

2025 11 20  
발표 자료

광운대학교 로봇학과  
FAIR Lab

김한서

# 이번 주 진행사항

- Triformer
  - 논문 리뷰
  - 실험 세팅
  - 실험 결과 및 시각화
  - 결과 정리

## **Triformer: Triangular, Variable-Specific Attentions for Long Sequence Multivariate Time Series Forecasting–Full Version**

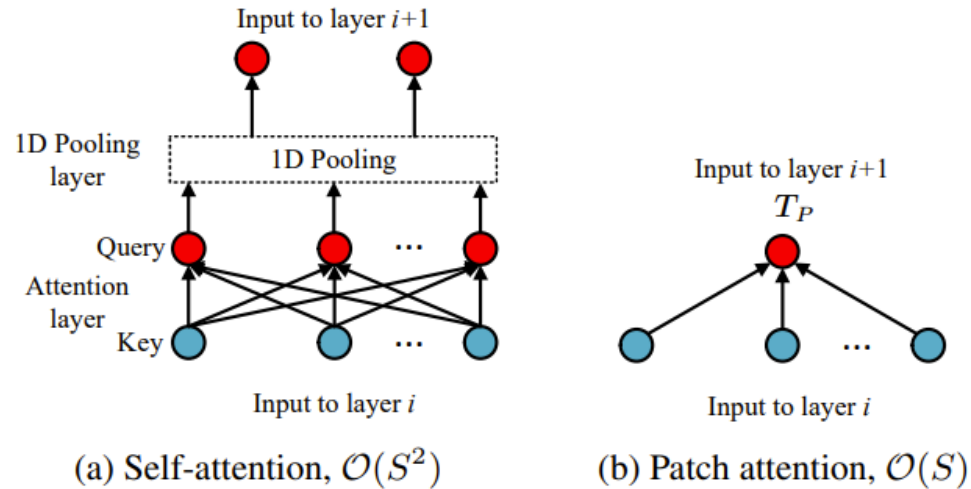
**Razvan-Gabriel Cirstea<sup>1</sup>, Chenjuan Guo<sup>1</sup>, Bin Yang<sup>1\*</sup>, Tung Kieu<sup>1</sup>, Xuanyi Dong<sup>2</sup>, Shirui Pan<sup>3</sup>**

<sup>1</sup>Aalborg University <sup>2</sup>Google Research, Brain Team <sup>3</sup>Monash University

{razvan, cguo, byang, tungkvt}@cs.aau.dk, xuanyi.dxy@gmail.com, shirui.pan@monash.edu

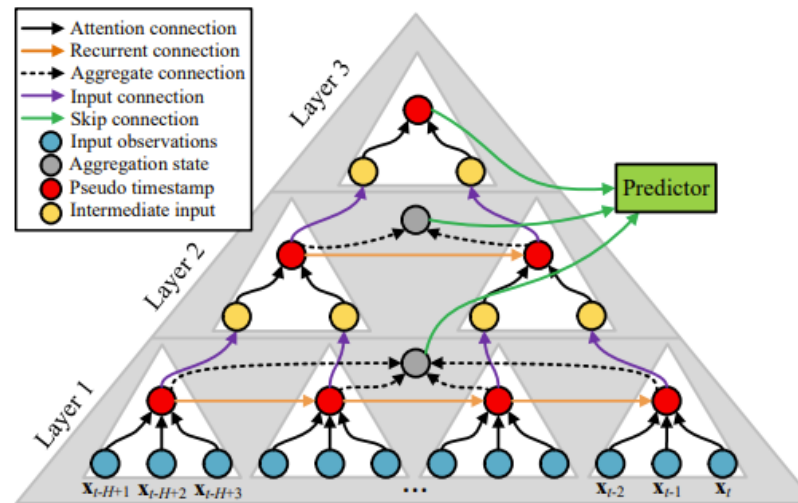
- arXiv 등록일: 2023-04-28
- 인용 수: 209회(Google Scholar, 2025-11-17)
- Published at IJCAI 2022
- Triformer는 기존 Transformer의 계산 복잡도와 변수별 고유 패턴 파악 실패를 해결하기 위해 Patch attention기반의 삼각형 구조와 경량 변수별 모델링을 제안

## Patch Attention



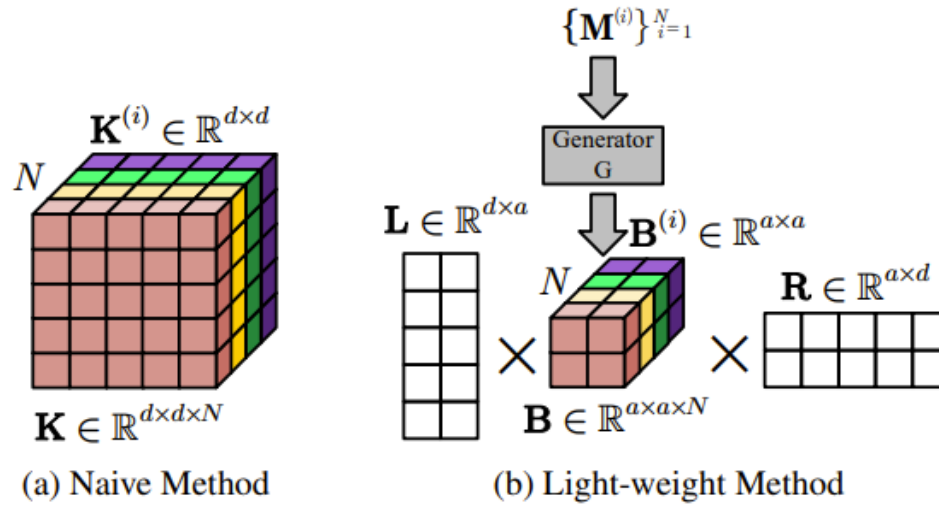
- (a) Self-attention
  - 모든 Query가 모든 Key를 참조하게 되면 계산 복잡도가 제곱에 비례하는 문제 발생
- (b) Patch Attention
  - 하나의 Query가 모든 Key를 참조하게 되면 계산 복잡도가 제곱이 아닌 입력의 개수에 정비례하여 계산량이 엄청나게 줄어들게 됨
- 시계열을 Patch 단위로 자르고, Linear Projection을 통해 고차원 임베딩 벡터로 변환 후에 입력으로 사용

## 모델 구조



- Layer 1
  - Patch attention에서 나온 요약된 벡터들을 순환 연결을 통해 지역적인 시계열 패턴 학습
- Layer 2, 3
  - 하위 층의 요약된 벡터들을 입력으로 받아 Patch attention 수행, 이후 나온 벡터들을 순환 연결을 통해 넓은 범위의 전역적인 추세 학습
- Predictor
  - 각 층의 벡터들의 정보들을 결합하여 Predictor로 전달
  - Predictor는 전달받은 모든 정보를 종합해 최종 예측 값 도출

## 경량 변수별 모델링



- Naive Method
  - 변수별 특성을 반영하기 위해 변수 각각에 고유한 행렬 생성
  - 변수 개수에 비례해 파라미터가 급증, 메모리 부족 발생
- Light-weight Method
  - Factorization(인수분해)을 통해 행렬을 변수 무관 행렬( $L, R$ )과 변수별 행렬  $B$ 로 분리  
모든 변수가  $L, R$ 을 공유하여 사용,  $B$ 는 Naive 행렬 크기보다 더 작은 단위를 사용
  - Generator (생성기)를 통해 변수별  $B$  부분을 직접 학습하지 않고, 메모리 벡터와 생성기를 통해 생성

# Triformer

## 주요 모델 성능 비교

Method		Reformer		LogTrans		StemGNN		AGCRN		Informer		Autoformer		Triformer	
	$F$	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
$ETT_{h1}$	24	0.991	0.754	0.686	0.604	0.488	0.508	0.438	0.461	0.577	0.549	<u>0.399</u>	<u>0.429</u>	<b>0.328</b>	<b>0.380</b>
	48	1.313	0.906	0.766	0.757	0.473	0.505	0.488	0.486	0.685	0.625	<u>0.405</u>	<u>0.439</u>	<b>0.359</b>	<b>0.401</b>
	168	1.824	1.138	1.002	0.846	0.630	0.577	0.553	0.540	0.931	0.752	<u>0.498</u>	<u>0.486</u>	<b>0.433</b>	<b>0.449</b>
	336	2.117	1.280	1.362	0.952	0.658	0.598	0.611	0.566	1.128	0.873	<u>0.500</u>	<u>0.480</u>	<b>0.487</b>	<b>0.475</b>
	720	2.415	1.520	1.397	1.291	0.764	0.675	0.778	0.643	1.215	0.896	<u>0.498</u>	<u>0.500</u>	<b>0.488</b>	<b>0.475</b>
$ETT_{m1}$	24	0.724	0.607	0.419	0.412	<u>0.302</u>	0.381	0.318	0.358	0.323	<u>0.369</u>	0.377	0.417	<b>0.252</b>	<b>0.329</b>
	48	1.098	0.777	0.507	0.583	<u>0.382</u>	<u>0.436</u>	0.426	0.429	0.494	<u>0.503</u>	0.429	0.442	<b>0.275</b>	<b>0.337</b>
	96	1.433	0.945	0.768	0.792	<u>0.419</u>	<u>0.461</u>	0.409	0.433	0.678	0.614	0.458	0.460	<b>0.314</b>	<b>0.371</b>
	288	1.820	1.094	1.462	1.320	<u>0.522</u>	<u>0.518</u>	0.545	0.531	1.056	0.786	0.632	0.526	<b>0.385</b>	<b>0.427</b>
	672	2.187	1.232	1.669	1.461	0.644	0.590	0.567	0.570	1.192	0.926	<u>0.602</u>	<u>0.540</u>	<b>0.437</b>	<b>0.448</b>
Weather	24	0.655	0.583	0.435	0.477	0.377	0.416	0.947	0.774	<u>0.335</u>	<u>0.381</u>	0.408	0.445	<b>0.323</b>	<b>0.364</b>
	48	0.729	0.666	0.426	0.495	0.438	0.467	0.948	0.777	<u>0.395</u>	<u>0.459</u>	0.475	0.487	<b>0.390</b>	<b>0.429</b>
	168	1.318	0.855	0.727	0.671	<u>0.554</u>	<u>0.545</u>	0.950	0.774	<u>0.608</u>	<u>0.567</u>	0.576	0.550	<b>0.497</b>	<b>0.501</b>
	336	1.930	1.167	0.754	0.670	<u>0.598</u>	<u>0.573</u>	0.946	0.772	0.702	0.620	<u>0.593</u>	<u>0.558</u>	<b>0.538</b>	<b>0.531</b>
	720	2.726	1.575	0.885	0.773	<u>0.676</u>	<u>0.619</u>	0.938	0.770	0.831	0.731	<u>0.682</u>	<u>0.634</u>	<b>0.587</b>	<b>0.563</b>
ECL	48	1.404	0.999	0.355	0.418	0.209	0.311	0.385	0.398	0.344	0.393	<u>0.193</u>	<u>0.310</u>	<b>0.183</b>	<b>0.279</b>
	168	1.515	1.069	0.368	0.432	0.265	0.355	0.481	0.509	0.368	0.424	<u>0.223</u>	<u>0.334</u>	<b>0.182</b>	<b>0.288</b>
	336	1.601	1.104	0.373	0.439	0.291	0.380	0.564	0.561	0.381	0.431	<u>0.232</u>	<u>0.345</u>	<b>0.202</b>	<b>0.309</b>
	720	2.009	1.170	0.409	0.454	0.317	0.400	0.725	0.678	0.406	0.443	<u>0.257</u>	<u>0.361</u>	<b>0.251</b>	<b>0.335</b>
	960	2.141	1.387	0.477	0.589	0.329	0.411	1.005	0.829	0.460	0.548	<u>0.265</u>	<u>0.364</u>	<b>0.248</b>	<b>0.339</b>

Table 2: Overall accuracy. **Bold** highlights the best results. Underline highlights the second best results.

- Triformer 모델은 Informer나 Autoformer와 같은 기존 모델 대비, 대부분의 데이터셋과 모든 예측 길이에서 가장 뛰어난 예측 성능을 보여줌

## 실험 세팅

- 사용한 모델: Triformer
- 재현 실험 데이터셋: ETTh1

Experiment	ETTh1
Learning rate	$10^{-4}$
Epoch	6
Batch size	32
Loss function	MSE Loss
Seq_len	96
input_feature	7
Pred_len	24/48/168/336/720
d_model	64
d_ff	2048

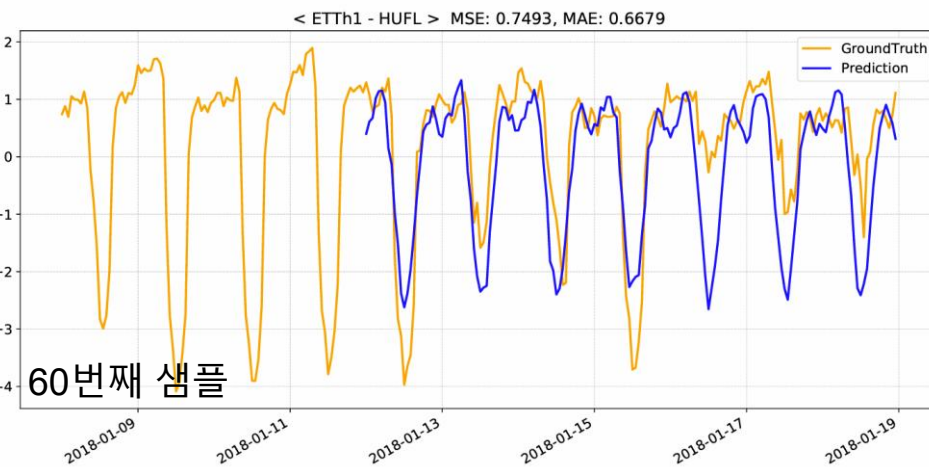
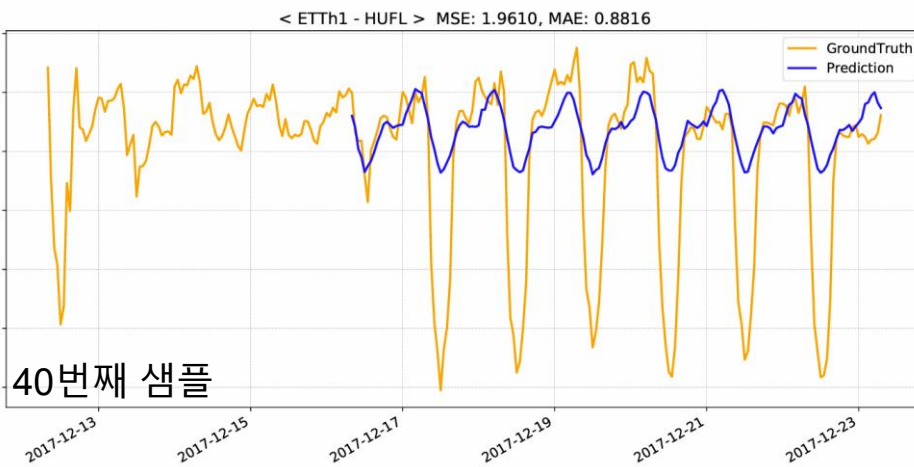
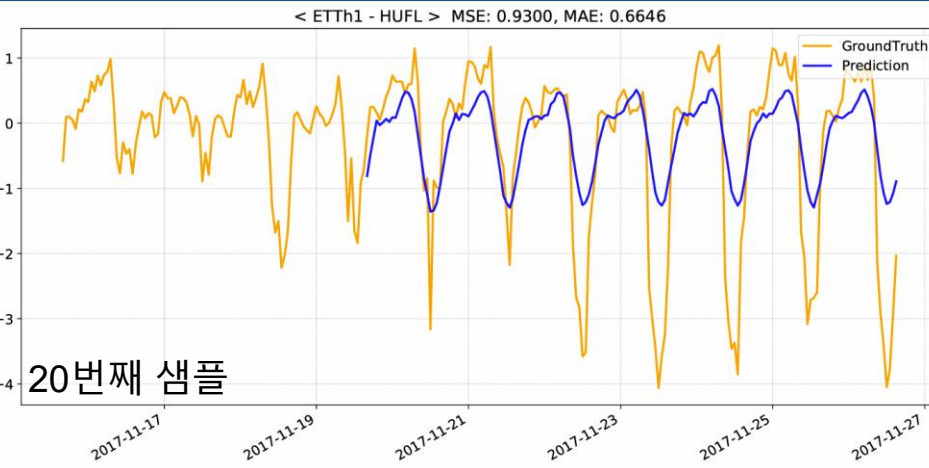
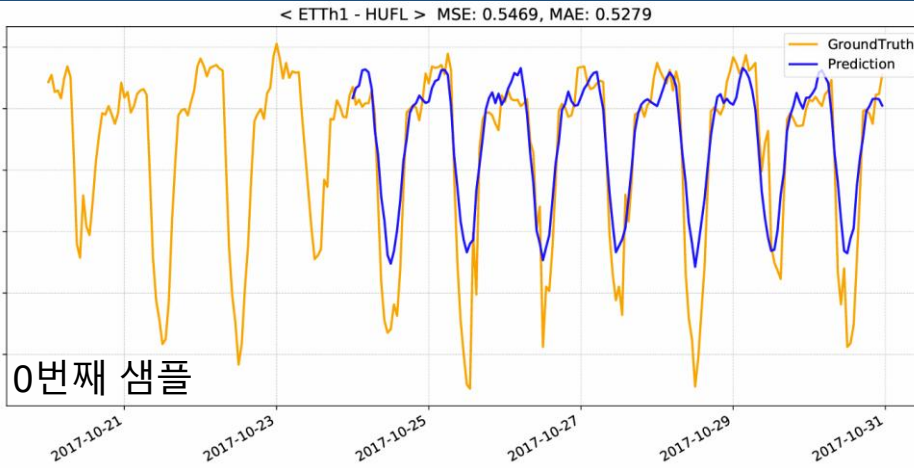


## 재현 실험 (ETTh1)

- 단기 예측 길이에선 논문 결과와 비슷하게 나왔지만, 장기 예측 길이에선 오차가 크게 발생하였음

	ETTh1 Paper		ETTh1 Reproduction	
Pred len	MSE	MAE	MSE	MAE
24	0.328	0.380	0.319	0.375
48	0.359	0.401	0.353	0.396
168	0.433	0.449	0.434	0.444
336	0.487	0.475	0.504	0.487
720	0.488	0.475	0.552	0.534

## 재현 실험 시각화 (ETTh1)



Seq\_len → 96  
Pred\_len → 168

## Triformer, Crossformer 비교 실험 (ETTh1)

- 비교적 최신 모델인 Crossformer가 대부분의 예측 길이에서 더 뛰어난 성능을 보였지만, 예측 길이 720에서는 Triformer가 조금 더 좋은 성능을 보임

	Triformer ETTh1		Crossformer ETTh1	
Pred len	MSE	MAE	MSE	MAE
24	<u>0.319</u>	<u>0.375</u>	<b>0.302</b>	<b>0.359</b>
48	<u>0.353</u>	<u>0.396</u>	<b>0.346</b>	<b>0.387</b>
168	<u>0.434</u>	<u>0.444</u>	<b>0.406</b>	<b>0.441</b>
336	<u>0.504</u>	<u>0.487</u>	<b>0.449</b>	<b>0.471</b>
720	<b>0.552</b>	<b>0.534</b>	<u>0.556</u>	<u>0.542</u>

## 실험 결과 정리

- 재현 실험
  - 장기 예측 길이인 720을 제외한 나머지에서 논문과 비슷한 수치가 나온 것을 확인
- 시각화
  - 예측 값이 정답 값을 어느정도 따라가는 모습을 보였지만 순간적인 변동을 따라가지 못함
- 비교 실험
  - 대부분의 예측 길이에서 Crossformer의 성능이 더 뛰어난 것을 확인, 하지만 예측 길이 720에서는 Triformer의 삼각형 구조가 정보 손실을 줄여 Triformer의 성능이 더 좋게 나옴