AutoTimes: Exploring LLM’s Potentials for TSF

AutoTimes: Autoregressive Time Series Forecasters via Large Language Models

Yong Liu [*1] **Guo Qin** [*1] **Xiangdong Huang** [1] **Jianmin Wang** [1] **Mingsheng Long** [1]

Yong Liu

Guo Qin

Xiangdon Huang

Jianmin Wang

Mingsheng Long

Text-Informed Time Series Forecasting

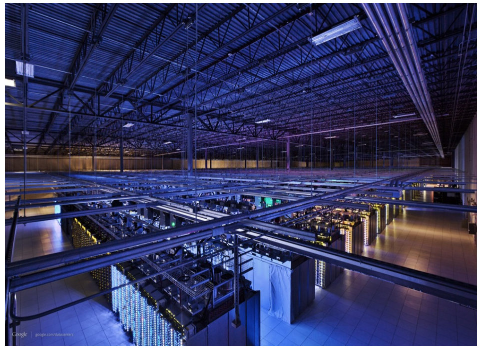
Industrial

Finance

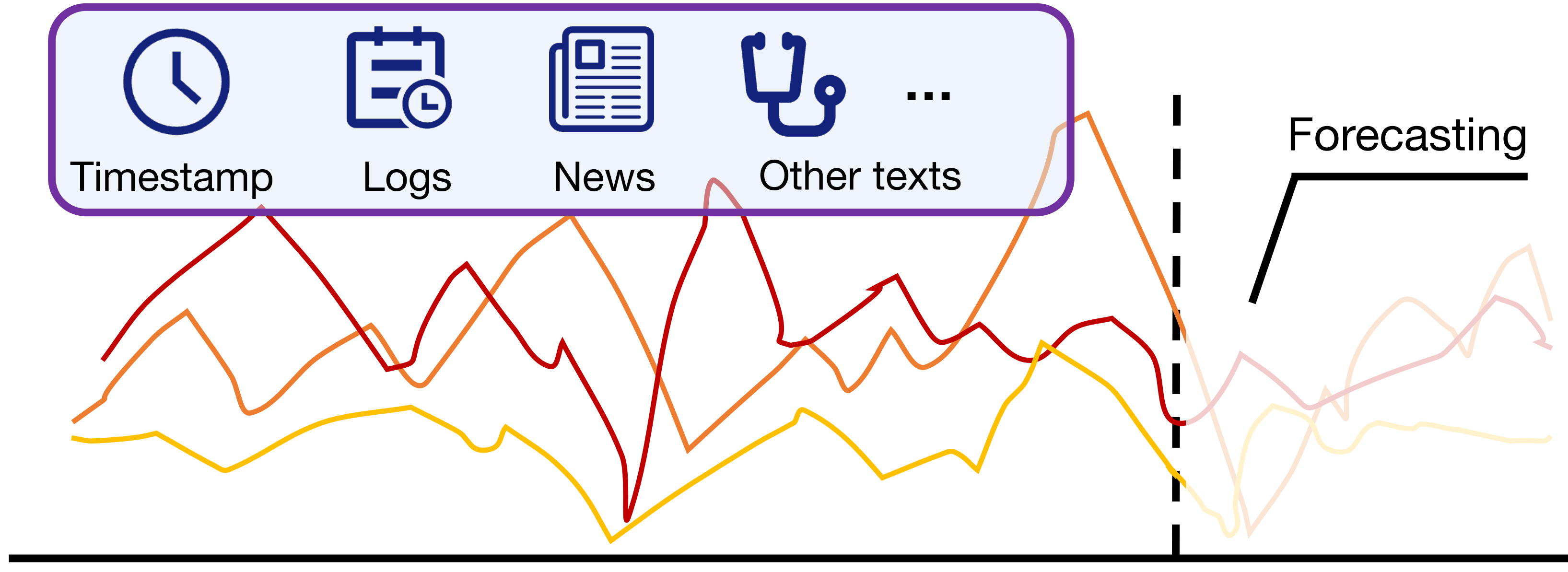
Climate

Health

IoT



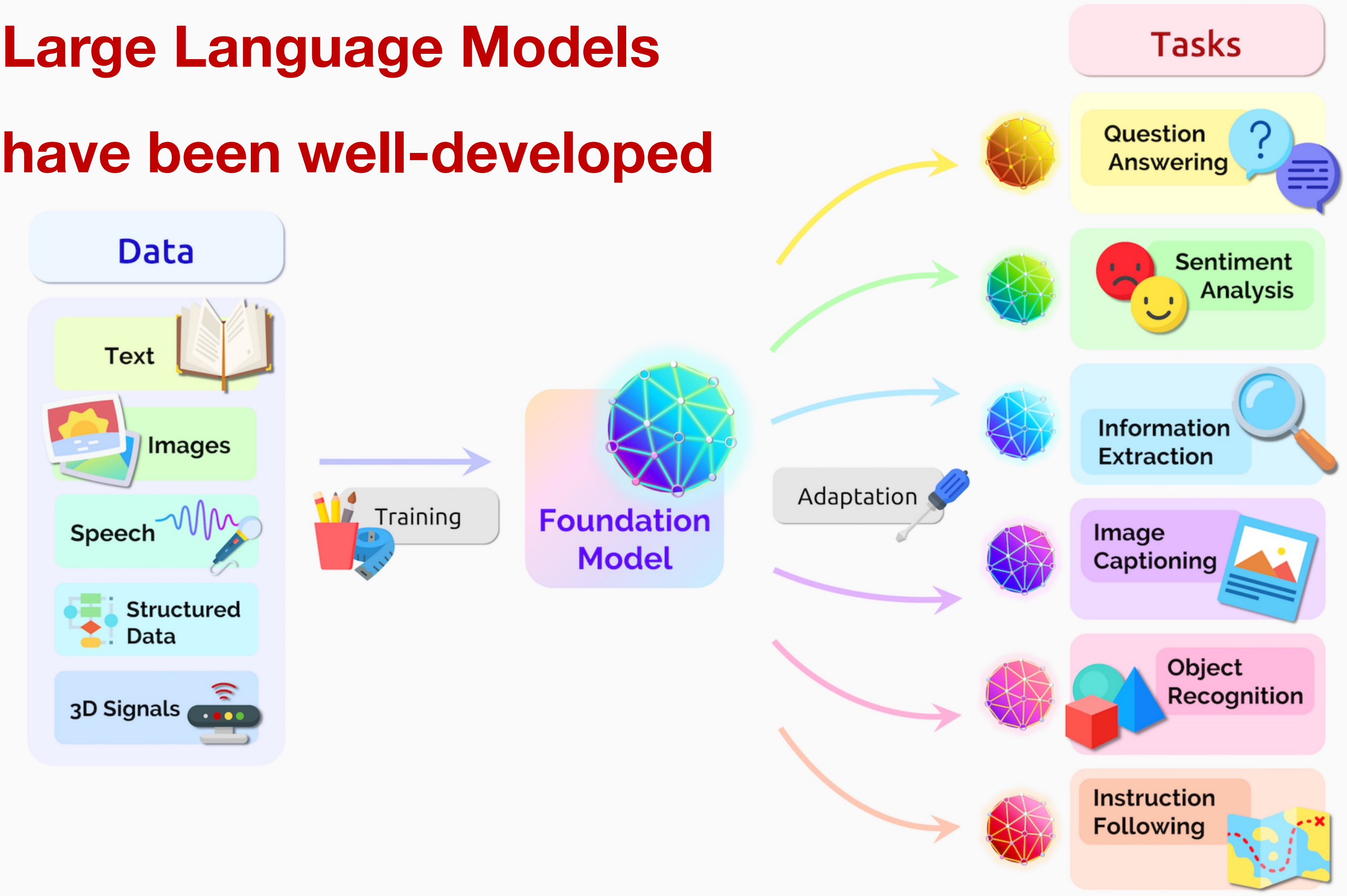
Time series and natural language always go together



- Process description
- Semantic token/variation
- Generative formulations
- …

Foundation Models

Large Language Models have been well-developed



Bommasani et al. On the Opportunities and Risks of Foundation Models. arXiv 2021.

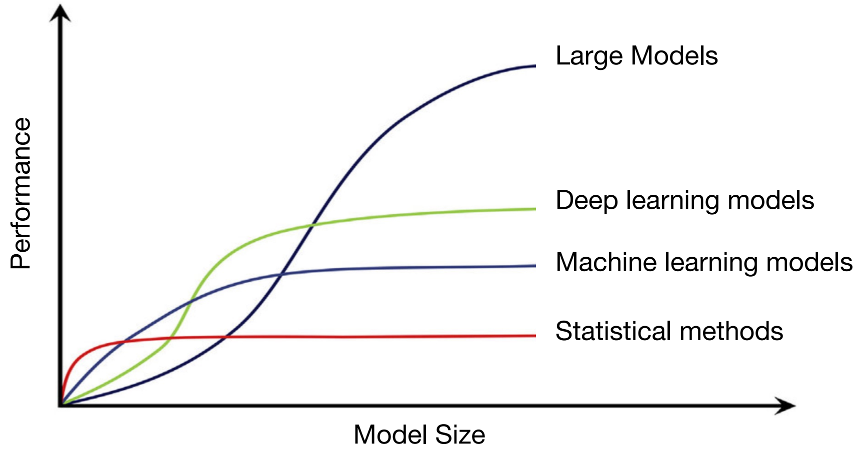
[Data General]

Learn from diverse modalities

[Task Universal]

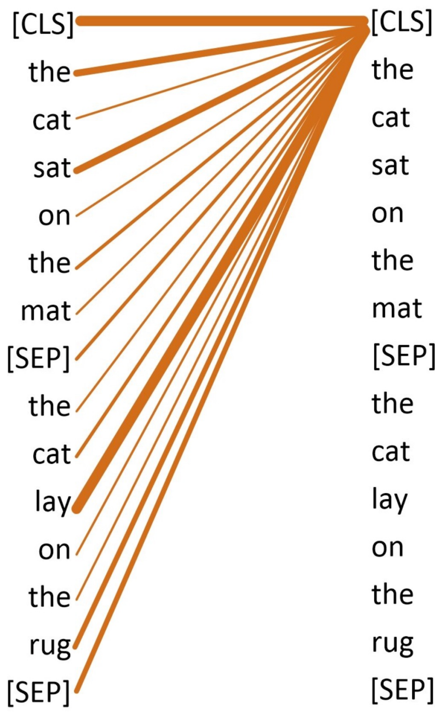
Adapt to diverse scenarios

[Scalable Backbone]



LLMs for Time Series: Motivations

Align time series and natural language



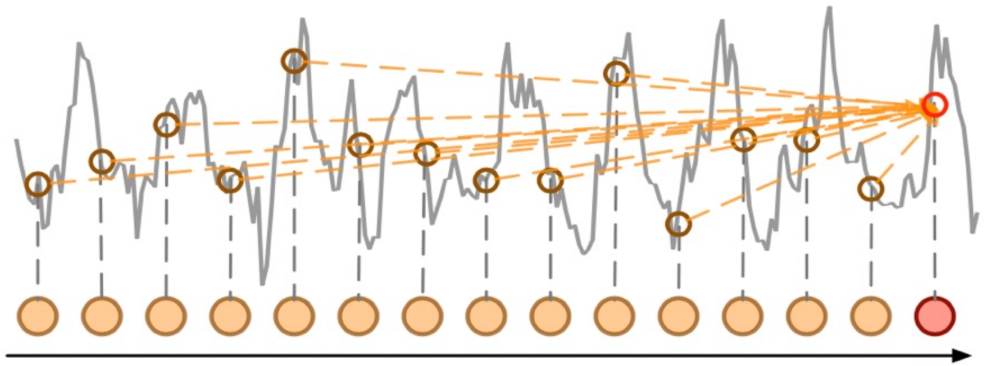
Dependencies of language tokens

Language modeling (Bengio et al., 2000):

$$P(\mathcal{U}) = \prod_{i=1}^{N} p(u_i | u_{<i})$$

Time series forecasting:

$$P(\mathbf{x}_{L+1:L+F} | \mathbf{x}_{1:L}), \mathbf{x} \in \mathbb{R}^C$$

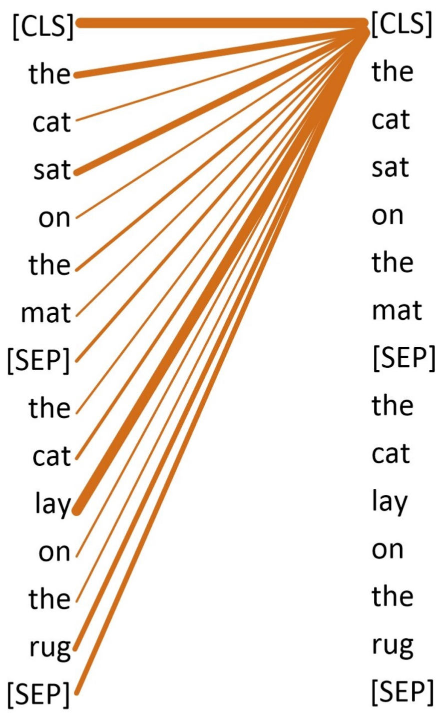


Dependencies of time points

Goal of LLM4TS: Leverage off-the-shelf LLMs as foundation models for time series

LLMs for Time Series: Motivations

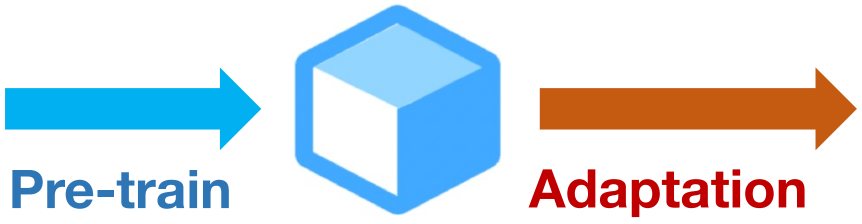
Align time series and natural language



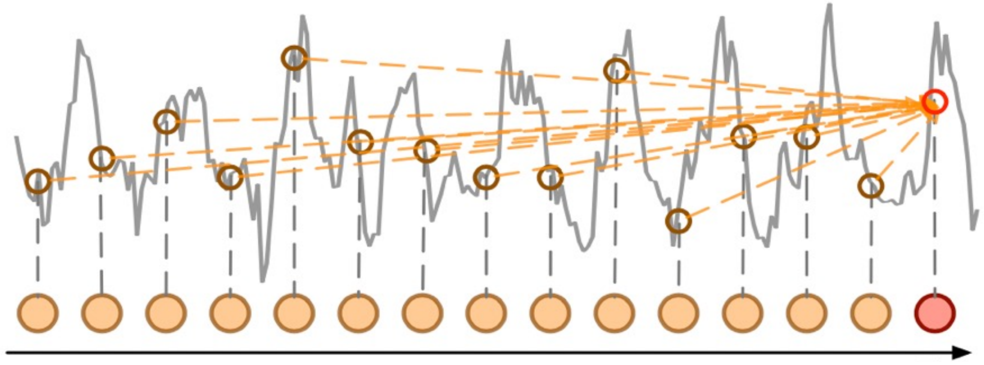
- **Large-scale text corpora**
- **Scalable and versatile architecture**

Language Models

- **Token Semantics**
- **Multimodality…**



Large Time Series Models



- **Limited scale of datasets**
- **Avoid case-by-case training**

Goal of LLM4TS: Leverage off-the-shelf LLMs as foundation models for time series

Insufficient Utilization of Language Models

Are Language Models Actually Useful for Time Series Forecasting?

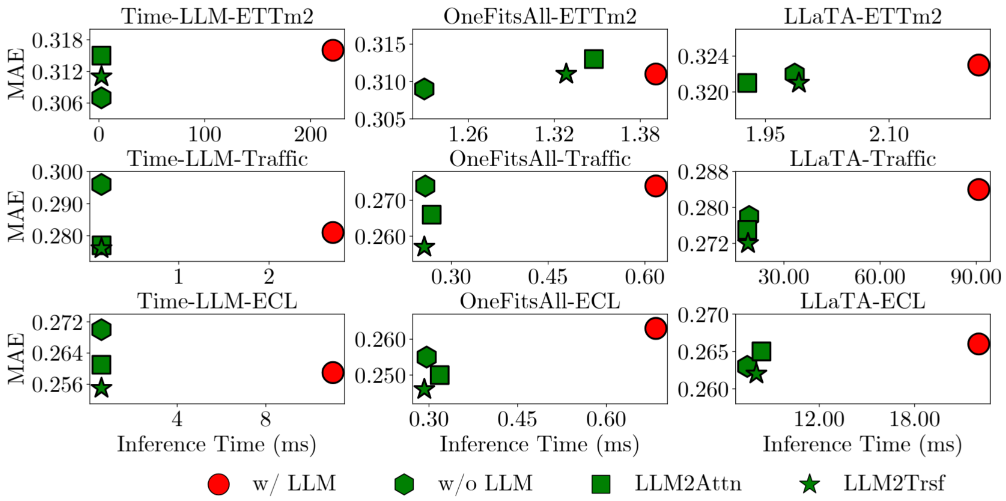
Mingtian Tan
University of Virginia
wtd3gz@virginia.edu

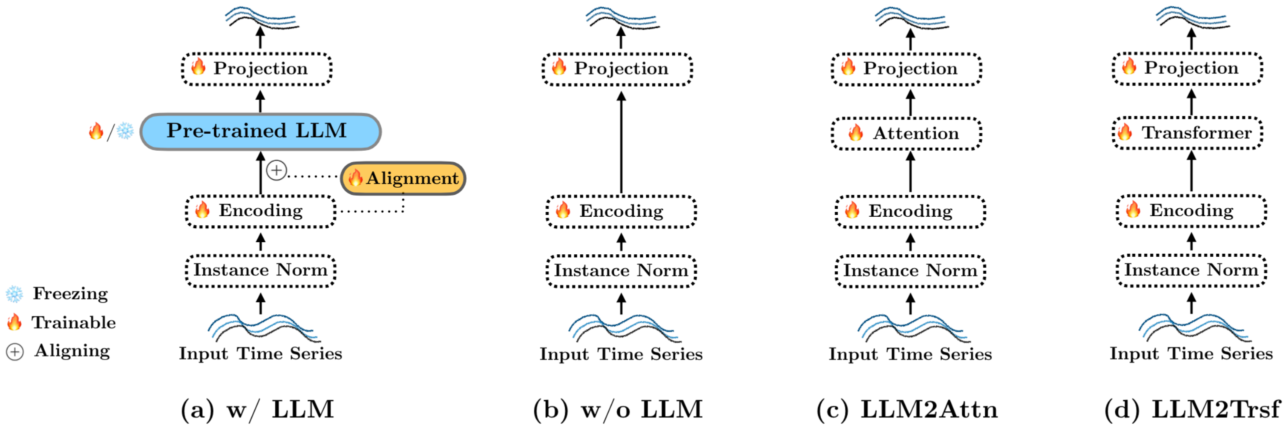
Mike A. Merrill
University of Washington
mikeam@cs.washington.edu

Vinayak Gupta
University of Washington
vinayak@cs.washington.edu

Tim Althoff
University of Washington
althoff@cs.washington

Thomas Hartvigsen
University of Virginia
hartvigsen@virginia.edu



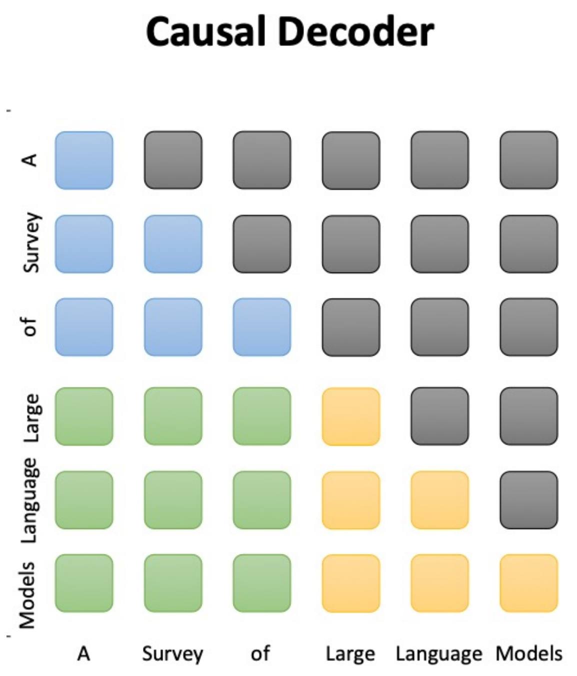


- ✗ High adaptation cost (**7B+ Params. In a LLM**)
- ✗ Results are still good **without LLMs**
- ✗ **Patch + Project** is already a simple & effective choice

Tan et al. Are Language Models Actually Useful for Time Series Forecasting? NeurIPS 2024.

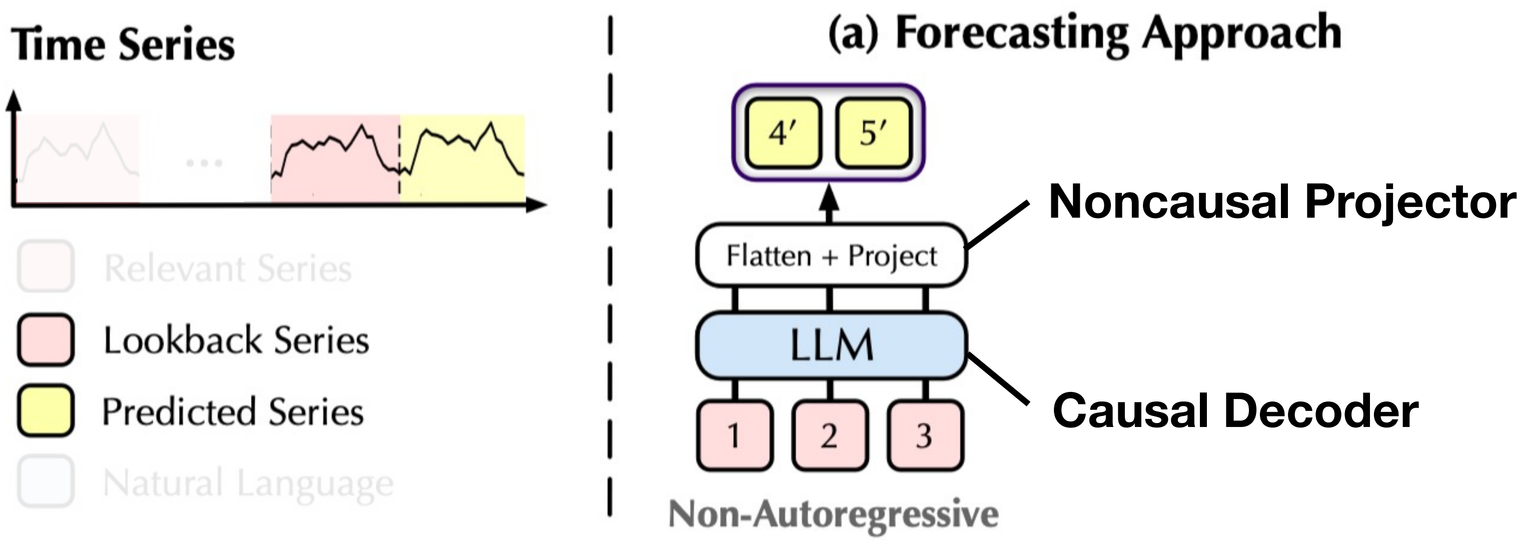
Rethinking Previous LLM4TS Methods

Insufficient utilization of LLMs is caused by several inconsistencies



Casual mask inside each LLM block

✗ **Architecture:** Previous works adapt LLMs, which are GPT-style causal decoders, as encoder-only models in a BERT-style



☹ **The token causality are broken in the last projector**

Rethinking Previous LLM4TS Methods

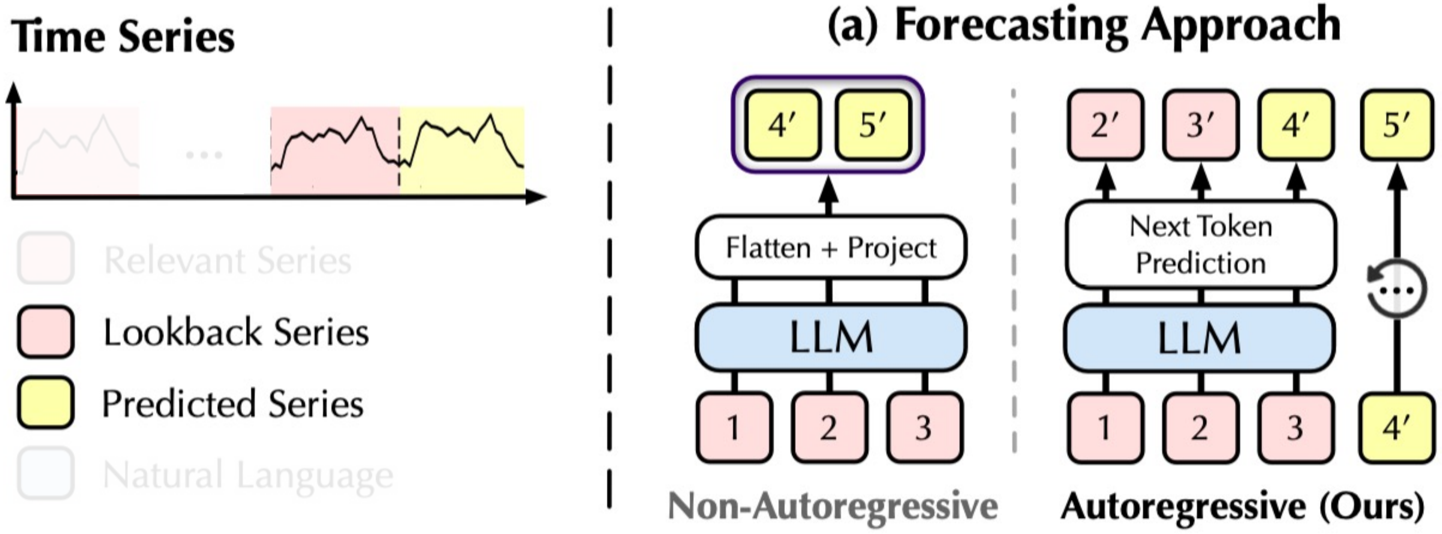
Insufficient utilization of LLMs is caused by several inconsistencies

$$P(\mathcal{U}) = \prod_{i=1}^{N} p(u_i | u_{<i})$$

Multiple supervision under different lengths

Inference with different lengths of input tokens

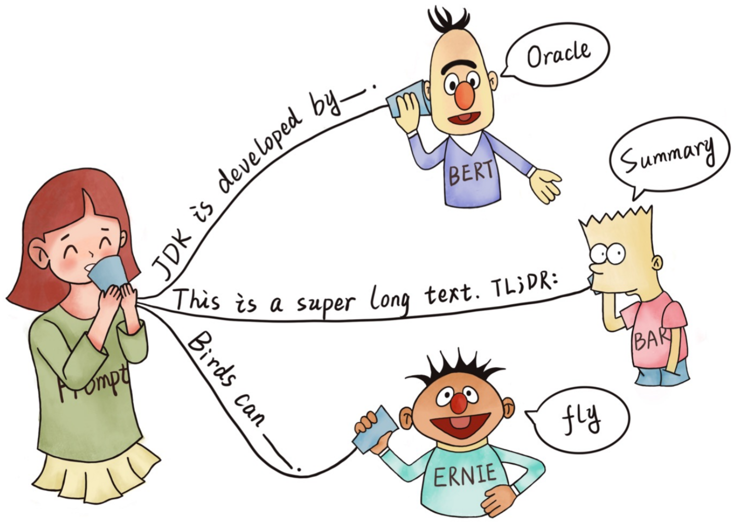
✗ **Autoregression:** LLM predicts the next tokens iteratively, while prevalent forecasters obtain all tokens in one step



☹ **The outcome forecaster is only available for specific length**

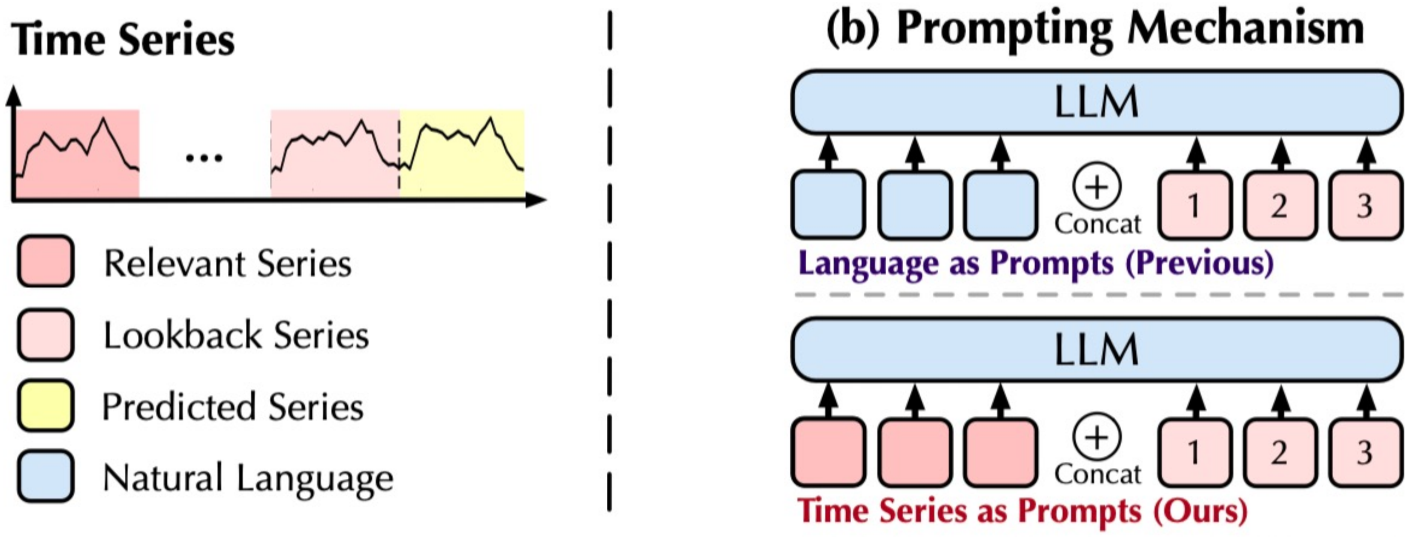
Revitalize LLMs for Time Series Modality

Exploration of advanced capabilities of language models



Prompts aim to elicit better responses from large models

- **Prompting:** we formulate time series as prompts, extending the context for prediction beyond the lookback window



☹ **Language prompts for TSF lead to modality gap**

Liu et al. Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing. ACM 2023.

Revitalize LLMs for Time Series Modality

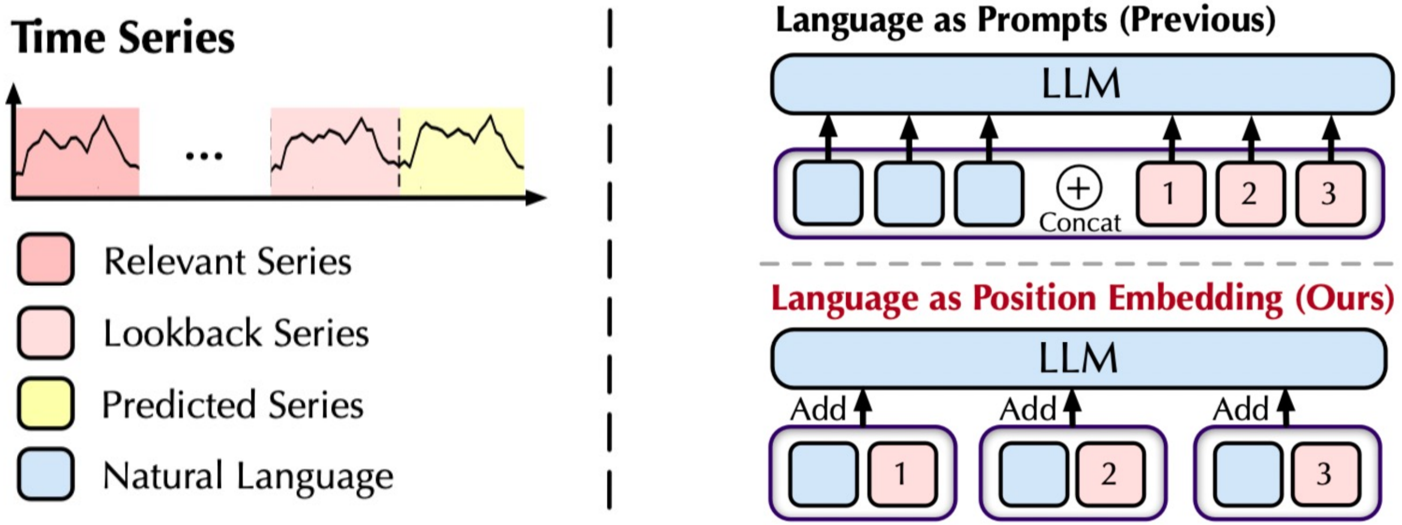
Exploration of advanced capabilities of language models

The Electricity Transformer Temperature (ETT) indicates the electric power long-term deployment. Each data point consists of the target oil temperature and 6 power load features …
Below is the information about the input time series:

[BEGIN DATA]
***
[Domain]: We usually observe that electricity consumption peaks at noon, with a significant increase in transformer load
***
[Instruction]: Predict the next <H> steps given the previous <T> steps information attached
***
[Statistics]: The input has a minimum of <min_val>, a maximum of <max_val>, and a median of <median_val>. The overall trend is <upward or downward>. The top five lags are <lag_val>.
[END DATA]

Delicate and long prompts designed for time series

- **Multimodal:** we use LLM-embedded textual timestamps to utilize chronological information and align multivariate series

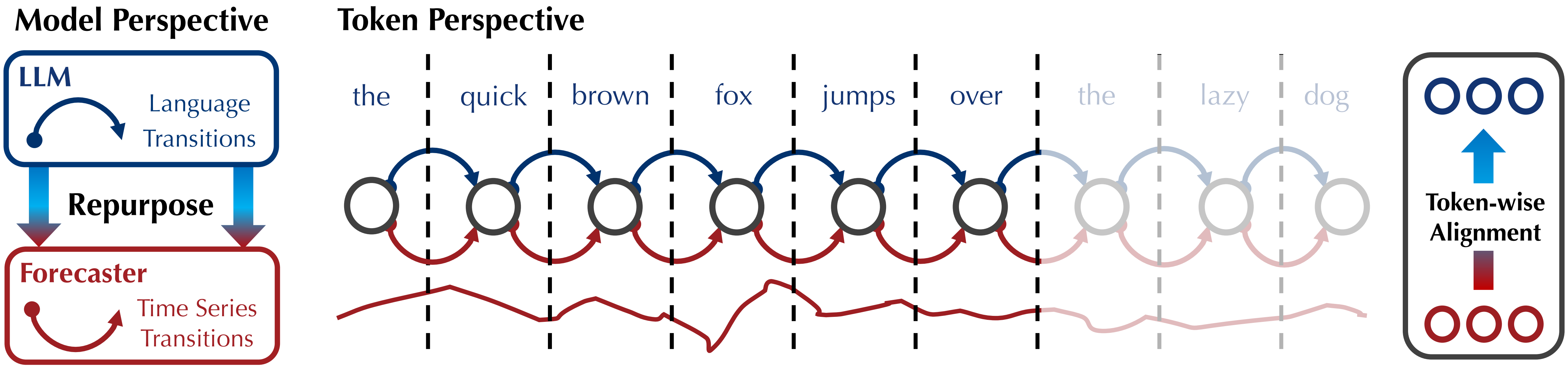


☹ **Language prompts for TSF lead to excessive contexts**

Jin et al. Time-LLM: Time Series Forecasting by Reprogramming Large Language Models. ICLR 2024.

Key Idea

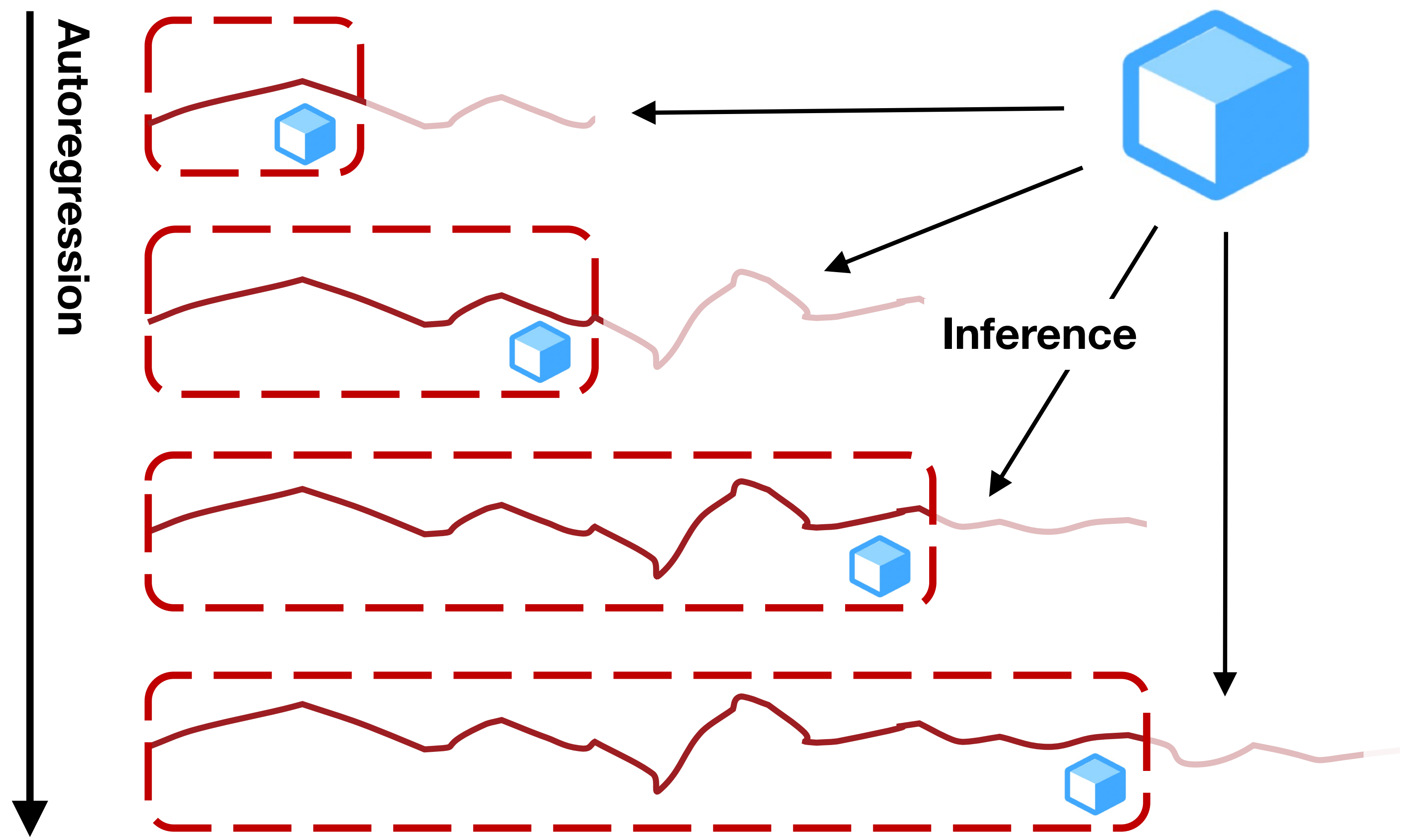
Language token transitions are general-purpose and transferable



- ✓ **Approach:** Reuse the general-purpose token transition
- ✓ **Alignment**: Embed time series into latent language representations
- ✓ **Potentials:** Autoregressive generation with inherited LLM capabilities

Key Idea

Autoregressive LLMs are arbitrary-length time series forecasters



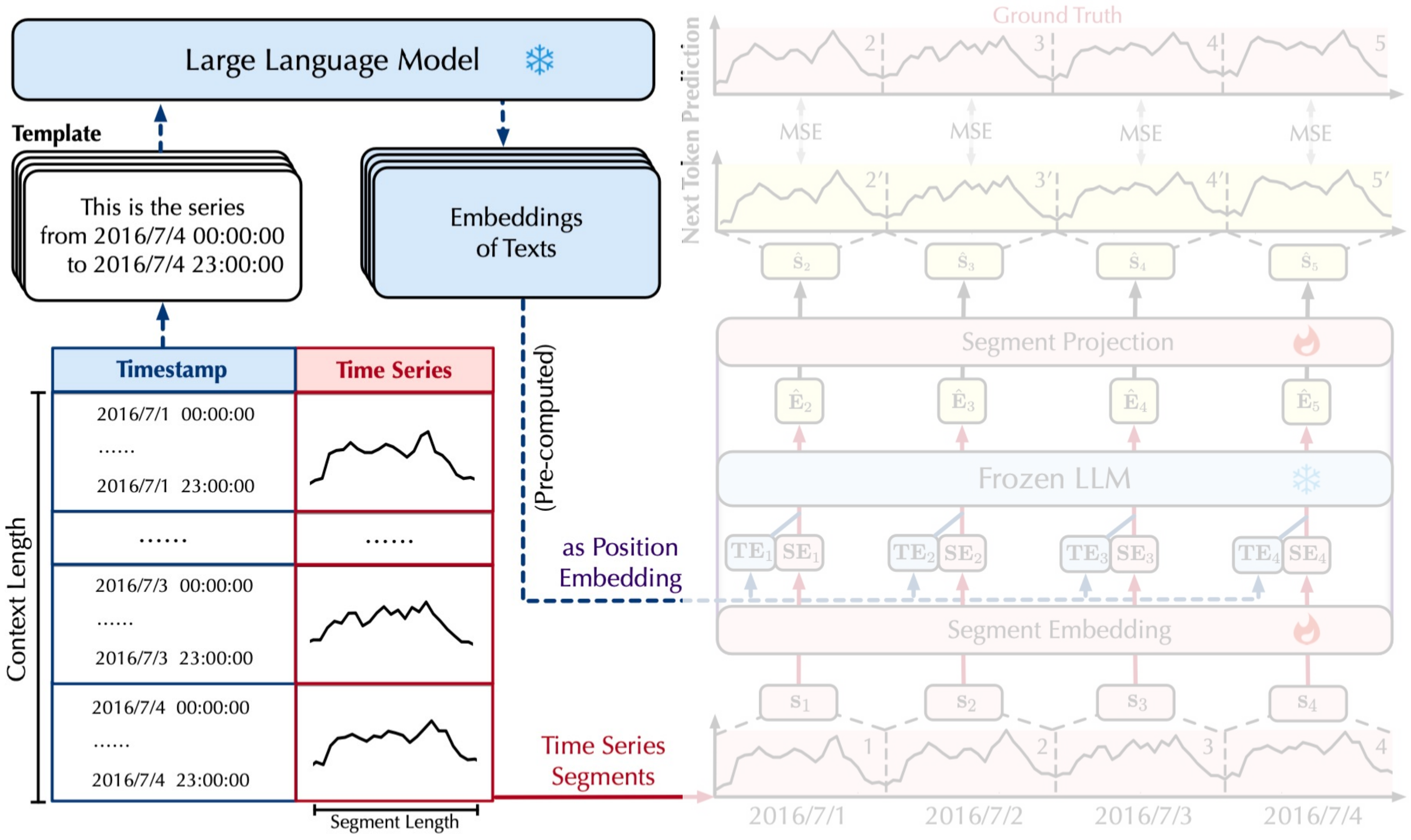
$$P(\mathcal{U}) = \prod_{i=1}^{N} p(u_i | u_{<i})$$

Token-wise supervision

$$f : (\mathbf{x}_{1:L}, \mathbf{a}_{1:L+F}) \mapsto \hat{\mathbf{x}}_{L+1:L+F}.$$

- ✓ Arbitrary lookback length L
- ✓ Arbitrary prediction length F
- ✓ Covariates: $\mathbf{a}_{1:L+F}$

Method Pipeline



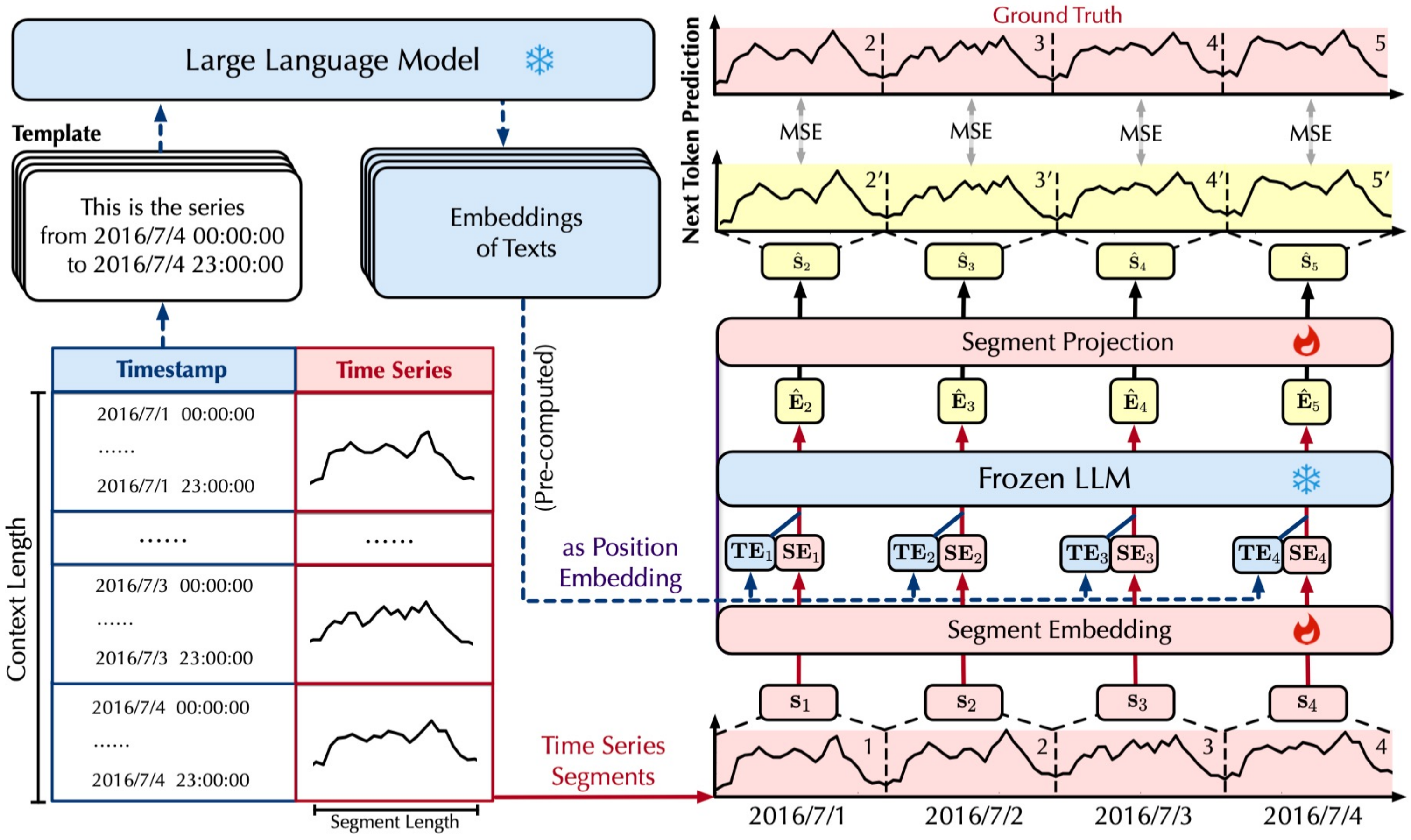
Tokenization: regard time series segments as basic language tokens

Modality-Mixing: Incorporate textual covariates (timestamp) to align variates

Freeze the LLM: Train minimal parameters by next token prediction

Inference: Generate arbitrary-length time series autoregressively like LLMs

Method Pipeline



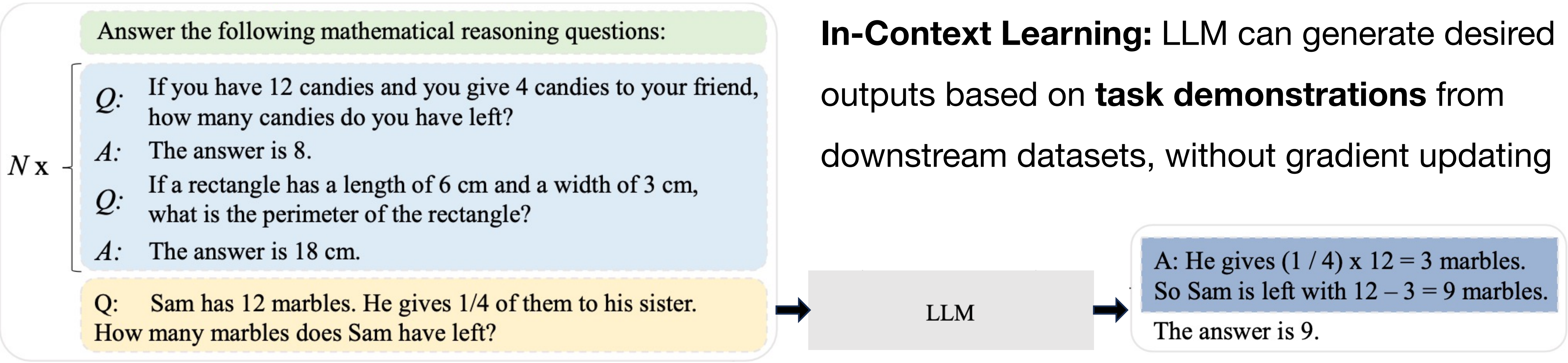
Tokenization: regard time series segments as basic language tokens

Modality-Mixing: Incorporate textual covariates (timestamp) to align variates

Freeze the LLM: Train minimal parameters by next token prediction

Inference: Generate arbitrary-length time series autoregressively like LLMs

In-Context Learning



In-Context Learning: LLM can generate desired outputs based on **task demonstrations** from downstream datasets, without gradient updating

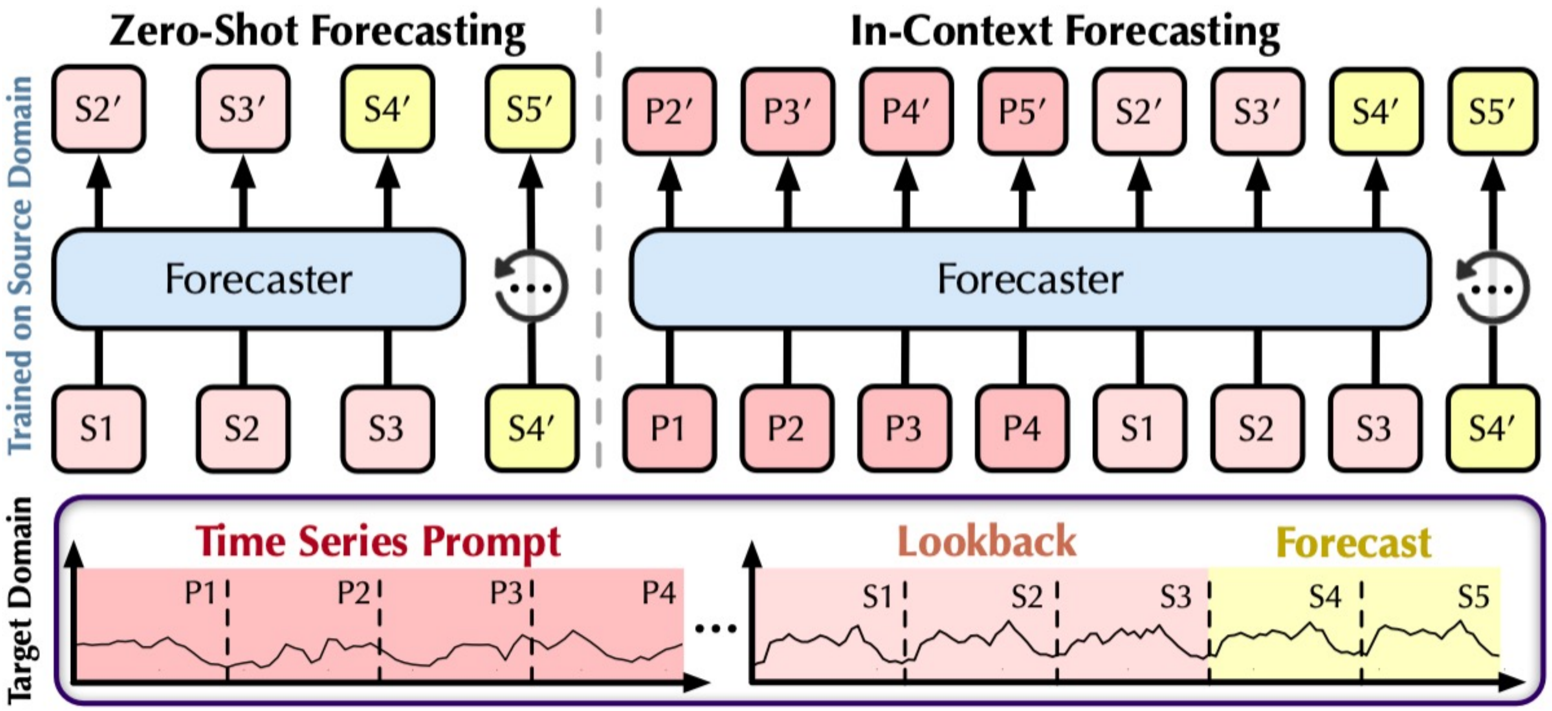
Task Demonstrations: Question-answer pairs in natural language, from an unseen task

Inference: Combine the current question with task demonstrations (prompt) as the input

Based on the token-wise alignment and full reutilization of token transition, AutoTimes can seamlessly transfer ICL to the time series modality

In-Context Forecasting

We propose in-context forecasting for time series



Time Series Forecasting:

$$f : (\mathbf{x}_{1:L}, \mathbf{a}_{1:L+F}) \mapsto \hat{\mathbf{x}}_{L+1:L+F}.$$

Time Series Prompt:

$$\mathcal{C} = \{tsp^{(j)} = \mathbf{x}_{\le t_j}\}, t_j \le L.$$

Earlier historical time series
(perhaps non-consecutive)

In-Context Forecasting:

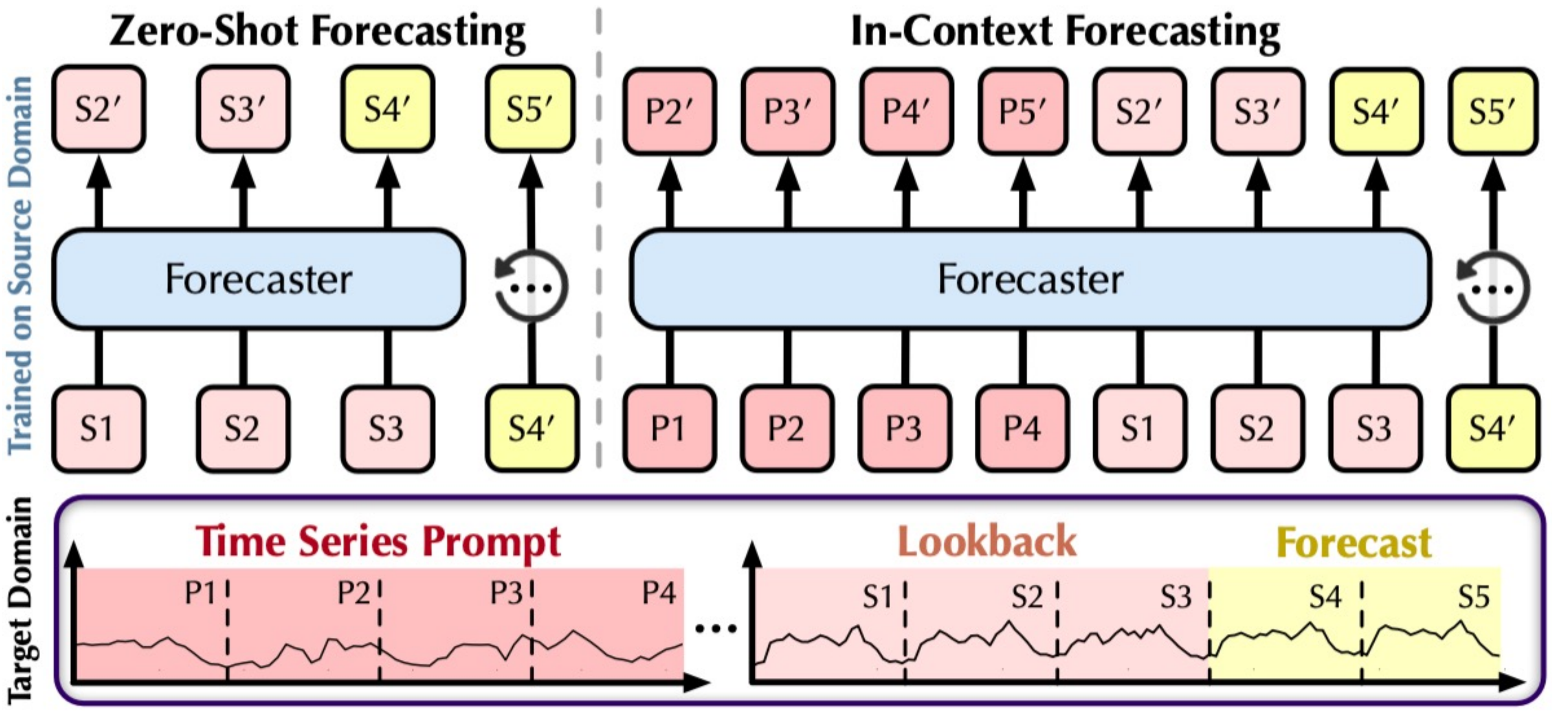
$$f : (\underline{\underline{\mathcal{C}}}, \mathbf{x}_{1:L}, \mathbf{a}_{1:L+F}) \mapsto \hat{\mathbf{x}}_{L+1:L+F}.$$

Prediction Demonstrations: Retrieve time series as prompts from the target domain

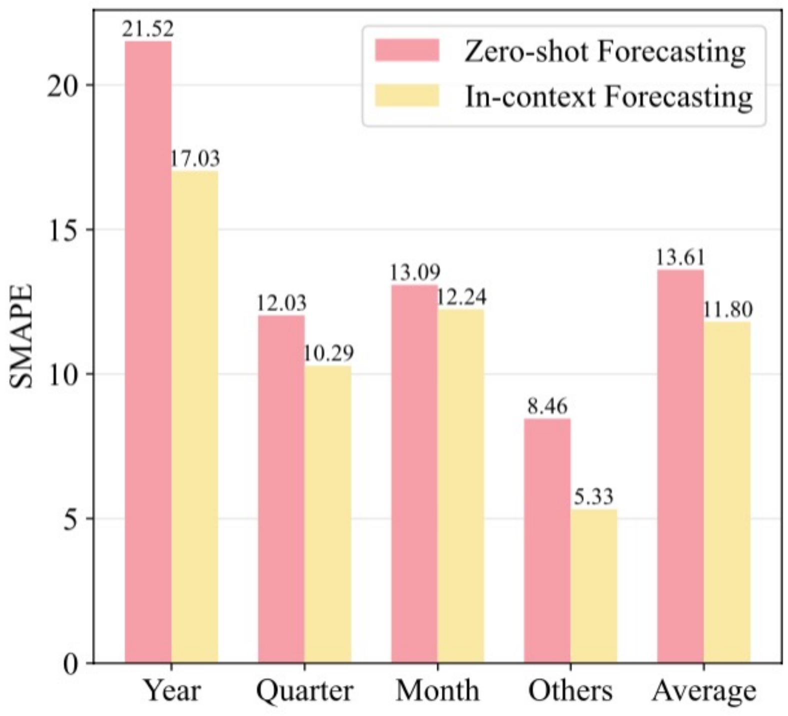
Inference: Input "prompt-lookback" sentence into our model without updating parameters

In-Context Forecasting

We propose in-context forecasting for time series



Enhanced performance with prompts



Prediction Demonstrations: Retrieve time series as prompts from the target domain

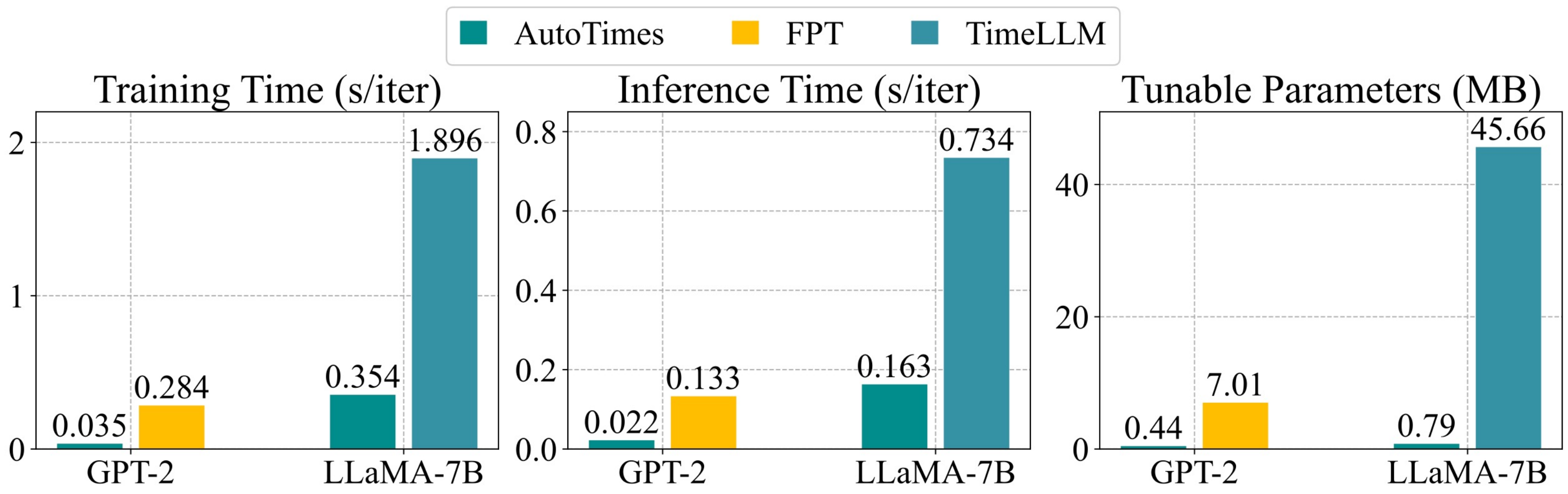
Inference: Input "prompt-lookback" sentence into our model without updating parameters

Comparison of LLM4TS

Quality assessments (none of prior LLM4TS methods achieved all three)

Method	**AutoTimes**	TimeLLM [15]	UniTime [21]	FPT [49]	LLMTime [13]	TEST [34]	TEMPO [7]	PromptCast [44]
Autoregressive	✓	✗	✗	✗	✓	✗	✗	✗
Freeze LLM	✓	✓	✗	✗	✓	✓	✗	✓
Multimodal	✓	✓	✓	✗	✗	✓	✓	✓

Minimal tunable parameters -> Better performance/model efficiency



15min to repurpose **LLaMA-7B** on **a RTX 3090-24G**

(8 x A100 for Time-LLM)

Ablation Study

True utilization of large language model (different from non-autoregressive LLM4TS methods)

Table 6: We follow the protocol of LLM4TS ablation studies [35] to verify whether the LLM is truly useful in our AutoTimes: (1) *w/o LLM* replaces the language model entirely and passing input tokens directly to the last layer; (2) *LLM2Attn* replaces the language model with a single multi-head attention layer; (3) *LLM2Trsf* replaces the language model with a single transformer block.

Dataset	ETTh1								ECL							
Type	**AutoTimes**		w/o LLM		LLM2Attn		LLM2Trsf		**AutoTimes**		w/o LLM		LLM2Attn		LLM2Trsf	
Metric	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
Pred-96	**0.360**	**0.400**	0.365	0.399	0.383	0.404	0.377	0.401	**0.129**	**0.225**	0.171	0.263	0.156	0.255	0.162	0.263
Pred-192	**0.388**	**0.419**	0.405	0.425	0.414	0.422	0.406	0.420	**0.147**	**0.241**	0.192	0.282	0.178	0.276	0.189	0.287
Pred-336	**0.401**	**0.429**	0.429	0.441	0.431	0.432	0.421	0.431	**0.162**	**0.258**	0.216	0.304	0.198	0.295	0.216	0.309
Pred-720	**0.406**	**0.440**	0.450	0.468	0.456	0.454	0.449	0.452	**0.199**	**0.288**	0.264	0.342	0.230	0.320	0.258	0.340

Tan et al. Are Language Models Actually Useful for Time Series Forecasting? NeurIPS 2024.

Forecasting Performance

Long-term forecasting (one-for-all rolling forecasting)

Models	**AutoTimes**		TimeLLM [15]		UniTime [21]		FPT [48]		iTrans. [22]		DLinear [44]		PatchTST [26]		TimesNet [41]	
Metric	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTh1	**0.389**	**0.422**	0.412	0.437	0.683	0.596	0.429	0.439	0.421	0.445	0.426	0.444	0.409	0.430	0.495	0.491
ECL	**0.159**	**0.253**	0.181	0.288	0.325	0.399	0.184	0.284	0.164	0.258	0.165	0.265	0.169	0.268	0.201	0.303
Weather	0.235	0.273	**0.225**	**0.266**	0.461	0.459	0.228	**0.266**	0.266	0.291	0.239	0.291	0.226	0.268	0.264	0.293
Traffic	**0.374**	**0.264**	0.410	0.303	0.584	0.367	0.461	0.326	0.384	0.274	0.423	0.298	0.391	0.275	0.602	0.322
Solar.	**0.197**	**0.242**	0.263	0.335	0.392	0.462	0.236	0.303	0.213	0.291	0.222	0.283	0.202	0.269	0.213	0.295

Short-term forecasting (in-distribution)

Models		**AutoTimes**	TimeLLM	FPT	Koopa	N-HiTS	DLinear	PatchTST	TimesNet	FiLM	N-BEATS
Average	sMAPE	**11.831**	11.983	11.991	11.863	11.960	12.418	13.022	11.930	12.489	11.910
	MASE	**1.585**	1.595	1.600	1.595	1.606	1.656	1.814	1.597	1.690	1.613
	OWA	**0.850**	0.859	0.861	0.858	0.861	0.891	0.954	0.867	0.902	0.862

Zero-shot forecasting (out-of-distribution)

Models	**AutoTimes**	FPT	DLinear	PatchTST	TimesNet	NSFormer	FEDFormer	Informer	Reformer
M4 → M3	**12.75**	13.06	14.03	13.06	14.17	15.29	13.53	15.82	13.37
M3 → M4	**13.036**	13.125	15.337	13.228	14.553	14.327	15.047	19.047	14.092

One LLM-forecasters can outperform each deep models trained on specific lengths

State-of-the-art performance

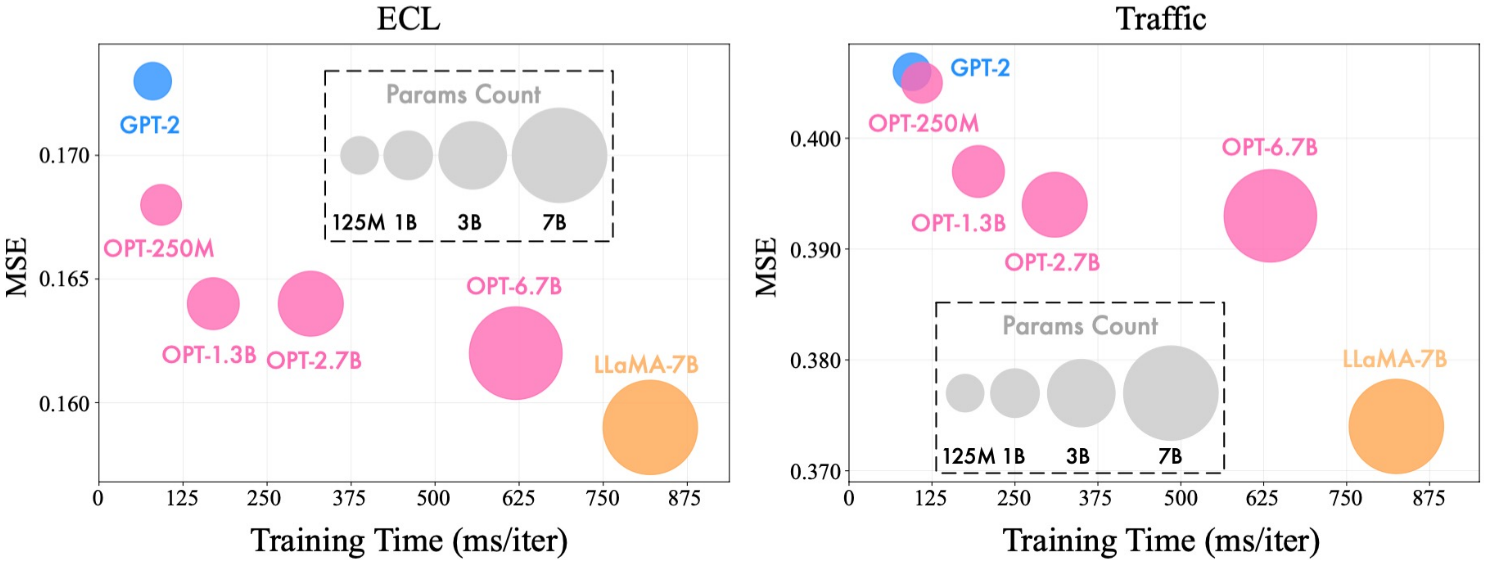
Compatibility of Language Models

AutoTimes configuration

Base LLM	GPT-2 (124M)	OPT-350M	OPT-1.3B	OPT-2.7B	OPT-6.7B	LLaMA-7B
Hidden Dim.	768	1024	2048	2560	4096	4096
Embedding	2-layer MLP	2-layer MLP	2-layer MLP	2-layer MLP	2-layer MLP	Linear
Trainable Param. (M)	0.44	0.58	1.10	1.36	2.15	0.79

Large model tuned with small amount of params

Scaling law of LLM-forecasters

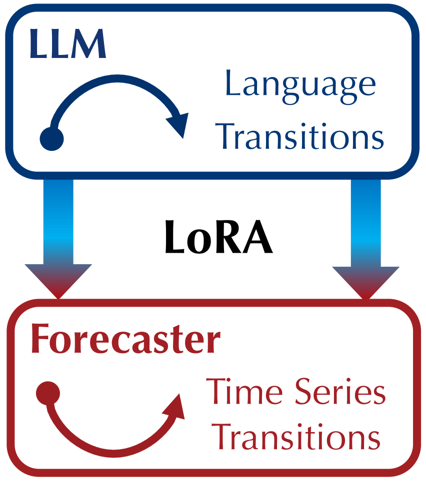


Larger language models, more accurate predictions

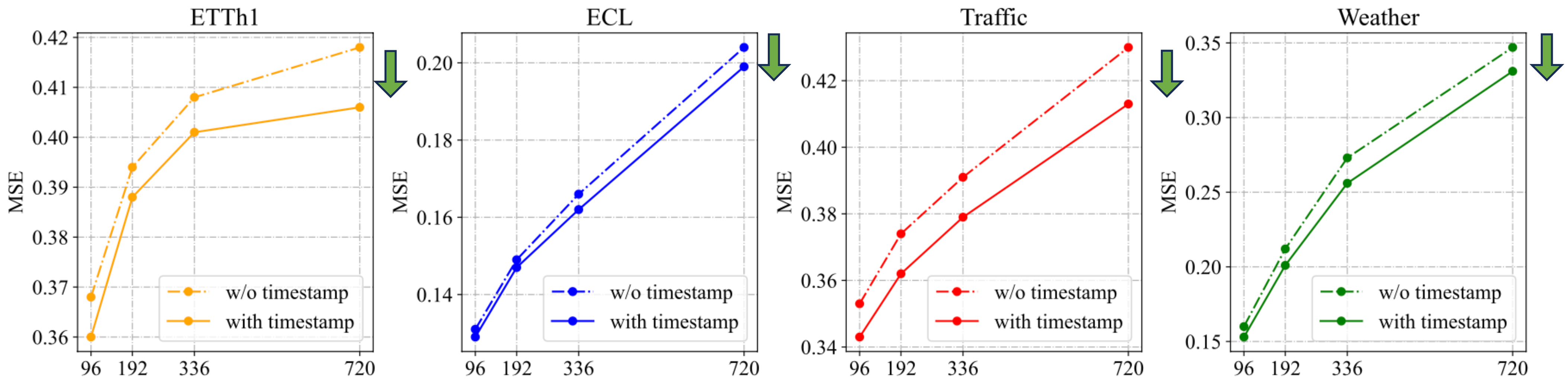
Method Analysis

Adopting low-rank adaptation can achieves better predictions

Datasets	ETTh1		ECL		Weather		Traffic		Solar-Energy	
Metric	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
AutoTimes	0.397	0.425	0.173	0.266	0.242	0.278	0.406	0.276	0.207	0.246
AutoTimes + LoRA	**0.396**	0.425	**0.161**	**0.255**	**0.231**	**0.268**	**0.396**	**0.275**	**0.201**	**0.243**

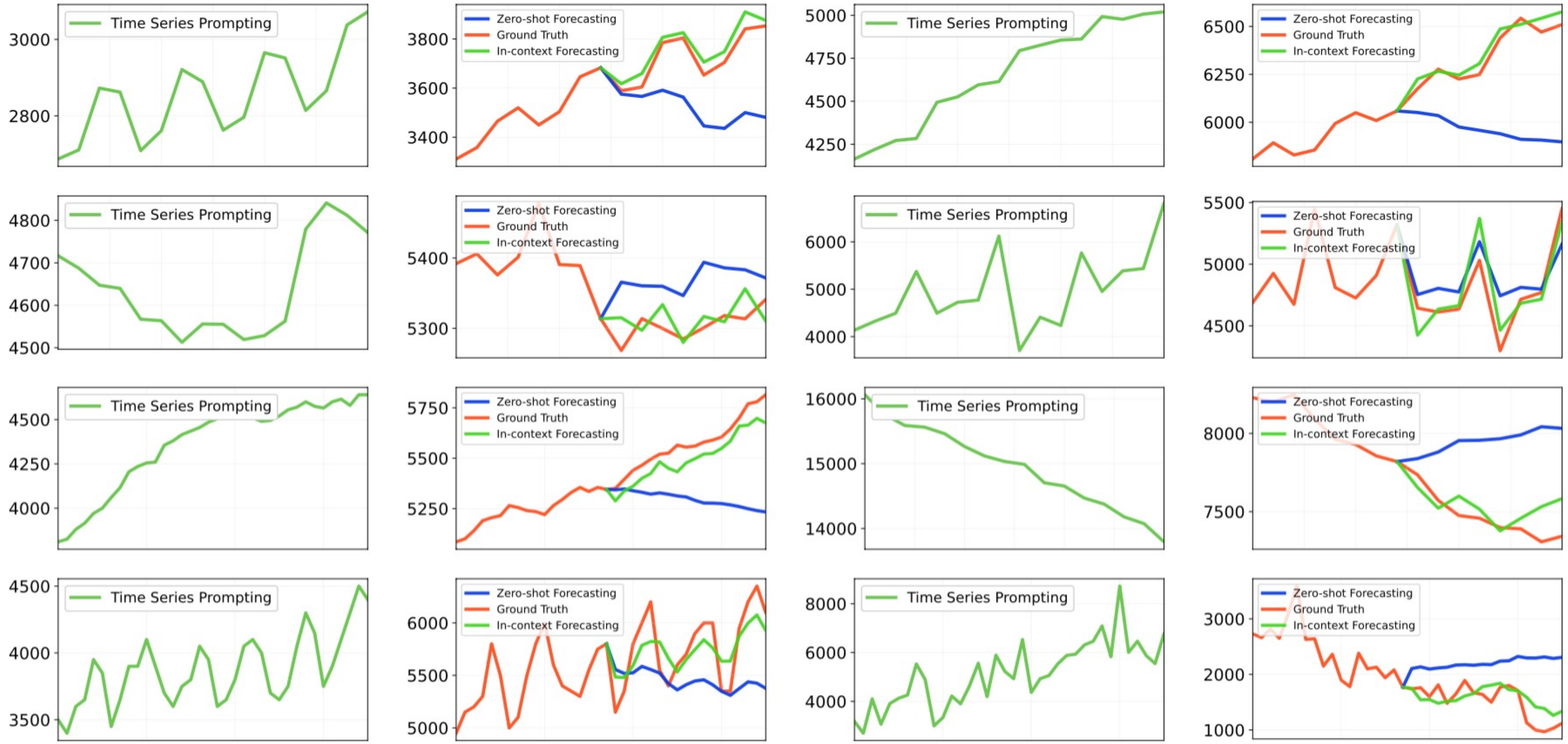


Textual Timestamps as position embeddings are effective



In-Context Forecasting Showcases

Facilitate an interactive experience of forecasting via prediction samples



In-Context Forecasting Showcases

Facilitate an interactive experience of forecasting via prediction samples

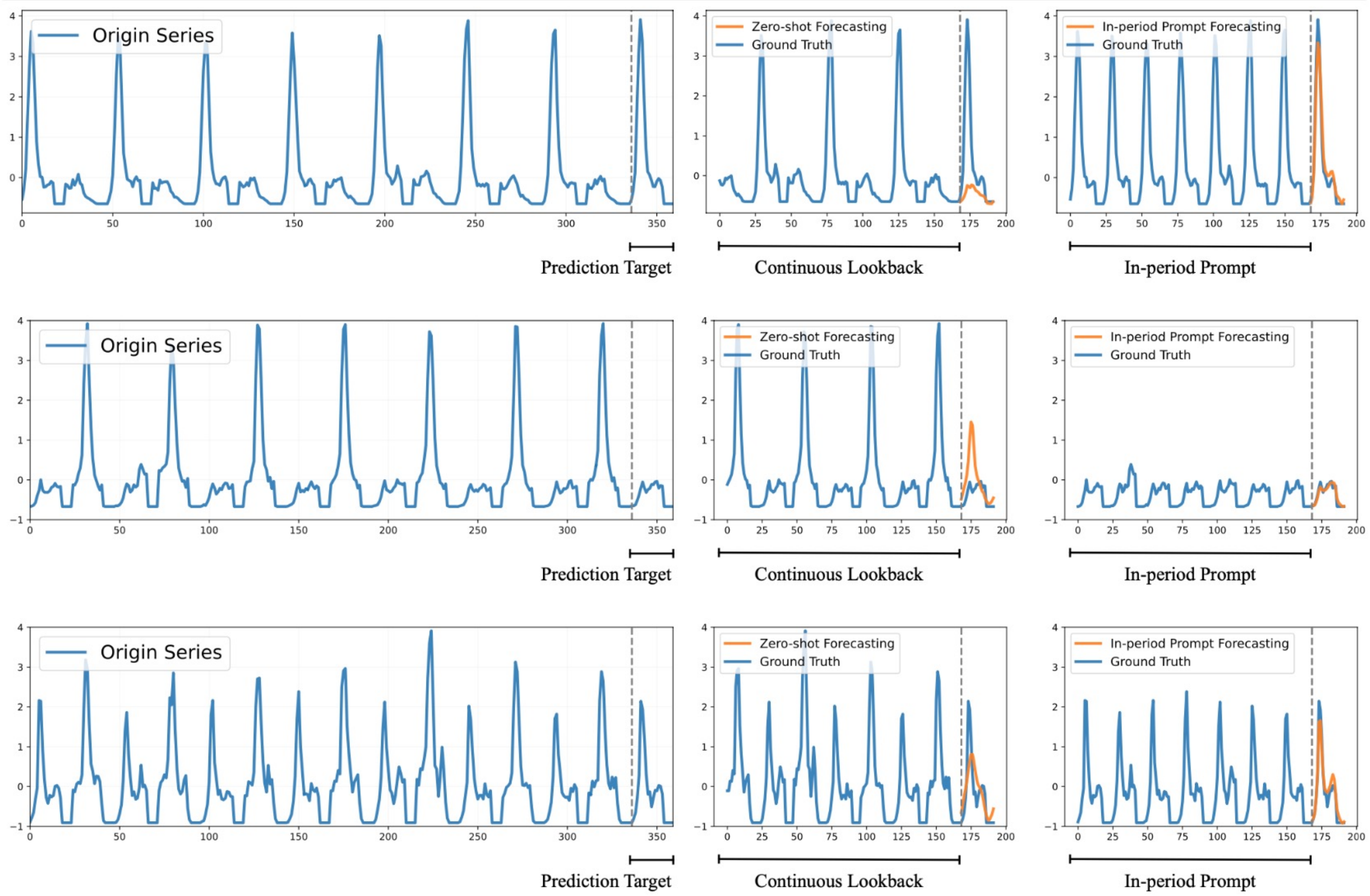
Table 20: Strategies to select time series prompts based on periodicity for in-context forecasting.

Context for prediction	ETTh1-OT	ETTh2-OT	ETTm1-OT	ETTm2-OT	Average Err.
P.0: Zero-Shot (Input-288))	0.0673	0.1637	0.0424	0.1669	0.1101
P.1: Zero-Shot (Input-672)	0.0657	0.1538	0.0415	0.1701	0.1078
P.2: Ahead-Period (Input-672)	**0.0645**	**0.1513**	**0.0399**	**0.1629**	**0.1047**
P.3: Ahead-Random (Input-672)	0.0666	0.1621	0.0407	0.1719	0.1103
P.4: Fixed Prompt (Input-672)	0.0769	0.1859	0.0512	0.2104	0.1311
P.5: Other-Variates (Input-672)	0.1263	0.1780	0.0852	0.2297	0.1548

- **Ahead-Period:** select the Ahead-24 (daily period) series of the original lookback series
- **Ahead-Random:** randomly select the previous series of the original lookback series
- **Fixed Prompt:** fixed as the first 384 time points from the same variate
- **Other Variate:** uniformly selected as Ahead-24 series, but comes from other variate

In-Context Forecasting Showcases

Facilitate an interactive experience of forecasting via prediction samples



Compared with simply extending lookback length, in-context forecasting aims to improve context efficiency

Take-away message: utilize inter-periodic, consecutive, and relevant prompts

Open Source

- ✓ **Efficient:** Only 15min to repurpose LLaMA-7B on **one single RTX 3090-24G** (8 x A100 for Time-LLM)
- ✓ **Compatible: Support any decoder-only LLMs**: GPT, LLaMA of different sizes, the OPT family…
- ✓ **Well-organized:** Pretty code implementations for multi-step **autoregressive forecasting** and **in-context forecasting**

GitHub: https://github.com/thuml/AutoTimes

README MIT license

AutoTimes (Large Language Models for Time Series Forecasting)

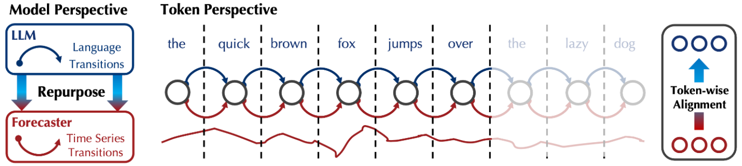
The repo is the official implementation: AutoTimes: Autoregressive Time Series Forecasters via Large Language Models.

> **Time Series Forecasting**: AutoTimes repurpose LLMs as autoregressive multivariate time series forecasters. Different from previous models, our repurposed forecaster can be applied on various lookback/forecast lengths.
>
> **Zero-Shot Forecasting**: AutoTimes takes advantage of LLM's general-purposed token transition as the future extrapolation of time series, demonstrating good performance without downstream samples.
>
> **In-Context Forecasting**: We propose in-context forecasting for the first time, where time series prompts can further incorporated into the context to enhance forecasting.
>
> **Easy-to-Use**: AutoTimes is compatiable with any decoder-only large language models, demonstrating generality and proper scaling behavior.

Updates

► News (2024.10): AutoTimes has been accepted by **NeurIPS 2024**. A revised version (**25 Pages**) is now available, including prompt engineering of in-context forecasting, adaptation cost evaluations, textual embeddings of metadata, and low-rank adaptation techique.

► News (2024.08): Recent work (code) has also raised questions about previous non-autoregressive LLM4TS methods. We conduct ablations here, highlighting AutoTimes can truly utilize LLMs. Instead of adopting LLMs in a BERT-style, **the general-purpose token transition is transferable among time series and natural language**.



► **News** (2024.2) Scripts for the above tasks in our paper are all available.

Thank You!

Yong Liu

https://wenweithu.github.io/

GitHub: https://github.com/thuml/AutoTimes