
Research Diary

PhD Research Journal

Author: **Annabel Jakob**

Start Date: 11 February 2026

Field: **Computer Science**



Contents

1	Resources	3
2	04 February 2026	4
2.1	Research Plan	4
2.2	Content Details	4
3	09 February 2026	5
4	10th February 2026	6
5	11th February 2026	7

1 Resources

- DPhil Progression Information and Resources

2 04 February 2026

2.1 Research Plan

Today's main tasks:

- Meeting with Seth and Gunes
- Familiarise with DPhil milestones and progression requirements
- Reading relevant literature suggested during supervisor meeting

2.2 Content Details

Meeting Summary

Meeting Notes:

- DPhil Milestones
 - Term 4, Week 0: Transfer status
 - 8th/9th Term: Confirmation of DPhil status
- Finding a research question
 - Bayesian Transformers
 - * The Bayesian Geometry of Transformer Attention
 - * Transformers Can Do Bayesian Inference
 - AI-assisted proofs
 - * Existing tools: LEAN Proof Checker and [Xena Project](#), Autodiff
 - * Possible steps:
 1. Compile dataset of theorems presented in previous conference papers (e.g. NeurIPS)
 2. Verify the theorems and proofs in the dataset
 3. Goal: Can we produce *new* theorems and proofs?
 - * Other resources:
 - https://en.wikipedia.org/wiki/Kevin_Buzzard
 - <https://terrytao.wordpress.com/2025/12/08/the-story-of-erdos-problem-126/>
 - Aristotle: IMO-level Automated Theorem Proving
- Mechanistic interpretability

- Potentially connected to encrypted backdoors
- Talk to Marek and Junayed
- Statistical Machine Learning
 - DeepRV: Accelerating spatiotemporal inference with pre-trained neural priors
- Random Number Generators
 - Can a backdoor be hidden in a RNG/ induced through an RNG? I.e. malicious signal induced via carefully chosen "random" numbers?
 - Planting Undetectable Backdoors in Machine Learning Models
 - Oblivious Defense in ML Models: Backdoor Removal without Detection

3 09 February 2026

Today's main tasks:

- Read Bayesian Transformers papers:
 - The Bayesian Geometry of Transformer Attention
 - Transformers Can Do Bayesian Inference

█ Paper Reading

Paper title: Transformers Can Do Bayesian Inference [1]

Authors: Mueller et al.

Summary: Bayesian methods are usually slow or mathematically intractable for large datasets. Deep learning is fast but often bad at uncertainty and priors. As a solution, the authors present Prior-Data Fitted Networks (PFNs), which learn the mapping of the data to Bayesian posterior prediction.

Key Contributions

- PFNs (transformers that learn to approximate Bayesian posterior predictive distributions); this includes architectural changes to the standard transformer architecture, such as the Riemann distribution (a discretized output distribution for regression) and removal of positional encodings for permutation invariance over dataset examples
- Demonstrate that PFNs can approximate the PPD of Gaussian processes and Bayesian neural networks (BNNs) + are much faster than standard methods for approximating Bayesian inference (e.g. MCMC, variational inference)

- A BNN prior over architectures outperforms XGBoost/CatBoost on small tabular datasets while being $\sim 5000\times$ faster, with strong calibration (low expected calibration error)
- Strong few-shot learning on Omniglot using a simple stroke-based prior

Notes

- Focuses on Bayesian posterior prediction for supervised learning problems, particularly small-data regimes (e.g. 30 training samples in tabular experiments)
- PFNs: train transformers to approximate Bayesian posterior predictive distributions by sampling synthetic datasets from a prior, masking labels, and learning to predict them. At inference time, a single forward pass approximates Bayesian inference.
- PFNs work for any prior distribution that can be sampled from → this is a very relaxed requirement compared to the standard assumptions of other Bayesian inference approximations. E.g. MCMC methods like NUTS [2] assume access to non-normalised posterior density, ability to evaluate the likelihood $p(D|\theta)$ and prior $p(\theta)$ pointwise, and gradients to the log-posterior.
- Deep learning models encode inductive bias through e.g. architecture, training data, and training procedure. PFNs let you inject an explicit prior by defining how to sample synthetic tasks.
- Limitation: PFN must be retrained for each new prior

Thoughts

- PFNs can effectively mimic Gaussian processes, but what about non-Gaussian processes (especially compared to MCMC and VI)?
 - The paper does test beyond GPs—the BNN experiments involve highly non-Gaussian, multimodal posteriors. PFNs handle these well precisely because they only need to sample from the prior, not evaluate densities. The limitation is more about whether you can design a prior that matches your real-world problem.

4 10th February 2026

💡 Daily Summary

Today's main tasks:

- Meeting with Seth

Meeting Summary

Meeting Notes:

5 11th February 2026

💡 Daily Summary

Today's main tasks:

- Read Bayesian Geometry of Transformer Attention [3]

📘 Paper Reading

Paper title: The Bayesian Geometry of Transformer Attention [3]

Authors: Agarwal et al.

Summary: Uses *Bayesian wind tunnels* (controlled prediction tasks with closed-form posteriors where the hypothesis space is too large for memorisation and in-context prediction requires genuine probabilistic inference) to empirically prove transformers perform exact Bayesian inference and understand how.

Key Contributions

- Transformers match analytic Bayesian posteriors with $10^{-3} - 10^{-4}$ bit accuracy on bijection learning and HMM filtering.
- Decompose Bayesian Learning into three primitives:
 - *Belief accumulation* integrating evidence into the posterior as observations arrive
 - *Belief transport* propagating beliefs forward through stochastic dynamics; i.e. transforming the belief vector according to the system dynamics.
 - *Random-access binding* Retrieving stored hypotheses by content rather than position.
- Architecture comparison: Transformers realise all three; Mamba handles accumulation/transport but struggles with binding; LSTMs only handle accumulation; MLPs fail entirely

Notes

- The authors used the different wind tunnels to identify which inference primitive the model realised. Belief accumulation was tested by bijection

elimination, belief transport was tested by HMM filtering, and random-access binding was tested by associative recall.

- How do transformers realise all three primitives?
 - The transformer layers can be broadly mapped to different inference steps:
 1. **Layer 0 - Set up representational infrastructure:** Build orthogonal frame (define hypothesis space). Keys at this layer form an approximately orthogonal basis over input tokens. Thus, each hypothesis occupies its own independent direction.
 2. **Middle layers - Perform actual computation:** Sharpen Q-K alignment (eliminate inconsistent hypotheses)
 3. **Late layers - Improve numerical accuracy:** Refine value manifold (encode posterior entropy precisely)

Thoughts

- The authors present three inference primitive, but they do not prove that these are *exhaustive*. Different tasks may reveal additional primitives.

References

- [1] Samuel Müller et al. “TRANSFORMERS CAN DO BAYESIAN INFERENCE”. In: *ICLR 2022 - 10th International Conference on Learning Representations*. 2022.
- [2] Matthew D. Hoffman and Andrew Gelman. “The no-U-turn sampler: Adaptively setting path lengths in Hamiltonian Monte Carlo”. In: *Journal of Machine Learning Research* 15 (2014). ISSN: 15337928.
- [3] Naman Agarwal and Siddhartha R Dalal. “The Bayesian Geometry of Transformer Attention Paper I of the Bayesian Attention Trilogy”. In: 1 (2026). URL: <https://arxiv.org/abs/2512.22471>.