Invited Talk at CS Dept. of Tsinghua University

Robust and Intelligent Time Series Analysis and Applications

Qingsong Wen

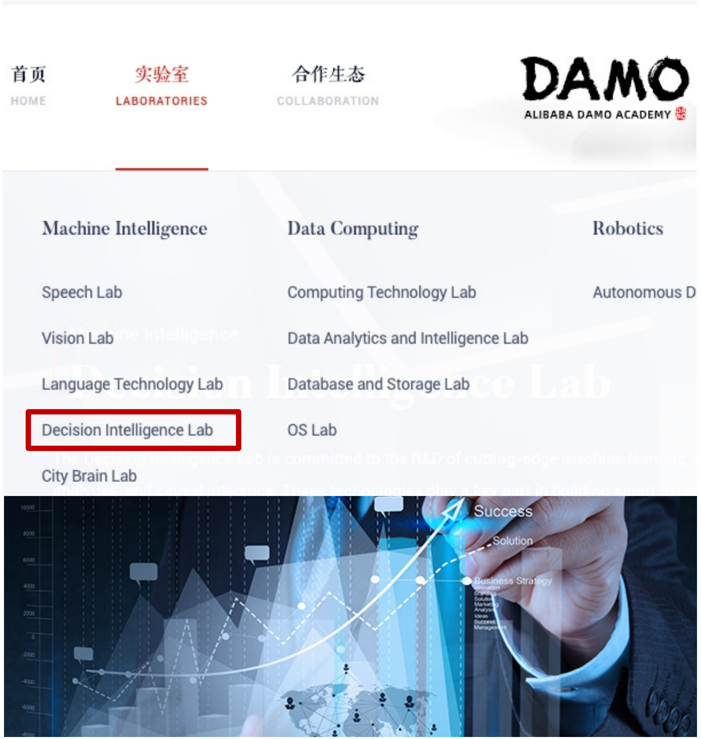
Staff Engineer / Manager, DAMO Academy, Alibaba Group (U.S.) Inc.

Oct 28, 2022

Credits to our group members:
Tian Zhou, Ziqing Ma, Weiqi Chen, Chaoli Zhang, Zhiqiang Zhou, Linxiao Yang, Yingying Zhang, Liang Sun, Wotao Yin, Rong Jin, etc.

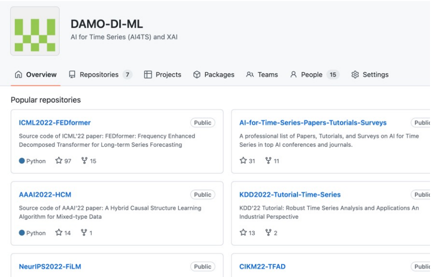
Who we are

- Alibaba DAMO Academy - Decision Intelligence Lab
  - Research Focus:
    - **AI for Time Series (AI4TS)**
    - **Explainable Artificial Intelligence (XAI)**
    - **Optimization**
    - **Others: ML, RL, ...**
  - Products and Applications:
    - **AIOps:** Cloud Autoscaling AHPA
    - **Green Energy AI:** eForecaster
    - **Optimization Solver:** MindOpt
    - **Others: ...**
  - More Information:
    - https://damo.alibaba.com/labs/decision-intelligence



Our R&D in *Time Series* Area

- Publication (2019~now):
  - 30+ in NeurIPS, ICML, AAAI, KDD, SIGMOD, ICDE, TKDE, PIEEE etc.
  - 20+ Patents: US, China
- Open Source: Paper code, Benchmark, Dataset, etc.
- Other Academic: Workshop, Tutorial, Competition, Journal SI, AIR with Prof.
- Products
  - Time Series Analysis: MindAnalytics (Internal, for 10+ BU, millions of calls per day)
  - Autoscaling: AHPA (at Alibaba Cloud)
  - Energy Forecasting: eForecaster (at Alibaba Cloud)
- Competition/Awards:
  - 2023 AAAI/IAAI Innovative Application Award (Autoscaling)
  - 2022 GDEC, Digital Economy Industrial Innovation Achievement
  - 2022 ICASSP Grand Challenge (AIOps in Network), First Place
  - 2021 State Grid New Energy Power Forecasting, Runner-up
  - 2021 International AIOps Challenge, Runner-up
  - ...



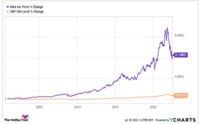
Outline

- ➢ ***Introduction***
- ❑ **Robust Time Series Processing Blocks**
- ❑ **Robust Time Series Applications**

Time Series Data is Ubiquitous

- A wide range of time series data
  - AIOps
  - IoT
  - Business data, e.g., sales volume, stock price
  - Many others

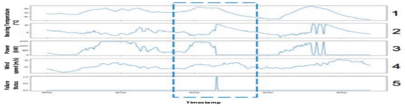
stocks



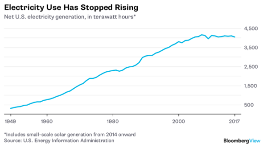
sales

goods consumption

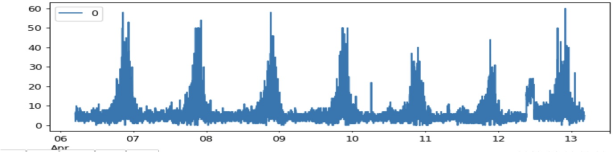
sensor



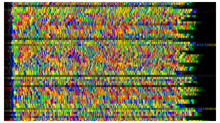
power demand



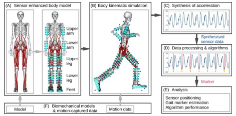
Cloud service monitoring



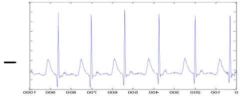
DNA sequence



motion detect



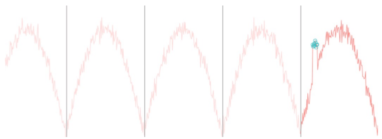
ECG



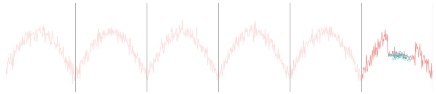
Typical Application: Time Series Anomaly Detection

Different Types of Anomalies

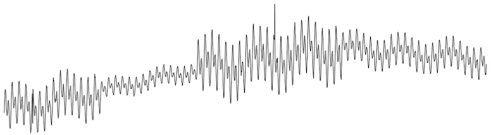
Spike & dip



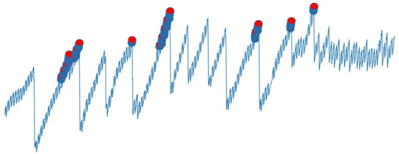
Change of mean



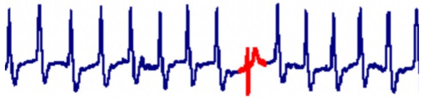
Change of variance



Long-term trend change



Subsequence anomaly



- Business service
- Software service
- Container and VM
- Server and components
- Network
- Data center infrastructure

In typical applications of Elastic Computing Service at Alibaba Cloud (2020):

- 5M+ servers monitored
- 50M+ metrics monitored
- In 2020 some typical root identification time is ~ 1 person-month
- Some cases incur 1M+ USD loss

- ***Huge amount of data calls for automatic and accurate anomaly detection algorithm***

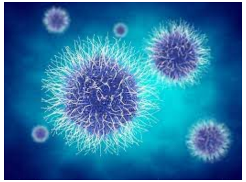
Typical Application: Time Series Forecasting

Energy Forecasting

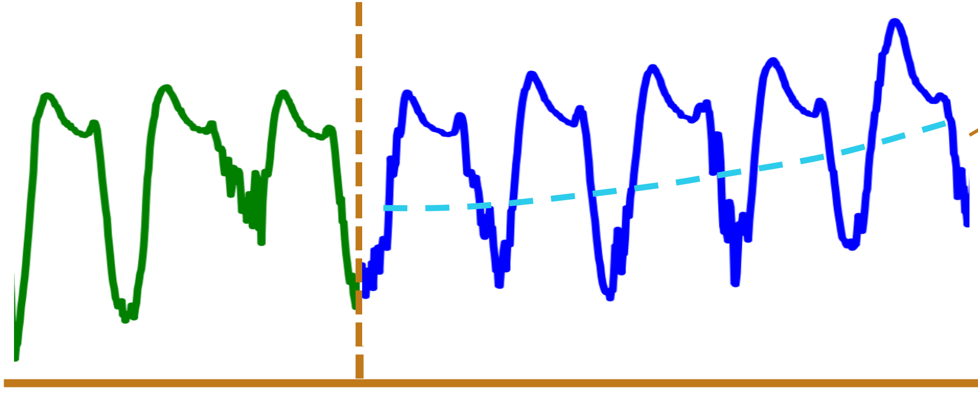
Traffic Forecasting

Economics Forecasting

Weather Forecasting



Disease Forecasting



Long-term forecasting
for decision and planning

- Modeling complicated temporal dependencies (periodicity and trend) for accurate prediction
- Less computational complexity

Time Series in Industry

- Time series with big data
  - Hundreds of millions metrics or even more
  - High frequency and real-time processing
  - Limited labels in many applications
  - Low deployment cost
- Time series data with complex patterns
  - Noises and outliers
  - Periodicity/Seasonality: no/single/multiple/variable periodic components
  - Fully automatic solution is challenging but highly preferred
- Connected with other applications
  - Anomaly detection is closely related to root cause analysis
  - Forecasting is closely related to decision-making, e.g., autoscaling

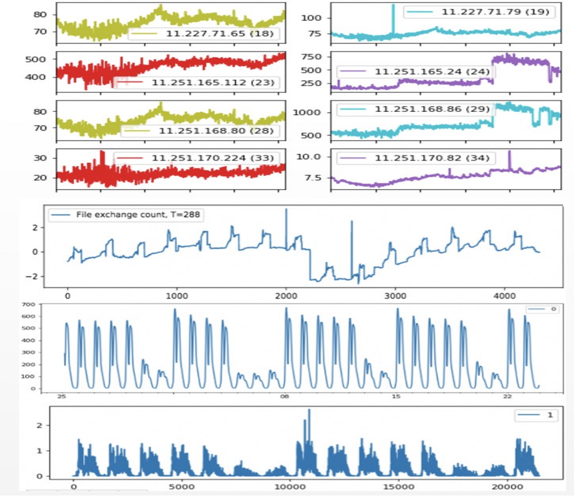
*Our solution: **robust and intelligent** algorithm family for time series analysis*

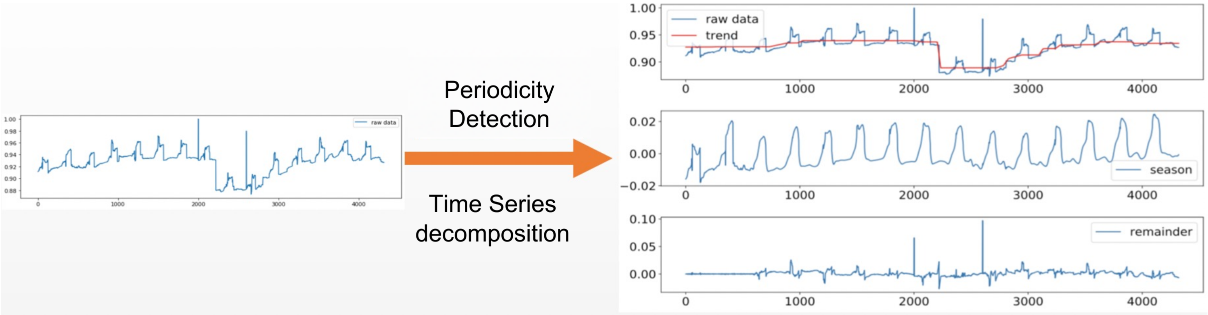
Outline

- ❑ **Introduction**
- ➢ ***Robust Time Series Processing Blocks***
    - Time Series Periodicity Detection
    - Time Series Trend Filtering
    - Time Series Seasonal-Trend Decomposition
- ❑ **Robust Time Series Applications**

Complex Periodic Time Series

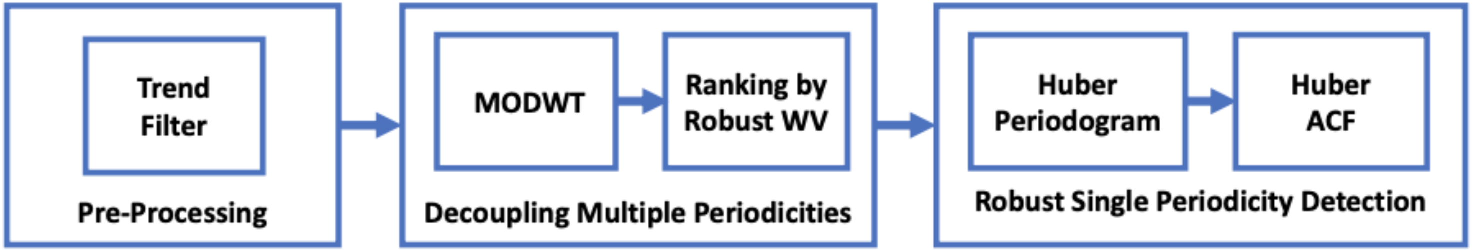
- Time series
  - Periodicity/seasonality, trend, and noises are common
  - Periodicity detection, decomposition, similarity are crucial for downstream tasks (forecasting, anomaly detection, etc.)
- Challenges of robust time series processing blocks
  - Noise, outliers, missing data, abrupt trend changes
  - None/single/multiple periodicity
  - Need automatic and accurate processing for large-scale industrial time series





RobustPeriod Algorithm

- Robust and general periodicity detection: RobustPeriod
  - Key idea*: first isolate periodic components*, then detect each one robustly ("divide and conquer")
  - Three blocks: Pre-processing, Decoupling Multiple Periodicities, Robust Single Periodicity Detection



$$y_t = \tau_t + \sum_{i=1}^{m} s_{i,t} + r_t$$

For periodic time series, period lengths are denoted as T_i ,i = 1, $\cdots m$, m is the number of periodic components

Qingsong Wen, Kai He, Liang Sun, Yingying Zhang, Min Ke, Huan Xu, "RobustPeriod: Time-Frequency Mining for Robust Multiple Periodicity Detection," in Proc. ACM SIGMOD International Conference on Management of Data (SIGMOD 2021), Xi'an, China, Jun. 2021.

Decouple Multiple Periodicities

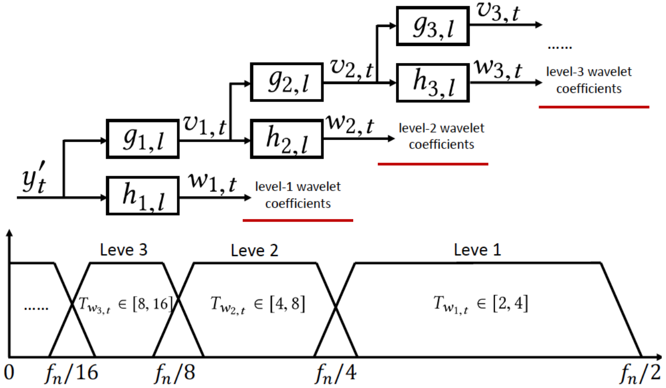
- MODWT for decoupling multiple periodicities
  - MODWT: maximal overlap discrete wavelet transform

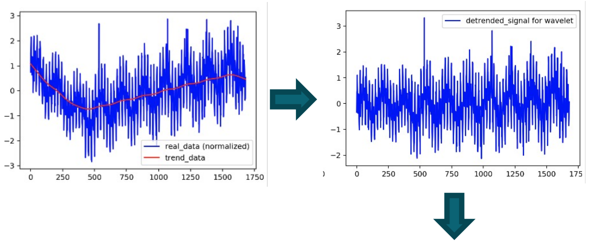
*j*th level *wavelet coefficients*

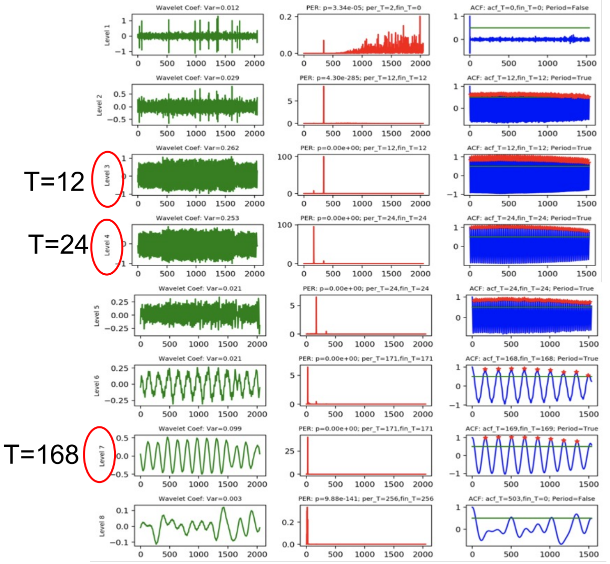
$$w_{j,t} = \sum_{l=0}^{L_j-1} h_{j,l} y'_{t-l \bmod N}$$

*j*th level scaling coefficients

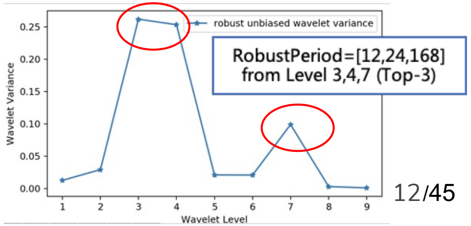
$$v_{j,t} = \sum_{l=0}^{L_j-1} g_{j,l} y'_{t-l \bmod N}$$







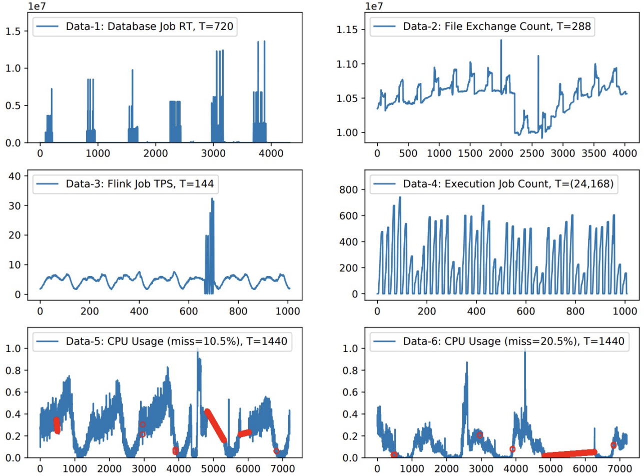
- Wavelet variance ranking for speedup
  - Wavelet variance decomposition: $\hat{\sigma}^2_{\mathbf{y}'} = \sum_{j=1}^{J_0} \hat{\sigma}^2_{\mathbf{w}_j} + \hat{\sigma}^2_{\mathbf{v}_{J_0}}$
  - Relationship to power spectral density (PSD): $\hat{\sigma}^2_{\mathbf{w}_j} \approx \int_{1/2^{j+1} \leq |f| \leq 1/2^j} S_{\mathbf{y}'}(f)df$
  - If there is a periodic component in the *j*th level wavelet coefficient, a large wavelet variance would be expected



Periodicity Detection Comparisons

- Real-world datasets:
  - Alibaba Cloud monitoring data with outliers, noise, trend, missing data (**single/multiple** periodicity)

Monitoring datasets from Alibaba Cloud



Algorithms	Data-1, T=720 Database RT	Data-2, T=288 File Exchange	Data-3, T=144 Flink TPS
Siegel	(655,769,...)	(288,576,...)	(141,144)
AUTOPERIOD	(353,241,9)	(288,439,...)	(68,141)
Wavelet-Fisher	(372,745,...)	(282,585,...)	(73,146)
RobustPeriod	**721**	**288**	**144**

Algorithms	Data-4, T=(24,168) Job Count	Data-5, T=1440 CPU Usage	Data-6, T=1440 CPU Usage
Siegel	(24,168)	(1459,2597,...)	(1575,1063,...)
AUTOPERIOD	(24,26)	(1488,739,...)	(366,2880,...)
Wavelet-Fisher	(12,24,...)	(1489,712,...)	(1489,364,...)
RobustPeriod	**(24,168)**	**1431**	**1426**

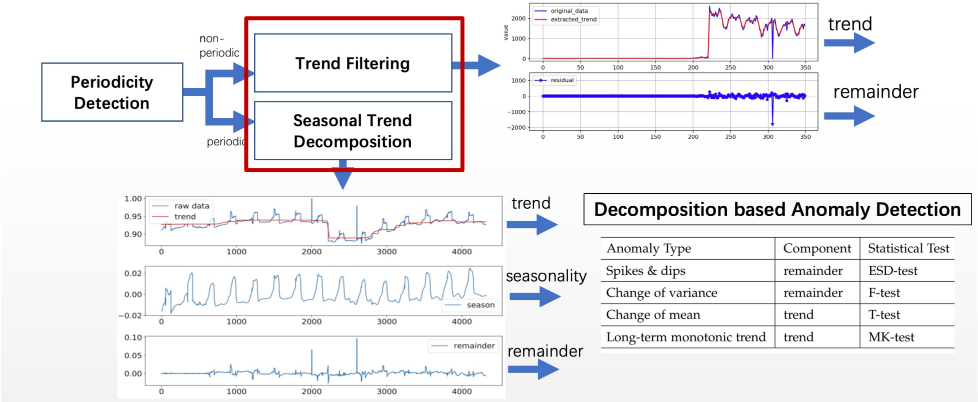
Algorithms	Robust to outliers	Robust to amplitude changes	Robust to trend changes	Support multiple periods	Not need priors for number of periods
ACF	×	×	×	×	×
FFT-Fisher	×	×	×	×	×
FFT-Siegel	×	×	×	✓	✓
Bandpass+ACF	×	✓	×	✓	×
Wavelet-Fisher	×	×	×	✓	✓
AUTOPERIOD	×	×	×	✓	✓
RobustPeriod	✓	✓	✓	✓	✓

Time Series Decomposition

- Trend filtering, seasonal-trend decomposition

$$y_t = \tau_t + r_t \qquad y_t = \tau_t + \sum_{i=1}^{m} s_{i,t} + r_t$$

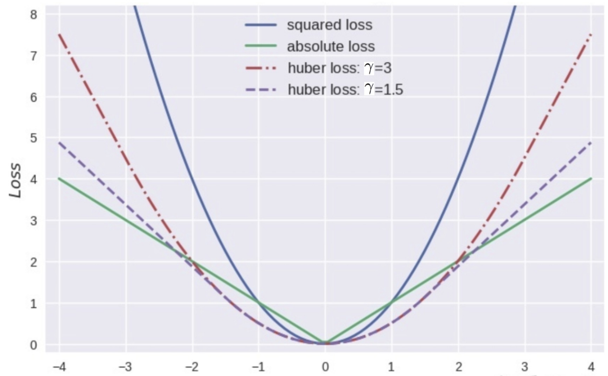
- Why need decomposition
  - Insights and interpretability from different components
  - Different utilization by components
    - e.g.: anomaly detection



Anomaly Type	Component	Statistical Test
Spikes & dips	remainder	ESD-test
Change of variance	remainder	F-test
Change of mean	trend	T-test
Long-term monotonic trend	trend	MK-test

RobustTrend Filter

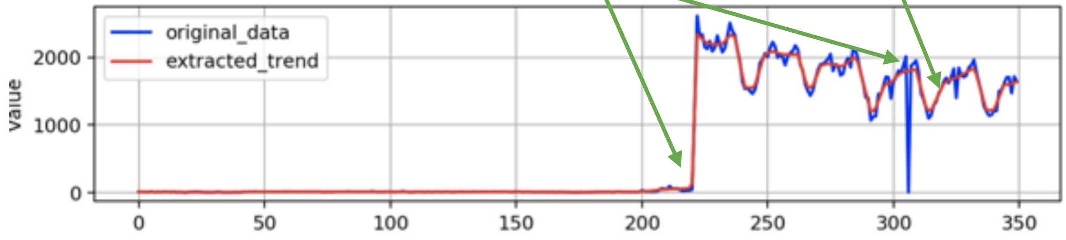
- Huber loss: robust to outliers
- 1st order L1 regularization: abrupt trend changes
- 2nd order L1 regularization: slow trend changes and can effectively reduce staircasing effect



$$\min_{\boldsymbol{\tau}} \ g_\gamma(\mathbf{y}-\boldsymbol{\tau}) + \lambda_1 ||\mathbf{D}^{(1)}\boldsymbol{\tau}||_1 + \lambda_2 ||\mathbf{D}^{(2)}\boldsymbol{\tau}||_1$$

where

$$g_\gamma(x_i) = \begin{cases} \frac{1}{2}x_i^2, & |x_i| \leq \gamma \\ \gamma|x_i| - \frac{1}{2}\gamma^2, & |x_i| > \gamma \end{cases}$$

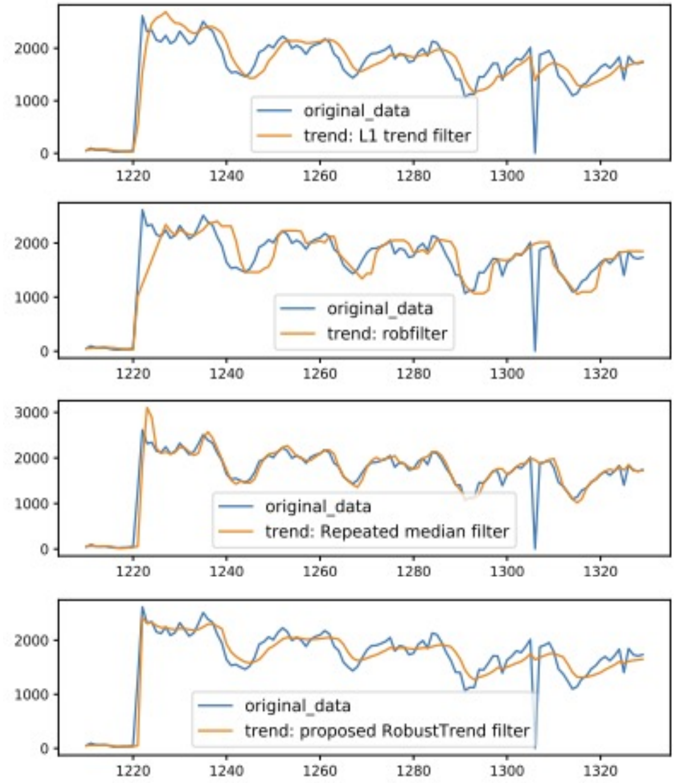


$$\mathbf{D}^{(1)} = \begin{bmatrix} 1 & -1 & & & \\ & 1 & -1 & & \\ & & \ddots & & \\ & & & 1 & -1 \end{bmatrix}, \ \mathbf{D}^{(2)} = \begin{bmatrix} 1 & -2 & 1 & & & \\ & 1 & -2 & 1 & & \\ & & & \ddots & & \\ & & & 1 & -2 & 1 \end{bmatrix}$$

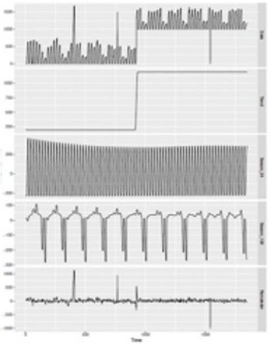
Qingsong Wen, Jingkun Gao, Xiaomin Song, Liang Sun, Jian Tan, "RobustTrend: A Huber Loss with a Combined First and Second Order Difference Regularization for Time Series Trend Filtering," in Proc. 28th International Joint Conference on Artificial Intelligence (IJCAI 2019), Macao, China, Aug. 2019.

Experiments on Real-World Data

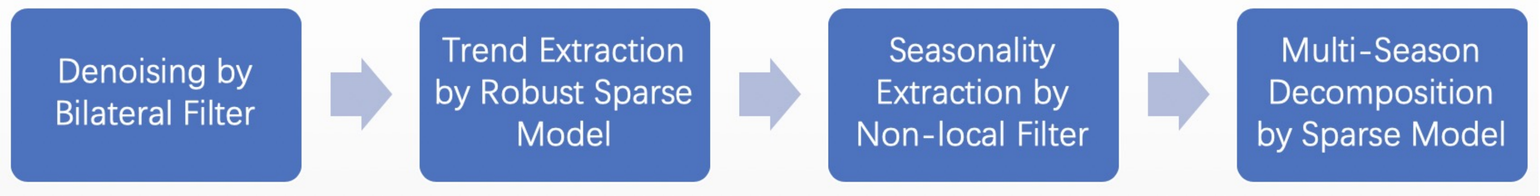
- Compare trend filters of SOTA models
- Performance highlights
  - L1 trend filter: sensitive to the outliers
  - robfilter: some delay when trend changes
  - Repeated median filter: overshoots trend estimation
  - **RobustTrend filter**: best tradeoff under outliers and abrupt trend changes



RobustSTL Algorithm



- Fast and robust seasonal-trend decomposition: $y_t = \tau_t + \sum_{i=1}^{m} s_{i,t} + r_t$
  - Key idea: *sequentially and robustly* extract components in time series
  - Four blocks: noise removal, trend extraction, seasonality extraction, multi-season decomposition
  - *Efficient* GADMM for trend extraction and multi-season decomposition: O(*NlogN*)



Qingsong Wen, Jingkun Gao, Xiaomin Song, Liang Sun, Huan Xu, Shenghuo Zhu, "RobustSTL: A Robust Seasonal-Trend Decomposition Algorithm for Long Time Series," in Proc. 33th AAAI Conference on Artificial Intelligence (AAAI 2019), Honolulu, Hawaii, Jan. 2019. (single periodicity version)

Qingsong Wen, Zhe Zhang, Yan Li, Liang Sun, "Fast RobustSTL: Efficient and Robust Seasonal-Trend Decomposition for Time Series with Complex Patterns," in Proc. 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD 2020), San Diego, CA, Aug. 2020. (multiple periodicities and high-speed version)

Linxiao Yang, Qingsong Wen, Bo Yang, Liang Sun, "A Robust and Efficient Multi-Scale Seasonal-Trend Decomposition," in Proc. IEEE 46th International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2021), Toronto, Canada, Jun. 2021. (multiple-scale version)

Trend Extraction

- Robust sparse model: LAD loss with two L1 regularizations
  - Mitigate season effect by *seasonal difference*

$$g_t = \nabla_T y'_t = y'_t - y'_{t-T} = \nabla_T \tau_t + \nabla_T s_t + \nabla_T r'_t$$

$$T = \max T_i, i = 1, 2, \cdots, m$$

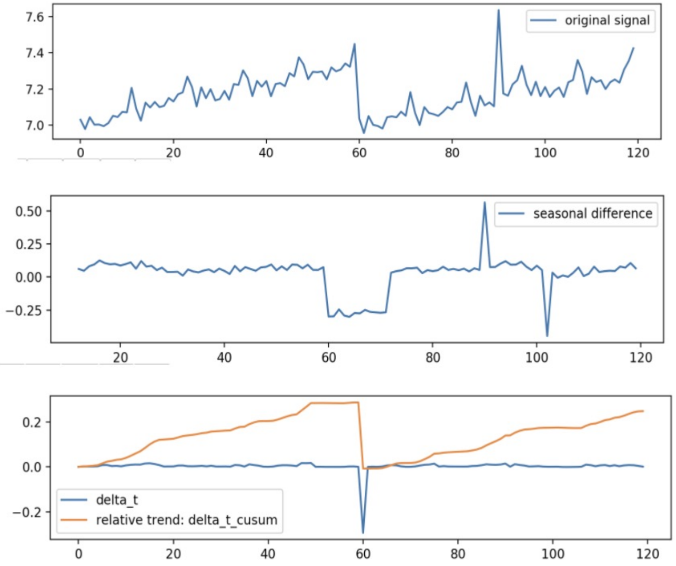
  - Estimate the *trend difference robustly*: LAD loss, two L1 regs

$$\sum_{t=T+1}^{N} |g_t - \sum_{i=0}^{T-1} \nabla \tau_{t-i}| + \lambda_1 \sum_{t=2}^{N} |\nabla \tau_t| + \lambda_2 \sum_{t=3}^{N} |\nabla^2 \tau_t|$$

$$\boxed{\begin{array}{l} \text{To capture abrupt change: } \nabla \tau_t = \tau_t - \tau_{t-1} \\ \text{To capture slow change: } \nabla^2 \tau_t = \tau_t - 2\tau_{t-1} + \tau_{t-2} \end{array}}$$

  - Finally, recover trend by *cumulative sum*

$$\tilde{\tau}^r_t = \tilde{\tau}_t - \tau_1 = \tilde{\tau}_t - \tilde{\tau}_1 = \begin{cases} 0, & t = 1 \\ \sum_{i=2}^{t} \nabla \tilde{\tau}_i, & t \geq 2 \end{cases}$$



Seasonality Extraction

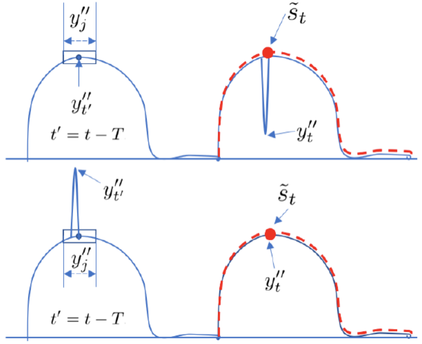
- Weighted non-local seasonal filtering
  - Consider K neighbourhoods for each season
  - Filter weights based on both value and position difference, similar to Bilateral filtering
  - Robust to *outlier* and adaptive to *seasonal shift*

$$\tilde{s}_t = \frac{1}{z}\sum_{i=1}^{m}\alpha_i \sum_{(t'_i,j)\in\Omega} w^t_{(t'_i,j)} y''_j$$

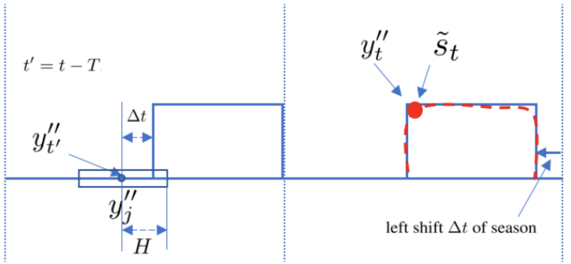
$$w^t_{(t'_i,j)} = e^{-\frac{|j-t'_i|^2}{2\delta_d^2} - \frac{|y''_j - y''_{t_i}|^2}{2\delta_i^2}}$$

$$\Omega = \{(t'_i,j) | (t'_i = t \pm k \times T_i, j = t'_i \pm h)\}$$

$$k = 1, 2, \cdots, K;\ h = 0, 1, \cdots, H$$



(a) Outlier robustness



(b) Season shift adaptation

GADMM for Efficient Computation: O(*NlogN*)

- ADMM→GADMM for trend extraction

ADMM

$$\mathbf{x}_{t+1} = \arg\min_{\mathbf{x}} L_\rho(\mathbf{x}, \mathbf{y}, \mathbf{u}) = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T(\mathbf{b} + \mathbf{y}_t - \frac{1}{\rho}\mathbf{u}_t) \quad \boxed{O(N^2)}$$

$$\mathbf{y}_{t+1} = \arg\min_{\mathbf{y}} L_\rho(\mathbf{x}, \mathbf{y}, \mathbf{u}) = S_{1/\rho}(\mathbf{A}\mathbf{x}_{t+1} - \mathbf{b} + \frac{1}{\rho}\mathbf{u}_t) \quad \boxed{O(N)}$$

$$\mathbf{u}_{t+1} = \arg\min_{\mathbf{u}} L_\rho(\mathbf{x}, \mathbf{y}, \mathbf{u}) = \mathbf{u}_t + \rho(\mathbf{A}\mathbf{x}_{t+1} - \mathbf{y}_{t+1} - \mathbf{b}) \quad \boxed{O(N)}$$

$$\mathbf{A} = \begin{bmatrix} \mathbf{M}_{(N-T)\times(N-1)} \\ \lambda_1 \mathbf{I}_{(N-1)\times(N-1)} \\ \lambda_2 \mathbf{D}_{(N-2)\times(N-1)} \end{bmatrix}, \mathbf{b} = \begin{bmatrix} \mathbf{g}_{(N-T)\times 1} \\ \mathbf{0}_{(2N-3)\times 1} \end{bmatrix}$$

- Similar GADMM can be formulated for multi-season decomposition

GADMM
X-update relaxation for trend extraction

$$\mathbf{x}_{t+1} = \arg\min_{\mathbf{x}} L_\rho(\mathbf{x}, \mathbf{y}, \mathbf{u}) = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T(\mathbf{b} + \mathbf{y}_t - \frac{1}{\rho}\mathbf{u}_t)$$

$$\mathbf{x}_{t+1} \approx \arg\min_{\mathbf{x}} \left(L_\rho(\mathbf{x}, \mathbf{y}, \mathbf{u}) + \underbrace{\|\mathbf{x} - \mathbf{x}_t\|^2_{\hat{\mathbf{H}}}}_{\text{extra proximal term}} \right) \quad O(N^2)$$

$$\hat{\mathbf{H}} = \mathbf{H} - \mathbf{A}^T\mathbf{A} = \tilde{\mathbf{A}}^T\tilde{\mathbf{A}} - \mathbf{A}^T\mathbf{A} = \hat{\mathbf{M}}^T\hat{\mathbf{M}} + \lambda_2^2\hat{\mathbf{D}}^T\hat{\mathbf{D}}$$

$$\mathbf{x}_{t+1} \approx \mathbf{H}^{-1}\mathbf{v} = \mathbf{F}^{-1}\mathrm{diag}^{-1}(\mathbf{F}\mathbf{h})\mathbf{F}\mathbf{v}$$
$$= \mathrm{IDFT}\left(\mathrm{DFT}(\mathbf{v})./\mathrm{DFT}(\mathbf{h})\right), \quad \boxed{O(NlogN)}$$

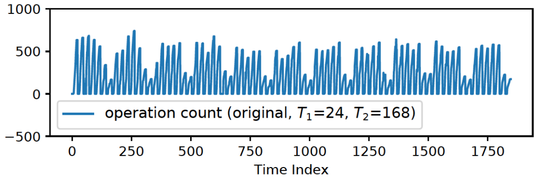
$$\tilde{\mathbf{A}} = \begin{bmatrix} \tilde{\mathbf{M}}_{(N-T)\times(N-1)} \\ \lambda_1 \mathbf{I}_{(N-1)\times(N-1)} \\ \lambda_2 \tilde{\mathbf{D}}_{(N-2)\times(N-1)} \end{bmatrix} \tilde{\mathbf{D}} = \begin{bmatrix} \mathbf{D} \\ \hat{\mathbf{D}} \end{bmatrix} = \left.\begin{bmatrix} 1 & -1 & & \\ & 1 & -1 & \\ & & \ddots & \\ & & 1 & -1 \\ -1 & & & 1 \end{bmatrix}\right\} \mathbf{D}$$

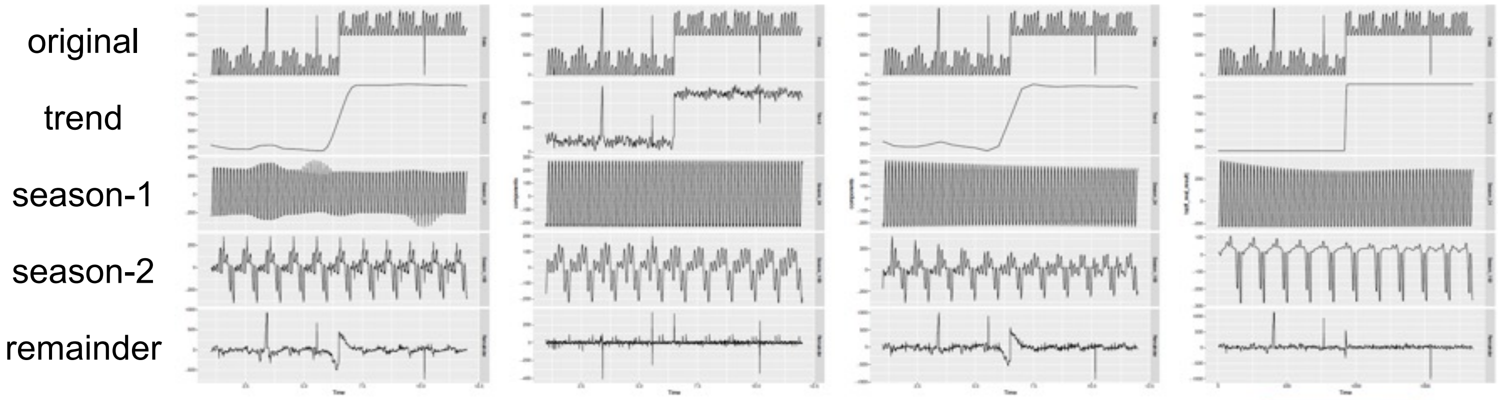
construct circulant matrix

$$\tilde{\mathbf{M}} = \begin{bmatrix} \mathbf{M} \\ \hat{\mathbf{M}} \end{bmatrix} = \left.\begin{bmatrix} 1 & \cdots & 1 & & & \\ & 1 & \cdots & 1 & & \\ & & \ddots & & & \\ & & & 1 & \cdots & 1 \\ 1 & & & & 1 & \cdots \\ \ddots & & & & & \ddots \\ \cdots & 1 & & & & 1 \end{bmatrix}\right\} \mathbf{M}$$

Decomposition on Real-World Data

- Real-world data with daily and weekly periodic components
  - adding abrupt trend change
  - adding 2 single-point outliers, 1 pattern outlier spanning 1 day





(a) MSTL (b) TBATS (c) AutoSTR (d) Fast RobustSTL

RobustSTL is robust to abrupt trend changes and outliers

Comparisons

- Speed experiments

N	Metrics	RobustSTL-ADMM	RobustSTL-PDHG	**Fast RobustSTL**
N=1080	Iter	41	310	37
	Time(Sec)	0.142	0.0319	0.0109
	SpeedUp		4.45x	**13.0x**
N=2160	Iter	48	319	40
	Time(Sec)	1.11	0.0571	0.0295
	SpeedUp		19.4x	**37.6x**
N=4320	Iter	68	602	37
	Time(Sec)	5.98	0.191	0.0988
	SpeedUp		31.3x	**60.5x**
N=8640	Iter	92	1377	53
	Time(Sec)	36.7	0.62	0.254
	SpeedUp		59.1x	**144x**

Algorithm	Time (N=4320)
AutoSTR	4441 seconds
TBATS	60 seconds
(M)STL	0.2 second
RobustSTL	**0.1 second**

RobustSTL is more efficient than others

- Algorithm comparisons

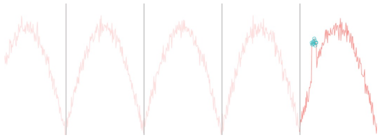
Algorithm	Robust to outlier	Robust to seasonal shift	Robust to trend changes	Support multiple periods	Efficient computation
ARIMA/SEATS	×	×	×	×	×
SSA	×	×	×	×	×
TBATS	×	×	✓	✓	×
STL	×	×	×	×	✓
MSTL	×	×	×	✓	×
STR	✓	✓	×	✓	×
RobustSTL	✓	✓	✓	✓	✓

Outline

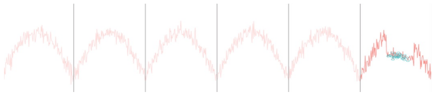
- ❑ **Introduction**
- ❑ **Robust Time Series Processing Blocks**
- ➢ ***Robust Time Series Applications***
    - Time Series Anomaly Detection
    - Time Series Forecasting
    - Other Time Series Applications

Time Series Anomaly Detection: Background

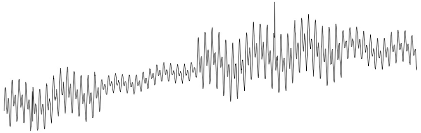
- Time-series anomalies
  - *Points/sequences* that significantly deviate *from the* normal pattern
  - **Differences** from <u>static data</u> anomaly detection:
    - Anomalies have temporal context
    - Noise is non-stationary (i.e., time-varying noise)
    - Lack of anomaly labels
- Types of anomalies
  - Point: Spikes/Dips, CoM, CoV, LMT
  - Subsequence anomaly: identifying anomalous subsequences (sequences of points)
- Models
  - *Traditional methods, deep models*



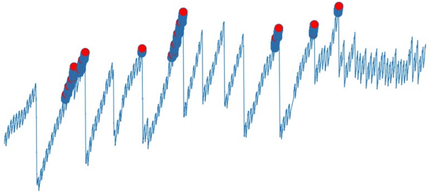
spike & dip



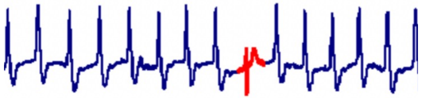
change of mean (CoM)



change of variance (CoV)



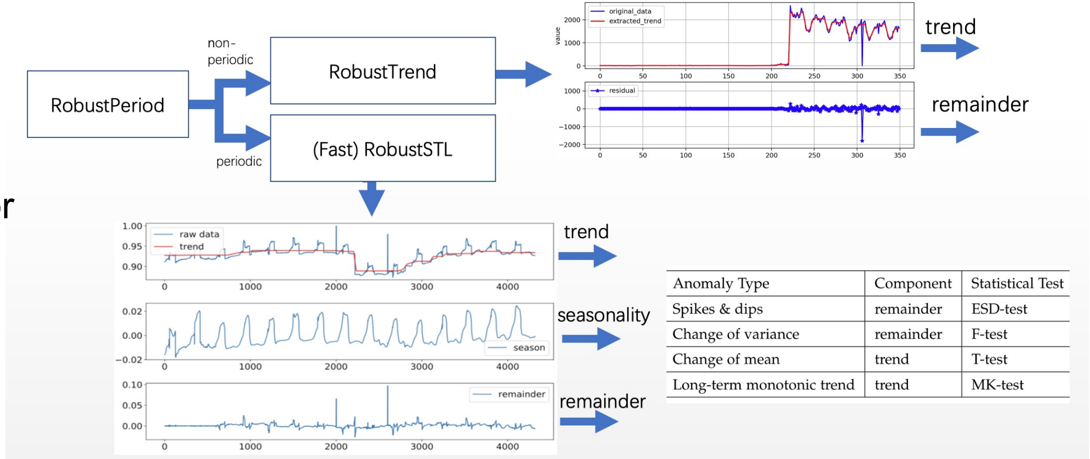
long-term trend change (LMT)



subsequence anomaly

Anomaly Detection: Decomposition based Model

- *Robust* Decomposition + Statistics
  - *Robust time series decompositions* reduce complexity and bring explainability
  - Robust statistical tests on each components lead to high accuracy for *different types of anomalies*

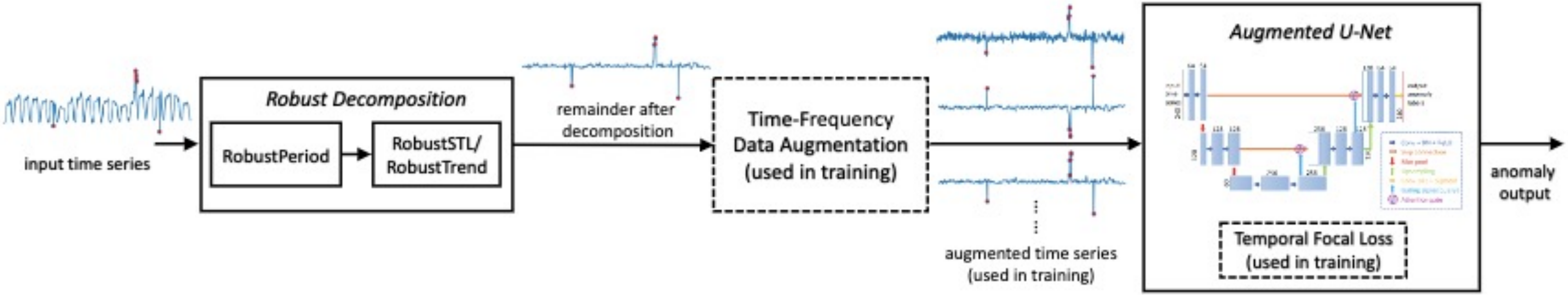


Anomaly Type	Component	Statistical Test
Spikes & dips	remainder	ESD-test
Change of variance	remainder	F-test
Change of mean	trend	T-test
Long-term monotonic trend	trend	MK-test

Yingying Zhang, Zhengxiong Guan, Huajie Qian, Leili Xu, Hengbo Liu, Qingsong Wen, Liang Sun, Junwei Jiang, Lunting Fan, Min Ke, "CloudRCA: A Root Cause Analysis Framework for Cloud Computing Platforms," in Proc. 30th ACM International Conference on Information and Knowledge Management (CIKM 2021), Queensland, Australia, Nov. 2021.

Anomaly Detection: RobustTAD

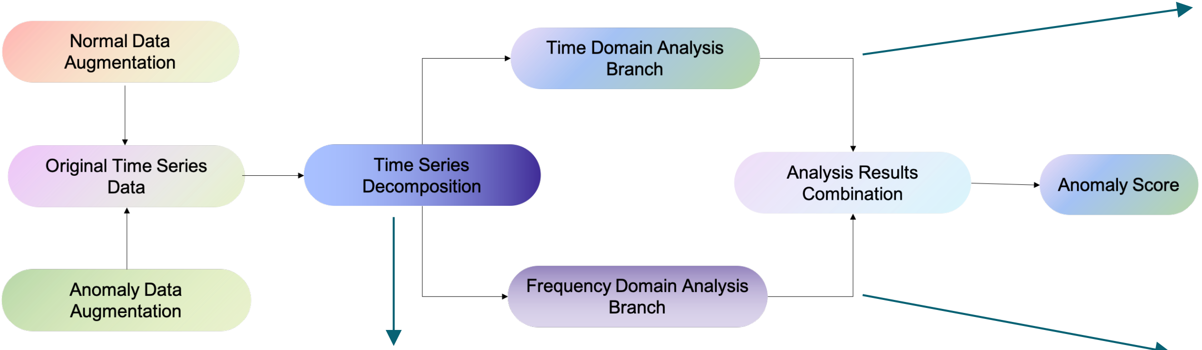
- Decomposition + *Data Augmentation* + *U-Net (deep model)*
  - Robust decomposition: handle complicated patterns, and simplify neural network
  - Data augmentation: mitigate the effects of limited labeled data
  - U-Net: capture multi-scale information of time series
  - Temporal focal loss: label-based weight and value-based weight for unbalanced label

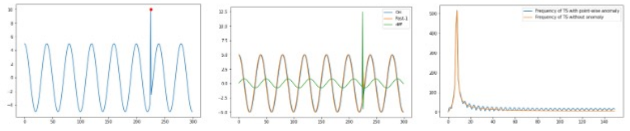


Jingkun Gao, Xiaomin Song, Qingsong Wen, Pichao Wang, Liang Sun, Huan Xu, "RobustTAD: Robust Time Series Anomaly Detection via Decomposition and Convolutional Neural Networks," in ACM SIGKDD Workshop on Mining and Learning from Time Series (KDD-MiLeTS 2020), San Diego, CA, Aug. 2020.

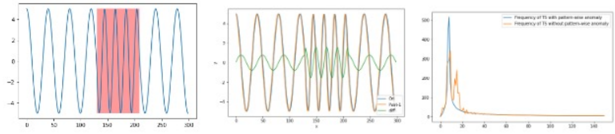
Anomaly Detection: TFAD

- Data Augmentation + Decomposition + TCN + *Time-Frequency Processing*
  - Anomalies types: seasonal anomaly, trend anomaly, global/context point anomaly
  - Point anomaly easier to detect in Time domain; seasonal anomaly easier to detect in Frequency domain

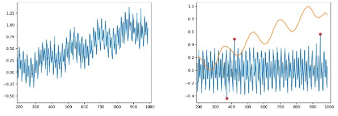




(a) Point-wise anomaly example (b) Diff in time domain w/o point-wise anomaly (c) Diff in frequency domain w/o point-wise anomaly



(a) Seasonality anomaly example. (b) Diff in time domain w/o seasonality anomaly (c) Diff in frequency domain w/o seasonality anomaly



(a) Original time series (b) Time series with decomposition

Chaoli Zhang, Tian Zhou, Qingsong Wen, Liang Sun, "TFAD: A Decomposition Time Series Anomaly Detection Architecture with Time-Freq Analysis," in Proc. 31st ACM International Conference on Information and Knowledge Management (CIKM 2022), Atlanta, GA, Oct. 2022.

Anomaly Detection: TFAD Experiments

Dataset characteristics

data set	#Curves/Dims	#Points	%Anomaly
KPI	58	5922913	2.26
Yahoo	367	572966	0.68
SMAP	55	429735	12.8
MSL	27	66709	10.5

- Univariate time series
    - Datasets: Yahoo, KPI
    - Algs: SPOT, DSPOT, DONUT, SR, SR-CNN, SR-DNN, NCAD
    - Main metric: F1 score

Results of univariate time series anomaly detection

Model	Yahoo (un.)	KPI (un.)	KPI (sup.)
SPOT	33.8	21.7	–
DSPOT	31.6	52.1	–
DONUT	2.6	34.7	–
SR	56.3	62.2	–
SR-CNN	65.2	77.1	–
SR-DNN	–	–	81.1
NCAD	**81.16** ± 1.43	76.64 ± 0.89	79.20 ± 0.92
TFAD	81.13 ± **0.52**	**79.80** ± **0.74**	**82.10** ± **0.42**

- Multivariate time series
    - Datasets: SMAP, MSL
    - Algs: AnoGAN, DeepSVDD, DAGMM, LSTM-VAE, MSCRED, OmniAnomaly, MTAD-GAT, THOC, NCAD
    - Main metric: F1 score

Results of multivariate time series anomaly detection

Model	SMAP (un.)	MSL (un.)
AnoGAN	74.59	86.39
DeepSVDD	71.71	88.12
DAGMM	82.04	86.08
LSTM-VAE	75.73	73.79
MSCRED	77.45	85.97
OmniAnomaly	84.34	89.89
MTAD-GAT	90.13	90.84
THOC	95.18	93.67
NCAD	94.45 ± **0.68**	95.60 ± 0.59
TFAD	**96.32** ± 1.57	**96.41** ± **0.34**

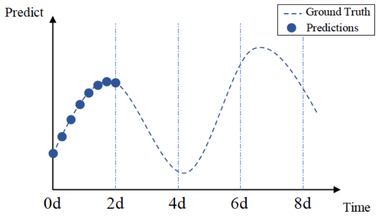
Anomaly Detection: TFAD Experiments

- Ablation studies
  - Decomposition module contributes 30%↑
  - Single domain analysis performs not well
  - F(NormalAug + AbnormalAug) > F(NormalAug) + F(AbnormalAug)
  - F1 score increases + variance decreases with frequency branch added

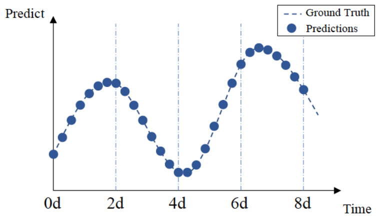
case	TCN	Dec	NorAug	TimeAnAug	FreqAnAug	FreqBran	Precision	Recall	F1 score
Freq Branch						✓	13.81 ± 3.08	35.97 ± 4.40	19.59 ± 2.81
Time Branch	✓						44.888 ± 0.095	57.227 ± 0.0381	50.312 ± 0.0569
(a)	✓	✓					57.557 ± 5.374	81.111 ± 5.339	66.968 ± 2.58
(b)	✓	✓	✓				57.869 ± 4.65	80.49 ± 4.85	67.099 ± 3.075
(c)	✓	✓		✓			62.569 ± 7.059	**89.45** ± 7.84	72.942 ± 2.644
(d)	✓	✓	✓	✓			68.528 ± 9.412	85.949 ± 11.3698	74.934 ± 2.908
(e)	✓	✓	✓	✓	✓		69.444 ± 7.133	86.638 ± 8.044	76.385 ± 2.387
TFAD	✓	✓	✓	✓	✓	✓	**79.176** ± 1.875	85.231 ± 1.464	**82.058** ± 0.4199

Time Series Forecasting: Background

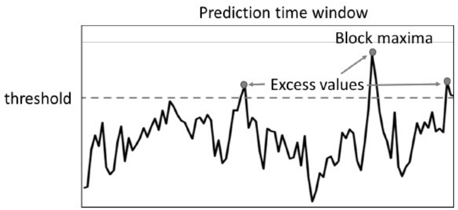
- Different forecasting types
  - **Short-term** forecasting: predict the near future
  - **Long-term** forecasting: predict the future with an extended period
  - **Extreme value** forecasting: predict the extreme values
  - **Point or Probabilistic** forecasting: predict point value or interval/probability distribution
- Challenges:
  - Accuracy, robustness
- Models:
  - Traditional: Statistical (ARIMA, ETS, Prophet)
  - *Ensemble: Tree, MLP*
  - *Deep Models: CNN, RNN, Transformers*



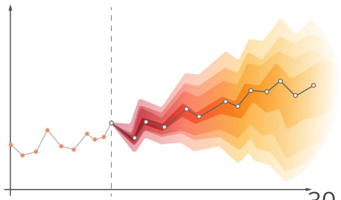
Short term forecasting



Long term forecasting

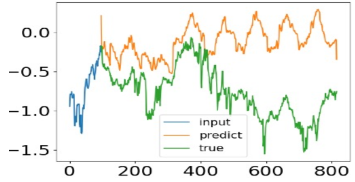


Extreme value forecasting



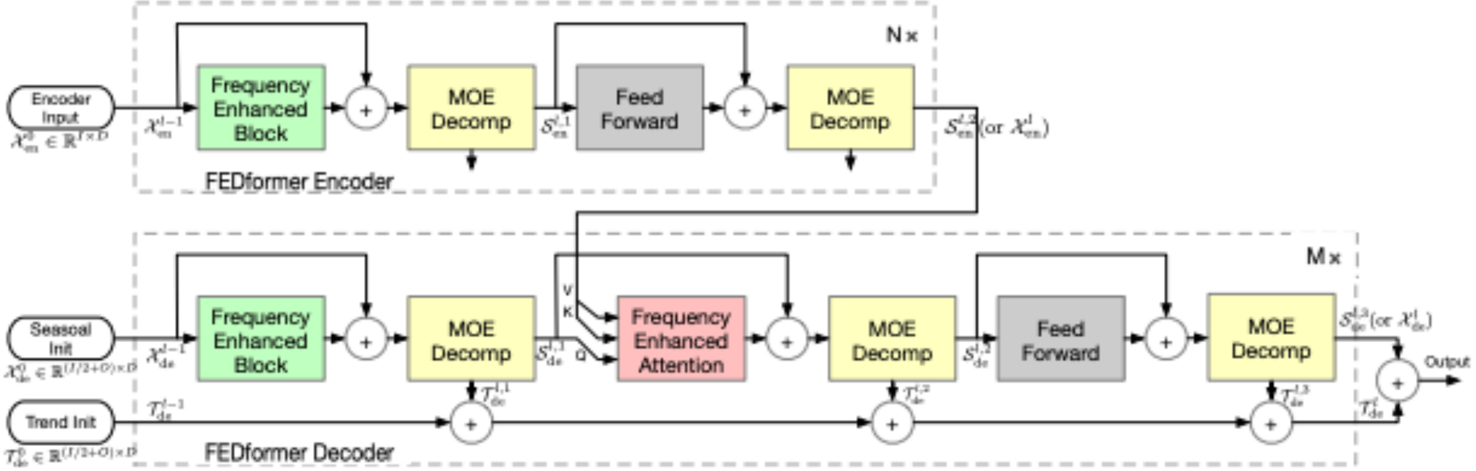
Probabilistic forecasting

Forecasting with Transformer: FEDformer



Trend and seasonality discrepancy of the conventional Transformer model

- FEDformer: frequency enhanced decomposed Transformer
  - Efficient and robust *frequency domain processing*: to capture key structures in time series
  - Mixture of experts *seasonal-trend decomposition*: to better capture global properties in time series



Tian Zhou, Ziqing Ma, Qingsong Wen, Xue Wang, Liang Sun, Rong Jin, "FEDformer: Frequency Enhanced Decomposed Transformer for Long-term Series Forecasting," ICML 2022.

Forecasting with Transformer: FEDformer

- FEDformer: frequency enhanced decomposed Transformer
  - Frequency enhanced block: substitute self-attention
  - Frequency enhanced attention: substitute cross-attention
  - Mixture of experts *STL*: to better capture global properties in time series

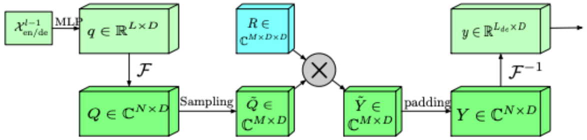


Figure 3. Frequency Enhanced Block with Fourier transform (FEB-f) structure.

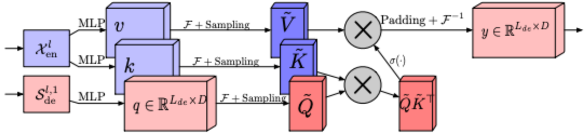


Figure 4. Frequency Enhanced Attention with Fourier transform (FEA-f) structure, $\sigma(\cdot)$ is the activation function.

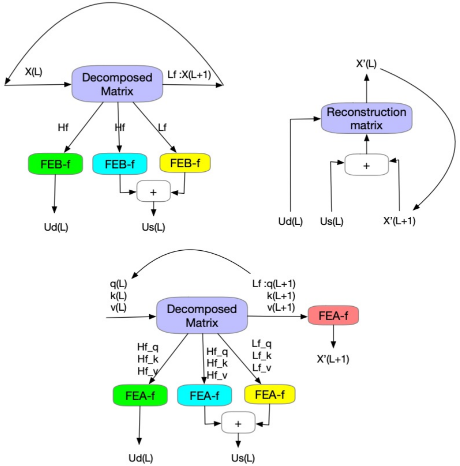


Figure 5. Top Left: Wavelet frequency enhanced block decomposition stage. Top Right: Wavelet block reconstruction stage shared by FEB-w and FEA-w. Bottom: Wavelet frequency enhanced cross attention decomposition stage.

MOE Seasonal-Trend Decomposition

$$\mathbf{X_{trend}} = \mathbf{Softmax}(L(x)) * (F(x)), \tag{10}$$

where $F(\cdot)$ is a set of average pooling filters and $\text{Softmax}(L(x))$ is the weights for mixing these extracted trends.

Forecasting with Transformer: FEDformer

Empirical comparison of FEDformer on six benchmark datasets

Table 2. Multivariate long-term series forecasting results on six datasets with input length $I = 96$ and prediction length $O \in \{96, 192, 336, 720\}$ (For ILI dataset, we use input length $I = 36$ and prediction length $O \in \{24, 36, 48, 60\}$). A lower MSE indicates better performance, and the best results are highlighted in bold.

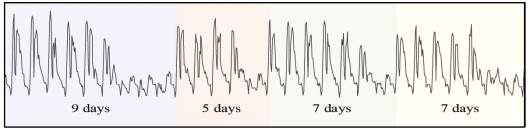
Methods	Metric	ETTm2				Electricity				Exchange				Traffic				Weather				ILI			
		96	192	336	720	96	192	336	720	96	192	336	720	96	192	336	720	96	192	336	720	24	36	48	60
FEDformer-f	MSE	**0.203**	**0.269**	**0.325**	**0.421**	0.193	0.201	0.214	0.246	0.148	0.271	0.460	1.195	0.587	0.604	0.621	0.626	**0.217**	**0.276**	**0.339**	**0.403**	3.228	2.679	2.622	2.857
	MAE	**0.287**	**0.328**	**0.366**	**0.415**	0.308	0.315	0.329	0.355	0.278	0.380	0.500	0.841	0.366	0.373	0.383	0.382	**0.296**	**0.336**	**0.380**	**0.428**	1.260	1.080	1.078	1.157
FEDformer-w	MSE	0.204	0.316	0.359	0.433	**0.183**	**0.195**	**0.212**	**0.231**	**0.139**	**0.256**	**0.426**	**1.090**	**0.562**	**0.562**	**0.570**	**0.596**	0.227	0.295	0.381	0.424	**2.203**	**2.272**	**2.209**	**2.545**
	MAE	0.288	0.363	0.387	0.432	**0.297**	**0.308**	**0.313**	**0.343**	**0.276**	**0.369**	**0.464**	**0.800**	**0.349**	**0.346**	**0.323**	**0.368**	0.304	0.363	0.416	0.434	**0.963**	**0.976**	**0.981**	**1.061**
Autoformer	MSE	0.255	0.281	0.339	0.422	0.201	0.222	0.231	0.254	0.197	0.300	0.509	1.447	0.613	0.616	0.622	0.660	0.266	0.307	0.359	0.419	3.483	3.103	2.669	2.770
	MAE	0.339	0.340	0.372	0.419	0.317	0.334	0.338	0.361	0.323	0.369	0.524	0.941	0.388	0.382	0.337	0.408	0.336	0.367	0.395	0.428	1.287	1.148	1.085	1.125
Informer	MSE	0.365	0.533	1.363	3.379	0.274	0.296	0.300	0.373	0.847	1.204	1.672	2.478	0.719	0.696	0.777	0.864	0.300	0.598	0.578	1.059	5.764	4.755	4.763	5.264
	MAE	0.453	0.563	0.887	1.338	0.368	0.386	0.394	0.439	0.752	0.895	1.036	1.310	0.391	0.379	0.420	0.472	0.384	0.544	0.523	0.741	1.677	1.467	1.469	1.564
LogTrans	MSE	0.768	0.989	1.334	3.048	0.258	0.266	0.280	0.283	0.968	1.040	1.659	1.941	0.684	0.685	0.7337	0.717	0.458	0.658	0.797	0.869	4.480	4.799	4.800	5.278
	MAE	0.642	0.757	0.872	1.328	0.357	0.368	0.380	0.376	0.812	0.851	1.081	1.127	0.384	0.390	0.408	0.396	0.490	0.589	0.652	0.675	1.444	1.467	1.468	1.560
Reformer	MSE	0.658	1.078	1.549	2.631	0.312	0.348	0.350	0.340	1.065	1.188	1.357	1.510	0.732	0.733	0.742	0.755	0.689	0.752	0.639	1.130	4.400	4.783	4.832	4.882
	MAE	0.619	0.827	0.972	1.242	0.402	0.433	0.433	0.420	0.829	0.906	0.976	1.016	0.423	0.420	0.420	423	0.596	0.638	0.596	0.792	1.382	1.448	1.465	1.483

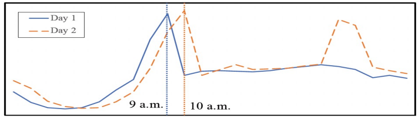
Linear complexity of FEDformer

Table 1. Complexity analysis of different forecasting models.

Methods	Training		Testing
	Time	Memory	Steps
FEDformer	$\mathcal{O}(L)$	$\mathcal{O}(L)$	1
Autoformer	$\mathcal{O}(L \log L)$	$\mathcal{O}(L \log L)$	1
Informer	$\mathcal{O}(L \log L)$	$\mathcal{O}(L \log L)$	1
Transformer	$\mathcal{O}(L^2)$	$\mathcal{O}(L^2)$	L
LogTrans	$\mathcal{O}(L \log L)$	$\mathcal{O}(L^2)$	1
Reformer	$\mathcal{O}(L \log L)$	$\mathcal{O}(L \log L)$	L
LSTM	$\mathcal{O}(L)$	$\mathcal{O}(L)$	L

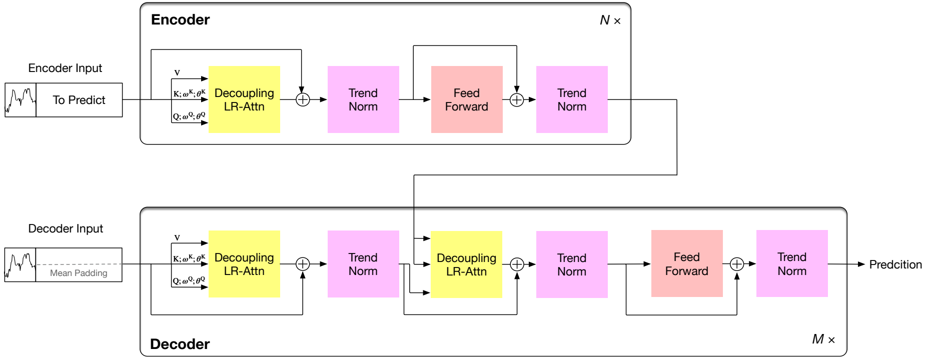
Forecasting with Transformer: Quatformer



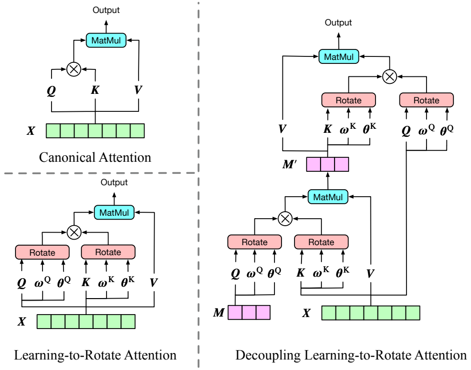


Multiple/variable periods and phase shift

- Quatformer: Transformer with quaternions for periodic time series
  - Learning-to-rotate attention (LR-Attn): modeling multiple *periods and periodic changes* by quaternions
  - Trend normalization: modeling slowly varying trend



Quatformer architecture



Learning-to-rotate attention

$$\frac{\gamma}{\sigma} \odot (\mathcal{X} - \mathrm{MovingAvg}(\mathcal{X})) + \mathcal{T},$$

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\mathcal{X}_i - \mu)^2}, \quad \mu = \frac{1}{N}\sum_{i=1}^{N}\mathcal{X}_i,$$

$$\mathcal{T} = \sum_{i=0}^{p} \boldsymbol{\beta}_i \mathrm{pos}^i, \quad \mathrm{pos} = [0, 1, 2, \cdots, N-1]^\top / N.$$

Trend normalization

Weiqi Chen, Wenwei Wang, Bingqing Peng, Qingsong Wen, Tian Zhou, Liang Sun, "Learning to Rotate: Quaternion Transformer for Complicated Periodical Time Series Forecasting", KDD 2022.

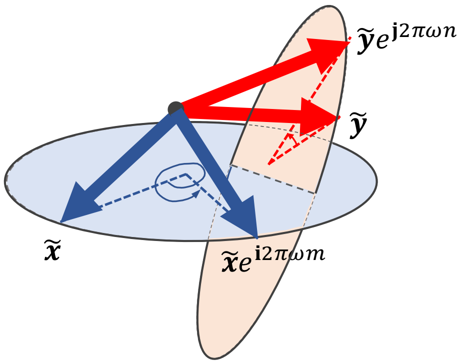
Forecasting with Transformer: Quatformer

- Quatformer: LR Attention

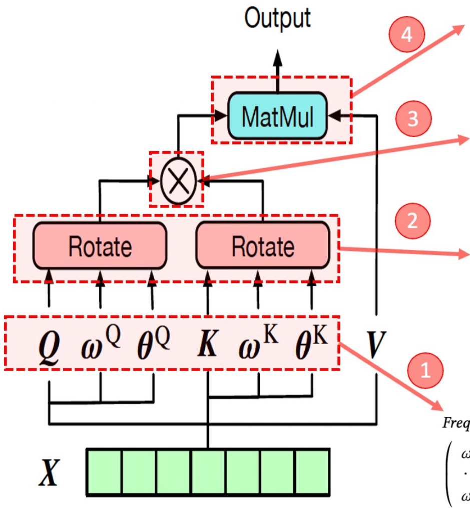
Rotatory Softmax-kernel with Quaternions

$$\mathrm{SM}^{\mathrm{rot}}(\phi(\boldsymbol{x}, m), \psi(\boldsymbol{y}, n)) = \exp(\mathrm{Re}[\phi(\boldsymbol{x}, m)^{\mathsf{H}} \psi(\boldsymbol{y}, n)]),$$

$$\phi(\boldsymbol{x}, m) = \tilde{\boldsymbol{x}} e^{\mathbf{i} 2\pi\omega m}, \quad \psi(\boldsymbol{y}, n) = \tilde{\boldsymbol{y}} e^{\mathbf{j} 2\pi\omega n},$$



Learning-to-Rotate Attention Structure



Series-Aggregation:

$$\mathcal{H} = \boldsymbol{S}\boldsymbol{V}.$$

Series-Attention with rotatory softmax-kernel :

$$\boldsymbol{S} = \mathrm{softmax}\left(\frac{1}{P\sqrt{d}} \sum_{p=1}^{P} \mathrm{Re}[\Phi_p(\boldsymbol{Q}, \mathrm{pos}^{\mathrm{Q}}) \Psi_p(\boldsymbol{K}, \mathrm{pos}^{\mathrm{K}})^{\mathsf{H}}]\right),$$

Series-Rotation:

$$\Phi_p(\boldsymbol{Q}, \mathrm{pos}^{\mathrm{Q}}) = \tilde{\boldsymbol{Q}} e^{\mathbf{i}(2\pi\omega_p^{\mathrm{Q}} \mathrm{pos}^{\mathrm{Q}} + \theta_p^{\mathrm{Q}})}, \quad p = 1, 2, \cdots, P$$

$$\Psi_p(\boldsymbol{K}, \mathrm{pos}^{\mathrm{K}}) = \tilde{\boldsymbol{K}} e^{\mathbf{j}(2\pi\omega_p^{\mathrm{K}} \mathrm{pos}^{\mathrm{K}} + \theta_p^{\mathrm{K}})}, \quad p = 1, 2, \cdots, P$$

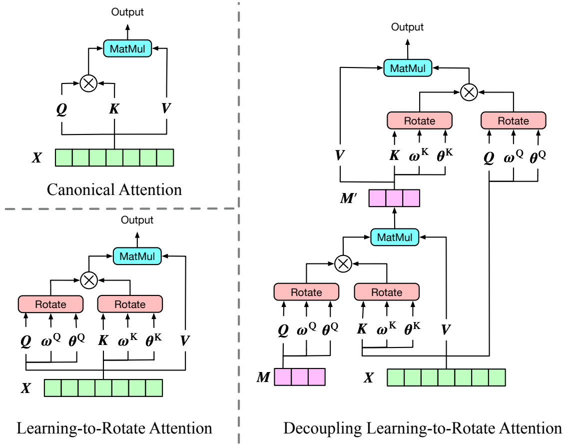
Frequency/phase-Generation:

$$\begin{pmatrix} \omega_1^{\mathrm{Q}} \\ \cdots \\ \omega_P^{\mathrm{Q}} \end{pmatrix}, \begin{pmatrix} \theta_1^{\mathrm{Q}} \\ \cdots \\ \theta_P^{\mathrm{Q}} \end{pmatrix} = \mathrm{Conv}(\boldsymbol{Q}; W_\omega^{\mathrm{Q}}), \mathrm{Conv}(\boldsymbol{Q}; W_\theta^{\mathrm{Q}}), \quad \begin{pmatrix} \omega_1^{\mathrm{K}} \\ \cdots \\ \omega_P^{\mathrm{K}} \end{pmatrix}, \begin{pmatrix} \theta_1^{\mathrm{K}} \\ \cdots \\ \theta_P^{\mathrm{K}} \end{pmatrix} = \mathrm{Conv}(\boldsymbol{K}; W_\omega^{\mathrm{K}}), \mathrm{Conv}(\boldsymbol{K}; W_\theta^{\mathrm{K}}),$$

(Execute multiple times representing multiple periods)

Forecasting with Transformer: Quatformer

- Quatformer: Decoupling LR-Attn, Performance
  - LR-Attn's complexity is $O(N^2)$
  - Decoupling LR-Attn introduces a momentum-updated c-length latent series $\mathcal{M} \in \mathbb{R}^{c \times d}$, and decouple $\mathcal{H} = \text{LR-Attn}(\mathcal{X}, \mathcal{Y})$ into



$$\mathcal{H} = \text{LR-Attn}(\mathcal{X}, \mathcal{M}') \in \mathbb{R}^{N \times d},$$
$$\mathcal{M}' = \text{LR-Attn}(\mathcal{M}, \mathcal{Y}) \in \mathbb{R}^{c \times d}.$$

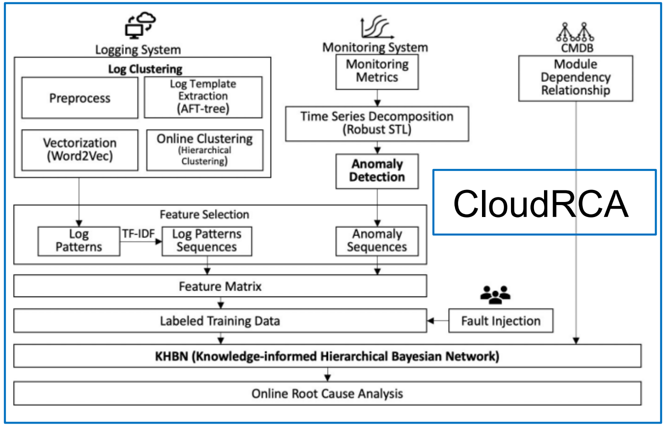
Complexity is decreased to $O(2cN)$.

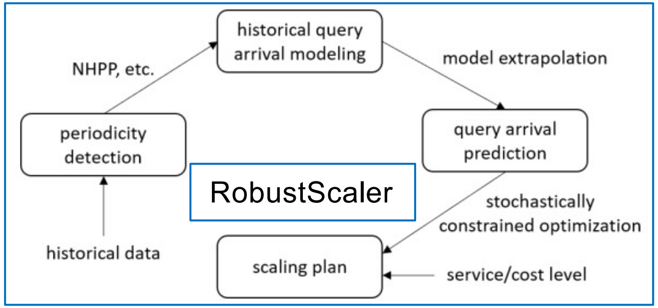
Performance of Quatformer

Models		Quatformer		Quatformer†		Autoformer	
		MSE	MAE	MSE	MAE	MSE	MAE
ETTh	96	**0.403**	**0.434**	0.426	0.450	0.442	0.451
	192	**0.444**	**0.463**	0.453	0.466	0.500	0.482
	336	**0.452**	**0.455**	0.464	0.474	0.512	0.492
	720	0.477	0.490	**0.474**	**0.487**	0.514	0.512
ETTm	96	0.220	0.301	**0.217**	**0.297**	0.247	0.325
	192	0.279	0.333	**0.269**	**0.329**	0.278	0.335
	336	0.331	**0.354**	**0.330**	0.366	0.336	0.370
	720	**0.422**	**0.413**	0.433	0.428	0.439	0.435
Weather	96	**0.211**	**0.279**	0.213	0.287	0.259	0.332
	192	**0.263**	**0.325**	0.265	0.326	0.300	0.359
	336	**0.310**	**0.344**	0.315	0.354	0.364	0.401
	720	**0.381**	**0.374**	0.382	0.378	0.439	0.440
Exchange	96	**0.147**	**0.274**	0.148	0.276	0.154	0.284
	192	**0.254**	**0.364**	0.255	0.365	0.272	0.381
	336	0.427	0.481	**0.425**	**0.480**	0.461	0.509
	720	**0.974**	**0.751**	1.095	0.800	1.100	0.813
Traffic	96	0.618	**0.384**	**0.617**	0.387	0.636	0.397
	192	0.619	0.384	**0.600**	**0.367**	0.618	0.381
	336	0.622	**0.384**	**0.618**	0.385	0.626	0.388
	720	0.629	0.383	**0.616**	**0.379**	0.653	0.400
Electricity	96	**0.197**	**0.308**	0.200	0.311	0.203	0.318
	192	**0.205**	**0.302**	0.216	0.330	0.233	0.338
	336	**0.220**	**0.329**	0.228	0.343	0.259	0.359
	720	0.245	0.350	**0.242**	**0.349**	0.255	0.361

Other Time Series Applications

- Fault Cause Localization
  - *From Anomaly Detection to Localization*
  - CloudRCA[1], NetRCA[2], Rule Set[3]
- Autoscaling
  - *from Forecasting to Decision-Making*
  - RobustScalar[4], AHPA[5]
- More…





[1] Yingying Zhang, Zhengxiong Guan, Huajie Qian, Leili Xu, Hengbo Liu, Qingsong Wen, Liang Sun, Junwei Jiang, Lunting Fan, Min Ke, "CloudRCA: A Root Cause Analysis Framework for Cloud Computing Platforms," CIKM 2021.

[2] Chaoli Zhang*, Zhiqiang Zhou*, Yingying Zhang*, Linxiao Yang*, Kai He*, Qingsong Wen*, Liang Sun* (*Equally Contributed), "NetRCA: An Effective Network Fault Cause Localization Algorithm," ICASSP 2022. **(ICASSP'22 AIOps Challenge, First Place (1/382))**

[3] Fan Yang, Kai He, Linxiao Yang, Hongxia Du, Jingbang Yang, Bo Yang, Liang Sun, "Learning Interpretable Decision Rule Sets: A Submodular Optimization Approach," **NeurIPS 2021 Spotlight**.

[4] Huajie Qian, Qingsong Wen, Liang Sun, Jing Gu, Qiulin Niu, Zhimin Tang, "RobustScaler: QoS-Aware Autoscaling for Complex Workloads," ICDE 2022.

[5] Zhiqiang Zhou, Chaoli Zhang, Lingna Ma, Jing Gu, Huajie Qian, Qingsong Wen, Liang Sun, Peng Li, Zhimin Tang, "AHPA: Adaptive Horizontal Pod Autoscaling Systems on Alibaba Cloud Container Service for Kubernetes," AAAI 2023. **(AAAI/IAAI 2023 Innovative Application Award)**

Talk Summary

- ❑ **Introduction:** Challenges and Needs for Robust and Intelligent Time Series Analysis
- ❑ **Robust Time Series Processing Blocks**
  - Time Series Periodicity Detection
  - Time Series Trend Filtering
  - Time Series Seasonal-Trend Decomposition
- ❑ **Robust Time Series Applications**
  - Time Series Anomaly Detection
  - Time Series Forecasting
  - Other Time Series Applications

Related Resource

➢ Source Code

- Time Series Forecasting
  - FEDformer(ICML'22): https://github.com/DAMO-DI-ML/ICML2022-FEDformer
  - Quatformer(KDD'22): https://github.com/DAMO-DI-ML/KDD2022-Quatformer
  - FiLM (NeurIPS'22): https://github.com/DAMO-DI-ML/NeurIPS2022-FiLM
- Time Series Anomaly Detection
  - TFAD (CIKM'22): https://github.com/DAMO-DI-ML/CIKM22-TFAD
- More
  - Group's Open Source: https://github.com/DAMO-DI-ML

➢ More Related Time Series Talk

- Tutorial @ KDD'22, "**Robust Time Series Analysis and Applications: An Industrial Perspective**"
  - https://qingsongedu.github.io/timeseries-tutorial-kdd-2022/
- Keynote @ CIKM'22 Time Series Workshop, "**Customized Transformers for Time Series Forecasting**"
  - https://drive.google.com/file/d/1Bg-NLscGfs5Cqw8q93FcysoGdj6uxgSf/view?usp=sharing

Thanks!
Q&A

My Homepage: https://sites.google.com/site/qingsongwen8/
Alibaba DAMO DI Lab: https://damo.alibaba.com/labs/decision-intelligence

Hiring: AI for Time Series, XAI

Research Interns
Postdocs
Applied/Research Scientists

Seattle (US), Hangzhou (China)

Email CV to

qingsongedu@gmail.com or qingsong.wen@alibaba-inc.com

References (in this talk)

1. Qingsong Wen, Kai He, Liang Sun, Yingying Zhang, Min Ke, Huan Xu, "RobustPeriod: Time-Frequency Mining for Robust Multiple Periodicity Detection," in Proc. ACM SIGMOD International Conference on Management of Data (SIGMOD 2021), Xi'an, China, Jun. 2021.
2. Qingsong Wen, Jingkun Gao, Xiaomin Song, Liang Sun, Jian Tan, "RobustTrend: A Huber Loss with a Combined First and Second Order Difference Regularization for Time Series Trend Filtering," in Proc. 28th International Joint Conference on Artificial Intelligence (IJCAI 2019), Macao, China, Aug. 2019.
3. Qingsong Wen, Jingkun Gao, Xiaomin Song, Liang Sun, Huan Xu, Shenghuo Zhu, "RobustSTL: A Robust Seasonal-Trend Decomposition Algorithm for Long Time Series," in Proc. 33th AAAI Conference on Artificial Intelligence (AAAI 2019), Honolulu, Hawaii, Jan. 2019.
4. Chaoli Zhang*, Zhiqiang Zhou*, Yingying Zhang*, Linxiao Yang*, Kai He*, Qingsong Wen*, Liang Sun* (*Equally Contributed), "NetRCA: An Effective Network Fault Cause Localization Algorithm," ICASSP 2022. **(ICASSP'22 AIOps Challenge, First Place (1/382))**

References (in this talk)

5. Linxiao Yang, Qingsong Wen, Bo Yang, Liang Sun, "A Robust and Efficient Multi-Scale Seasonal-Trend Decomposition," in Proc. IEEE 46th International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2021), Toronto, Canada, Jun. 2021.
6. Fan Yang, Kai He, Linxiao Yang, Hongxia Du, Jingbang Yang, Bo Yang, Liang Sun, "Learning Interpretable Decision Rule Sets: A Submodular Optimization Approach," **NeurIPS 2021 Spotlight**.
7. Jingkun Gao, Xiaomin Song, Qingsong Wen, Pichao Wang, Liang Sun, Huan Xu, "RobustTAD: Robust Time Series Anomaly Detection via Decomposition and Convolutional Neural Networks," in SIGKDD Workshop on Mining and Learning from Time Series (KDD-MiLeTS 2020), San Diego, CA, Aug. 2020.
8. Chaoli Zhang, Tian Zhou, Qingsong Wen, Liang Sun, "TFAD: A Decomposition Time Series Anomaly Detection Architecture with Time-Freq Analysis," in Proc. 31st ACM International Conference on Information and Knowledge Management (CIKM 2022), Atlanta, GA, Oct. 2022.
9. Yingying Zhang, Zhengxiong Guan, Huajie Qian, Leili Xu, Hengbo Liu, Qingsong Wen, Liang Sun, Junwei Jiang, Lunting Fan, Min Ke, "CloudRCA: A Root Cause Analysis Framework for Cloud Computing Platforms," in Proc. 30th ACM International Conference on Information and Knowledge Management **(CIKM 2021 Oral)**, Queensland, Australia, Nov. 2021.

References (in this talk)

9. Qingsong Wen, Zhe Zhang, Yan Li, Liang Sun, "Fast RobustSTL: Efficient and Robust Seasonal-Trend Decomposition for Time Series with Complex Patterns," in Proc. 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD 2020), San Diego, CA, Aug. 2020.
10. Tian Zhou, Ziqing Ma, Qingsong Wen, Xue Wang, Liang Sun, Rong Jin, "FEDformer: Frequency Enhanced Decomposed Transformer for Long-term Series Forecasting," ICML 2022.
11. Weiqi Chen, Wenwei Wang, Bingqing Peng, Qingsong Wen, Tian Zhou, Liang Sun, "Learning to Rotate: Quaternion Transformer for Complicated Periodical Time Series Forecasting", KDD 2022.
12. Huajie Qian, Qingsong Wen, Liang Sun, Jing Gu, Qiulin Niu, Zhimin Tang, "RobustScaler: QoS-Aware Autoscaling for Complex Workloads," ICDE 2022.
13. Zhiqiang Zhou, Chaoli Zhang, Lingna Ma, Jing Gu, Huajie Qian, Qingsong Wen, Liang Sun, Peng Li, Zhimin Tang, "AHPA: Adaptive Horizontal Pod Autoscaling Systems on Alibaba Cloud Container Service for Kubernetes," AAAI 2023. **(AAAI/IAAI 2023 Innovative Application Award)**

References (not included in this talk)

15. Qingsong Wen, Liang Sun, Fan Yang, Xiaomin Song, Jingkun Gao, Xue Wang, Huan Xu, "Time Series Data Augmentation for Deep Learning: A Survey," in the 30th International Joint Conference on Artificial Intelligence (IJCAI 2021), Montreal, Canada, Aug. 2021. **Selected by Paper Digest into Most Influential IJCAI Papers (Version: 2022-02), Rank 1st (1/600+ IJCAI'21 papers).**
16. Qingsong Wen, Linxiao Yang, Tian Zhou, Liang Sun, "Robust Time Series Analysis and Applications: An Industrial Perspective", in Proc. 28th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD 2022), Washington DC, Aug. 2022.
17. Qingsong Wen, Zhengzhi Ma, Liang Sun, "On Robust Variance Filtering and Change of Variance Detection," in Proc. IEEE 45th International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2020), Barcelona, Spain, May 2020.
18. Qingyang Xu, Qingsong Wen, Liang Sun, "Two-Stage Framework for Seasonal Time Series Forecasting," in Proc. IEEE 46th International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2021), Toronto, Canada, Jun. 2021.
19. Xiaomin Song, Qingsong Wen, and Liang Sun, "Robust Time Series Dissimilarity Measure for Outlier Detection and Periodicity Detection," in Proc. 31st ACM International Conference on Information and Knowledge Management (CIKM 2022), Atlanta, GA, Oct. 2022.

References (not included in this talk)

20. Tian Zhou, Ziqing Ma, Xue Wang, Qingsong Wen, Liang Sun, Tao Yao, Wotao Yin, Rong Jin, “FiLM: Frequency improved Legendre Memory Model for Long-term Time Series Forecasting”, in Proc. 36th Conf. on Neural Information Processing Systems **(NeurIPS 2022 Oral)**, New Orleans, LA, Dec. 2022.
21. Chenxiao Yang, Qitian Wu, Qingsong Wen, Zhiqiang Zhou, Liang Sun, Junchi Yan, "Towards Out-of-Distribution Sequential Event Prediction: A Causal Treatment," in Proc. 36th Annual Conference on Neural Information Processing Systems (NeurIPS 2022), New Orleans, LA, Dec. 2022.
22. Kexin Zhang, Qingsong Wen, Chaoli Zhang, Liang Sun, Yong Liu, "Time Series Anomaly Detection using Skip-Step Contrastive Predictive Coding," in NeurIPS 2022 Workshop on Self-Supervised Learning - Theory and Practice, New Orleans, LA, Dec. 2022.
23. Peisong_Niu, Tian Zhou, Qingsong Wen, Liang Sun, Tao Yao, "Chemistry Guided Molecular Graph Transformer," in NeurIPS 2022 Workshop on AI for Science: Progress and Promises, New Orleans, LA, Dec. 2022.
24. Yi Wang, Chien-fei Chen, Peng-Yong Kong, Husheng Li, and Qingsong Wen, “A Cyber-Physical-Social Perspective on Future Smart Distribution Systems,” **Proceedings of the IEEE** (PIEEE), 2022.
25. Longyuan Li, Junchi Yan, Qingsong Wen, Yaohui Jin, and Xiaokang Yang, "Learning Robust Deep State Space for Unsupervised Anomaly Detection in Contaminated Time-Series," IEEE Transactions on Knowledge and Data Engineering (TKDE), 2022.