The SWIFT Model of Eye Movements

— Project Report —
Simulation Based Inference

Abdul Muqsit Farooqi, 230635

Muhammad Areeb, 236888

Ahsan Bin Qasim, 236889

August 17, 2025

TU Dortmund University

Contents

1 Introduction

To better understand how people read, it's important to see how the settings of cognitive models show up in real eye movement patterns. This connection helps us test and improve reading theories. Models such as SWIFT offer mechanistic accounts of fixation durations, saccade directions, and word-level processing in natural reading. However, estimating these parameters directly from experimental data is difficult, as the likelihood function is often unknown or too computationally expensive to calculate. This work tackles that challenge by employing a Simulation-Based Inference (SBI) framework, implemented through `BayesFlow`, to estimate parameters of a SWIFT-inspired reading model using both simulated and empirical eye fixation data. The primary objective is to recover posterior distributions for the parameters governing attention allocation, lexical activation, and fixation timing, relying solely on the observable patterns of fixations as input.

This project applies a Simulation-Based Inference framework, implemented with `BayesFlow`, to infer parameters of a SWIFT-inspired eye movement model from simulated fixation data. By combining principled Bayesian methods with neural inference networks, from summary features of fixation sequences the approach aims to recover posterior distributions over key cognitive parameters. The workflow of the project is the following:

- Defined priors for attention spread, controlling the spatial extent of attentional processing (ν), maximum activation, determining peak lexical activation levels (r), and mean fixation duration, setting the temporal scale of word processing (μ_T) to capture plausible cognitive variability.
- Simulated fixation sequences using a custom saliency-driven model with gamma-distributed durations.
- Extracted eight descriptive summary features from each sequence.
- Trained on summary network to learn embeddings, enabling the comparison with handcrafted features.
- Used BayesFlow CouplingFlow network to train on 10,000 simulated (parameter, summary) pairs.
- Obtain posterior samples and estimates using real fixation data.
- Assessed calibration with SBC and compared saliency, non-saliency, and random generative models.

The report is structured as follows: Chapter 2 defines the objectives of the project and details the data material. Chapter 3 details statistical models. Chapter 4 presents the inference results for both the manual & bayesflow network summary, and model comparison results. Chapter 5 concludes the results of all the models.

2 Description of the Problem

It is essential for testing and refining theories of reading to Understand how cognitive model parameters relate to observable eye movement patterns. Models such as SWIFT provide explanations for fixation durations, saccade directions, and word-level processing during natural reading. However, directly estimating the parameters from experimental data is challenging, as the likelihood function is typically unknown.

To addresses this challenge, a Simulation-Based Inference (SBI) approach is applied with BayesFlow to estimate parameters of a SWIFT model from observed eye and simulated fixation data. The goal of this project is to retrieve posterior distributions over key parameters controlling attention allocation, lexical activation, and fixation timing, using only observable fixation patterns (work index and fixation duration) as input.

2.1 The Objectives of the Project

The aim of this project is to use Simulation-Based Inference (SBI) to infer cognitive model parameters from eye-tracking data and evaluate these estimates. The central questions are:

- Can the parameters ν (attention spread), r (activation), and μ_T (fixation duration) be reliably estimated from fixation sequences?
- How does the performance differ when using manually generated summary statistics versus BayesBlow summary network?
- How do posterior distributions change across different generative model variants (with saliency, no saliency, random)?

The project involves several steps that are defining informative priors, simulating fixation sequences from a SWIFT-inspired generative model, extracting summary statistics, training a neural density estimator with BayesFlow, and validating (model daignostics) the inferred posteriors using Simulation-Based Calibration (SBC).

2.2 The Data Material

In this project, data is based on trial-level eye movement records simulated from the generative model. Each row corresponds to a single fixation event and contains the variables given in Table 1.

Variable Name	Variable Type	Description
`Trial_ID`	Integer	Unique ID for each reading trial.
`Word_ID`	Integer	Position of the fixated word in the sentence or passage.
`Fixation_Start`	Numeric	Start time of the fixation (in milliseconds).
`Fixation_End`	Numeric	End time of the fixation (in milliseconds).
`Word_Frequency`	Numeric	Frequency of the fixated word in a reference corpus.
`Word_Length`	Integer	Number of characters in the fixated word.
`Saccade_Direction`	Categorical	Direction of the preceding saccade (e.g., forward, backward).

Table 1: Eye-Tracking Dataset Variable Descriptions

From these raw fields, a derived variable was computed:

$$\text{Fixation Duration} = \text{Fixation End} - \text{Fixation Start}$$

This measure represents the duration of each fixation in milliseconds and is a primary dependent variable for modeling.

For parameter inference, the analysis focused on `Word_ID` (as a positional index) and calculated `Fixation_duration`, as these directly relate to the model's temporal and spatial predictions. The simulated dataset contained 10,000 examples, split into training and validation sets for the SBI pipeline. The real fixation data (test data) were of 40 observations. Potential issues with the data might affect the project:

- **Noise in fixation timing** due to tracker precision limits,
- **Outlier fixations** Extremely short or long durations that may arise from recording errors or blinks,
- **Trial variability** in sentence length or word difficulty influenced fixation patterns.

The original dataset was used to train the BayesFlow inference network and to evaluate the model's ability to recover parameters from both synthetic and real fixation data (simulated data).

3 Statistical Models

This section describes the statistical modeling approaches used to infer cognitive parameters of fixation data. Two variants of the Simulation-Based Inference (Tejero-Cantero et al. (2020)) pipeline were compared:

1. **Model 1:** Manual (handcrafted) summary statistics as model input.
2. **Model 2:** Learned summary representations via a neural summary network in `BayesFlow`.

Both models shared the same generative simulator, priors, and inference network architecture (CouplingFlow), differing only in how fixation sequences were summarized prior to inference.

The packages used in this project for the `Bayesflow` are Radev et al. (2023b), Radev et al. (2023a),

The SWIFT-inspired simulation is governed by three key parameters (ν (attention spread, controlling the spatial extent of lexical activation), r (maximum activation, setting the peak activation level), and μ_T (mean fixation duration, determining the temporal scale of processing)), controlling spatial and temporal dynamics of fixations. Priors and generative processes are defined as follows:

Priors Model parameters are drawn from uniform distributions:

$$\begin{aligned} \nu &\sim \text{Uniform}(0.1,\ 0.9) \\ r &\sim \text{Uniform}(5.0,\ 15.0) \\ \mu_T &\sim \text{Uniform}(150.0,\ 150.0) \end{aligned}$$

Here, ν controls the spatial spread of activation, r determines the peak activation strength, and μ_T is the mean fixation duration.

Fixation Duration Generation Durations are sampled from a Gamma distribution ((Engbert and Rabe, 2024, p. 4)):

$$\text{duration} \sim \text{Gamma}(\alpha,\ \mu_T/\alpha)$$

The shape parameter α controls variability, while μ_T sets the mean.

Spatial Activation and Normalization The activation of word i is computed from its distance d_i to the current fixation ((Engbert and Rabe, 2024, p. 3)):

$$\begin{aligned} \text{activation}_i &= r \cdot \exp\left(-\frac{d_i^2}{2\nu^2}\right) \\ \text{norm}_i &= \frac{\text{activation}_i}{\max(\text{activation})} \end{aligned}$$

Saliency and Selection Probability Normalized activations are transformed into saliency scores ((Engbert and Rabe, 2024, p. 3)):

$$\text{saliency}_i = \begin{cases} \sin^2(\pi \cdot \text{norm}_i) + \eta, & \text{if norm}_i < 1 \\ \eta, & \text{otherwise} \end{cases}$$

$$p_i = \frac{\text{saliency}_i}{\sum_{j=1}^{N} \text{saliency}_j}$$

The η term ensures non-zero probabilities for all words.

Position Sampling The next fixation position is sampled from a categorical distribution:

$$\text{position}_{t+1} \sim \text{Categorical}(p_1, p_2, \ldots, p_N)$$

This process repeats for a fixed number of steps or until convergence. (Engbert and Rabe (2024))

3.1 Model 1: Manual Summary Statistics

Approximator: For Model 1, each simulated fixation sequence was transformed into a fixed set of eight manual summary statistics designed to capture key aspects of eye movement behavior, such as mean and skewness of fixation durations, and proportions of different saccade directions(Schwetlick et al. (2020)). These features were passed directly into the `BayesFlow` `CouplingFlow` inference network via an `Adapter` configured to map simulation outputs to model inputs. No summary network was used in this model, as the handcrafted features already provided fixed-size embeddings suitable for posterior estimation.

Training: The model was trained using *offline amortized Bayesian inference* (Radev et al. (2023b)). A dataset of 10,000 (θ, summary) pairs was pre-simulated from the SWIFT-inspired generative model (Engbert et al. (2005)). Training used the Adam optimizer with default `BayesFlow` settings, processing batches of simulations for multiple epochs until convergence in the training and validation loss curves was achieved. Since amortization was employed, the computationally expensive training phase was performed once, after which inference could be run efficiently for any new dataset without retraining.

Diagnostics: The computational faithfulness and model sensitivity of the approximator were evaluated using `BayesFlow's` diagnostic suite:

- **Posterior Predictive Check:** Figure 1 shows the posterior predictive overlay for fixation durations (Berkhof et al. (2000)). Blue lines indicate posterior predictive samples, the dashed orange line shows the posterior predictive mean, and the black line is the observed data. The predictions broadly match the overall trend of the observed sequence, though extreme peaks in the empirical data remain underestimated.

- **Simulation-Based Calibration (SBC):** ECDF difference plots (Figure 2) revealed that parameter ν exhibited strong tail deviations well outside the 95% confidence bands, r showed mild mid-range bias but remained largely within calibration limits, and μ_T displayed systematic underestimation in the upper rank spectrum.
- **Parameter Recovery:** Figures 10, 3, and 4 reports recovery performance: ν (RMSE = 0.2682, Corr = 0.1018) showed poor recovery, r (RMSE = 1.4680, Corr = 0.9528) was recovered with high accuracy, and μ_T (RMSE = 11.6340, Corr = 0.9860) achieved near-perfect recovery.

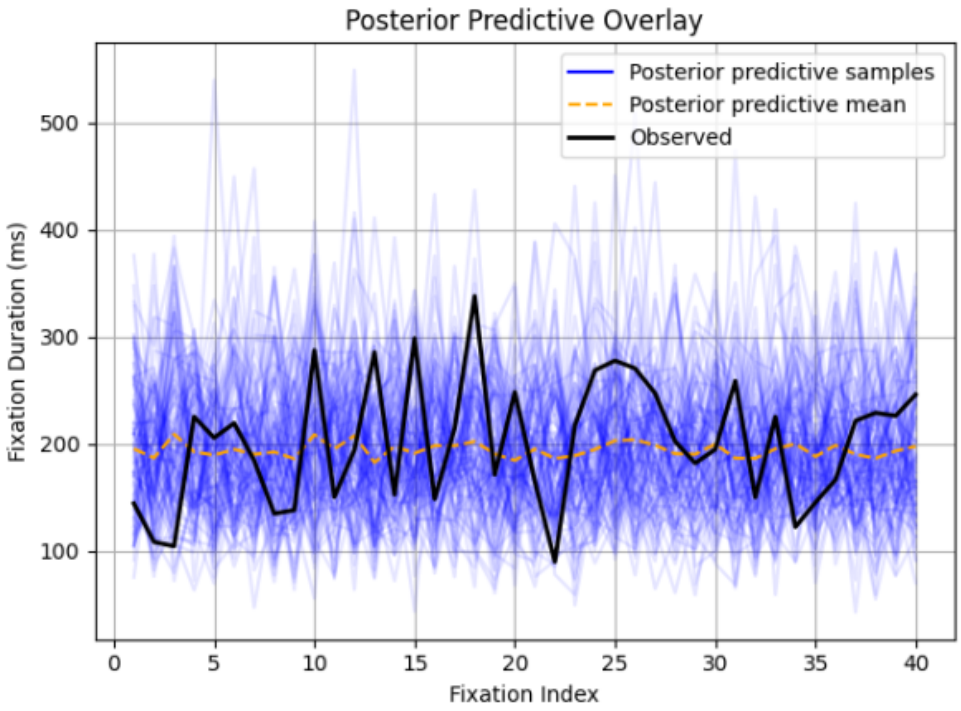

Figure 1: Posterior predictive overlay for Model 1.

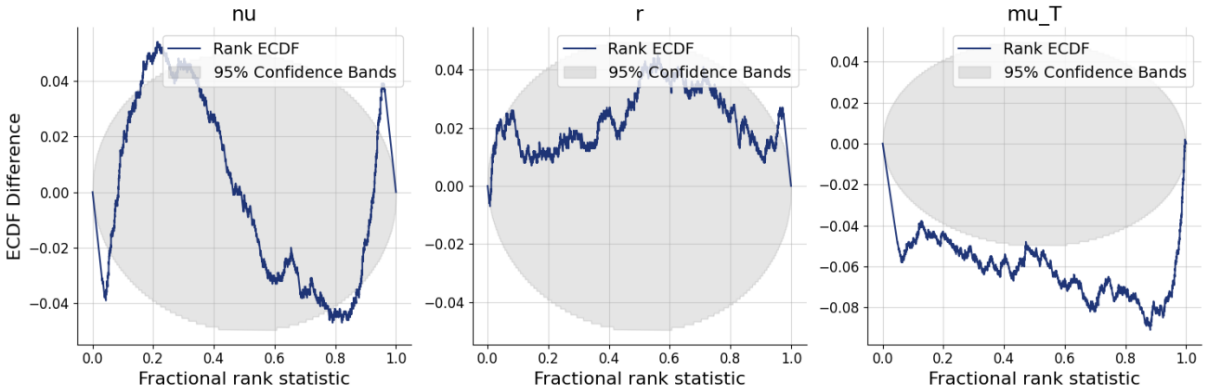

Figure 2: SBC ECDF difference plots for Model 1 parameters (ν, r, μ_T) with 95% confidence bands.

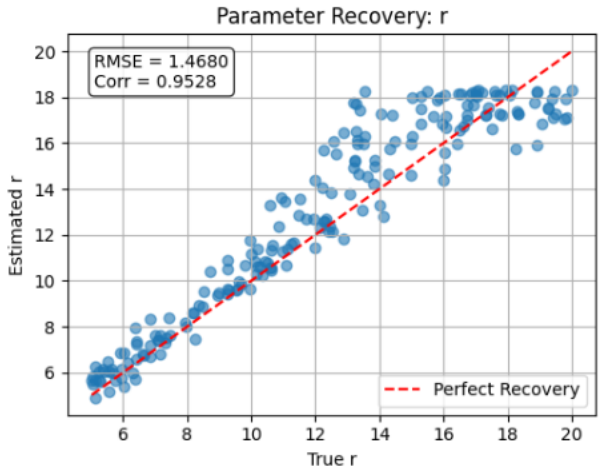

Figure 3: Parameter recovery plot for Model 1: r with RMSE and correlation values. The red dashed line denotes perfect recovery.

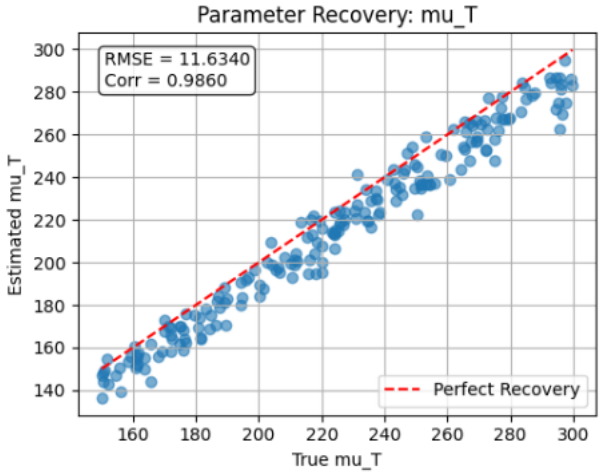

Figure 4: Parameter recovery plot for Model 1: μ_T with RMSE and correlation values. The red dashed line denotes perfect recovery.

Inference Once trained, the model was applied to observed fixation data. The handcrafted summaries of the observed sequence were passed through the trained approximator to generate posterior samples for ν, r, and μ_T. From these, posterior means and credible intervals were computed. The results confirm that handcrafted summaries are effective for well-identifiable parameters such as r and μ_T, but provide insufficient information for reliable inference on ν, which remains poorly recovered and systematically biased.

3.2 Model 2: BayesFlow-Based Summary Network

Approximator: For Model 2, the handcrafted summary statistics of Model 1 were replaced with a learned summary network implemented within the `BayesFlow` framework. The network received raw fixation sequences (fixation durations and associated indices) and transformed them into low-dimensional embeddings via a dense neural architecture. This approach aimed to automatically discover non-linear and potentially more informative feature representations, thereby avoiding manual feature engineering biases.

Training: The learned summary network was jointly trained with a CouplingFlow-based inference network and an Adapter module for conditioning. The training dataset comprised 10,000 simulated (parameter, sequence) pairs generated from the saliency-driven fixation model (Bruce and Tsotsos (2005)) with gamma-distributed fixation durations. Optimization was performed offline using the Adam optimizer, minimizing the negative log-likelihood of the true parameters under the amortized posterior.

Diagnostics: Posterior predictive checks (Figure 5) reveal that the learned summaries capture the central trend of the observed fixation durations, with the posterior predictive mean (orange dashed line) closely following the empirical mean. However, similar to Model 1, extreme peaks in the observed data remain underestimated by the model. The

simulation-based calibration (SBC) results (Figure 6) show reduced calibration deviations for parameter r compared to Model 1, indicating improved robustness in its estimation. However, ν still exhibits substantial tail bias and μ_T continues to show significant deviations across the rank spectrum.

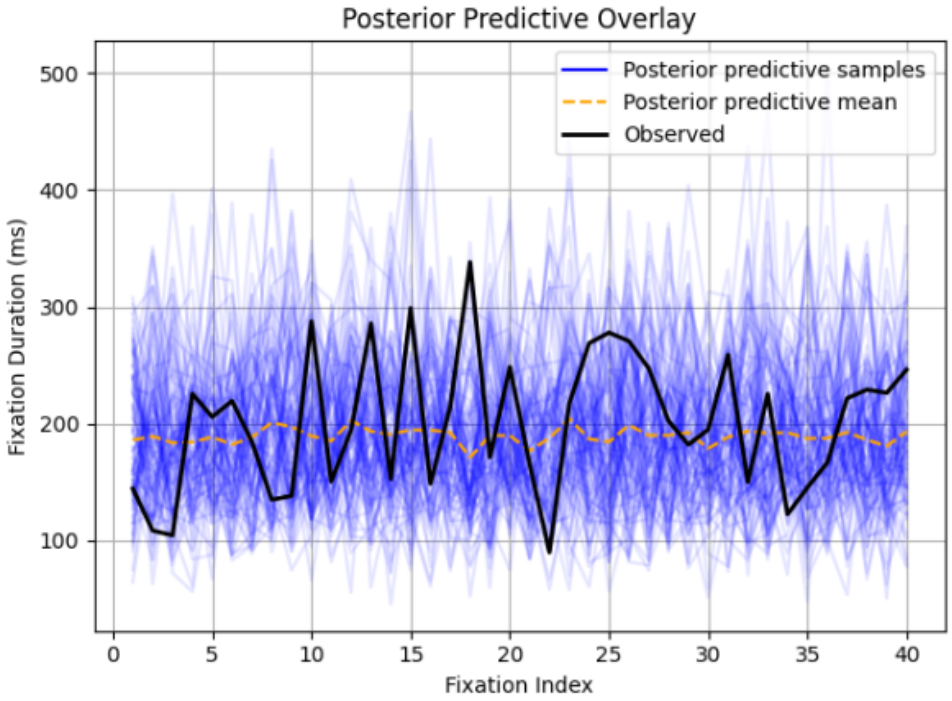

Figure 5: Posterior predictive overlay for Model 2. Blue lines show posterior predictive samples, orange dashed line is the posterior predictive mean, and the black line denotes observed fixation durations.

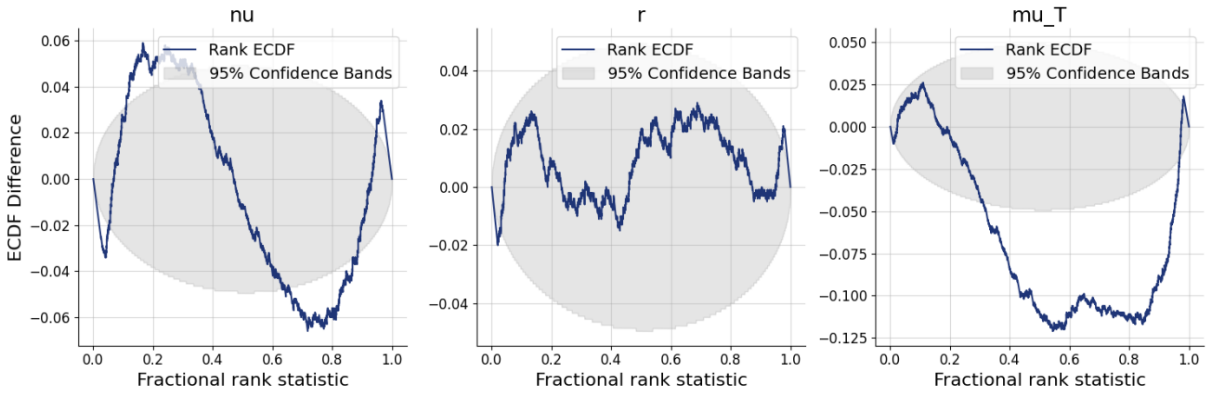

Figure 6: SBC ECDF difference plots for Model 2 parameters (ν, r, μ_T) with 95% confidence bands.

Inference: The parameter recovery plots (Figures 11, 7, and 8) show that r and μ_T maintain strong recovery performance with correlations of 0.8371 and 0.9598, respectively, although slightly lower than Model 1. Recovery of ν remains poor (Corr $= -0.0575$), with estimates that are largely uncorrelated with their true values. RMSE values for ν, r, and μ_T were 0.2791, 2.4535, and 16.2151, respectively. These results suggest that while the learned summary network performs similarly to manual summaries for well-recovered parameters, it does not substantially improve inference for poorly identifiable parameters such as ν.

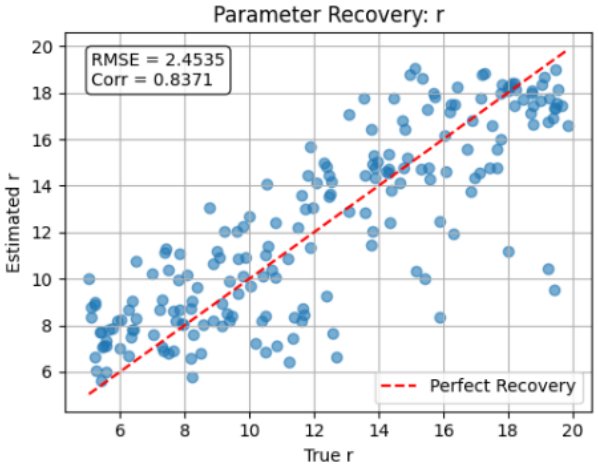

Figure 7: Parameter recovery plot for Model 2: r with RMSE and correlation values. The red dashed line denotes perfect recovery.

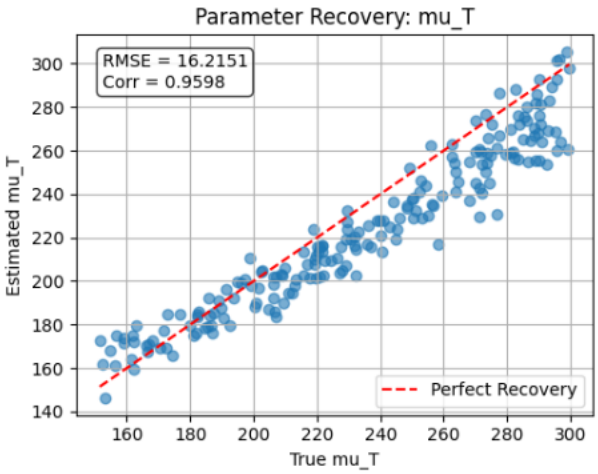

Figure 8: Parameter recovery plot for Model 2: μ_T with RMSE and correlation values. The red dashed line denotes perfect recovery.

3.3 Comparison of Model Variants

Three simulation-based inference models were evaluated, differing in the way eye-movement sequences were generated from latent parameters ν, r, and μ_T:

1. **Full Model:** The SWIFT-inspired simulation with full saliency dynamics, where word activations decayed with distance and guided probabilistic saccade selection.
2. **No Saliency Model:** A control variant without saliency-based selection, producing random word positions but retaining parameter-driven fixation durations.
3. **Random Model:** A baseline model generating both fixation positions and durations entirely at random, with no dependence on ν or r.

All models used a `SummaryNetwork` to embed padded fixation sequences (word index, duration) into fixed-length representations, followed by a `CouplingFlow` inference network for posterior estimation. Each model was trained with *offline amortized Bayesian inference* using $N = 10{,}000$ simulated (θ, x) pairs and the Adam optimizer with default `BayesFlow` settings.

Parameter Recovery Performance Figures 9, 12, and 13 compare the absolute parameter recovery errors for the three variants when estimating parameters from sequences generated by the full SWIFT-inspired simulation. The **Full Model** achieved the lowest recovery error across all parameters, with near-perfect accuracy for ν (0.163) and μ_T (4.108), and low error for r (2.012). The **No Saliency Model** exhibited a substantial increase in error for all parameters, with ν error rising to 1.165, r error to 4.339, and μ_T error to 44.052. The **Random Model** showed intermediate performance, with ν error of 0.233, r error of 1.349, and a notably high μ_T error of 32.482.

These results indicate that incorporating saliency dynamics is critical for accurate recovery of the rate parameter r and for stable estimation of μ_T, while ν can still be moderately recovered even in the absence of structured spatial guidance.

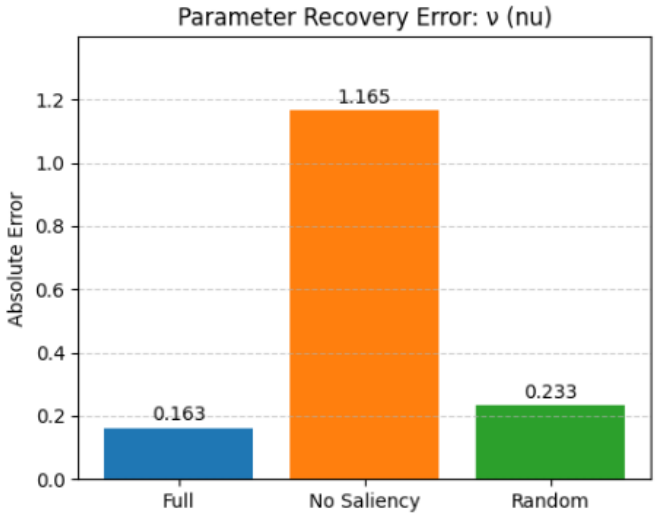

Figure 9: ν Full Model, No Siliency Model, and Random Model.

4 Discussion & Conclusion

This project aimed to develop and evaluate simulation-based Bayesian inference models for estimating cognitive parameters of eye-movement behavior from fixation sequence data. The primary objectives were to compare handcrafted summary statistics with `Bayesflow` summary network, and check parameter recovery performance across different generative assumptions (with saliency, no saliency, and random models). The analysis produced the following key findings:

- **Model 1 (Manual Summaries)**: Features captured core aspects of fixation behavior and enabled accurate recovery for r (RMSE = 1.4680, Corr = 0.9528) and μ_T (RMSE = 11.6340, Corr = 0.9860), but recovery for ν remained poor (RMSE = 0.2682, Corr = 0.1018). Posterior predictive checks indicated a good fit to mean fixation durations but underestimated extreme peaks.
- **Model 2 (`Bayesflow` Inference Network)**: Replacing handcrafted features with a dense neural summary network did not substantially improve ν recovery (Corr = -0.0575) and slightly reduced r and μ_T correlations, although SBC results suggested improved calibration for r.
- **Model Variants with Different Generative Assumptions**: The *Full Model* yielded the lowest recovery errors across all parameters, with near perfect estimation for ν (0.163) and μ_T (4.108), and low error for r (2.012). Removing saliency dynamics (*No Saliency*) substantially increased errors, raising ν to 1.165, r to 4.339, and μ_T to 44.052. The *Random Model* showed intermediate performance, with ν error of 0.233, r error of 1.349, and high μ_T error of 32.482.
- Across models, μ_T was generally recovered with higher accuracy than ν, especially with different summary approaches. While the full generative model achieved strong recovery for both parameters, removing saliency dynamics led to severe degradation in μ_T estimates, underscoring that fixation duration-related parameters are more robust overall but still sensitive to generative assumptions.

These results suggest that while handcrafted statistics suffice for well-identified parameters, learned summaries may offer calibration benefits for some parameters but not necessarily accuracy gains. Moreover, incorporating realistic saliency-driven saccade targeting is critical for recovering rate parameters such as r.

Limitations are the reliance on simulated data from specific generative settings, which may not capture all aspects of real reading behavior, and the observed persistent difficulty in recovering ν across approaches. Future work would be:

- Explore hybrid summary approaches combining handcrafted and learned features.
- Increase model capacity or sequence length to better capture spatial structure.
- Extend simulations to heterogeneous readers or varying text complexities to assess model generalizability.

Overall, this study demonstrates the utility of amortized simulation-based inference for modeling cognitive parameters in reading and highlights the interplay between summary representation choice and generative assumptions in parameter identifiability.

Bibliography

Johannes Berkhof, Iven Van Mechelen, and Herbert Hoijtink. Posterior predictive checks: Principles and discussion. *Computational Statistics*, 15(3):337–354, 2000.

Neil Bruce and John Tsotsos. Saliency based on information maximization. *Advances in neural information processing systems*, 18, 2005.

Ralf Engbert and Maximilian M Rabe. A tutorial on bayesian inference for dynamical modeling of eye-movement control during reading. *Journal of Mathematical Psychology*, 119:102843, 2024.

Ralf Engbert, Antje Nuthmann, Eike M Richter, and Reinhold Kliegl. Swift: a dynamical model of saccade generation during reading. *Psychological review*, 112(4):777, 2005.

Stefan T. Radev, Marvin Schmitt, Valentin Pratz, Umberto Picchini, Ullrich Köthe, and Paul-Christian Bürkner. JANA: Jointly amortized neural approximation of complex Bayesian models. In *Proceedings of the Thirty-Ninth Conference on Uncertainty in Artificial Intelligence*, volume 216 of *Proceedings of Machine Learning Research*, pages 1695–1706. PMLR, 2023a.

Stefan T. Radev, Marvin Schmitt, Lukas Schumacher, Lasse Elsemüller, Valentin Pratz, Yannik Schälte, Ullrich Köthe, and Paul-Christian Bürkner. BayesFlow: Amortized Bayesian workflows with neural networks. *Journal of Open Source Software*, 8(89):5702, 2023b.

Lisa Schwetlick, Lars Oliver Martin Rothkegel, Hans Arne Trukenbrod, and Ralf Engbert. Modeling the effects of perisaccadic attention on gaze statistics during scene viewing. *Communications biology*, 3(1):727, 2020.

Alvaro Tejero-Cantero, Jan Boelts, Michael Deistler, Jan-Matthis Lueckmann, Conor Durkan, Pedro J Gonçalves, David S Greenberg, and Jakob H Macke. Sbi–a toolkit for simulation-based inference. *arXiv preprint arXiv:2007.09114*, 2020.

A Additional figures

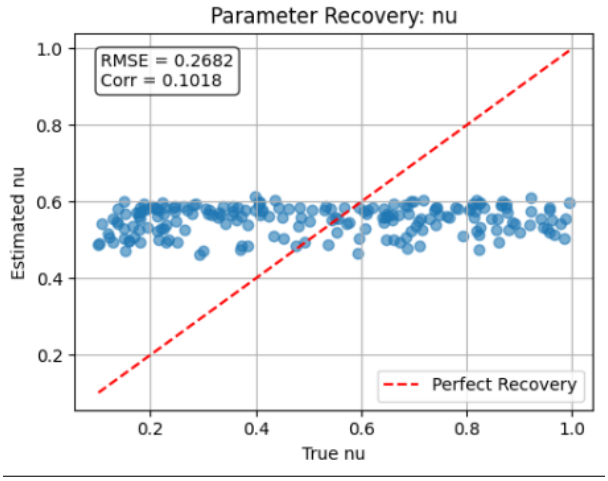

Figure 10: Parameter recovery plot for Model 1: ν with RMSE and correlation values. The red dashed line denotes perfect recovery.

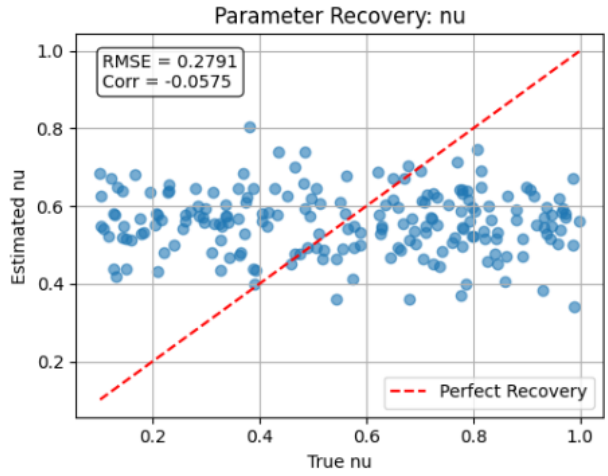

Figure 11: Parameter recovery plot for Model 2: ν with RMSE and correlation values. The red dashed line denotes perfect recovery.

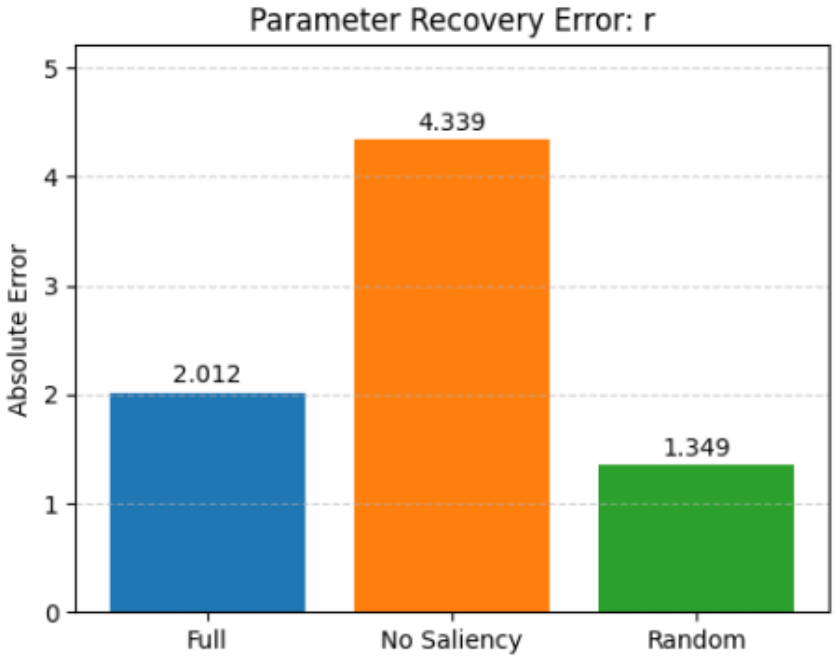

Figure 12: r Full Model, No Siliency Model, and Random Model.

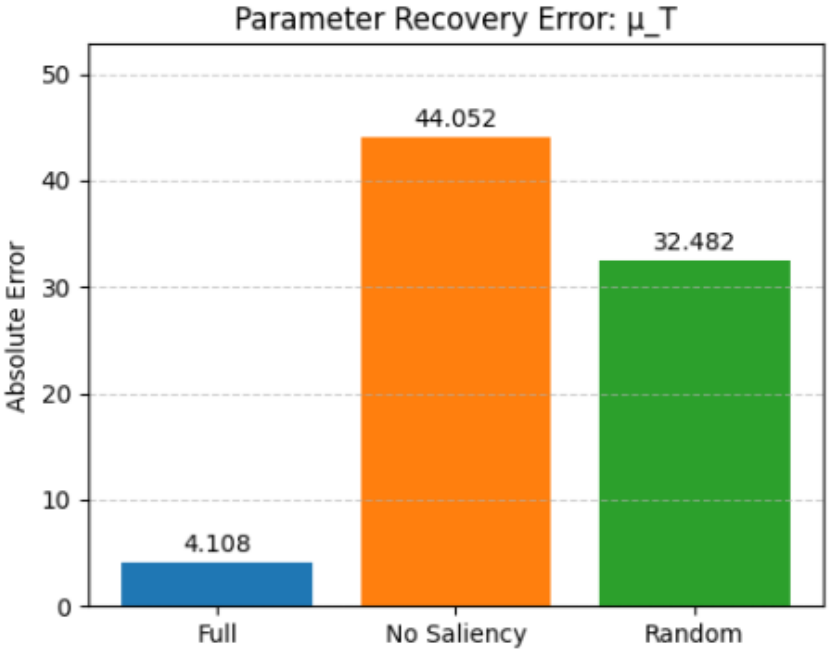

Figure 13: μ_T Full Model, No Siliency Model, and Random Model.