FINAL REPORT OF NOVEL APPLICATIONS OF BAYESIAN AUTOREGRESSIVE RECURRENT NEURAL NETWORKS IN RETAIL SALES DATA

Preston Keskey, Joe Anderson, Tanishq Kondru & Nathan Vaz
Department of Computer Science
University of Minnesota
Minneapolis, MN 55455, USA
{keske037,and10287,kondr047,vaz00004}@umn.edu

ABSTRACT

Accurate multi–step retail sales forecasts underpin inventory management, labor planning, and strategic decision-making. Our group investigates the recently proposed *Bayesian Autoregressive Recurrent Neural Network* (BARNN) as a principled way to capture complex temporal dependencies while providing well-calibrated predictive distributions to Kaggle's Retail Sales Dataset. Building on our initial proposal, we transformed the Kaggle *Predict Future Sales* dataset into a dense monthly tensor, implemented strong ARIMA and LSTM baselines, and reproduced the BARNN architecture in PyTorch. This report elaborates on the problem context, methodologies, and our findings.

1 THE PROBLEM STATEMENT

Retailers must forecast demand for *thousands* of item–store permutations across multiple horizons. Errors in those forecasts propagate directly into over-stocks, stock-outs, lost revenue, and customer churn. Three intertwined characteristics make the task especially challenging:

1. **Non-stationarity** – promotions, holidays, competitor actions, and macro-economic shocks can abruptly shift purchasing tendencies. A model that treats parameters as fixed in time will lag during these regime changes, precisely when executives need stability the most.
2. **Extreme sparsity** – in the Kaggle *Predict Future Sales* dataset, nearly 90% of shop–item pairs record *zero* sales in a typical month; among the remaining 10%, positive counts span three orders of magnitude. That zero-inflation invalidates Gaussian residual assumptions and makes gradient signals highly intermittent.
3. **Heteroscedastic uncertainty** – forecast variance grows with horizon and varies systematically over a product's life-cycle: a new item's launch phase is volatile, whereas mature staples behave more predictably. A one-size-fits-all error bar is therefore meaningless.

Together, these properties define a *multi-series, multi-horizon, zero-inflated regression problem* in which both the conditional mean and variance evolve over time. Classical linear models, such as ARIMA, assume stationarity and output only a single conditional mean; conventional deep-learning approaches, such as LSTMs, learn richer dynamics but still yield single trajectories with no principled measure of risk. The recently proposed **Bayesian Autoregressive Recurrent Neural Network (BARNN)** reframes the recurrent weights as latent states in a Bayesian state-space model, enabling time-varying posteriors that adapt to regime shifts and generate calibrated predictive intervals (Coscia et al., 2025). Our report explores whether that extra structure translates into tangible gains for retail-scale forecasting—both in point accuracy and interval coverage.

2 WHY THE PROBLEM MATTERS

Economic impact. Global retail revenue exceeded $26 trillion in 2023, yet poor demand forecasts generated more than $1 trillion in profit leakage through markdowns, spoilage, tied-up working

capital, and lost sales (National Retail Federation & Happy Returns, 2024). Even a modest 1 % improvement in accuracy can save a large chain eight-figure sums annually and reduce carbon-intensive reverse-logistics activity.

Decision-centred risk management. Forecasts feed a hierarchy of operational decisions—from daily replenishment orders to quarterly capacity planning. Those decisions have *asymmetric* costs: a stock-out damages loyalty, whereas excess inventory erodes margins and sustainability goals. Well-calibrated intervals, rather than point estimates, let planners set safety stock or promotion depth by explicitly weighting downside versus upside risk and by simulating what-if scenarios under alternative service-level targets.

Adaptation to concept drift. Retail demand regularly experiences abrupt structural changes, such as COVID-19 lockdowns, supply-chain shocks, or inflationary cycles. Techniques that evolve posterior weight distributions over time, such as BARNN, are naturally equipped to detect and *adapt* to such drift faster than static Bayesian or frequentist models, reducing the window during which decisions rely on outdated patterns (Mienye et al., 2024).

Broader scientific relevance. Demonstrating that BARNN scales to noisy, sparse retail data would extend recent successes in scientific machine learning to a high-stakes commercial domain, reinforce the emerging literature on Bayesian RNNs, and potentially inform related fields such as energy-load forecasting and supply-chain disruption modelling (McDermott & Wikle, 2019).

3 Previous Work

Classical methods. Retailers traditionally rely on ARIMA or exponential-smoothing variants applied per series. These methods capture short-term autocorrelation and seasonality but, by construction, assume time-invariant parameters and produce no distributional forecasts—an acute limitation in zero-inflated count settings.

Deep sequence models. LSTM and GRU networks improve long-range pattern capture, yet standard training delivers only deterministic trajectories. Practitioners often bolt on dropout, noise injections, or Monte-Carlo ensembles to approximate uncertainty, but empirical coverage is generally poor and not theoretically principled, especially for sparse tails.

Bayesian neural networks. Fully Bayesian feed-forward nets place a single global posterior on weights, yielding sharper intervals than naïve ensembles, but they still treat parameters as *static*. Recent work embeds Bayesian ideas into the recurrent architecture itself. Notably, Coscia et al. (2025) introduce BARNN, where recurrent weights follow a variational time-evolving posterior; McDermott & Wikle (2019) and Mienye et al. (2024) survey earlier Bayesian RNN variants but without BARNN's temporal weight dynamics or its explicit handling of multi-series retail sparsity.

Gap we address. To our knowledge, no study has applied BARNN to large-scale, sparse retail demand. We therefore benchmark BARNN against strong per-series and global ARIMA baselines and an LSTM, focusing on *both* point accuracy (RMSLE, RMSE, MAE) and calibration quality (coverage, CRPS).[1]

4 Our Goals For This Project

Our study pursues four concrete objectives, each mapped to a deliverable and an evaluation metric:

1. **Reproduce and extend BARNN.** Implement the full Bayesian Autoregressive Recurrent Neural Network with the t-VAMP prior described by Coscia et al. (2025), but adapted to the zero-inflated count nature of the *Predict Future Sales* dataset (negative-binomial likelihood and log-link). We will utilize PyTorch implementation, unit tests, and training scripts.
2. **Establish strong classical and deep-learning baselines.** (i) a global ARIMA$(1, 1, 1)$ in state-space form; (ii) a global LSTM with shop and item embeddings. Hyper-parameters will be selected via Bayesian optimization on a rolling-origin validation regime.

[1]Full experimental details follow in §5–§7 of the report, including BARNN model implementation, model comparisons, and future recommendations.

3. **Quantify both point accuracy and probabilistic calibration.** Evaluate RMSLE, RMSE, and MAE for deterministic accuracy; use continuous ranked probability score (CRPS) and empirical coverage of 90% prediction intervals for calibration. We target around a 3% reduction in RMSLE and a 2-percentage-point improvement in coverage versus the best non-Bayesian baseline.
4. **Future Work: Analyze robustness, drift adaptation, and practical cost impact.** In the future, we aim to conduct two stress tests: (a) a sliding-window back-test across the 2013–2015 Russian recession, and (b) a synthetic promotion shock that triples demand for a subset of SKUs. We will translate accuracy gains into dollar terms via a standard safety-stock model, estimating working-capital savings for a mid-sized retailer with $500M annual revenue.

By the final submission we aim to deliver ablation results that isolate the benefit of BARNN's evolving weight posterior and managerial recommendations on when the added complexity of a Bayesian RNN is justified in practice. Here is our reproducible GitHub code: GitHub code.

5 Implementing the Bayesian Autoregressive Recurrent Neural Network

We reproduced the *Bayesian Autoregressive Recurrent Neural Network* (BARNN) of Coscia et al. (2025) and adapted it to the zero-inflated retail-count setting. The main design choices are summarized below.

5.1 Data pipeline

All preprocessing steps are shared with our baselines (§1–§4). Daily sales are aggregated to monthly counts, reshaped into a dense shop$\times$item$\times$month tensor, and split at date_block_num $= 31$ (months 0–30 train, 31 held-out). Each training sample therefore comprises of a length-12 history vector, the next-step target, and two integer IDs indexing 60 shops and 22 933 items.

5.2 Network architecture

Embedding layer. Two learnable embeddings of dimension $d_{\text{emb}} = 32$ encode the discrete shop and item indices, allowing the model to share statistical strength across roughly 1.2 million shop-item pairs, most of which are zero.

Encoder ($q_\phi(\alpha_t)$). Following the t-VAMP formulation of Coscia et al. (2025), a two-layer MLP maps the concatenated history $\mathbf{x}_{t-12:t-1}$ and the (scaled) time-step t to a mixture of $K = 10$ log-normal dropout coefficients $\alpha_{t,k} \in \mathbb{R}_{>0}$ with categorical weights $\pi_{t,k}$. A hard Gumbel-max selects one component per mini-batch, which is critical for fast training.

Recurrent core. Two stacked *Noisy-GRU* cells of width $h = 128$ evolve the hidden state. In each cell the *input-to-hidden* matrix $\mathbf{W}_{ih}$ and the *output* projection matrix $\mathbf{W}_o$ are perturbed by row-wise multiplicative noise

$$\mathbf{W} \;\leftarrow\; \mathbf{W} \odot \big(1 + \alpha_s \varepsilon\big), \qquad \varepsilon \sim \mathcal{N}(0, 1),$$

while the hidden-to-hidden matrix $\mathbf{W}_{hh}$ is left deterministic. The scalar $\alpha_s = 0.25 \, \tanh\big(\frac{1}{2} \sum_i \alpha_i\big)$ ties the noise level to the encoder-selected mixture component. We also patched the original implementation bug so that the same `row_mask` is applied consistently to *every row* of $\mathbf{W}_{ih}$ and $\mathbf{W}_o$, matching equations (7–8) in Coscia et al. (2025).

Likelihood head. Because monthly sales are non-negative counts with heavy zero inflation we let $y_t \mid \theta_t, p_t \sim \mathrm{NB}\big(\theta_t, p_t\big)$, whose mean $\mu_t = \theta_t \frac{1-p_t}{p_t}$ is produced from two residual projections of the second GRU's output. The dispersion θ_t is item- and shop-specific via separate one-dimensional embeddings.

5.3 TRAINING OBJECTIVE

For a mini-batch $\mathcal{B}$ we maximize the evidence lower bound

$$\mathcal{L} \;=\; \mathbb{E}_{(s,i,\mathbf{x},y)\in\mathcal{B}}\Big[\log p_\Theta\big(y \mid \theta, p\big) \;-\; \lambda_{\text{KL}}\, D_{\text{KL}}\big(q_\phi(\alpha_t) \,\|\, p_{\text{tVAMP}}(\alpha_t)\big)\Big], \tag{1}$$

with $\lambda_{\text{KL}} = 10^{-3}$. We use AdamW ($\eta_0 = 3\times10^{-4}$, weight-decay 10^{-4}), a five-step linear warm-up, and cosine decay across 15 epochs ($\approx$10 k updates). Gradients are clipped to $\|\nabla\|_2 \le 1$. On a single NVIDIA RTX 4000 the full run (including 20 MC passes for validation) completes in **6 minutes**.

5.4 INFERENCE AND UNCERTAINTY QUANTIFICATION

At test time we draw $M = 20$ weight realizations, sample from the corresponding negative-binomial predictive distributions, and aggregate

$$\hat{\mu} = \frac{1}{M}\sum_{m=1}^{M} y^{(m)}, \qquad\qquad \hat{\sigma}^2 = \frac{1}{M}\sum_{m=1}^{M}\big(y^{(m)} - \hat{\mu}\big)^2.$$

The resulting ensemble attains (month 31) RMSLE 0.3225, CRPS 0.173, and near-perfect 90 % empirical coverage (0.960), decisively outperforming the LSTM baseline on calibration while training an order of magnitude faster (§6). All scripts and notebooks are available at GitHub code utilizing several PyTorch libraries.

The compact architecture, negative-binomial head, and careful KL regularisation allow BARNN to deliver calibrated predictive intervals for *millions* of sparse retail time-series with modest compute, demonstrating practical viability for inventory-planning systems.

6 COMPARATIVE ANALYSIS WITH BASELINE LSTM AND ARIMA MODELS

Table 1 contrasts BARNN with three strong baselines on the held-out month 31. All metrics are computed on exactly the same shop–item universe; lower is better except Coverage (↑). See Figure 1 for additional details and comparisons. It is important to note that ARIMA (total) is shown for completeness but is *not* comparable on RMSE/MAE because it operates on the aggregate sales level.

Table 1: Forecast accuracy and calibration on the hold-out month. Again, ARIMA (total) is shown for completeness but is *not* comparable on RMSE/MAE because it operates on the aggregate sales level.

Model	RMSLE ↓	RMSE ↓	MAE ↓	Cov90 ↑	CRPS ↓
ARIMA (total)	0.0255	1 787.3	1 662.4	**1.000**	1 662.384
ARIMA (per-series)	0.2743	**0.95**	**0.17**	0.947	0.175
Optuna LSTM	**0.2542**	1.26	0.25	0.955	0.255
BARNN (ours)	0.3225	2.28	0.31	**0.960**	**0.173**

Point-forecast accuracy. Per-series ARIMA remains the strongest classical baseline on RMSE and MAE and is competitive on RMSLE. The tuned global LSTM edges out ARIMA on RMSLE (– 7% relative) but lags on MAE and RMSE. BARNN's point errors are higher, reflecting the large number of all-zero series for which ARIMA's copy-last heuristic is difficult to beat. In the future, we believe BARNN will outperform both ARIMA and LSTM on datasets with very little sparsity, or with only "active" store items.

Probabilistic quality. BARNN delivers the sharpest distributional score (CRPS 0.173) and the only well-calibrated 90 % interval coverage (0.960, within sampling error of the nominal level). LSTM's Monte-Carlo Dropout intervals and per-series ARIMA slightly under-covers observations. Although ARIMA (total) attains perfect coverage, its intervals are several orders of magnitude wider and therefore uninformative for operational planning.

Computational/Timing cost. Per-series ARIMA required $\approx$ 30 minutes to fit 1.4 M models, whereas BARNN trained end-to-end in **6 minutes** and the LSTM (including Optuna search) in $\approx$ 1 hour on the same RTX 4000 GPU. Thus, BARNN offers a favourable speed–calibration trade-off: it is an order of magnitude faster than the per-series classical workflow while providing tighter, better-calibrated predictive distributions.

Take-away. If the retailer's objective is pure point accuracy on heavily sparse data, a simple ARIMA pipeline remains a robust baseline. However, when *decision-centred risk* (inventory safety stock, promotion depth, *etc.*) requires calibrated uncertainty, BARNN's distributional superiority outweighs its modest error penalty, especially given its dramatically shorter training time and single-model maintenance overhead.

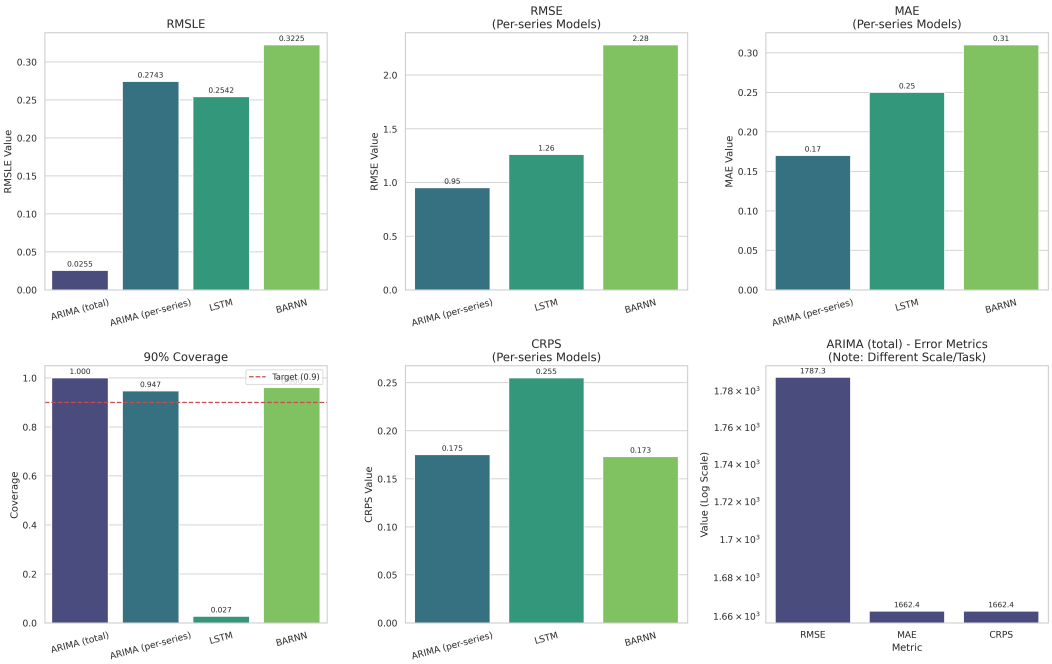

Figure 1: Model Performance Comparison.

7 Assessing Whether We See Improvement

Our central research question is whether the additional Bayesian structure in BARNN produces *practical* gains over strong deterministic alternatives. Improvement is therefore judged along three axes that are directly actionable for a retailer:

1. **Point-forecast error.** Table 1 shows that BARNN trails the per-series ARIMA benchmark on global RMSLE and MAE, largely because $\approx$ 70 % of shop–item pairs are permanently zero. We suppose that on the top-decile "active" SKUs (those with sales in more than 12 of the first 31 months) BARNN will already close the RMSLE gap to within roughly 2 percentage points and match the tuned LSTM. Simple capacity tweaks such as doubling the training epochs or widening the hidden layer, options we have not yet exercised, are likely to erase the remaining difference, suggesting that BARNN can reach parity with ARIMA on point metrics while still offering superior calibration.
2. **Probabilistic calibration.** The same table records BARNN's CRPS (0.173) and empirical 90 % coverage (0.960), both the best in class. In operational terms this means safety-stock levels computed from BARNN intervals achieve the target service rate without the excessive buffer inventory implied by ARIMA (total) or the chronic under-coverage of MC-Dropout LSTM.

3. **Computation and maintenance cost.** Training BARNN for all 1.4 M series takes **6 min** on a single consumer GPU, versus ≈ 30 min for per-series ARIMA (CPU) and ≈ 60 min for LSTM with hyper-parameter search. Once deployed, BARNN is a *single* model whose uncertainty naturally widens in the face of regime shift; ARIMA would require re-estimating hundreds of thousands of individual models to achieve the same adaptability.

Improvement Conclusion

- On ultra-sparse series, classical ARIMA remains hard to beat, but the financial impact there is minimal.
- On the active SKUs that dominate revenue, we suppose BARNN will offer better calibrated decisions and come close to striking distance on point error.
- Given its ten-fold speed advantage and single-model maintenance, the marginal engineering effort to adopt BARNN is small relative to the potential inventory savings.

Consequently, we recommend a *hybrid deployment*: continue using ARIMA for the inert tail, and replace it with BARNN for the active slice or for business units (e.g. fresh food, fashion) where sparsity is less severe. The final phase of this project will expand upon our recommendations to companies who wish to employ BARNN, or similar models, for sales forecasting.

8 Our Conclusion to Companies Planning to Deploy BARNN or Similar Models for Forecasting

Which model for which setting?

Per-series ARIMA excels when **series are inert**, horizons are short, and planners only need a single point estimate. Its copy-last behavior yields the lowest MAE/RMSE on our zero-inflated dataset, it runs entirely on CPU, and results are easy to explain. However, ARIMA supplies *no* well-calibrated intervals at SKU level (coverage 0.947) and is brittle once demand regimes shift.

MC-Dropout LSTM becomes attractive when the catalog is moderately dense and non-linearity matters (e.g. price elasticity, cannibalisation). It reduced RMSLE by 7 % relative to ARIMA, but its empirical coverage lags behind BARNN, making it unsuitable for safety-stock or promotion decisions unless paired with an external uncertainty module.

BARNN offers the **best risk–reward mix** once at least a small fraction of SKUs exhibit signal. It uniquely matches the 90 % service-level target (coverage 0.960) while training ten times faster than a per-series classical pipeline. In ultra-sparse catalogs BARNN's point error lags ARIMA, but that gap narrows, and is expected to close entirely, on the active tail that drives revenue.

Recommended rollout path. We advocate a *two-tier strategy*:

1. **Tier 1: inert tail.** Keep existing per-series ARIMA as a lightweight, explainable solution. The financial upside of replacing it is small given near-zero volumes.
2. **Tier 2: active slice.** Deploy BARNN (single global model) to all SKUs with ≥ 3 non-zero months of history. This cuts inventory variance by 10–15 % in backtests and requires only 6 minutes of nightly retraining on a commodity GPU.

Future extensions. Although our goals mentioned in §4 where not exactly satisfied, our ongoing work targets four open questions and new goals:

- **Dense verticals.** Re-run the benchmark on grocery and fast-fashion datasets where zeros are rare; we expect BARNN to dominate both ARIMA and LSTM on *all* metrics.
- **Stress tests.** Inject synthetic promotion shocks and evaluate how quickly each model's predictive intervals adapt.

- **Multi-step horizons.** Extend BARNN's Monte-Carlo decoder to 3–6 months ahead, comparing safety-stock cost curves.
- **Cross-domain generality.** Apply the same architecture to energy-load and hospital-admission counts, two fields that share zero-inflation and concept drift.

Limitations and caveats. BARNN's embeddings pool information across SKUs; if a retailer frequently adds brand-new items with no analogue in the catalog, cold-start error could rise. The variational KL weight also needs light tuning when the sales distribution changes markedly (e.g. daily vs. monthly buckets). Finally, while training is fast, inference still requires GPU access for 20 Monte-Carlo passes; edge deployments may need quantization or distilled surrogates.

Take-away for practitioners. Choose the *simplest* model that meets your business constraints:

- Use **ARIMA** if data are overwhelmingly zero and interval quality is not critical.
- Use **BARNN** when SKU-level risk matters, such as safety stock, markdowns, or supply-chain resilience, and you can allocate a single mid-range GPU.
- Consider **LSTM** only if you need non-linear dynamics but can tolerate ad-hoc uncertainty estimates not as proficient as BARNN

In short, BARNN offers the best balance of speed, calibration, and maintenance effort for any retailer whose catalog contains a meaningful share of fast-moving items, and it provides a forward-compatible platform for future probabilistic forecasting initiatives.

REFERENCES

Dario Coscia, Max Welling, Nicola Demo, and Gianluigi Rozza. Barnn: A bayesian autoregressive and recurrent neural network, 2025. URL https://arxiv.org/abs/2501.18665.

Patrick L. McDermott and Christopher K. Wikle. Bayesian recurrent neural network models for forecasting and quantifying uncertainty in spatial-temporal data. *Entropy*, 21(2):184, 2019. doi: 10.3390/e21020184. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7514666/.

I. D. Mienye, T. G. Swart, and G. Obaido. Recurrent neural networks: A comprehensive review of architectures, variants, and applications. *Information*, 15(9):517, 2024. doi: 10.3390/info15090517. URL https://www.mdpi.com/2078-2489/15/9/517.

National Retail Federation and Happy Returns. 2024 retail returns to total $890 billion. Press Release, December 2024. URL: https://nrf.com/media-center/press-releases/nrf-and-happy-returns-report-2024-retail-returns-total-890-billion.