41st IEEE International Conference on Data Engineering

HONG KONG SAR, CHINA | MAY 19 – 23, 2025

Accurate and Efficient Multivariate Time Series Forecasting via Offline Clustering

Yiming Niu[1], Jinliang Deng[2,3], Lulu Zhang[1], Zimu Zhou[4], Yongxin Tong[1]

[1]Beihang University, [2]The Hong Kong University of Science and Technology,

[3]Southern University of Science and Technology, [4]City University of Hong Kong

Outline

- Background and Motivations
- Problem Statement
- Our Solutions
- Experiments
- Conclusion

Outline

- Background and Motivations
- Problem Statement
- Our Solutions
- Experiments
- Conclusion

Background and Motivation

- **Multivariate Time Series Forecasting (MTSF)**

MTS in Real Life

Traffic Flow

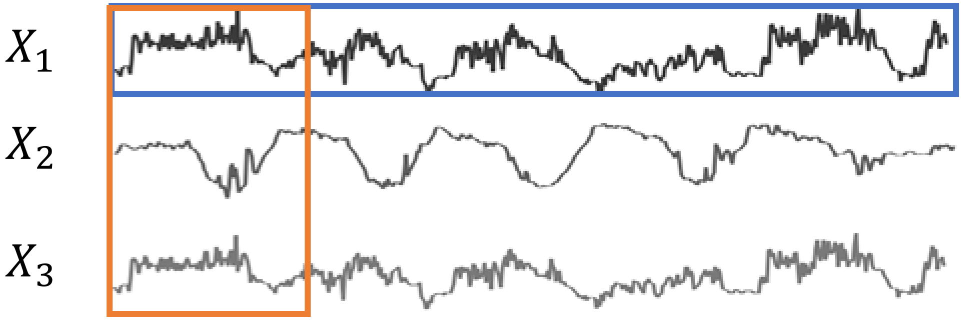
Weather Condition

Equipment Status

Power Consumption

……etc.

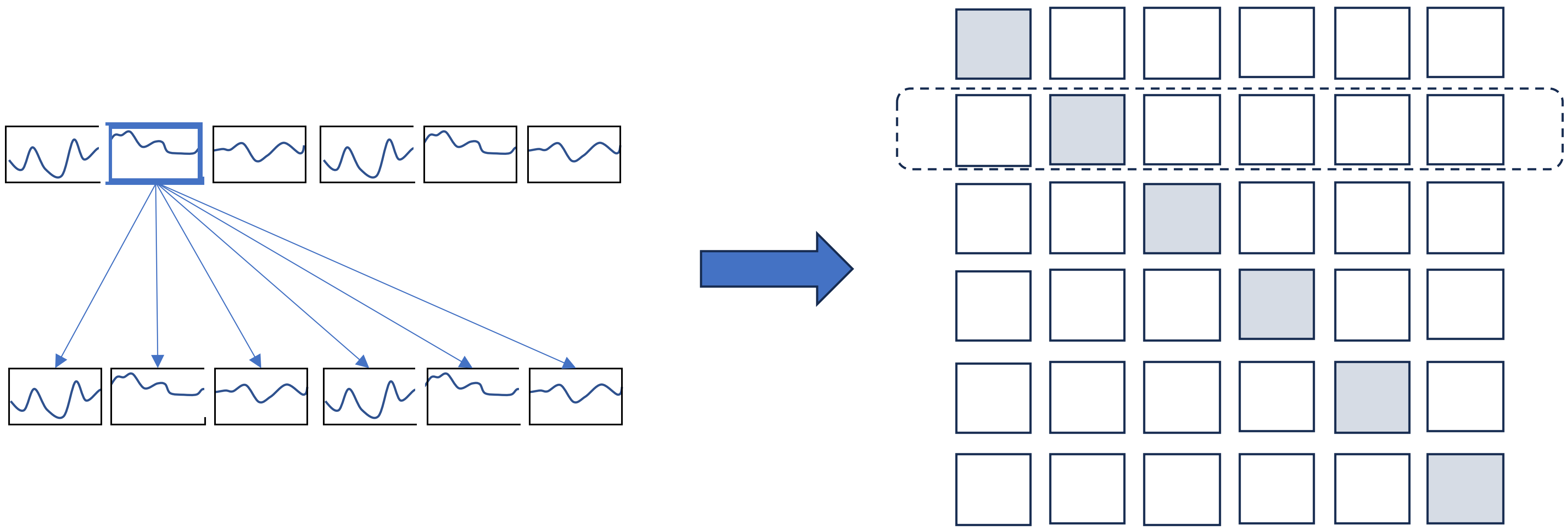
Characteristics



- **Temporal Correlation**: The dependence of a variable's values on its historical observations, capturing dynamic patterns like trends, seasonality, and periodic cycles over time.
- **Channel Correlation**: The interdependence among different variables (i.e. channels), highlighting cross-variable relationships.

Background and Motivation

- **Attention mechanism is a widely−used[1] method for modeling correlations in MTSF**



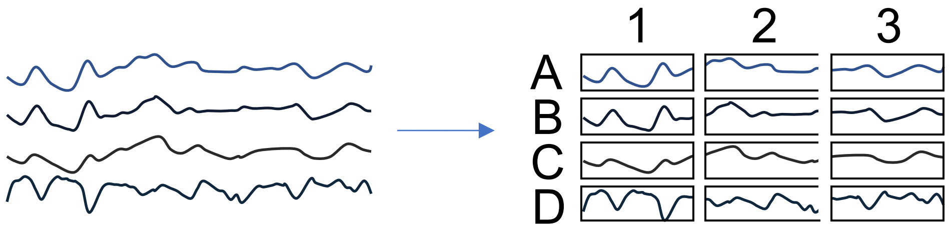
Correlations Matrix
(Attention Scores)

[1] Wen, Qingsong, et al. "Transformers in time series: a survey." *Proceedings of the International Joint Conference on Artificial Intelligence*. 2023.

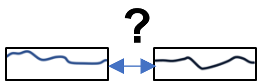
Background and Motivation

- **High cost of modeling correlations with attention mechanism leads to a tradeoff between performance and accuracy**

Correlation Modeling



For $N = 4$ channels with $L = 3$ time segments



Correlations To Be Modeled

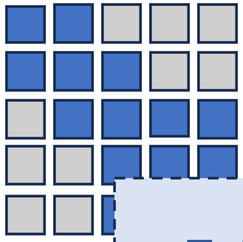
Temporal Correlation

(A1, A3), (A1, A2), (A2, A3) $O(L^2)$

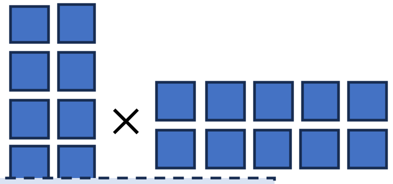
Channel Correlation

(A1, B1), (A1, C1), (A1, D1)
(B1, C1), (B1, D1), (C1, D1) $O(N^2)$

Existing Solutions

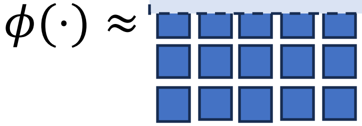


Sparse

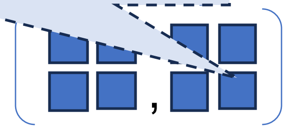


oximated
n

Existing efficient attention mechanisms are universally designed, balancing performance and accuracy



Kernelized Approximated Attention



Hierarchical Attention

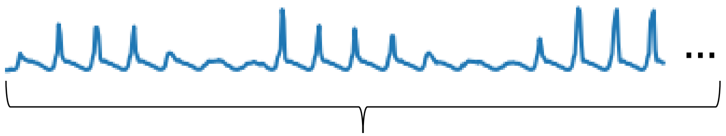
……etc.

Background and Motivation

- **Taking into account the repeating patterns inherent in time series, we may be able to push the boundary further**

Repeating Patterns

Take 1024 time steps from Electricity[2] for example

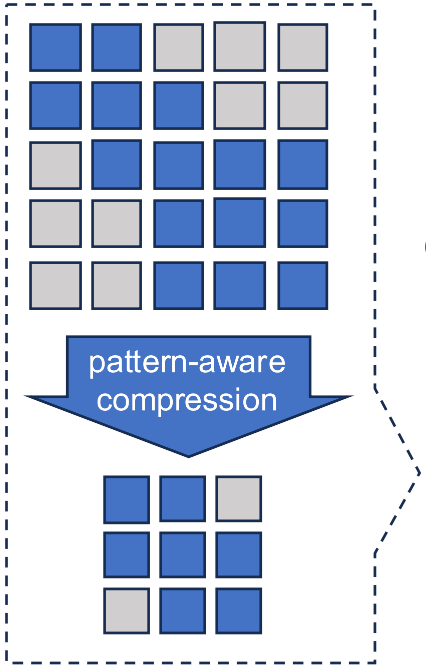


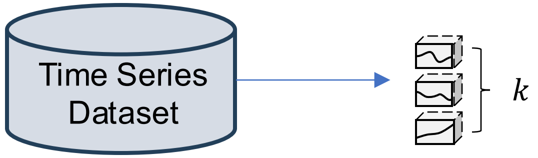
$t = 1024$ time steps

$\Updownarrow$

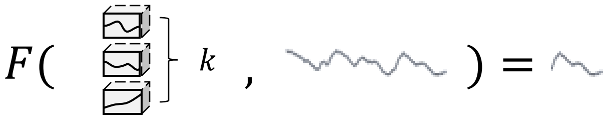
$k = 4$ patterns

Reducing the computation





① Find representative patterns



② Forecast using patterns

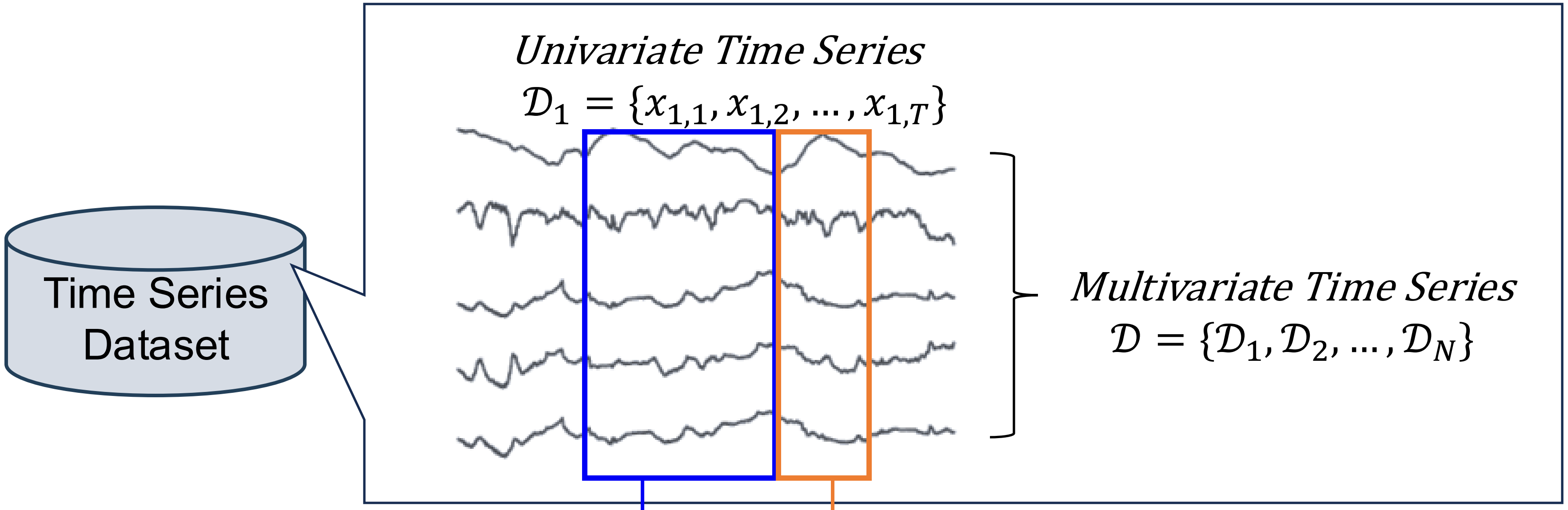
[2] https://archive.ics.uci.edu/dataset/321/electricityloaddiagrams20112014

Outline

- Background and Motivations
- Problem Statement
- Our Solutions
- Experiments
- Conclusion

Problem Statement

- **Multivariate Time Series Forecasting (MTSF)**



$\boldsymbol{x_e} = \{x_{e,1}, x_{e,2}, \dots, x_{e,L}\}$ $\boldsymbol{y_e} = \{x_{e,L+1}, x_{e,L+2}, \dots, x_{e,L+L_f}\}$

$\mathcal{X} = \{\boldsymbol{x_1}, \boldsymbol{x_2}, \dots, \boldsymbol{x_N}\}$ $\mathcal{Y} = \{\boldsymbol{y_1}, \boldsymbol{y_2}, \dots, \boldsymbol{y_N}\}$

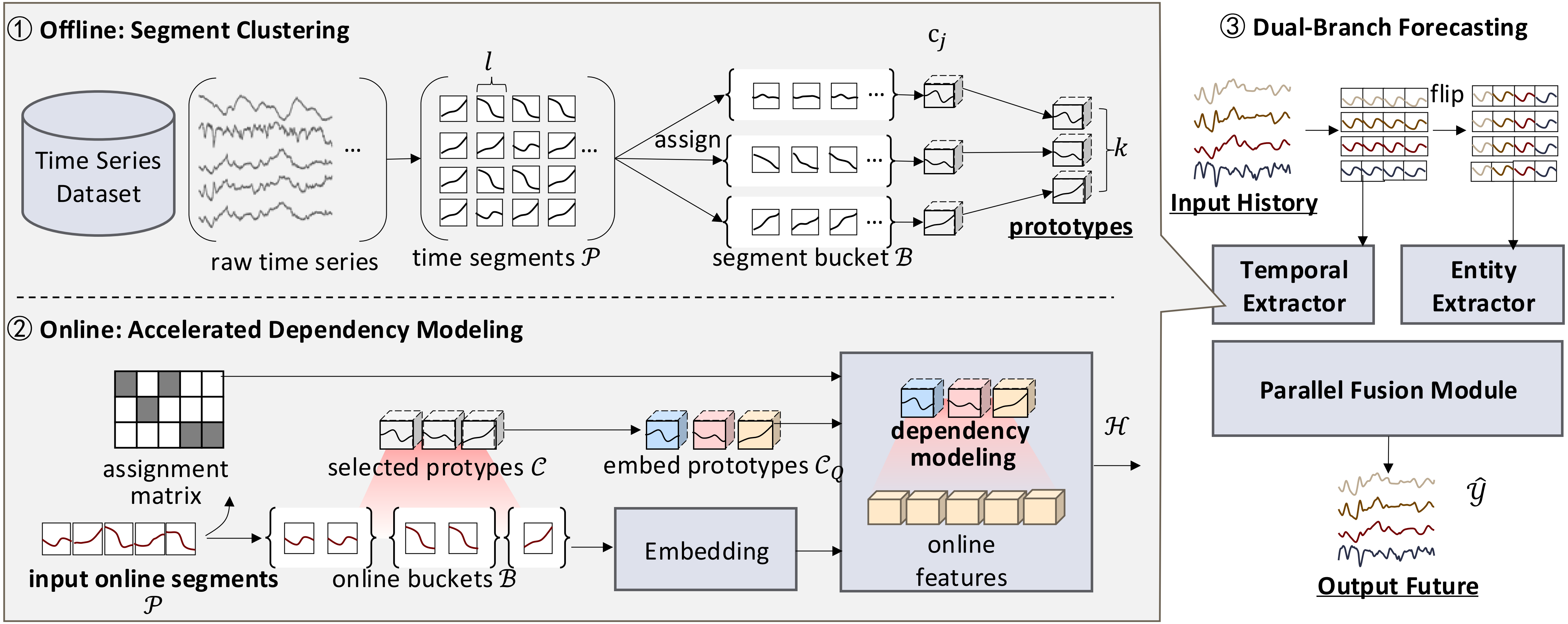
Find the forecasting function $F(\mathcal{X}) = \hat{\mathcal{Y}}$ that minimize the difference between $\mathcal{Y}$ and $\hat{\mathcal{Y}}$

Outline

- Background and Motivations
- Problem Statement
- Our Solutions
- Experiments
- Conclusion

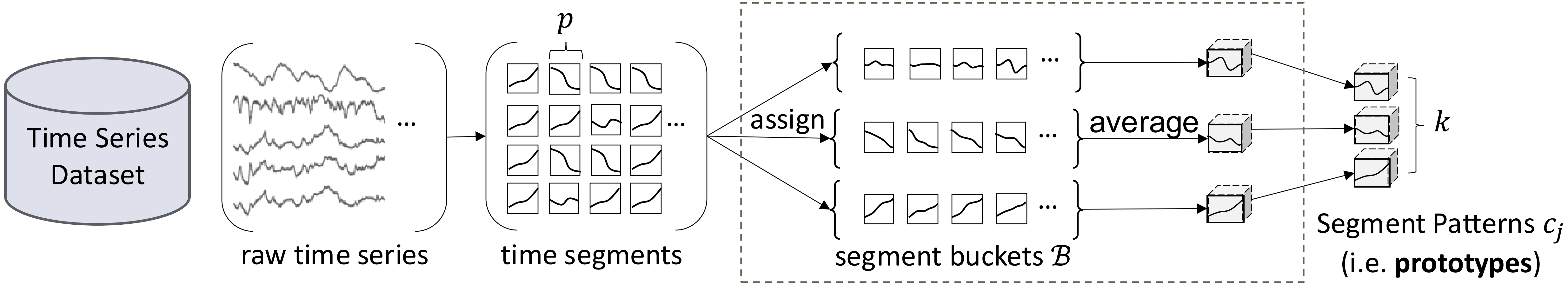
Our Solutions

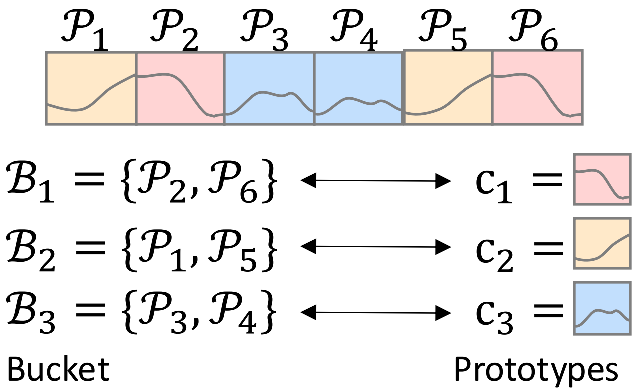
- **Overview of Forecasting via Offline Clustering (FOCUS)**



Our Solutions

- **Find patterns via segment clustering**





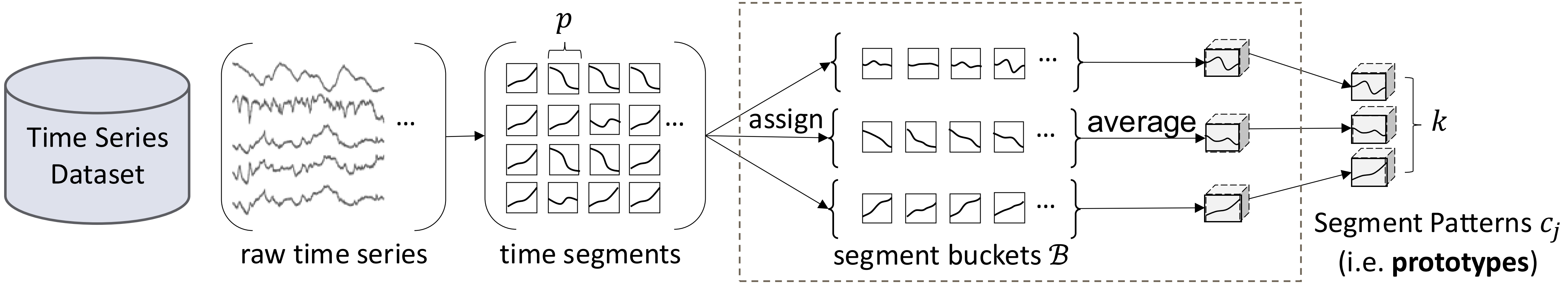
To minimize:

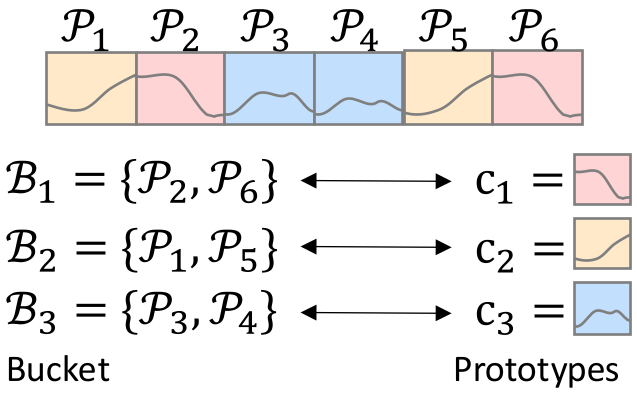
$$\mathcal{L} = \mathcal{L}_{rec}$$

Numerical Reconstruction Loss: $\mathcal{L}_{rec} = \sum_{j=1}^{k} \left\| c_j - \mathrm{mean}(\boldsymbol{\mathcal{B}}_j) \right\|$

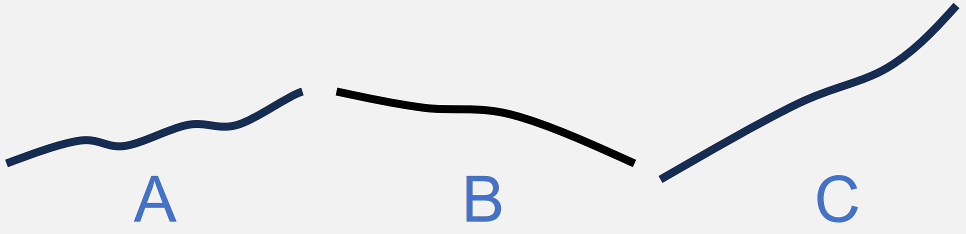
Our Solutions

- **Find patterns via segment clustering**





Is Euclidean distance suitable?

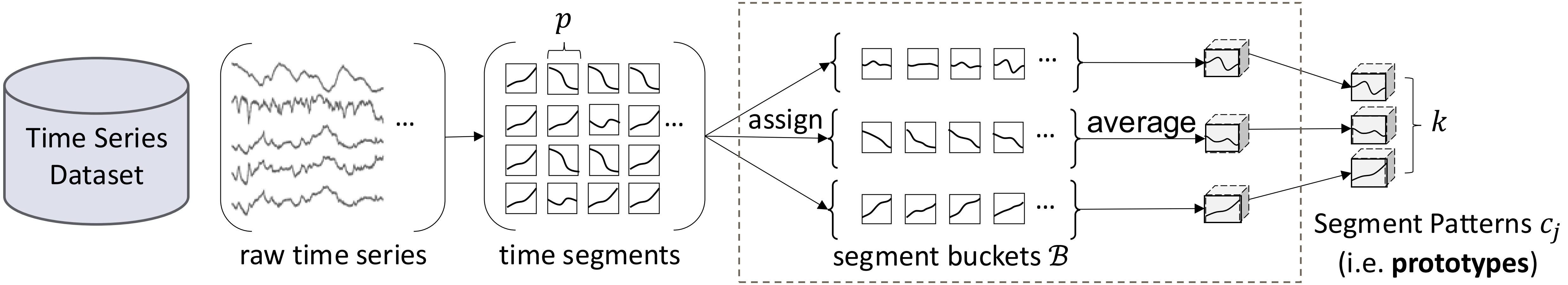


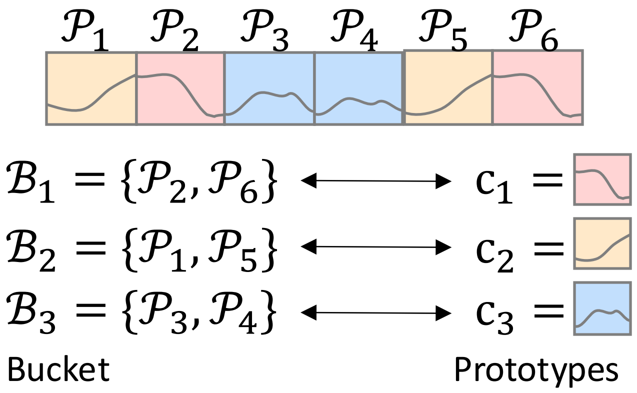
Euclidean(A,B) < Euclidean(A,C)

But A and C are more likely to indicate similar situations.

Our Solutions

- **Find patterns via segment clustering**





To minimize:

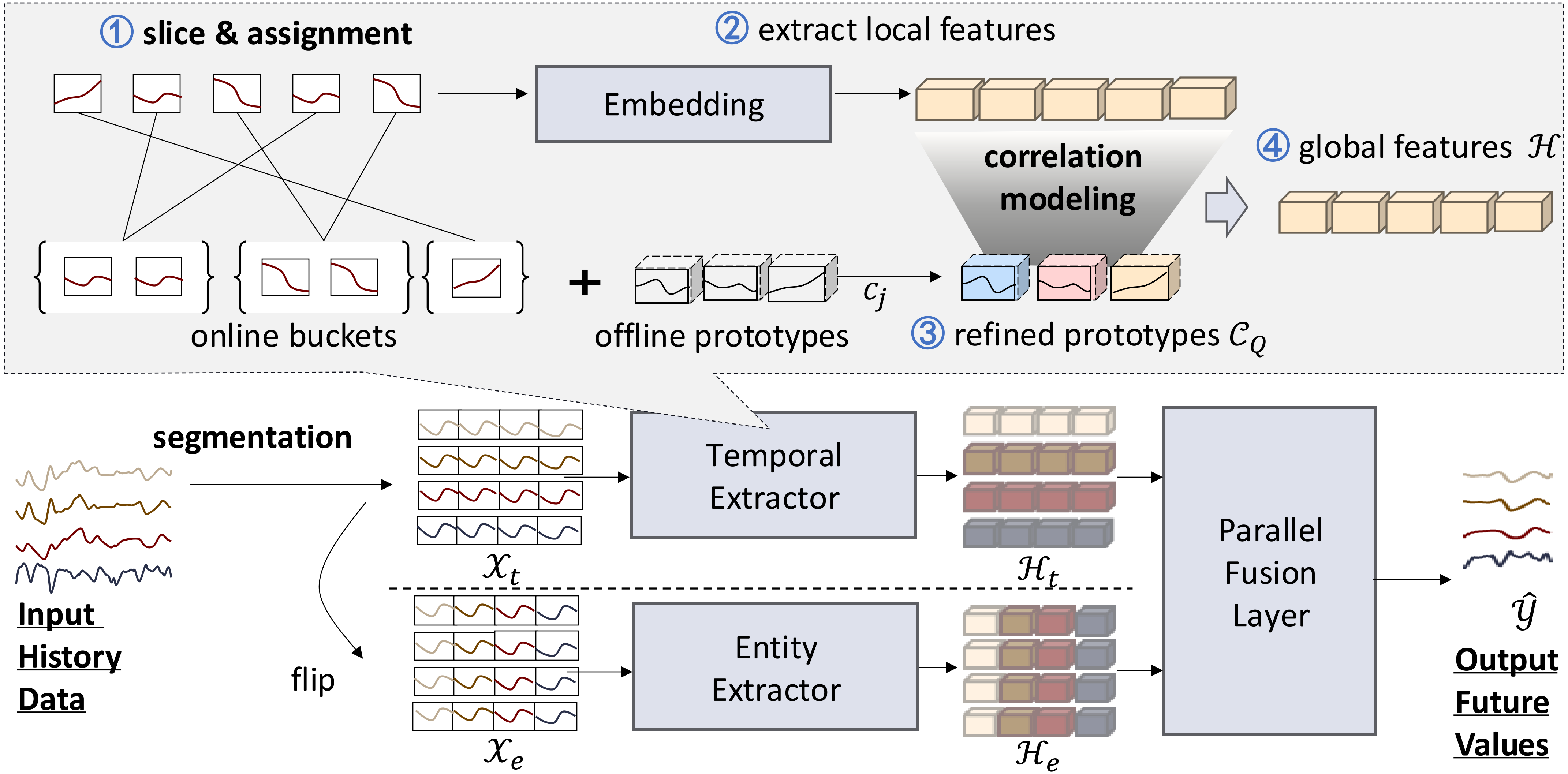
$$\mathcal{L} = \mathcal{L}_{rec} + \alpha L_{corr}$$

Numerical Reconstruction Loss: $\mathcal{L}_{rec} = \sum_{j=1}^{k} \left\| c_j - \mathrm{mean}(\boldsymbol{\mathcal{B}}_j) \right\|$

Correlation Matching Loss: $\mathcal{L}_{corr} = -\sum_{j=1}^{k} \frac{1}{|\boldsymbol{\mathcal{B}}_j|} \sum_{\mathcal{P}_{e,i} \in \boldsymbol{\mathcal{B}}_j} \mathrm{corr}\left(\mathcal{P}_{e,i}, c_j\right)$

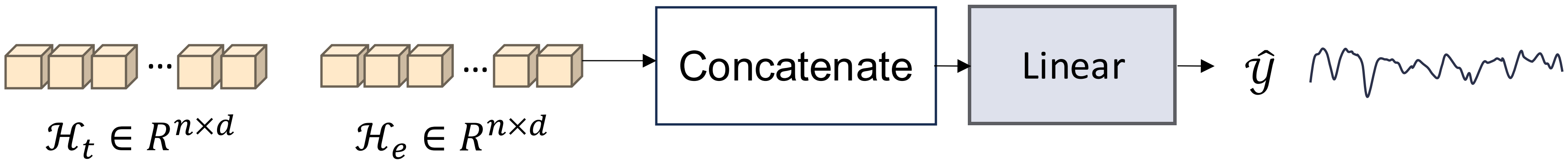
Our Solutions

- **Accelerated correlation modeling with prototypes**



Our Solutions

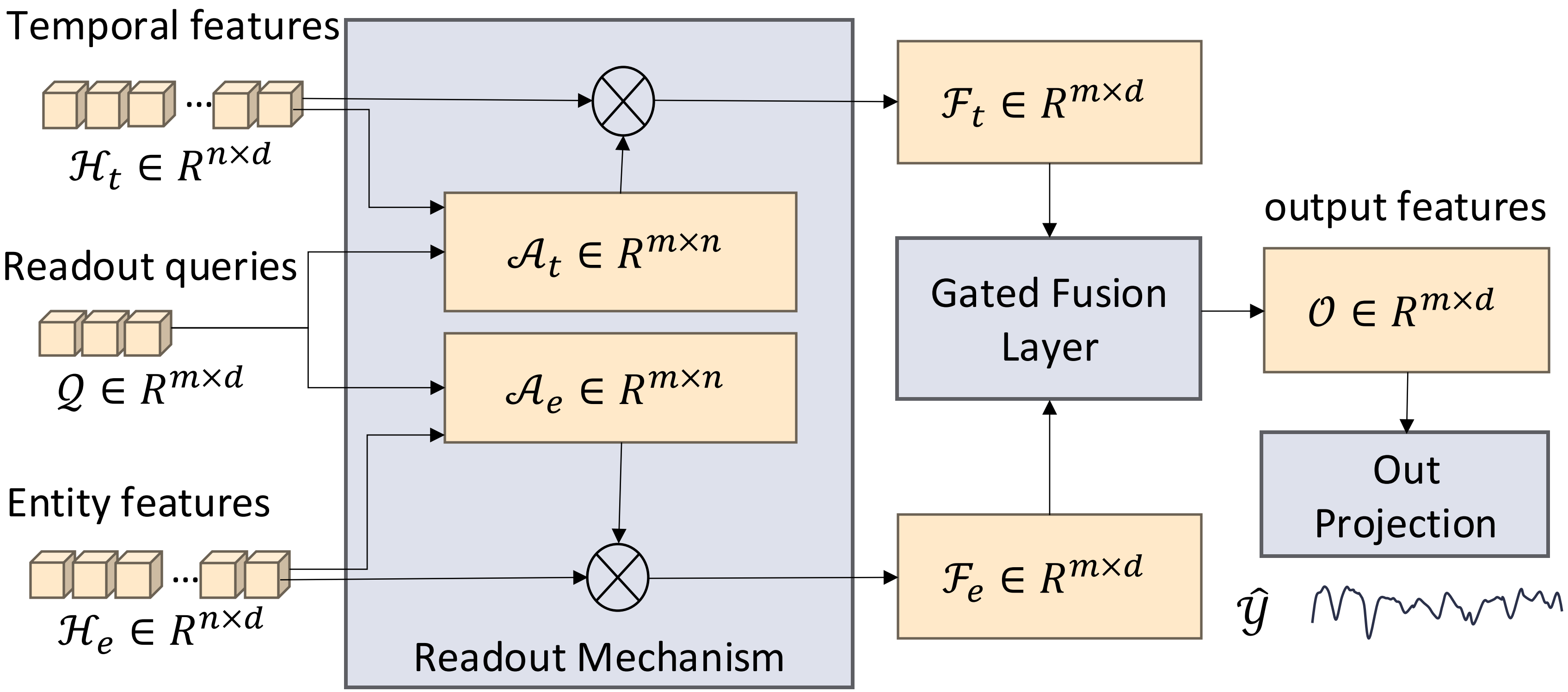
- **Parameter-efficient feature fusion with arbitrary input length**



Params: $2 * n * d * L_f$

Linear Layer

Fixed input length



$m \ll n$

Params: $2 * m + m * d * L_f$

Parallel Fusion Layer

Arbitrary input length

Outline

- Background and Motivations
- Problem Statement
- Our Solutions
- Experiments
- Conclusion

Experiments

- ## Datasets for evaluation

Input & Output: 512 timesteps as input and 96 or 336 timesteps as output

① **PEMS04**: contains 16,992 timesteps of traffic flow from 307 locations.

② **PEMS08**: contains 17,856 timesteps of traffic flow from 170 locations.

③ **ETTh1**: contains 14,400 timesteps of 7 indicators collected from electricity transformers every hour.

④ **ETTm1**: contains 57,600 timesteps of 7 indicators collected from electricity transformers every 15 minutes.

⑤ **Weather**: contains 52,696 timesteps recording 21 meteorological indicators every 10 minutes.

⑥ **Electricity**: contains 26,304 timesteps recording the hourly electricity consumption of 321.

⑦ **Traffic**: contains 17,544 timesteps recording hourly data from California Department of Transportation, which describes the road occupancy rates measured by 862 sensors on San Francisco Bay area freeways.

Experiments

- ## Baselines

Environment: Using PyTorch implemented on NVIDIA Tesla V100 GPU

① **FOCUS**: Our proposed method.

② **PatchTST**: A univariate forecasting model based on Transformer.

③ **Crossformer**: A sophisticated multivariate forecasting model based on Transformer.

④ **MTGNN**:A widely adopted multivariate forecasting model based on GCN.

⑤ **Graph Wavenet**: An efficient multivariate forecasting model based on Adaptive GCN.

⑥ **TimesNet**: An efficient multivariate forecasting model with Temporal 2D-Variation modeling.

⑦ **LightCTS**: A recent model that employs a refined structural design for efficient forecasting.

⑧ **Dlinear**: A simple forecasting model with only linear layers.

Experiments

- ## Evaluation of forecasting accuracy

FOCUS achieves **most accurate in 26 out of total 28 test settings**, gain 8.43% average improvement over PatchTST and 19.5% average improvement over LightCTS.

TABLE III: Comparison of long-range forecasting accuracy with baselines

Models		FOCUS		PatchTST		Crossformer		MTGNN		Graph Wavenet		TimesNet		LightCTS		DLinear	
Metrics		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
PEMS04	96	**0.0758**	**0.170**	0.102	0.228	0.103	0.209	0.0835	0.189	0.0838	0.186	0.0950	0.203	0.115	0.229	0.129	0.219
	336	**0.0936**	**0.190**	0.120	0.240	0.125	0.234	0.108	0.218	0.105	0.209	0.101	0.208	0.123	0.230	0.161	0.245
PEMS08	96	**0.0504**	**0.139**	0.0775	0.200	0.0773	0.190	0.0581	0.158	0.0615	0.157	0.66	0.172	0.0835	0.196	0.115	0.207
	336	**0.066**	**0.161**	0.0833	0.201	0.0929	0.207	0.0757	0.184	0.0793	0.183	0.07	0.177	0.0900	0.200	0.140	0.233
ETTh1	96	**0.372**	**0.402**	0.391	0.422	0.418	0.449	0.454	0.472	0.458	0.472	0.428	0.452	0.401	0.429	0.401	0.424
	336	**0.391**	**0.423**	0.434	0.463	0.751	0.681	0.457	0.474	0.543	0.531	0.454	0.425	0.506	0.495	0.424	0.446
ETTm1	96	0.304	**0.352**	**0.297**	0.354	0.324	0.370	0.366	0.427	0.358	0.402	0.316	0.369	0.312	0.363	0.307	0.358
	336	**0.366**	**0.394**	0.368	0.395	0.418	0.442	0.523	0.525	0.449	0.459	0.381	0.410	0.392	0.417	0.371	0.399
Traffic	96	**0.387**	**0.275**	0.388	0.299	0.520	0.287	0.492	0.291	0.523	0.292	0.623	0.337	0.596	0.415	0.395	0.276
	336	**0.414**	**0.285**	0.418	0.313	0.525	0.286	0.535	0.321	0.570	0.325	0.652	0.374	0.623	0.383	0.421	0.331
Electricity	96	**0.132**	**0.227**	0.155	0.274	0.136	0.236	0.137	0.238	0.146	0.247	0.190	0.296	0.177	0.279	0.135	0.233
	336	**0.164**	**0.259**	0.192	0.305	0.166	0.266	0.176	0.281	0.191	0.292	0.206	0.309	0.208	0.307	0.166	0.267
Weather	96	**0.148**	**0.199**	0.152	0.205	0.154	0.220	0.1613	0.227	0.161	0.221	0.170	0.229	0.149	0.203	0.155	0.223
	336	0.236	**0.280**	0.240	0.284	0.264	0.333	0.260	0.330	0.259	0.323	0.260	0.305	**0.234**	0.281	0.239	0.302

Experiments

- **Evaluation of forecasting efficiency**

FOCUS achieves **lowest FLOPS** (except DLinear) and **20x less memory usage** than PatchTST

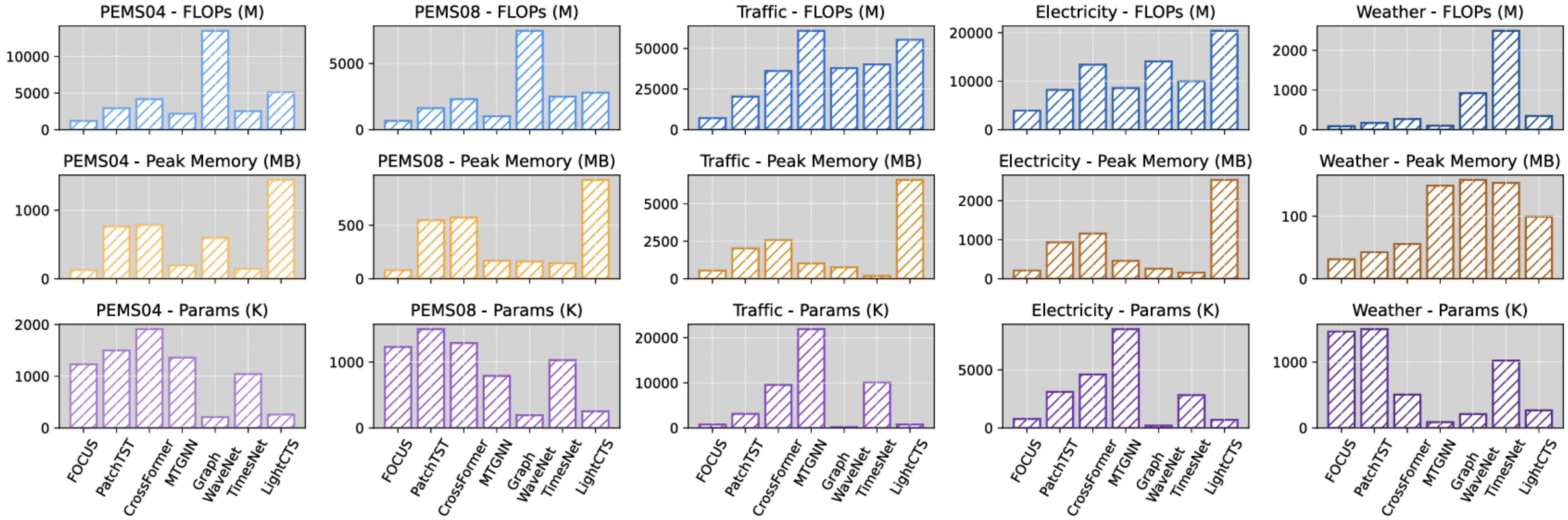


Fig. 6: Comparison of FLOPs, Peak Memory Occupation, Number of Parameters with baselines.

Experiments

• Ablation Study

In the comparison between FOCUS and the following three variants, the effectiveness of two-stage correlation modeling and the Parallel Fusion Layer is demonstrated:

- FOCUS-Attn: Directly replaces the two-stage association modeling with a full attention mechanism.
- FOCUS-LnrFusion: Directly replaces the Parallel Fusion Layer with a full linear layer.
- FOCUS-AllLnr: Replaces all modules with a full linear layer network (with nonlinear activation functions).

TABLE IV: Ablation Study

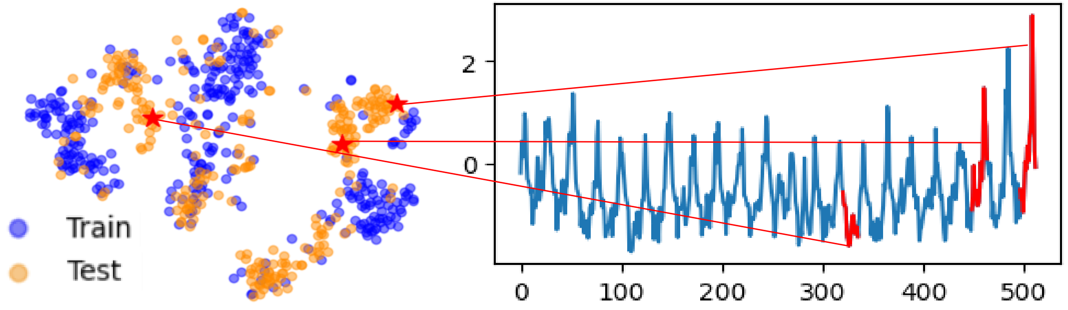
Dataset	Model	MSE	MAE	FLOPs(M)	Mem(MB)	Param(K)
PEMS08	FOCUS	**0.0711**	**0.168**	673	79.23	1227
	FOCUS-Attn	0.0864	0.178	1235	96.13	1233
	FOCUS-LnrFusion	0.0875	0.191	559	54.93	1438
	FOCUS-AllLnr	0.0897	0.193	**519**	**50.96**	1429
Electricity	FOCUS	**0.162**	**0.258**	3929	266	2617
	FOCUS-Attn	0.163	0.258	4434	305	2650
	FOCUS-LnrFusion	0.167	0.264	3134	177	2989
	FOCUS-AllLnr	0.173	0.281	**2966**	**168**	2956

Experiments

- ## Visualization

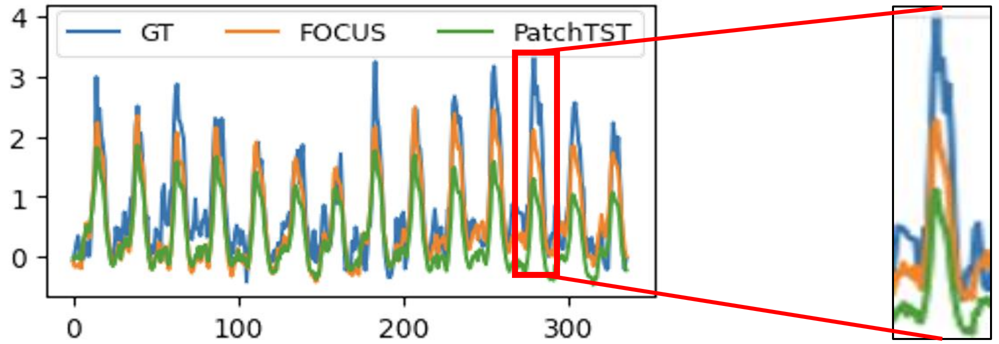
We selected the Traffic dataset for testing and chose the statistical results of the first variable.

Unseen Segment Patterns



We extracted the time segment representations constructed by FOCUS during the prediction process and visualized them via t-SNE.
The results show that there are some patterns in the test set that do not appear in the training set, and FOCUS can well distinguish these unseen patterns in the representation space.

Sequence-level Distribution Shift



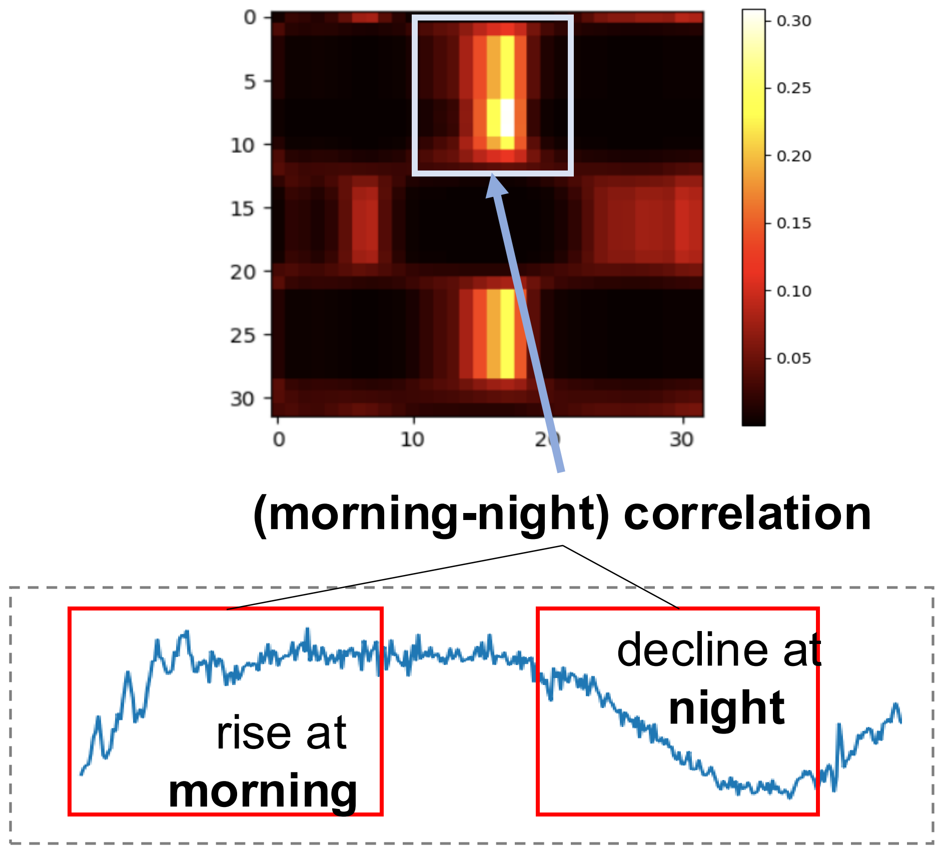
We selected a sequence from the Traffic test set, which has a significantly higher peak than those in the training set. We found that FOCUS tends to predict results with higher peak values compared to PatchTST. This indicates that FOCUS has stronger generalization ability in the face of such distribution shifts.

Experiments

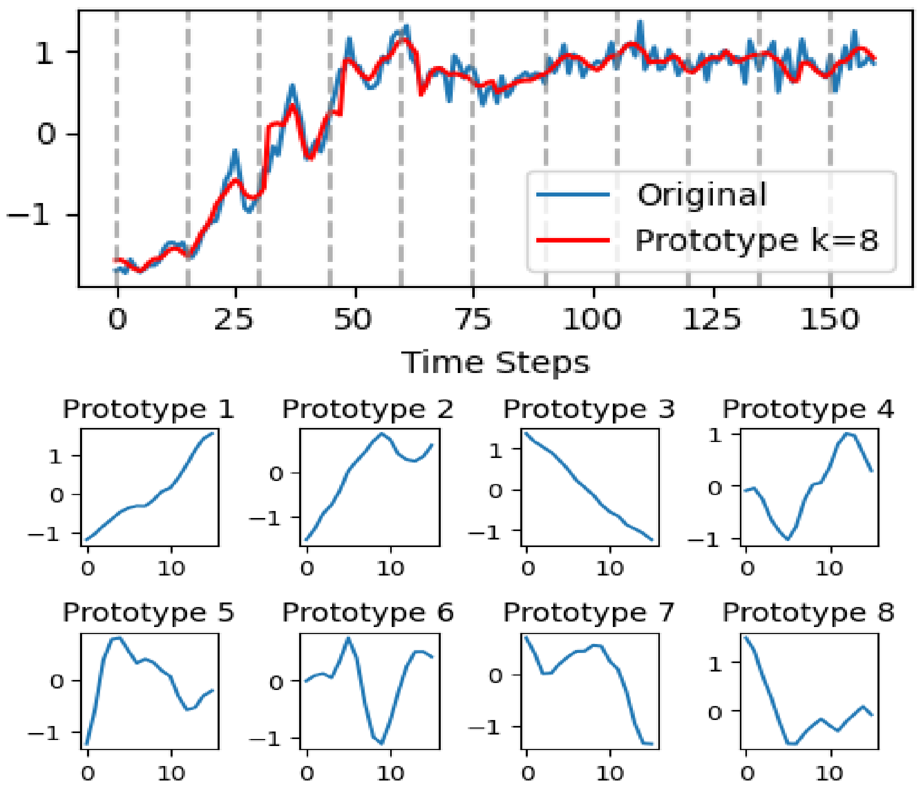
- ## Case Study

We randomly selected 512 timesteps from PEMS08 dataset for case study.

Correlation Matrix



Mined Prototypes



Outline

- Background and Motivations
- Problem Statement
- Our Solutions
- Experiments
- Conclusion

Conclusion

- **We propose FOCUS, a novel forecasting model that leverages the segment patterns of time series to accelerate attention-based modeling of long-range dependencies in time series.**

- **We design an efficient two-phase framework, including an offline phase for mining segment patterns and an online phase for achieving linear-complexity forecasting.**

- **Experiments on multi-domain real-world datasets demonstrate that FOCUS performs well in both efficiency and accuracy.**

Q & A

Thank You