Mixture of Experts for Time Series Foundation Models

Xu Liu[1,2]*, **Juncheng Liu**[1], **Gerald Woo**[1*], **Taha Aksu**[1], **Chenghao Liu**[1†]
Silvio Savarese[1], **Caiming Xiong**[1], **Doyen Sahoo**[1]
[1]Salesforce AI Research, [2]National University of Singapore

Abstract

Time series foundation models have shown exceptional zero-shot forecasting capabilities. However, achieving effectively unified training on time series remains an open challenge. Existing approaches like MOIRAI pursues unified training by employing multiple input/output projection layers, each tailored to handle time series at a specific frequency. We identify two major drawbacks to this human-imposed frequency-level model specialization: (1) Frequency is not a reliable indicator of the underlying patterns in time series. (2) The Non-stationarity of time series leads to varied distributions even within a short context window of a single time series. Frequency-level specialization is too coarse-grained to capture this level of diversity. To address these limitations, this paper introduces MOIRAI-MOE, using a single input/output projection layer while delegating the modeling of diverse time series patterns to the sparse mixture of experts (MoE) within Transformers. With these designs, MOIRAI-MOE reduces reliance on human-defined heuristics and enables automatic token-level specialization. Extensive experiments on 39 datasets demonstrate the superiority of MOIRAI-MOE over existing foundation models in both in-distribution and zero-shot scenarios.

1 Introduction

Time series forecasting is experiencing a major shift. The traditional method of building individual models for each dataset is giving way to the concept of universal forecasting [23]. In this approach, a pretrained model is capable of being applied across diverse downstream tasks in a zero-shot manner, regardless of variations in domain, frequency, dimensionality, context, or prediction length.

To succeed in zero-shot forecasting, time series foundation models are pretrained on data spanning multiple sources. However, time series data is inherently heterogeneous, posing significant challenges for *unified time series training*. Existing solutions such as UniTime [14] utilize language prompts to achieve model specialization at the dataset level. MOIRAI [23] introduces a finer-grained categorization based on time series frequency. They employ multiple input/output projection layers with each tailored to a specific frequency, thereby enabling frequency-level specialization.

However, we argue that *human-imposed frequency-level specialization lacks generalizability and introduces several limitations*. (1) Frequency is not always a reliable indicator of the true structure of time series. As shown in Figure 1, time series with different frequencies can exhibit similar patterns, while those with the same frequency may display diverse and unrelated patterns. This human-imposed mismatch between frequency and pattern undermines the efficacy of model specialization, resulting in inferior performance. (2) Furthermore, real-world time series are inherently non-stationary [15], displaying varied distributions even within a short context window of a single time series. Clearly, frequency-level specialization is too coarse-grained to capture this level of diversity, underscoring the need for more fine-grained modeling approaches.

*Work done during internship/industrial PhD at Salesforce AI Research.
†Corresponding author: chenghao.liu@salesforce.com. Extended paper: https://arxiv.org/abs/2410.10469

NeurIPS 2024 Workshop on Time Series in the Age of Large Models.

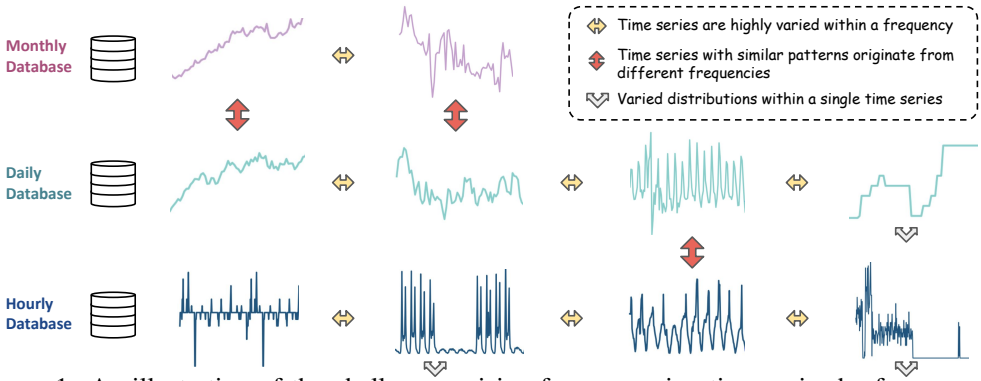

Figure 1: An illustration of the challenges arising from grouping time series by frequency and imposing frequency-level model specialization: the diversity of patterns within the same frequency group, the similarity of patterns across different frequencies, and the variability of distributions within a single time series. The examples are derived from **real time series** in Monash [8].

To address the aforementioned issues, this paper introduces **MOIRAI-MOE**, an innovative solution for effective time series unified training, inspired by recent developments of Sparse Mixture of Experts (MoE) Transformers [13, 7, 3]. The core idea of MOIRAI-MOE is to utilize a single input/output projection layer while delegateing the modeling of diverse time series patterns to the sparse specialized experts in Transformer layers. With these designs, specialization of MOIRAI-MOE is achieved in a data-driven manner and operates at the token level. Moreover, this study introduces a new function that leverages cluster centroids derived from a pretrained model to guide expert allocations. Our contributions are summarized as follows:

- We propose MOIRAI-MOE, the first mixture-of-experts time series foundation model, achieving token-level model specialization in a data-driven manner. We introduce a new expert gating function for accurate expert assignments and improved performance.
- Extensive experiments on 39 datasets demonstrate the superiority of MOIRAI-MOE over existing foundation models in both in-distribution and zero-shot scenarios.

2 Method

In this section, we introduce MOIRAI-MOE, a mixture-of-experts model built upon the time series foundation model MOIRAI [23]. Figure 2 presents a comparison. While MOIRAI-MOE inherits many of the strengths of MOIRAI, it significantly improves upon it by: rather than using multi input/output projection layers to model time series with different frequencies, MOIRAI-MOE employs a single projection layer while delegating the task of capturing diverse time series patterns to the mixture of experts in the Transformer. In addition, MOIRAI-MOE proposes a novel gating function that leverages knowledge from a pretrained model, and adopts a decoder-only training objective to improve training efficiency by enabling parallel learning of various context lengths in a single model update.

2.1 Time series token construction

By aggregating adjacent time series data into patches, patching techniques [17, 5, 14, 23] effectively capture local semantic information and significantly reduce computational overhead when processing long inputs. Given a time series with length S, we segment it into non-overlapping patches of size P, resulting in a sequence of patches $\boldsymbol{x} \in \mathbb{R}^{N \times P}$, where $N = \lceil \frac{S}{P} \rceil$. We then normalize the patches to mitigate distribution shift issues [15, 25]. In a decoder-only (autoregressive) model, where each patch predicts its succeeding patch, applying a causal normalizer to each patch is the most effective way to achieve accurate normalization. However, this approach generates N subsequences with different lengths, diminishing the parallel training that decoder-only models typically offer. To address this, we introduce the masking ratio r as a hyperparameter, which specifies the portion of the entire sequence used exclusively for robust normalizer calculation, without contributing to the prediction loss. Finally, we forward the patches through a single projection layer to generate time series tokens $\boldsymbol{x} \in \mathbb{R}^{N \times D}$, where D is the dimension of the Transformers. This layer is implemented as a residual multi-layer perceptron to enhance representation capacity [4].

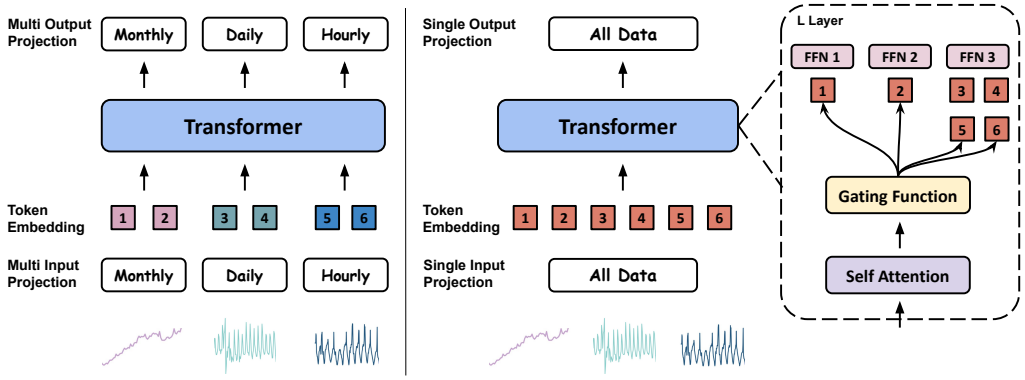

Figure 2: Comparison of MOIRAI (left) and MOIRAI-MOE (right).

2.2 Mixture of experts for transformers

A decoder-only Transformer [6] is constructed by stacking L layers of Transformer blocks. We establish the mixture of experts by replacing each feed-forward network (FFN) with a MoE layer, which is composed of M expert networks $\{E_1, \ldots, E_M\}$ and a gating function G. Only a subset of experts is activated for each token, allowing experts to specialize in distinct patterns of time series and ensuring computational efficiency. The output of the MoE layer is computed as:

$$\sum_{i=1}^{M} G(\boldsymbol{x})_i \cdot E_i(\boldsymbol{x}) \tag{1}$$

where $E_i(\boldsymbol{x})$ is the output of the i-th expert network, and $G(\boldsymbol{x})_i$ is the i-th token-to-expert affinity score generated by the gating function. In this work, we propose a new gating mechanism that leverages cluster centroids derived from the token representations of a pretrained model to guide expert allocations. Specifically, we utilize the self-attention output representations of a pretrained model (in our case, we use the MOIRAI model) and apply k-means clustering to generate clusters. The number of clusters is set to match the total number of experts. During MoE training, each token computes the Euclidean distance to each cluster centroid $\boldsymbol{C} \in \mathbb{R}^{M \times D}$, and these distances serve as token-to-expert affinity scores for expert assignments:

$$G(\boldsymbol{x}) = \text{Softmax}(\text{TopK}(\text{Euclidean}(\boldsymbol{x}, \boldsymbol{C}))) \tag{2}$$

2.3 Training objective

Let $\boldsymbol{x}_{t-l+1:t} = \{\boldsymbol{x}_{t-l+1}, \ldots, \boldsymbol{x}_t\}$ denote the context window of length l for a token at position t. In this study, to facilitate both point and probabilistic forecasting, our goal is formulated as forecasting the predictive distribution of the next token $p(\boldsymbol{x}_{t+1}|\phi)$ by predicting the mixture distribution parameters $\hat{\phi}$ [23]. These parameters are derived from the output tokens of the Transformer, followed by a single output projection layer. The following negative log-likelihood is minimized during training:

$$\mathcal{L}_{\text{pred}} = -\log p(\boldsymbol{x}_{t+1}|\,\hat{\phi}),\ \hat{\phi} = f_{\boldsymbol{\theta}}(\boldsymbol{x}_{t-l+1:t}) \tag{3}$$

3 Results

To ensure a fair comparison with MOIRAI in terms of activated parameters, we configure the number of activated experts as $K = 2$ for MOIRAI-MOE, resulting in 11M/86M activated parameters per token for MOIRAI-MOE$_\text{S}$/MOIRAI-MOE$_\text{B}$, closely matching the dense model MOIRAI$_\text{S}$/MOIRAI$_\text{B}$ that contains 14M/91M activated parameters. The total number of experts M is set to 32, yielding total parameter sizes of 117M for MOIRAI-MOE$_\text{S}$ and 935M for MOIRAI-MOE$_\text{B}$.

We begin with an in-distribution evaluation using a total of 29 datasets from the Monash benchmark [8]. Their training set are included in LOTSA [23], holding out the test set which we now use for assessments. The evaluation results in Figure 3 show that MOIRAI-MOE beats all competitors. In

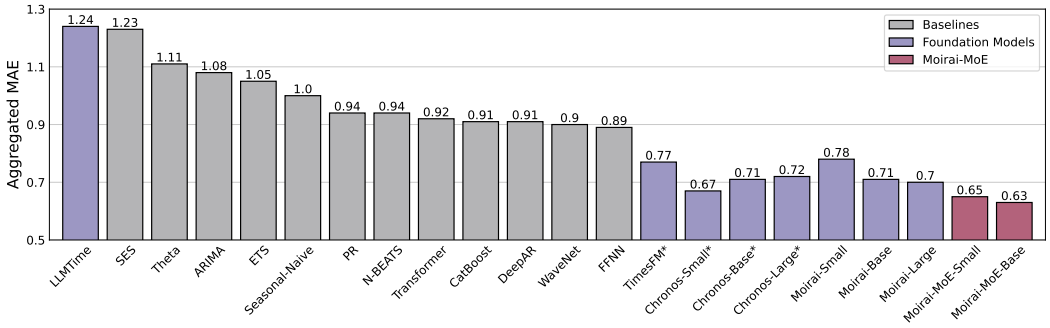

Figure 3: In-distribution forecasting evaluation. We use asterisks (*) to mark the methods that used the evaluation datasets here in their pretraining corpora. Values are normalized by seasonal naive, followed by geometric mean.

particular, MOIRAI-MOE$_S$ drastically surpasses its dense counterpart MOIRAI$_S$ by 17%, and also outperforms the larger models MOIRAI$_B$ and MOIRAI$_L$ by 8% and 7%, respectively. Compared to the foundation model Chronos, which MOIRAI could not surpass, MOIRAI-MOE successfully bridges the gap and delivers superior results with up to 65× fewer activated parameters. Next, we conduct an out-of-distribution evaluation on 10 datasets not included in LOTSA. To establish a comprehensive comparison, we report results for both probabilistic and point forecasting, using continuous ranked probability score (CRPS) and mean absolute scaled error (MASE) as evaluation metrics. The results are presented in Table 1. MOIRAI-MOE$_B$ achieves the best zero-shot performance, even outperforming TimesFM and Chronos, which include partial evaluation data in their pretraining corpora. When compared to all sizes of MOIRAI, MOIRAI-MOE$_S$ delivers a 3%–14% improvement in CRPS and an 8%–16% improvement in MASE. These improvements are remarkable, considering that MOIRAI-MOE$_S$ has only 11M activated parameters – 28× fewer than MOIRAI$_L$.

Table 1: Zero-shot performance of probabilistic and point forecasting. We use asterisks (*) to mark the non-zero-shot datasets because they were used in the pretraining corpus of TimesFM and Chronos. The Average column is normalized by seasonal naive, followed by geometric mean. Best average results are highlighted in **red**, and second best results are in **blue**. Power: Turkey Power. Traffic: Istanbul Traffic. Weather: Jena Weather. BizITObs: BizITObs-L2C.

Method	Metric	Electricity	Solar	Power	ETT1	ETT2	Traffic	MDENSE	Walmart	Weather	BizITObs	Average
Seasonal Naive	CRPS	0.070	0.512	0.085	0.515	0.205	0.257	0.294	0.151	0.068	0.262	1.000
	MASE	0.881	1.203	0.906	1.778	1.390	1.137	1.669	1.236	0.782	0.986	1.000
TimesFM	CRPS	0.045*	0.456	0.037	0.280	0.113	0.131	0.070	0.067	0.042	0.080	**0.488**
	MASE	0.655*	1.391	0.851	1.700	1.644	0.678	0.702	0.735	0.440	0.310	0.689
Chronos$_S$	CRPS	0.043*	0.389*	0.038	0.360	0.097	0.124	0.087	0.079	0.089	0.087	0.543
	MASE	0.629*	1.193*	0.717	1.799	1.431	0.622	0.834	0.849	0.606	0.301	0.694
Chronos$_B$	CRPS	0.041*	0.341*	0.039	0.387	0.092	0.109	0.075	0.080	0.058	0.084	0.499
	MASE	0.617*	1.002*	0.722	1.898	1.265	0.553	0.712	0.849	0.583	0.301	**0.656**
Chronos$_L$	CRPS	0.041*	0.339*	0.038	0.404	0.091	0.117	0.075	0.073	0.062	0.084	0.500
	MASE	0.615*	0.987*	0.702	1.959	1.270	0.597	0.724	0.788	0.601	0.310	0.660
MOIRAI$_S$	CRPS	0.072	0.471	0.048	0.275	0.101	0.173	0.084	0.103	0.049	0.081	0.578
	MASE	0.981	1.465	0.948	1.701	1.417	0.990	0.836	1.048	0.521	0.301	0.798
MOIRAI$_B$	CRPS	0.055	0.419	0.040	0.301	0.095	0.116	0.104	0.093	0.041	0.078	0.520
	MASE	0.792	1.292	0.888	1.736	1.314	0.644	1.101	0.964	0.487	0.291	0.736
MOIRAI$_L$	CRPS	0.050	0.406	0.036	0.286	0.094	0.112	0.095	0.098	0.051	0.079	0.514
	MASE	0.751	1.237	0.870	1.750	1.436	0.631	0.957	1.007	0.515	0.285	0.729
MOIRAI-MOE$_S$	CRPS	0.046	0.429	0.036	0.288	0.093	0.108	0.071	0.090	0.056	0.081	0.497
	MASE	0.719	1.222	0.737	1.750	1.248	0.563	0.746	0.927	0.476	0.298	0.670
MOIRAI-MOE$_B$	CRPS	0.041	0.382	0.034	0.296	0.091	0.100	0.071	0.088	0.057	0.079	**0.478**
	MASE	0.638	1.161	0.725	1.748	1.247	0.510	0.721	0.918	0.509	0.290	**0.651**

4 Conclusion

In this work, we introduce MOIRAI-MOE, the first time series MoE foundation model. By utilizing token-level specialization in a data-driven approach, MOIRAI-MOE delivers significant performance improvements over its predecessor MOIRAI and other competitive time series foundation models.

References

[1] Abdul Fatir Ansari, Lorenzo Stella, Caner Turkmen, Xiyuan Zhang, Pedro Mercado, Huibin Shen, Oleksandr Shchur, Syama Sundar Rangapuram, Sebastian Pineda Arango, Shubham Kapoor, et al. Chronos: Learning the language of time series. *arXiv preprint arXiv:2403.07815*, 2024.

[2] Defu Cao, Furong Jia, Sercan O Arik, Tomas Pfister, Yixiang Zheng, Wen Ye, and Yan Liu. Tempo: Prompt-based generative pre-trained transformer for time series forecasting. In *International Conference on Learning Representations*, 2024.

[3] Damai Dai, Chengqi Deng, Chenggang Zhao, RX Xu, Huazuo Gao, Deli Chen, Jiashi Li, Wangding Zeng, Xingkai Yu, Y Wu, et al. Deepseekmoe: Towards ultimate expert specialization in mixture-of-experts language models. In *Association for Computational Linguistics*, 2024.

[4] Abhimanyu Das, Weihao Kong, Andrew Leach, Shaan Mathur, Rajat Sen, and Rose Yu. Long-term forecasting with tide: Time-series dense encoder. *Transactions on Machine Learning Research*, 2023.

[5] Abhimanyu Das, Weihao Kong, Rajat Sen, and Yichen Zhou. A decoder-only foundation model for time-series forecasting. In *International Conference on Machine Learning*, 2024.

[6] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

[7] William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, pages 1–39, 2022.

[8] Rakshitha Godahewa, Christoph Bergmeir, Geoffrey I Webb, Rob J Hyndman, and Pablo Montero-Manso. Monash time series forecasting archive. *arXiv preprint arXiv:2105.06643*, 2021.

[9] Mononito Goswami, Konrad Szafer, Arjun Choudhry, Yifu Cai, Shuo Li, and Artur Dubrawski. Moment: A family of open time-series foundation models. *arXiv preprint arXiv:2402.03885*, 2024.

[10] Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. Mixtral of experts. *arXiv preprint arXiv:2401.04088*, 2024.

[11] Jiawei Jiang, Chengkai Han, Wenjun Jiang, Wayne Xin Zhao, and Jingyuan Wang. Towards efficient and comprehensive urban spatial-temporal prediction: A unified library and performance benchmark. *arXiv e-prints*, pages arXiv–2304, 2023.

[12] Guokun Lai, Wei-Cheng Chang, Yiming Yang, and Hanxiao Liu. Modeling long-and short-term temporal patterns with deep neural networks. In *The 41st international ACM SIGIR conference on research & development in information retrieval*, pages 95–104, 2018.

[13] Dmitry Lepikhin, HyoukJoong Lee, Yuanzhong Xu, Dehao Chen, Orhan Firat, Yanping Huang, Maxim Krikun, Noam Shazeer, and Zhifeng Chen. Gshard: Scaling giant models with conditional computation and automatic sharding. In *International Conference on Learning Representations*, 2021.

[14] Xu Liu, Junfeng Hu, Yuan Li, Shizhe Diao, Yuxuan Liang, Bryan Hooi, and Roger Zimmermann. Unitime: A language-empowered unified model for cross-domain time series forecasting. In *Proceedings of the ACM on Web Conference 2024*, pages 4095–4106, 2024.

[15] Yong Liu, Haixu Wu, Jianmin Wang, and Mingsheng Long. Non-stationary transformers: Exploring the stationarity in time series forecasting. In *Advances in Neural Information Processing Systems*, pages 9881–9893, 2022.

[16] Yong Liu, Haoran Zhang, Chenyu Li, Xiangdong Huang, Jianmin Wang, and Mingsheng Long. Timer: Generative pre-trained transformers are large time series models. In *Forty-first International Conference on Machine Learning*, 2024.

[17] Yuqi Nie, Nam H Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. In *International Conference on Learning Representations*, 2023.

[18] Santosh Palaskar, Vijay Ekambaram, Arindam Jati, Neelamadhav Gantayat, Avirup Saha, Seema Nagar, Nam H Nguyen, Pankaj Dayama, Renuka Sindhgatta, Prateeti Mohapatra, et al. Automixer for improved multivariate time-series forecasting on business and it observability data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 22962–22968, 2024.

[19] Samyam Rajbhandari, Conglong Li, Zhewei Yao, Minjia Zhang, Reza Yazdani Aminabadi, Ammar Ahmad Awan, Jeff Rasley, and Yuxiong He. Deepspeed-moe: Advancing mixture-of-experts inference and training to power next-generation ai scale. In *International Conference on Machine Learning*, pages 18332–18346, 2022.

[20] Kashif Rasul, Arjun Ashok, Andrew Robert Williams, Arian Khorasani, George Adamopoulos, Rishika Bhagwatkar, Marin Biloš, Hena Ghonia, Nadhir Vincent Hassen, Anderson Schneider, et al. Lag-llama: Towards foundation models for time series forecasting. *arXiv preprint arXiv:2310.08278*, 2023.

[21] Artur Trindade. Electricityloaddiagrams20112014. UCI Machine Learning Repository, 2015. DOI: https://doi.org/10.24432/C58C86.

[22] Will Cukierski Walmart Competition Admin. Walmart recruiting - store sales forecasting, 2014.

[23] Gerald Woo, Chenghao Liu, Akshat Kumar, Caiming Xiong, Silvio Savarese, and Doyen Sahoo. Unified training of universal time series forecasting transformers. In *International Conference on Machine Learning*, 2024.

[24] Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. In *Advances in neural information processing systems*, pages 22419–22430, 2021.

[25] Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. Timesnet: Temporal 2d-variation modeling for general time series analysis. In *International Conference on Learning Representations*, 2023.

[26] Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Proceedings of the AAAI conference on artificial intelligence*, pages 11106–11115, 2021.

[27] Tong Zhu, Xiaoye Qu, Daize Dong, Jiacheng Ruan, Jingqi Tong, Conghui He, and Yu Cheng. Llama-moe: Building mixture-of-experts from llama with continual pre-training. *arXiv preprint arXiv:2406.16554*, 2024.

A Related Work

Foundation Models for Time Series Forecasting Time series foundation models serve as versatile zero-shot forecasting tools. A key challenge in training these models is accommodating the high diversity of time series data, underscoring the possible need for designing specialization modules. Current approaches like TEMPO [2] and UniTime [14] utilize language-based prompts to identify data sources, facilitating model specialization at the dataset level. MOIRAI [23] advances this by focusing on a time series meta feature – frequency. This method designs separate input/output projection layers for specific frequencies, allowing for frequency-specific specialization. Similarly, TimesFM [5] operates at this level of specialization by incorporating a frequency embedding dictionary to differentiate data. Some methods, like Chronos [1], Lag-LLaMA [20], Moment [9], and Timer [16], do not incorporate any specialization modules. Instead, they utilize the same architecture for all time series data, which can potentially increase the learning complexity and demand a large number of parameters to memorize the diverse input patterns. In this work, we propose to achieve automatic token-level specialization by using sparse mixture of experts, where diverse time series tokens are processed by specialized experts, while similar tokens share parameter space, thereby reducing learning complexity.

Sparse Mixture of Experts Mixture of experts (MoE) has emerged as an effective method for significantly scaling up model capacity while minimizing computation overhead in Large Language Models (LLMs) [7, 3, 27]. In this study, our motivation for using MoE is primarily centered on its capacity to enable token-level model specialization. A common approach for integrating MoE into Transformers involves replacing Feed-Forward Networks (FFNs) with MoE layers. An MoE layer consists of multiple expert networks and a gating function, where each expert shares the same structure as a standard FFN. The gating function is responsible for producing a gating vector that indicates the expert assignment. The assignment is usually sparse to maintain computational efficiency in the MoE layer, meaning that each token is generally processed by only one [7] or two [19, 10] experts.

B More experimental results

In the main results, we simultaneously enable the mixture of experts and switch the training objective from a masked encoder approach to a decoder-only approach. To ensure a more rigorous comparison, we conduct further experiments where only the learning objective is changed. Table 2 presents the Monash evaluation results using the small model, with the first and last rows representing MOIRAI$_S$ and MOIRAI-MOE$_S$, respectively. This outcome suggests that altering the learning objective alone yields modest performance improvements, while the major gains stem from leveraging experts for automatic token-level specialization.

Table 2: Model variants performance on Monash.

Model Variant	Aggregated MAE
Multi Projection w/ Masked Encoder	0.78
Multi Projection w/ Decoder-Only	0.75
Single Projection & MoE w/ Decoder-Only	0.65

C More evaluation details

C.1 In-distribution forecasting

Following MOIRAI [23], we perform evaluations on 29 datasets from the Monash benchmark [8], including M1 Monthly, M3 Monthly, M3 Other, M4 Monthly, M4 Weekly, M4 Daily, M4 Hourly, Tourism Quarterly, Tourism Monthly, CIF 2016, Australian Electricity Demand, Bitcoin, Pedestrian Counts, Vehicle Trips, KDD Cup 2018, Australia Weather, NN5 Daily, NN5 Weekly, Carparts, FRED-MD, Traffic Hourly, Traffic Weekly, Rideshare, Hospital, COVID Deaths, Temperature Rain, Sunspot, Saugeen River Flow, and US Births. The full results of time series foundation models are shown in Table 3.

Table 3: Full MAE results of time series foundation models on the Monash Benchmark. The other baseline results can be found in [23].

Dataset	Seasonal Naive	LLMTime	TimesFM	MOIRAI$_{\text{Small}}$	MOIRAI$_{\text{Base}}$	MOIRAI$_{\text{Large}}$	Chronos$_{\text{Small}}$	Chronos$_{\text{Base}}$	Chronos$_{\text{Large}}$	MOIRAI-MOE$_{\text{Small}}$	MOIRAI-MOE$_{\text{Base}}$
M1 Monthly	2,011.96	2,562.84	1,673.60	2,082.26	2,068.63	1,983.18	1,797.78	1,637.68	1,627.11	1,992.49	1,811.94
M3 Monthly	788.95	877.97	653.57	713.41	658.17	664.03	644.38	622.27	619.79	646.07	617.31
M3 Other	375.13	300.30	207.23	263.54	198.62	202.41	196.59	191.80	205.93	185.89	179.92
M4 Monthly	700.24	728.27	580.20	597.60	592.09	584.36	592.85	598.46	584.78	569.25	544.08
M4 Weekly	347.99	518.44	285.89	339.76	328.08	301.52	264.56	252.26	248.89	302.65	278.37
M4 Daily	180.83	266.52	172.98	189.10	192.66	189.78	169.91	177.49	168.41	172.45	163.40
M4 Hourly	353.86	576.06	196.20	268.04	209.87	197.79	214.18	230.70	201.14	241.58	217.35
Tourism Quarterly	11,405.45	16,918.86	10,568.92	18,352.44	17,196.86	15,820.02	7,823.27	8,835.52	8,521.70	9,508.07	7,374.27
Tourism Monthly	1,980.21	5,608.61	2,422.01	3,569.85	2,862.06	2,688.55	2,465.10	2,358.67	2,140.73	2,523.66	2,268.31
CIF 2016	743,512.31	599,313.84	819,922.44	655,888.58	539,222.03	695,156.92	649,110.99	604,088.54	728,981.15	453,631.21	568,283.48
Aus. Elec. Demand	455.96	760.81	525.73	266.57	201.39	177.68	267.18	236.27	330.04	215.28	227.92
Bitcoin	7.78E+17	1.74E+18	7.78E+17	1.76E+18	1.62E+18	1.87E+18	2.34E+18	2.27E+18	1.88E+18	1.55E+18	1.90E+18
Pedestrian Counts	65.60	97.77	45.03	54.88	54.08	41.66	29.77	27.34	26.95	41.35	32.37
Vehicle Trips	32.48	31.48	21.93	24.46	23.17	21.85	19.38	19.25	19.19	21.62	21.65
KDD Cup 2018	47.09	42.72	40.86	39.81	38.66	39.09	38.60	42.36	38.83	40.21	40.86
Australia Weather	2.36	2.17	2.07	1.96	1.80	1.75	1.96	1.84	1.85	1.76	1.75
NN5 Daily	8.26	7.10	3.85	5.37	4.26	3.77	3.83	3.67	3.53	4.04	3.49
NN5 Weekly	16.71	15.76	15.09	15.07	16.42	15.30	15.03	15.12	15.09	15.74	15.29
Carparts	0.67	0.44	0.50	0.53	0.47	0.49	0.52	0.54	0.53	0.45	0.44
FRED-MD	5,385.53	2,804.64	2,237.63	2,568.48	2,679.29	2,792.55	938.46	1,036.67	863.99	1,651.76	2,273.61
Traffic Hourly	0.013	0.030	0.009	0.020	0.020	0.010	0.013	0.012	0.010	0.013	0.014
Traffic Weekly	1.19	1.15	1.06	1.17	1.14	1.13	1.14	1.12	1.12	1.13	1.14
Rideshare	1.60	6.28	1.36	1.35	1.39	1.29	1.27	1.33	1.30	1.26	1.26
Hospital	20.01	25.68	18.54	23.00	19.40	19.44	19.74	19.75	19.88	20.17	19.60
COVID Deaths	353.71	653.31	623.47	124.32	126.11	117.11	207.47	118.26	190.01	119.00	102.92
Temperature Rain	9.39	6.37	5.27	5.30	5.08	5.27	5.35	5.17	5.19	5.33	5.36
Sunspot	3.93	5.07	1.07	0.11	0.08	0.13	0.20	2.45	3.45	0.10	0.08
Saugeen River Flow	21.50	34.84	25.16	24.07	24.40	24.76	23.57	25.54	26.25	23.05	24.40
US Births	1,152.67	1,374.99	461.58	872.51	624.30	476.50	432.14	420.08	432.14	411.61	385.24

C.2 Zero-shot forecasting

We conduct zero-shot evaluations on the datasets listed in Table 4, which cover five domains and span frequencies ranging from minute-level to weekly. We use a non-overlapping rolling window approach, where the stride equals the prediction length. The test set consists of the last $h * r$ time steps, where h is the forecast horizon and r is the number of rolling evaluation windows. The validation set is defined as the last forecast horizon before the test set, while the training set includes all preceding data.

Table 4: Summary of datasets used in the zero-shot forecasting evaluations.

Dataset	Domain	Frequency	Prediction Length	Rolling Evaluations
Electricity [21]	Energy	H	24	7
Solar [12]	Energy	H	24	7
Turkey Power [3]	Energy	H	24	7
ETT1 [26]	Energy	D	30	3
ETT2 [26]	Energy	D	30	3
Istanbul Traffic [4]	Transport	H	24	7
M-DENSE [11]	Transport	D	30	3
Walmart [22]	Sales	W	8	4
Jena Weather [24]	Nature	10T	144	7
BizITObs-L2C [18]	Web/CloudOps	5T	48	20

[3] https://www.kaggle.com/datasets/dharanikra/electrical-power-demand-in-turkey

[4] https://www.kaggle.com/datasets/leonardo00/istanbul-traffic-index