Are Self-Attentions Effective for Time Series Forecasting?

Dongbin Kim (dongbin413@snu.ac.kr)

NeurIPS 2024.

Dongbin Kim, Jinseong Park, Jaewook Lee*, Hoki Kim*.

Are Self-Attentions Effective for Time Series Forecasting?. NeurIPS 2024

https://github.com/dongbeank/CATS

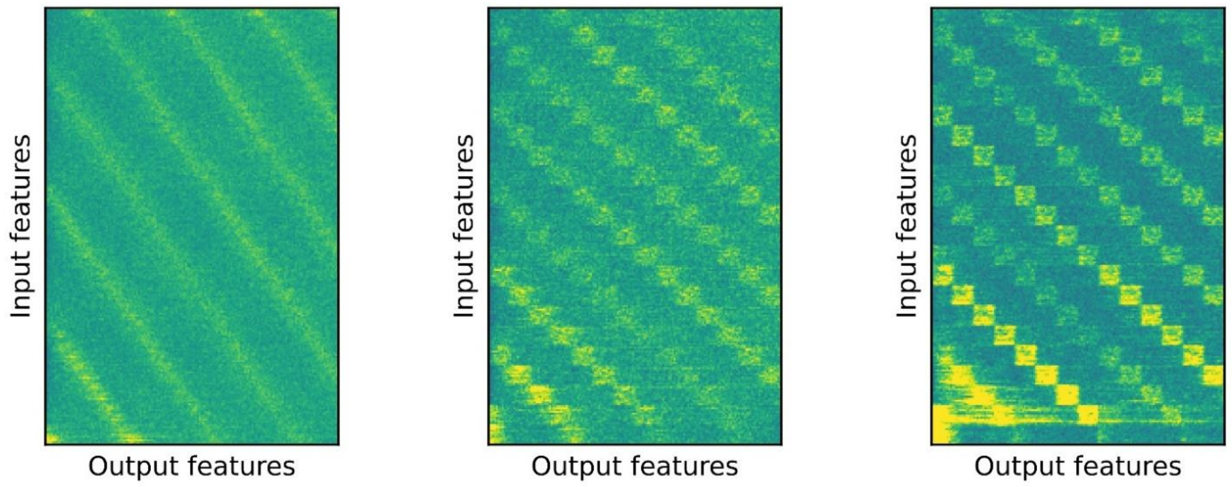
1 Introduction

NEURAL INFORMATION PROCESSING SYSTEMS

- Time series forecasting is crucial across domains, but **Transformer's effectiveness remains debated.**
- While recent works question the effectiveness of Transformers, with simpler linear models often outperforming them, we argue that **the core issue may lie in self-attention.**
- The success of linear models suggests **we need to rethink which components are truly necessary.**
- **Key question: Are self-attentions truly effective for time series forecasting?**

Motivation of Self-Attention Removal

- Conducted experiments with three PatchTST variations (overlapping, non-overlapping, w/o self-attention).
- Weight patterns in final linear layer reveal clearer temporal capture without self-attention.
- Performance analysis suggests **self-attention might be unnecessary for time series forecasting.**



(a) Original PatchTST (b) PatchTST w/ non-overlapping (c) PatchTST w/o self-attn

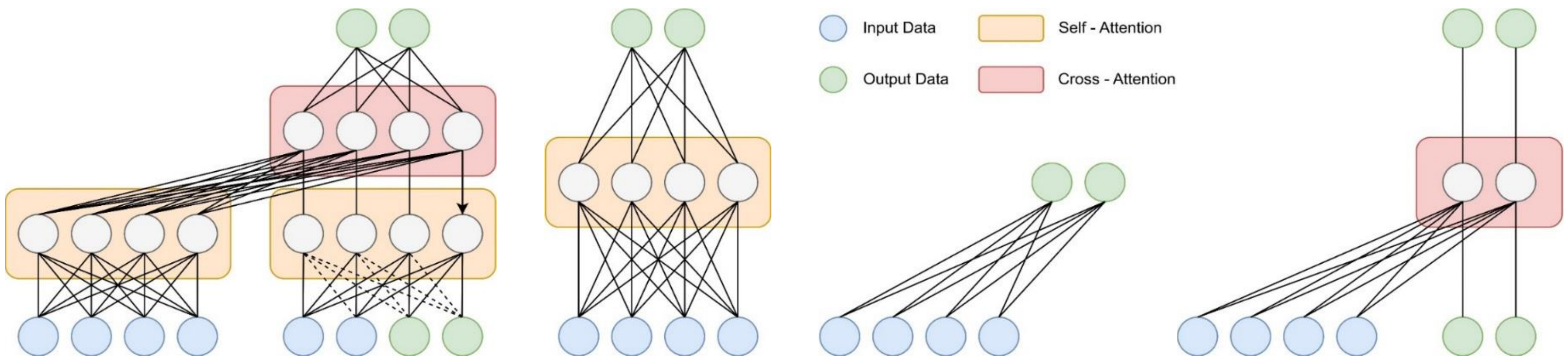
Effect of self-attention in PatchTST on forecasting performance

Horizon	original	w/o self-attn
96	**0.290**	**0.290**
192	0.332	**0.328**
336	0.366	**0.359**
720	0.416	**0.414**

Rethinking Transformer Design

NEURAL INFORMATION PROCESSING SYSTEMS

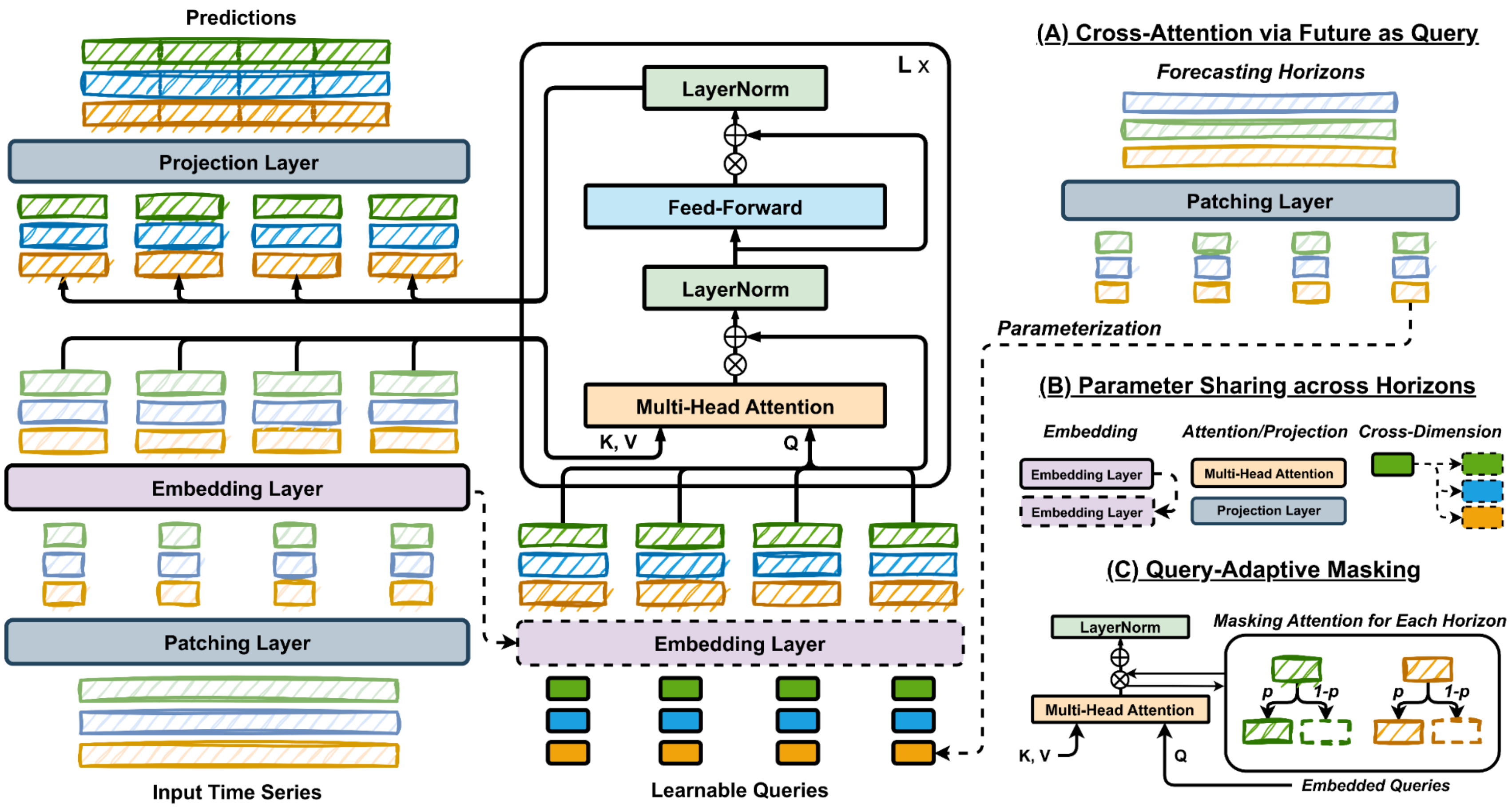
- Given the challenges associated with self-attention in time series forecasting, we propose a fundamental rethinking of the Transformer architecture:
  - Traditional Transformers rely heavily on self-attention mechanisms, potentially leading to temporal information loss.
  - Linear models remove all transformer-based components but may struggle with complex temporal dependencies.
  - Our approach removes self-attention layers while preserving the advantages of cross-attention.



(a) Transformer **(b) Encoder** **(c) Linear model** **(d) Ours**

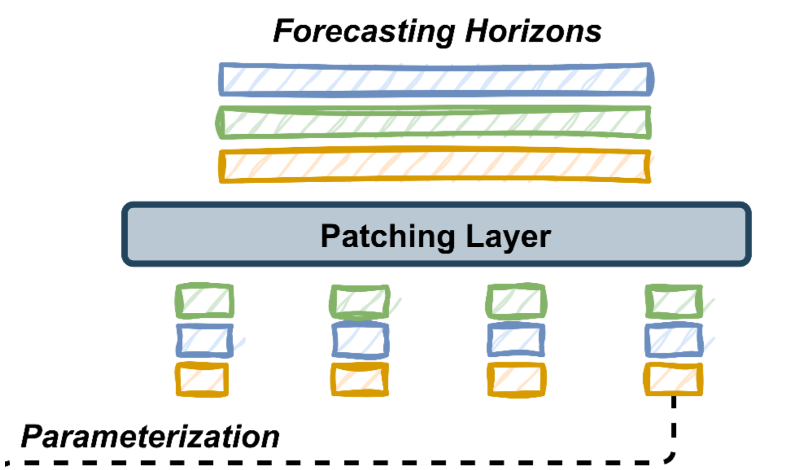
Proposed Methodology

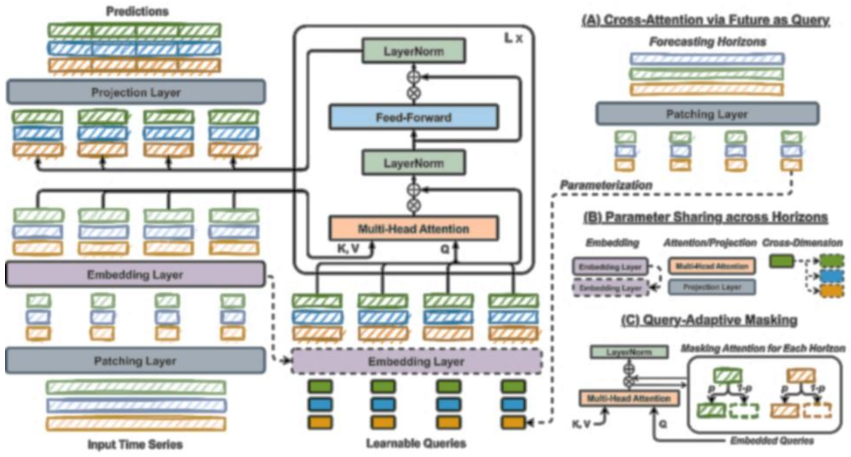
- We introduce a Cross-Attention-only Time Series transformer (CATS), that rethinks the traditional Transformer framework by eliminating self-attention and leveraging cross-attention mechanisms instead.



Proposed Methodology

NEURAL INFORMATION PROCESSING SYSTEMS





(A) Cross-Attention via Future as Query

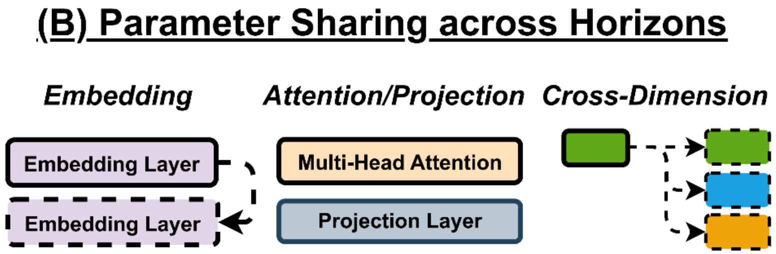
- Novel query conceptualization
  - Future horizons as learnable parameters
- Direct temporal pattern capture without information loss
- Time and memory complexity reduced to $O(LT/P^2)$ from $O(L^2)$

Table 2: Time complexity of transformer-based models to calculate attention outputs. Time refers to the inference time obtained by averaging 10 runs under $L = 96$ and $T = 720$ on Electricity.

Method	Encoder	Decoder	Time	Method	Encoder	Decoder	Time
Transformer [13]	$\mathcal{O}(L^2)$	$\mathcal{O}(T(T+L))$	10.4ms	Informer [29]	$\mathcal{O}(L\log L)$	$\mathcal{O}(T(T+\log L))$	13.5ms
Autoformer [23]	$\mathcal{O}(L\log L)$	$\mathcal{O}\left((L/2+H)\log\left(L/2+T\right)\right)$	24.1ms	Pyraformer [11]	$\mathcal{O}(L)$	$\mathcal{O}(T(T+L))$	11.2ms
FEDformer [31]	$\mathcal{O}(L)$	$\mathcal{O}\left(L/2+H\right)$	69.3ms	Crossformer [28]	$\mathcal{O}\left(ML^2/P^2\right)$	$\mathcal{O}\left(MT(T+L)/P^2\right)$	30.6ms
PatchTST [14]	$\mathcal{O}\left(L^2/P^2\right)$	-	7.6ms	CATS (Ours)	-	$\mathcal{O}\left(LT/P^2\right)$	7.0ms

Proposed Methodology

(B) Parameter Sharing across Horizons



- Comprehensive sharing across all network layers
  - Embedding Layer, Attention blocks, Projection Layer
- Cross-dimension parameter sharing
- Significant reduction in model parameters while maintaining performance

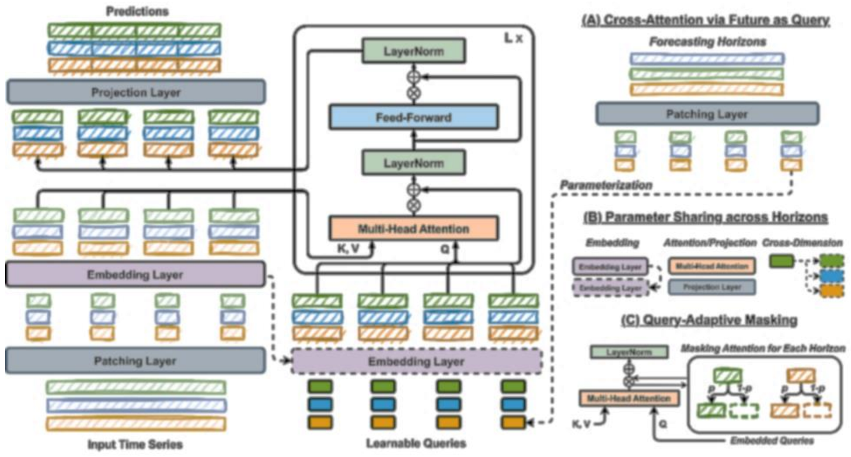


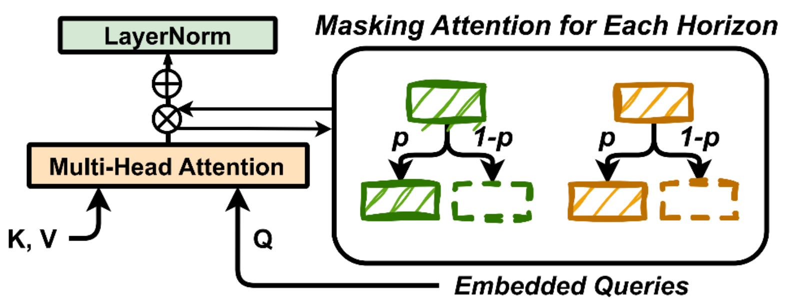
Table 3: Effect of parameter sharing across horizons on the number of parameters for different forecasting horizons on ETTh1.

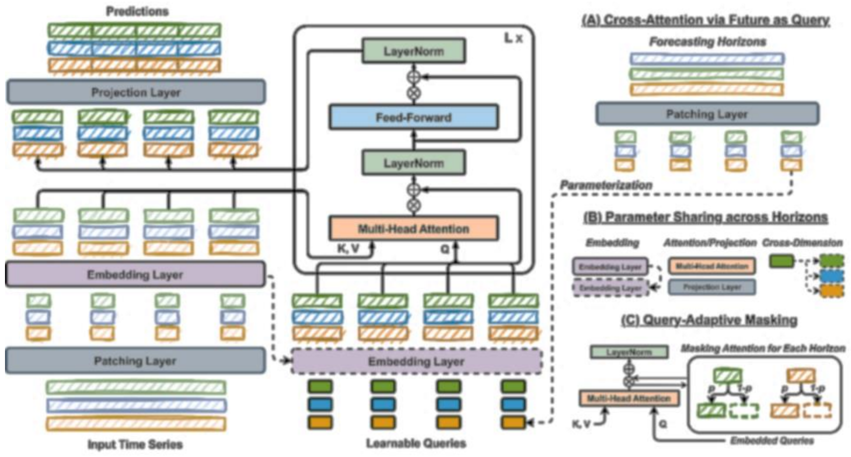
Horizon	w/ sharing	w/o sharing
96	355,320	404,672
192	355,416	552,320
336	355,560	958,112
720	355,944	3,121,568

Proposed Methodology

NEURAL INFORMATION PROCESSING SYSTEMS

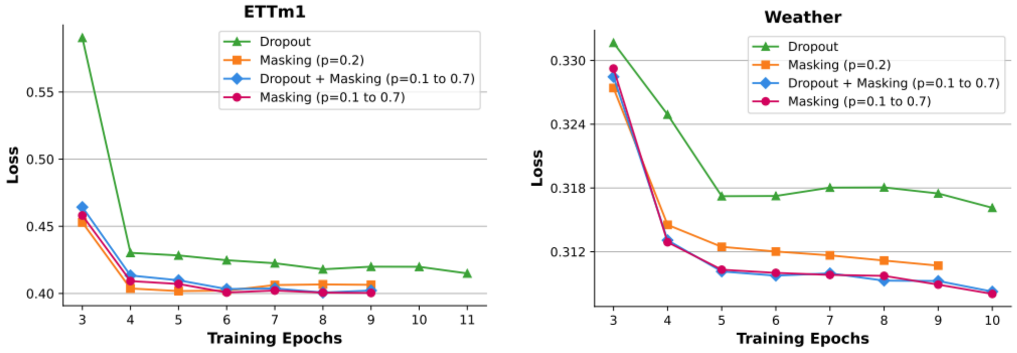
(C) Query-Adaptive Masking





(C) Query-Adaptive Masking

- Novel selective masking technique
- Prevention of overfitting to keys/values
- Enhanced focus on horizon-specific patterns through probabilistic masking



(a) ETTm1 with $T = 720$ (b) Weather with $T = 720$

Figure 8: Comparison of performance with query-adaptive masking with two different probabilities, dropout, and using both query-adaptive masking and dropout. The results of $p = 0.1$ to 0.7 indicate a probability masking that is linearly increased proportionally to the horizon predicted by the query.

5 Experimental Results

- Performance Comparison

Multivariate Long-term Time Series Forecasting Results

Models		CATS		TimeMixer		PatchTST		Timesnet		Crossformer		MICN		FiLM		DLinear		Autoformer		Informer	
Metric		MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
Weather	96	**0.161**	**0.207**	0.163	0.209	0.186	0.227	0.172	0.220	0.195	0.271	0.198	0.261	0.195	0.236	0.195	0.252	0.266	0.336	0.300	0.384
	192	**0.208**	**0.250**	**0.208**	**0.250**	0.234	0.265	0.219	0.261	0.209	0.277	0.239	0.299	0.239	0.271	0.237	0.295	0.307	0.367	0.598	0.544
	336	0.264	0.290	0.251	**0.287**	0.284	0.301	**0.246**	0.337	0.273	0.332	0.285	0.336	0.289	0.306	0.282	0.331	0.359	0.395	0.578	0.523
	720	0.342	**0.341**	**0.339**	**0.341**	0.356	0.349	0.365	0.359	0.379	0.401	0.351	0.388	0.361	0.351	0.345	0.382	0.419	0.428	1.059	0.741
Electricity	96	**0.149**	**0.237**	0.153	0.247	0.190	0.296	0.168	0.272	0.219	0.314	0.180	0.293	0.198	0.274	0.210	0.302	0.201	0.317	0.274	0.368
	192	**0.163**	**0.250**	0.166	0.256	0.199	0.304	0.184	0.322	0.231	0.322	0.189	0.302	0.198	0.278	0.210	0.305	0.222	0.334	0.296	0.386
	336	**0.180**	**0.268**	0.185	0.277	0.217	0.319	0.198	0.300	0.246	0.337	0.198	0.312	0.217	0.300	0.223	0.319	0.231	0.443	0.300	0.394
	720	0.219	**0.302**	0.225	0.310	0.258	0.352	0.220	0.320	0.280	0.363	**0.217**	0.330	0.278	0.356	0.258	0.350	0.254	0.361	0.373	0.439
Traffic	96	**0.421**	**0.270**	0.462	0.285	0.526	0.347	0.593	0.321	0.644	0.429	0.577	0.350	0.647	0.384	0.650	0.396	0.613	0.388	0.719	0.391
	192	**0.436**	**0.275**	0.473	0.296	0.522	0.332	0.617	0.336	0.665	0.431	0.589	0.356	0.600	0.361	0.598	0.370	0.616	0.382	0.696	0.379
	336	**0.453**	**0.284**	0.498	0.296	0.517	0.334	0.629	0.336	0.674	0.420	0.594	0.358	0.610	0.367	0.605	0.373	0.622	0.337	0.777	0.420
	720	**0.484**	**0.303**	0.506	0.313	0.552	0.352	0.640	0.350	0.683	0.424	0.613	0.361	0.691	0.425	0.645	0.394	0.660	0.408	0.864	0.472
ETT (Avg)	96	**0.289**	**0.339**	0.290	**0.339**	0.326	0.362	0.312	0.355	0.465	0.456	0.340	0.388	0.324	0.358	0.319	0.368	0.389	0.415	1.414	0.816
	192	**0.348**	0.374	0.350	**0.373**	0.388	0.397	0.365	0.385	0.553	0.518	0.408	0.431	0.384	0.393	0.399	0.418	0.448	0.443	1.985	0.989
	336	**0.376**	**0.395**	0.390	0.404	0.426	0.423	0.455	0.421	0.686	0.584	0.479	0.476	0.428	0.423	0.469	0.463	0.491	0.473	2.101	1.101
	720	**0.434**	**0.433**	0.439	0.438	0.464	0.455	0.467	0.455	1.038	0.754	0.597	0.541	0.481	0.459	0.596	0.537	0.533	0.504	2.343	1.163

Short-term Time Series Forecasting Results

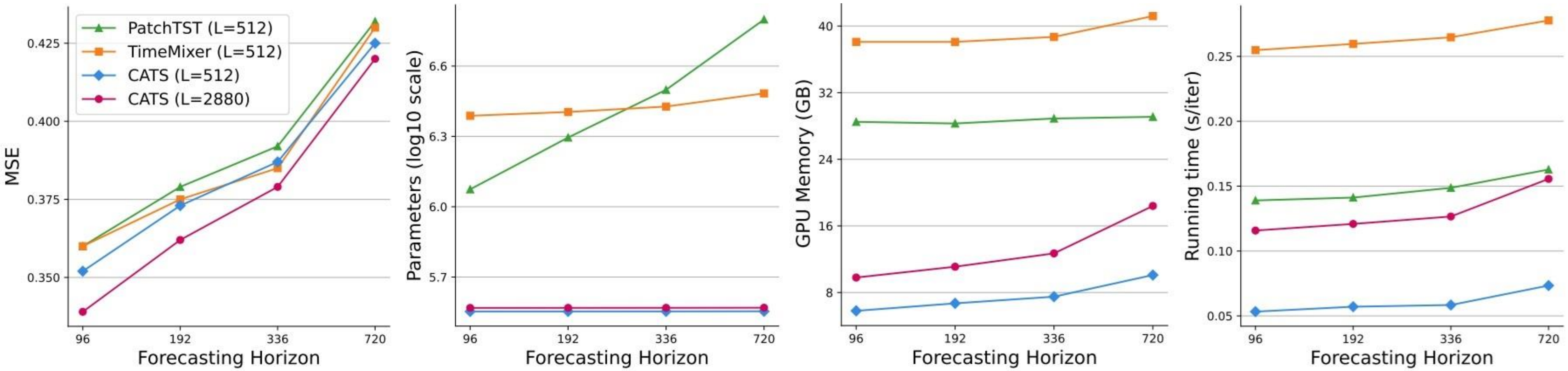
Models		CATS	TimeMixer	Timesnet	PatchTST	MICN	FiLM	DLinear	Autoformer	Informer
Average	SMAPE	**11.701**	11.723	11.829	13.152	19.638	14.863	13.639	12.909	14.086
	MASE	**1.557**	1.559	1.585	1.945	5.947	2.207	2.095	1.771	2.718
	OWA	**0.838**	0.840	0.851	0.998	2.279	1.125	1.051	0.939	1.230

- Efficiency Analysis

Model Scalability with Input Length

Input Length	Parameters 336	Parameters 720	Parameters 1440	Parameters 2880	GPU Memory 336	GPU Memory 720	GPU Memory 1440	GPU Memory 2880	MSE 336	MSE 720	MSE 1440	MSE 2880
PatchTST	4.3M	8.7M (2.0x)	17.0M (4.0x)	33.6M (7.9x)	3.5GB	7.4GB (2.1x)	22.0GB (6.3x)	58.6GB (16.9x)	0.418	0.418	0.420	0.412
TimeMixer	1.1M	4.1M (3.6x)	14.2M (12.6x)	52.9M (46.8x)	2.9GB	3.9GB (1.3x)	5.9GB (2.0x)	10.3GB (3.6x)	0.428	0.425	0.414	0.472
DLinear	0.5M	1.0M (2.1x)	2.1M (4.2x)	4.2M (8.5x)	1.1GB	1.1GB (1.0x)	1.2GB (1.0x)	1.2GB (1.1x)	0.426	0.422	0.401	0.408
CATS	0.4M	0.4M (1.0x)	0.4M (1.0x)	0.4M (1.1x)	1.9GB	2.1GB (1.1x)	2.7GB (1.4x)	3.8GB (2.0x)	0.407	0.402	0.399	0.395

Superior Efficiency with Longer Sequences



Conclusion

- We introduce **CATS**, a novel architecture that **simplifies the Transformer by eliminating all self-attentions** and focusing on cross-attention potential.

- We propose **three specialized techniques** tailored for cross-attention-only transformer: **(i) cross-attention via future as query**, **(ii) parameter sharing across horizons**, and **(iii) query-adaptive masking**.

- Our model achieves **state-of-the-art performance with significantly fewer parameters**, providing new insights into designing efficient architectures for time series forecasting.

- For more results and source code, please visit:

Paper

Code