictor. **b)** Performance of the MC predictor (green) with various number of samples compared with the original sequence model (gray, 1.1M). Y-axis is Div. **c)** Average $\text{Div}(t)$ (across sequence length) of the MC predictor (green) with $S = 50$ compared to the underlying model (gray) for the different model sizes in Figure 1.

from $x_{<t}$ for $t \in [0, 50]$. Note that we deliberately used a larger diffusion than necessary to ensure that capacity of the task inference machinery was not the bottleneck. Studying the total parameter efficiency of our method (i.e. including the diffusion module) is left for future work.

Improved performance for small models

For small model sizes (e.g. 1.1M), we observe a clear improvement of the Monte-Carlo predictor over $p_{\phi_0}(x_t \mid x_{<t})$, see Figure 2**b)**. Further, as expected, divergence to oracle monotonically decreases with additional MC samples, demonstrating the test-time scaling potential of our approach. We also confirm the importance of having long sequences x_{ctx} for our latent proxy z in Figure 5. Observe that the Div of the MC predictor for this 1.1M model with $S = 5$ is similar to the Div of the 6.3M model in Figure 1**c)**.

Diminishing returns with scale

However, as model size increases, our method has diminishing returns Figure 2**c)**: for the biggest model, the MC predictor underperforms the traditional model for all number of samples. We attribute this discrepancy to multiple possible causes. On one hand, as the model size increases, its divergence to the Bayesian oracle at short context decreases, which sets the bar higher for the MC estimate. This reflects the insight from The Bitter Lesson (Sutton, 2019): as model capacity increases, architectural priors matter less. Our approach is thus most applicable in settings where the underlying sequence model underfits the Bayesian oracle—likely the case for foundation models faced with the complexities of natural language. In such scenarios, our method offers a mechanism for enhancing performance in high-ambiguity regimes without increasing compute for easier parts of the sequence. At the same time, it is also plausible that engineering issues are at play, which we discuss in Appendix B.

3. Discussion

Using our synthetic benchmark, MetaHMM, we have demonstrated that Transformers fail to approximate the Bayesian posterior predictive in high-ambiguity regimes. As noted, a similar issue is faced by humans, who address it with adaptive behavior (Gigerenzer & Goldstein, 1996). This highlights a key limitation of current approaches to sequence modeling, where fixed compute budget is used regardless of contextual uncertainty. We argue that foundation models should be *ambiguity-sensitive*, adapting their inference effort to the difficulty of the prediction.

As a step in this direction, we proposed a modular method that bootstraps a standard autoregressive model into a two-stage predictor: a diffusion-based context sampler and a conditional transformer. This architecture enables test-time scalable approximate Bayesian inference through Monte Carlo sampling. Our experiments show improved performance under ambiguity, though further work is needed to improve efficiency and decide if it can be applied to larger models.

More broadly, our contribution is to clearly identify a structural problem—handling epistemic uncertainty—and provide a foundation for future solutions. Beyond scalable inference, alternative directions include learned heuristics tailored to ambiguous contexts, and mechanisms for information-seeking behavior. While our framework does not support explicit actions, recent trends in RL-finetuned models (OpenAI, 2024; Guo et al., 2025) may be implicitly addressing ambiguity through learned clarification or retrieval behaviors. We hope that this work stimulates a comprehensive integration of past and current research related to the identified ambiguity problem; towards foundation models which go beyond Bayes-optimality (Grau-Moya et al., 2022).

References

Akyürek, E., Schuurmans, D., Andreas, J., Ma, T., and Zhou, D. What learning algorithm is in-context learning? investigations with linear models. *arXiv preprint arXiv:2211.15661*, 2022.

Akyürek, E., Wang, B., Kim, Y., and Andreas, J. In-context language learning: Architectures and algorithms. *arXiv preprint arXiv:2401.12973*, 2024.

Binz, M., Gershman, S. J., Schulz, E., and Endres, D. Heuristics from bounded meta-learned inference. *Psychological review*, 129(5):1042, 2022.

Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M. S., Bohg, J., Bosselut, A., Brunskill, E., et al. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021.

Bradbury, J., Frostig, R., Hawkins, P., Johnson, M. J., Leary, C., Maclaurin, D., Necula, G., Paszke, A., VanderPlas, J., Wanderman-Milne, S., et al. Jax: composable transformations of python+ numpy programs. 2018.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33: 1877–1901, 2020.

Chan, S., Santoro, A., Lampinen, A., Wang, J., Singh, A., Richemond, P., McClelland, J., and Hill, F. Data distributional properties drive emergent in-context learning in transformers. *Advances in neural information processing systems*, 35:18878–18891, 2022.

Chen, T., Zhang, R., and Hinton, G. Analog bits: Generating discrete data using diffusion models with self-conditioning. *arXiv preprint arXiv:2208.04202*, 2022.

Friston, K., FitzGerald, T., Rigoli, F., Schwartenbeck, P., and Pezzulo, G. Active inference: a process theory. *Neural computation*, 29(1):1–49, 2017.

Gigerenzer, G. and Goldstein, D. G. Reasoning the fast and frugal way: models of bounded rationality. *Psychological review*, 103(4):650, 1996.

Gilboa, I., Postlewaite, A., and Schmeidler, D. Is it always rational to satisfy savage's axioms? *Economics & Philosophy*, 25(3):285–296, 2009.

Gottlieb, J. and Oudeyer, P.-Y. Towards a neuroscience of active sampling and curiosity. *Nature Reviews Neuroscience*, 19(12):758–770, 2018.

Grau-Moya, J., Delétang, G., Kunesch, M., Genewein, T., Catt, E., Li, K., Ruoss, A., Cundy, C., Veness, J., Wang, J., et al. Beyond bayes-optimality: meta-learning what you know you don't know. *arXiv preprint arXiv:2209.15618*, 2022.

Grau-Moya, J., Genewein, T., Hutter, M., Orseau, L., Delétang, G., Catt, E., Ruoss, A., Wenliang, L. K., Mattern, C., Aitchison, M., et al. Learning universal predictors. *arXiv preprint arXiv:2401.14953*, 2024.

Graves, A. Adaptive computation time for recurrent neural networks. *arXiv preprint arXiv:1603.08983*, 2016.

Guo, D., Yang, D., Zhang, H., Song, J., Zhang, R., Xu, R., Zhu, Q., Ma, S., Wang, P., Bi, X., et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Hahn, M. and Goyal, N. A theory of emergent in-context learning as implicit structure induction, 2023. URL https://arxiv.org/abs/2303.07971.

Hendel, R., Geva, M., and Globerson, A. In-context learning creates task vectors. *arXiv preprint arXiv:2310.15916*, 2023.

Ho, J. and Salimans, T. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

Izacard, G. and Grave, E. Distilling knowledge from reader to retriever for question answering. *arXiv preprint arXiv:2012.04584*, 2020.

Keluskar, A., Bhattacharjee, A., and Liu, H. Do llms understand ambiguity in text? a case study in open-world question answering. In *2024 IEEE International Conference on Big Data (BigData)*, pp. 7485–7490. IEEE, 2024.

Kobayashi, G., Kuribayashi, T., Yokoi, S., and Inui, K. Transformer language models handle word frequency in prediction head. *arXiv preprint arXiv:2305.18294*, 2023.

Kojima, T., Gu, S. S., Reid, M., Matsuo, Y., and Iwasawa, Y. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35: 22199–22213, 2022.

Lampinen, A. K., Chan, S. C., Singh, A. K., and Shanahan, M. The broader spectrum of in-context learning. *arXiv preprint arXiv:2412.03782*, 2024.

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.-t., Rocktäschel, T., et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33:9459–9474, 2020.

Li, L., Zhang, H., Zhang, X., Zhu, S., Yu, Y., Zhao, J., and Heng, P.-A. Towards an information theoretic framework of context-based offline meta-reinforcement learning. *arXiv preprint arXiv:2402.02429*, 2024.

Lieder, F. and Griffiths, T. L. Resource-rational analysis: Understanding human cognition as the optimal use of limited computational resources. *Behavioral and brain sciences*, 43:e1, 2020.

Linderman, S. W., Chang, P., Harper-Donnelly, G., Kara, A., Li, X., Duran-Martin, G., and Murphy, K. Dynamax: A python package for probabilistic state space modeling with jax. *Journal of Open Source Software*, 10(108):7069, 2025.

Liu, A., Wu, Z., Michael, J., Suhr, A., West, P., Koller, A., Swayamdipta, S., Smith, N. A., and Choi, Y. We're afraid language models aren't modeling ambiguity. *arXiv preprint arXiv:2304.14399*, 2023.

Lovelace, J., Kishore, V., Wan, C., Shekhtman, E., and Weinberger, K. Q. Latent diffusion for language generation. *Advances in Neural Information Processing Systems*, 36: 56998–57025, 2023.

Lovelace, J., Kishore, V., Chen, Y., and Weinberger, K. Q. Diffusion guided language modeling. *arXiv preprint arXiv:2408.04220*, 2024.

Millidge, B., Seth, A., and Buckley, C. Understanding the origin of information-seeking exploration in probabilistic objectives for control. *arXiv preprint arXiv:2103.06859*, 2021.

Mittal, S., Elmoznino, E., Gagnon, L., Bhardwaj, S., Sridhar, D., and Lajoie, G. Does learning the right latent variables necessarily improve in-context learning? *arXiv preprint arXiv:2405.19162*, 2024.

Nakano, R., Hilton, J., Balaji, S., Wu, J., Ouyang, L., Kim, C., Hesse, C., Jain, S., Kosaraju, V., Saunders, W., et al. Webgpt: Browser-assisted question-answering with human feedback. *arXiv preprint arXiv:2112.09332*, 2021.

Niwa, A. and Iso, H. Ambignlg: Addressing task ambiguity in instruction for nlg. *arXiv preprint arXiv:2402.17717*, 2024.

Oord, A. v. d., Li, Y., and Vinyals, O. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

OpenAI. Openai o1 system card, 2024. URL https://arxiv.org/abs/2412.16720.

Ortega, P. A., Wang, J. X., Rowland, M., Genewein, T., Kurth-Nelson, Z., Pascanu, R., Heess, N., Veness, J., Pritzel, A., Sprechmann, P., et al. Meta-learning of sequential strategies. *arXiv preprint arXiv:1905.03030*, 2019.

Oudeyer, P.-Y. and Kaplan, F. What is intrinsic motivation? a typology of computational approaches. *Frontiers in neurorobotics*, 1:108, 2007.

Panwar, M., Ahuja, K., and Goyal, N. In-context learning through the bayesian prism, 2024. URL https://arxiv.org/abs/2306.04891.

Peebles, W. and Xie, S. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4195–4205, 2023.

Rabiner, L. and Juang, B. An introduction to hidden markov models. *ieee assp magazine*, 3(1):4–16, 1986.

Rizvi-Martel, M., Lizaire, M., Lacroce, C., and Rabusseau, G. Simulating weighted automata over sequences and trees with transformers. In *International Conference on Artificial Intelligence and Statistics*, pp. 2368–2376. PMLR, 2024.

Robert, C. P., Casella, G., and Casella, G. *Monte Carlo statistical methods*, volume 2. Springer, 1999.

Salimans, T. and Ho, J. Progressive distillation for fast sampling of diffusion models. *arXiv preprint arXiv:2202.00512*, 2022.

Sanborn, A. N., Griffiths, T. L., and Navarro, D. J. Rational approximations to rational models: Alternative algorithms for category learning. *Psychological Review*, 117(4):1144–1167, 2010. doi: 10.1037/a0020511.

Schick, T., Dwivedi-Yu, J., Dessì, R., Raileanu, R., Lomeli, M., Hambro, E., Zettlemoyer, L., Cancedda, N., and Scialom, T. Toolformer: Language models can teach themselves to use tools. *Advances in Neural Information Processing Systems*, 36:68539–68551, 2023.

Snell, C., Lee, J., Xu, K., and Kumar, A. Scaling llm test-time compute optimally can be more effective than scaling model parameters. *arXiv preprint arXiv:2408.03314*, 2024.

Sutton, R. The bitter lesson. *Incomplete Ideas (blog)*, 13(1): 38, 2019.

Todd, E., Li, M. L., Sharma, A. S., Mueller, A., Wallace, B. C., and Bau, D. Function vectors in large language models. *arXiv preprint arXiv:2310.15213*, 2023.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Von Oswald, J., Niklasson, E., Randazzo, E., Sacramento, J., Mordvintsev, A., Zhmoginov, A., and Vladymyrov, M. Transformers learn in-context by gradient descent. In *International Conference on Machine Learning*, pp. 35151–35174. PMLR, 2023.

Wang, X., Zhu, W., Saxon, M., Steyvers, M., and Wang, W. Y. Large language models are latent variable models: Explaining and finding good demonstrations for incontext learning. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=BGvkwZEGt7.

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Xia, F., Chi, E., Le, Q. V., Zhou, D., et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

Xie, S. M., Raghunathan, A., Liang, P., and Ma, T. An explanation of in-context learning as implicit bayesian inference. *arXiv preprint arXiv:2111.02080*, 2021.

Yang, L., Lin, Z., Lee, K., Papailiopoulos, D., and Nowak, R. Task vectors in in-context learning: Emergence, formation, and benefit. *arXiv preprint arXiv:2501.09240*, 2025.

Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K., and Cao, Y. React: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations (ICLR)*, 2023.

Zhang, M. J. and Choi, E. Clarify when necessary: Resolving ambiguity through interaction with lms. *arXiv preprint arXiv:2311.09469*, 2023.

Zhu, W. Leebert: Learned early exit for bert with cross-level optimization. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 2968–2980, 2021.

Zhuang, Y., Singh, C., Liu, L., Shang, J., and Gao, J. Vector-icl: In-context learning with continuous vector representations. *arXiv preprint arXiv:2410.05629*, 2024.

A. Extented related work

A.1. Sequence Meta-Learning and In-Context Learning

The link between in-context learning (ICL) and meta-learning was established early on (Brown et al., 2020) and has become central to understanding foundation model behavior. Xie et al. (2021) were among the first to formalize this connection by framing foundation model pre-training as meta-learning over a distribution of tasks. This perspective has since inspired a wide range of studies (Von Oswald et al., 2023; Hahn & Goyal, 2023; Chan et al., 2022; Akyürek et al., 2022; 2024; Grau-Moya et al., 2024), most of which focus on few-shot ICL (Lampinen et al., 2024), where the model infers a classification function in-context: $x_0, y_0, \ldots, x_q \to y_q$.

Our work instead focuses on sequence meta-learning, a more general—and we argue more accurate—framing of what foundation models do : model sequences. In this view, each sequence is generated by a latent task θ, and the model must perform implicit inference over θ to make accurate predictions.

Several synthetic distributions have been proposed for studying sequence meta-learning, but, as shown in Figure 1, none satisfy all the properties we require. GINC (Xie et al., 2021) uses sequences drawn from a mixture of HMMs but lacks compositional structure and evaluates models mainly through predictive accuracy. RegBench (Akyürek et al., 2024) also evaluates models on probabilistic sequence data and compares them to a Bayesian oracle, but focuses on architecture comparisons (e.g., RNNs vs. Transformers) rather than the ambiguity-computation mismatch we investigate. Other works study non-Markovian sequence distributions, such as those based on PCFGs (Hahn & Goyal, 2023) or Turing machines (Grau-Moya et al., 2024), but these lack a tractable Bayesian oracle, limiting their utility for quantitative evaluation.

A.2. Latent variables in Transformers

Multiple previous works have explored to what extent Transformers explicitly represent the latent variables underlying an in-context problem. (Todd et al., 2023; Hendel et al., 2023) have shown that in some few-show ICL tasks, such word associations, Transformer indeed represent the task latent θ, allowing for manipulation of the inference process. (Yang et al., 2025) have expanded on the conditions necessary for *task vectors* to appear, and how they often don't. Authors have also proposed a method to force the appearance of task vectors using an auxiliary loss. (Mittal et al., 2024) also demonstrated that task vectors sometimes do not appear in function approximation few-shot ICL tasks and attributed it to the fact that Transformers had trouble representing the functional form $p^*_\theta(x \mid y)$. Lastly, (Zhuang et al., 2024) have trained Transformers with continuous task vectors in order to increase the performance on some ICL tasks.

A.3. Ambiguity and the Limits of Bayesian Inference

The challenge of inference under ambiguity has long been studied in cognitive science and decision theory. While Bayesian inference offers a normative ideal, exact inference becomes intractable—or even behaviorally irrational—when the posterior is broad (Gigerenzer & Goldstein, 1996; Gilboa et al., 2009; Lieder & Griffiths, 2020).

To account for how humans reason effectively despite such limitations, resource-rational analysis posits that people optimize a trade-off between accuracy and cognitive cost (Lieder & Griffiths, 2020). Rather than marginalizing over all hypotheses, humans rely on heuristics or approximate inference mechanisms—often learned through experience—that deliver fast, "good enough" solutions (Sanborn et al., 2010; Binz et al., 2022).

Another human strategy is information-seeking, where agents act to reduce uncertainty before committing to a belief or decision. This is formalized in active inference (Friston et al., 2017), in which agents take epistemic actions—e.g., querying, exploring, or deferring—to gather evidence and improve predictions. Related ideas appear in reinforcement learning under the banner of curiosity and intrinsic motivation (Oudeyer & Kaplan, 2007; Gottlieb & Oudeyer, 2018).

These insights motivate our central hypothesis: when the posterior over latent tasks $p(\theta \mid x_{<t})$ is broad, prediction becomes not only statistically harder, but also computationally more demanding. While humans respond flexibly through heuristics and exploration, transformers by default apply a fixed computation budget to all inputs.

A.4. Existing solutions

Although rarely framed in terms of latent task inference, many approaches in the foundation model literature implicitly address ambiguity. One prominent strategy is retrieval-augmented generation (RAG) (Lewis et al., 2020; Izacard & Grave, 2020), which enriches the input context with relevant documents, enabling models to disambiguate queries using external

knowledge. RAG is especially helpful in settings with underspecified or ambiguous inputs.

Other approaches equip models with clarification-seeking capabilities (Zhang & Choi, 2023), allowing them to request more information before answering. Similarly, tool-augmented models (Schick et al., 2023; Yao et al., 2023) can interact with APIs or calculators, actively reducing uncertainty—akin to epistemic actions in humans. Further, system prompts, such as those used in chatbots, serve to disambiguate the model's role and task (Niwa & Iso, 2024).

As a side note, chain-of-thought prompting (Wei et al., 2022; Kojima et al., 2022) methods improve reasoning by encouraging intermediate steps. However, theses methods enhance can be seen as addressing cases where the conditional prediction $p(x_t \mid x_{<t}, \theta)$, not the task inference, is too difficult for the model.

Finally, reinforcement learning from human feedback (RLHF) can lead to emergent ambiguity-sensitive behaviors (OpenAI, 2024; Guo et al., 2025). Models fine-tuned with RLHF sometimes defer responses, ask clarifying questions, or invoke tools—particularly when direct answers are penalized for inaccuracy (Nakano et al., 2021). These behaviors align with active inference (Friston et al., 2017; Millidge et al., 2021), though ambiguity resolution is not explicitly optimized.

A.5. Test-Time Scaling and Adaptive Inference

Our method also contributes to the growing literature on test-time scalable inference, where computation is dynamically adjusted based on input difficulty. Early work on adaptive computation time (ACT) (Graves, 2016) allowed models to learn how many steps to take. In transformers, this has evolved into early exiting mechanisms (Zhu, 2021), which conditionally terminate processing. See Snell et al. (2024) for a modern discussion of test-time scaling. Our approach offers a specific, and principled interpretation of test-time scaling from a Bayesian perspective.

B. Additional details

B.1. Engineering challenges of the Monte-Carlo predictor

As model size increases, the latent representation z also grows in dimensionality, potentially making the diffusion sampling task more difficult. Despite the DiT's large capacity, it may still underfit. Improved diffusion design—e.g., via classifier-free guidance (Ho & Salimans, 2022) or self-conditioning (Chen et al., 2022)—could help bridge this gap. Additionally, the embedding z may encode high- frequency details from x_{ctx} that are hard to sample accurately. This issue could be exacerbated by increasing the dimension of x. A promising direction would be to regularize z to capture only information relevant to task identity θ. For example, maximizing the mutual information $I(z;\theta)$ via a contrastive loss (Oord et al., 2018; Li et al., 2024) could encourage more robust low-dimensional. We leave a full exploration of these directions to future work, viewing our method as a first step toward separating task inference and prediction in foundation models.

B.2. MetaHMM details

All experiments in this paper are performed on three seeds of a MetaHMM with a fixed size. HMMs have a hidden state space of dimension 20 and an observational space of size 50. Other hyperparamters are described in Figure 3a). The total number of HMMs can be computed as

$$\underbrace{(n_b \cdot s_b \cdot d_b)}_{\text{Base cycles}} \cdot \underbrace{(g_f^{n_f} \cdot d_f \cdot s_f)}_{\text{Cycle families}} \cdot \underbrace{(\alpha_e^{g_e} \cdot \beta_e)}_{\text{Emission groups}} \tag{4}$$

which in our case gives 12,288 HMMs.

B.3. Architecture details

All causal transformers use the Adam optimizer with learning rate 0.001, batch size 256 and $50,000$ updates. Further hyperparameters of the causal Transformers are given by Figure 3b). When training on a MetaHMM environment, we hold out 1000 HMMs ($\sim \frac{1}{12}$ of all) for validation, and report validation metrics throughout.

All diffusion models are DiTs (Peebles & Xie, 2023) trained with the Adam optimizer with learning rate 0.0001, batch size 512 and $100,000$ updates. The conditioning information, i.e. $x_{<t}$, is first passed through an Transformer encoder (without causal masking) and then both through a cross-attention block and a adaLN-Zero block. Hyperparameters are given by Figure 3c). When training, we hold out $\frac{1}{10}$th of all HMMs for validation. The DiT uses the velocity parameterization (Salimans & Ho, 2022) with an $L2$ loss and a cosine noise schedule. Sampling is performed using the DDPM sampler with

Base cycles	
Cycles n_b	4
Step-size s_b	2
Directions d_b	2
Cycle families	
Families n_f	3
Groups per family g_f	2
Directions d_f	2
Step-sizes s_f	2
Emission matrix	
Groups g_e	3
Emission per group α_e	2
Shifts β_e	3

(a) **MetaHMM**

Size/Hyperparmeters	Layers	Heads	Dimension
1.1M	4	4	128
2.0M	6	6	128
6.3M	6	8	256
25.0M	8	8	512

(b) **Causal Transformer**

DiT Layers	8
DiT Heads	8
Dimension	512
Enc. Layers	8
Enc. Heads	8

(c) **DiT**

Figure 3: **Hyperparameters**

50 timesteps. Other hyperparameters are the same as in (Lovelace et al., 2023).

B.4. Figures

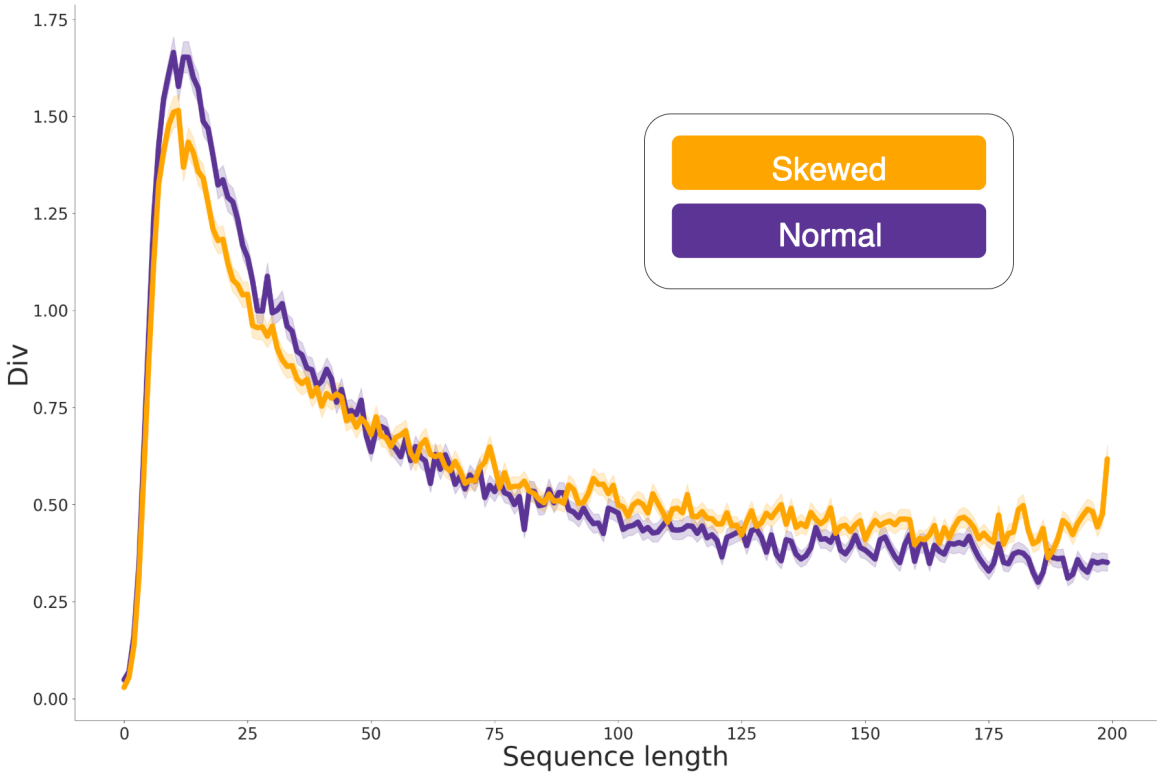

Figure 4: **Transformer trained on random sequence lengths** Variation of the next-token training setup of Figure 1c) where sequences have random lengths uniformly sampled between 1 and 200. This puts more pressure on the predictors to perform well at low context lengths. Batch size are adjusted so that the amount of tokens per batch remains constant (and equal to full-length training).

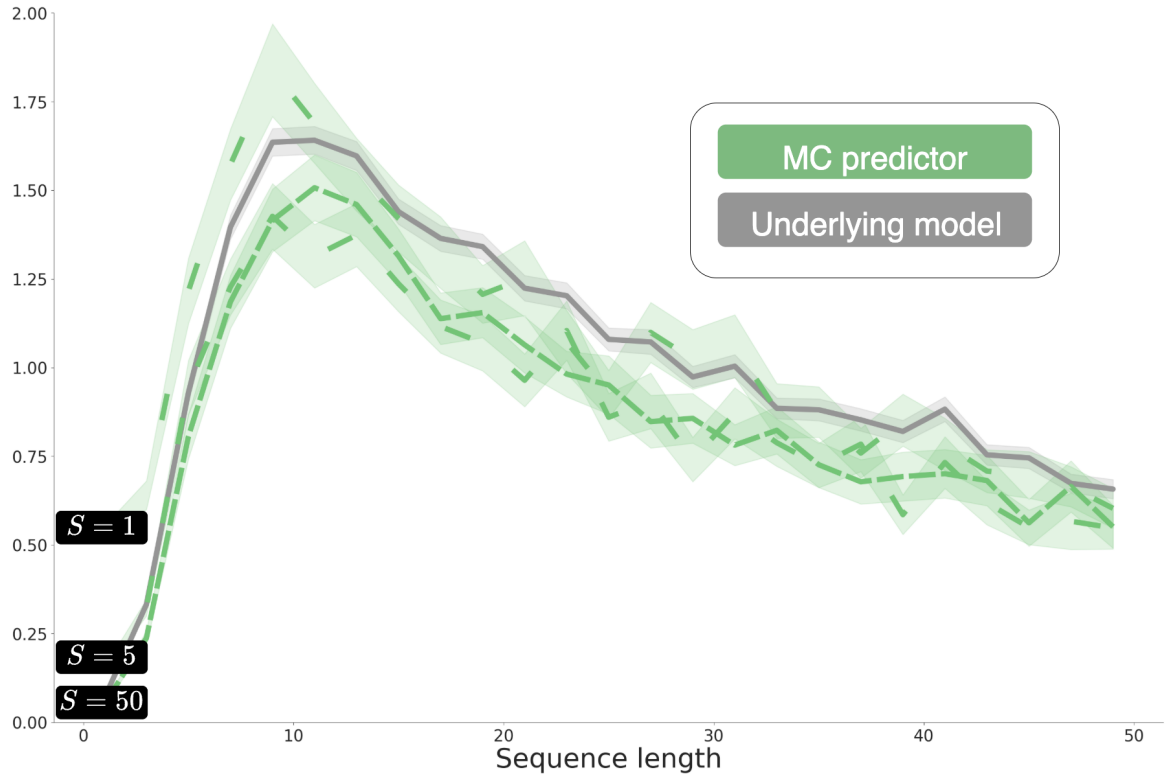

Figure 5: **Short** x_{ctx} variation of Figure 2b) where x_{ctx} has length 10. This means that z should be a poor proxy for θ and the MC predictor should do poorly.