Explore the Time Series Forecasting Potential of TabPFN Leveraging the Intrinsic Periodicity of Data

Sibo Cai [1 2] **Xi Sun** [3] **Hui Zhong** [3]

Abstract

Time series forecasting has extensive and significant applications in various fields such as transportation, energy, finance, etc. In recent years, time series foundation models have emerged prominently in the prediction field due to their ability to achieve accurate predictions with minimal fine-tuning or even in zero-shot scenarios. Among these models, TabPFN and its derivative TabPFN-TS stand out as notable examples. This paper introduces a TabPFN-based time series prediction method that capitalizes on the intrinsic periodicity of data. Experimental evaluations across multiple time series datasets reveal that our proposed method outperforms TabPFN-TS. This outcome validates the efficacy of incorporating data periodicity into TabPFN-based time series prediction. The code of our method can be found here: https://github.com/sibo-cai/TabPFN-TSP.

1. Introduction

Time series forecasting is a crucial research area in machine learning, attracting extensive attention from both academia and industry. Traditional methods such as ARIMA and ETS can perform satisfactorily when the data scale is small. However, they often struggle to handle complex time series patterns. Although deep learning methods can achieve good results with large-scale data, they typically rely on a vast amount of training data, complex model architectures, and have limited generalization capabilities.

The rise of foundation models has opened up new avenues for time series prediction. TabPFN (Hollmann et al., 2023; 2025), a foundation model designed for tabular data, transforms complex prediction tasks into tabular regression tasks. Combined with feature engineering, it demonstrates performance comparable to or even superior to professional time series models in zero-shot prediction scenarios. TabPFN-TS (TabPFN for Time Series) (Hoo et al., 2025), developed based on TabPFN, is specifically tailored for time series prediction. It converts time series data into tabular form and uses calendar features of timestamps, such as sine-cosine encodings of year, month, and day, along with running indices as input features, enabling TabPFN to process time series data.

However, TabPFN-TS is highly dependent on external timestamp calendar features during feature engineering, showing deficiencies in mining the internal periodicity of time series itself. When dealing with non-standard cycles (e.g. semester cycles in universities), since the model fails to fully consider the intrinsic periodic laws of the data during feature construction, it has difficulty capturing complex and variable periodic patterns, thus restricting the application scope of TabPFN-TS to a certain extent.

This paper presents a TabPFN-based time series prediction method based on the intrinsic periodicity of data. By analyzing the spectral characteristics of time series through FFT, extracting the periods corresponding to the frequencies with the highest amplitudes, and constructing input feature matrices based on these periods, the method enables TabPFN to more accurately capture the internal periodic patterns of data, thereby improving prediction performance. Experimental results on multiple datasets show that our proposed method outperforms TabPFN-TS, offering a promising alternative for time series prediction using TabPFN.

2. Related Work

2.1. Traditional Time Series Prediction Methods

Traditional methods like ARIMA (AutoRegressive Integrated Moving Average) and ETS (Error, Trend, Seasonal) are grounded in statistical theories, predicting by fitting the trend and seasonal components of data. These methods are simple in principle and efficient in computation. However, they have limited capabilities in modeling non-linear patterns and complex cycles. Their prediction accuracy sig-

[1]The Open University of China, Beijing, China [2]Engineering Research Center of Integration and Application of Digital Learning Technology, Ministry of Education, Beijing, China [3]MetaLight Inc., Beijing, China. Correspondence to: Sibo Cai <caisb@ouchn.edu.cn>.

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

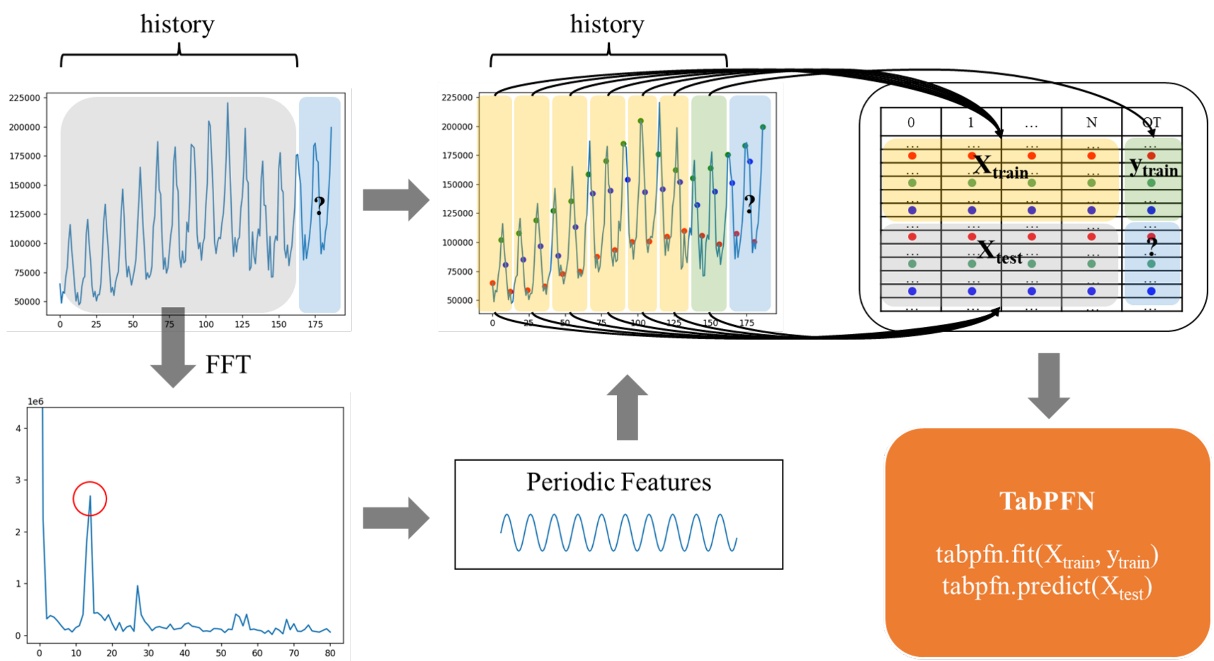


Figure 1. The overall working framework of our method.

nificantly declines, especially when the data contains high noise or shows variable patterns.

2.2. Deep Learning Models

Deep learning models such as DeepAR (Salinas et al., 2019), TimesNet (Wu et al., 2023), and several Transformer-based models including PatchTST (Nie et al., 2023), GPT4TS (Zhou et al., 2023), iTransformer (Liu et al., 2023), TimeXer (Wang et al., 2024), etc., automatically extract time series features through neural networks and can handle high-dimensional and non-linear data. Nevertheless, these methods require a large amount of labeled data for training, have numerous model parameters, incur high training costs, and are prone to overfitting in small dataset scenarios.

2.3. Foundation Models

In recent years, the application of foundation models in the time series field has gradually increased. Models like Chronos (Ansari et al., 2024), Lag-Llama (Rasul et al., 2024), TimesFM (Das et al., 2024), Moirai (Liu et al., 2024a), Moment (Goswami et al., 2024), Timer (Liu et al., 2024c) and Timer-XL (Liu et al., 2024b), etc., achieve zero-shot prediction through pre-training on large-scale time series datasets. TabPFN-TS, on the other hand, takes a different approach. By transforming time series problems into tabular regression tasks, leveraging timestamp features and the generalization ability of TabPFN, it achieves zero-shot prediction with much smaller parameter size and outperforms professional models on multiple datasets. However, its feature engineering is solely based on external timestamps, failing to utilize the data's intrinsic periodic characteristics.

3. Methodology

3.1. Overview

This paper proposes a TabPFN-based time series prediction method that capitalizes on the data's intrinsic periodicity. The method seeks to demonstrate the effectiveness of TabPFN in time series prediction tasks by integrating the intrinsic periodic patterns within time series data and refining the construction of input feature matrices.The method mainly consists of four core steps: First, divide the given time series to obtain subsequences for fitting and prediction; second, use the Fast Fourier Transform to extract key periodic information from the historical sequence; third, construct input feature matrices and label sequences based on the extracted periods; finally, input the constructed data into the TabPFN model for prediction, and evaluate the prediction results. The overall working framework is illustrated in Figure 1.

3.2. Problem Definition

Consider a time series $L = [x_1, x_2, \ldots, x_N]$. The objective is to forecast its terminal subsequence $y_{\text{test}} =$

Table 1. MAE comparison of our method and TabPFN-TS on the test datasets. The optimal performance is highlighted in bold, and the second-best performance is underlined. The lengths l and m are shown as integer multiples of the period length.

		monash_tourism_monthly (period=12)				m4_hourly (period=24)			
		item id 0		item id 1		item id 0		item id 1	
		m: 2x	m: 1x	m: 2x	m: 1x	m: 2x	m: 1x	m: 2x	m: 1x
TabPFN-TS		**144.48**	200.06	14435.49	9574.02	27.34	24.07	379.12	118.52
Ours	l: 1x	182.68	268.59	**7321.70**	**7946.79**	23.31	23.15	220.93	160.81
	l: 2x	183.21	**187.26**	9131.62	8408.83	33.93	**14.34**	**148.63**	194.97
	l: 3x	155.97	231.12	13747.95	10455.99	23.39	27.06	197.43	**70.83**
	l: 4x	214.78	199.16	19525.28	10603.79	**17.41**	21.97	208.44	71.88

$[x_{N-m+1}, \ldots, x_N]$, which has a length of m. To begin, partition L into a historical sequence $R = [x_1, \ldots, x_{N-m}]$ and the target prediction sequence y_{test} (denoted as question mark in Figure 1). Here, R is used to construct the input feature matrices X_{train}, X_{test} and y_{train} used in TabPFN.

3.3. Periodic Feature Extraction

Apply the Fast Fourier Transform to the historical sequence R, converting the time-domain signal into a frequency-domain representation to obtain the spectrum. In the spectrum, the amplitudes corresponding to different frequencies reflect the intensity of the frequency components in the original time series. Select the k frequencies $\{f_1, f_2, \ldots, f_k\}$ with the highest amplitudes, and calculate the corresponding periods $\{T_i = 1/f_i \mid i = 1, 2, \ldots, k\}$ according to the reciprocal relationship between period and frequency. These periods reflect the periodic patterns of the data.

3.4. Feature Matrix Construction

Selection of y_{train}: Extract a subsequence with a length of l from the end of R as $y_{\text{train}} = [x_{N-m-l+1}, \ldots, x_{N-m}]$. Note that both y_{train} and y_{test} are one-dimensional column vectors.

Construction of X_{train} and X_{test}: For each element y_j in y_{train} and y_{test}, extract historical values from R at intervals of T_i according to the extracted period T_i to form feature vectors. Specifically, for each element y_j, for each period $T_i (i = 1, 2, \ldots, k)$ in turn, trace back along the historical sequence R starting from the position corresponding to y_j until reaching the beginning of R. Extract all the values at the positions separated by period T_i during the tracing process. Each set of extracted values for a single period forms a feature sub-vector. Combine these feature sub-vectors to create a feature vector containing multiple elements for each period, resulting in a total feature vector with a length related to the number of traced elements and periods. Combine the feature vectors corresponding to all elements to form the two-dimensional matrices X_{train} and X_{test}. Here, X_{train} is used for model fitting and X_{test} is used to generate prediction results. It should be emphasized that, according to the construction rules, each column in X_{train} and X_{test} will be composed of sequential elements from R. Columns containing y_{test} values within X_{test} (if any) must be excluded. This step is to prevent the model from having prior knowledge of y_{test}.

3.5. Prediction with TabPFN

Feed the constructed X_{train} and y_{train} into TabPFN for fitting. Subsequently, input X_{test} into TabPFN to generate the predicted values $\hat{y}_{\text{test}}$. Finally, compare the prediction results $\hat{y}_{\text{test}}$ with the true values y_{test} by calculating MAE (Mean Absolute Error).

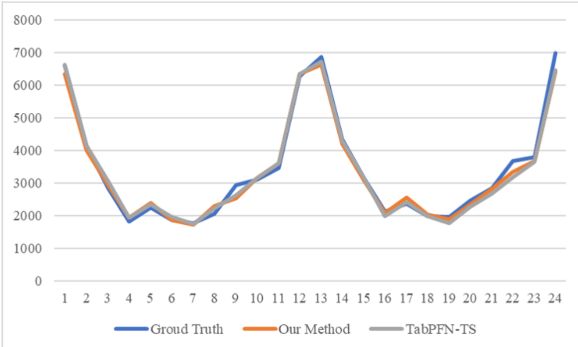
4. Experiments

4.1. Experimental Design

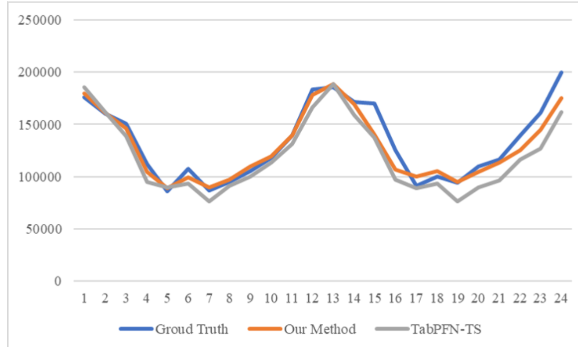
Test Datasets: We followed the same experimental datasets as those utilized by TabPFN-TS (Hoo et al., 2025) in their script in Google Colab . Consistent with the procedures described in the script, we selected the time series datasets $monash\_tourism\_monthly$ and $m4\_hourly$. Similarly, we picked the first two time series from each of these datasets, denoted as item id 0 and 1.

Comparison Method: This paper mainly compares with TabPFN-TS, which constructs the feature space by performing sine-cosine encoding on the calendar features of timestamps and combining it with running indices, serving as the basis for time series prediction.

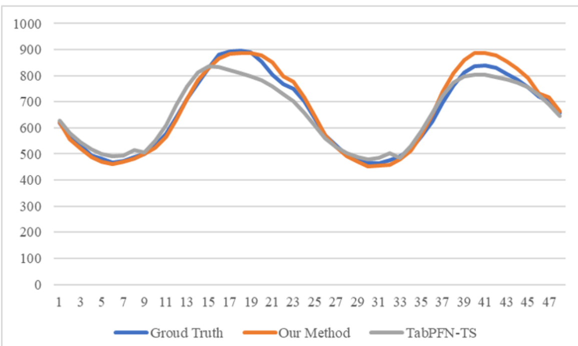
Parameter Configuration: In this preliminary study, parameter k was set to 1 empirically. But note that k can be fine-tuned according to the characteristics of different datasets. The lengths l and m of y_{train} and y_{test}, respectively, were configured as integer multiples of the period length. Specifically, the prediction length m was set to 1-multiple and 2-multiples of the period length, while the length of y_{train} ranged from 1 to 4 times the period length. For TabPFN model, we utilized version 2.0, configured it for regression tasks, and set the output type to "mean".



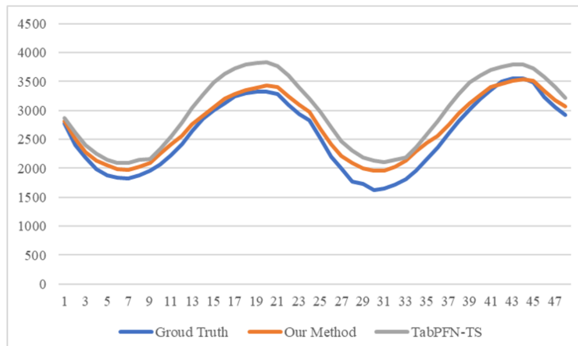
(a) monash_tourism_monthly (item id 0, 2x periods)



(b) monash_tourism_monthly (item id 1, 2x periods)



(c) m4_hourly (item id 0, 2x periods)



(d) m4_hourly (item id 1, 2x periods)

Figure 2. Prediction results visualization of our method and TabPFN-TS.

4.2. Experimental Results

In this section, the experimental results are presented in two parts to evaluate the performance of the proposed method.

4.2.1. MAE Comparison with TabPFN-TS

The proposed method was compared with TabPFN-TS using MAE as evaluation metrics across the test datasets. Table 1 summarizes the comparative results, where the optimal performance is highlighted in bold, and the second-best performance is underlined. The period values were computed via the Fast Fourier Transform (FFT). Specifically, for the $monash\_tourism\_monthly$ dataset, the identified period is 12, and for the $m4\_hourly$ dataset, the period is 24. From the results, except for a slight inferiority of our method compared to TabPFN-TS in the first case, our method outperforms TabPFN-TS in all other cases.

4.2.2. Prediction Results Visualization

Figure 2 illustrates the prediction results of our method and TabPFN-TS, providing a visual comparison between the predicted and actual time series values. A prediction length of 2-multiples period was chosen. The best results obtained by our method were selected for presentation. For the specific case of $monash\_tourism\_monthly$ with item id 0, although the MAE of our method is marginally higher than that of TabPFN-TS, the two prediction curves are almost overlapping. While in all other cases, our method shows better performance.

5. Conclusion

This paper presents a TabPFN-based time series prediction method that capitalizes on the intrinsic periodicity within data. By extracting the dominant periods of time series through FFT and constructing feature matrices, a new mode of time series prediction based on TabPFN is explored. The experimental results demonstrate that the proposed method outperforms TabPFN-TS.

Given that the work in this paper is still in its preliminary stages, future research endeavors will encompass comprehensive experiments across a more extensive and diverse datasets. We will also investigate in more detail the utilization of parameter k and incorporate a wider variety of covariates like timestamp features used in TabPFN-TS, along with weather and other relevant factors to enable practical application in real-world time series forecasting.

Acknowledgements

This work is supported by the scientific research project of Engineering Research Center of Integration and Application of Digital Learning Technology, Ministry of Education (Grant No.20220106).

References

Ansari, A. F., Stella, L., Turkmen, C., Zhang, X., Mercado, P., Shen, H., Shchur, O., Rangapuram, S. S., Pineda Arango, S., Kapoor, S., Zschiegner, J., Maddix, D. C., Mahoney, M. W., Torkkola, K., Gordon Wilson, A., Bohlke-Schneider, M., and Wang, Y. Chronos: Learning the language of time series. *Transactions on Machine Learning Research*, 2024. ISSN 2835-8856. URL https://openreview.net/forum?id=gerNCVqqtR.

Das, A., Kong, W., Sen, R., and Zhou, Y. A decoder-only foundation model for time-series forecasting, 2024. URL https://arxiv.org/abs/2310.10688.

Goswami, M., Szafer, K., Choudhry, A., Cai, Y., Li, S., and Dubrawski, A. Moment: A family of open time-series foundation models. In *International Conference on Machine Learning*, 2024.

Hollmann, N., Müller, S., Eggensperger, K., and Hutter, F. Tabpfn: A transformer that solves small tabular classification problems in a second. In *International Conference on Learning Representations*, 2023.

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*, 01 2025. doi: 10.1038/s41586-024-08328-6. URL https://www.nature.com/articles/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, 2025. URL https://arxiv.org/abs/2501.02945.

Liu, X., Liu, J., Woo, G., Aksu, T., Liang, Y., Zimmermann, R., Liu, C., Savarese, S., Xiong, C., and Sahoo, D. Moirai-moe: Empowering time series foundation models with sparse mixture of experts. *arXiv preprint arXiv:2410.10469*, 2024a.

Liu, Y., Hu, T., Zhang, H., Wu, H., Wang, S., Ma, L., and Long, M. itransformer: Inverted transformers are effective for time series forecasting. *arXiv preprint arXiv:2310.06625*, 2023.

Liu, Y., Qin, G., Huang, X., Wang, J., and Long, M. Timer-xl: Long-context transformers for unified time series forecasting. *arXiv preprint arXiv:2410.04803*, 2024b.

Liu, Y., Zhang, H., Li, C., Huang, X., Wang, J., and Long, M. Timer: Generative pre-trained transformers are large time series models. In *Forty-first International Conference on Machine Learning*, 2024c.

Nie, Y., H. Nguyen, N., Sinthong, P., and Kalagnanam, J. A time series is worth 64 words: Long-term forecasting with transformers. In *International Conference on Learning Representations*, 2023.

Rasul, K., Ashok, A., Williams, A. R., Ghonia, H., Bhagwatkar, R., Khorasani, A., Bayazi, M. J. D., Adamopoulos, G., Riachi, R., Hassen, N., Biloš, M., Garg, S., Schneider, A., Chapados, N., Drouin, A., Zantedeschi, V., Nevmyvaka, Y., and Rish, I. Lag-llama: Towards foundation models for probabilistic time series forecasting, 2024.

Salinas, D., Flunkert, V., and Gasthaus, J. Deepar: Probabilistic forecasting with autoregressive recurrent networks, 2019. URL https://arxiv.org/abs/1704.04110.

Wang, Y., Wu, H., Dong, J., Liu, Y., Qiu, Y., Zhang, H., Wang, J., and Long, M. Timexer: Empowering transformers for time series forecasting with exogenous variables. *Advances in Neural Information Processing Systems*, 2024.

Wu, H., Hu, T., Liu, Y., Zhou, H., Wang, J., and Long, M. Timesnet: Temporal 2d-variation modeling for general time series analysis. In *International Conference on Learning Representations*, 2023.

Zhou, T., Niu, P., Wang, X., Sun, L., and Jin, R. One fits all:power general time series analysis by pretrained lm, 2023. URL https://arxiv.org/abs/2302.11939.