# TiRex: Zero-Shot Forecasting Across Long and Short Horizons

**Andreas Auer** [1 2] **Patrick Podest** [2] **Daniel Klotz** [3] **Sebastian Böck** [1] **Günter Klambauer** [1 2] **Sepp Hochreiter** [1 2]

## Abstract

In-context learning, the ability of large language models to perform tasks using only examples provided in the prompt, has recently been adapted for time series forecasting. This paradigm enables zero-shot prediction, where past values serve as context for forecasting future values, making powerful forecasting tools accessible to non-experts and increasing the performance when training data are scarce. Most existing zero-shot forecasting approaches rely on transformer architectures, which, despite their success in language, often fall short of expectations in time series forecasting, where recurrent models like LSTMs frequently have the edge. Conversely, while LSTMs are well-suited for time series modeling due to their state-tracking capabilities, they lack strong in-context learning abilities. We introduce TiRex that closes this gap by leveraging xLSTM, an enhanced LSTM with competitive in-context learning skills. Unlike transformers, state-space models, or parallelizable RNNs such as RWKV, TiRex retains state-tracking, a critical property for long-horizon forecasting. To further facilitate its state-tracking ability, we propose a training-time masking strategy called CPM. TiRex sets a new state of the art in zero-shot time series forecasting on the HuggingFace benchmarks *GiftEval* and *Chronos-ZS*, outperforming significantly larger models including *TabPFN-TS* (Prior Labs), *Chronos Bolt* (Amazon), *TimesFM* (Google), and *Moirai* (Salesforce) across both short- and long-term forecasts.

## 1. Introduction

Recent research in time series forecasting has adopted in-context learning through large-scale pre-trained models, analogous to large language models (Woo et al., 2024; Ansari et al., 2024a; Das et al., 2024). These models enable zero-shot forecasting, allowing them to generalize to unseen datasets without parameter updates, akin to meta-learning (Hochreiter et al., 2001). This capability empowers practitioners without machine learning expertise to use advanced forecasting tools. More importantly, zero-shot forecasting significantly improves performance in data-scarce settings, where training task-specific models often fail to generalize. As a result, in-context learning models hold promise for broad adoption in domains such as energy, retail, or healthcare.

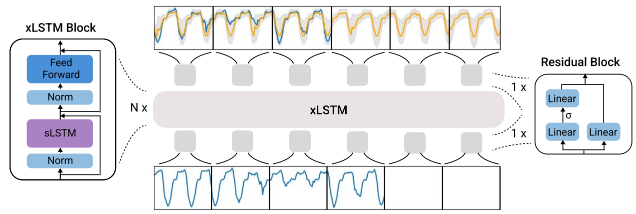



*Figure 1.* Architecture overview of TiRex, which comprises two main components: the xLSTM blocks and a residual block in the input and output layers. The illustrated forecast shows the forecasted series is in blue and the forecast of TiRex in orange. During inference, only the last three output windows are of interest.

Most pre-trained time series models are based on transformer architectures (Vaswani et al., 2017), which are well suited for in-context learning, but despite their success in language, often fall short of expectations in time series forecasting (e.g., Zeng et al., 2023). In contrast, LSTMs (Hochreiter, 1991; Hochreiter & Schmidhuber, 1997) have demonstrated strong results in time series forecasting due to their recurrence and effective state-tracking (e.g., Nearing et al., 2024). Therefore, LSTMs are more expressive than state-space models (SSMs), parallelizable RNNs like RWKV (Peng et al., 2023), and transformers (Merrill & Sabharwal, 2023; Merrill et al., 2024; Delétang et al., 2023). However, they lack strong in-context learning capabilities. To bridge this gap, we adopt xLSTM (Beck et al., 2024), a modern LSTM variant that incorporates architectural enhancements for scalability and improved generalization. In particular, xLSTM has demonstrated in-context learning performance comparable to that of transformer-based large

[1]NXAI GmbH, Linz, Austria [2]ELLIS Unit, LIT AI Lab, Institute for Machine Learning, JKU Linz, Austria [3]Interdisciplinary Transformation University Austria, Linz, Austria. Correspondence to: Andreas Auer <auer@ml.jku.at>.

*Proceedings of the 1st ICML Workshop on Foundation Models for Structured Data*, Vancouver, Canada. 2025. 

language models (Beck et al., 2025).

To fully unlock xLSTM's state-tracking abilities, we introduce Contiguous Patch Masking (CPM), a novel training-time masking strategy. CPM enhances xLSTM's ability to produce coherent long-horizon predictions by mitigating degradation common in autoregressive multi-step forecasting. While synthetic datasets are frequently used for pre-training forecasting models, the potential of data augmentation strategies remains largely untapped, unlike their established role in vision pre-training (Tian et al., 2020). To address this, we design and utilize a suite of augmentations.

**Our key contributions are:**

- TiRex: We present TiRex, a pre-trained time series model based on xLSTM, which sets a new state of the art in zero-shot forecasting. It achieves superior performance across standardized benchmarks, improving both short- and long-term forecasting accuracy.
- Contiguous Patch Masking (CPM): We propose a novel masking strategy that enhances state-tracking abilities, therefore enabling pre-trained time series models to produce reliable uncertainty estimates over long prediction horizons, effectively addressing autoregressive error accumulation.
- Data Augmentation Strategies: We introduce three augmentation techniques for time series model pre-training and demonstrate their effectiveness in enhancing the robustness and overall performance of TiRex.

### 1.1. Problem Setup: Zero-Shot Forecasting

Time series forecasting aims to predict future values of a time series based on its past values. Given a time series $(y_1, y_2, \ldots, y_T)$ with $y_t \in \mathbb{R}$ denoting the value at time $t$, the forecasting objective is to predict its future horizon $(y_{T+1}, \ldots, y_{T+h})$, where $h$ is the forecast horizon's length. Throughout the paper, we adopt Python-style array notation and denote a contiguous sequence of values by $\mathbf{y}_{1:T} := (y_t)_{t=1}^{T} = (y_1, y_2, \ldots, y_T)$. Probabilistic forecasting extends this setup by modeling the uncertainty inherent in most time series data. Instead of producing point estimates, the model learns to approximate the conditional distribution over future outcomes,

$$\mathcal{P}(\mathbf{y}_{T+1:T+h} \mid \mathbf{y}_{1:T}). \tag{1}$$

In the zero-shot forecasting setting, the prediction model is pre-trained on a corpus of time series datasets $C = \{D_1, D_2, \ldots, D_N\}$, where each $D_n, 1 \leq n \leq N$ is a time series dataset, e.g. a set of time series from a particular domain. At inference the model is applied directly to time series of new, unseen dataset, i.e., $\mathbf{y} \in D^{\text{test}}$ and $\mathbf{y} \notin \bigcup C$, without any fine-tuning or task-specific supervision.

## 2. TiRex

TiRex is a decoder-only xLSTM-based (Beck et al., 2024) architecture for time series forecasting. It consists of stacked xLSTM blocks between lightweight input and output layers (Figure 1). Following Beck et al. (2025), each *xLSTM block* comprises an RMSNorm, xLSTM module, and feed-forward network with residual connections. In contrast to Beck et al. (2025) TiRex utilizes the sLSTM module instead of the mLSTM module, because sLSTM supports real recurrence, enabling state-tracking (Beck et al., 2024). sLSTM remains efficient via optimized kernels (Pöppel et al., 2024). To handle varying scales, the *input layer* of TiRex applies $z$-score instance normalization (Kim et al., 2021). The time series is then segmented into windows, with length $m_{\text{in}}$ and mapped to the xLSTM input token via a shared two-layer residual block (He et al., 2016; Srivastava et al., 2015). A binary mask for missing values is concatenated to the time series values before the residual block, yielding an input transformation $\mathbb{R}^{2m_{\text{in}}} \rightarrow \mathbb{R}^{d}$. The *output layer* mirrors the input layer but outputs $|Q|$ quantiles for each time step: $\mathbb{R}^{d} \rightarrow \mathbb{R}^{m_{\text{out}} \times |Q|}$, with quantile levels ($Q = \{0.1, \ldots, 0.9\}$). More architectural details are provided in Appendix B. The model is trained using quantile loss, which is averaged across patches:

$$L = \frac{1}{|Q| \cdot m_{\text{out}}} \sum_{t=1}^{m_{\text{out}}} \sum_{q \in Q} \begin{cases} q(y_t - \hat{y}_t^q) & \text{if } \hat{y}_t^q \leq y_t \\ (1-q)(\hat{y}_t^q - y_t) & \text{else .} \end{cases} \tag{2}$$

**Multi-Patch Horizon Forecasts** When the forecast horizon $h$ exceeds the output patch length, multiple future patches must be predicted. We refer to this as multi-patch prediction. Existing pre-trained models (Das et al., 2024; Ansari et al., 2024b) typically address this via autoregressive generation, using point estimates (say, mean or median) of previous outputs as inputs for subsequent patches. However, this approach reinitializes the probabilistic forecast at each step, disrupting the propagation of uncertainty. In contrast, TiRex treats future inputs as missing values, allowing the internal memory to propagate both predictive state and uncertainty across patches. This results in more coherent probabilistic and overall better forecasts, as our quantitative and qualitative experiments show.

### 2.1. Contiguous Patch Masking (CPM)

To facilitate the stable multi-patch prediction capability of TiRex, we propose Contiguous Patch Masking (CPM). CPM randomly masks full and consecutive patches in pre-training. Such a masked patch is represented as "missing values" in the model input, hence corresponds to the structure of the input when multi-patch forecasts are used in the inference. The procedure is as follows: For each training sample, we first uniformly sample the amount of consecutive patches

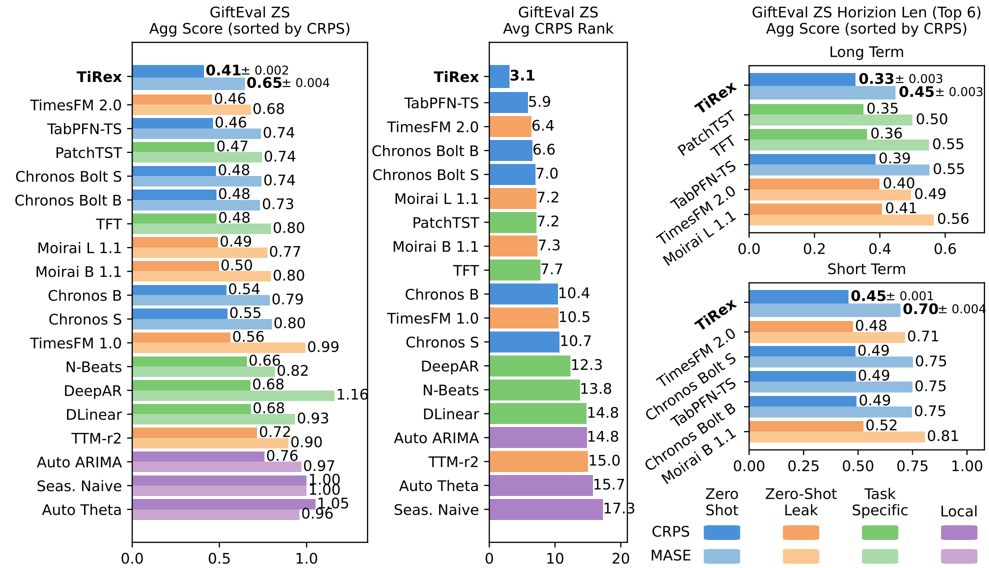



*Figure 2.* Results of the GiftEval-ZS benchmark. The aggregated scores of the overall benchmark and the short- and long-term subresults are shown. Additionally, the average rank in terms of CRPS, as in the public leaderboard, is presented. Lower values are better. "Zero-shot Leak" refers to models which are partly trained on the benchmark datasets (Overlap:: Moirai 19%, TimesFM 10%, TTM: 16%). We trained TiRex with 6 different seeds and report the observed standard deviation in the plot.

$c_{\text{mask}} \sim U(1, c_{\text{mask}}^{\max})$ and the masking probability $p_{\text{mask}} \sim U(0, p_{\text{mask}}^{\max})$. Afterwards, we mask the time series: For a time series of length $T$ we sample a binary mask of length $\lfloor \frac{T}{c_{\text{mask}} m_{\text{out}}} \rfloor$ with Bernoulli probability $p_{\text{mask}}$ and repeat each element $c_{\text{mask}} \cdot m_{\text{out}}$ so that the mask has a length of $T$ too. Note that when neighboring elements are masked, the actual maximum of consecutive masked patches can be greater than $c_{\text{mask}}^{\max}$. While CPM incorporates elements from BERT-style (Devlin et al., 2019) masked-modeling, our training remains more similar to the typical causal-style masking of decoder-only approaches (Radford et al., 2018) since the target is shifted and the information flow is uni-directional.

## 3. Data Augmentation

To facilitate more diverse time series patterns and enhance the model's exposure to a wider range of potentially relevant dynamics, we propose three augmentations for pre-training. The employed augmentations: (1) *Amplitude Modulation*, which introduces trends and changepoints in the scale of the time series. Formally, a time series $\mathbf{y}$ is transformed by $y'_t = y_t \cdot a_t$, where $a_t$ follows a linear trend, potentially with changepoints. (2) *Censor Augmentation*, which censors values within the time series at a random threshold. The augmented respective series is computed by $y'_t = \max/\min(y_t, c)$, The censor threshold $c$ is sampled by drawing a quantile uniformly from the empirical distribution of the signal. (3) *Spike Injection*, which adds short, periodic spike signals to the time series. The augmented time series is computed by $y'_t = y_t + s_t$, where $s_t$ represents the added spike signals. The shape of each spike is defined by a kernel, which can be a tophat, a radial basis function (RBF), or a linear kernel. The augmentations are applied with a probability of $0.5$ for amplitude modulation and censor augmentation and $0.05$ for spike augmentation. Appendix D provides a more detailed description of the augmentation procedures.

## 4. Experiments

TiRex is trained on $47.5$ million time series, combining real-world with synthetic data. We apply the proposed augmentations and CPM during training and utilize a context length of 2048 with a size of 32 for in- and output patches ($m_{\text{in/out}} = 32$). We evaluate TiRex on two standardized public benchmarks: Chronos-ZS benchmark (Ansari et al., 2024a) and GiftEval (Aksu et al., 2024), covering a total of 108 diverse evaluation settings across various frequencies and forecast horizons. As we are mostly interested in the zero-shot performance, we remove 16 of the 96 evaluations of GiftEval, which have an overlap with TiRex training data and denote this benchmark as *GiftEval-ZS benchmark*. Evaluation follows the official benchmark protocols using MASE and CRPS (via WQL), normalized by a seasonal naive baseline and aggregated via geometric mean and average rank. We compare against state-of-the-art zero-shot models, including Chronos (Bolt) (Ansari et al., 2024a;b), TimesFM (Das et al., 2024), and Moirai (Woo et al., 2024), TTM (Ekambaram et al., 2024) as well as non-zero-shot task-specific and local models. Training and evaluation details are provided in Appendix E, full results in Appendix F.

### 4.1. Zero-Shot Forecasting

TiRex consistently outperforms all competing methods across both short- and long-term forecasting tasks on the GiftEval-ZS benchmark (Figure 2). In terms of CRPS, TiRex achieves a score of $0.411$ (standard deviation over training with 6 seeds is $\pm 0.002$), notably surpassing the next best models. This performance gap is further reflected in the average rank, where TiRex shows a substantial lead over the second-best model, while the following models (TimesFM-2.0, TabPFN, and Chronos-Bolt) achieve very similar scores among themselves. Importantly, while other models tend to peak either in short- (e.g., Chronos-Bolt) or long-term (e.g., TabPFN-TS) forecasting, TiRex is the only model to excel simultaneously at both. Moreover, TiRex attains these results with significantly fewer parameters (35M) compared to Chronos-Bolt-Base (200M) and TimesFM-2.0 (500M). The advantage is most pronounced in long-term forecasting, where TiRex becomes the first zero-shot model to surpass the performance of PatchTST and TFT. Extended results, results for the Chronos-ZS benchmark, an efficiency analysis, and qualitative examples are presented in Appendix F.2.

### 4.2. Ablations

We conduct ablation studies to assess the impact of CPM, the proposed augmentations, and the xLSTM backbone — with extended results in Appendix F.3.

**Contiguous Patch Masking and Multi-Patch-Inference** We analyze the effectiveness of the proposed Contiguous Patch Masking (CPM) procedure by comparing three configurations: (1) standard decoder-only training with autoregressive inference — this setting is similar, e.g., to TimesFM; (2) training with naive multiple-patch procedure where we place the "rollout" always at the end of the sequence; and (3) training with CPM. Configurations (2) and (3) use the same inference procedure with masked tokens for multi-step forecasting, whereas (1) relies on patch-wise autoregressive decoding. Only CPM enables strong long-term performance without degrading short-term performance (see Table 1). These findings suggest that CPM is essential for improved performance under multi-patch prediction.

**Augmentation** To assess the impact of our augmentations, we trained our model without any augmentations. The results, detailed in Table 1, indicate that including the augmentations is beneficial, i.e., improves the performance of the model.

**Backbone** To assess the architectural choice of xLSTM with sLSTM modules, we replace it with mLSTM (Beck et al., 2024) and transformer blocks (Touvron et al., 2023) while keeping the patching and training procedure unchanged. In the Transformer variant, rotary positional embeddings (Su et al., 2024) are added to mitigate the absence of inherent positional information, due to their permutation-invariance property. Additionally, we ablate our overall architecture by comparing to a Chronos-Bolt architecture that we train with our training procedure, using the same datasets and augmentations as for TiRex. Table 1 summarizes the results. TiRex yields the best performance, especially on long-term forecasts. We hypothesize this is because its explicit state-tracking capabilities (Beck et al., 2024), which might facilitate uncertainty propagation and enable accurate modeling of periodic temporal structures over extended horizons. The comparison to a Chronos Bolt architecture, which is consistently outperformed by TiRex, highlights that our overall architecture is critical to achieve good performance.

*Table 1.* Ablation study of individual components (CRPS on GiftEval ZS). The top two rows report the mean and standard deviation of TiRex over 6 runs with different random seeds. For the ablation variants, results that degrade performance by more than $3\times$ the standard deviation relative to TiRex are underlined.

| | Benchmark | Gift-ZS (CRPS) | | |
|---|---|---|---|---|
| | | Overall | Long | Short |
| | TiRex | 0.411 | 0.325 | 0.455 |
| | $\pm$6 seeds | 0.002 | 0.003 | 0.001 |
| CPM | naive multi-patch | <u>0.424</u> | <u>0.335</u> | <u>0.475</u> |
| CPM | w/o multi-patch | <u>0.445</u> | <u>0.370</u> | <u>0.471</u> |
| | w/o Augment | <u>0.430</u> | <u>0.339</u> | <u>0.473</u> |
| Backbone | Transformer | <u>0.422</u> | <u>0.342</u> | <u>0.461</u> |
| Backbone | mLSTM | <u>0.457</u> | <u>0.430</u> | 0.456 |
| Backbone | Chronos Bolt B | <u>0.454</u> | <u>0.418</u> | 0.458 |

## 5. Conclusion

This work introduces TiRex, a pre-trained time series forecasting model based on xLSTM. To fully unlock the state-tracking capabilities of xLSTM, we further propose Contiguous Patch Masking, a training-time masking strategy tailored for in-context learning. Contiguous Patch Masking is a crucial component in our modeling pipeline, since it enables strong long-term forecasting performance without sacrificing short-term capabilities. TiRex establishes a new state-of-the-art in zero-shot forecasting, outperforming prior methods on both short- and long-term horizons across the Chronos-ZS and GiftEval benchmarks. Our ablation studies highlight the individual contributions of each component to overall performance.

## Acknowledgements

We thank Maximilian Beck and Korbinian Pöppel for the discussions and advice in regard to xLSTM. The ELLIS Unit Linz, the LIT AI Lab, and the Institute for Machine Learning are supported by the Federal State Upper Austria.

## References

Aksu, T., Woo, G., Liu, J., Liu, X., Liu, C., Savarese, S., Xiong, C., and Sahoo, D. GIFT-eval: A benchmark for general time series forecasting model evaluation. In *NeurIPS Workshop on Time Series in the Age of Large Models*, 2024.

Ansari, A. F., Stella, L., Turkmen, A. C., Zhang, X., Mercado, P., Shen, H., Shchur, O., Rangapuram, S. S., Arango, S. P., Kapoor, S., Zschiegner, J., Maddix, D. C., Wang, H., Mahoney, M. W., Torkkola, K., Wilson, A. G., Bohlke-Schneider, M., and Wang, B. Chronos: Learning the Language of Time Series. *Transactions on Machine Learning Research*, May 2024a. ISSN 2835-8856.

Ansari, A. F., Turkmen, C., Shchur, O., and Stella, L. Fast and accurate zero-shot forecasting with Chronos-Bolt and AutoGluon, December 2024b. URL https://aws.amazon.com/blogs/machine-learning/fast-and-accurate-zero-shot-forecasting-with-chronos-bolt-and-autogluon/.

Beck, M., Pöppel, K., Spanring, M., Auer, A., Prudnikova, O., Kopp, M. K., Klambauer, G., Brandstetter, J., and Hochreiter, S. xLSTM: Extended Long Short-Term Memory. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, November 2024.

Beck, M., Pöppel, K., Lippe, P., Kurle, R., Blies, P. M., Klambauer, G., Böck, S., and Hochreiter, S. xLSTM 7B: A Recurrent LLM for Fast and Efficient Inference, 2025.

Box, G. E. P. and Jenkins, G. M. Some Recent Advances in Forecasting and Control. *Journal of the Royal Statistical Society Series C*, 17(2):91–109, 1968.

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33: 1877–1901, 2020.

Chen, S.-A., Li, C.-L., Yoder, N., Arik, S. O., and Pfister, T. TSMixer: An All-MLP Architecture for Time Series Forecasting. *ArXiv*, 2303.06053, 2023.

Das, A., Kong, W., Sen, R., and Zhou, Y. A decoder-only foundation model for time-series forecasting. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 10148–10167. PMLR, July 2024.

Delétang, G., Ruoss, A., Grau-Moya, J., Genewein, T., Wenliang, L. K., Catt, E., Cundy, C., Hutter, M., Legg, S., Veness, J., and Ortega, P. A. Neural networks and the Chomsky hierarchy. In *International Conference on Learning Representations (ICLR)*, volume 11, 2023.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*, pp. 4171–4186, 2019.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *International Conference on Learning Representations*, October 2020.

Ekambaram, V., Jati, A., Dayama, P., Mukherjee, S., Nguyen, N. H., Gifford, W. M., Reddy, C., and Kalagnanam, J. Tiny Time Mixers (TTMs): Fast Pretrained Models for Enhanced Zero/Few-Shot Forecasting of Multivariate Time Series. *ArXiv*, 2401.03955, 2024.

Gardner, J. R., Pleiss, G., Bindel, D., Weinberger, K. Q., and Wilson, A. G. Gpytorch: Blackbox matrix-matrix gaussian process inference with gpu acceleration. In *Advances in Neural Information Processing Systems*, 2018.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016. doi: 10.1109/CVPR.2016.90.

Hochreiter, S. Untersuchungen zu dynamischen neuronalen Netzen. Diploma thesis, Institut für Informatik, Lehrstuhl Prof. Brauer, Technische Universität München, 1991.

Hochreiter, S. and Schmidhuber, J. Long short-term memory. *Neural Comput.*, 9(8):1735–1780, 1997.

Hochreiter, S., Younger, A. S., and Conwell, P. R. Learning to learn using gradient descent. In Dorffner, G., Bischof, H., and Hornik, K. (eds.), *Proc. Int. Conf. on Artificial Neural Networks (ICANN 2001)*, pp. 87–94. Springer, 2001.

Hollmann, N., Müller, S., Purucker, L., Krishnakumar, A., Körfer, M., Hoo, S. B., Schirrmeister, R. T., and Hutter, F. Accurate predictions on small data with a tabular foundation model. *Nature*, 637(8045):319–326, January 2025. ISSN 1476-4687. doi: 10.1038/s41586-024-08328-6.

Hoo, S. B., Müller, S., Salinas, D., and Hutter, F. The tabular foundation model tabpfn outperforms specialized time series forecasting models based on simple features. *ArXiv*, 2501.02945, 2025.

Hyndman, R., Koehler, A., Ord, K., and Snyder, R. *Forecasting with Exponential Smoothing*. Springer Series in Statistics. Springer, Berlin, Heidelberg, 2008. ISBN 978-3-540-71916-8 978-3-540-71918-2. doi: 10.1007/978-3-540-71918-2.

Kim, T., Kim, J., Tae, Y., Park, C., Choi, J.-H., and Choo, J. Reversible Instance Normalization for Accurate Time-Series Forecasting against Distribution Shift. In *International Conference on Learning Representations*, October 2021.

Kraus, M., Divo, F., Dhami, D. S., and Kersting, K. xlstm-mixer: Multivariate time series forecasting by mixing via scalar memories. *ArXiv*, 2410.16928, 2024.

Lim, B., Arık, S. Ö., Loeff, N., and Pfister, T. Temporal Fusion Transformers for interpretable multi-horizon time series forecasting. *International Journal of Forecasting*, 37(4):1748–1764, October 2021. ISSN 0169-2070. doi: 10.1016/j.ijforecast.2021.03.012.

Merrill, W. and Sabharwal, A. The parallelism tradeoff: Limitations of log-precision transformers. *Transactions of the Association for Computational Linguistics*, 11:531–545, 2023. doi: 10.1162/tacl_a_00562.

Merrill, W., Petty, J., and Sabharwal, A. The illusion of state in state-space models. *ArXiv*, 2404.08819, 2024.

Nearing, G., Cohen, D., Dube, V., Gauch, M., Gilon, O., Harrigan, S., Hassidim, A., Klotz, D., Kratzert, F., Metzger, A., et al. Global prediction of extreme floods in ungauged watersheds. *Nature*, 627(8004):559–563, 2024.

Nie, Y., Nguyen, N. H., Sinthong, P., and Kalagnanam, J. A Time Series is Worth 64 Words: Long-term Forecasting with Transformers. In *The Eleventh International Conference on Learning Representations*, September 2022.

Oreshkin, B. N., Carpov, D., Chapados, N., and Bengio, Y. N-BEATS: Neural basis expansion analysis for interpretable time series forecasting. In *International Conference on Learning Representations*, September 2019.

Peng, B., Alcaide, E., Anthony, Q., Albalak, A., Arcadinho, S., Biderman, S., Cao, H., Cheng, X., Chung, M., Grella, M., et al. Rwkv: Reinventing rnns for the transformer era. *ArXiv*, 2305.13048, 2023.

Pöppel, K., Beck, M., and Hochreiter, S. FlashRNN: I/O-Aware Optimization of Traditional RNNs on modern hardware. In *International Conference on Learning Representations*, October 2024.

Radford, A., Narasimhan, K., Salimans, T., Sutskever, I., et al. Improving language understanding by generative pre-training. 2018.

Salinas, D., Flunkert, V., Gasthaus, J., and Januschowski, T. DeepAR: Probabilistic forecasting with autoregressive recurrent networks. *International Journal of Forecasting*, 36(3):1181–1191, July 2020. ISSN 0169-2070. doi: 10.1016/j.ijforecast.2019.07.001.

Srivastava, R. K., Greff, K., and Schmidhuber, J. Training Very Deep Networks. *ArXiv*, 1507.06228, 2015.

Su, J., Ahmed, M., Lu, Y., Pan, S., Bo, W., and Liu, Y. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

Tian, Y., Sun, C., Poole, B., Krishnan, D., Schmid, C., and Isola, P. What makes for good views for contrastive learning? *Advances in neural information processing systems*, 33:6827–6839, 2020.

Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., Rodriguez, A., Joulin, A., Grave, E., and Lample, G. Llama: Open and efficient foundation language models. *ArXiv*, 2302.13971, 2023.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. Attention is All you Need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

Woo, G., Liu, C., Kumar, A., Xiong, C., Savarese, S., and Sahoo, D. Unified Training of Universal Time Series Forecasting Transformers. In *Forty-First International Conference on Machine Learning*, June 2024.

Zeng, A., Chen, M., Zhang, L., and Xu, Q. Are transformers effective for time series forecasting? In *Proceedings of the Thirty-Seventh AAAI Conference on Artificial Intelligence and Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence and Thirteenth Symposium on Educational Advances in Artificial Intelligence*, volume 37 of *AAAI'23/IAAI'23/EAAI'23*, pp. 11121–11128. AAAI Press, February 2023. ISBN 978-1-57735-880-0. doi: 10.1609/aaai.v37i9.26317.

Zhang, B. and Sennrich, R. Root mean square layer normalization. *Advances in Neural Information Processing Systems*, 32, 2019.

## A. Related Work

Statistical models such as ARIMA (Box & Jenkins, 1968) and exponential smoothing (Hyndman et al., 2008) are classical approaches in time series forecasting. In the last decades, however, neural network-based models have emerged as effective alternatives: Notable examples include DeepAR (Salinas et al., 2020), based on a LSTM with a mixture density head; N-BEATS (Oreshkin et al., 2019), the first approach that employed a deep block architecture; PatchTST (Nie et al., 2022), a patch-based attention approach; and TFT (Lim et al., 2021), which combines LSTM and transformer components. These models are trained on multiple time series from a single dataset and require retraining when applied to new tasks.

Currently, pre-trained time series models, with the capability of zero-shot generalization across datasets, predominantly adopt different transformer architectures. For instance, Chronos (Ansari et al., 2024a) and its successor Chronos-Bolt (Ansari et al., 2024b) use an encoder-decoder variant. Moirai (Woo et al., 2024) adopts an encoder-only design with a masked modeling objective, and TimesFM (Das et al., 2024) follows a decoder-only causal modeling strategy for autoregressive generation. TabPFN (Hollmann et al., 2025) and its adaptation to time series TabPFN-TS (Hoo et al., 2025) use a modified transformer encoder and pre-train only on synthetic data. A notable exception is TTM (Ekambaram et al., 2024), since it builds on the MLP-based TSMixer architecture (Chen et al., 2023).

The dominance of transformer architectures echoes their strong in-context learning capabilities, essentially for zero-shot forecasting, which are known from the language domain (Brown et al., 2020). However, despite their success in language, they often fall short of expectations in time series forecasting. For example, Zeng et al. (2023) show that DLinear, a simple linear model, can outperform transformers in multiple scenarios. Moreover, classical models like LSTMs are still widely used and remain competitive. While LSTMs are well-suited for time series modeling, they lack strong in-context learning abilities. Recent advancements in recurrent architectures — such as the xLSTM (Beck et al., 2025) — did close this gap. xLSTM has shown promising results in task-specific time series applications (Kraus et al., 2024), yet its potential for pre-trained, general-purpose time series modeling remains underexplored.

## B. TiRex Architecture

TiRex utilizes xLSTM as its backbone architecture, and adopts a decoder-only mode, which allows for efficient training. It stacks multiple xLSTM blocks between a lightweight input and output layer. The input layer preprocesses the time series via scaling and patching operations, producing tokens that are subsequently processed by the xLSTM blocks. The output tokens correspond to forecasted patches of the target series and are mapped back to the forecast horizon. For multi-patch forecasting, additional inputs are encoded as missing values. The individual components are described in detail below and an overview of the architecture is provided in Figure 1.

**xLSTM Block** TiRex adpots the block design proposed by Beck et al. (2025), but substitutes the mLSTM with a sLSTM module as the sequence mixing component. Both module options were introduced in the original publication, but only sLSTM allows for state-tracking (by trading it for reduced memory capacity Beck et al., 2024). Each block comprises an sLSTM module followed by a feed-forward network, with both components preceded by an RMSNorm (Zhang & Sennrich, 2019). Additionally, both the sLSTM module and feedforward submodules include residual skip connections. sLSTM supports real recurrence, to enable state-tracking, yet still is efficient in training and inference due to an optimized kernel architecture (Pöppel et al., 2024). TiRex stacks multiple of these blocks, depending on the model size. After the last block, an additional RMSNorm is applied. More details on the general xLSTM architecture are provided in Appendix C.

**Input/Output Layer and Loss** TiRex is designed to generalize across diverse time series domains, which often exhibit significant variation in scale. To ensure robustness, TiRex applies instance normalization to each time series (Kim et al., 2021). Specifically, $z$-score normalization is used. That is, $\tilde{\mathbf{y}}_{0:T} = \frac{\mathbf{y}_{0:T} - \bar{y}_{0:T}}{\sigma_{y_{0:T}}}$, where, $\bar{y}$ and, $\sigma_y$ denote the mean and standard deviation of the time series sample.

TiRex segments time series into non-overlapping windows, and maps each window to the input space of the xLSTM using a two-layer residual block (He et al., 2016; Srivastava et al., 2015). This patching mechanism is inspired by vision transformers (Dosovitskiy et al., 2020) and was adapted for time series by Nie et al. (2022) and Woo et al. (2024). It reduces the effective sequence length of the xLSTM blocks by a factor defined by the window size. To account for missing values, a binary mask indicating presence or absence is concatenated to the time series values before the residual block. Given an input window of size $m_{\text{in}}$ and xLSTM hidden dimension $d$, the patching block defines a mapping $\mathbb{R}^{2m_{\text{in}}} \to \mathbb{R}^d$. The same residual block is shared across all time windows.

Mirroring the input layer, the decoder's output tokens are transformed back to the dimensions of the output-patch window using a residual block and subsequently scaled back to the original target space. Hereby, the model outputs provides $|Q|$ quantile values for each time step of the output-patch window, rather than single-point predictions. Hence, the output block defines a mapping $\mathbb{R}^d \to \mathbb{R}^{m_{\text{out}} \times |Q|}$. Specifically, TiRex predict nine equidistant quantile levels, $Q = \{0.1, 0.2, \ldots, 0.9\}$. The model's parameters are optimized by minimizing the quantile loss. The loss is calculated for each output token, therefore, the loss does not distinguish context and forecast for a training sample, but implicitly "forecast after each input token". Formally, the loss for an output window, given the true value $y_t$ at time $t$ and its corresponding quantile predictions $\hat{y}_t^q$ for quantile level $q$ is computed as:

$$L = \frac{1}{|Q|\, m_{\text{out}}} \sum_{t=1}^{m_{\text{out}}} \sum_{q \in Q} \begin{cases} q\,(y_t - \hat{y}_t^q) & \text{if } \hat{y}_t^q \le y_t \\ (1-q)\,(\hat{y}_t^q - y_t) & \text{else} \end{cases} . \tag{3}$$

The losses of all output tokens of a training sample are averaged — missing values in the output window are ignored for the loss calculation.

## C. xLSTM

TiRex utilizes xLSTM (Beck et al., 2024) as its backbone architecture. xLSTM extends the classical LSTM (Hochreiter, 1991; Hochreiter & Schmidhuber, 1997) by incorporating modern design principles to improve its scalability, parallelization, and in-context modeling capabilities.

xLSTM introduces two cell types: the matrix LSTM (mLSTM) and the scalar LSTM (sLSTM). The mLSTM is designed to increase memory capacity through a matrix-based memory representation, and enables efficient parallel computation. In contrast, the sLSTM preserves a true recurrent pathway as in the original LSTM, enabling strong state-tracking capabilities (Beck et al., 2024). The recurrent pathway makes LSTM more expressive than State Space Models (SSMs), parallelizable RNNs like RWKV, and transformers (Merrill & Sabharwal, 2023; Merrill et al., 2024; Delétang et al., 2023). Figure 3 illustrates the respective expressivity hierarchies. Delétang et al. (2023) demonstrate on synthetic language tasks that this state-tracking capability of LSTMs yields empirical advantages. Beck et al. (2024) show that sLSTM retains these advantages. sLSTM employs a multi-head strategy along the recurrent pathway to improve efficiency.

TiRex exclusively employs sLSTM cells. Given a sequence of $T$ embedded input patches $\boldsymbol{X}_{1:T} = (\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_T) \in \mathbb{R}^{d \times T}$, the forward computation of an sLSTM cell for a given time step is defined as follows:

$$\boldsymbol{c}_t = \mathbf{f}_t \odot \boldsymbol{c}_{t-1} + \mathbf{i}_t \odot \boldsymbol{z}_t , \qquad \text{cell state} \tag{4}$$

$$\boldsymbol{n}_t = \mathbf{f}_t \odot \boldsymbol{n}_{t-1} + \mathbf{i}_t , \qquad \text{normalizer state} \tag{5}$$

$$\boldsymbol{h}_t = \mathbf{o}_t \odot \tilde{\boldsymbol{h}}_t , \qquad \tilde{\boldsymbol{h}}_t = \boldsymbol{c}_t \odot \boldsymbol{n}_t^{-1} \qquad \text{hidden state} \tag{6}$$

$$\boldsymbol{z}_t = \varphi(\tilde{\boldsymbol{z}}_t) , \qquad \tilde{\boldsymbol{z}}_t = \boldsymbol{W}_{\boldsymbol{z}}\, \boldsymbol{x}_t + \boldsymbol{R}_{\boldsymbol{z}}\, \boldsymbol{h}_{t-1} + \boldsymbol{b}_{\boldsymbol{z}} \qquad \text{cell input} \tag{7}$$

$$\mathbf{i}_t = \exp\left(\tilde{\mathbf{i}}_t\right) , \qquad \tilde{\mathbf{i}}_t = \boldsymbol{W}_{\mathbf{i}}\, \boldsymbol{x}_t + \boldsymbol{R}_{\mathbf{i}}\, \boldsymbol{h}_{t-1} + \boldsymbol{b}_{\mathbf{i}} \qquad \text{input gate} \tag{8}$$

$$\mathbf{f}_t = \exp\left(\tilde{\mathbf{f}}_t\right) , \qquad \tilde{\mathbf{f}}_t = \boldsymbol{W}_{\mathbf{f}}\, \boldsymbol{x}_t + \boldsymbol{R}_{\mathbf{f}}\, \boldsymbol{h}_{t-1} + \boldsymbol{b}_{\mathbf{f}} \qquad \text{forget gate} \tag{9}$$

$$\mathbf{o}_t = \sigma(\tilde{\mathbf{o}}_t) , \qquad \tilde{\mathbf{o}}_t = \boldsymbol{W}_{\mathbf{o}}\, \boldsymbol{x}_t + \boldsymbol{R}_{\mathbf{o}}\, \boldsymbol{h}_{t-1} + \boldsymbol{b}_{\mathbf{o}} \qquad \text{output gate,} \tag{10}$$

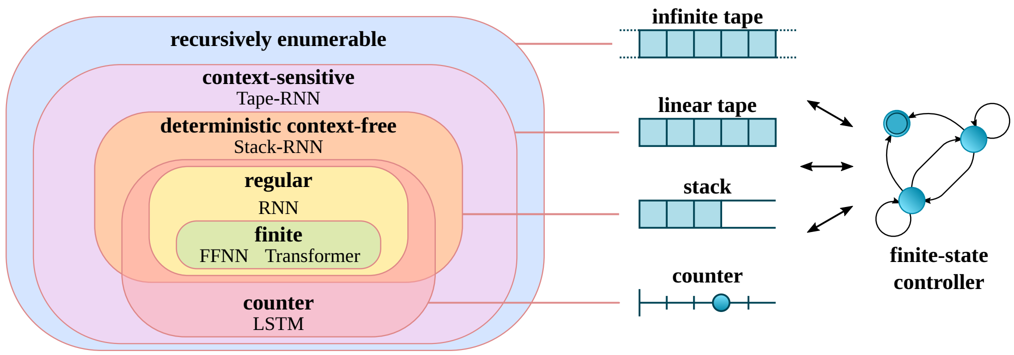
cell where $\boldsymbol{h}_t \in \mathbb{R}^d$ denotes the hidden state, $\boldsymbol{c}_t \in \mathbb{R}^d$ denotes the cell states and, $\boldsymbol{n}_t \in \mathbb{R}^d$ denotes a normalizer state. Further, $\mathbf{i}_t, \mathbf{o}_t, \mathbf{f}_t \in \mathbb{R}^d$ are the input, output and forget gate, respectively, $\boldsymbol{W}_{\boldsymbol{z}}, \boldsymbol{W}_{\mathbf{i}}, \boldsymbol{W}_{\mathbf{f}}, \boldsymbol{W}_{\mathbf{o}} \in \mathbb{R}^{d \times D}$, $\boldsymbol{R}_{\boldsymbol{z}}, \boldsymbol{R}_{\mathbf{i}}, \boldsymbol{R}_{\mathbf{f}}, \boldsymbol{R}_{\mathbf{o}} \in \mathbb{R}^{d \times d}$, and $\boldsymbol{b}_{\boldsymbol{z}}, \boldsymbol{b}_{\mathbf{i}}, \boldsymbol{b}_{\mathbf{f}}, \boldsymbol{b}_{\mathbf{o}} \in \mathbb{R}^d$ are trainable weight matrices and biases. The matrices $\boldsymbol{R}_{\boldsymbol{z}}$, $\boldsymbol{R}_{\mathbf{i}}$, $\boldsymbol{R}_{\mathbf{f}}$, $\boldsymbol{R}_{\mathbf{o}}$ are block-diagonal, where each block represents one head. This way, the parameters reduce to $d^2/(N_h)$, where $N_h$ is the number of heads, limiting the cell interactions to individual heads. The input-, output-, and forget-gates are activated by exponential ($\exp$) or sigmoid functions ($\sigma$); The cell inputs use a hyperbolic tangent function ($\varphi$).

To enable deep modeling, xLSTM organizes its recurrent layers into blocks that combine sLSTM and/or mLSTM layers with additional architectural components. Specifically, TiRex uses the block structure from Beck et al. (2025) that consists of

1. an sLSTM module

2. a feed forward network
3. residual connections around each subcomponent,
4. and pre-normalization layers (RMSNorm)

This block architecture is illustrated in Figure 1 and allows for the training of deep networks. To achieve scalability, xLSTM employs custom CUDA kernels that enable high-throughput training and inference on modern hardware (Pöppel et al., 2024).




| Grammar type (low → high) | Automaton | Memory |
|---|---|---|
| Regular (R) | Finite-state automaton (FSA) | Automaton state |
| Context-free (CF) | Push-down automaton (PDA) | + infinite stack |
| Context-sensitive (CS) | Linear bounded automaton (LBA) | + bounded tape |
| Recursively enumerable (RE) | Turing machine (TM) | + infinite tape |

*Figure 3.* Formal language classes and their correspondence with neural network architectures and $k$-counter machines that have a counting mechanism (from: Delétang et al., 2023).

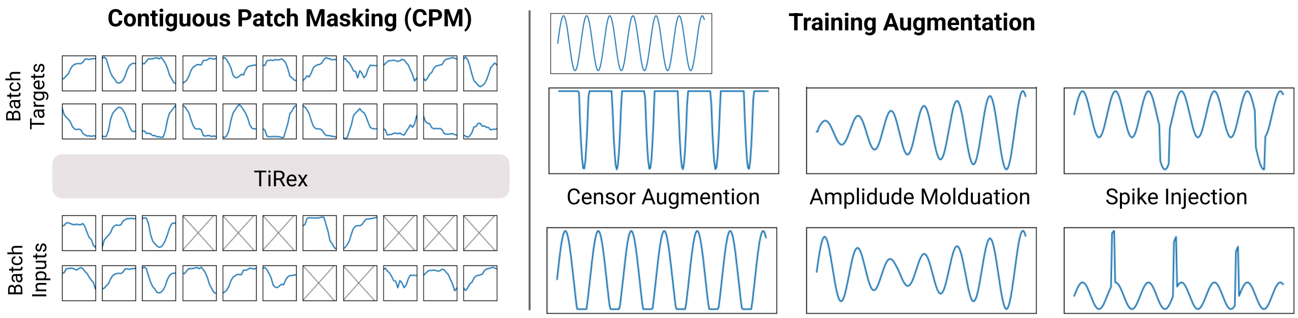



*Figure 4.* Illustration of Contiguous Patch Masking and the different training augmentations.

## D. Data Augmentation

This section details the time series augmentations proposed and used for pre-training TiRex. The augmentations are illustrated in Figure 4.

**Amplitude Modulation** This augmentation introduces scale trends and changepoints into the time series by multiplying the signal with a piecewise linear trend. The modulation trend is generated by sampling changepoints and interpolating amplitudes between them. See Algorithm 1.

**Censor Augmentation** This augmentation censors (clips) the input signal either from below or above, depending on a randomly sampled direction. The clipping threshold is determined by drawing a quantile uniformly from the empirical distribution of the signal. See Algorithm 2.

**Spike Injection** This augmentation injects a structured additive signal in the form of sparse, periodic spikes. First, a periodic pattern is sampled, which is then tiled across the time axis with a sampled periodicity. Each spike label in the pattern is mapped to a kernel (selected from a fixed set) with randomized parameters that control its shape and magnitude. The final augmentation is the sum of all such kernel evaluations added to the original signal. See Algorithm 3. The spike kernel types and their parameter ranges are defined in Table 2, while the available temporal patterns and their sampling probabilities are described in Table 3.

**Algorithm 1** Amplitude Modulation

**Require:** Time Series $\mathbf{y} \in \mathbb{R}^T$
**Ensure:** Augmented Time Series $\mathbf{y}_{\text{aug}}$
1: $k \sim \text{Uniform}(0, 5)$ {Number of changepoints}
2: Sample $k$ changepoints $\{c_1, \ldots, c_k\} \subset \{1, \ldots, T-1\}$
3: $\mathbf{c} \leftarrow [0, \text{sorted}(\{c_1, \ldots, c_k\}), T]$
4: Sample amplitudes $\mathbf{a} \sim \mathcal{N}(1, 1)^{k+2}$
5: Interpolate trend $\mathbf{t} \in \mathbb{R}^T$ from $(\mathbf{c}, \mathbf{a})$
6: $\mathbf{y}_{\text{aug}} \leftarrow \mathbf{y} \odot \mathbf{t}$
7: **return** $\mathbf{y}_{\text{aug}}$

*Table 2.* Kernel types and parameterizations (as functions of the periodicity $\pi$) employed for the spike injection augmentations

| Kernel | Width Param | Amplidue Param |
|---|---|---|
| Tophat | $w \sim [0.05\pi, 0.2\pi]$ | $h \sim [0.5, 3]$ |
| RBF | $\sigma_{\text{RBF}} \sim [0.05\pi, 0.2\pi]$ | $h \sim [0.5, 3]$ |
| Linear | $w \sim [0.05\pi, 0.2\pi]$ | $h \sim [0.5, 3]$ |

**Algorithm 2** Censor Augmentation

**Require:** Time series $\mathbf{y} \in \mathbb{R}^T$
**Ensure:** Augmented time series $\mathbf{y}_{\text{aug}}$
Sample quantile level $q \sim \mathrm{Uniform}(0, 1)$
Compute threshold $c \leftarrow \mathrm{Quantile}(\mathbf{y}, q)$
Sample censor direction $b \sim \mathrm{Bernoulli}(0.5)$ {Bottom or top censor}
**if** $b = 1$ **then**
    $\mathbf{y}_{\text{aug}} \leftarrow \max(y_t, c)$ for all $t$ {Bottom censoring}
**else**
    $\mathbf{y}_{\text{aug}} \leftarrow \min(y_t, c)$ for all $t$ {Top censoring}
**end if**
**return** $\mathbf{y}_{\text{aug}}$

**Algorithm 3** Spike Injection

**Require:** Time series $\mathbf{y} \in \mathbb{R}^T$, spike patterns $\mathcal{P}$, spike kernel set $\mathcal{K}$
**Ensure:** Augmented time series $\mathbf{y}_{\text{aug}}$
Sample periodicity $\pi \sim \mathcal{U}(10, \min(512, T))$
Sample periodic pattern $z \in \mathcal{P}$ and shift it randomly
Sample $s \sim \mathcal{U}(T - \pi, T)$
Generate spike positions $\mathbf{m} \in \mathbb{Z}^T$ by repeating $z$ with spacing $\pi$ and shifting to align the last spike at $s$
Sample kernel type $\kappa \in \mathcal{K}$
**for** each unique spike label in $\mathbf{m}$ **do**
    Sample kernel parameters
    Generate kernel centered at each occurrence
**end for**
Sum kernels to obtain additive spike signal $\mathbf{s} \in \mathbb{R}^T$
$\mathbf{y}_{\text{aug}} \leftarrow \mathbf{y} + \mathbf{s}$
**return** $\mathbf{y}_{\text{aug}}$

*Table 3.* Spike pattern, representative patterns, and sampling probabilities employed for the spike injection augmentations.

| **Category** | **Utilized Pattern** | **Sample Probability** |
|---|---|---|
| Simple | $[0]$, $[0, 1]$ | 0.75 |
| 3-periodic | $[0, 1, 2]$, $[0, 0, 1]$ | 0.10 |
| 4-periodic | $[0, 0, 1, 1]$, $[0, 1, 0, 2]$ | 0.10 |
| Weekly-like | $[0, 0, 0, 0, 0, 1, 1]$, $[0, 0, 0, 0, 0, 1, 2]$ | 0.05 |

## E. Experiment Details

This section outlines the experimental setup, including training procedures, evaluation benchmarks, and state-of-the-art comparisons.

### E.1. Model and Training Hyperparameter

TiRex utilizes a xLSTM-based architecture with hyperparameters summarized in Table 4. The model has 35 million model parameters.

**Pre-Training** TiRex is pre-trained for $500{,}000$ steps with a batch size of $256$ using the AdamW optimizer with a learning rate of $0.001$ and weight decay of $0.01$. We also employ a cosine learning rate scheduler with linear warm-up (the warm-up ratio is set to 5 % and the minimum learning rate is $0.0001$). The context length of TiRex is $2048$.

*Table 4.* TiRex model architecture hyperparameters.

| Parameter | Value |
|---|---|
| Input patch size ($m_{\text{in}}$) | 32 |
| Output patch size ($m_{\text{out}}$) | 32 |
| Embedding dimension ($d$) | 512 |
| Feed-forward dimension ($d_{\text{ff}}$) | 2048 |
| Number of heads | 4 |
| Number of blocks | 12 |

### E.2. Compared Baselines

We compare TiRex against a broad set of state-of-the art models, including zero-shot pre-trained models and task-specific models. The zero-shot models include Chronos and Chronos-Bolt (Ansari et al., 2024a;b), TimesFM (v1.0, v2.0) (Das et al., 2024), Moirai (all variants) (Woo et al., 2024), TabPFN-TS (Hollmann et al., 2025), and Tiny Time Mixer (TTM) (Ekambaram et al., 2024). The task-specific models are trained individually on each dataset. Hence, they do not operate in 'zero-shot' and at-best provide an asymmetric comparison. We still found them useful to contextualize the current strengths and limitations of zero-shot approaches. Specifically, we compare against PatchTST (Nie et al., 2022), TFT (Lim et al., 2021), DLinear (Zeng et al., 2023), DeepAR (Salinas et al., 2020), and N-BEATS (Oreshkin et al., 2019). We report the results of the public leaderboards when available and also replicated the results of pre-trained models within our evaluation pipeline to ensure its validity.

### E.3. Pre-Training Data Corpus

We construct a diverse training corpus by combining real and synthetic time series to support robust generalization across heterogeneous forecasting tasks. Our training dataset has three components:

1. **Chronos Training Data** (30 million time series): We incorporate the training datasets from Chronos and adopt their proposed time series mixup augmentation strategy (Ansari et al., 2024a). Unlike Chronos, we generate significantly more and longer time series, expanding the diversity and temporal span of the data. Table 5 lists the respective datasets, which are provided on hugginface: `https://huggingface.co/datasets/autogluon/chronos_datasets`. Details on the TsMixup procedure are provided in the paragraph below.

2. **Synthetic Gaussian Process Data** (15 million time series): Inspired by KernelSynth (Ansari et al., 2024a), we generate synthetic time series using Gaussian Processes (GPs). Details on the synthetic data generation procedure are provided in the paragraph below.

3. **GiftEval Pre-training Data** ($\approx 2.5$ million time series): We integrate a subset of the pre-training corpus from GiftEval and use it for pre-training. It does not overlap with the GiftEval benchmark evaluation data. Table 6 lists the respective datasets, which are provided on hugginface: `https://huggingface.co/datasets/Salesforce/GiftEvalPretrain`.

**Data Mix Probabilities** For the Chronos training data and the synthetic Gaussian process data, each time series is sampled with equal probability. The GiftEval Pre-training data is sampled with a probability of approximately $8\%$, hence slightly oversampled compared to the share of series due to technical implementation details.

**TsMixup** To augment data diversity, we apply TsMixup (Ansari et al., 2024a), a convex combination of $k$ time series of length $l$ to the training data from Chronos. Each series is $z$-score normalized (Chronos used mean normalization) prior to combination to ensure comparable magnitude. The number $k$ is sampled uniformly from $\{1, \ldots, K_{\max}\}$, the length $l$ is sampled uniformly from $[L_{\min}, L_{\max}]$, and the mixing weights $\lambda_i$ are drawn from a Dirichlet distribution:

$$\mathbf{x}_{1:l}^{\text{mix}} = \sum_{i=1}^{k} \lambda_i \cdot \tilde{\mathbf{x}}_{1:l}^{i}, \tag{11}$$

where $\tilde{\mathbf{x}}_i \in \mathbb{R}^l$ is a normalized time series segment, and $\boldsymbol{\lambda} \sim \text{Dir}(\alpha)$. As $k = 1$ is sampled with non-zero probability, the augmented dataset includes original sequences, thereby preserving base data fidelity while enhancing variability. In our procedure we utilize $K_{\max} = 4$, $L_{\min} = 128$, $L_{\min} = 4096$, and $\alpha = 1.5$; we generate 30 million time series.

**Synthetic GP Data** We generate synthetic time series using Gaussian Processes (GPs), building upon the core ideas of KernelSynth (Ansari et al., 2024a). Each synthetic time series $\mathbf{x} \in \mathbb{R}^{L_{\text{syn}}}$ is sampled from a GP:

$$\mathbf{x} \sim \mathcal{GP}(0, \tilde{\kappa}(t, t')), \tag{12}$$

where $\tilde{\kappa}(t, t')$ is a composite kernel constructed by randomly sampling and combining kernels from a kernel bank $\mathcal{K}$. We sample $j \sim U\{1, J\}$ base kernels (with replacement) from $\mathcal{K}$ and combine them using random binary operations from $\{+, \times\}$ to obtain $\tilde{\kappa}$. Kernel parameters (e.g., length scale, periodicity) are sampled from predefined priors. In our procedure, we utilize $J = 4$ and $L_{\text{syn}} = 4096$; Our kernel bank $\mathcal{K}$ includes periodic, Radial Basis Function (RBF), Rational Quadratic (RQ), and Piecewise Polynomial kernels. We generate $15$ million time series with this procedure.

In contrast to KernelSynth, we introduce the following adaptations:

1. We sample periodicities from both fixed sets (as KernelSynth) and additionally from continuous distributions to increase temporal diversity.
2. We employ a more scalable GP sampler with GPU support and approximations for longer series (Gardner et al., 2018). This enables the generation of more and longer sequences.
3. We use a modified kernel bank.

*Table 5.* Training Datasets published by (Ansari et al., 2024a) — (https://huggingface.co/datasets/autogluon/chronos_datasets) — that were utilized to train TiRex.

| Name | #Series | Avg. Length |
|---|---|---|
| Mexico City Bikes | 494 | 78313 |
| Brazilian Cities Temperature | 12 | 757 |
| Solar (5 Min.) | 5166 | 105120 |
| Solar (Hourly) | 5166 | 105120 |
| Spanish Energy and Weather | 66 | 35064 |
| Taxi (Hourly) | 2428 | 739 |
| USHCN | 6090 | 38653 |
| Weatherbench (Hourly) | 225280 | 350639 |
| Weatherbench (Daily) | 225280 | 14609 |
| Weatherbench (Weekly) | 225280 | 2087 |
| Wiki Daily (100k) | 100000 | 2741 |
| Wind Farms (Hourly) | 100000 | 8514 |
| Wind Farms (Daily) | 100000 | 354 |
| Electricity (15 Min.) | 370 | 113341 |
| Electricity (Hourly) | 321 | 26304 |
| Electricity (Weekly) | 321 | 156 |
| KDD Cup 2018 | 270 | 10897 |
| London Smart Meters | 5560 | 29951 |
| M4 (Daily) | 4227 | 2371 |
| M4 (Hourly) | 414 | 901 |
| M4 (Monthly) | 48000 | 234 |
| M4 (Weekly) | 359 | 1035 |
| Pedestrian Counts | 66 | 47459 |
| Rideshare | 2340 | 541 |
| Taxi (30 Min.) | 2428 | 1478 |
| Temperature-Rain | 32072 | 725 |
| Uber TLC (Hourly) | 262 | 4344 |
| Uber TLC (Daily) | 262 | 181 |

*Table 6.* Subset of pre-training datasets published by (Aksu et al., 2024) — (`https://huggingface.co/datasets/Salesforce/GiftEvalPretrain`) — that were utilized to train TiRex.

| Name | #Series | Avg. Length |
|---|---:|---:|
| azure vm traces 2017 | 159472 | 5553 |
| borg cluster data 2011 | 143386 | 3749 |
| bdg-2 panther | 105 | 8760 |
| bdg-2 fox | 135 | 17219 |
| bdg-2 rat | 280 | 16887 |
| bdg-2 bear | 91 | 16289 |
| lcl | 713 | 13385 |
| smart | 5 | 19142 |
| ideal | 217 | 5785 |
| sceaux | 1 | 34223 |
| borealis | 15 | 5551 |
| buildings 900k | 1795256 | 8761 |
| largest 2017 | 8196 | 105120 |
| largest 2018 | 8428 | 105120 |
| largest 2019 | 8600 | 105120 |
| largest 2020 | 8561 | 105408 |
| largest 2021 | 8548 | 105120 |
| PEMS03 | 358 | 26208 |
| PEMS04 | 307 | 16992 |
| PEMS07 | 883 | 28224 |
| PEMS08 | 170 | 17856 |
| PEMS BAY | 325 | 52128 |
| LOS LOOP | 207 | 34272 |
| BEIJING SUBWAY 30MIN | 276 | 1572 |
| SHMETRO | 288 | 8809 |
| HZMETRO | 80 | 2377 |
| Q-TRAFFIC | 45148 | 5856 |
| subseasonal | 862 | 16470 |
| subseasonal precip | 862 | 11323 |
| wind power | 1 | 7397147 |
| solar power | 1 | 7397222 |
| kaggle web traffic weekly | 145063 | 114 |
| kdd2022 | 134 | 35280 |
| godaddy | 3135 | 41 |
| favorita sales | 111840 | 1244 |
| china air quality | 437 | 13133 |
| beijing air quality | 12 | 35064 |
| residential load power | 271 | 538725 |
| residential pv power | 233 | 537935 |
| cdc fluview ilinet | 75 | 852 |
| cdc fluview who | 74 | 564 |

### E.4. Benchmarks and Metrics

We evaluate our models on two standardized benchmarks — GiftEval (Aksu et al., 2024) and the Chronos Zero-Shot benchmark (Ansari et al., 2024a). Both are hosted on public HuggingFace leaderboards. These benchmarks offer transparent, reproducible evaluations across a wide range of datasets, domains, and forecast horizons, enabling a comprehensive assessment of generalization capabilities. The benchmarks not only specify the datasets and forecast horizons but also metric computations, ensuring valid baseline comparisons.

**GiftEval.** GiftEval (Aksu et al., 2024) comprises 23 datasets totaling over 144,000 time series, spanning seven domains, ten sampling frequencies, and a wide range of forecast horizons from short- to long-term. In sum the benchmark evaluates 97 different evaluation settings. The benchmark includes evaluations of 17 models, covering classical statistical methods, deep learning approaches, and recent foundation models, including all pre-trained models relevant to our study.

In total, 16 out of the 97 evaluation settings in GiftEval overlap with those used for pre-training in our work (i.e., the Chronos pre-train collection). To ensure comparability while avoiding data leakage, we restrict our main evaluation to non-overlapping datasets. We denote this benchmark as **GiftEval-ZS benchmark**. This is possible because of the dataset-level granularity of the leaderboard submissions, which allows custom aggregations of results while preserving fidelity to the original benchmark. Full benchmark results, including the overlapping datasets, are reported in Appendix F.1. The individual datasets and evaluation settings of the benchmark are listed in Table 8. For additional details, please refer to Aksu et al. (2024).

GiftEval uses the Mean Absolute Scaled Error (MASE) for point forecasts and the Continuous Ranked Probability Score (CRPS) for probabilistic forecasts as performance metrics. Equation 13 defines the MASE: $\hat{y}_t$ is the point forecast, $y_t$ is the observed value at time $t$. MASE scales the error by a naïve seasonal forecast, given the seasonal period $s$. Equation 14 respectively defines the CRPS: $F(u)$ is the predictive cumulative distribution and $\mathbf{1}\{y_t \leq u\}$ is the indicator function for the observed value. To evaluate performance over a full forecast horizon, the CRPS is averaged across time steps. In practice, CRPS is approximated by computing the average weighted quantiles loss over a fixed set of quantile levels $Q = \{0.1, 0.2, \ldots, 0.9\}$, with $\hat{y}_t^q$ as the quantile prediction of quantile $q$ at time step $t$ (see Equation 15).

$$\text{MASE} = \frac{\frac{1}{h}\sum_{t=T+1}^{T+h} |\hat{y}_t - y_t|}{\frac{1}{h}\sum_{t=s+1}^{T} |y_t - y_{t-s}|} \tag{13}$$

$$\text{CRPS} = \frac{1}{h}\sum_{t=T+1}^{T+h} \int_{-\infty}^{\infty} \left(F(u) - \mathbf{1}\{y_t \leq u\}\right)^2 \, du \tag{14}$$

$$\text{CRPS} \approx \frac{1}{|Q| \cdot h} \sum_{q \in Q} \frac{2\sum_{t=T+1}^{T+h} \text{QL}(q, \hat{y}_t^q, y_t)}{\sum_{t=T+1}^{T+h} |y_t|} \tag{15}$$

$$\text{QL}(q, \hat{y}^q, y_t) = \begin{cases} q(y_t - \hat{y}_t^q) & \text{if } \hat{y}_t^q \leq y_t \\ (1-q)(\hat{y}_t^q - y_t) & \text{else .} \end{cases} \tag{16}$$

Before aggregation, the metric values of both metrics are normalized per dataset using a seasonal naïve baseline to mitigate scale effects. The aggregated metric scores are computed using the geometric mean of these normalized scores. Additionally, the average rank of the CRPS across evaluation settings is reported to increase robustness against outlier results.

**Chronos-ZS benchmark.** The Chronos-ZS benchmark (Ansari et al., 2024a) consists of 27 datasets, with a focus on short-term forecasting. TiRex's pre-training data has no overlap with Chronos-ZS benchmark, hence we can use it to extend the assessment of its zero-shot capabilities. The evaluation metrics are identical in structure to GiftEval: MASE for point forecasts and Weighted Quantile Loss (WQL) for probabilistic forecasts, with WQL evaluated over the same set of quantiles, making it computationally equivalent to the CRPS approximation of GiftEval. Aggregation procedures, including baseline normalization and geometric mean computation, are also consistent across both benchmarks. The datasets and settings of the benchmark are presented in Table 8; for additional details, please refer to Ansari et al. (2024a).

*Table 7.* GiftEval (Aksu et al., 2024) benchmark datasets and evaluation settings. Evaluation settings that are part of GiftEval-ZS benchmark are marked in the respective column. Forecast Horizion is abbreviated as "Hor" and the number of evaluated windows is abbreviated as "Win".

| Name | GiftEval-ZS | Freq | #Series | Short | | Medium | | Long | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | Hor | Win | Hor | Win | Hor | Win |
| bitbrains_fast_storage | x | 5T | 1250 | 48 | 18 | 480 | 2 | 720 | 2 |
| bitbrains_fast_storage | x | H | 1250 | 48 | 2 | - | - | - | - |
| bitbrains_rnd | x | 5T | 500 | 48 | 18 | 480 | 2 | 720 | 2 |
| bitbrains_rnd | x | H | 500 | 48 | 2 | - | - | - | - |
| bizitobs_application | x | 10S | 1 | 60 | 15 | 600 | 2 | 900 | 1 |
| bizitobs_l2c | x | 5T | 1 | 48 | 20 | 480 | 7 | 720 | 5 |
| bizitobs_l2c | x | H | 1 | 48 | 6 | 480 | 1 | 720 | 1 |
| bizitobs_service | x | 10S | 21 | 60 | 15 | 600 | 2 | 900 | 1 |
| car_parts | x | M | 2674 | 12 | 1 | - | - | - | - |
| covid_deaths | x | D | 266 | 30 | 1 | - | - | - | - |
| electricity | | 15T | 370 | 48 | 20 | 480 | 20 | 720 | 20 |
| electricity | x | D | 370 | 30 | 5 | - | - | - | - |
| electricity | | H | 370 | 48 | 20 | 480 | 8 | 720 | 5 |
| electricity | | W | 370 | 8 | 3 | - | - | - | - |
| ett1 | x | 15T | 1 | 48 | 20 | 480 | 15 | 720 | 10 |
| ett1 | x | D | 1 | 30 | 3 | - | - | - | - |
| ett1 | x | H | 1 | 48 | 20 | 480 | 4 | 720 | 3 |
| ett1 | x | W | 1 | 8 | 2 | - | - | - | - |
| ett2 | x | 15T | 1 | 48 | 20 | 480 | 15 | 720 | 10 |
| ett2 | x | D | 1 | 30 | 3 | - | - | - | - |
| ett2 | x | H | 1 | 48 | 20 | 480 | 4 | 720 | 3 |
| ett2 | x | W | 1 | 8 | 2 | - | - | - | - |
| hierarchical_sales | x | D | 206 | 30 | 4 | - | - | - | - |
| hierarchical_sales | x | W | 118 | 8 | 4 | - | - | - | - |
| hospital | x | M | 767 | 12 | 1 | - | - | - | - |
| jena_weather | x | 10T | 1 | 48 | 20 | 480 | 11 | 720 | 8 |
| jena_weather | x | D | 1 | 30 | 2 | - | - | - | - |
| jena_weather | x | H | 1 | 48 | 19 | 480 | 2 | 720 | 2 |
| kdd_cup_2018 | | D | 270 | 30 | 2 | - | - | - | - |
| kdd_cup_2018 | | H | 270 | 48 | 20 | 480 | 2 | 720 | 2 |
| loop_seattle | x | 5T | 323 | 48 | 20 | 480 | 20 | 720 | 15 |
| loop_seattle | x | D | 323 | 30 | 2 | - | - | - | - |
| loop_seattle | x | H | 323 | 48 | 19 | 480 | 2 | 720 | 2 |
| m4_daily | | D | 4227 | 14 | 1 | - | - | - | - |
| m4_hourly | | H | 207 | 48 | 2 | - | - | - | - |
| m4_monthly | | M | 2400 | 18 | 20 | - | - | - | - |
| m4_quarterly | x | Q | 24000 | 8 | 1 | - | - | - | - |
| m4_weekly | | W | 359 | 13 | 1 | - | - | - | - |
| m4_yearly | x | A | 22974 | 6 | 1 | - | - | - | - |
| m_dense | x | D | 30 | 30 | 3 | - | - | - | - |
| m_dense | x | H | 30 | 48 | 20 | 480 | 4 | 720 | 3 |
| restaurant | x | D | 403 | 30 | 2 | - | - | - | - |
| saugeen | x | D | 1 | 30 | 20 | - | - | - | - |
| saugeen | x | M | 1 | 12 | 7 | - | - | - | - |
| saugeen | x | W | 1 | 8 | 20 | - | - | - | - |
| solar | x | 10T | 137 | 48 | 20 | 480 | 11 | 720 | 8 |
| solar | x | D | 137 | 30 | 2 | - | - | - | - |
| solar | x | H | 137 | 48 | 19 | 480 | 2 | 720 | 2 |
| solar | x | W | 137 | 8 | 1 | - | - | - | - |
| sz_taxi | x | 15T | 156 | 48 | 7 | 480 | 1 | 720 | 1 |
| sz_taxi | x | H | 156 | 48 | 2 | - | - | - | - |
| temperature_rain | | D | 32072 | 30 | 3 | - | - | - | - |
| us_births | x | D | 1 | 30 | 20 | - | - | - | - |
| us_births | x | M | 1 | 12 | 2 | - | - | - | - |
| us_births | x | W | 1 | 8 | 14 | - | - | - | - |

*Table 8.* Chronos-ZS benchmark benchmark datasets (Ansari et al., 2024a) and evalution setting.

| Name | Horizon | Periodicty |
|---|---|---|
| traffic | 24 | 24 |
| australian electricity | 48 | 48 |
| ercot | 24 | 24 |
| ETTm | 24 | 96 |
| ETTh | 24 | 24 |
| exchange rate | 30 | 5 |
| nn5 | 56 | 1 |
| nn5 weekly | 8 | 1 |
| weather | 30 | 1 |
| covid deaths | 30 | 1 |
| fred md | 12 | 12 |
| m4 quarterly | 8 | 4 |
| m4 yearly | 6 | 1 |
| dominick | 8 | 1 |
| m5 | 28 | 1 |
| tourism monthly | 24 | 12 |
| tourism quarterly | 8 | 4 |
| tourism yearly | 4 | 1 |
| car parts | 12 | 12 |
| hospital | 12 | 12 |
| cif 2016 | 12 | 12 |
| m1 yearly | 6 | 1 |
| m1 quarterly | 8 | 4 |
| m1 monthly | 18 | 12 |
| m3 monthly | 18 | 12 |
| m3 yearly | 6 | 1 |
| m3 quarterly | 8 | 4 |

### E.5. Computation & Hardware

We conducted all experiments on Nvidia A40 and H100 GPUS — A40's provide enough GPU memory to conduct all training runs. The inference experiments (GPU memory and Inference Speed) were conducted on a Nivida A40 GPU. CPU requirements are flexible; we utilized a 64-core Xeon(R) Platinum 8358.

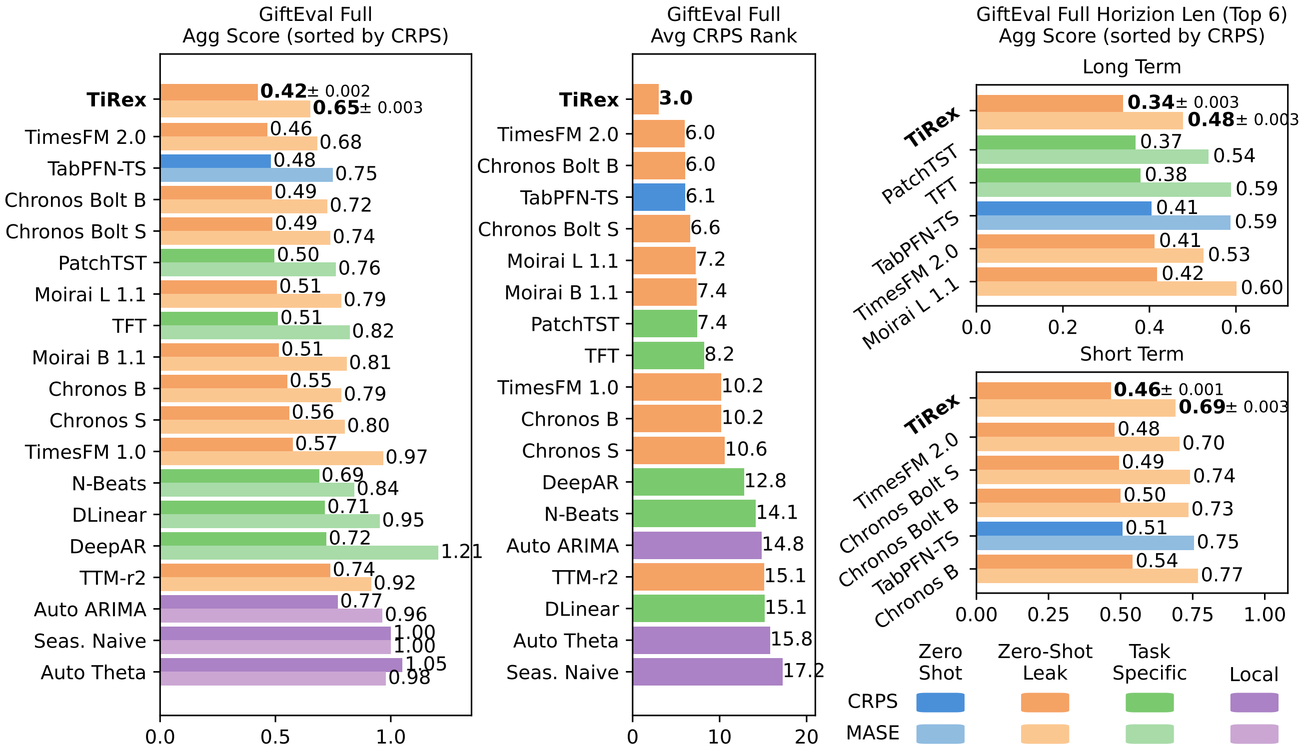



*Figure 5.* Results of the **full GifEval benchmark**. The aggregated scores of the overall benchmark and the short- and long-term subresults are shown. Additionally, the average rank in terms of CRPS, as in the public leaderboard, is presented. Lower values are better. "Zero-shot Leak" refers to models which are partly trained on the benchmark datasets. We trained TiRex with 6 different seeds and report the observed standard deviation in the plot.

## F. Extended Results

This section presents additional experimental results complementing Section 4. The structure is the same as in the main paper, preceded by results for the full GiftEval benchmark: First, extended results on the zero-shot evaluation are presented, followed by extended ablation studies results. Additionally, we provide an inference efficiency analysis, a qualitative analysis, and fine-tuning results.

### F.1. Full GiftEval leaderboard

Figure 5 presents the evaluation results on the full GiftEval benchmark, including settings excluded from GiftEval-ZS benchmark due to training data overlap with TiRex. These results align with those reported on the HuggingFace GiftEval leaderboard. The results are consistent with the trends observed in the main GiftEval-ZS benchmark evaluation. TiRex outperforms all baseline models by a substantial margin overall, with the largest performance gap observed in the long-term forecasting tasks.

### F.2. Zero-Shot Forecasting

**GiftEval-ZS benchmark** Complementing the main paper, Figures 6–8 present the short-, medium-, and long-term evaluation sub-results for all models. Both aggregated scores and average CRPS rank metrics are reported. As discussed in the main text, TiRex consistently achieves the best performance across all settings. The results of the individual evaluation settings are reported in the Tables 10-17.

**Chronos-ZS benchmark** Figure 9 shows the main results on the Chronos-ZS benchmark, which focuses on short-term forecasting. Task-specific and local models not covered by the official leaderboard, which reports only pre-trained models.

These additional results are taken from the benchmark's original publication (Ansari et al., 2024a). The results on the Chronos-ZS benchmark exhibit similar performance patterns to those observed on GiftEval-ZS (Figure 9): TiRex again achieves the best results in terms of WQL score and rank. In the MASE score TiRex has second best score, closely behind TabPFN-TS. Notably, Moirai performs substantially better on the Chronos-ZS benchmark (compared to the GiftEval-ZS benchmark). However, this improvement is likely due to the substantial overlap of $82\%$ between its pre-training data and the Chronos-ZS test set. These results highlight the robustness of TiRex in zero-shot generalization. The results of the individual evaluation settings are reported in the Tables 18-19.

**Inference Speed & Memory** Apart from the forecasting performance, we analyze the GPU memory consumption and inference runtime across all models (see Figure 10). Specifically, we evaluate samples with a context length of 2048 and a prediction length of 32 over multiple batch sizes. As expected, given TiRex substantially smaller size compared to the next best models (TimesFM-2.0 and Chronos-Bolt-Base), TiRex requires significantly less GPU memory and achieves faster inference speeds. Specifically, TiRex is over $11\times$ faster than TimesFM-2.0, over $4\times$ faster than Chronos-Bolt Base, and over $2176\times$ faster than TabPFN-TS. Furthermore, TiRex even outperforms Chronos-Bolt Small, a similarly sized transformer-based architecture for larger batch sizes. The differences in maximum GPU memory consumption follow a similar order.

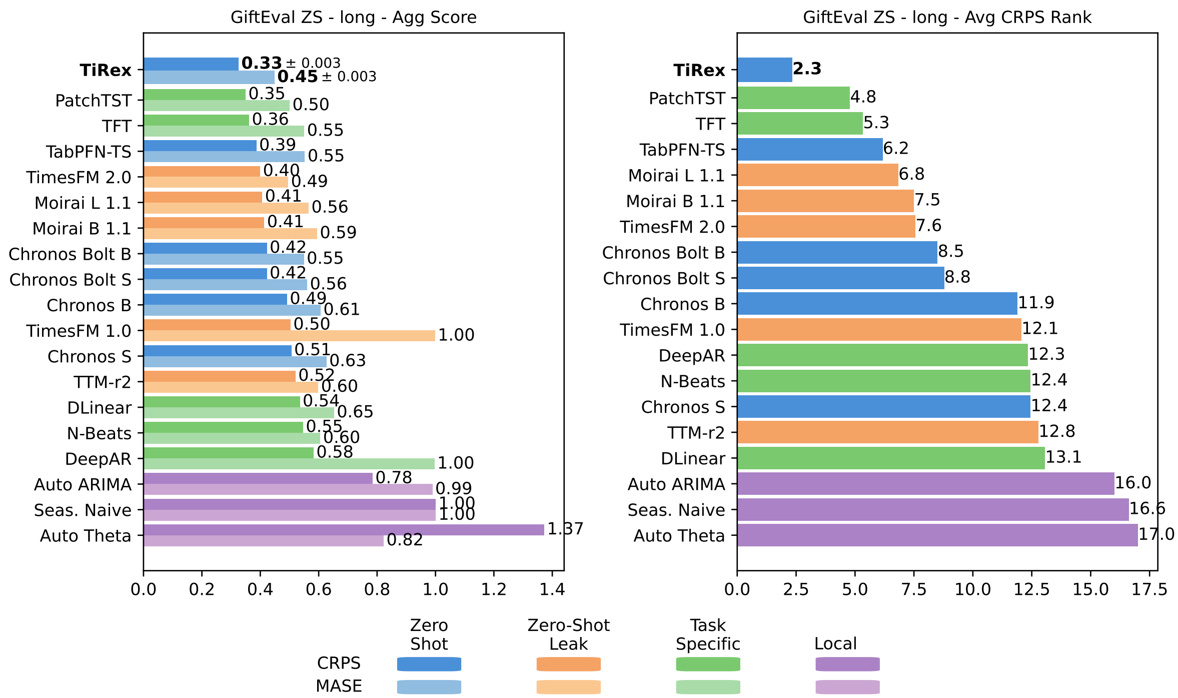



*Figure 6.* Results of the GiftEval-ZS benchmark: The aggregated scores and the average CRPS rank of the benchmarks' **long-term** sub-results are shown. Lower values are better. "Zero-shot Leak" refers to models that are partly trained on the benchmark datasets. We trained TiRex with 6 different seeds and report the observed standard deviation of the aggregated scores.

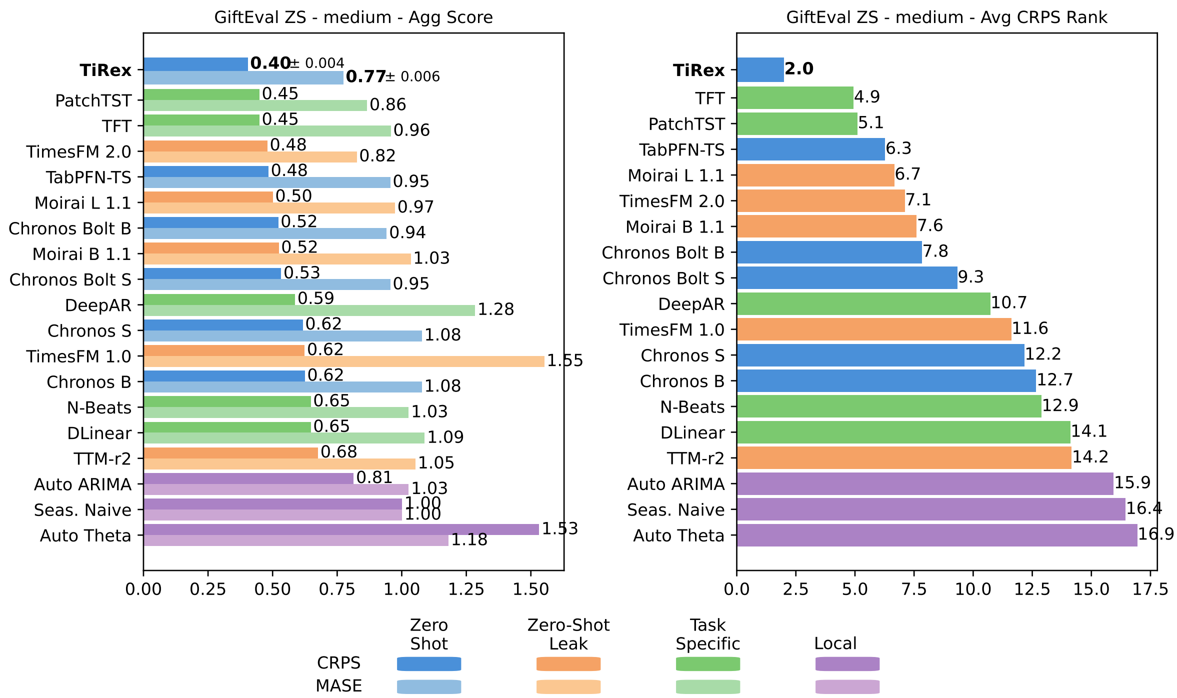



*Figure 7.* Results of the GiftEval-ZS benchmark: The aggregated scores and the average CRPS rank of the benchmarks' **medium-term** sub-results are shown. Lower values are better. "Zero-shot Leak" refers to models that are partly trained on the benchmark datasets. We trained TiRex with 6 different seeds and report the observed standard deviation of the aggregated scores.

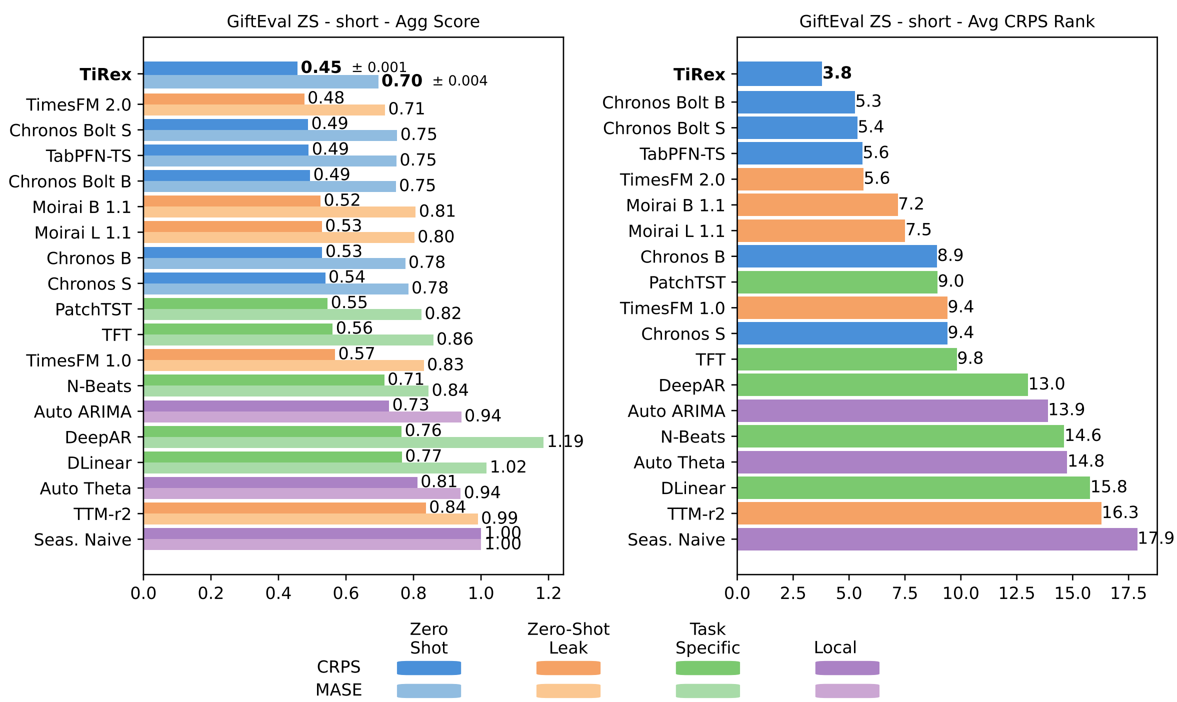



*Figure 8.* Results of the GiftEval-ZS benchmark: The aggregated scores and the average CRPS rank of the benchmarks' **short-term** sub-results are shown. Lower values are better. "Zero-shot Leak" refers to models that are partly trained on the benchmark datasets. We trained TiRex with 6 different seeds and report the observed standard deviation of the aggregated scores.

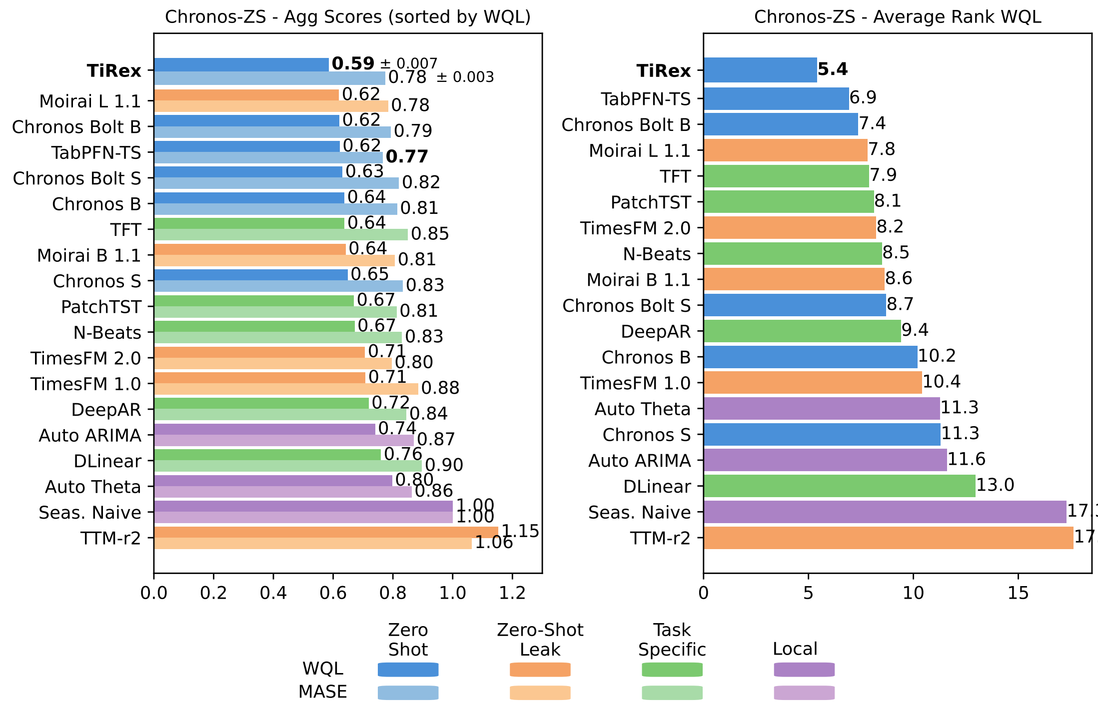



*Figure 9.* Results of the Chronos-ZS benchmark: The aggregated scores and the average WQL rank. Lower values are better. "Zero-shot Leak" refers to models that are partly trained on the benchmark datasets (Overlap: Moirai 82%, TimesFM 15%, TTM: 11%). We trained TiRex with 6 different seeds and report the observed standard deviation of the aggregated scores.

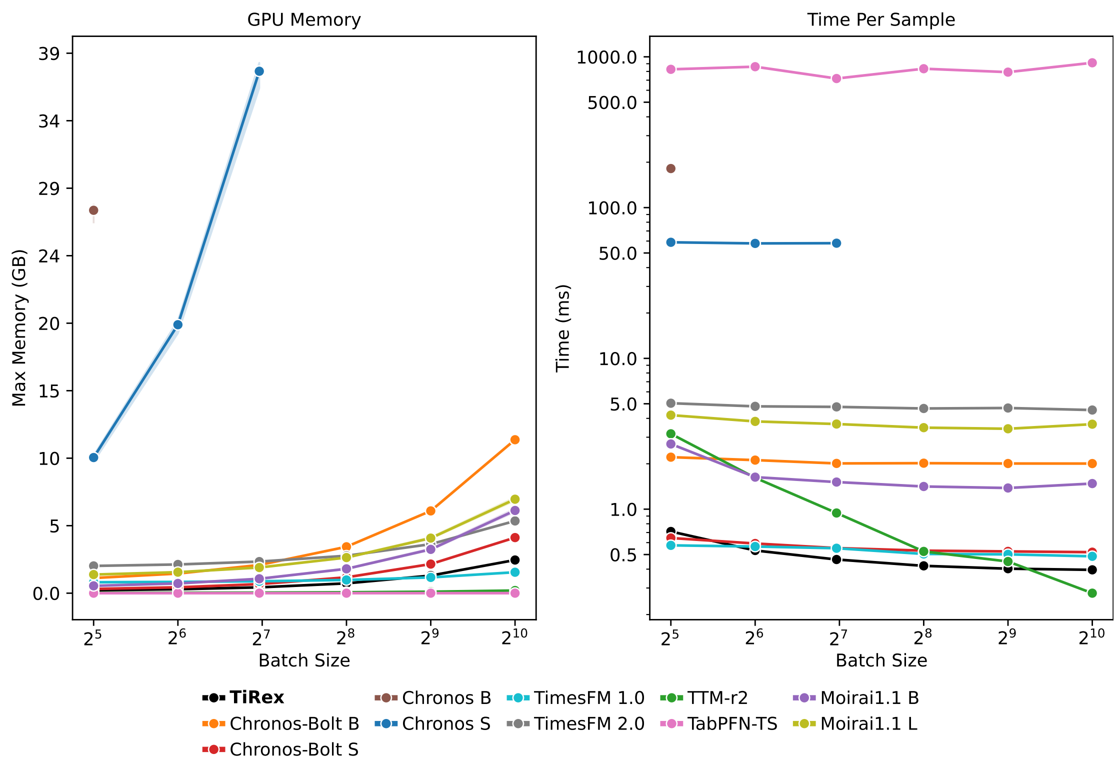



*Figure 10.* Inference efficiency of different pre-trained forecasting models. Left: GPU Memory depending on the batch size. The maximum available GPU memory was 48 GB in the experiment (Nvidia A40). Right: Inference Time per sample depending on the batch size.

### F.3. Ablations

Table 9 provides extended results of the ablation studies.

We also analyze mLSTM and sLSTM mix architectures as proposed in the original xLSTM paper. We denote xLSTM[i:j] for an architecture where $i$ sLSTM blocks are combined with $j$ mLSTM blocks. While using only mLSTM performs worst, switching just one of these blocks back to a sLSTM improves the results close to the sLSTM-only architecture of TiRex.

*Table 9.* Ablation study of individual components. The top two rows report the mean and standard deviation of TiRex over six runs with different random seeds. For the ablation variants, results that degrade performance by more than $3\times$ the standard deviation relative to TiRex are underlined. Columns correspond to evaluation settings: GiftEval-ZS benchmark (overall, short-term, and long-term) and Chronos-ZS benchmark. Lower values indicate better performance.

| | Benchmark | Gift-ZS Overall | | Gift-ZS Long | | Gift-ZS Short | | Chronos-ZS | |
|---|---|---|---|---|---|---|---|---|---|
| | | CRPS | MASE | CRPS | MASE | CRPS | MASE | WQL | MASE |
| | TiRex | 0.411 | 0.647 | 0.325 | 0.45 | 0.455 | 0.696 | 0.592 | 0.776 |
| | $\pm$6 seeds | 0.002 | 0.004 | 0.003 | 0.003 | 0.001 | 0.004 | 0.007 | 0.003 |
| CPM | naïve multi-patch | $\underline{0.424}$ | $\underline{0.662}$ | $\underline{0.335}$ | $\underline{0.460}$ | $\underline{0.475}$ | $\underline{0.718}$ | $\underline{0.650}$ | $\underline{0.817}$ |
| CPM | w/o multi-patch | $\underline{0.445}$ | $\underline{0.704}$ | $\underline{0.370}$ | $\underline{0.518}$ | $\underline{0.471}$ | $\underline{0.719}$ | 0.589 | 0.777 |
| Augment | w/o any | $\underline{0.430}$ | $\underline{0.682}$ | $\underline{0.339}$ | $\underline{0.478}$ | $\underline{0.473}$ | $\underline{0.722}$ | $\underline{0.623}$ | $\underline{0.800}$ |
| Augment | w/o censor | 0.417 | 0.652 | $\underline{0.336}$ | 0.457 | 0.458 | 0.699 | 0.595 | 0.767 |
| Augment | w/o spike | 0.415 | $\underline{0.660}$ | 0.328 | 0.459 | $\underline{0.462}$ | $\underline{0.710}$ | 0.591 | 0.773 |
| Augment | w/o amplidude modulation | 0.409 | 0.644 | 0.323 | 0.448 | 0.455 | 0.694 | $\underline{0.618}$ | $\underline{0.798}$ |
| Backbone | Transformer | $\underline{0.422}$ | $\underline{0.662}$ | $\underline{0.342}$ | $\underline{0.472}$ | $\underline{0.461}$ | 0.702 | 0.597 | 0.768 |
| Backbone | mLSTM | $\underline{0.457}$ | $\underline{0.718}$ | $\underline{0.430}$ | $\underline{0.589}$ | 0.456 | 0.699 | 0.588 | 0.775 |
| Backbone | xLSTM[1:11] | 0.414 | 0.652 | 0.330 | 0.456 | 0.455 | 0.698 | $\underline{0.631}$ | $\underline{0.807}$ |
| Backbone | xLSTM[1:5] | 0.412 | 0.651 | 0.330 | $\underline{0.460}$ | 0.450 | 0.693 | 0.611 | $\underline{0.791}$ |
| Backbone | Chronos Bolt S | $\underline{0.456}$ | $\underline{0.676}$ | $\underline{0.413}$ | $\underline{0.498}$ | $\underline{0.463}$ | 0.705 | 0.609 | $\underline{0.791}$ |
| Backbone | Chronos Bolt B | $\underline{0.454}$ | $\underline{0.670}$ | $\underline{0.418}$ | $\underline{0.493}$ | 0.458 | 0.701 | $\underline{0.627}$ | $\underline{0.807}$ |

**Chronos-Bolt architecture trained with our data pipeline** To further analyze the impact of our dataset and data augmentation pipeline — and separate its impact from our additional methodological contributions, we trained a Chronos-Bolt architecture, as representative of a state-of-the-art pre-trained model. Chronos-Bolt was selected due to its publicly available code and configuration, though the exact training data and procedure remain undisclosed. We used the same training hyperparameters (e.g., learning rate, warmup schedule, . . . ) as for TiRex. As shown in Figure 13, integrating our data pipeline leads to improved performance compared to the published Chronos-Bolt results on the GiftEval-ZS benchmark, while we see a mixed effect on the Chronos-ZS benchmark. Nonetheless, TiRex consistently outperforms the optimized Chronos-Bolt variant, which highlights the contributions of both CPM and the xLSTM backbone to TiRex's effectiveness. As expected, this difference is most pronounced for long-term forecasts.

**Contiguous Patch Masking (CPM)** CPM applies 2 hyperparameters: $p_{\text{mask}}^{\text{max}}$, which defines the maximum masking probability sampled per sample, and $c_{\text{mask}}^{\text{max}}$, which defines the maximum for the number of consecutive patches sampled per sample. As pre-training computational demand hinders an extended hyperparameter search, we heuristically selected $p_{\text{mask}}^{\text{max}} = 0.25$, which corresponds to a typical dropout probability of time series models, and $c_{\text{mask}}^{\text{max}} = 5$ to ensure multi-patch forecasts spanning multiple tokens. To analyze the sensitivity of the model performance to these parameters, we additionally trained TiRex variants spanning the combinations of $p_{\text{mask}}^{\text{max}} = \{0.1, 0.25, 0.5\}$ and $c_{\text{mask}}^{\text{max}} = \{0, 1, 3, 5, 7, 9\}$. Figure 11 illustrates the results: The results are not very sensitive to the parameters as long as CPM is utilized ($c_{\text{mask}}^{\text{max}} > 0$). Additionally, good long-term forecasts require sufficient training samples with multi-patch forecasts spanning multiple tokens, i.e.,$c_{\text{mask}}^{\text{max}} > 3$ or $p_{\text{mask}}^{\text{max}} \geq 0.5$ (the latter more likely leads to neighboring masked patches that effectively mask out more than $c_{\text{mask}}$ patches).

**Augmentations** In addition to the main paper, Table 9 also shows ablation of the individual augmentations. The results indicate that including the augmentations is beneficial, i.e., improves the performance of the model. Performance consistently decreased in at least one benchmark metric when any single augmentation was removed. The most substantial decline occurs when no augmentations were applied. This underscores the effectiveness of each augmentation and their combined positive impact on the model's generalization capabilities.

We do not apply each augmentation to every sample but apply them with a selected probability. Due to the computational cost of pre-training, an extensive hyperparameter search was not feasible. Instead, we heuristically selected application probabilities under the hypothesis that augmentations are beneficial, but excessive use may lead to diminished returns. We chose an application probability of $0.5$ for both Censor and Amplitude Modulation. For Spike Injection, we used a lower probability of $0.05$, as its computational cost is higher, and frequent application creates a speed bottleneck in training.

To analyze the sensitivity of TiRex's performance to these application probabilities, we trained additional variants with altered values. Specifically, we tested probabilities in $\{0, 0.1, 0.25, 0.75, 1\}$ for Censor and Amplitude Modulation, and probabilities in $\{0, 0.01, 0.2, 0.3\}$ for Spike Injection. Figure 12 summarizes the results. The analysis indicates that model performance is relatively robust to variations in application probability as long as the augmentations are utilized at all. However, targeted tuning may yield marginal improvements.

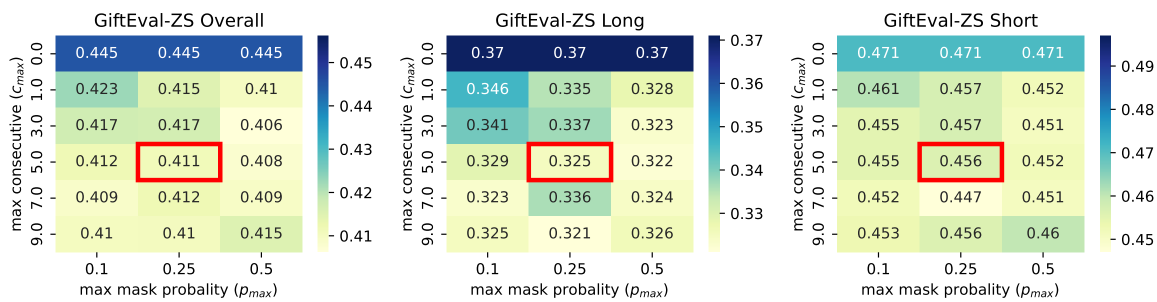



*Figure 11.* CRPS results on the GiftEval-ZS benchmark of TiRex variants trained with a different set of hyperparameters for Contiguous Patch Masking ($c_{\text{mask}}^{\text{max}} = 0$ indicates that Contiguous Patch Masking is not used). The parameters used for training TiRex are enclosed in a red frame.

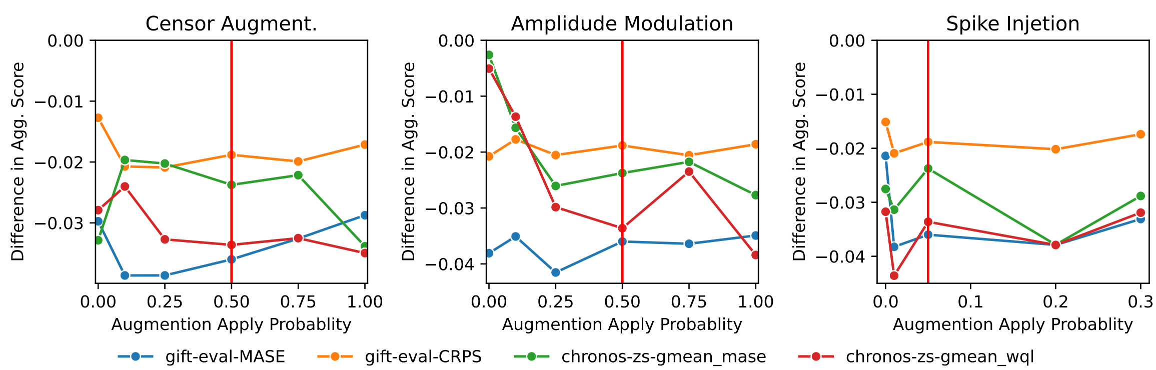



*Figure 12.* CRPS performance variation on the GiftEval-ZS benchmark for TiRex variants trained with different augmentation application probabilities, relative to the baseline TiRex without augmentations. Red vertical lines indicate the parameters used in the actual model configuration.

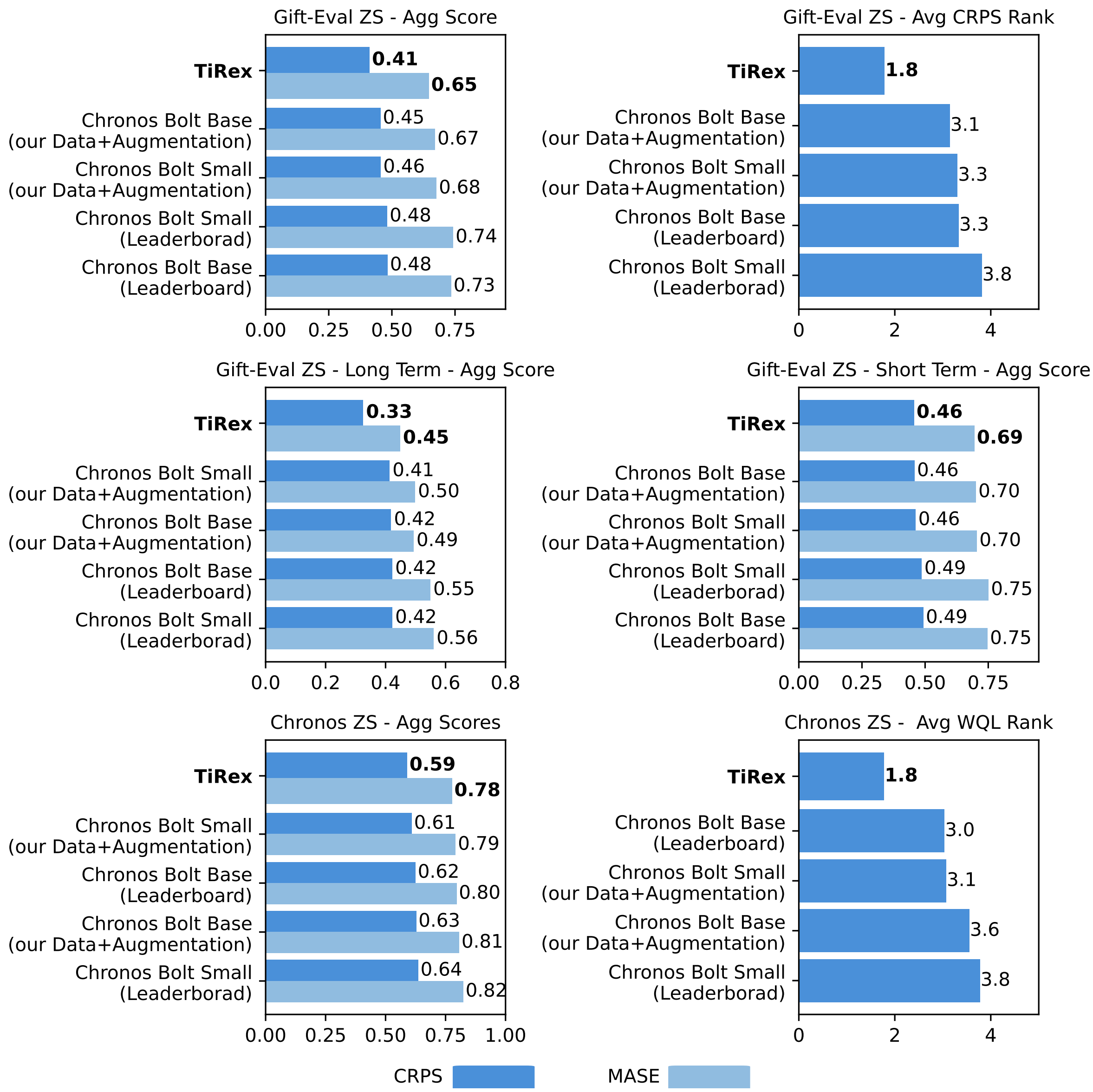



*Figure 13.* Results of training a Chronos-Bolt architecture with the same data and data augmentations as TiRex — compared to the published Chronos-Bolt model and TiRex. The results are from the GiftEval-ZS benchmark and the Chronos-ZS benchmark.

### F.4. Qualitative Analysis

We also qualitatively analyzed the predictions of TiRex and compared them against current state-of-the-art methods (Figure 16 and 15). Beyond its generally higher forecasting accuracy, the analysis shows that TiRex demonstrates robust multi-patch prediction capabilities, maintaining coherent uncertainty estimates across different forecast horizons — and more accurately forecasts short periodic spikes, which are often missed or smoothed out by other models. We hypothesize that these improvements mainly stem from (i) the effective multi-patch forecasting due to CPM, (ii) the sLSTM architecture that provides state-tracking for uncertainty propagation and strong periodicity modeling, and (iii) the spike augmentation strategy enhancing the model's sensitivity to rare, sharp events during training.

### F.5. Finetuned-Forecasting

To further explore the capabilities of our already strong pre-trained model, we finetune TiRex with the training split of the GiftEval benchmark (as defined by Aksu et al., 2024). Fine-tuning is performed jointly across all training datasets. To avoid overfitting, we mix the training data with our pre-training data, using a $20/80$ ratio. For sampling, we use a uniform distribution to choose the dataset to draw the next training sample from. We freeze the input and output layers of TiRex and run over $40$k steps with an initial learning rate of $1 \times 10^{-3}$ and a linear learning rate decay that reaches $0$ at the end of the run. In contrast to the pre-training regime, we do not apply any data augmentation techniques (see Section 3) but still employ CPM.

In the finetuning setting, we observe an incremental improvement over the pre-trained model, especially in the MASE metric (Figure 14). Specifically, we compare the pre-trained to its finetuned version as well as to a fine-tuned TTM (Ekambaram et al., 2024), the only zero-shot model that provides fine-tune results on the GiftEval leaderboard.

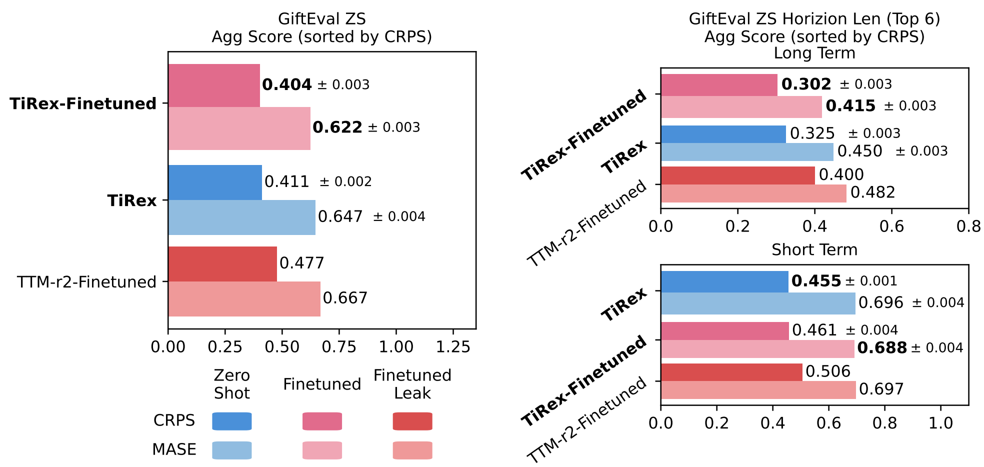



*Figure 14.* GiftEval-ZS benchmark evaluation results comparing the finetuned models from the GiftEval leaderboard to our pre-trained and finetuned models. We trained TiRex with 6 different seeds, finetuned each of these models with 4 different seeds (24 seeds in total), and report the mean and the observed standard deviation of the aggregated scores.

## G. Limitations & Future Work

Like most pre-trained forecasting models, TiRex focuses on univariate time series. Although modeling multivariate series as independent univariate signals often performs well — as reflected in the GiftEval results and, for example, shown in Nie et al. (2022) — future work could incorporate multivariate data, for example in the form of extended contexts or modified input layers. Due to computational constraints, we did not extensively hyperparameters and only conducted a sensitivity analysis on key parameters. Future work should explore more comprehensive tuning for additional performance gains.

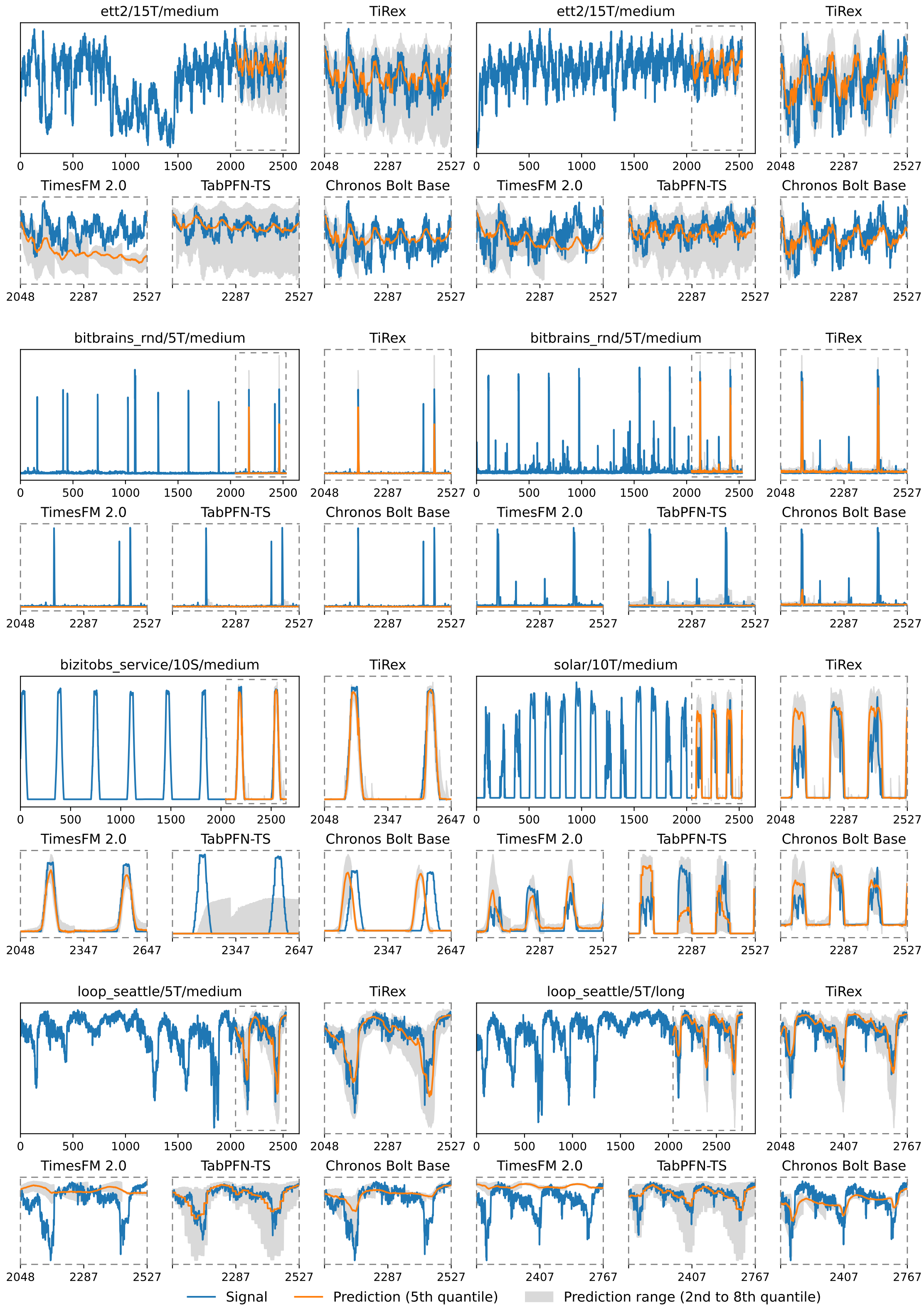



*Figure 15.* Examples of **medium- and long-term forecasts** from the GiftEval benchmark. For each example, we show one plot with the full context and the TiRex prediction, as well as zoomed-in forecasts of the best-performing zero-shot models.

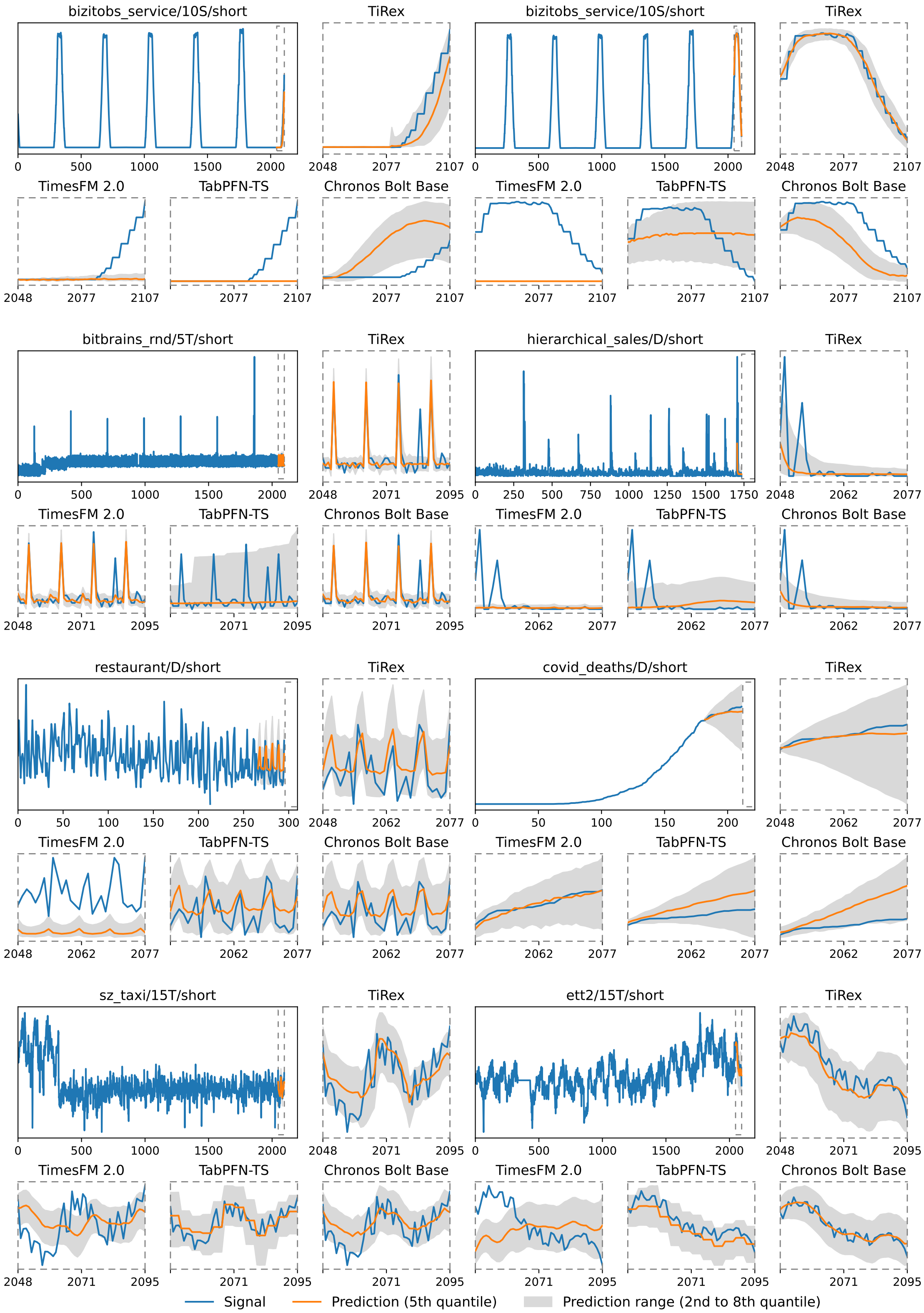



*Figure 16.* Examples of **short-term forecasts** from the GiftEval benchmark. For each example, we show one plot with the full context and the TiRex prediction, as well as zoomed-in forecasts of the best-performing zero-shot models.

*Table 10.* MASE scores of different zero-shot models on the GiftEval benchmark evaluation settings (Part 1/2). The models achieving the **best** and second-best scores are highlighted. Results for datasets that are part of the training data for the respective models are shaded in grey, and these results are excluded from the calculation of the best score. We trained TiRex with 6 different seeds and report the observed standard deviation in the plot.

| | TiRex | Chronos Bolt B | Chronos Bolt S | TimesFM 2.0 | TimesFM 1.0 | TabPFN-TS | Moirai L 1.1 | Moirai B 1.1 | TTM-r2 | Chronos B | Chronos S |
|---|---|---|---|---|---|---|---|---|---|---|---|
| bitbrains_fast_storage/5T/long | **0.916** ± 0.006 | 0.948 | 0.962 | 0.980 | 32.0 | 1.04 | 0.955 | 0.967 | 1.16 | 1.01 | 1.08 |
| bitbrains_fast_storage/5T/medium | **1.00** ± 0.011 | 1.06 | 1.06 | 1.08 | 20.3 | 1.19 | 1.02 | 1.05 | 1.23 | 1.11 | 1.14 |
| bitbrains_fast_storage/5T/short | **0.692** ± 0.007 | 0.752 | 0.778 | 0.731 | 0.874 | 0.888 | 0.827 | 0.792 | 0.966 | 0.850 | 0.833 |
| bitbrains_fast_storage/H/short | **1.06** ± 0.013 | 1.07 | 1.08 | 1.09 | 1.18 | 1.15 | 1.09 | 1.18 | 1.28 | 1.11 | 1.14 |
| bitbrains_rnd/5T/long | **3.35** ± 0.013 | 3.40 | 3.41 | 3.60 | 91.0 | 3.51 | 3.42 | 3.45 | 3.78 | 3.77 | 3.83 |
| bitbrains_rnd/5T/medium | **4.40** ± 0.007 | 4.45 | 4.47 | 4.59 | 49.8 | 4.59 | 4.46 | 4.53 | 4.81 | 4.60 | 4.63 |
| bitbrains_rnd/5T/short | **1.66** ± 0.004 | 1.71 | 1.72 | 1.77 | 1.98 | 1.87 | 1.75 | 1.82 | 2.05 | 1.79 | 1.81 |
| bitbrains_rnd/H/short | 5.84 ± 0.011 | 5.90 | 5.88 | 5.99 | 6.09 | 5.96 | 5.93 | 6.07 | 6.16 | **5.79** | 5.89 |
| bizitobs_application/10S/long | **3.67** ± 0.069 | 10.5 | 9.65 | 4.07 | 16.3 | 7.73 | 7.84 | 13.5 | 9.59 | 9.25 | 9.67 |
| bizitobs_application/10S/medium | **2.85** ± 0.057 | 9.72 | 9.15 | 3.08 | 11.2 | 6.93 | 7.39 | 12.8 | 9.02 | 9.87 | 10.2 |
| bizitobs_application/10S/short | **1.40** ± 0.112 | 5.53 | 5.41 | 1.56 | 4.36 | 3.19 | 4.51 | 5.32 | 4.21 | 3.01 | 3.34 |
| bizitobs_l2c/5T/long | 1.19 ± 0.040 | 1.24 | 1.36 | 1.24 | 1.28 | 1.22 | **1.12** | **1.12** | 1.32 | 1.20 | 1.25 |
| bizitobs_l2c/5T/medium | **0.849** ± 0.028 | 0.878 | 0.920 | 1.02 | 0.887 | 0.870 | 0.987 | 0.853 | 0.992 | 0.942 | 0.880 |
| bizitobs_l2c/5T/short | 0.290 ± 0.005 | 0.278 | **0.272** | 0.312 | 0.310 | 0.338 | 0.285 | 0.291 | 0.324 | 0.301 | 0.301 |
| bizitobs_l2c/H/long | 0.590 ± 0.022 | **0.556** | 0.612 | 1.30 | 1.27 | 0.919 | 1.27 | 1.09 | 1.24 | 1.25 | 1.31 |
| bizitobs_l2c/H/medium | 0.525 ± 0.020 | **0.495** | 0.570 | 1.18 | 1.49 | 0.748 | 1.25 | 1.32 | 1.24 | 1.34 | 1.32 |
| bizitobs_l2c/H/short | 0.528 ± 0.021 | **0.432** | 0.485 | 0.782 | 1.05 | 0.634 | 1.15 | 0.999 | 0.983 | 0.990 | 0.905 |
| bizitobs_service/10S/long | **1.57** ± 0.057 | 5.30 | 4.85 | 2.15 | 7.77 | 3.90 | 4.33 | 6.08 | 5.34 | 4.22 | 3.89 |
| bizitobs_service/10S/medium | **1.32** ± 0.058 | 4.98 | 4.64 | 1.53 | 6.46 | 3.78 | 3.87 | 5.99 | 5.12 | 4.58 | 4.06 |
| bizitobs_service/10S/short | **0.884** ± 0.052 | 3.32 | 2.90 | 1.04 | 2.90 | 2.00 | 2.31 | 3.43 | 2.73 | 1.88 | 1.91 |
| car_parts/M/short | **0.838** ± 0.004 | 0.855 | 0.858 | 0.922 | 0.893 | 0.843 | 0.903 | 0.835 | 1.57 | 0.908 | 0.885 |
| covid_deaths/D/short | 39.5 ± 0.803 | 38.9 | **36.5** | 47.4 | 55.6 | 37.8 | 36.5 | 34.6 | 53.5 | 42.7 | 42.2 |
| electricity/15T/long | 0.891 ± 0.008 | 0.933 | 0.953 | **0.904** | 1.50 | 1.02 | 1.31 | 1.32 | 1.35 | 1.01 | 1.06 |
| electricity/15T/medium | 0.841 ± 0.006 | 0.862 | 0.896 | **0.845** | 1.49 | 0.977 | 1.29 | 1.33 | 1.32 | 0.990 | 1.03 |
| electricity/15T/short | 0.945 ± 0.008 | 0.935 | 0.936 | **0.907** | 1.48 | 1.23 | 1.71 | 1.54 | 1.43 | 1.05 | 1.13 |
| electricity/D/short | **1.43** ± 0.010 | 1.45 | 1.48 | 1.49 | 1.75 | 1.49 | 1.51 | 1.50 | 1.66 | 1.56 | 1.60 |
| electricity/H/long | 1.21 ± 0.018 | 1.24 | 1.26 | **1.05** | 1.21 | 1.34 | 1.36 | 1.26 | 1.38 | 1.20 | 1.23 |
| electricity/H/medium | 1.08 ± 0.010 | 1.08 | 1.10 | **0.929** | 1.07 | 1.18 | 1.20 | 1.19 | 1.25 | 1.06 | 1.09 |
| electricity/H/short | 0.869 ± 0.009 | 0.873 | 0.914 | **0.763** | 0.878 | 1.04 | 1.08 | 1.09 | 1.16 | 0.902 | 0.951 |
| electricity/W/short | 1.46 ± 0.010 | 1.48 | 1.50 | **1.45** | 1.86 | **1.45** | 1.79 | 1.92 | 2.52 | 1.49 | 1.54 |
| ett1/15T/long | **1.05** ± 0.009 | 1.14 | 1.19 | 1.11 | 1.32 | 1.11 | 1.40 | 1.12 | 1.20 | 1.35 | 1.50 |
| ett1/15T/medium | **1.04** ± 0.009 | 1.06 | 1.11 | 1.08 | 1.27 | 1.05 | 1.30 | 1.24 | 1.14 | 1.32 | 1.36 |
| ett1/15T/short | 0.706 ± 0.007 | **0.680** | 0.704 | 0.719 | 0.875 | 0.787 | 0.925 | 0.825 | 0.812 | 0.801 | 0.872 |
| ett1/D/short | 1.71 ± 0.016 | 1.67 | 1.70 | **1.65** | 1.70 | 1.77 | 1.75 | 1.74 | 1.96 | 1.90 | 1.80 |
| ett1/H/long | **1.34** ± 0.030 | 1.35 | 1.44 | 1.51 | 1.41 | 1.46 | 1.45 | 1.38 | 1.36 | 1.43 | 1.42 |
| ett1/H/medium | **1.25** ± 0.017 | 1.37 | 1.37 | 1.31 | 1.44 | 1.36 | 1.34 | 1.35 | 1.32 | 1.37 | 1.31 |
| ett1/H/short | **0.827** ± 0.007 | 0.828 | 0.834 | 0.866 | 0.938 | 0.891 | 0.855 | 0.885 | 0.882 | 0.840 | 0.898 |
| ett1/W/short | 1.72 ± 0.044 | 1.70 | 1.70 | 1.65 | 1.73 | 1.58 | **1.51** | 1.54 | 1.54 | 1.66 | 1.65 |
| ett2/15T/long | **0.932** ± 0.012 | 0.940 | 0.991 | 0.941 | 1.03 | 0.958 | 1.14 | 1.30 | 0.986 | 1.14 | 1.11 |
| ett2/15T/medium | **0.910** ± 0.010 | 0.922 | 0.987 | 0.938 | 1.01 | 0.939 | 1.06 | 1.10 | 0.987 | 1.06 | 1.02 |
| ett2/15T/short | 0.749 ± 0.010 | 0.766 | 0.788 | **0.747** | 0.898 | 0.845 | 1.00 | 0.959 | 0.832 | 0.857 | 0.885 |
| ett2/D/short | 1.28 ± 0.019 | 1.32 | **1.22** | 1.56 | 1.64 | 1.54 | 1.44 | 1.31 | 1.56 | 1.26 | 1.43 |
| ett2/H/long | 1.16 ± 0.037 | **1.04** | 1.07 | 1.13 | 1.09 | 1.37 | 1.28 | 1.12 | 1.13 | 1.12 | **1.04** |
| ett2/H/medium | 1.05 ± 0.021 | **1.03** | 1.05 | 1.05 | 1.12 | 1.28 | 1.18 | **1.03** | 1.10 | 1.15 | 1.14 |
| ett2/H/short | 0.742 ± 0.006 | **0.733** | 0.744 | 0.755 | 0.821 | 0.787 | 0.783 | 0.807 | 0.790 | 0.781 | 0.790 |
| ett2/W/short | 0.797 ± 0.040 | **0.739** | 0.791 | 1.12 | 1.13 | 0.959 | 1.31 | 0.851 | 1.36 | 0.749 | 0.807 |
| hierarchical_sales/D/short | 0.744 ± 0.002 | **0.743** | 0.749 | 0.752 | 0.745 | 0.766 | 0.745 | 0.746 | 0.834 | 0.774 | 0.801 |
| hierarchical_sales/W/short | 0.721 ± 0.001 | 0.733 | 0.733 | **0.703** | 0.725 | 0.723 | 0.749 | 0.747 | 1.09 | 0.764 | 0.756 |
| hospital/M/short | 0.767 ± 0.003 | 0.791 | 0.801 | 0.755 | 0.783 | **0.753** | 0.768 | 0.775 | 1.05 | 0.816 | 0.813 |

*Table 11.* MASE scores of different zero-shot models on the GiftEval benchmark evaluation settings (Part 2/2). The models achieving the **best** and second-best scores are highlighted. Results for datasets that are part of the training data for the respective models are shaded in grey, and these results are excluded from the calculation of the best score. We trained TiRex with 6 different seeds and report the observed standard deviation in the plot.

| | TiRex | Chronos Bolt B | Chronos Bolt S | TimesFM 2.0 | TimesFM 1.0 | TabPFN-TS | Moirai L 1.1 | Moirai B 1.1 | TTM-r2 | Chronos B | Chronos S |
|---|---|---|---|---|---|---|---|---|---|---|---|
| jena_weather/10T/long | **0.641** ± 0.015 | 0.657 | 0.703 | 0.231 | 1.36 | 0.657 | 0.792 | 0.762 | 0.649 | 0.802 | 0.956 |
| jena_weather/10T/medium | **0.610** ± 0.005 | **0.610** | 0.646 | 0.191 | 1.10 | 0.625 | 0.694 | 0.712 | 0.656 | 0.722 | 0.744 |
| jena_weather/10T/short | **0.297** ± 0.006 | 0.306 | 0.320 | 0.091 | 0.318 | 0.325 | 0.338 | 0.350 | 0.346 | 0.366 | 0.359 |
| jena_weather/D/short | **1.02** ± 0.014 | 1.05 | 1.03 | 1.24 | 1.20 | 1.23 | 1.14 | 1.15 | 2.53 | 1.12 | 1.14 |
| jena_weather/H/long | 0.987 ± 0.031 | 1.03 | 1.06 | 1.06 | 1.38 | 1.10 | **0.881** | 1.06 | 1.07 | 1.11 | 1.16 |
| jena_weather/H/medium | 0.828 ± 0.023 | 0.747 | **0.721** | 0.864 | 1.05 | 0.954 | 0.891 | 0.817 | 0.815 | 0.883 | 0.842 |
| jena_weather/H/short | **0.516** ± 0.003 | 0.536 | 0.540 | 0.525 | 0.589 | 0.595 | 0.585 | 0.554 | 0.575 | 0.567 | 0.583 |
| kdd_cup_2018/D/short | 1.21 ± 0.009 | 1.20 | 1.19 | 1.21 | 1.21 | **1.17** | 1.20 | 1.20 | **1.17** | 1.37 | 1.40 |
| kdd_cup_2018/H/long | 0.759 ± 0.027 | 0.684 | 0.925 | 1.03 | 1.09 | 1.06 | 0.867 | 0.960 | **1.00** | 1.14 | 1.24 |
| kdd_cup_2018/H/medium | 0.825 ± 0.024 | 0.700 | 0.857 | **1.03** | 1.14 | 1.14 | 0.954 | 1.05 | 1.07 | 1.24 | 1.34 |
| kdd_cup_2018/H/short | 0.657 ± 0.006 | 0.601 | 0.667 | **0.941** | 1.09 | 1.10 | 0.894 | 0.944 | 1.01 | 1.04 | 1.06 |
| loop_seattle/5T/long | **1.02** ± 0.038 | 1.24 | 1.19 | 1.13 | 1.43 | 1.05 | 0.556 | 0.591 | 1.11 | 1.31 | 1.38 |
| loop_seattle/5T/medium | **0.941** ± 0.019 | 1.14 | 1.15 | 1.12 | 1.49 | 0.972 | 0.450 | 0.523 | 1.07 | 1.61 | 1.61 |
| loop_seattle/5T/short | **0.572** ± 0.005 | 0.628 | 0.631 | 0.583 | 0.836 | 0.588 | 0.486 | 0.536 | 0.631 | 0.764 | 0.768 |
| loop_seattle/D/short | 0.878 ± 0.005 | 0.903 | 0.919 | **0.859** | 0.880 | 0.899 | 0.916 | 0.903 | 1.74 | 0.912 | 1.00 |
| loop_seattle/H/long | 0.917 ± 0.011 | 0.996 | 1.08 | **0.906** | 1.26 | 0.922 | 1.05 | 1.15 | 1.11 | 1.01 | 1.12 |
| loop_seattle/H/medium | 0.944 ± 0.012 | 1.02 | 1.10 | **0.934** | 1.32 | 0.948 | 1.00 | 1.14 | 1.19 | 1.05 | 1.14 |
| loop_seattle/H/short | 0.850 ± 0.005 | 0.900 | 0.915 | **0.832** | 1.15 | 0.912 | 0.945 | 1.06 | 1.08 | 0.926 | 0.967 |
| m4_daily/D/short | 3.15 ± 0.063 | 3.20 | 3.19 | 3.09 | 3.27 | **4.31** | 4.18 | 5.37 | 4.40 | 3.18 | 3.16 |
| m4_hourly/H/short | 0.719 ± 0.026 | 0.837 | 0.866 | 0.596 | 0.768 | **0.780** | 0.886 | 0.971 | 2.78 | 0.693 | 0.739 |
| m4_monthly/M/short | 0.929 ± 0.004 | 0.949 | 0.954 | 0.600 | 0.957 | **0.895** | 0.977 | 0.953 | 1.53 | 0.973 | 0.982 |
| m4_quarterly/Q/short | 1.18 ± 0.013 | 1.22 | 1.25 | 0.965 | 1.40 | **1.17** | 1.14 | 1.14 | 2.03 | 1.23 | 1.24 |
| m4_weekly/W/short | 1.90 ± 0.029 | 2.08 | 2.11 | 2.22 | 2.42 | **2.07** | 2.58 | 2.81 | 3.48 | 2.08 | 2.09 |
| m4_yearly/A/short | 3.45 ± 0.075 | 3.51 | 3.69 | **2.54** | 3.35 | 3.16 | 2.97 | 3.01 | 5.13 | 3.64 | 3.74 |
| m_dense/D/short | 0.688 ± 0.016 | 0.716 | 0.742 | 0.636 | 0.702 | **0.634** | 0.957 | 1.10 | 1.21 | 0.712 | 0.834 |
| m_dense/H/long | 0.730 ± 0.013 | 0.938 | 0.913 | 0.795 | 0.787 | 1.04 | **0.696** | 0.734 | 1.06 | 0.773 | 0.737 |
| m_dense/H/medium | 0.736 ± 0.016 | 0.881 | 0.820 | 0.771 | 0.765 | 1.02 | **0.684** | 0.734 | 1.02 | 0.757 | 0.712 |
| m_dense/H/short | 0.788 ± 0.009 | **0.775** | 0.805 | 0.848 | 0.849 | 0.916 | 0.777 | 0.837 | 1.09 | 0.800 | 0.803 |
| restaurant/D/short | **0.677** ± 0.002 | 0.700 | 0.700 | 0.692 | 0.704 | 0.782 | 0.715 | 0.704 | 0.897 | 0.728 | 0.758 |
| saugeen/D/short | 3.12 ± 0.108 | **2.84** | 2.96 | 3.34 | 3.30 | 3.30 | 3.29 | 2.91 | 4.03 | 3.29 | 2.98 |
| saugeen/M/short | 0.750 ± 0.020 | 0.739 | 0.727 | 0.836 | 0.814 | **0.707** | 0.756 | 0.834 | 0.790 | 0.854 | 0.992 |
| saugeen/W/short | **1.18** ± 0.028 | 1.22 | 1.24 | 1.95 | 1.28 | 1.25 | 1.38 | 1.41 | 1.87 | 1.35 | 1.37 |
| solar/10T/long | **0.828** ± 0.021 | 1.07 | 1.19 | 1.15 | 1.58 | 0.915 | 1.95 | 2.02 | 1.10 | 1.66 | 1.98 |
| solar/10T/medium | **0.879** ± 0.037 | 1.03 | 1.06 | 1.17 | 1.38 | 0.935 | 1.82 | 1.89 | 1.13 | 1.54 | 1.79 |
| solar/10T/short | 1.05 ± 0.032 | 0.991 | **0.947** | 1.48 | 1.49 | 1.08 | 1.11 | 1.10 | 1.18 | 1.11 | 1.22 |
| solar/D/short | **0.971** ± 0.005 | 0.982 | 0.995 | **0.971** | 0.990 | 0.985 | 0.987 | 1.02 | 1.07 | 1.01 | 1.08 |
| solar/H/long | **0.697** ± 0.024 | 1.03 | 0.957 | 1.27 | 1.44 | 0.977 | 1.02 | 1.07 | 1.12 | 1.07 | 1.01 |
| solar/H/medium | **0.731** ± 0.026 | 0.931 | 0.926 | 0.959 | 1.04 | 0.921 | 0.917 | 0.892 | 1.05 | 0.806 | 0.815 |
| solar/H/short | **0.699** ± 0.019 | 0.813 | 0.852 | 1.02 | 1.04 | 0.937 | 0.875 | 0.893 | 0.980 | 0.827 | 0.855 |
| solar/W/short | 1.13 ± 0.075 | 0.980 | 0.991 | 1.28 | 1.15 | 0.878 | 1.53 | 1.66 | 2.88 | 1.15 | **0.877** |
| sz_taxi/15T/long | **0.509** ± 0.003 | 0.545 | 0.538 | 0.514 | 0.535 | 0.560 | 0.554 | 0.537 | 0.531 | 0.567 | 0.584 |
| sz_taxi/15T/medium | **0.536** ± 0.001 | 0.559 | 0.562 | 0.546 | 0.558 | 0.585 | 0.569 | 0.558 | 0.566 | 0.597 | 0.618 |
| sz_taxi/15T/short | **0.544** ± 0.001 | 0.548 | 0.550 | 0.539 | 0.558 | 0.571 | 0.581 | 0.576 | 0.574 | 0.589 | 0.598 |
| sz_taxi/H/short | 0.563 ± 0.002 | 0.562 | 0.567 | **0.560** | 0.566 | 0.582 | 0.601 | 0.588 | 0.597 | 0.576 | 0.579 |
| temperature_rain/D/short | 1.34 ± 0.003 | 1.30 | 1.32 | 1.43 | 1.42 | **1.38** | 1.20 | 1.31 | 1.66 | 1.41 | 1.44 |
| us_births/D/short | 0.404 ± 0.017 | 0.485 | 0.528 | 0.370 | 0.552 | **0.320** | 0.503 | 0.509 | 1.63 | 0.420 | 0.436 |
| us_births/M/short | 0.808 ± 0.066 | 0.924 | 0.756 | **0.497** | 0.622 | 0.588 | 0.771 | 0.723 | 1.32 | 0.778 | 0.572 |
| us_births/W/short | 1.08 ± 0.032 | 1.09 | 1.12 | 1.10 | 1.06 | 0.929 | 1.47 | 1.44 | 1.78 | 0.932 | **0.921** |

*Table 12.* CRPS scores of different zero-shot models on the GiftEval benchmark evaluation settings (Part 1/2). The models achieving the **best** and second-best scores are highlighted. Results for datasets that are part of the training data for the respective models are shaded in grey, and these results are excluded from the calculation of the best score. We trained TiRex with 6 different seeds and report the observed standard deviation in the plot.

| | TiRex | Chronos Bolt B | Chronos Bolt S | TimesFM 2.0 | TimesFM 1.0 | TabPFN-TS | Moirai L 1.1 | Moirai B 1.1 | TTM-r2 | Chronos B | Chronos S |
|---|---|---|---|---|---|---|---|---|---|---|---|
| bitbrains_fast_storage/5T/long | **0.655** ± 0.028 | 0.748 | 0.753 | 0.908 | 0.806 | 0.760 | 0.716 | 0.732 | 0.939 | 0.711 | 0.709 |
| bitbrains_fast_storage/5T/medium | **0.605** ± 0.016 | 0.755 | 0.867 | 0.881 | 0.746 | 0.735 | 0.636 | 0.662 | 0.906 | 0.804 | 0.803 |
| bitbrains_fast_storage/5T/short | **0.408** ± 0.005 | 0.454 | 0.435 | 0.447 | 0.476 | 0.456 | 0.412 | 0.413 | 0.596 | 0.463 | 0.446 |
| bitbrains_fast_storage/H/short | 0.699 ± 0.013 | 0.774 | **0.589** | 0.688 | 0.699 | 0.591 | 0.646 | 0.613 | 0.926 | 0.622 | 0.614 |
| bitbrains_rnd/5T/long | **0.660** ± 0.065 | 0.756 | 0.756 | 0.706 | 0.806 | 0.730 | 0.678 | 0.665 | 0.779 | 1.05 | 1.08 |
| bitbrains_rnd/5T/medium | **0.594** ± 0.019 | 0.605 | 0.792 | 0.727 | 0.734 | 0.710 | **0.594** | 0.616 | 0.835 | 0.644 | 0.615 |
| bitbrains_rnd/5T/short | **0.403** ± 0.001 | 0.438 | 0.453 | 0.461 | 0.519 | 0.455 | 0.418 | 0.446 | 0.605 | 0.507 | 0.493 |
| bitbrains_rnd/H/short | 0.631 ± 0.013 | 0.624 | 0.623 | 0.649 | 0.654 | 0.637 | **0.566** | 0.580 | 1.06 | 0.665 | 0.614 |
| bizitobs_application/10S/long | **0.053** ± 0.004 | 0.109 | 0.092 | 0.057 | 0.128 | 0.088 | 0.094 | 0.120 | 0.144 | 0.093 | 0.093 |
| bizitobs_application/10S/medium | 0.041 ± 0.003 | 0.104 | 0.085 | **0.033** | 0.136 | 0.070 | 0.084 | 0.104 | 0.126 | 0.117 | 0.078 |
| bizitobs_application/10S/short | **0.013** ± 0.001 | 0.054 | 0.035 | 0.014 | 0.056 | 0.031 | 0.038 | 0.033 | 0.058 | 0.031 | 0.031 |
| bizitobs_l2c/5T/long | 0.581 ± 0.032 | 0.738 | 0.790 | 0.748 | 0.734 | 0.511 | **0.508** | 0.554 | 0.785 | 0.722 | 0.743 |
| bizitobs_l2c/5T/medium | 0.366 ± 0.015 | 0.445 | 0.462 | 0.529 | 0.449 | **0.345** | 0.410 | 0.380 | 0.540 | 0.484 | 0.465 |
| bizitobs_l2c/5T/short | 0.078 ± 0.002 | 0.074 | **0.073** | 0.084 | 0.080 | 0.099 | 0.079 | 0.078 | 0.107 | 0.084 | 0.087 |
| bizitobs_l2c/H/long | **0.276** ± 0.010 | 0.278 | 0.295 | 0.728 | 0.724 | 0.440 | 0.600 | 0.495 | 0.751 | 0.738 | 0.780 |
| bizitobs_l2c/H/medium | **0.253** ± 0.008 | 0.254 | 0.285 | 0.640 | 0.856 | 0.380 | 0.619 | 0.688 | 0.782 | 0.793 | 0.780 |
| bizitobs_l2c/H/short | 0.227 ± 0.011 | **0.189** | 0.204 | 0.345 | 0.485 | 0.290 | 0.559 | 0.493 | 0.549 | 0.469 | 0.428 |
| bizitobs_service/10S/long | **0.054** ± 0.002 | 0.113 | 0.096 | 0.062 | 0.137 | 0.091 | 0.104 | 0.115 | 0.140 | 0.094 | 0.093 |
| bizitobs_service/10S/medium | **0.034** ± 0.002 | 0.096 | 0.082 | 0.038 | 0.109 | 0.067 | 0.069 | 0.090 | 0.117 | 0.073 | 0.068 |
| bizitobs_service/10S/short | **0.013** ± 0.000 | 0.051 | 0.032 | 0.015 | 0.051 | 0.031 | 0.032 | 0.042 | 0.053 | 0.027 | 0.027 |
| car_parts/M/short | 0.990 ± 0.010 | 0.995 | 1.01 | 1.05 | 1.02 | **0.955** | 1.18 | 0.999 | 2.29 | 1.07 | 1.03 |
| covid_deaths/D/short | **0.037** ± 0.004 | 0.047 | 0.043 | 0.062 | 0.204 | 0.040 | 0.046 | 0.044 | 0.123 | 0.045 | 0.061 |
| electricity/15T/long | 0.075 ± 0.001 | 0.084 | 0.086 | **0.083** | 0.137 | 0.089 | 0.099 | 0.115 | 0.143 | 0.095 | 0.098 |
| electricity/15T/medium | 0.075 ± 0.001 | 0.083 | 0.087 | **0.080** | 0.138 | 0.092 | 0.103 | 0.106 | 0.142 | 0.095 | 0.096 |
| electricity/15T/short | 0.082 ± 0.000 | 0.082 | 0.082 | **0.079** | 0.130 | 0.104 | 0.128 | 0.120 | 0.152 | 0.092 | 0.099 |
| electricity/D/short | 0.056 ± 0.001 | **0.055** | 0.058 | 0.060 | 0.077 | 0.060 | 0.069 | 0.061 | 0.093 | 0.061 | 0.071 |
| electricity/H/long | 0.092 ± 0.003 | 0.098 | 0.102 | 0.089 | 0.101 | 0.112 | 0.103 | **0.086** | 0.128 | 0.105 | 0.107 |
| electricity/H/medium | 0.078 ± 0.002 | 0.081 | 0.084 | **0.073** | 0.082 | 0.091 | 0.087 | 0.082 | 0.109 | 0.087 | 0.088 |
| electricity/H/short | 0.061 ± 0.001 | 0.064 | 0.067 | **0.054** | 0.064 | 0.072 | 0.077 | 0.075 | 0.097 | 0.064 | 0.070 |
| electricity/W/short | 0.046 ± 0.001 | 0.047 | 0.048 | **0.049** | 0.088 | 0.051 | 0.062 | 0.077 | 0.159 | 0.049 | 0.052 |
| ett1/15T/long | **0.246** ± 0.003 | 0.298 | 0.296 | 0.283 | 0.358 | 0.260 | 0.358 | 0.273 | 0.352 | 0.400 | 0.450 |
| ett1/15T/medium | 0.251 ± 0.002 | 0.281 | 0.288 | 0.278 | 0.329 | **0.248** | 0.342 | 0.324 | 0.333 | 0.379 | 0.390 |
| ett1/15T/short | 0.161 ± 0.003 | **0.158** | 0.169 | 0.168 | 0.193 | 0.183 | 0.226 | 0.193 | 0.235 | 0.198 | 0.217 |
| ett1/D/short | 0.282 ± 0.004 | 0.287 | 0.283 | 0.281 | **0.280** | 0.292 | 0.286 | 0.301 | 0.416 | 0.387 | 0.360 |
| ett1/H/long | **0.263** ± 0.004 | 0.311 | 0.337 | 0.310 | 0.317 | 0.290 | 0.296 | 0.287 | 0.342 | 0.350 | 0.360 |
| ett1/H/medium | **0.253** ± 0.002 | 0.303 | 0.295 | 0.282 | 0.304 | 0.276 | 0.270 | 0.282 | 0.339 | 0.330 | 0.327 |
| ett1/H/short | **0.179** ± 0.002 | 0.181 | 0.189 | 0.192 | 0.209 | 0.194 | 0.189 | 0.197 | 0.250 | 0.194 | 0.222 |
| ett1/W/short | 0.306 ± 0.009 | 0.296 | 0.293 | 0.272 | 0.307 | **0.256** | 0.260 | 0.261 | 0.448 | 0.312 | 0.317 |
| ett2/15T/long | **0.097** ± 0.001 | 0.111 | 0.118 | 0.106 | 0.119 | 0.101 | 0.115 | 0.137 | 0.126 | 0.134 | 0.129 |
| ett2/15T/medium | **0.093** ± 0.001 | 0.110 | 0.119 | 0.105 | 0.112 | 0.098 | 0.105 | 0.109 | 0.128 | 0.122 | 0.117 |
| ett2/15T/short | 0.066 ± 0.001 | 0.067 | 0.070 | **0.065** | 0.077 | 0.073 | 0.080 | 0.078 | 0.093 | 0.071 | 0.073 |
| ett2/D/short | 0.092 ± 0.001 | 0.094 | **0.091** | 0.108 | 0.113 | 0.129 | 0.094 | 0.095 | 0.119 | 0.092 | 0.097 |
| ett2/H/long | 0.116 ± 0.004 | 0.117 | 0.121 | 0.125 | 0.125 | 0.136 | 0.125 | **0.110** | 0.144 | 0.136 | 0.122 |
| ett2/H/medium | 0.106 ± 0.002 | 0.115 | 0.118 | 0.110 | 0.126 | 0.128 | 0.118 | **0.100** | 0.139 | 0.132 | 0.136 |
| ett2/H/short | 0.065 ± 0.001 | **0.063** | 0.065 | 0.066 | 0.074 | 0.070 | 0.069 | 0.072 | 0.088 | 0.071 | 0.072 |
| ett2/W/short | 0.088 ± 0.002 | 0.088 | 0.094 | 0.110 | 0.111 | 0.120 | 0.109 | 0.087 | 0.200 | **0.077** | **0.077** |
| hierarchical_sales/D/short | **0.572** ± 0.002 | 0.576 | 0.582 | 0.576 | 0.573 | 0.593 | 0.580 | 0.575 | 0.792 | 0.600 | 0.619 |
| hierarchical_sales/W/short | 0.349 ± 0.003 | 0.353 | 0.354 | **0.330** | 0.343 | 0.342 | 0.359 | 0.357 | 0.725 | 0.367 | 0.367 |
| hospital/M/short | 0.052 ± 0.000 | 0.057 | 0.058 | **0.050** | 0.052 | **0.050** | 0.051 | 0.051 | 0.123 | 0.056 | 0.056 |

*Table 13.* CRPS scores of different zero-shot models on the GiftEval benchmark evaluation settings (Part 2/2). The models achieving the **best** and second-best scores are highlighted. Results for datasets that are part of the training data for the respective models are shaded in grey, and these results are excluded from the calculation of the best score. We trained TiRex with 6 different seeds and report the observed standard deviation in the plot.

| | TiRex | Chronos Bolt B | Chronos Bolt S | TimesFM 2.0 | TimesFM 1.0 | TabPFN-TS | Moirai L 1.1 | Moirai B 1.1 | TTM-r2 | Chronos B | Chronos S |
|---|---|---|---|---|---|---|---|---|---|---|---|
| jena_weather/10T/long | **0.053** ± 0.001 | 0.064 | 0.063 | 0.035 | 0.069 | 0.055 | 0.077 | 0.070 | 0.068 | 0.080 | 0.096 |
| jena_weather/10T/medium | **0.051** ± 0.001 | 0.057 | 0.060 | 0.031 | 0.067 | 0.057 | 0.072 | 0.068 | 0.069 | 0.076 | 0.089 |
| jena_weather/10T/short | **0.030** ± 0.001 | 0.033 | 0.037 | 0.016 | 0.036 | 0.035 | 0.051 | 0.053 | 0.045 | 0.044 | 0.047 |
| jena_weather/D/short | 0.046 ± 0.001 | **0.045** | 0.047 | 0.058 | 0.059 | 0.047 | 0.051 | 0.050 | 0.124 | 0.049 | 0.051 |
| jena_weather/H/long | **0.057** ± 0.002 | 0.062 | 0.068 | 0.068 | 0.089 | 0.066 | 0.061 | 0.065 | 0.084 | 0.074 | 0.072 |
| jena_weather/H/medium | **0.051** ± 0.002 | 0.054 | 0.058 | 0.066 | 0.065 | 0.060 | 0.058 | 0.057 | 0.073 | 0.070 | 0.069 |
| jena_weather/H/short | **0.041** ± 0.000 | 0.042 | 0.043 | 0.045 | 0.048 | 0.043 | 0.045 | 0.044 | 0.060 | 0.046 | 0.047 |
| kdd_cup_2018/D/short | 0.381 ± 0.005 | 0.372 | 0.373 | 0.378 | 0.380 | **0.359** | 0.381 | 0.376 | 0.452 | 0.503 | 0.512 |
| kdd_cup_2018/H/long | 0.341 ± 0.014 | 0.300 | 0.419 | 0.518 | 0.537 | **0.462** | 0.378 | 0.418 | 0.542 | 0.624 | 0.712 |
| kdd_cup_2018/H/medium | 0.337 ± 0.011 | 0.301 | 0.364 | 0.466 | 0.493 | **0.462** | 0.387 | 0.441 | 0.532 | 0.664 | 0.706 |
| kdd_cup_2018/H/short | 0.270 ± 0.004 | 0.246 | 0.267 | **0.376** | 0.446 | 0.437 | 0.362 | 0.389 | 0.514 | 0.459 | 0.459 |
| loop_seattle/5T/long | **0.090** ± 0.004 | 0.129 | 0.125 | 0.114 | 0.148 | 0.094 | 0.049 | 0.052 | 0.125 | 0.143 | 0.150 |
| loop_seattle/5T/medium | **0.083** ± 0.002 | 0.116 | 0.119 | 0.110 | 0.151 | 0.087 | 0.038 | 0.045 | 0.121 | 0.176 | 0.175 |
| loop_seattle/5T/short | **0.049** ± 0.000 | 0.055 | 0.055 | 0.051 | 0.075 | 0.052 | 0.041 | 0.046 | 0.068 | 0.070 | 0.070 |
| loop_seattle/D/short | 0.042 ± 0.000 | 0.044 | 0.045 | **0.041** | 0.043 | 0.044 | 0.045 | 0.044 | 0.101 | 0.045 | 0.048 |
| loop_seattle/H/long | **0.063** ± 0.001 | 0.076 | 0.082 | 0.066 | 0.097 | **0.063** | 0.074 | 0.080 | 0.097 | 0.082 | 0.089 |
| loop_seattle/H/medium | **0.065** ± 0.001 | 0.076 | 0.082 | 0.067 | 0.100 | **0.065** | 0.070 | 0.080 | 0.104 | 0.084 | 0.088 |
| loop_seattle/H/short | **0.059** ± 0.000 | 0.065 | 0.066 | **0.059** | 0.082 | 0.064 | 0.066 | 0.074 | 0.095 | 0.066 | 0.069 |
| m4_daily/D/short | 0.021 ± 0.000 | 0.021 | 0.021 | 0.021 | 0.021 | **0.024** | 0.030 | 0.040 | 0.035 | 0.022 | 0.021 |
| m4_hourly/H/short | 0.021 ± 0.000 | 0.025 | 0.020 | 0.011 | 0.021 | **0.028** | 0.020 | 0.022 | 0.040 | 0.024 | 0.025 |
| m4_monthly/M/short | 0.093 ± 0.000 | 0.094 | 0.094 | 0.067 | 0.097 | **0.088** | 0.095 | 0.094 | 0.177 | 0.104 | 0.103 |
| m4_quarterly/Q/short | **0.074** ± 0.000 | 0.077 | 0.078 | 0.062 | 0.085 | 0.075 | 0.073 | 0.073 | 0.139 | 0.083 | 0.084 |
| m4_weekly/W/short | 0.035 ± 0.001 | 0.038 | 0.038 | 0.042 | 0.041 | **0.036** | 0.046 | 0.048 | 0.069 | 0.037 | 0.040 |
| m4_yearly/A/short | 0.119 ± 0.002 | 0.121 | 0.128 | **0.091** | 0.117 | 0.113 | 0.104 | 0.105 | 0.197 | 0.135 | 0.139 |
| m_dense/D/short | 0.066 ± 0.002 | 0.069 | 0.072 | 0.060 | 0.070 | **0.057** | 0.095 | 0.104 | 0.151 | 0.075 | 0.087 |
| m_dense/H/long | 0.122 ± 0.003 | 0.170 | 0.146 | 0.127 | 0.135 | 0.164 | **0.114** | 0.122 | 0.222 | 0.135 | 0.133 |
| m_dense/H/medium | 0.121 ± 0.002 | 0.157 | 0.134 | 0.127 | 0.132 | 0.159 | **0.112** | 0.123 | 0.213 | 0.136 | 0.128 |
| m_dense/H/short | 0.130 ± 0.001 | **0.125** | 0.133 | 0.139 | 0.140 | 0.154 | 0.128 | 0.140 | 0.225 | 0.137 | 0.140 |
| restaurant/D/short | **0.254** ± 0.001 | 0.264 | 0.264 | 0.261 | 0.265 | 0.297 | 0.270 | 0.266 | 0.438 | 0.279 | 0.292 |
| saugeen/D/short | 0.382 ± 0.014 | **0.338** | 0.354 | 0.408 | 0.417 | 0.384 | 0.406 | 0.354 | 0.589 | 0.432 | 0.387 |
| saugeen/M/short | 0.303 ± 0.009 | 0.296 | 0.293 | 0.342 | 0.328 | **0.278** | 0.324 | 0.348 | 0.405 | 0.408 | 0.464 |
| saugeen/W/short | **0.353** ± 0.009 | 0.363 | 0.372 | 0.601 | 0.382 | 0.380 | 0.430 | 0.423 | 0.696 | 0.473 | 0.482 |
| solar/10T/long | **0.328** ± 0.008 | 0.443 | 0.497 | 0.498 | 0.703 | 0.352 | 0.771 | 0.903 | 0.545 | 0.748 | 0.901 |
| solar/10T/medium | **0.354** ± 0.016 | 0.436 | 0.453 | 0.516 | 0.623 | 0.359 | 0.747 | 0.832 | 0.573 | 0.686 | 0.796 |
| solar/10T/short | 0.542 ± 0.017 | 0.511 | **0.498** | 0.804 | 0.871 | 0.545 | 0.596 | 0.614 | 0.785 | 0.579 | 0.635 |
| solar/D/short | 0.281 ± 0.002 | 0.287 | 0.286 | 0.278 | 0.288 | **0.269** | 0.292 | 0.295 | 0.396 | 0.326 | 0.337 |
| solar/H/long | **0.243** ± 0.008 | 0.405 | 0.373 | 0.493 | 0.572 | 0.324 | 0.347 | 0.360 | 0.512 | 0.464 | 0.441 |
| solar/H/medium | **0.260** ± 0.010 | 0.368 | 0.356 | 0.376 | 0.425 | 0.324 | 0.346 | 0.331 | 0.493 | 0.356 | 0.361 |
| solar/H/short | **0.259** ± 0.009 | 0.298 | 0.303 | 0.406 | 0.403 | 0.358 | 0.333 | 0.338 | 0.468 | 0.334 | 0.345 |
| solar/W/short | 0.154 ± 0.008 | 0.133 | 0.136 | 0.171 | 0.157 | **0.124** | 0.213 | 0.235 | 0.531 | 0.161 | **0.124** |
| sz_taxi/15T/long | **0.197** ± 0.001 | 0.248 | 0.245 | 0.227 | 0.238 | 0.248 | 0.213 | 0.209 | 0.260 | 0.265 | 0.275 |
| sz_taxi/15T/medium | **0.202** ± 0.001 | 0.244 | 0.246 | 0.229 | 0.233 | 0.245 | 0.215 | 0.211 | 0.270 | 0.268 | 0.279 |
| sz_taxi/15T/short | **0.200** ± 0.000 | 0.202 | 0.203 | 0.199 | 0.206 | 0.215 | 0.215 | 0.213 | 0.268 | 0.236 | 0.241 |
| sz_taxi/H/short | 0.136 ± 0.000 | 0.136 | 0.137 | **0.135** | 0.137 | 0.144 | 0.146 | 0.143 | 0.183 | 0.149 | 0.149 |
| temperature_rain/D/short | 0.550 ± 0.002 | 0.538 | 0.544 | 0.586 | 0.581 | **0.565** | 0.479 | 0.535 | 0.791 | 0.610 | 0.627 |
| us_births/D/short | 0.021 ± 0.001 | 0.026 | 0.028 | 0.019 | 0.029 | **0.018** | 0.027 | 0.027 | 0.104 | 0.022 | 0.023 |
| us_births/M/short | 0.017 ± 0.002 | 0.019 | 0.016 | **0.011** | 0.013 | 0.013 | 0.016 | 0.015 | 0.036 | 0.018 | 0.013 |
| us_births/W/short | 0.013 ± 0.000 | 0.013 | 0.013 | 0.013 | 0.013 | **0.011** | 0.018 | 0.017 | 0.027 | **0.011** | **0.011** |

*Table 14.* MASE scores of TiRex compared with various task-specific and local models on the GiftEval benchmark evaluation settings (Part 1/2). Models achieving the **best** and second-best scores are highlighted. We trained TiRex with 6 different seeds and report the observed standard deviation in the plot.

| | TiRex | DeepAR | PatchTST | DLinear | TFT | N-Beats | Auto ARIMA | Auto Theta | Seas. Naive |
|---|---|---|---|---|---|---|---|---|---|
| bitbrains_fast_storage/5T/long | **0.916** ± 0.006 | 7.33 | 1.14 | 3.47 | 1.21 | 1.40 | 1.14 | 1.61 | 1.14 |
| bitbrains_fast_storage/5T/medium | **1.00** ± 0.011 | 8.50 | 1.20 | 3.33 | 1.38 | 1.60 | 1.22 | 1.42 | 1.22 |
| bitbrains_fast_storage/5T/short | **0.692** ± 0.007 | 0.945 | 0.973 | 1.48 | 0.996 | 1.09 | 1.14 | 1.15 | 1.14 |
| bitbrains_fast_storage/H/short | **1.06** ± 0.013 | 6.06 | 1.34 | 2.65 | 1.73 | 1.37 | 1.43 | 1.35 | 1.30 |
| bitbrains_rnd/5T/long | **3.35** ± 0.013 | 4.44 | 3.72 | 6.35 | 3.71 | 3.95 | 3.50 | 4.11 | 3.50 |
| bitbrains_rnd/5T/medium | **4.40** ± 0.007 | 4.89 | 4.65 | 7.08 | 4.81 | 4.79 | 4.54 | 4.88 | 4.54 |
| bitbrains_rnd/5T/short | **1.66** ± 0.004 | 2.10 | 1.98 | 2.63 | 2.27 | 2.18 | 1.97 | 2.07 | 1.97 |
| bitbrains_rnd/H/short | 5.84 ± 0.011 | 6.06 | 6.11 | 8.50 | 6.19 | 6.16 | 6.08 | **5.75** | 6.04 |
| bizitobs_application/10S/long | 3.67 ± 0.069 | 4.47 | 3.19 | 4.25 | 14.8 | 3.83 | 36400 | **2.93** | 36400 |
| bizitobs_application/10S/medium | 2.85 ± 0.057 | 3.22 | 2.77 | 3.88 | 13.8 | 2.57 | 2.69 | **1.78** | 2.69 |
| bizitobs_application/10S/short | 1.40 ± 0.112 | 4.16 | 2.24 | 7.65 | 9.11 | 2.57 | 2.24 | **1.11** | 2.24 |
| bizitobs_l2c/5T/long | 1.19 ± 0.040 | 1.32 | **0.686** | 1.12 | 1.06 | 0.968 | 1.45 | 1.24 | 1.45 |
| bizitobs_l2c/5T/medium | 0.849 ± 0.028 | 1.21 | 0.787 | 0.894 | **0.786** | 0.935 | 1.24 | 0.868 | 1.24 |
| bizitobs_l2c/5T/short | 0.290 ± 0.005 | 0.613 | 0.266 | **0.243** | 0.278 | 0.297 | 0.986 | 0.292 | 0.986 |
| bizitobs_l2c/H/long | **0.590** ± 0.022 | 0.727 | 0.617 | 0.744 | 0.599 | 0.811 | 1.54 | 1.41 | 4.04 |
| bizitobs_l2c/H/medium | **0.525** ± 0.020 | 0.737 | 0.537 | 0.676 | 0.693 | 0.701 | 1.56 | 1.65 | 1.65 |
| bizitobs_l2c/H/short | 0.528 ± 0.021 | 1.50 | **0.495** | 0.640 | 0.862 | 0.536 | 1.25 | 1.19 | 1.21 |
| bizitobs_service/10S/long | 1.57 ± 0.057 | 3.96 | 1.69 | 2.29 | 1.75 | 2.62 | **1.37** | 1.62 | **1.37** |
| bizitobs_service/10S/medium | 1.32 ± 0.058 | 2.17 | 1.49 | 2.23 | 1.68 | 2.59 | 1.32 | **1.06** | 1.32 |
| bizitobs_service/10S/short | 0.884 ± 0.052 | 2.67 | 1.24 | 1.87 | 2.15 | 1.12 | 1.23 | **0.791** | 1.23 |
| car_parts/M/short | 0.838 ± 0.004 | 0.835 | **0.797** | 0.997 | 0.807 | 0.810 | 0.958 | 1.23 | 1.20 |
| covid_deaths/D/short | 39.5 ± 0.803 | 50.7 | 37.7 | 33.2 | 32.9 | 32.8 | **31.4** | 45.4 | 46.9 |
| electricity/15T/long | 0.891 ± 0.008 | 2.28 | **0.960** | 1.24 | 1.03 | 1.15 | 1.16 | 1.50 | 1.16 |
| electricity/15T/medium | 0.841 ± 0.006 | 1.39 | **0.977** | 1.31 | 1.11 | 1.62 | 1.15 | 1.43 | 1.15 |
| electricity/15T/short | 0.945 ± 0.008 | 1.67 | 1.47 | 1.64 | 2.07 | 1.69 | 1.72 | **1.35** | 1.72 |
| electricity/D/short | **1.43** ± 0.010 | 1.89 | 1.85 | 3.56 | 1.86 | 1.85 | 1.82 | 1.88 | 1.99 |
| electricity/H/long | 1.21 ± 0.018 | 2.67 | **1.39** | 2.21 | 1.41 | 1.42 | 1.52 | 2.05 | 1.52 |
| electricity/H/medium | 1.08 ± 0.010 | 6.76 | **1.16** | 2.26 | 1.31 | 1.35 | 1.39 | 1.78 | 1.39 |
| electricity/H/short | 0.869 ± 0.009 | 1.23 | **1.08** | 1.31 | 1.29 | 1.44 | 1.36 | 1.74 | 1.36 |
| electricity/W/short | 1.46 ± 0.010 | 2.25 | 1.96 | **1.84** | 2.10 | 2.10 | 2.09 | 2.14 | 2.09 |
| ett1/15T/long | **1.05** ± 0.009 | 9.34 | 1.10 | 1.19 | 1.34 | 1.42 | 1.19 | 1.76 | 1.19 |
| ett1/15T/medium | **1.04** ± 0.009 | 1.35 | 1.08 | 1.20 | 1.08 | 1.41 | 1.19 | 1.25 | 1.19 |
| ett1/15T/short | **0.706** ± 0.007 | 1.44 | 0.835 | 0.804 | 1.05 | 0.870 | 0.934 | 0.863 | 0.934 |
| ett1/D/short | 1.71 ± 0.016 | 1.69 | **1.68** | 1.98 | 1.86 | 2.04 | 1.85 | 1.75 | 1.78 |
| ett1/H/long | **1.34** ± 0.030 | 2.68 | 1.47 | 1.46 | 1.55 | 1.96 | 1.65 | 2.51 | 1.48 |
| ett1/H/medium | **1.25** ± 0.017 | 3.12 | 1.39 | 1.66 | 1.58 | 1.67 | 1.57 | 1.84 | 1.57 |
| ett1/H/short | **0.827** ± 0.007 | 1.06 | 0.893 | 0.945 | 0.947 | 0.930 | 0.995 | 1.28 | 0.977 |
| ett1/W/short | 1.72 ± 0.044 | 4.16 | 1.89 | 2.16 | **1.61** | 1.63 | 1.99 | 1.89 | 1.77 |
| ett2/15T/long | **0.932** ± 0.012 | 3.70 | 0.961 | 1.10 | 1.15 | 0.980 | 1.01 | 1.10 | 1.01 |
| ett2/15T/medium | **0.910** ± 0.010 | 3.27 | 0.933 | 1.21 | 1.10 | 1.09 | 1.05 | 1.04 | 1.05 |
| ett2/15T/short | **0.749** ± 0.010 | 4.11 | 0.879 | 0.937 | 1.06 | 1.01 | 1.07 | 0.832 | 1.07 |
| ett2/D/short | **1.28** ± 0.019 | 3.64 | 2.17 | 3.25 | 1.31 | 1.54 | 1.45 | 1.85 | 1.39 |
| ett2/H/long | 1.16 ± 0.037 | 2.49 | 1.43 | 1.58 | 1.45 | 1.26 | 1.28 | 1.46 | **1.13** |
| ett2/H/medium | **1.05** ± 0.021 | 2.52 | 1.27 | 1.36 | 1.32 | **1.05** | 1.46 | 1.30 | 1.24 |
| ett2/H/short | **0.742** ± 0.006 | 1.48 | 0.858 | 0.817 | 0.956 | 0.819 | 0.952 | 1.02 | 0.923 |
| ett2/W/short | 0.797 ± 0.040 | 7.17 | 1.49 | 1.93 | 1.60 | 2.69 | 1.13 | 1.41 | **0.779** |
| hierarchical_sales/D/short | **0.744** ± 0.002 | 0.757 | 0.756 | 0.860 | 0.771 | 0.773 | 0.813 | 0.932 | 1.13 |
| hierarchical_sales/W/short | **0.721** ± 0.001 | 0.781 | 0.771 | 0.993 | 0.793 | 0.778 | 0.850 | 0.849 | 1.03 |
| hospital/M/short | 0.767 ± 0.003 | 0.834 | 0.820 | 0.811 | 0.833 | 0.771 | 0.826 | **0.761** | 0.921 |

*Table 15.* MASE scores of TiRex compared with various task-specific and local models on the GiftEval benchmark evaluation settings (Part 2/2). Models achieving the **best** and second-best scores are highlighted. We trained TiRex with 6 different seeds and report the observed standard deviation in the plot.

| | TiRex | DeepAR | PatchTST | DLinear | TFT | N-Beats | Auto ARIMA | Auto Theta | Seas. Naive |
|---|---|---|---|---|---|---|---|---|---|
| jena_weather/10T/long | **0.641** ± 0.015 | 3.15 | 1.07 | 0.912 | 0.741 | 0.855 | 0.761 | 0.990 | 0.761 |
| jena_weather/10T/medium | **0.610** ± 0.005 | 1.20 | 0.943 | 1.17 | 0.737 | 0.749 | 0.716 | 0.806 | 0.716 |
| jena_weather/10T/short | **0.297** ± 0.006 | 0.574 | 0.552 | 1.96 | 0.450 | 0.527 | 0.743 | 0.368 | 0.743 |
| jena_weather/D/short | **1.02** ± 0.014 | 1.30 | 1.39 | 1.60 | 1.80 | 1.83 | 1.45 | 1.60 | 1.57 |
| jena_weather/H/long | **0.987** ± 0.031 | 6.89 | 1.31 | 1.90 | 1.15 | 1.13 | 1.98 | 2.64 | 1.27 |
| jena_weather/H/medium | **0.828** ± 0.023 | 1.30 | 1.09 | 0.997 | 0.939 | 0.902 | 1.45 | 1.36 | 0.889 |
| jena_weather/H/short | **0.516** ± 0.003 | 18.8 | 0.641 | 0.982 | 0.634 | 0.763 | 1.08 | 0.878 | 0.723 |
| kdd_cup_2018/D/short | 1.21 ± 0.009 | 1.23 | 1.22 | 1.23 | 1.21 | 1.35 | **1.18** | 1.38 | 1.50 |
| kdd_cup_2018/H/long | 0.759 ± 0.027 | 3.34 | **1.02** | 1.09 | 1.11 | 1.18 | 1.18 | 1.37 | 1.34 |
| kdd_cup_2018/H/medium | 0.825 ± 0.024 | 1.17 | **1.05** | 1.12 | 1.16 | 1.29 | 1.42 | 1.33 | 1.43 |
| kdd_cup_2018/H/short | 0.657 ± 0.006 | 1.28 | **1.12** | 1.13 | 1.15 | 1.30 | 1.34 | 1.27 | 1.34 |
| loop_seattle/5T/long | 1.02 ± 0.038 | 1.96 | 1.06 | 1.17 | **0.977** | 1.05 | 1.25 | 1.44 | 1.25 |
| loop_seattle/5T/medium | **0.941** ± 0.019 | 1.27 | 1.05 | 1.12 | 1.01 | 1.05 | 1.15 | 2.06 | 1.15 |
| loop_seattle/5T/short | **0.572** ± 0.005 | 0.803 | 0.744 | 0.895 | 0.731 | 0.735 | 0.762 | 0.780 | 0.762 |
| loop_seattle/D/short | **0.878** ± 0.005 | 1.08 | 0.934 | 0.900 | 0.973 | 0.898 | 1.49 | 1.39 | 1.73 |
| loop_seattle/H/long | **0.917** ± 0.011 | 0.985 | 0.979 | 1.03 | 0.972 | 0.932 | 2.59 | 2.02 | 1.55 |
| loop_seattle/H/medium | **0.944** ± 0.012 | 1.03 | 1.03 | 1.20 | 0.971 | 1.03 | 2.00 | 1.61 | 1.48 |
| loop_seattle/H/short | **0.850** ± 0.005 | 0.941 | 1.07 | 1.13 | 1.04 | 1.03 | 1.29 | 1.40 | 1.29 |
| m4_daily/D/short | 3.15 ± 0.063 | 4.58 | **3.22** | 3.42 | 3.29 | 3.35 | 3.26 | 3.34 | 3.28 |
| m4_hourly/H/short | 0.719 ± 0.026 | 3.53 | 1.40 | 1.69 | 2.47 | 1.34 | **1.03** | 2.46 | 1.19 |
| m4_monthly/M/short | 0.929 ± 0.004 | 3.18 | 1.06 | 1.13 | 1.21 | 1.05 | 0.976 | **0.966** | 1.26 |
| m4_quarterly/Q/short | **1.18** ± 0.013 | 1.44 | 1.32 | 1.46 | 1.30 | 1.21 | 1.28 | 1.19 | 1.60 |
| m4_weekly/W/short | 1.90 ± 0.029 | 4.62 | 2.34 | 4.64 | 2.68 | **1.97** | 2.36 | 2.66 | 2.78 |
| m4_yearly/A/short | 3.45 ± 0.075 | 3.40 | 3.29 | 4.16 | **3.09** | 3.15 | 3.71 | 3.11 | 3.97 |
| m_dense/D/short | **0.688** ± 0.016 | 0.793 | 0.732 | 1.01 | 0.799 | 0.706 | 1.34 | 1.22 | 1.67 |
| m_dense/H/long | 0.730 ± 0.013 | 0.805 | 0.738 | 1.24 | **0.723** | 1.18 | 1.21 | 2.29 | 1.48 |
| m_dense/H/medium | 0.736 ± 0.016 | 0.738 | 0.757 | 0.930 | **0.732** | 0.890 | 1.27 | 1.74 | 1.57 |
| m_dense/H/short | **0.788** ± 0.009 | 0.795 | 1.03 | 1.04 | 0.878 | 0.915 | 1.49 | 1.69 | 1.49 |
| restaurant/D/short | **0.677** ± 0.002 | 0.713 | 0.690 | 0.706 | 0.750 | 0.712 | 0.929 | 0.843 | 1.01 |
| saugeen/D/short | **3.12** ± 0.108 | 4.31 | 3.28 | 4.20 | 3.22 | 3.28 | 3.74 | 3.60 | 3.41 |
| saugeen/M/short | 0.750 ± 0.020 | 1.63 | 0.893 | 0.955 | 0.865 | 0.758 | **0.725** | 0.912 | 0.976 |
| saugeen/W/short | **1.18** ± 0.028 | 1.31 | 1.55 | 1.81 | 1.55 | 1.54 | 1.55 | 2.12 | 1.99 |
| solar/10T/long | **0.828** ± 0.021 | 1.28 | 0.912 | 1.18 | 1.00 | 2.03 | 0.871 | 4.53 | 0.871 |
| solar/10T/medium | **0.879** ± 0.037 | 1.21 | 0.913 | 1.08 | 0.931 | 1.98 | 0.927 | 2.69 | 0.927 |
| solar/10T/short | 1.05 ± 0.032 | 1.47 | 2.20 | 1.24 | 1.11 | **0.848** | 1.11 | 1.80 | 1.11 |
| solar/D/short | 0.971 ± 0.005 | 2.49 | **0.962** | 1.03 | 0.999 | 1.21 | 1.01 | 1.05 | 1.16 |
| solar/H/long | **0.697** ± 0.024 | 0.972 | 0.978 | 1.35 | 1.12 | 2.21 | 0.995 | 5.24 | 1.07 |
| solar/H/medium | **0.731** ± 0.026 | 0.992 | 0.965 | 1.17 | 0.884 | 2.12 | 0.848 | 2.87 | 0.935 |
| solar/H/short | **0.699** ± 0.019 | 1.02 | 0.954 | 1.06 | 0.960 | 1.04 | 0.952 | 2.05 | 0.952 |
| solar/W/short | 1.13 ± 0.075 | 1.69 | 1.10 | 1.13 | **0.691** | 2.33 | 1.12 | 1.15 | 1.47 |
| sz_taxi/15T/long | **0.509** ± 0.003 | 0.733 | 0.761 | 0.841 | 0.535 | 0.666 | 0.598 | 0.759 | 0.691 |
| sz_taxi/15T/medium | **0.536** ± 0.001 | 0.558 | 0.588 | 0.629 | 0.548 | 0.662 | 0.632 | 0.716 | 0.713 |
| sz_taxi/15T/short | **0.544** ± 0.001 | 0.602 | 0.560 | 0.582 | 0.603 | 0.604 | 0.764 | 0.649 | 0.764 |
| sz_taxi/H/short | **0.563** ± 0.002 | 0.576 | 0.591 | 0.667 | 0.595 | 0.624 | 0.624 | 0.691 | 0.738 |
| temperature_rain/D/short | 1.34 ± 0.003 | 1.72 | 1.51 | 1.83 | **1.44** | 1.90 | 1.71 | 1.93 | 2.01 |
| us_births/D/short | 0.404 ± 0.017 | 0.535 | 0.487 | 0.645 | **0.315** | 0.456 | 1.58 | 1.63 | 1.86 |
| us_births/M/short | 0.808 ± 0.066 | 0.760 | 0.782 | 1.17 | 0.871 | 0.928 | **0.466** | 0.883 | 0.761 |
| us_births/W/short | **1.08** ± 0.032 | 1.45 | 1.23 | 1.46 | 1.59 | 1.40 | 1.48 | 1.49 | 1.56 |

*Table 16.* CRPS scores of TiRex compared with various task-specific and local models on the GiftEval benchmark evaluation settings (Part 1/2). Models achieving the **best** and second-best scores are highlighted. We trained TiRex with 6 different seeds and report the observed standard deviation in the plot.

| | TiRex | DeepAR | PatchTST | DLinear | TFT | N-Beats | Auto ARIMA | Auto Theta | Seas. Naive |
|---|---|---|---|---|---|---|---|---|---|
| bitbrains_fast_storage/5T/long | **0.655** $\pm$ 0.028 | 1.01 | 0.669 | 1.33 | 0.734 | 0.791 | 1.29 | 1.36 | 1.29 |
| bitbrains_fast_storage/5T/medium | **0.605** $\pm$ 0.016 | 0.990 | 0.642 | 0.967 | 0.610 | 0.841 | 1.27 | 1.45 | 1.27 |
| bitbrains_fast_storage/5T/short | **0.408** $\pm$ 0.005 | 0.493 | 0.471 | 0.577 | 0.451 | 0.622 | 1.21 | 0.731 | 1.21 |
| bitbrains_fast_storage/H/short | 0.699 $\pm$ 0.013 | 0.778 | **0.549** | 0.803 | 0.595 | 0.812 | 0.844 | 1.15 | 1.08 |
| bitbrains_rnd/5T/long | 0.660 $\pm$ 0.065 | 0.672 | 0.664 | 1.30 | **0.624** | 1.06 | 1.29 | 1.60 | 1.29 |
| bitbrains_rnd/5T/medium | **0.594** $\pm$ 0.019 | 0.647 | 0.620 | 1.01 | 0.628 | 0.699 | 1.26 | 1.47 | 1.26 |
| bitbrains_rnd/5T/short | **0.403** $\pm$ 0.001 | 0.557 | 0.474 | 0.571 | 0.486 | 0.656 | 1.10 | 0.741 | 1.10 |
| bitbrains_rnd/H/short | 0.631 $\pm$ 0.013 | **0.585** | 0.603 | 1.07 | 0.650 | 0.715 | 0.874 | 1.38 | 1.30 |
| bizitobs_application/10S/long | 0.053 $\pm$ 0.004 | 0.083 | 0.054 | 0.070 | 0.056 | 0.062 | 0.973 | **0.035** | 0.973 |
| bizitobs_application/10S/medium | 0.041 $\pm$ 0.003 | 0.053 | 0.047 | 0.056 | 0.047 | 0.047 | 0.042 | **0.024** | 0.042 |
| bizitobs_application/10S/short | 0.013 $\pm$ 0.001 | 0.064 | 0.022 | 0.079 | 0.090 | 0.043 | 0.035 | **0.010** | 0.035 |
| bizitobs_l2c/5T/long | 0.581 $\pm$ 0.032 | 0.719 | **0.324** | 0.653 | 0.472 | 0.546 | 0.674 | 0.632 | 0.674 |
| bizitobs_l2c/5T/medium | 0.366 $\pm$ 0.015 | 0.589 | **0.332** | 0.490 | 0.346 | 0.505 | 0.530 | 0.415 | 0.530 |
| bizitobs_l2c/5T/short | 0.078 $\pm$ 0.002 | 0.179 | **0.074** | 0.080 | 0.077 | 0.100 | 0.262 | 0.080 | 0.262 |
| bizitobs_l2c/H/long | **0.276** $\pm$ 0.010 | 0.338 | 0.291 | 0.422 | 0.286 | 0.479 | 0.787 | 0.819 | 1.82 |
| bizitobs_l2c/H/medium | **0.253** $\pm$ 0.008 | 0.345 | 0.263 | 0.398 | 0.345 | 0.420 | 0.813 | 0.892 | 1.42 |
| bizitobs_l2c/H/short | 0.227 $\pm$ 0.011 | 0.789 | **0.217** | 0.336 | 0.401 | 0.288 | 0.547 | 0.507 | 0.536 |
| bizitobs_service/10S/long | 0.054 $\pm$ 0.002 | 0.070 | 0.057 | 0.067 | 0.056 | 0.061 | 0.056 | **0.052** | 0.056 |
| bizitobs_service/10S/medium | 0.034 $\pm$ 0.002 | 0.044 | 0.045 | 0.053 | 0.044 | 0.043 | 0.049 | **0.027** | 0.049 |
| bizitobs_service/10S/short | **0.013** $\pm$ 0.000 | 0.032 | 0.025 | 0.032 | 0.025 | 0.021 | 0.040 | **0.013** | 0.040 |
| car_parts/M/short | 0.990 $\pm$ 0.010 | 0.953 | 1.00 | 1.26 | **0.890** | 1.02 | 1.29 | 1.34 | 1.72 |
| covid_deaths/D/short | 0.037 $\pm$ 0.004 | 0.177 | 0.067 | 0.063 | 0.037 | 0.071 | **0.030** | 0.095 | 0.125 |
| electricity/15T/long | 0.075 $\pm$ 0.001 | 0.155 | **0.081** | 0.129 | 0.084 | 0.123 | 0.129 | 0.401 | 0.129 |
| electricity/15T/medium | 0.075 $\pm$ 0.001 | 0.119 | **0.086** | 0.142 | 0.094 | 0.176 | 0.124 | 0.328 | 0.124 |
| electricity/15T/short | 0.082 $\pm$ 0.000 | 0.152 | **0.134** | 0.177 | 0.184 | 0.180 | 0.165 | 0.140 | 0.165 |
| electricity/D/short | **0.056** $\pm$ 0.001 | 0.078 | 0.083 | 0.169 | 0.084 | 0.110 | 0.083 | 0.088 | 0.122 |
| electricity/H/long | 0.092 $\pm$ 0.003 | 0.176 | 0.104 | 0.203 | **0.094** | 0.126 | 0.190 | 0.300 | 0.190 |
| electricity/H/medium | 0.078 $\pm$ 0.002 | 0.454 | **0.081** | 0.206 | 0.091 | 0.115 | 0.156 | 0.254 | 0.156 |
| electricity/H/short | 0.061 $\pm$ 0.001 | 0.094 | **0.079** | 0.112 | 0.089 | 0.123 | 0.109 | 0.177 | 0.109 |
| electricity/W/short | 0.046 $\pm$ 0.001 | **0.092** | 0.095 | 0.111 | 0.107 | 0.123 | 0.100 | 0.101 | 0.099 |
| ett1/15T/long | **0.246** $\pm$ 0.003 | 2.22 | 0.247 | 0.343 | 0.280 | 0.431 | 0.396 | 1.39 | 0.396 |
| ett1/15T/medium | 0.251 $\pm$ 0.002 | 0.315 | 0.250 | 0.347 | **0.247** | 0.430 | 0.352 | 1.13 | 0.352 |
| ett1/15T/short | **0.161** $\pm$ 0.003 | 0.320 | 0.191 | 0.233 | 0.245 | 0.254 | 0.241 | 0.410 | 0.241 |
| ett1/D/short | 0.282 $\pm$ 0.004 | 0.293 | 0.304 | 0.376 | 0.330 | 0.387 | **0.279** | 0.341 | 0.515 |
| ett1/H/long | **0.263** $\pm$ 0.004 | 0.469 | 0.297 | 0.363 | 0.313 | 0.567 | 0.430 | 1.94 | 0.616 |
| ett1/H/medium | **0.253** $\pm$ 0.002 | 0.535 | 0.273 | 0.455 | 0.316 | 0.450 | 0.384 | 1.65 | 0.540 |
| ett1/H/short | **0.179** $\pm$ 0.002 | 0.233 | 0.190 | 0.256 | 0.199 | 0.249 | 0.223 | 0.668 | 0.250 |
| ett1/W/short | 0.306 $\pm$ 0.009 | 0.686 | 0.323 | 0.447 | 0.406 | 0.372 | **0.305** | 0.319 | 0.338 |
| ett2/15T/long | **0.097** $\pm$ 0.001 | 0.304 | 0.098 | 0.141 | 0.109 | 0.126 | 0.165 | 0.169 | 0.165 |
| ett2/15T/medium | **0.093** $\pm$ 0.001 | 0.258 | 0.094 | 0.151 | 0.104 | 0.138 | 0.143 | 0.150 | 0.143 |
| ett2/15T/short | **0.066** $\pm$ 0.001 | 0.378 | 0.076 | 0.102 | 0.081 | 0.109 | 0.096 | 0.077 | 0.096 |
| ett2/D/short | **0.092** $\pm$ 0.001 | 0.207 | 0.131 | 0.218 | 0.096 | 0.140 | 0.125 | 0.164 | 0.205 |
| ett2/H/long | **0.116** $\pm$ 0.004 | 0.196 | 0.130 | 0.165 | 0.138 | 0.156 | 0.272 | 0.336 | 0.287 |
| ett2/H/medium | **0.106** $\pm$ 0.002 | 0.281 | 0.125 | 0.166 | 0.122 | 0.130 | 0.245 | 0.284 | 0.241 |
| ett2/H/short | **0.065** $\pm$ 0.001 | 0.122 | 0.074 | 0.088 | 0.078 | 0.091 | 0.089 | 0.102 | 0.094 |
| ett2/W/short | **0.088** $\pm$ 0.002 | 0.728 | 0.142 | 0.194 | 0.160 | 0.294 | 0.136 | 0.160 | 0.169 |
| hierarchical_sales/D/short | **0.572** $\pm$ 0.002 | 0.600 | 0.590 | 0.817 | 0.600 | 0.728 | 0.735 | 0.967 | 2.36 |
| hierarchical_sales/W/short | **0.349** $\pm$ 0.003 | 0.379 | 0.358 | 0.582 | 0.382 | 0.439 | 0.485 | 0.474 | 1.03 |
| hospital/M/short | **0.052** $\pm$ 0.000 | 0.062 | 0.064 | 0.076 | 0.058 | 0.068 | 0.060 | 0.055 | 0.062 |

*Table 17.* CRPS scores of TiRex compared with various task-specific and local models on the GiftEval benchmark evaluation settings (Part 2/2). Models achieving the **best** and second-best scores are highlighted. We trained TiRex with 6 different seeds and report the observed standard deviation in the plot.

| | TiRex | DeepAR | PatchTST | DLinear | TFT | N-Beats | Auto ARIMA | Auto Theta | Seas. Naive |
|---|---|---|---|---|---|---|---|---|---|
| jena_weather/10T/long | 0.053 ± 0.001 | 0.143 | 0.066 | 0.093 | **0.052** | 0.134 | 0.304 | 0.424 | 0.304 |
| jena_weather/10T/medium | **0.051** ± 0.001 | 0.073 | 0.065 | 0.098 | 0.052 | 0.089 | 0.277 | 0.350 | 0.277 |
| jena_weather/10T/short | **0.030** ± 0.001 | 0.063 | 0.064 | 0.129 | 0.069 | 0.104 | 0.155 | 0.130 | 0.155 |
| jena_weather/D/short | **0.046** ± 0.001 | 0.062 | 0.053 | 0.073 | 0.069 | 0.193 | 0.080 | 0.082 | 0.297 |
| jena_weather/H/long | **0.057** ± 0.002 | 0.197 | 0.076 | 0.139 | 0.090 | 0.131 | 0.230 | 1.29 | 0.598 |
| jena_weather/H/medium | **0.051** ± 0.002 | 0.078 | 0.069 | 0.093 | 0.073 | 0.097 | 0.211 | 0.832 | 0.486 |
| jena_weather/H/short | **0.041** ± 0.000 | 0.699 | 0.050 | 0.086 | 0.048 | 0.098 | 0.143 | 0.296 | 0.173 |
| kdd_cup_2018/D/short | 0.381 ± 0.005 | 0.383 | 0.401 | 0.482 | **0.380** | 0.529 | 0.393 | 0.459 | 0.888 |
| kdd_cup_2018/H/long | 0.341 ± 0.014 | 1.09 | **0.477** | 0.583 | 0.503 | 0.660 | 1.05 | 0.970 | 1.25 |
| kdd_cup_2018/H/medium | 0.337 ± 0.011 | **0.442** | **0.442** | 0.548 | 0.472 | 0.660 | 0.851 | 0.791 | 0.949 |
| kdd_cup_2018/H/short | 0.270 ± 0.004 | 0.517 | **0.457** | 0.581 | 0.467 | 0.683 | 0.559 | 0.531 | 0.559 |
| loop_seattle/5T/long | 0.090 ± 0.004 | 0.184 | 0.095 | 0.131 | **0.088** | 0.117 | 0.137 | 0.231 | 0.137 |
| loop_seattle/5T/medium | **0.083** ± 0.002 | 0.118 | 0.095 | 0.126 | 0.092 | 0.118 | 0.123 | 0.240 | 0.123 |
| loop_seattle/5T/short | **0.049** ± 0.000 | 0.072 | 0.066 | 0.099 | 0.065 | 0.080 | 0.081 | 0.082 | 0.081 |
| loop_seattle/D/short | **0.042** ± 0.000 | 0.052 | 0.046 | 0.053 | 0.048 | 0.053 | 0.078 | 0.072 | 0.131 |
| loop_seattle/H/long | **0.063** ± 0.001 | 0.068 | 0.069 | 0.088 | 0.068 | 0.080 | 0.193 | 0.468 | 0.245 |
| loop_seattle/H/medium | **0.065** ± 0.001 | 0.072 | 0.071 | 0.104 | 0.069 | 0.088 | 0.154 | 0.390 | 0.206 |
| loop_seattle/H/short | **0.059** ± 0.000 | 0.066 | 0.076 | 0.099 | 0.073 | 0.090 | 0.108 | 0.165 | 0.108 |
| m4_daily/D/short | 0.021 ± 0.000 | 0.030 | **0.023** | 0.029 | **0.023** | 0.029 | **0.023** | 0.024 | 0.026 |
| m4_hourly/H/short | 0.021 ± 0.000 | 0.133 | 0.039 | 0.055 | 0.040 | 0.050 | **0.034** | 0.041 | 0.040 |
| m4_monthly/M/short | 0.093 ± 0.000 | 0.184 | 0.102 | 0.129 | 0.113 | 0.122 | **0.098** | **0.098** | 0.126 |
| m4_quarterly/Q/short | **0.074** ± 0.000 | 0.083 | 0.083 | 0.110 | 0.083 | 0.096 | 0.082 | 0.079 | 0.099 |
| m4_weekly/W/short | 0.035 ± 0.001 | 0.062 | **0.040** | 0.070 | 0.049 | 0.047 | 0.050 | 0.053 | 0.073 |
| m4_yearly/A/short | 0.119 ± 0.002 | 0.113 | 0.117 | 0.168 | **0.110** | 0.134 | 0.130 | 0.115 | 0.138 |
| m_dense/D/short | **0.066** ± 0.002 | 0.076 | 0.070 | 0.123 | 0.077 | 0.087 | 0.135 | 0.126 | 0.294 |
| m_dense/H/long | 0.122 ± 0.003 | 0.130 | 0.120 | 0.259 | **0.115** | 0.243 | 0.270 | 1.43 | 0.552 |
| m_dense/H/medium | 0.121 ± 0.002 | 0.118 | 0.127 | 0.191 | **0.114** | 0.184 | 0.255 | 1.21 | 0.479 |
| m_dense/H/short | 0.130 ± 0.001 | **0.128** | 0.173 | 0.214 | 0.139 | 0.190 | 0.281 | 0.549 | 0.281 |
| restaurant/D/short | **0.254** ± 0.001 | 0.270 | 0.262 | 0.340 | 0.284 | 0.342 | 0.362 | 0.329 | 0.907 |
| saugeen/D/short | **0.382** ± 0.014 | 0.572 | 0.408 | 0.613 | 0.419 | 0.478 | 0.564 | 0.669 | 0.754 |
| saugeen/M/short | **0.303** ± 0.009 | 0.689 | 0.372 | 0.490 | 0.340 | 0.388 | 0.326 | 0.373 | 0.445 |
| saugeen/W/short | **0.353** ± 0.009 | 0.397 | 0.484 | 0.673 | 0.491 | 0.574 | 0.549 | 0.734 | 0.855 |
| solar/10T/long | **0.328** ± 0.008 | 0.549 | 0.339 | 0.585 | 0.379 | 1.01 | 0.786 | 6.64 | 0.786 |
| solar/10T/medium | **0.354** ± 0.016 | 0.485 | 0.356 | 0.552 | 0.362 | 1.00 | 0.771 | 5.67 | 0.771 |
| solar/10T/short | **0.542** ± 0.017 | 0.933 | 1.37 | 0.824 | 0.618 | 0.576 | 0.860 | 2.36 | 0.860 |
| solar/D/short | 0.281 ± 0.002 | 0.682 | 0.287 | 0.383 | **0.277** | 0.450 | 0.282 | 0.286 | 0.757 |
| solar/H/long | **0.243** ± 0.008 | 0.381 | 0.353 | 0.612 | 0.401 | 1.00 | 0.607 | 7.32 | 1.47 |
| solar/H/medium | **0.260** ± 0.010 | 0.352 | 0.344 | 0.552 | 0.330 | 1.00 | 0.557 | 6.13 | 1.27 |
| solar/H/short | **0.259** ± 0.009 | 0.389 | 0.340 | 0.507 | 0.367 | 0.498 | 0.628 | 2.33 | 0.628 |
| solar/W/short | 0.154 ± 0.008 | 0.242 | 0.162 | 0.210 | **0.114** | 0.432 | 0.152 | 0.155 | 0.236 |
| sz_taxi/15T/long | **0.197** ± 0.001 | 0.286 | 0.281 | 0.396 | 0.241 | 0.323 | 0.398 | 0.629 | 0.554 |
| sz_taxi/15T/medium | **0.202** ± 0.001 | 0.210 | 0.220 | 0.296 | 0.206 | 0.314 | 0.351 | 0.529 | 0.454 |
| sz_taxi/15T/short | **0.200** ± 0.000 | 0.219 | 0.207 | 0.271 | 0.222 | 0.281 | 0.309 | 0.288 | 0.309 |
| sz_taxi/H/short | **0.136** ± 0.000 | 0.139 | 0.144 | 0.201 | 0.144 | 0.190 | 0.170 | 0.232 | 0.229 |
| temperature_rain/D/short | 0.550 ± 0.002 | 0.682 | 0.644 | 0.830 | **0.592** | 0.840 | 0.694 | 0.761 | 1.63 |
| us_births/D/short | 0.021 ± 0.001 | 0.028 | 0.025 | 0.041 | **0.016** | 0.029 | 0.074 | 0.075 | 0.144 |
| us_births/M/short | 0.017 ± 0.002 | 0.016 | 0.017 | 0.032 | 0.021 | 0.025 | **0.010** | 0.019 | 0.017 |
| us_births/W/short | **0.013** ± 0.000 | 0.017 | 0.015 | 0.022 | 0.019 | 0.021 | 0.018 | 0.018 | 0.022 |

*Table 18.* MASE scores of different zero-shot models on the Chronos-ZS benchmark datasets. The models achieving the **best** and second-best scores are highlighted. Results for datasets that are part of the training data for the respective models are shaded in grey, and these results are excluded from the calculation of the best score. We trained TiRex with 6 different seeds and report the observed standard deviation in the plot.

| | TiRex | Chronos Bolt B | Chronos Bolt S | TimesFM 2.0 | TimesFM 1.0 | TabPFN-TS | Moirai L 1.1 | Moirai B 1.1 | TTM-r2 | Chronos B | Chronos S |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ETTh | 0.793 $\pm$ 0.019 | **0.748** | 0.793 | 0.854 | 0.890 | 0.903 | 0.825 | 0.902 | 0.973 | 0.774 | 0.827 |
| ETTm | 0.641 $\pm$ 0.006 | 0.633 | 0.621 | **0.620** | 1.04 | 0.806 | 0.755 | 0.826 | 0.781 | 0.803 | 0.718 |
| australian electricity | 1.06 $\pm$ 0.096 | **0.740** | 0.813 | 1.32 | 1.63 | 0.997 | 0.930 | 1.25 | 1.22 | 1.11 | 1.28 |
| car parts | **0.838** $\pm$ 0.004 | 0.855 | 0.858 | 0.986 | 0.901 | **0.838** | 0.846 | 0.863 | 1.16 | 0.889 | 0.875 |
| cif 2016 | 0.923 $\pm$ 0.005 | 0.999 | 1.02 | 0.949 | 1.04 | **0.894** | 1.03 | 1.06 | 1.61 | 0.985 | 0.996 |
| covid deaths | 39.5 $\pm$ 0.803 | 38.9 | **36.5** | 42.2 | 55.6 | 37.8 | 32.1 | 34.3 | 49.5 | 43.6 | 42.6 |
| dominick | **0.797** $\pm$ 0.004 | 0.856 | 0.871 | 0.935 | 1.13 | 0.864 | 0.830 | 0.828 | 1.31 | 0.824 | 0.812 |
| ercot | 0.630 $\pm$ 0.040 | 0.693 | 0.756 | 0.688 | 0.589 | 0.603 | 0.568 | 0.569 | 1.03 | **0.463** | 0.571 |
| exchange rate | 2.20 $\pm$ 0.157 | 1.71 | 1.77 | 1.98 | 3.34 | **1.44** | 1.74 | 1.81 | 2.01 | 2.18 | 1.78 |
| fred md | 0.451 $\pm$ 0.023 | 0.646 | 0.612 | 0.498 | 0.650 | 0.530 | 0.564 | 0.572 | 0.698 | **0.445** | **0.445** |
| hospital | 0.767 $\pm$ 0.003 | 0.791 | 0.801 | 0.755 | 0.783 | **0.686** | 0.830 | 0.820 | 0.843 | 0.812 | 0.815 |
| m1 monthly | 1.05 $\pm$ 0.008 | 1.10 | 1.13 | 1.05 | 1.07 | **1.04** | 1.14 | 1.14 | 1.47 | 1.12 | 1.16 |
| m1 quarterly | 1.69 $\pm$ 0.011 | 1.77 | 1.84 | 1.71 | 1.67 | **1.66** | 1.79 | 1.64 | 2.12 | 1.76 | 1.81 |
| m1 yearly | 3.67 $\pm$ 0.095 | 4.40 | 5.10 | 3.60 | 4.00 | **3.59** | 3.52 | 3.61 | 5.78 | 4.48 | 4.90 |
| m3 monthly | 0.847 $\pm$ 0.009 | 0.851 | 0.869 | **0.832** | 0.935 | 0.845 | 0.902 | 0.897 | 1.23 | 0.863 | 0.893 |
| m3 quarterly | 1.17 $\pm$ 0.011 | 1.29 | 1.33 | 1.20 | 1.15 | **1.10** | 1.12 | 1.15 | 1.72 | 1.18 | 1.29 |
| m3 yearly | 2.76 $\pm$ 0.070 | 2.91 | 3.41 | 2.82 | **2.70** | 2.73 | 2.75 | 2.73 | 3.87 | 3.17 | 3.42 |
| m4 quarterly | 1.18 $\pm$ 0.013 | 1.22 | 1.25 | 0.965 | 1.40 | **1.17** | 1.17 | 1.17 | 1.81 | 1.24 | 1.25 |
| m4 yearly | 3.46 $\pm$ 0.076 | 3.51 | 3.69 | 2.55 | 3.37 | **3.18** | 3.04 | 3.07 | 4.89 | 3.69 | 3.80 |
| m5 | **0.904** $\pm$ 0.001 | 0.915 | 0.920 | 0.919 | 0.919 | 0.927 | 0.927 | 0.924 | 1.10 | 0.934 | 0.931 |
| nn5 | **0.563** $\pm$ 0.007 | 0.576 | 0.583 | 0.608 | 0.629 | 0.665 | 0.586 | 0.638 | 0.998 | 0.594 | 0.618 |
| nn5 weekly | 0.913 $\pm$ 0.010 | 0.916 | 0.927 | **0.865** | 0.949 | 0.882 | 0.979 | 0.939 | 0.994 | 0.941 | 0.946 |
| tourism monthly | 1.47 $\pm$ 0.014 | 1.53 | 1.61 | 1.62 | 1.92 | **1.46** | 1.65 | 1.75 | 3.21 | 1.84 | 1.93 |
| tourism quarterly | 1.64 $\pm$ 0.026 | 1.76 | 1.84 | 1.80 | 2.06 | **1.59** | 1.82 | 1.92 | 3.77 | 1.81 | 1.77 |
| tourism yearly | 3.50 $\pm$ 0.100 | 3.69 | 3.89 | 3.49 | 3.23 | **3.02** | 3.21 | 3.15 | 3.56 | 3.84 | 3.98 |
| traffic | 0.799 $\pm$ 0.013 | **0.784** | 0.860 | 0.726 | 0.641 | 0.791 | 0.758 | 0.768 | 0.835 | 0.812 | 0.831 |
| weather | **0.790** $\pm$ 0.008 | 0.812 | 0.802 | 0.822 | 0.913 | 0.811 | 0.810 | 0.823 | 0.896 | 0.809 | 0.850 |

*Table 19.* WQL scores of different zero-shot models on the Chronos-ZS benchmark datasets. The models achieving the **best** and second-best scores are highlighted. Results for datasets that are part of the training data for the respective models are shaded in grey, and these results are excluded from the calculation of the best score. We trained TiRex with 6 different seeds and report the observed standard deviation in the plot.

| | TiRex | Chronos Bolt B | Chronos Bolt S | TimesFM 2.0 | TimesFM 1.0 | TabPFN-TS | Moirai L 1.1 | Moirai B 1.1 | TTM-r2 | Chronos B | Chronos S |
|---|---|---|---|---|---|---|---|---|---|---|---|
| ETTh | 0.079 ± 0.003 | **0.071** | 0.076 | 0.085 | 0.092 | 0.098 | 0.082 | 0.091 | 0.118 | 0.080 | 0.083 |
| ETTm | 0.054 ± 0.001 | 0.052 | **0.051** | 0.052 | 0.084 | 0.077 | 0.070 | 0.076 | 0.076 | 0.070 | 0.063 |
| australian electricity | 0.059 ± 0.005 | **0.036** | 0.042 | 0.067 | 0.089 | 0.045 | 0.038 | 0.054 | 0.074 | 0.067 | 0.072 |
| car parts | 0.990 ± 0.010 | 0.995 | 1.01 | 1.65 | 1.04 | **0.949** | 0.990 | 1.02 | 1.38 | 1.07 | 1.03 |
| cif 2016 | 0.012 ± 0.002 | 0.016 | 0.016 | 0.053 | 0.020 | **0.009** | 0.015 | 0.016 | 0.038 | 0.012 | 0.012 |
| covid deaths | **0.037** ± 0.004 | 0.047 | 0.043 | 0.215 | 0.204 | 0.040 | 0.036 | 0.045 | 0.108 | 0.045 | 0.063 |
| dominick | **0.321** ± 0.001 | 0.345 | 0.348 | 0.371 | 0.412 | 0.335 | 0.344 | 0.343 | 0.552 | 0.331 | 0.336 |
| ercot | 0.019 ± 0.002 | 0.021 | 0.026 | 0.021 | 0.021 | 0.020 | 0.017 | 0.018 | 0.044 | **0.013** | 0.015 |
| exchange rate | 0.013 ± 0.002 | 0.012 | 0.011 | 0.015 | 0.013 | **0.010** | 0.011 | 0.011 | 0.377 | 0.012 | 0.012 |
| fred md | 0.023 ± 0.006 | 0.042 | 0.037 | 0.027 | 0.036 | 0.055 | 0.042 | 0.050 | 0.050 | 0.020 | **0.015** |
| hospital | 0.052 ± 0.000 | 0.057 | 0.058 | **0.050** | 0.052 | 0.063 | 0.059 | 0.058 | 0.079 | 0.056 | 0.057 |
| m1 monthly | 0.136 ± 0.004 | 0.139 | 0.134 | 0.130 | **0.123** | 0.150 | 0.177 | 0.169 | 0.215 | 0.131 | 0.138 |
| m1 quarterly | 0.099 ± 0.004 | 0.101 | 0.094 | 0.113 | **0.087** | 0.090 | 0.093 | 0.076 | 0.156 | 0.102 | 0.106 |
| m1 yearly | 0.135 ± 0.008 | 0.151 | 0.157 | 0.145 | 0.163 | **0.118** | 0.127 | 0.123 | 0.246 | 0.198 | 0.183 |
| m3 monthly | 0.091 ± 0.000 | 0.093 | 0.094 | **0.089** | 0.098 | 0.090 | 0.099 | 0.101 | 0.153 | 0.096 | 0.100 |
| m3 quarterly | 0.070 ± 0.000 | 0.076 | 0.077 | 0.075 | 0.072 | **0.067** | 0.070 | 0.070 | 0.115 | 0.074 | 0.080 |
| m3 yearly | 0.131 ± 0.004 | 0.129 | 0.155 | 0.144 | **0.123** | 0.130 | 0.130 | 0.131 | 0.191 | 0.149 | 0.157 |
| m4 quarterly | **0.074** ± 0.000 | 0.077 | 0.078 | 0.062 | 0.085 | 0.075 | 0.076 | 0.076 | 0.125 | 0.083 | 0.083 |
| m4 yearly | 0.119 ± 0.002 | 0.121 | 0.128 | 0.091 | 0.117 | **0.114** | 0.108 | 0.109 | 0.192 | 0.135 | 0.138 |
| m5 | **0.551** ± 0.001 | 0.562 | 0.567 | 0.557 | 0.561 | 0.565 | 0.594 | 0.583 | 0.668 | 0.585 | 0.586 |
| nn5 | **0.146** ± 0.002 | 0.150 | 0.151 | 0.155 | 0.160 | 0.173 | 0.152 | 0.165 | 0.311 | 0.162 | 0.169 |
| nn5 weekly | 0.083 ± 0.001 | 0.084 | 0.085 | **0.079** | 0.086 | 0.081 | 0.091 | 0.089 | 0.113 | 0.089 | 0.089 |
| tourism monthly | **0.077** ± 0.002 | 0.090 | 0.094 | 0.085 | 0.101 | 0.084 | 0.098 | 0.099 | 0.283 | 0.101 | 0.108 |
| tourism quarterly | **0.061** ± 0.002 | 0.065 | 0.067 | 0.070 | 0.085 | 0.093 | 0.067 | 0.070 | 0.200 | 0.077 | 0.070 |
| tourism yearly | **0.148** ± 0.006 | 0.166 | 0.168 | 0.163 | **0.148** | **0.148** | 0.137 | 0.138 | 0.212 | 0.199 | 0.201 |
| traffic | 0.235 ± 0.004 | 0.231 | 0.252 | 0.212 | 0.185 | **0.229** | 0.236 | 0.238 | 5.54 | 0.255 | 0.258 |
| weather | **0.127** ± 0.000 | 0.134 | 0.133 | 0.133 | 0.150 | 0.132 | 0.133 | 0.135 | 0.159 | 0.137 | 0.147 |