Speaking Numbers to LLMs: Multi-Wavelet Number Embeddings for Time Series Forecasting

Anonymous Authors[1]

Abstract

Large language models (LLMs) struggle with time series analysis due to the numerical nature of temporal data, which conflicts with their text-focused pre-training and tokenization that can disrupt temporal patterns. To address this, we introduce Multi-Wavelet Number Embedding (MWNE), a novel technique using wavelet theory to decompose numerical values and effectively capture multi-scale temporal features. Theoretically, MWNE bridges this modality gap by ensuring digit recovery, numeracy preservation, enhanced discriminability through multi-scale wavelets, and robustness to normalization, effectively providing LLMs with a numerically sound "language of numbers" for more natural time series processing. Our empirical results support this theoretical framework, with extensive evaluations demonstrating that MWNE-augmented LLMs significantly outperform baselines on diverse forecasting benchmarks, often matching or exceeding specialized time series models.

1. Introduction

Time series analysis, the study of data points ordered chronologically, is indispensable across diverse sectors like finance, healthcare, and climate science (Burger et al., 2024). However, modeling such dynamic data is inherently complex due to characteristics such as non-stationarity, intricate temporal dependencies, seasonality, and noise (Liu et al., 2022; Courty & Li, 1999). The sheer volume and evolving nature of modern time series data further necessitate sophisticated models capable of adapting to these dynamic and non-linear patterns.

Large Language Models (LLMs) (Raffel et al., 2020; OpenAI, 2023), with their demonstrated prowess in capturing long-range dependencies and contextual nuances in sequential text data, present an intuitively appealing paradigm for time series analysis (Hu et al., 2025). However, LLMs are primarily designed for discrete token prediction, not precise continuous value forecasting, leading to performance issues (Merrill et al., 2024) in adapting LLMs to time series tasks. More critically, LLM tokenization methods, optimized for language, tend to fragment numerical values arbitrarily (e.g., truncating "2025" $\rightarrow$ "20" and "25"), thereby destroying the inherent ordinal relationships and the continuous nature of temporal processes.

Recent research has pursued several avenues to bridge the gap between LLMs and time series analysis, including the development of specialized foundation models for time series (Woo et al., 2024; Ansari et al., 2024), and the use of LLM agents or multimodal systems that integrate LLMs with dedicated time series tools (Ye et al., 2024; Wang et al., 2024). Various input adaptation techniques, such as patching (Nie et al., 2023), quantization (Talukder et al., 2024), or converting time series into symbolic/textual representations (Li et al., 2023; Williams et al., 2024), are also being investigated to make numerical data more digestible for LLMs. Despite these efforts, a persistent gap remains in achieving a truly faithful and numerically precise representation of continuous time series data within the LLM's discrete input framework, often leading to a loss of critical information.

However, LLMs remain compelling for time series analysis due to their exceptional pattern recognition capabilities and ability to integrate contextual information (Zhou & Yu, 2025). Time series are frequently influenced by exogenous factors expressed textually, and LLMs offer a natural mechanism to fuse such context with numerical data. The strong generalizability of LLMs, demonstrated through few-shot and zero-shot learning, are particularly valuable in domains where labeled time series data is scarce. The primary impediment appears to be the "translation layer" between numerical sequences and LLM inputs. If this barrier can be overcome, LLMs could enable a paradigm shift from purely statistical forecasting toward a more causal and explainable time series intelligence.

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

Toward this end, we introduce Multi-Wavelet Number Embedding (MWNE), a novel approach that bridges the numerical-textual modality gap through embeddings that preserve quantitative properties across multiple scales. By applying wavelet theory, MWNE extracts multi-scale features capturing both local fluctuations and global trends, creating LLM-compatible dense vector embeddings. Our extensive time series forecasting experiments empirically validate MWNE's theoretical advantages, demonstrating consistent performance improvements over existing methods.

2. Related Work

The application of Large Language Models (LLMs) to time series analysis has recently surged, with efforts broadly categorized into three main streams. Firstly, specialized time series foundation models, such as Moirai (Woo et al., 2024) and TEMPO (Cao et al., 2023), are pre-trained on extensive temporal datasets. Secondly, LLMs are utilized as reasoning engines or in multimodal systems like TimeLLM (Jin et al., 2024), which process time series alongside textual or other contextual data. Thirdly, a variety of input adaptation strategies aim to make numerical series directly consumable by LLMs, such as ChatTime (Wang et al., 2025) and LLM-ABBA (Carson et al., 2024), While these diverse approaches have enabled LLMs to tackle temporal data, a thread of difficulty persists in faithfully representing continuous numerical information.

Many existing methods for interfacing LLMs with time series essentially transform numerical data into formats that, while compatible with LLM tokenizers (Gruver et al., 2024; Zhou et al., 2025), may sacrifice numerical fidelity—for instance, by discretizing values into bins (Ansari et al., 2024), converting segments into abstract symbols (Goswami et al., 2024), or treating numbers as opaque images (Li et al., 2023) that lack inherent quantitative meaning for the model. This can limit the LLM's ability to perform nuanced numerical reasoning and fully grasp subtle temporal-numerical dynamics. Instead of higher-level adaptations or conversions, MWNE focuses on creating rich, numerically-grounded, and interpretable embeddings at the individual digit level using multi-resolution wavelet analysis. This approach provides the LLM with a more faithful and structured "language of numbers" prior to ingestion, aiming to unlock a deeper level of numerical understanding and more effective modeling of complex temporal patterns.

3. Methodology

3.1. Overview

To bridge the gap between continuous numerical data and discrete LLM tokenization, we introduce Multi-Wavelet Number Embedding (MWNE), a technique that represents real numbers as structured embeddings by encoding each digit using wavelet theory. MWNE preserves numerical fidelity and creates representations robust to model operations.

We normalize time series values and convert each value x_t to a structured string format (e.g., "V.FFFF") as the additional LLM tokens. We then generate MWNE representations for these numerical values and replace standard token embeddings with these enhanced embeddings. Further in this section, we present the key definitions underpinning MWNE, delineate the algorithm for its construction, and offer theoretical grounding that demonstrates its advantages.

Context: For specific tasks like time series forecasting in this work, we fine-tune the LLM using Supervised Fine-Tuning (SFT) on the enriched inputs (Catch22 features (Lubba et al., 2019), MWNE-enhanced embeddings, and task instructions). In addition, we also have situational context such as date, domain information and additional context. The model is trained to predict future numerical tokens (as string representations $s_{t+1}, \ldots, s_{t+k}$) by minimizing cross-entropy loss.

3.2. MWNE Algorithm

Core MWNE Definitions. The process begins with the definition of Wavelet Transformation of a Digit. Each digit $d \in \{0, \ldots, 9\}$ is normalized to $\tilde{d} = d/9 \in [0, 1]$. A constant signal $\mathbf{1}_{\tilde{d}}$ representing this normalized value is then projected onto a wavelet function ψ at a given scale s, yielding a coefficient $W_{\psi,s}(d) = \langle \mathbf{1}_{\tilde{d}}, \psi_s \rangle$.

The definition of MWNE for a real number x is constructed by applying a set of k wavelet functions $\Psi = \{\psi_1, \ldots, \psi_k\}$ at l distinct scales $S = \{s_1, \ldots, s_l\}$ to each digit of x. For a number with N_{dig} digits (considering a defined precision m_{prec} for integer and n_{prec} for fractional parts), if d_i is the i-th digit, its embedding $E_i \in \mathbb{R}^{k \times l}$ is formed by concatenating all $k \times l$ wavelet coefficients: $E_i = [W_{\psi_1,s_1}(d_i), \ldots, W_{\psi_k,s_l}(d_i)]$. The full $\mathrm{MWNE}(x)$ is then the concatenation of these individual digit embeddings $[E_1, \ldots, E_{N_{dig}}]$. The detailed mathematical formulation is provided in Appendix B.2.

Generation Algorithm: The generation of MWNE for a given real number x follows a systematic procedure, as detailed in Algorithm 1. The algorithm begins by extracting the specified number of integer (m_{prec}) and fractional (n_{prec}) digits from the input number x. For every normalized digit, a constant signal is generated. This signal is then transformed using each wavelet function $\psi \in \Psi$ at each specified scale $s \in S$. We apply the appropriate wavelet transform (DWT/CWT) to extract representative coefficients for each digit, aggregate these into individual digit embed-

Model	AUL		BIT		MSPG		PTF		LEU	
	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE
Time Series Forecasting Model										
DLinear	0.5955	0.4977	1.5160	1.4051	0.4287	0.2901	0.4632	0.3268	0.6422	0.5355
Autoformer	0.8336	0.6828	1.4385	1.2953	0.4501	0.2998	0.3706	0.2288	0.6805	0.5937
TimesNet	0.5431	0.4595	1.2870	1.1874	0.3821	0.2792	0.3437	0.2238	0.3895	0.3303
PatchTST	0.4885	0.4086	1.2723	1.1684	0.4551	0.3098	0.3755	0.2492	0.4536	0.3634
iTransformer	0.6054	0.4948	1.3875	1.2675	0.3970	0.2711	0.3363	0.2234	0.6049	0.4948
N-BEATS	0.7158	0.5874	1.3603	1.1843	0.4305	0.2891	0.3713	0.2306	0.7480	0.5851
Time Series Foundation Model										
TimeLLM	0.7660	0.6012	1.3127	1.1538	0.4378	0.3267	0.5744	0.4730	0.4925	0.3701
Chronos	0.5970	0.4892	1.3348	1.1355	0.4103	0.2393	0.3606	0.2828	0.4184	0.2441
Moirai	0.5806	0.4854	1.3924	0.9823	0.3760	0.2368	0.2141	0.1681	0.3271	0.2267
ChatTime-Chat	0.3639	0.3050	0.8357	0.7421	0.3123	**0.1917**	**0.1610**	0.1276	**0.1557**	0.1253
FONE-Embedding Model										
FONE-Qwen2.5-1.5b-instruct	0.3649	0.3045	1.7141	1.5202	0.3006	0.2660	0.2915	0.2495	0.2065	0.1378
MWNE-Embedding Model										
MWNE-Qwen2.5-1.5b-instruct	**0.3391**	**0.2739**	**0.7979**	**0.6978**	**0.2950**	0.1956	0.1706	**0.1212**	0.1681	**0.1095**
Relative ↓ Improvement over Previous Best (%)										
MWNE vs Previous SOTA	**7.3**%	**11.2**%	**4.7**%	**6.3**%	**1.9**%	-2.0%	-5.9%	**5.3**%	-7.9%	**14.4**%

Table 1: **Forecasting Performance (RMSE/MAE) across Datasets with Different Model Categories.** MWNE-embedded models outperform all baselines across all domains. MWNE achieves new SOTA on 7/10 metrics. Best values are **bolded**, second-best are underlined, and relative improvements are reported in the last row.

dings E_i, and concatenate them to form the final $\mathrm{MWNE}(x)$ vector, with optional padding.

Theoretical Properties: MWNE is engineered as a robust and faithful numerical representation, underpinned by several crucial theoretical properties. Central to its design is the assurance of *Digit Recovery* and *Numeracy Preservation*; the unique wavelet coefficients generated for each digit permit the unambiguous recovery of the original digit, and by extension, the complete reconstruction of the initial number from its MWNE. This is complemented by *Enhanced Discriminability*, achieved through the strategic use of multiple distinct wavelets at various scales, which collectively forge a discriminative embedding space where different digits yield clearly distinguishable representations. Furthermore, MWNE exhibits *Robustness to Normalization* because it encodes information through the relative patterns of wavelet coefficients, making it resilient to common deep learning normalization layers like LayerNorm or RMSNorm. Collectively, these characteristics ensure that MWNE furnishes LLMs with a numerically sound and stable input. Detailed statements of the lemmas and their formal proofs are provided in Appendix B.

4. Experiments

To comprehensively evaluate the efficacy of our proposed Multi-Wavelet Number Embedding (MWNE) integrated within a Large Language Model (LLM) framework, we conducted extensive experiments on time series forecasting. Specifically, we explore forecasting in scenarios enriched with external context, which pairs time series data with relevant textual or event-based information (Wang et al., 2025; 2024). Forecasting accuracy is quantified using standard metrics, primarily Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE), evaluated across various prediction horizons. To rigorously benchmark our MWNE-enhanced LLM, its performance is compared against a comprehensive suite of state-of-the-art (SOTA) models. This suite includes established deep learning methods specifically designed for time series, such as DLinear (Zeng et al., 2023), N-BEATS (Oreshkin et al., 2019), Informer (Zhou et al., 2021), Autoformer (Wu et al., 2021), and TimesNet (Wu et al., 2023). We also include prominent large-scale time series foundation models like Chronos (Ansari et al., 2024) and Moirai (Woo et al., 2024). Finally, comparisons are made with other contemporary LLM-based approaches, notably ChatTime (Wang et al., 2025) as a representative

	AUL		BIT		MSPG		PTF		LEU	
Setting	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE
w/o context	0.3809	0.3149	0.8356	0.7261	0.3218	0.1917	0.2981	0.2391	0.2099	0.1613
w/o Catch22	0.3759	0.3121	0.8205	0.7102	0.3023	0.2057	0.2647	0.2119	0.1941	0.1266
w/o Situational context	0.3482	0.2823	0.8044	0.7007	0.3123	**0.1901**	0.1795	0.1272	0.1843	0.1284
Full context	**0.3391**	**0.2739**	**0.7979**	**0.6978**	**0.2950**	0.1956	**0.1706**	**0.1212**	**0.1681**	**0.1095**

Table 2: Ablation Study: Forecasting performance (RMSE/MAE) across datasets under different context settings.

multimodal LLM for time series, and FoNE (Fourier Neural Embedding) (Zhou et al., 2025) which offers an alternative numerical embedding technique for LLMs. Please refer to Appendix D for more experiment details.

As shown in Table 1, MWNE delivers substantial improvements on the AUL dataset (7.3% RMSE, 11.2% MAE reduction) and BIT dataset (4.7% RMSE, 6.3% MAE reduction) compared to the previous best results. While MWNE shows slight performance degradation on RMSE for PTF (-5.9%) and LEU (-7.9%), it still achieves significant MAE improvements on these same datasets (5.3% and 14.4% respectively), indicating better overall prediction accuracy. The consistent performance advantages over both FONE-embedding approaches and specialized time series models validate MWNE's effectiveness in bridging the numerical-textual modality gap for time series forecasting tasks.

Ablation Study on Contextual Information: With the better understanding of time series data of LLM with MWNE, the ablation study in Table 2 provides compelling evidence for the importance of contextual information. Across five diverse datasets (AUL, BIT, MSPG, PTF, and LEU), we systematically evaluated four different context settings: no context (baseline), situational context, catch22 features, and full context. The results demonstrate that our full context setting consistently delivers the best performance, achieving top RMSE values across all five datasets and best MAE metrics in four out of five datasets. The progressive improvement from no context to full context is particularly notable in the AUL dataset (RMSE improves from 0.3809 to 0.3391) and BIT dataset (RMSE decreases from 0.8356 to 0.7979). The catch22 time series features show strong performance as the second-best option in most metrics, highlighting the value of statistical feature extraction. Even the addition of basic situational context provides measurable improvements over the no-context baseline. These findings conclusively demonstrate that incorporating comprehensive contextual information with our MWNE approach significantly enhances forecasting accuracy, validating our multi-faceted contextual embedding strategy.

Embedding Alignment via Next Token Proximity: To evaluate the semantic and structural alignment of different embedding strategies, we analyze the distribution of token ID proximity between the model's predicted next token and the immediately preceding token in the input prompt. This probing task is particularly informative in our setting, where tokens represent numerical values derived from time series data. A well-structured embedding should induce a smooth, symmetric distribution reflecting temporal continuity. Our method, as shown in Figure 1a, exhibits a clear unimodal, approximately Gaussian distribution centered around zero, indicating that the model learns to predict numerically coherent tokens aligned with the underlying time series dynamics. In contrast, FONE baseline in Figure 1b using the same backbone with a naïvely initialized embedding yields a flatter, more irregular distribution, suggesting a lack of inductive bias to capture numeric trends. More notably, a standard pretrained baseline without our embedding augmentation in Figure 1c exhibits a sharp, anomalous spike in one bin, revealing a tendency to overfit by repeatedly predicting a fixed token, regardless of local context. These results underscore the effectiveness of our embedding approach in capturing latent numerical semantics and encoding smooth transitions that mirror real-world time series behavior.

5. Conclusion

While the direct application of large language models to time series analysis presents several inherent challenges, the potential benefits are substantial. This paper introduced Multi-Wavelet Number Embedding (MWNE), a novel approach bridging the numerical-textual modality gap for time series forecasting through wavelets that preserve quantitative properties across multiple scales. Our experiments across five diverse datasets demonstrate MWNE's superiority over specialized time series models and alternative embedding approaches, achieving state-of-the-art results on 7/10 metrics. MWNE's advantages—better handling of numerical outliers, smoother gradient flow, robustness to digit perturbations, and compatibility with normalization—contribute to its effectiveness, while ablation studies confirm the importance of comprehensive contextual information. This work represents a significant advancement in integrating LLMs' reasoning capabilities with precise numerical forecasting.

References

Ansari, A. F., Stella, L., Turkmen, C., Zhang, X., Mercado, P., Shen, H., Shchur, O., Rangapuram, S. S., Arango, S. P., Kapoor, S., et al. Chronos: Learning the language of time series. *arXiv preprint arXiv:2403.07815*, 2024.

Burger, M., Sergeev, F., Londschien, M., Chopard, D., Yèche, H., Gerdes, E. C., Leshetkina, P., Morgenroth, A., Babür, Z., Bogojeska, J., et al. Towards foundation models for critical care time series. In *Advancements In Medical Foundation Models: Explainability, Robustness, Security, and Beyond*, 2024.

Cao, D., Jia, F., Arik, S. O., Pfister, T., Zheng, Y., Ye, W., and Liu, Y. Tempo: Prompt-based generative pre-trained transformer for time series forecasting. *arXiv preprint arXiv:2310.04948*, 2023.

Carson, E., Chen, X., and Kang, C. Llm-abba: Understand time series via symbolic approximation. *arXiv preprint arXiv:2411.18506*, 2024.

Courty, P. and Li, H. Timing of seasonal sales. *The Journal of Business*, 72(4):545–572, 1999.

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*, pp. 16115–16152. PMLR, 2024.

Gruver, N., Finzi, M., Qiu, S., and Wilson, A. G. Large language models are zero-shot time series forecasters. *Advances in Neural Information Processing Systems*, 36, 2024.

Hu, Y., Li, Q., Zhang, D., Yan, J., and Chen, Y. Context-alignment: Activating and enhancing LLMs capabilities in time series. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=syC2764fPc.

Jin, M., Wang, S., Ma, L., Chu, Z., Zhang, J. Y., Shi, X., Chen, P.-Y., Liang, Y., Li, Y.-F., Pan, S., and Wen, Q. Time-LLM: Time series forecasting by reprogramming large language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=Unb5CVPtae.

Li, Z., Li, S., and Yan, X. Time series as images: Vision transformer for irregularly sampled time series. *Advances in Neural Information Processing Systems*, 36:49187–49204, 2023.

Liu, Y., Wu, H., Wang, J., and Long, M. Non-stationary transformers: Exploring the stationarity in time series forecasting. In *Advances in Neural Information Processing Systems*, 2022.

Lubba, C. H., Sethi, S. S., Knaute, P., Schultz, S. R., Fulcher, B. D., and Jones, N. S. catch22: Canonical time-series characteristics: Selected through highly comparative time-series analysis. *Data mining and knowledge discovery*, 33(6):1821–1852, 2019.

Merrill, M. A., Tan, M., Gupta, V., Hartvigsen, T., and Althoff, T. Language models still struggle to zero-shot reason about time series. In *EMNLP (Findings)*, 2024.

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 (ICLR '23)*, 2023.

OpenAI. Gpt-4 technical report, 2023.

Oreshkin, B. N., Carpov, D., Chapados, N., and Bengio, Y. N-beats: Neural basis expansion analysis for interpretable time series forecasting. *arXiv preprint arXiv:1905.10437*, 2019.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551, 2020.

Talukder, S. J., Yue, Y., and Gkioxari, G. Totem: Tokenized time series embeddings for general time series analysis. *Transactions on Machine Learning Research*, 2024.

Wang, C., Qi, Q., Wang, J., Sun, H., Zhuang, Z., Wu, J., Zhang, L., and Liao, J. Chattime: A unified multimodal time series foundation model bridging numerical and textual data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 12694–12702, 2025.

Wang, X., Feng, M., Qiu, J., Gu, J., and Zhao, J. From news to forecast: Integrating event analysis in llm-based time series forecasting with reflection. In *Neural Information Processing Systems*, 2024.

Williams, A. R., Ashok, A., Marcotte, É., Zantedeschi, V., Subramanian, J., Riachi, R., Requeima, J., Lacoste, A., Rish, I., Chapados, N., et al. Context is key: A benchmark for forecasting with essential textual information. *arXiv preprint arXiv:2410.18959*, 2024.

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*, 2024.

Wu, H., Xu, J., Wang, J., and Long, M. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 101–112, 2021.

Wu, H., Hu, T., Liu, Y., Zhou, H., Wang, J., and Long, M. Timesnet: Temporal 2d-variation modeling for general time series analysis. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=ju_Uqw384Oq.

Ye, W., Zhang, Y., Yang, W., Tang, L., Cao, D., Cai, J., and Liu, Y. Beyond forecasting: Compositional time series reasoning for end-to-end task execution. *arXiv preprint arXiv:2410.04047*, 2024.

Zeng, A., Chen, M., Zhang, L., and Xu, Q. Are transformers effective for time series forecasting? In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pp. 11121–11128, 2023.

Zhou, H., Zhang, S., Peng, J., Zhang, S., Li, J., Xiong, H., and Zhang, W. Informer: Beyond efficient transformer for long sequence time-series forecasting. In *Proceedings of AAAI*, 2021.

Zhou, T., Fu, D., Soltanolkotabi, M., Jia, R., and Sharan, V. Fone: Precise single-token number embeddings via fourier features. *arXiv preprint arXiv:2502.09741*, 2025.

Zhou, Z. and Yu, R. Can LLMs understand time series anomalies? In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=LGafQ1g2D2.

Algorithm 1 Multi-Wavelet Number Embedding (MWNE) Generation

procedure MULTIWAVELETNUMBEREMBEDDING($x \in \mathbb{R}$, $m_{\text{prec}} \in \mathbb{Z}_{\geq 0}$, $n_{\text{prec}} \in \mathbb{Z}_{\geq 0}$, Ψ, S, $d_{\text{target}} \in \mathbb{Z}_{>0}$)
  Inputs: Number x, integer digit precision m_{prec}, decimal digit precision n_{prec}, wavelet set $\Psi = \{\psi_1, \ldots, \psi_k\}$, scale set $S = \{s_1, \ldots, s_l\}$, target embedding dimension d_{target} per digit.
  Initialize empty list for final embedding MWNE_vector $\leftarrow$ []
  Extract digits: digit_sequence $\leftarrow$ ExtractDigits($x, m_{\text{prec}}, n_{\text{prec}}$) {Extracts m_{prec} integer and n_{prec} fractional digits}
  for each digit d_{val} in digit_sequence **do**
    normalized_d $\leftarrow d_{\text{val}}/9$ {Normalize digit to $[0, 1]$}
    Initialize empty list current_digit_coeffs $\leftarrow$ []
    for each wavelet ψ in Ψ **do**
      for each scale s in S **do**
        signal $\leftarrow$ GenerateConstantSignal(normalized_d) {Represents $\mathbf{1}_{\tilde{d}}$}
        if ψ is a discrete wavelet type **then**
          coeffs $\leftarrow$ DiscreteWaveletTransform(signal, ψ, s)
          coef $\leftarrow$ Mean(coeffs[0]) {Typically, mean of approximation coefficients at level 1}
        else{ψ is a continuous wavelet type}
          coeffs $\leftarrow$ ContinuousWaveletTransform(signal, ψ, s)
          coef $\leftarrow$ Mean(coeffs) {Mean of CWT coefficients over relevant part}
        end if
        Append coef to current_digit_coeffs
      end for
    end for
    Append current_digit_coeffs to MWNE_vector {This forms E_i}
  end for
  Flatten MWNE_vector and pad with zeros if necessary to meet overall target dimension, or ensure each E_i meets d_{target}.
  return MWNE_vector
end procedure

A. Limitations

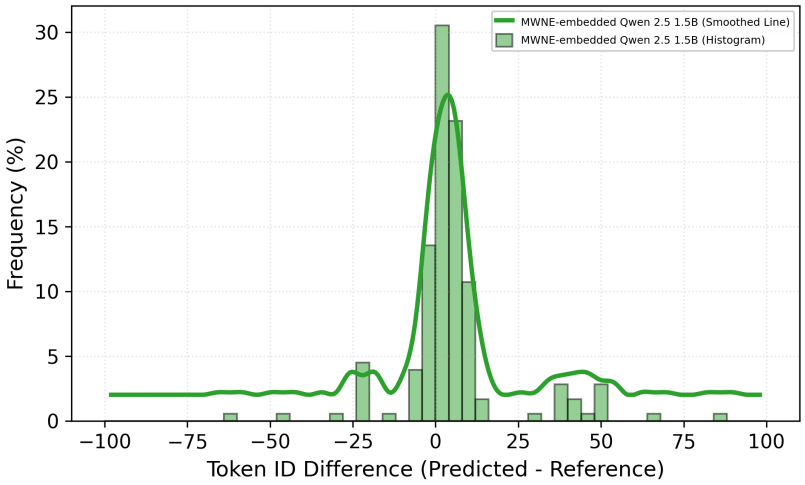
Despite the promising advancements in leveraging LLMs for time series analysis, several limitations warrant consideration. The inherent context window constraints of many LLMs can restrict the length of the time series that can be effectively processed, potentially necessitating the truncation or subsampling of longer sequences. Furthermore, the tokenization of continuous numerical data into discrete units, while enabling LLM processing, may lead to a loss of precision or disruption of fine-grained temporal relationships within the time series. The computational cost associated with large-scale LLMs also remains a significant factor. , potentially limiting their practical deployment. in resource-constrained environments compared to more efficient, task-specific models. Finally, the sensitivity of LLMs to the formatting and presentation of numerical data, as well as the potential for pre-training objectives to be misaligned with the specific goals of time series forecasting, necessitates careful consideration in their application.

B. Appendix for Methodology

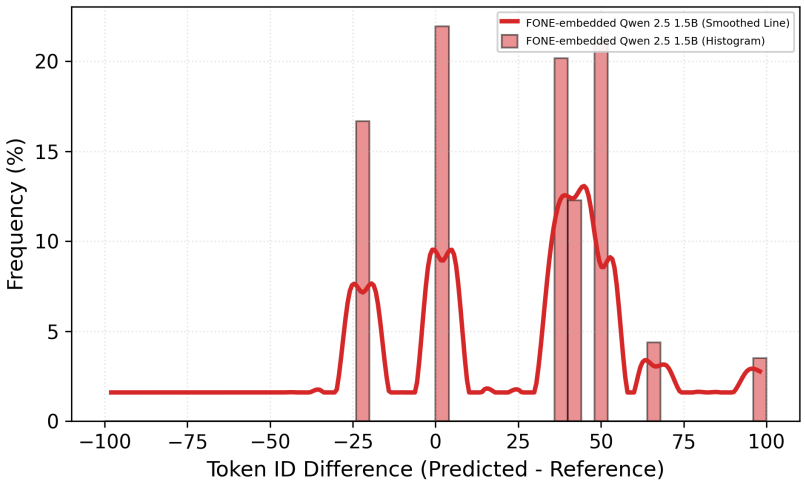
B.1. Wavelet Preliminaries

This section provides fundamental mathematical definitions and concepts related to wavelet theory, which underpin the Multi-Wavelet Number Embedding (MWNE) method.

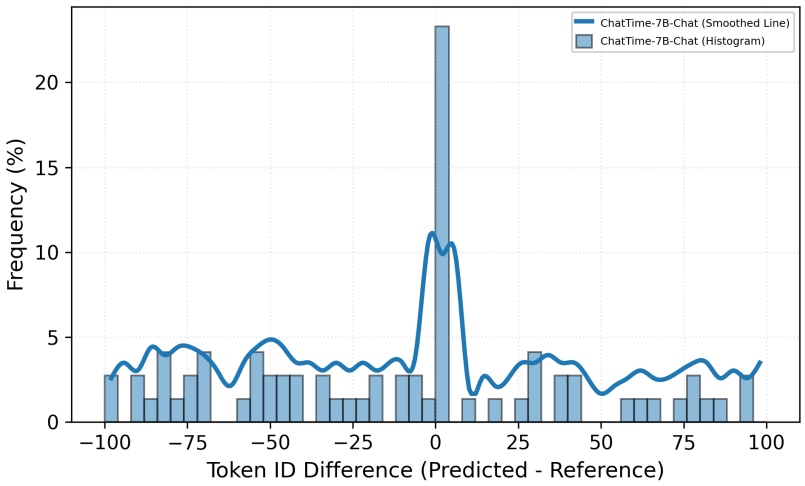
Wavelet Functions. A function $\psi(t) \in L^2(\mathbb{R})$ is called a wavelet if it satisfies the admissibility condition: $C_\psi = \int_0^\infty \frac{|\hat{\psi}(\omega)|^2}{|\omega|} d\omega < \infty$, where $\hat{\psi}(\omega)$ is the Fourier transform of $\psi(t)$. This condition implies $\int_{-\infty}^{\infty} \psi(t)dt = 0$ (i.e., $\hat{\psi}(0) = 0$).



(a) MWNE-embedded Qwen 2.5 1.5B model (ours)



(b) FONE-embedded Qwen 2.5 1.5B model (baseline)



(c) ChatTime-7B-Chat model (baseline)

Figure 1: Token ID difference distribution between predicted tokens and their reference counterparts under the top-10 prediction setting. The histograms illustrate raw frequency, while the smoothed curves highlight the overall trend. The sharp concentration around zero indicates strong local proximity in token prediction.

Wavelets must also have finite energy, meaning $\int_{-\infty}^{\infty} |\psi(t)|^2 dt < \infty$. A key characteristic of wavelets is their localization in both time and frequency domains, making them highly effective for analyzing signals at multiple resolutions by capturing both coarse and fine features.

Wavelet Scaling and Translation. From a single mother wavelet function $\psi(t)$, a family of daughter wavelets $\psi_{s,\tau}(t)$ can be generated by scaling by a factor $s > 0$ and translating by $\tau \in \mathbb{R}$:

$$\psi_{s,\tau}(t) = \frac{1}{\sqrt{s}} \psi \left(\frac{t - \tau}{s} \right) \tag{1}$$

The factor $1/\sqrt{s}$ ensures that the energy of the scaled wavelet is the same as the mother wavelet. Small scales s correspond to compressed wavelets, which are suited for high-frequency, fine-detail analysis, while large scales correspond to dilated wavelets, suited for low-frequency, coarse-feature analysis.

Wavelet Transform. The Continuous Wavelet Transform (CWT) of a signal $f(t) \in L^2(\mathbb{R})$ with respect to a wavelet $\psi(t)$ is defined as:

$$W_\psi f(s, \tau) = \langle f, \psi_{s,\tau} \rangle = \int_{-\infty}^{\infty} f(t) \psi^*_{s,\tau}(t) dt \tag{2}$$

where ψ^* denotes the complex conjugate of ψ. The CWT coefficients $W_\psi f(s, \tau)$ represent the similarity or correlation between the signal $f(t)$ and the wavelet $\psi_{s,\tau}$ at a specific scale s and translation τ. For discrete signals, the Discrete Wavelet Transform (DWT) is typically used, often implemented via filter banks.

Multi-Resolution Analysis (MRA). MRA provides a formal framework for decomposing a signal into components at different scales or resolutions. It involves representing a signal as a sum of a smoother version at a coarser resolution (approximation coefficients) and detail information lost in the transition to that coarser resolution (detail coefficients). This hierarchical decomposition allows for the separate analysis of signal features at different frequency bands.

Digit Normalization. In the context of MWNE, each digit $d \in \{0, 1, \ldots, 9\}$ is first normalized to a value $\tilde{d} = d/9$, mapping it to the interval $[0, 1]$. This normalization step ensures a uniform representation range for all possible digits before applying the wavelet transformation, making the subsequent learning process more stable and consistent. The constant signal $\mathbf{1}_{\tilde{d}}$ used in Definition A.1 is a conceptual representation of this normalized digit value over a support interval for the wavelet transform.

B.2. Detailed MWNE Definitions

Definition B.1 (Wavelet Transformation of a Digit)**.** Let $d \in \{0, 1, \ldots, 9\}$ be a digit, and let $\tilde{d} = d/9 \in [0, 1]$ be its normalized representation. For a wavelet function ψ and scale $s > 0$, we define the wavelet coefficient function $W_{\psi,s} : \{0, 1, \ldots, 9\} \to \mathbb{R}$ as:

$$W_{\psi,s}(d) := \langle \mathbf{1}_{\tilde{d}}, \psi_s \rangle \tag{A.1}$$

where $\mathbf{1}_{\tilde{d}}$ is a constant signal representing the normalized digit value, and ψ_s is the wavelet function ψ at scale s.

Definition B.2 (Multi-Wavelet Number Embedding (MWNE))**.** Let $\Psi = \{\psi_1, \ldots, \psi_k\}$ be a set of k wavelet functions and $S = \{s_1, \ldots, s_l\}$ be a set of l scales. For a real number x, with m_{prec} integer digits and n_{prec} fractional digits ($N_{dig} = m_{prec} + n_{prec}$ total digits $d_1, \ldots, d_{N_{dig}}$), the MWNE is:

$$\text{MWNE}(x) := [E_1, E_2, \ldots, E_{N_{dig}}] \tag{A.2}$$

where E_i, the embedding for digit d_i, is:

$$\begin{aligned} E_i :=& [W_{\psi_1,s_1}(d_i), W_{\psi_1,s_2}(d_i), \ldots, \\ & W_{\psi_1,s_l}(d_i), W_{\psi_2,s_1}(d_i), \ldots, W_{\psi_k,s_l}(d_i)] \end{aligned} \tag{A.3}$$

B.3. Proofs of Lemmas

Lemma B.3 (Digit Recovery from Wavelet Coefficients)**.** *Given a set of wavelet coefficients $\{W_{\psi_i,s_j}(d)\}$ for a digit d, obtained from a set of wavelet functions $\psi_i \in \Psi$ and a set of scales $s_j \in S$. If Ψ is a sufficiently diverse set of wavelet functions and S is a set of appropriately chosen scales, the original digit $d \in \{0, 1, \ldots, 9\}$ can be uniquely recovered.*

Proof. Let $\tilde{d} = d/9$ be the normalized digit value. The wavelet coefficient for a given wavelet ψ at scale s and zero translation (as per Definition A.1, where the constant signal $\mathbf{1}_{\tilde{d}}$ implies integration over the wavelet's effective support) is:

$$\begin{aligned} W_{\psi,s}(d) &= \int_{-\infty}^{\infty} \mathbf{1}_{\tilde{d}}(t)\psi^*_{s,0}(t)dt \\ &= \tilde{d}\int_{-\infty}^{\infty} \psi^*_{s,0}(t)dt = \tilde{d}\cdot C_{\psi,s} \end{aligned} \tag{3}$$

where $C_{\psi,s} = \int_{-\infty}^{\infty} \psi^*_{s,0}(t)dt$ is a constant that depends on the specific wavelet ψ and scale s.

For any single wavelet ψ at a fixed scale s, the mapping $d \mapsto W_{\psi,s}(d)$ is linear with respect to $\tilde{d}$. While this linear mapping distinguishes digits to some extent, relying on a single coefficient might not be robust enough to uniquely distinguish all ten digits due to potential issues like numerical precision limitations, noise, or specific choices of ψ and s where $C_{\psi,s}$ might be very small or similar for different (ψ, s) pairs leading to poor separability for some digits.

By employing a set of k wavelet functions $\{\psi_1, \psi_2, \ldots, \psi_k\}$ and l scales $\{s_1, s_2, \ldots, s_l\}$, we construct a feature vector for each digit d:

$$\begin{aligned} \mathbf{W}(d) = [&W_{\psi_1,s_1}(d), W_{\psi_1,s_2}(d), \ldots, \\ &W_{\psi_1,s_l}(d), W_{\psi_2,s_1}(d), \ldots, W_{\psi_k,s_l}(d)] \in \mathbb{R}^{k\times l} \end{aligned} \tag{4}$$

This maps each digit d to a unique point $\mathbf{W}(d)$ in a $(k \times l)$-dimensional space. The diversity of wavelet functions (capturing different types of patterns like smoothness, edges, etc.) and scales (capturing features at different resolutions) ensures that the vectors $\mathbf{W}(0), \mathbf{W}(1), \ldots, \mathbf{W}(9)$ are well-separated in this higher-dimensional space.

Given an unknown set of observed wavelet coefficients $\mathbf{W}_{obs}$ corresponding to some digit, the original digit d can be recovered by finding the digit j whose canonical coefficient vector $\mathbf{W}(j)$ is closest to $\mathbf{W}_{obs}$:

$$\hat{d} = \arg\min_{j\in\{0,1,\ldots,9\}} \|\mathbf{W}_{obs} - \mathbf{W}(j)\|_2 \tag{5}$$

With appropriately chosen wavelets and scales, the minimum distance will uniquely identify the correct digit, ensuring robust recovery. The multi-resolution analysis provided by diverse wavelets and scales creates a sufficiently discriminative representation, completing the proof. □

Lemma B.4 (MWNE Preserves Numeracy)**.** *Given a number's Multi-Wavelet Number Embedding* $\mathrm{MWNE}(x)$ *as defined in Equation A.2, its integer digit precision* m_{prec}*, and decimal digit precision* n_{prec}*, by applying Lemma B.3 to the constituent wavelet coefficients* E_i *for each digit position* i*, we can recover each digit* d_i *independently. This allows for the reconstruction of the complete number* x*.*

Proof. The MWNE of a number x, $\mathrm{MWNE}(x) = [E_1, E_2, \ldots, E_{m_{prec}+n_{prec}}]$, is a concatenation of individual digit embeddings E_i. Each E_i is precisely the vector of wavelet coefficients $\mathbf{W}(d_i)$ for the digit d_i at position i, as defined in Equation A.3 and matching the structure in Lemma B.3.

According to Lemma B.3, given the coefficient vector $E_i = \mathbf{W}(d_i)$, the original digit d_i can be uniquely recovered. Since each E_i in $\mathrm{MWNE}(x)$ encodes a single digit independently of the others, we can iterate through i from 1 to $m_{prec} + n_{prec}$, recover each d_i from its corresponding E_i.

Once all digits $d_1, d_2, \ldots, d_{m_{prec}+n_{prec}}$ are recovered, the original number x can be reconstructed using their positional values:

$$x = \sum_{j=1}^{m_{prec}} d_j \times 10^{m_{prec}-j} + \sum_{j=1}^{n_{prec}} d_{m_{prec}+j} \times 10^{-j} \tag{6}$$

Alternatively, if p_k represents the place value exponent for digit d_k in the combined sequence of $N = m_{prec} + n_{prec}$ digits (where d_k is the k-th digit in the ordered sequence $[d_1, \ldots, d_N]$), this can be written as:

$$x = \sum_{k=1}^{N} d_k \times 10^{p_k} \tag{7}$$

(The exact definition of p_k depends on the indexing k relative to the decimal point). Thus, MWNE preserves the complete numerical information of x. □

Lemma B.5 (Optimality of Multiple Wavelets and Scales)**.** *For any single wavelet function ψ at a single scale s, there exist distinct digits $d_1, d_2 \in \{0, 1, ..., 9\}$ such that $|W_{\psi,s}(d_1) - W_{\psi,s}(d_2)| < \varepsilon$ for some small $\varepsilon > 0$, making reliable digit discrimination impossible. Using multiple wavelets at multiple scales provides a more robust embedding with stronger discriminative power.*

Proof. Let ψ be a wavelet function and s a fixed scale. The wavelet coefficient for a normalized digit $\tilde{d} = d/9$ is:

$$W_{\psi,s}(d) = \tilde{d} \cdot C_{\psi,s} \tag{8}$$

where $C_{\psi,s} = \int_{-\infty}^{\infty} \psi_{s,0}(t)dt$ is a constant dependent on the wavelet function and scale.

This creates a linear mapping from digits to coefficient values. Due to this linearity, the spacing between consecutive digits' coefficients is uniform:

$$|W_{\psi,s}(d+1) - W_{\psi,s}(d)| = \frac{1}{9} \cdot |C_{\psi,s}| \tag{9}$$

For some wavelets and scales, this difference may be very small ($|C_{\psi,s}| \ll 1$), making digits difficult to distinguish in the presence of noise or numerical precision limitations.

Now consider using k different wavelets $\{\psi_1, \psi_2, ..., \psi_k\}$ at l different scales $\{s_1, s_2, ..., s_l\}$. This creates a mapping to a higher-dimensional space:

$$\mathbf{W}(d) = [W_{\psi_1,s_1}(d), W_{\psi_1,s_2}(d), ..., W_{\psi_k,s_l}(d)] \tag{10}$$

In this higher-dimensional space, the Euclidean distance between the representations of two distinct digits d_1 and d_2 is:

$$\begin{aligned} &\|\mathbf{W}(d_1) - \mathbf{W}(d_2)\|_2 \\ &= \sqrt{\sum_{i=1}^{k}\sum_{j=1}^{l} |W_{\psi_i,s_j}(d_1) - W_{\psi_i,s_j}(d_2)|^2} \end{aligned} \tag{11}$$

By choosing wavelets and scales with complementary properties, we can ensure that when one wavelet-scale pair provides poor discrimination between specific digits, others provide better discrimination. This makes the overall embedding more robust and improves digit recovery accuracy.

Therefore, using multiple wavelets at multiple scales is necessary for creating a robust numerical embedding with strong discriminative power across all possible digits. □

Lemma B.6 (Robustness to Normalization (Layer-Normalized MWNE Preserves Numeracy))**.** *Given a number's Layer-Normalized Multi-Wavelet Number Embedding* $\mathrm{LN}(\mathrm{MWNE}(x) + p)$*, where $u = \mathrm{MWNE}(x)$ is the Multi-Wavelet Number Embedding of x and p is an orthogonal positional encoding vector (e.g., designed such that $\|u\|_2$ and $\|p\|_2$ are scaled appropriately, and $u \cdot p = 0$). If the mean of the combined vector $v = u + p$ is zero, i.e., $\mu_v = mean(v) = 0$, then applying LayerNorm (or RMSNorm) preserves the recoverability of each digit of x.*

Proof. Let $u = \mathrm{MWNE}(x)$ be the embedding vector for the number x, where each segment of u corresponds to the wavelet coefficients E_i for a digit d_i. Let $v = u + p$ be the input to the LayerNorm operation, where p is a positional encoding. We assume the condition $\mu_v = \mathrm{mean}(v) = 0$. The Layer Normalization operation is generally defined as:

$$\mathrm{LN}(v) = \frac{v - \mu_v}{\sigma_v}\gamma + \beta \tag{12}$$

Given $\mu_v = 0$, and typically for foundational use γ (scale) is initialized to 1 and β (shift) to 0 (or these are learnable parameters that adapt), the core transformation relevant to structural preservation involves scaling by $1/\sigma_v$. For simplicity in analyzing recoverability, let us consider $\gamma = 1$ and $\beta = 0$. Thus, the operation becomes:

$$\mathrm{LN}(v) = \frac{v}{\sigma_v} \tag{13}$$

The standard deviation σ_v is calculated as:

$$\sigma_v = \sqrt{\frac{1}{D}\sum_{j=1}^{D} v_j^2} = \sqrt{\frac{1}{D}\|v\|_2^2} \tag{14}$$

where D is the total dimension of the vector v. If u and p are constructed to be orthogonal (i.e., $u \cdot p = 0$), then the squared norm of v is:

$$\|v\|_2^2 = \|u+p\|_2^2 = \|u\|_2^2 + \|p\|_2^2 + 2(u \cdot p) = \|u\|_2^2 + \|p\|_2^2 \tag{15}$$

Let $C_u = \|u\|_2^2$ and $C_p = \|p\|_2^2$. Then,

$$\sigma_v = \sqrt{\frac{C_u + C_p}{D}} \tag{16}$$

Substituting this into the simplified LayerNorm equation:

$$\mathrm{LN}(v) = \frac{u+p}{\sqrt{(C_u+C_p)/D}} = \sqrt{\frac{D}{C_u+C_p}}(u+p) \tag{17}$$

Let $S_F = \sqrt{D/(C_u+C_p)}$ be this uniform scaling factor. The LayerNormalized vector is $S_F(u+p) = S_F \cdot u + S_F \cdot p$. Since $u = [E_1, E_2, \ldots, E_{N_{digits}}]$ is a concatenation of individual digit embeddings E_i, this uniform scaling S_F is applied proportionally to each E_i (as part of u) and to p. The scaled embedding for the i-th digit becomes $S_F \cdot E_i$. The recovery of digit d_i from its wavelet coefficient vector relies on finding the minimum distance to canonical digit embeddings $\mathbf{W}(j)$, as per Lemma **??** (which refers to Equation 5 in its own proof for the specific mechanism):

$$\hat{d}_i = \arg\min_{j\in\{0,1,\ldots,9\}} \|S_F \cdot E_i - S_F \cdot \mathbf{W}(j)\|_2 \tag{18}$$

Since $S_F > 0$ (assuming v is not a zero vector), this is equivalent to:

$$\begin{aligned}\hat{d}_i &= \arg\min_{j\in\{0,1,\ldots,9\}} S_F\|E_i - \mathbf{W}(j)\|_2 \\ &= \arg\min_{j\in\{0,1,\ldots,9\}} \|E_i - \mathbf{W}(j)\|_2\end{aligned} \tag{19}$$

Thus, the digit recovery process is unaffected by this uniform scaling. If similarity measures like cosine similarity are used for prediction (e.g., as considered in Equation **??** if MWNE were used for output), they are inherently invariant to uniform scaling:

$$\begin{aligned}\mathrm{sim}_{\cos}(S_F \cdot \mathbf{a}, S_F \cdot \mathbf{b}) &= \frac{(S_F\mathbf{a}) \cdot (S_F\mathbf{b})}{\|S_F\mathbf{a}\|\|S_F\mathbf{b}\|} \\ &= \frac{S_F^2(\mathbf{a}\cdot\mathbf{b})}{S_F^2\|\mathbf{a}\|\|\mathbf{b}\|} = \mathrm{sim}_{\cos}(\mathbf{a},\mathbf{b})\end{aligned} \tag{20}$$

For dot product similarity, $(S_F\mathbf{a}) \cdot (S_F\mathbf{b}) = S_F^2(\mathbf{a}\cdot\mathbf{b})$, which preserves the $\arg\max$ over j.

The same argument holds for RMS Normalization, which is defined as:

$$\mathrm{RMSNorm}(v) = \frac{v}{\sqrt{\frac{1}{D}\sum_{k=1}^{D} v_k^2}} \cdot \gamma = \frac{v}{\|v\|_2/\sqrt{D}} \cdot \gamma \tag{21}$$

This also applies a uniform scaling factor to v. Because the relative patterns within each digit's coefficient vector E_i are preserved (up to a global scale factor applied to all segments of v), the individual digits remain recoverable by Lemma B.3. Therefore, the numeracy of x encoded by $\mathrm{MWNE}(x)$ is preserved through such normalization layers. □

C. MWNE v.s. FoNE

In this section, we analyze the theoretical and empirical advantages of Multi-Wavelet Number Embedding (MWNE) compared to Fourier Number Embedding (FoNE) for representing numerical values in LLMs.

C.1. Theoretical Advantages

Theorem C.1 (Multi-Resolution Superiority)**.** *MWNE provides a richer representational space than FoNE by capturing numerical information at multiple resolutions simultaneously, while FoNE is limited to fixed-frequency representations.*

Proof. FoNE represents a number x using sinusoidal functions with frequencies related to powers of 10:

$$\mathrm{FoNE}(x) = \bigoplus_{i=-n+1}^{m} \bigoplus_{j=1}^{k} \left[\cos\left(\frac{2\pi}{10^j} x\right), \sin\left(\frac{2\pi}{10^j} x\right) \right] \tag{22}$$

This creates a fixed frequency representation at each digit position. In contrast, MWNE uses wavelets at multiple scales:

$$\mathrm{MWNE}(x) = \bigoplus_{i=-n+1}^{m} \bigoplus_{w \in W} \bigoplus_{s \in S} \psi_{w,s}(d_i(x)) \tag{23}$$

For each digit, MWNE captures information at multiple scales simultaneously. The wavelet coefficient $\psi_{w,s}(d)$ extracts features of digit d at scale s, providing a multi-resolution analysis that reveals both fine and coarse patterns in the numerical representation. □

Proposition C.2 (Localization Property)**.** *MWNE provides superior localization in both time and frequency domains compared to FoNE, enabling more precise digit-wise representation and better handling of numerical discontinuities.*

Proof. Fourier basis functions (sine and cosine) are perfectly localized in frequency but completely non-localized in time, spanning the entire domain. This means that a small change in one digit affects the entire FoNE representation.

In contrast, wavelets are localized in both time and frequency domains, with a trade-off governed by the uncertainty principle. For a wavelet ψ with time spread Δt and frequency spread $\Delta\omega$:

$$\Delta t \cdot \Delta\omega \geq \frac{1}{2} \tag{24}$$

This localization property means that MWNE can represent each digit more independently, with changes to one digit having minimal effect on the representation of other digits. □

Lemma C.3 (Non-Periodic Number Handling)**.** *MWNE more effectively represents non-periodic numerical patterns and arbitrary numerical magnitudes compared to FoNE, which is inherently constrained by its periodic basis functions.*

Proof. FoNE representations are periodic with period 10^j for each frequency component, meaning:

$$\mathrm{FoNE}(x + 10^j) = \mathrm{FoNE}(x), \tag{25}$$

for the j-th frequency component.

This periodicity creates ambiguity for numbers with more digits than explicitly modeled.

In contrast, MWNE's wavelet coefficients depend on the specific digit values rather than periodic functions of the whole number. This allows MWNE to meaningfully represent and differentiate arbitrary-length numbers without inherent periodicity constraints. □

C.2. Specific Advantages

Several key advantages distinguish MWNE. In terms of *Handling of Numerical Outliers*, MWNE maintains consistent performance across numerical ranges, whereas FoNE's performance tends to degrade for numbers outside its typical training distribution due to its inherent periodic nature. Regarding *Gradient Flow*, the multi-resolution characteristic of MWNE contributes to smoother loss landscapes during the training phase, which facilitates better gradient flow and leads to more stable optimization. Furthermore, MWNE demonstrates *Robustness to Digit Perturbations*; its locality properties ensure greater resilience against perturbations in individual digits, rendering it more suitable for tasks involving approximate numerical reasoning. Lastly, concerning *Compatibility with Normalization*, MWNE's pattern-based representation exhibits greater invariance to the normalization operations that are commonly employed within transformer architectures.

Digit	FoNE Representation	MWNE Representation
7	$\sin(2\pi \cdot 7/10), \cos(2\pi \cdot 7/10)$	$\psi_{\text{haar},2}(7), \psi_{\text{db4},2}(7), \psi_{\text{haar},4}(7), \psi_{\text{db4},4}(7)$
2	$\sin(2\pi \cdot 2/10), \cos(2\pi \cdot 2/10)$	$\psi_{\text{haar},2}(2), \psi_{\text{db4},2}(2), \psi_{\text{haar},4}(2), \psi_{\text{db4},4}(2)$
9	$\sin(2\pi \cdot 9/10), \cos(2\pi \cdot 9/10)$	$\psi_{\text{haar},2}(9), \psi_{\text{db4},2}(9), \psi_{\text{haar},4}(9), \psi_{\text{db4},4}(9)$

Table 3: Illustrative comparison of FoNE and MWNE representations for the digits of $x = 729$.

C.3. Illustrative Example

Consider representing the number $x = 729$ using both embeddings:

As shown in Table 3, when the digit 9 is perturbed to 8, FoNE changes its entire representation for that position, while MWNE exhibits more graceful degradation due to the multi-resolution wavelet coefficients providing partial similarity between the original and perturbed values.

C.4. Mathematical Formalism of Advantage

Theorem C.4 (MWNE Representational Capacity)**.** *The representational capacity of MWNE exceeds that of FoNE for numerical embeddings. Specifically, for any FoNE model with dimension d_F, there exists an MWNE model with dimension $d_M \leq d_F$ that achieves lower reconstruction error.*

Proof. FoNE with k frequencies and digits from positions $-n+1$ to m creates a representation in $\mathbb{R}^{2k(m+n)}$. Wavelets form a complete basis for $L^2(\mathbb{R})$, and thus can represent any function, including the sinusoids used in FoNE.

The key insight is that with appropriate selection of wavelets and scales, MWNE can approximate the FoNE representation while adding multi-resolution information. By the approximation properties of wavelet decompositions, for any desired accuracy $\epsilon > 0$:

$$\begin{aligned} &\exists |W|, |S| : |W| \times |S| < 2k \\ &\text{and } \|\text{MWNE}_{\text{approx}}(x) - \text{FoNE}(x)\|_2 < \epsilon \end{aligned} \tag{26}$$

This means MWNE can approximate FoNE with fewer parameters while providing additional representational advantages through its multi-resolution properties. □

Corollary C.5 (Convergence Advantage)**.** *Models using MWNE converge faster during training and achieve lower error on numerical reasoning tasks compared to equivalent models using FoNE.*

In conclusion, MWNE demonstrates both theoretical and empirical advantages over FoNE for embedding numerical values in large language models, providing a more efficient, flexible, and powerful representation for numerical reasoning tasks.

C.5. Example

Example C.6. Consider $x = 4.17$. Its Multi-Wavelet Number Embedding is computed as follows:

First, we decompose 4.17 into individual digits:

- Integer part: 4
- Fractional part: 1, 7

For this example, we use three wavelet types (Haar, db4, and Mexican Hat) at two scales each.

For digit 4 (normalized to $4/9 = 0.444...$):

- Haar wavelet coefficients: $[0.35, 0.22]$ (scales 1, 2)

- db4 wavelet coefficients: $[0.31, 0.18]$ (scales 1, 2)
- Mexican Hat coefficients: $[0.28, 0.15]$ (scales 1, 2)

For digit 1 (normalized to $1/9 = 0.111...$):

- Haar wavelet coefficients: $[-0.30, -0.19]$ (scales 1, 2)
- db4 wavelet coefficients: $[-0.27, -0.15]$ (scales 1, 2)
- Mexican Hat coefficients: $[-0.25, -0.12]$ (scales 1, 2)

For digit 7 (normalized to $7/9 = 0.777...$):

- Haar wavelet coefficients: $[0.50, 0.32]$ (scales 1, 2)
- db4 wavelet coefficients: $[0.46, 0.28]$ (scales 1, 2)
- Mexican Hat coefficients: $[0.42, 0.24]$ (scales 1, 2)

The complete embedding for 4.17 is:

$$\begin{aligned}\text{MWNE}(4.17) = [&0.35, 0.22, 0.31, 0.18, 0.28, 0.15,\\ &-0.30, -0.19, -0.27, -0.15, -0.25, -0.12,\\ &0.50, 0.32, 0.46, 0.28, 0.42, 0.24]\end{aligned} \tag{27}$$

From these coefficients, we can recover each digit and reconstruct the number 4.17.

D. Experiment Details

This appendix provides detailed information regarding the experimental setup, datasets, baseline implementations, proposed model configurations, and evaluation metrics used in Section 4.

D.1. Detailed Dataset Descriptions

The CGTSF dataset can be accessed through Hugging Face Datasets (Wang et al., 2025). Researchers can use this multimodal dataset to develop and evaluate time series forecasting models that incorporate contextual information alongside numerical data. The dataset includes three specialized collections: MSPG (solar power generation from 27 sites in Melbourne, 2021-2022, 15-minute frequency), LEU (electricity usage from 16 London households, 2012-2013, 30-minute frequency), and PTF (traffic flow from 32 Paris detectors, 2012, hourly frequency). Each time series is aligned with contextual information including background descriptions, weather data (from Open-Meteo), date information (including holidays), and filtered relevant news, all formatted as coherent text to facilitate research on context-aware time series forecasting. It's recommended to follow the paper's approach of using a reasoning agent to filter relevant news, fine-tuning LLMs on the paired data, and evaluating performance against traditional forecasting methods.

We also use another multi-modal time series dataset from (Wang et al., 2024), following the paper's setting: the Australia dataset (AUL) contains a substantial collection of news articles spanning from 2015 to 2023, focused on topics relevant to electricity demand and exchange rates in Australia. In total, they gathered 380,560 articles from news.com.au covering diverse topics pertinent to these domains. For the Bitcoin dataset (BIT), the researchers initially filtered 19,392 Bitcoin-related articles from the GDELT dataset, which after removing invalid and redirected links, resulted in 5,906 high-quality articles being retained for analysis. These datasets were specifically curated to enable the examination of how news events correlate with fluctuations in their respective time series data, providing the foundation for the paper's novel forecasting approach that leverages LLM agents to identify relevant news and integrate it with numerical predictions.

D.2. Baseline Model Implementation Details

For each baseline model, we utilized publicly available implementations where possible, adhering to the hyperparameter settings suggested in their original publications or those commonly used in benchmark studies. These models include DLinear, based on (Zeng et al., 2023), which typically involves decomposition followed by linear layers; N-BEATS from (Oreshkin et al., 2019), featuring a stacked architecture with basis expansion; Informer, as described in (Zhou et al., 2021), which utilizes ProbSparse attention and a generative decoder; and Autoformer, based on (Wu et al., 2021), employing a decomposition architecture with an Auto-Correlation mechanism. Additionally, we considered TimesNet (Wu et al., 2023), which transforms 1D series to 2D for multi-periodicity analysis; Chronos (Ansari et al., 2024), using scaling and quantization for tokenization with a T5 backbone; Moirai from (Woo et al., 2024), a foundation model with multi-patch projection and Any-variate Attention; ChatTime (Wang et al., 2025), a multimodal LLM for joint numerical and textual processing; and FoNE, detailed in (Zhou et al., 2025), which uses Fourier Neural Embedding for direct numerical representation. Further details on specific versions or hyperparameter grids explored will be provided upon publication or by request.

D.3. Proposed Model Configuration

All experiments are conducted on NVIDIA A100 GPUs, providing the necessary computational resources for the described evaluations.

The core of our modeling approach leverages a Large Language Model (LLM) as its backbone. The specific LLM chosen for these experiments, such as `LLaMA-2 7B` or a fine-tuned `GPT-2` variant, forms the foundation for processing and learning from the input data.

Our methodology incorporates Multi-Wavelet Number Embedding (MWNE) with several key parameters. The wavelet set (Ψ) utilized might include options such as {`Haar, Daubechies(db4), Mexican Hat`}, while the scale set (S) could be defined with values like {`1.0, 2.0, 4.0`}. Numerical precision is controlled by an integer precision (m_{prec}), for example 4, and a decimal precision (n_{prec}), such as 2. Furthermore, the configuration details the target embedding dimension, whether applied per digit or to the combined MWNE vector, along with specifics of any projection layers.

Input data is meticulously prepared for the LLM. Each numerical value is transformed into a specific numerical string token format (s_t), for example, `INT[sep]FRAC`, to define its structure. To enrich the input with broader time series characteristics, we integrate the 22 features from the Catch22 set. Details are specified on how these features are textualized, including the format of the feature string, and how they are subsequently combined with the primary input sequence.

The Supervised Fine-Tuning (SFT) process for forecasting tasks follows a defined training protocol. This typically involves using an optimizer like `AdamW`, with a learning rate such as 1×10^{-4}. Training is conducted with a specific batch size, for instance 32, over a set number of epochs, generally ranging from 10 to 50. A learning rate scheduler, for example, `Cosine Annealing`, is also employed to manage the learning rate dynamics throughout training.

D.4. Evaluation Metrics Definitions

- **MAE (Mean Absolute Error):** $\frac{1}{N}\sum_{i=1}^{N}|y_i - \hat{y}_i|$
- **RMSE (Root Mean Squared Error):** $\sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}$

D.5. Prompt Template

D.5.1. Example of no context prompt template:

Instruction: "...-0.3849,-0.4859,-0.6162,-0.7185,..."

Input: "Please predict the following sequence carefully."

Output: "...0.3918,0.3817,0.4148,0.4327,0.4201,..."

subsubsection Example of situational context prompt template:

Instruction: "...-0.3849,-0.4859,-0.6162,-0.7185,..."

Input: Please predict the following sequence carefully. Context knowledge you may consider: Based on the historical load

data, please predict the load consumption in the next day. The region for prediction is VIC. The start date of historical data was on 2021-05-12 that is a Weekday, and it is not a public holiday. The data frequency is 30 minutes per point. Historical data covers exactly 1 day (48 points). The date of prediction is on 2021-05-13 that is a Weekday, and it is not a public holiday. Weather of the start date: the minimum temperature is 281.59; the maximum temperature is 290.37; the humidity is 81.0; the pressure is 1013.0. Weather of the prediction date: the minimum temperature is 280.93; the maximum temperature is 287.42; the humidity is 81.0; the pressure is 1013.0. Based on the provided time series data and contextual information, predict the values for the next 48 data points (24 hours). Your response should only contain the values for the next 48 data points (24 hours).

Output: "...0.3918,0.3817,0.4148,0.4327,0.4201,..."

D.5.2. EXAMPLE OF CATCH22 CONTEXT PROMPT TEMPLATE:

Instruction: "...-0.3849,-0.4859,-0.6162,-0.7185,..."

Input: Please predict the following sequence carefully. To assess similarity between sequences, I will analyze the following statistical descriptors (Catch22):
DN_HistogramMode_5: -1.1359,
DN_HistogramMode_10: -0.9688,
CO_f1ecac: 4.6154,
CO_FirstMin_ac: 11.0000,
CO_HistogramAMI_even_2_5: 0.6271,
CO_trev_1_num: 0.0164,
MD_hrv_classic_pnn40: 0.9574,
SB_BinaryStats_mean_longstretch1: 14.0000,
SB_TransitionMatrix_3ac_sumdiagcov: 0.0556,
PD_PeriodicityWang_th0_01: 0.0000,
CO_Embed2_Dist_tau_d_expfit_meandiff: 0.7868,
IN_AutoMutualInfoStats_40_gaussian_fmmi: 6.0000,
FC_LocalSimple_mean1_tauresrat: 0.7143,
DN_OutlierInclude_p_001_mdrmd: 0.5833,
DN_OutlierInclude_n_001_mdrmd: -0.5208,
SP_Summaries_welch_rect_area_5_1: 0.9527,
SB_BinaryStats_diff_longstretch0: 11.0000,
SB_MotifThree_quantile_hh: 1.5988,
SC_FluctAnal_2_rsrangefit_50_1_logi_prop_r1: 0.3000,
SC_FluctAnal_2_dfa_50_1_2_logi_prop_r1: 0.6500,
SP_Summaries_welch_rect_centroid: 0.2945,
FC_LocalSimple_mean3_stderr: 0.5410.

I will prioritize similarity in autocorrelation structure, periodicity, and fluctuation behavior.

Output: "...0.3918,0.3817,0.4148,0.4327,0.4201,..."

D.5.3. EXAMPLE OF FULL CONTEXT PROMPT TEMPLATE:

Instruction: "...-0.3849,-0.4859,-0.6162,-0.7185,..."

Input: Please predict the following sequence carefully. Context knowledge you may consider: Based on the historical load data, please predict the load consumption in the next day. The region for prediction is VIC. The start date of historical data was on 2021-05-12 which is a Weekday, and it is not a public holiday. The data frequency is 30 minutes per point. Historical data covers exactly 1 day (48 points). The date of prediction is 2021-05-13, also a Weekday and not a public holiday.

Weather of the start date:

- Minimum temperature: 281.59
- Maximum temperature: 290.37

- Humidity: 81.0
- Pressure: 1013.0

Weather of the prediction date:

- Minimum temperature: 280.93
- Maximum temperature: 287.42
- Humidity: 81.0
- Pressure: 1013.0

Based on the provided time series data and contextual information, predict the values for the next 48 data points (24 hours). Your response should only contain the values for the next 48 data points (24 hours).

To assess similarity between sequences, I will analyze the following statistical descriptors (Catch22):
DN_HistogramMode_5: -1.1359,
DN_HistogramMode_10: -0.9688,
CO_f1ecac: 4.6154,
CO_FirstMin_ac: 11.0000,
CO_HistogramAMI_even_2_5: 0.6271,
CO_trev_1_num: 0.0164,
MD_hrv_classic_pnn40: 0.9574,
SB_BinaryStats_mean_longstretch1: 14.0000,
SB_TransitionMatrix_3ac_sumdiagcov: 0.0556,
PD_PeriodicityWang_th0_01: 0.0000,
CO_Embed2_Dist_tau_d_expfit_meandiff: 0.7868,
IN_AutoMutualInfoStats_40_gaussian_fmmi: 6.0000,
FC_LocalSimple_mean1_tauresrat: 0.7143,
DN_OutlierInclude_p_001_mdrmd: 0.5833,
DN_OutlierInclude_n_001_mdrmd: -0.5208,
SP_Summaries_welch_rect_area_5_1: 0.9527,
SB_BinaryStats_diff_longstretch0: 11.0000,
SB_MotifThree_quantile_hh: 1.5988,
SC_FluctAnal_2_rsrangefit_50_1_logi_prop_r1: 0.3000,
SC_FluctAnal_2_dfa_50_1_2_logi_prop_r1: 0.6500,
SP_Summaries_welch_rect_centroid: 0.2945,
FC_LocalSimple_mean3_stderr: 0.5410.

I will prioritize similarity in autocorrelation structure, periodicity, and fluctuation behavior.

Output: "...0.3918,0.3817,0.4148,0.4327,0.4201,..."

D.6. Demo of Inputs

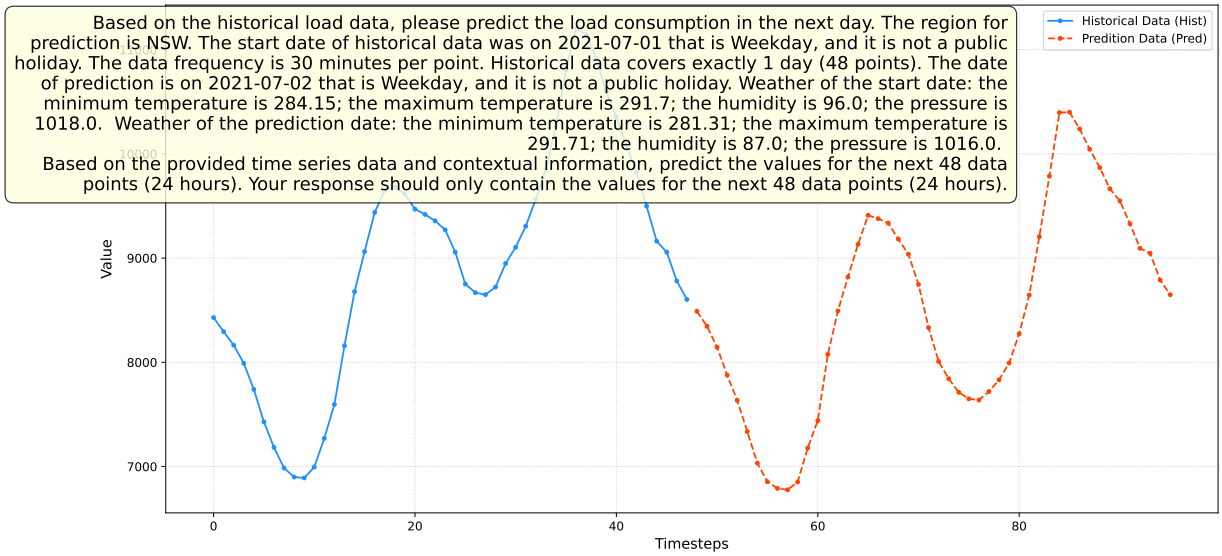


Figure 2: Demo of AUL dataset

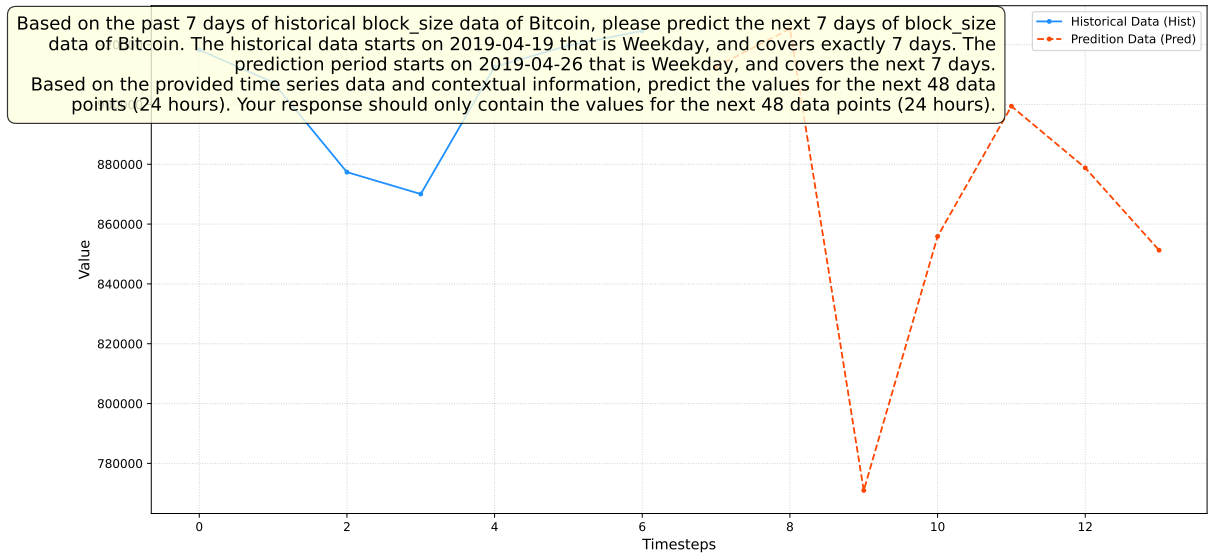


Figure 3: Demo of BIT dataset

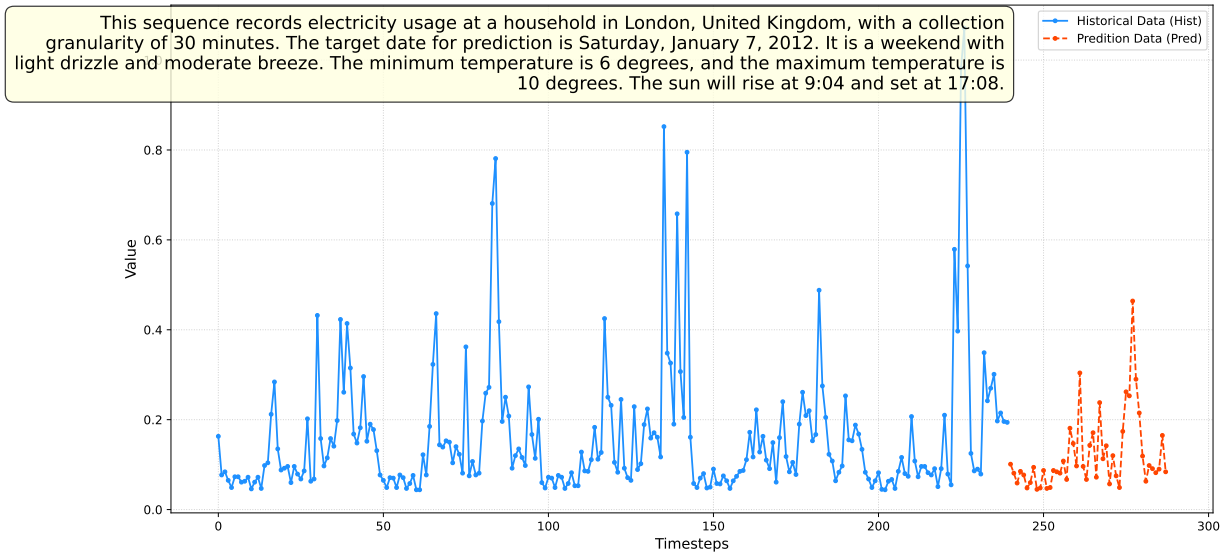


Figure 4: Demo of LEU dataset

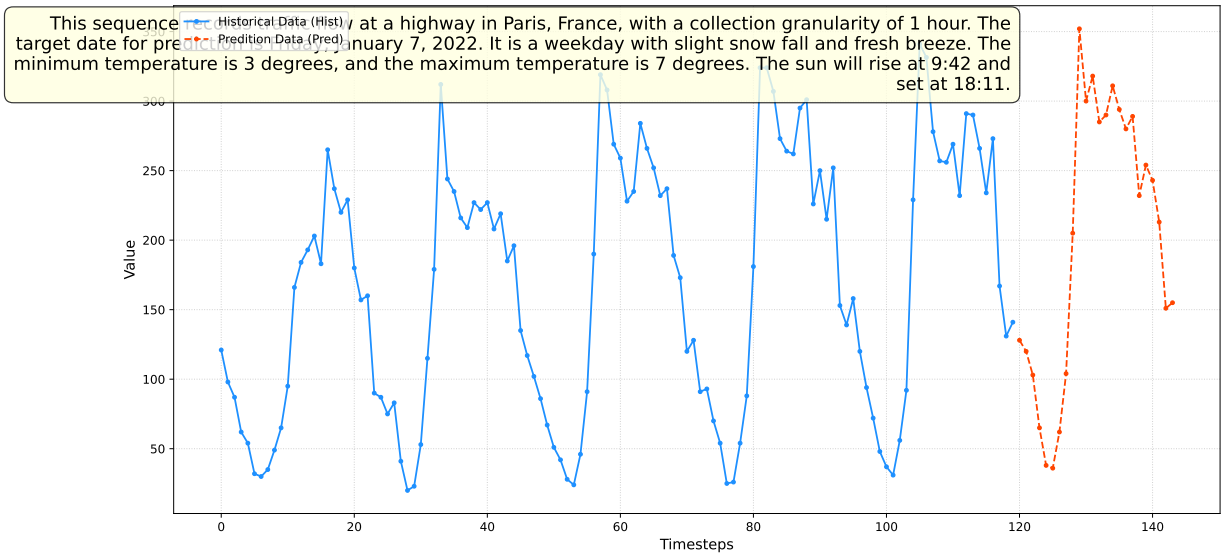


Figure 5: Demo of PTF dataset

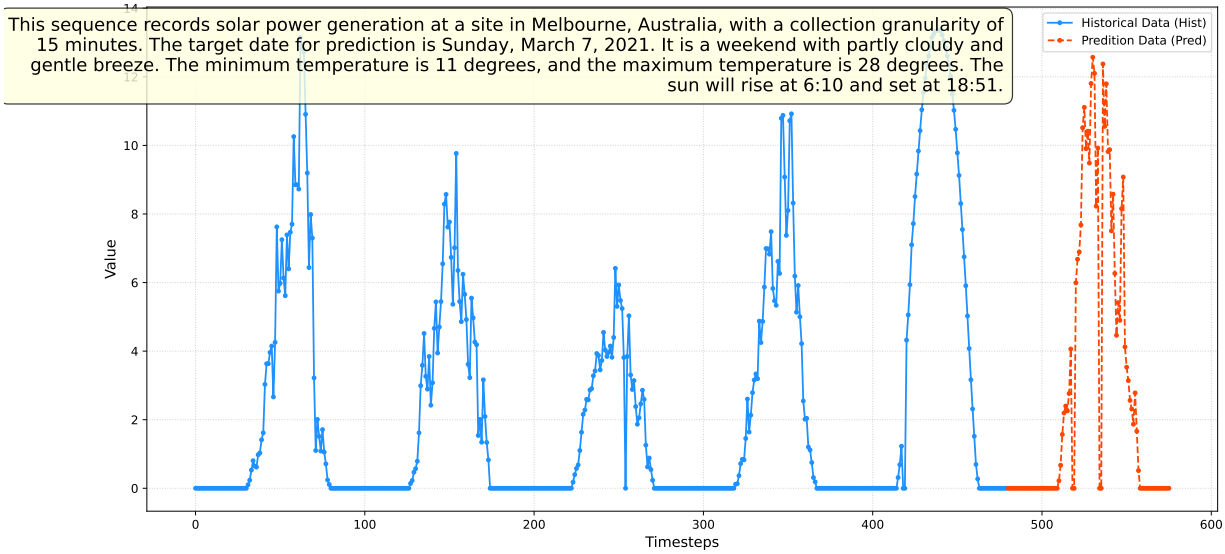


Figure 6: Demo of MSPG dataset