Switch-Hurdle: A MoE Encoder with AR Hurdle Decoder for Intermittent Demand Forecasting

Fabian Mușat, Simona Căbuz

eMAG, Romania
fabian.musat@emag.ro, simona.cabuz@emag.ro

Abstract

Intermittent demand, a pattern characterized by long sequences of zero sales punctuated by sporadic, non-zero values, poses a persistent challenge in retail and supply chain forecasting. Both traditional methods, such as ARIMA, exponential smoothing, or Croston variants, as well as modern neural architectures such as DeepAR and Transformer-based models often underperform on such data, as they treat demand as a single continuous process or become computationally expensive when scaled across many sparse series. To address these limitations, we introduce Switch-Hurdle: a new framework that integrates a Mixture-of-Experts (MoE) encoder with a Hurdle-based probabilistic decoder. The encoder uses a sparse Top-1 expert routing during the forward pass yet approximately dense in the backward pass via a straight-through estimator (STE). The decoder follows a cross-attention autoregressive design with a shared hurdle head that explicitly separates the forecasting task into two components: a binary classification component estimating the probability of a sale, and a conditional regression component, predicting the quantity given a sale. This structured separation enables the model to capture both occurrence and magnitude processes inherent to intermittent demand. Empirical results on the M5 benchmark and a large proprietary retail dataset show that Switch-Hurdle achieves state-of-the-art prediction performance while maintaining scalability.

1 Introduction

Accurate prediction of retail demand directly impacts inventory optimization, warehousing efficiency and capital utilization. Inaccurate results, particularly in high-volume, high-SKU environments lead to significant financial penalties, ranging from increased rental costs due to overstocking to reduced customer service levels caused by stock-outs. The most difficult challenge in this domain stems from intermittent demand, a common pattern across retail, characterized by sporadic bursts of sales combined with long periods of zero activity (Garza, Challu, and Mergenthaler-Canseco 2023). Both traditional forecasting methods, such as ARIMA or exponential smoothing, as well as contemporary state-of-art deep learning methods, like DeepAR or Transformer variants, often struggle with this dual-state data. These models typically treat the entire time series as single processes, leading to biased architectures towards smoothing.

Modern Transformer based approaches leverage attention and encoder-decoder structures to capture long-range dependencies in time series patches. However, scaling these solutions is computationally expensive and most often they still fail to capture the fundamental heterogeneity of intermittent data.

To overcome these limitations we introduce Switch-Hurdle: a new Mixture-of-Experts (MoE) encoder decoder architecture designed to address probabilistic forecast of intermittent demand. Our contributions are as follows:

- Switch–Hurdle architecture: A Transformer with an approximately dense expert routing in the encoder and a lightweight cross-attention autoregressive decoder tailored for intermittent demand.
- Hurdle head for zeros and deficits: A shared hurdle head that separates "whether a sale happens" from "how much is sold," handling both zero-inflation and zero-deflation.
- Covariate integration and training dynamics: A simple AR decoding scheme that combines prior predictions with future covariates, alongside a stable training recipe supporting probabilistic and point-wise objectives.
- Empirical gains: State-of-the-art WRMSSE on M5 and improved WAPE on an internal dataset, supported by ablations and expert-routing analyses.

The remainder of the paper is organized as follows: Section 2 reviews related work. Section 3 presents the proposed Switch-Hurdle architecture, detailing the Switch MoE encoder, cross-attention autoregressive decoder, hurdle head, and training objectives. Section 4 describes datasets, metrics, and the experimental setup, reports results on M5 and the internal dataset, including baselines and ablations and provides a conditional expert routing analysis. Section 5 concludes with limitations and future directions.

2 Related Work

While tree-based ensembles have long dominated time-series forecasting, Transformer-based methods have recently gained prominence. Contemporary approaches fall into three groups: (i) Transformer-based models, (ii) Mixture-of-Experts (MoE) architectures, and (iii) large Foundation Models (FMs).

Transformers for time series forecasting The shift from recurrent and convolutional models to Transformers improved the modeling of both short and long range dependencies via self-attention. On M5 and related benchmarks, vanilla Transformer, Informer, and TFT outperform AutoARIMA and AutoETS, with MASE gains of about 26–29% and WQL reductions (Caetano, Oliveira, and Ramos 2025; Jin et al. 2024), albeit at higher computational cost (Oliveira and Ramos 2024). Efficiency oriented variants target attention complexity or tokenization: Autoformer introduces sparse mechanisms (Wu et al. 2021); PatchTST uses patch tokenization and improves robustness and accuracy (Nie et al. 2023); LipFormer removes costly components, like LayerNorm or positional encodings, and integrates future covariates via a Dual Encoder, yielding gains across backbones (Wang et al. 2025; Suresh 2025).

Mixture of Experts for time series forecasting MoE scales the model capacity by activating a sparse subset of experts per token. Switch Transformers popularized Top-1 token-wise routing (Fedus, Zoph, and Shazeer 2022). Subsequent work explored soft or continuous mixtures and improved balancing (Puigcerver et al. 2024), and Dense-to-Sparse training that begins dense and gradually sparsifies routing (Nie et al. 2022).

Foundation Models for time series forecasting Large FMs pre-trained on broad time-series corpora demonstrate zero-shot generalization. MOMENT (Goswami et al. 2024), which is trained on the Time-series Pile, uses a patch-style tokenization similar to PatchTST. LLM-based approaches adapt prompting for time series (Jia et al. 2024), while domain specific time-series FMs scaled via MoE (e.g., Time-MoE (Shi et al. 2025), Moirai MoE (Liu et al. 2025)) reach billions of parameters and strong benchmark performance.

Despite recent advancements, in real retail settings, a gap remains: even though Transformers beat statistical baselines, they often lag behind feature-rich methods like LightGBM and DeepAR on the hierarchical WRMSSE metric, which favors good aggregation and strong use of external features. Moreover, often times, dense models can bias toward smoothing, which is problematic for sparse, zero-inflated retail demand. This points to the need for models that better use covariates and explicitly handle zero inflation.

Compared to prior work, we pair a lightweight cross-attention AR decoder with a hurdle head to explicitly model zero occurrence and positive-demand magnitude, while using MoE routing to specialize representations.

3 Method

3.1 Model Architecture

As depicted in Figure 1, our solution combines a dense-to-sparse MoE encoder with a lightweight autoregressive (AR) decoder and a shared hurdle head. This design addresses two key challenges in intermittent demand forecasting: (i) sparsity and zero-inflation, and (ii) the need for multi-step forecasts. The encoder captures temporal and categorical structure through a two-layer MoE; the decoder models the conditional distribution of future demand via a probabilistic hurdle mechanism.

3.2 The Switch MoE Encoder

Inspired from the Switch Transformer's work, (Fedus, Zoph, and Shazeer 2022) our Switch Block works as a pattern recognition component, routing the essential information to the decoder. This mechanism ensures experts' unsupervised specialization, partitioning the token space into clusters of dedicated patterns.

The encoder essentially routes each token representation through a Mixture-of-Experts (MoE) layer composed of E SwiGLU experts $f_e(\cdot)$. Each expert is a position-wise feed-forward network parametrized by $\{W_{1,e}, W_{2,e}, W_{3,e}\}$ that applies a gated activation using SwiGLU, i.e.

$$f_e(x) \;=\; \big(\mathrm{Swish}(xW_{1,e}) \odot (xW_{2,e})\big)W_{3,e}, \qquad (1)$$

where $\mathrm{Swish}(\cdot)$ is the Swish function with $\beta = 1$, i.e. Sigmoid Linear Unit (SiLU).

Now let $x_t \in \mathbb{R}^d$ denote the per-token embedding entering the MoE, with $t \in [1 \dots L]$. A learned router $W_g \in \mathbb{R}^{E\times d}$ produces logits $r_t = W_g x_t$ and routing probabilities $p_t = \mathrm{softmax}(r_t)$.

To encourage expert specialization and improve prediction performance, we adopt a Top-1 straight-through (STE) gate: the forward pass makes a hard, one-hot expert choice per token, while the backward pass uses the soft probabilities to provide stable gradients. At inference, we evaluate only the selected expert per token. This discrete routing concentrates capacity into distinct experts rather than diluting it in a single dense FFN. A secondary benefit is practical efficiency at inference time, where only the selected expert is evaluated. Given the routing probabilities $p_t \in \mathbb{R}^E$, we obtain:

$$g_t \;=\; \mathrm{one\_hot}\big(\arg\max(p_t)\big) \;+\; \big(p_t - \mathrm{stopgrad}(p_t)\big). \quad (2)$$

Here, $\mathrm{stopgrad}(\cdot)$ returns its argument in the forward pass but blocks gradients, making the gate forward-sparse (Top-1) and backward-dense.

The MoE output for step t is

$$x'_t \;=\; \sum_{e=1}^{E} g_{t,e}\, f_e(x_t) \;\in\; \mathbb{R}^d. \qquad (3)$$

Stacking over the context yields the encoder memory

$$X' \;=\; \big[x'_1, \dots, x'_L\big]^\top \;\in\; \mathbb{R}^{L\times d}, \qquad (4)$$

which serves as the decoder's cross-attention memory.

To ensure balanced expert utilization, we regularize the pre-gate probabilities with a KL-to-uniform term:

$$\bar{p}_e = \frac{1}{N}\sum_{i=1}^{N} p_{i,e}, \qquad (5)$$

$$\mathcal{L}_{\text{balance}} = \sum_{e=1}^{E} \bar{p}_e \log \frac{\bar{p}_e}{1/E}, \qquad (6)$$

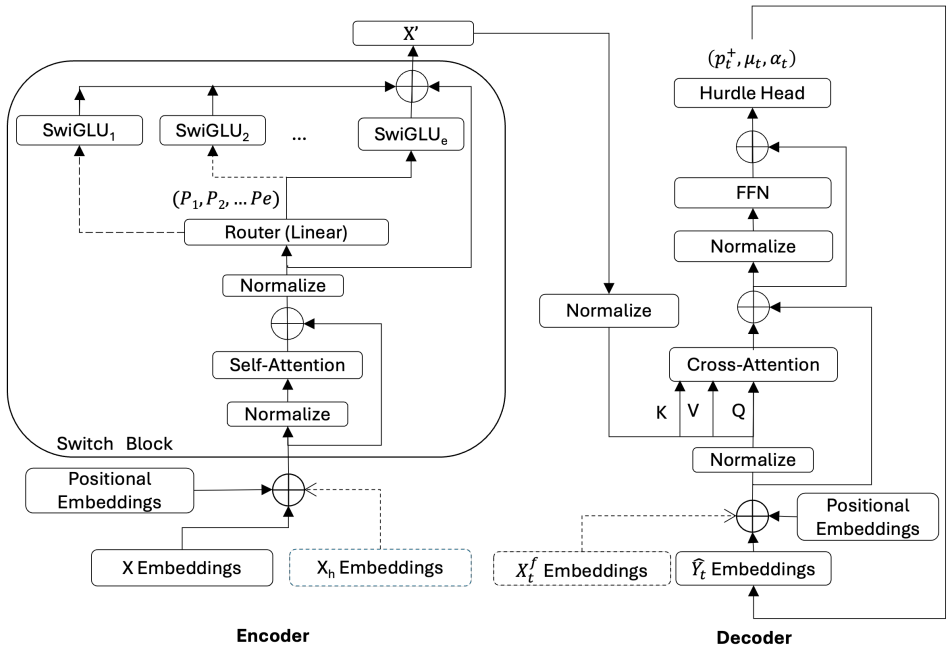

Figure 1: *Main architecture for the Switch-Hurdle Transformer.* The encoder (left) uses Top-1 MoE routing with SwiGLU experts to extract specialized representations of demand and covariate embeddings. The decoder (right) applies cross-attention over the encoder's context memory to generate step-wise probabilistic forecasts (p_t^+, μ_t, α_t). Each step conditions on the previous prediction, future covariates, and positional embeddings, while the shared hurdle head jointly models zero-demand probability and the conditional Negative-Binomial distribution for positive demand.

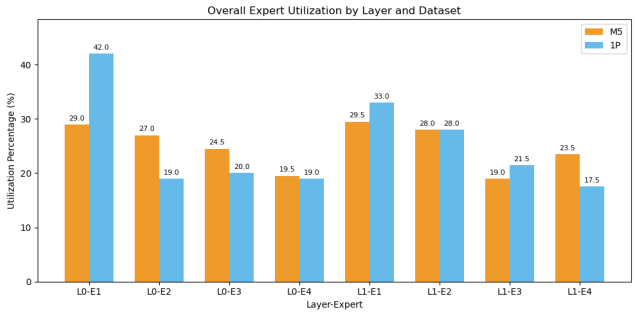

Figure 2: *Overall expert utilization by layer and dataset.* Bars show the percentage of tokens routed to each expert for the two Switch-encoder layers (L0, L1) on M5 and internal 1P data after introducing the KL-to-uniform regularizer. The distribution is balanced without collapse while still reflecting dataset- and layer-specific specialization.

where N is the number of tokens in the minibatch (time steps $\times$ series). This replaces the Switch auxiliary loss and promotes stable specialization without collapse.

Figure 2 illustrates balanced routing induced by the KL regularizer, while the Top-1 STE preserves specialization across layers and datasets. This supports our design choices: Top-1 straight-through routing provides sparse forward selection with dense gradient updates, and the KL-to-uniform regularizer prevents collapse while allowing specialization.

3.3 The Autoregressive Hurdle Decoder

The decoder is a cross-attention only block that produces hidden states h_t for $t = (1, \ldots, T)$. Each h_t is subsequently mapped to the forecast parameters (p_t^+, μ_t, α_t) by the hurdle head, defined in Sec. 3.4. At step t, the decoder conditions on (i) the previously predicted demand $\hat{y}_{t-1}$, (ii) the step-specific future covariates X_t^f (e.g., promotions or planned price changes), and (iii) a learned positional embedding $p_t^{(\text{dec})}$:

$$q_t \;=\; W_p \,\log\!\big(1+\hat{y}_{t-1}\big) \;+\; W_f\, X_t^f \;+\; p_t^{(\text{dec})}, \qquad (7)$$

where W_p and W_f are learned projections.

We use $\log(1+\hat{y}_{t-1})$ to stabilize scale and gradients for count data since it handles zeros and downweights large spikes better; other monotone links such as $\log(\varepsilon + \cdot)$ or a learned embedding are also viable.

Each q_t attends to the encoder output $X' \in \mathbb{R}^{L\times d}$ via cross-attention, yielding a hidden state h_t that integrates historical context with planned future signals. The decoder has no self-attention. Autoregression arises solely from the inclusion of $\hat{y}_{t-1}$ in q_t. We apply teacher forcing with a scheduled ratio, gradually replacing ground-truth y_{t-1} with model predictions $\hat{y}_{t-1}$ during training.

The reason we adopt an AR decoder, conditioning on $\hat{y}_{t-1}$ is to (i) reduce exposure bias while keeping inference simple and fast, and (ii) avoid redundant lag inputs, since long-range information is already available through cross-attention to X'.

3.4 Hurdle Head for Demand Distribution

The decoder output h_t is passed to a shared hurdle head that jointly predicts: (i) the probability of a positive demand event p_t^+, and (ii) the parameters (μ_t, α_t) of a conditional Negative-Binomial (NB) distribution for positive counts:

$$p_t^+ = \sigma(w_p^\top h_t), \tag{8}$$

$$\mu_t = \text{softplus}(w_\mu^\top h_t + b_\mu), \tag{9}$$

$$\alpha_t = \text{softplus}(w_\alpha^\top h_t + b_\alpha). \tag{10}$$

The NB probability of zero is:

$$p_{0,t} = (1 + \alpha_t \mu_t)^{-1/\alpha_t}. \tag{11}$$

The resulting hurdle distribution is

$$P(Y_t = 0) = 1 - p_t^+, \tag{12}$$

$$P(Y_t = y > 0) = \frac{p_t^+}{1 - p_{0,t}} \tag{13}$$

$$P_{\text{NB}}(Y_t = y; \mu_t, \alpha_t), \tag{14}$$

where P_{NB} denotes the NB pmf under the mean–dispersion parameterization:

$$P_{\text{NB}}(y; \mu, \alpha) = \frac{\Gamma(y + \frac{1}{\alpha})}{\Gamma(\frac{1}{\alpha})\, y!} \left(\frac{1}{1 + \alpha\mu}\right)^{\frac{1}{\alpha}} \left(\frac{\alpha\mu}{1 + \alpha\mu}\right)^{y}.$$

This formulation separates the zero-occurrence process from positive-demand magnitude, yielding calibrated probabilistic forecasts for intermittent series.

We use NB for the positive-count component because it models overdispersion, $\text{Var} = \mu_t + \alpha_t \mu_t^2$, common in retail demand, while Poisson forces $\text{Var} = \mu$ and underestimates uncertainty. The hurdle head is modular and can be swapped with any count distribution.

3.5 Training Objectives

The Switch-Hurdle model is optimized with one of two composite objectives, both including the load-balancing term $\mathcal{L}_{\text{balance}}$ in Eq. (6).

Probabilistic Objective. For distributional calibration, we optimize the hurdle negative log-likelihood (NLL) with load balancing:

$$\mathcal{L}_{\text{Prob}} = \mathcal{L}_{\text{Hurdle}} + \lambda_{\text{aux}}\, \mathcal{L}_{\text{balance}}. \tag{15}$$

Point-wise Hybrid Objective. For deterministic accuracy, we combine MAE with the probabilistic structure and anneal its weight:

$$\mathcal{L}_{\text{Total}} = \mathcal{L}_{\text{MAE}} + \lambda_{\text{decay}} \big(\mathcal{L}_{\text{Hurdle}} + \mathcal{L}_{\text{balance}}\big), \tag{16}$$

where λ_{decay} is initialized to 1.0 and reduced by 30% after each epoch until a floor of 0.05, allowing the model to focus increasingly on MAE in later epochs.

4 Experiments

We evaluated our variant of the Switch-Hurdle Transformer model on proprietary, internal datasets and on the external, popular, M5 Walmart dataset (Makridakis et al., 2022). Both tasks aim to forecast demand at the daily level per SKU, with a horizon of a month. M5 contains $30,490$ time series with static features such as store, item and state id, and time-varying features such as past events, campaigns, etc. Our internal dataset contains about $40,000$ time series with static features such as product and vendor id and time-varying features such as past events, promos, campaigns, price-related features, etc. The complexity of both datasets is similar, with M5 being more challenging to predict due to the smaller feature range and variable demand types and peaks. We use a horizon of $T = 28$ and a context or input length of $L = 56$.

For both datasets, we tested two types of training:

- probabilistic: we optimize the loss of the hurdle model, made up of a binary cross-entropy part, for the Bernoulli process, combined with a negative log-likelihood of a truncated negative binomial distribution, for the positive demand part;
- point-wise: we optimize the loss of the hurdle model and a point loss such as mean absolute error (MAE).

4.1 Base results

For our internal datasets, we first took a small (10%) sample that uniformly represents our data, so we can draw some quick conclusions. We started by measuring basic models to establish a baseline.

Table 1: *Classical baseline metrics on the internal sample dataset.* This table compares traditional forecasting methods on a 10% stratified sample of our internal data. Results show that the Croston model, specifically designed for intermittent demand, achieves the best overall accuracy, followed by SARIMAX. These results confirm the zero-inflated nature of the dataset and motivate the use of models that explicitly separate demand occurrence and magnitude, such as the proposed hurdle-based approach.

Model	WAPE	MASE
Naïve	111.14%	1.2064
SARIMAX (1,1,0)	101.71%	1.0757
SARIMAX (1,1,1)	<u>97.67%</u>	<u>1.0330</u>
Croston (Biased)	**93.15%**	**0.9852**

Unsurprisingly, Croston - which is very well suited for intermittend demand - has the best metrics, with SARIMAX coming in a close second.

4.2 Sample results

On this small (10%) sample, we trained some well-established deep learning models using both methods described above and chose the top Transformer-based model to put it up against our Switch-Hurdle Transformer variant.

We also compare our results against zero-shot Time-MoE, a recent time-series foundation model with a Mixture-of-

Experts architecture, on our internal data. We evaluate its point forecasts using WAPE and MASE.

Notes

- **Note 1** - Switch-Hurdle Transformer was trained using a combination of binary cross-entropy and truncated negative binomial negative log-likelihood as the hurdle model imposes it and auxiliary loss (load balancing).
- **Note 2** - Switch-Hurdle Transformer was trained using a combination of mean absolute error and auxiliary loss (load balancing).
- **Note 3** - Tables presenting metrics on M5 contain three entries per row, i.e the WRMSSE and RMSSE metrics corresponds to the model trained using a distribution, and the MASE metric corresponds to the model trained using mean absolute error.

4.3 Main results

We selected TFT (Lim et al., 2021) as the main competitor to our model, because they were very close in terms of metrics and because they support both historical and future covariates, so we trained both models using the methods described above and benchmarked both using WRMSSE and MASE for M5 and WAPE and MASE for the internal dataset. We have added RMSE results on the M5 dataset, for completion. The results show Switch-Hurdle Transformer being capable of state-of-the-art performance, coming slightly ahead of the TFT models in some cases.

4.4 Conditional Expert Routing Analysis

To investigate functional specialization, we analyze expert utilization *conditioned* on the z-score–defined demand regime (Zero, Low, Normal, Spike). We report the conditional routing distribution $P(e \mid \text{regime})$, normalized to 100% per regime and aggregated over the validation split.

As shown in Figure 4, for layer 0, there is a split emerging early in the stack: Expert 3 dominates *Zero* periods (67.5%), Expert 1 captures most *Low/Spike* tokens (90.8% / 94%), and Expert 0 contributes most in *Normal* conditions (67.5%). This suggests Layer 0 forms regime-aware gates that separate no-sale / rare-sale contexts from typical demand.

Figure 5 sharpens this partitioning in layer 1: Expert 2 handles the most frequent regimes, dominating *Zero* (59.3%) and *Normal* (51.3%), while Expert 1 specializes in tail events, capturing *Low* (93.9%) and *Spike* (97.3%). Expert 3 is effectively unused across regimes, indicating stable specialization rather than collapse.

These patterns provide strong empirical evidence that Top-1 STE with KL-to-uniform regularization leads to balanced but regime-specific experts, aligning with the accuracy gains reported in the ablations (Tables 6, 7) and main results (Tables 4, 5).

4.5 Ablation studies

To better understand the contribution of individual architectural components, we conducted a series of ablation experiments on both our internal dataset and the external M5 benchmark. Each variant isolates a single design choice, such as expert activation function, gating mechanism, and number of experts, while keeping all the other factors constant, such as training schedule, optimizer, loss function and data splits. This allows us to quantify the trade-offs between computational cost, convergence stability and prediction performance.

Types of experts. The first ablation examines whether SwiGLU-based experts justify their additional computational cost compared to simpler MLP layers using GELU. SwiGLUs introduce a multiplicative term that improves gradient flow and representation sparsity, helping experts to specialize in different demand regimes better. As shown in Tables 6 and 7, replacing SwiGLU with GELU experts slightly increases both WAPE and MASE, suggesting that simpler experts are slightly worse, especially when models are trained for longer periods. This confirms that the nonlinear gating in SwiGLU contributes meaningfully to expert diversity and effective generalization.

Gating method. While our simple straight-through (ST) estimator is sparse, choosing only one expert, in the backward pass it's approximately dense because it goes through every expert, even though the results are dropped. This design achieves the best of both worlds: sparse expert activation for inference efficiency and dense gradient updates for stable training. To assess the impact of this mechanism, we compared it against a soft-gating variant, where routing probabilities are used directly as continuous mixture weights. As shown in both datasets, soft-gating results in a measurable degradation in performance: roughly +0.05 WRMSSE on M5 and +2-3% WAPE on internal sample. This suggests that enforcing a discrete expert selection is beneficial: it helps each expert specialize and prevents mode collapse, where all experts converge to similar behaviors. In other words, the ST gating acts as a structural regularizer, encouraging diversity.

Number of experts. We further examined how model capacity and depth distribution affect performance. Our baseline uses four experts per layer across two Switch encoder layers, chosen to balance the model prediction performance and compute cost. We then compared this baseline with a shallower configuration that varies the number of experts within a single layer:

- Medium (Shallow) with 1 layer and 8 experts,
- Large (Shallow) with 1 layer and 32 experts.

Results in Table 6 show that scaling up the number of experts without depth degrades the model performance. Specifically, in Large (Shallow) variant, the model performs worse than the baseline despite having 8x more experts. We attribute this to expert underutilization: with too many experts and only one routing stage, the model struggles to learn meaningful specialization, leading to overfragmented token assignments. Conversely, a smaller number of experts distributed across multiple layers encourage hierarchical specialization, where earlier layers capture broader demand dynamics, and later layers focus on fine-grained adjustments. These findings highlight that depth and sparsity must be optimized simultaneously, not scaled independently. Increas-

Table 2: *Probabilistic (NLL) Objective.*

Model	WAPE	MASE
PatchTST	121.13%	1.2976
TFT	*80.23%*	*0.8595*
DeepAR	79.27%	0.8492
Switch-Hurdle (ours)	**74.70%**	**0.8108**

Table 3: *Point-Wise (MAE) Objective.*

Model	WAPE	MASE
PatchTST	92.82%	0.9944
DeepAR	66.39%	0.7112
Time-MoE (Zero-Shot)†	84.77%	1.0438
TFT	61.92%	0.6634
Switch-Hurdle (ours)	**56.97%**	**0.6184**

Deep learning benchmark metrics on the sample dataset. **Left Table** (Probabilistic Objective): Models trained with the Hurdle Loss ($\mathcal{L}_{\text{Hurdle}}$) and load balancing ($\mathcal{L}_{\text{balance}}$). **Right Table** (Point-Wise Objective): Models trained with the hybrid loss ($\mathcal{L}_{\text{Total}}$), optimized for deterministic accuracy.

†Time-MoE results are obtained by applying the released Time-MoE model on our internal dataset and converting its probabilistic output to point forecasts via the predictive mean.

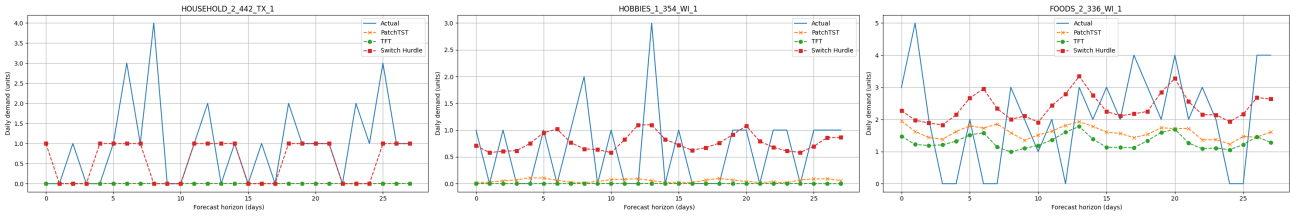

Figure 3: Qualitative comparison of 28-day forecasts on three representative M5 series: `HOUSEHOLD_2_442_TX_1`, `HOBBIES_1_354_WI_1`, and `FOODS_2_336_WI_1`. For each series we plot the actual daily demand (blue) together with predictions from PatchTST, TFT, and Switch-Hurdle. PatchTST and TFT tend to produce nearly flat forecasts and under-react to intermittent spikes, while Switch-Hurdle better tracks both the occurrence and magnitude of spikes and remains close to zero in no-demand periods.

Table 4: *Benchmark metrics on the M5 dataset.* The table presents the performance of leading Transformer-based forecasting models under both probabilistic (WRMSSE) and point-wise (MASE) evaluation. The Switch-Hurdle Transformer achieves the lowest WRMSSE across all methods, demonstrating improved robustness and distributional calibration on the challenging, hierarchical M5 dataset.

Model	WRMSSE	RMSE	MASE
PatchTST	1.0393	**2.4562**	0.9471
DeepAR	0.7895	2.9534	0.9087
TFT	0.6932	2.4686	**0.8983**
TSMixer	0.6403	-	-
Switch-Hurdle (ours)	**0.6307**	2.4744	0.8992

Table 5: *Benchmark metrics on the full internal dataset.* This comparison highlights the model's performance in a production-scale retail setting with rich feature space and varied demand regimes. The Switch-Hurdle Transformer achieves the lowest WAPE and nearly the best MASE, outperforming TFT and DeepAR while maintaining computational efficiency and stability during training.

Model	WAPE	MASE
PatchTST	81.22%	0.8478
DeepAR	64.86%	0.6770
TFT	55.60%	**0.5803**
Switch-Hurdle (ours)	**53.99%**	0.5865

ing the number of experts beyond a certain threshold offers little gain unless accompanied by deeper routing hierarchies.

The utilization patterns in Figure 2 align with the accuracy gains from hard gating (Table 7) and SwiGLU experts (Tables 6, 7).

The ablation findings explain better why the Switch-Hurdle Transformer consistently surpasses other architectures, including TFT, across both internal and external datasets (see Tables 4 and 5). By combining SwiGLU-based experts with hard Top-1 routing, the model achieves a balanced capacity allocation across demand patterns, improving its robustness to intermittent and low-signal periods that characterize retail data. Meanwhile, the compact two-layer design ensures that computational costs remain practical for large-scale deployment. Overall, these studies validate that the model's performance stems not merely from increased capacity, but from architectural choices that align the biases of sparse expert specialization with the statistical structure of retail demand.

5 Conclusion and Future Work

This work introduced Switch-Hurdle Transformer, a novel encoder-decoder architecture for large-scale retail demand forecasting that combines dense-to-sparse expert routing with probabilistic hurdle modeling. This model directly

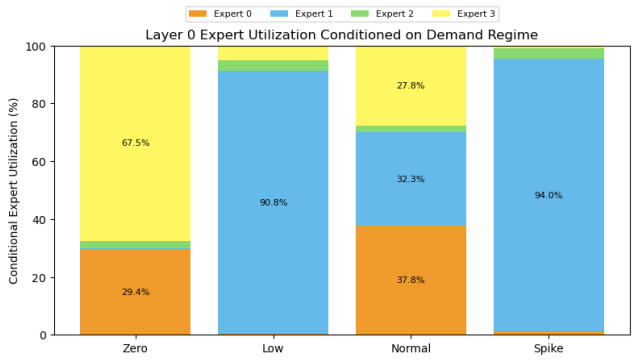

Figure 4: *Layer 0 expert specialization conditioned on demand regime.* Each bar shows $P(e \mid \text{regime})$ and is normalized to 100% per regime (Zero, Low, Normal, Spike). Values are normalized per regime to correct for minor measurement drift.

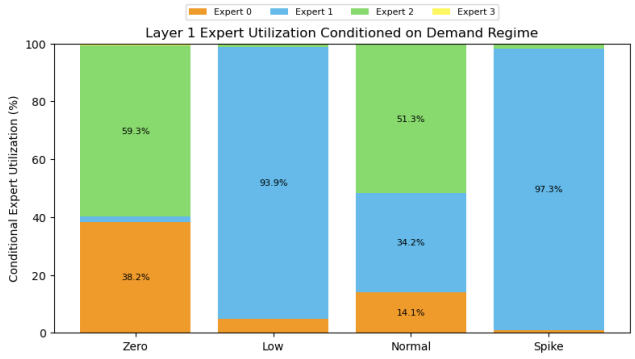

Figure 5: *Layer 1 expert specialization conditioned on demand regime.* Normalization as in Figure 4. This pattern supports the claim that Top-1 STE yields sparse forward routing with dense gradient updates, promoting stable yet differentiated experts across layers.

addresses two challenges in industrial forecasting: (i) the heavy-tailed, zero inflated type of data in demand, and (ii) the need for scalable and interpretable architectures. Our encoder leverages a Mixture-of-Experts (MoE) design with a Top-1 routing and SwiGLU experts, enabling sparse specialization without compromising gradient stability. The decoder's autoregressive hurdle head explicitly models zero-demand probability and positive demand magnitude through a Negative-Binomial distribution, producing probabilistic forecasts. Together, these components deliver state-of-the-art performance on both M5 benchmark and a large proprietary dataset, outperforming established Transformer variants such as TFT and PatchTST. Ablation studies further highlight how structured sparsity, discrete routing, and balanced depth jointly improve generalization and training efficiency. These findings not only validate our architectural design but also shed a light on how MoE principles can be systematically applied to practical, data-sparse domains, like retail demand.

Table 6: *Ablation results on the internal sample dataset.* Each variant isolates a specific design factor in the encoder. Results confirm that SwiGLU experts and ST Top-1 gating achieve the best trade-off between accuracy and efficiency, while excessively shallow or wide configurations degrade performance. Lower WAPE and MASE indicate better forecasting accuracy.

Model	WAPE	MASE
Baseline	56.97%	0.6184
GELU Experts	58.36%	0.6298
Soft-gating	59.23%	0.6413
Medium (Shallow)	58.02%	0.6298
Large (Shallow)	59.03%	0.6407

Table 7: *Ablation metrics on the M5 dataset.* Experiments on the large-scale M5 benchmark confirm the trends observed on internal data: SwiGLU experts and discrete Top-1 gating deliver better performance, while soft-gating or simpler MLP experts reduce precision. Metrics are reported as WRMSSE (lower is better) and MASE.

Model	WRMSSE	MASE
Baseline	0.6307	0.8627
GELU Experts	0.6401	0.8921
Soft-gating	0.6823	0.9173

Future Work. There are several promising directions for extending this work, including:

- *Adaptive Expert Specialization*: future research could explore dynamic expert assignment based on demand clustering, allowing the routing mechanism to evolve with market seasonality or promotions.
- *Cross-Domain Pretraining*: similar to recent time-series foundation models, pretraining the Switch-Hurdle Transformer across multiple domains, such as retail, logistics or energy, may enhance zero-shot generalization and calibration across heterogeneous distributions.

In summary, the Switch-Hurdle Transformer demonstrates that sparse specialization and probabilistic reasoning can coexist within a single forecasting framework, yielding a model that is accurate, interpretable, and deployable at industrial scale. We believe this work provides a foundational step toward a new generation of demand forecasting systems where neural architectures not only predict outcomes but also reveal the structure of the underlying economic processes they model.

Acknowledgments

We thank Fabian Muşat for implementing the initial version of Switch-Hurdle, and our colleagues in Data & AI for helpful discussions. We also thank the reviewers for their feedback.

References

Caetano, R.; Oliveira, J. M.; and Ramos, P. 2025. Transformer-Based Models for Probabilistic Time Series Forecasting with Explanatory Variables. *Mathematics*, 13(5): 814.

Fedus, W.; Zoph, B.; and Shazeer, N. 2022. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120): 1–39.

Garza, A.; Challu, C.; and Mergenthaler-Canseco, M. 2023. TimeGPT-1. *arXiv preprint arXiv:2310.03589*.

Goswami, M.; Szafer, K.; Choudhry, A.; Cai, Y.; Li, S.; and Dubrawski, A. 2024. MOMENT: A Family of Open Time-series Foundation Models. In *International Conference on Machine Learning*.

Jia, F.; Wang, K.; Zheng, Y.; Cao, D.; and Liu, Y. 2024. Gpt4mts: Prompt-based large language model for multimodal time-series forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 23343–23351.

Jin, M.; Wang, S.; Ma, L.; Chu, Z.; Zhang, J. Y.; Shi, X.; Chen, P.-Y.; Liang, Y.; Li, Y.-F.; Pan, S.; et al. 2024. Time-LLM: Time Series Forecasting by Reprogramming Large Language Models. In *The Twelfth International Conference on Learning Representations*.

Liu, X.; Liu, J.; Woo, G.; Aksu, T.; Liang, Y.; Zimmermann, R.; Liu, C.; Li, J.; Savarese, S.; Xiong, C.; et al. 2025. Moirai-MoE: Empowering Time Series Foundation Models with Sparse Mixture of Experts. In *Forty-second International Conference on Machine Learning*.

Nie, X.; Cao, S.; Miao, X.; Ma, L.; Xue, J.; Miao, Y.; Yang, Z.; Yang, Z.; and CUI, B. 2022. Dense-to-Sparse Gate for Mixture-of-Experts.

Nie, Y.; Nguyen, N. H.; Sinthong, P.; and Kalagnanam, J. 2023. A Time Series is Worth 64 Words: Long-term Forecasting with Transformers. In *The Eleventh International Conference on Learning Representations*.

Oliveira, J. M.; and Ramos, P. 2024. Evaluating the effectiveness of time series transformers for demand forecasting in retail. *Mathematics*, 12(17): 2728.

Puigcerver, J.; Ruiz, C. R.; Mustafa, B.; and Houlsby, N. 2024. From Sparse to Soft Mixtures of Experts. In *The Twelfth International Conference on Learning Representations*.

Shi, X.; Wang, S.; Nie, Y.; Li, D.; Ye, Z.; Wen, Q.; and Jin, M. 2025. Time-MoE: Billion-Scale Time Series Foundation Models with Mixture of Experts. In *The Thirteenth International Conference on Learning Representations*.

Suresh, V. 2025. Benchmarking Transformer Variants for Hour-Ahead PV Forecasting: PatchTST with Adaptive Conformal Inference. *Energies*, 18(18).

Wang, M.; Yang, J.; Yang, B.; Li, H.; Gong, T.; Yang, B.; and Cui, J. 2025. Towards Lightweight Time Series Forecasting: a Patch-wise Transformer with Weak Data Enriching. *CoRR*.

Wu, H.; Xu, J.; Wang, J.; and Long, M. 2021. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. *Advances in neural information processing systems*, 34: 22419–22430.