CATCH: Channel-Aware Multivariate Time Series Anomaly Detection via Frequency Patching

汇报人：吴行健

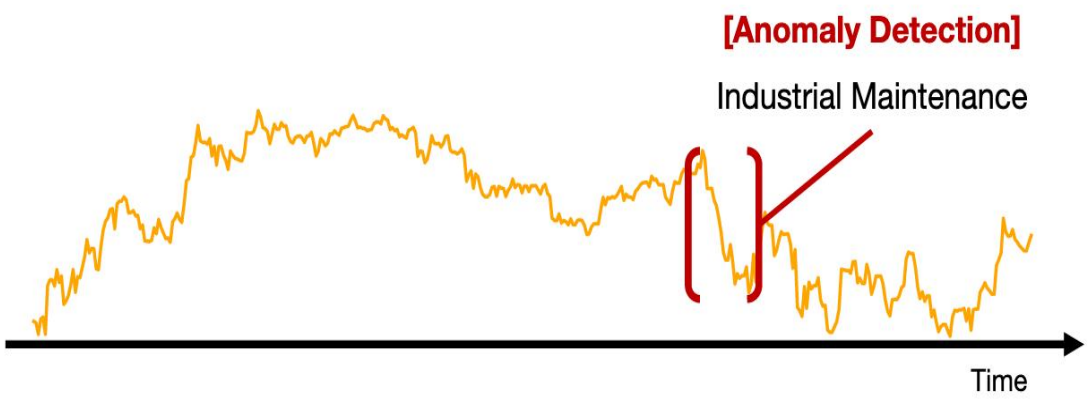
华东师范大学在读博士

时间序列异常检测介绍

华東師範大學 EAST CHINA NORMAL UNIVERSITY

- 时间序列数据指按时间顺序记录的数据，**遍布日常生活和工业生产的方方面面**

In the context of time series anomaly detection, $\mathrm{X} \in \mathbb{R}^{N \times T}$ denotes a time series with N channels and length T. For clear delineation, we separate dimensions with commas and use this format throughout this paper. For example, we denote $\mathrm{X}_{i,j}$ as the i-th channel at the j-th timestamp, $\mathrm{X}_{n,:} \in \mathbb{R}^{T}$ as the time series of n-th channel, where $n = 1, 2, \cdots, N$. The multivariate time series anomaly detection problem is to determine whether $\mathrm{X}_{:,t}$ is anomaly or not.



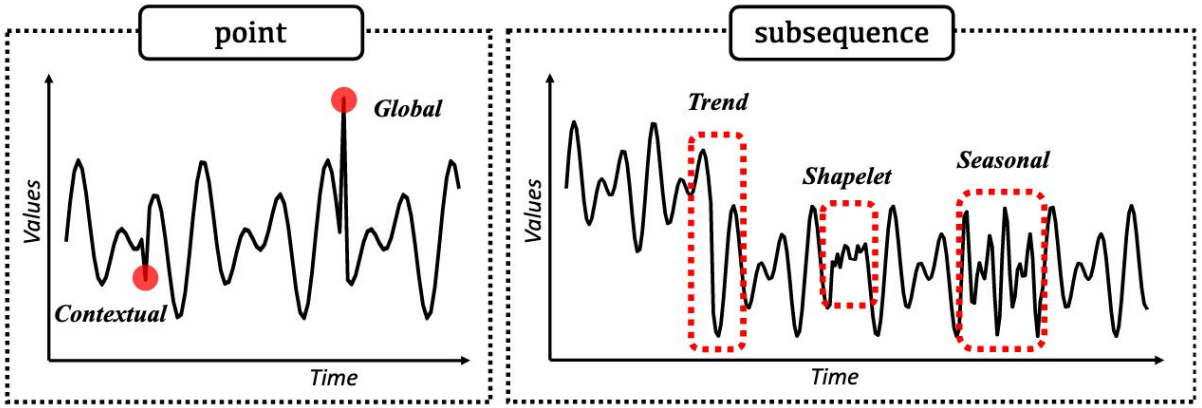
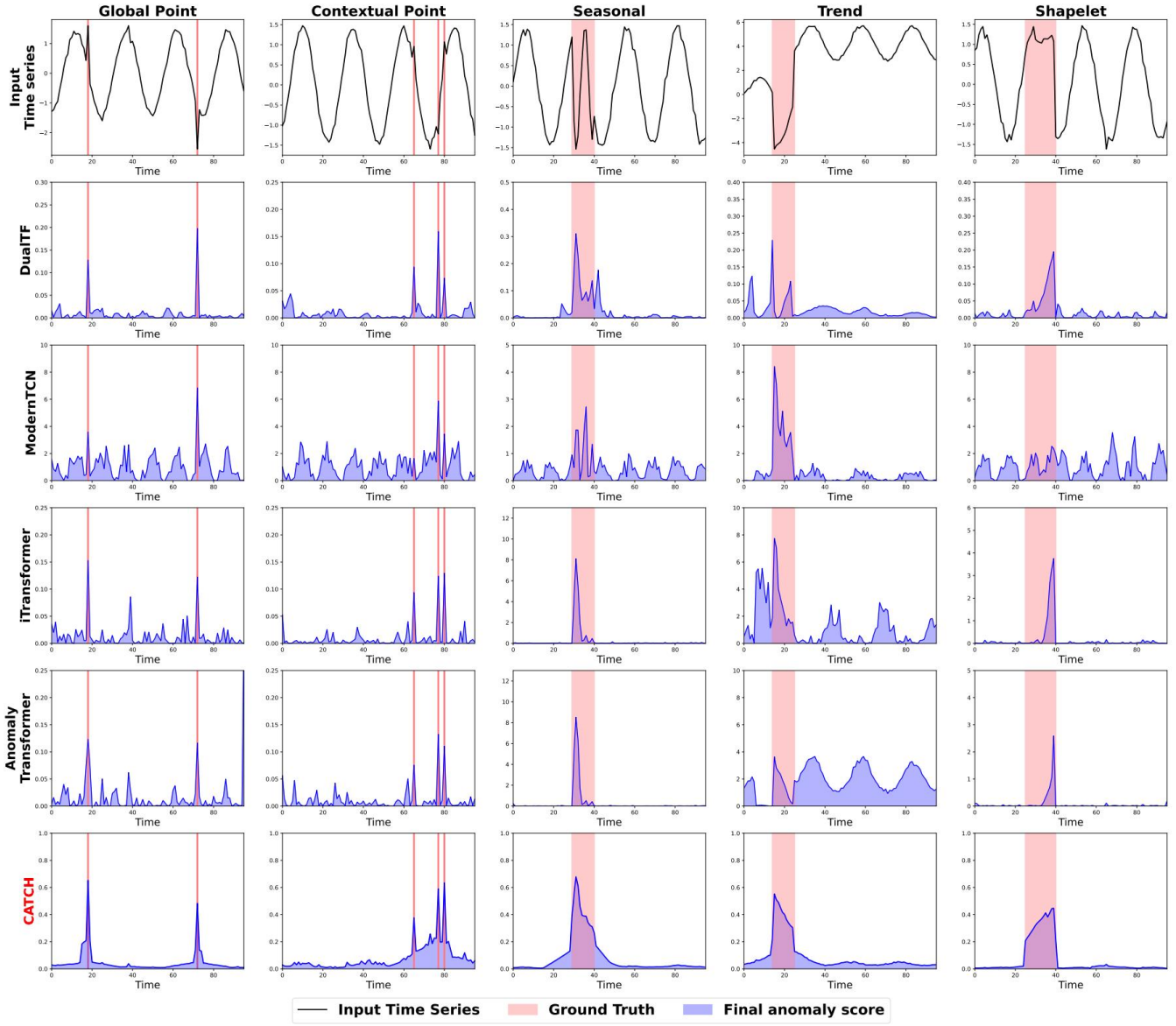


Figure 1: Types of time series anomalies.

时间序列异常检测介绍

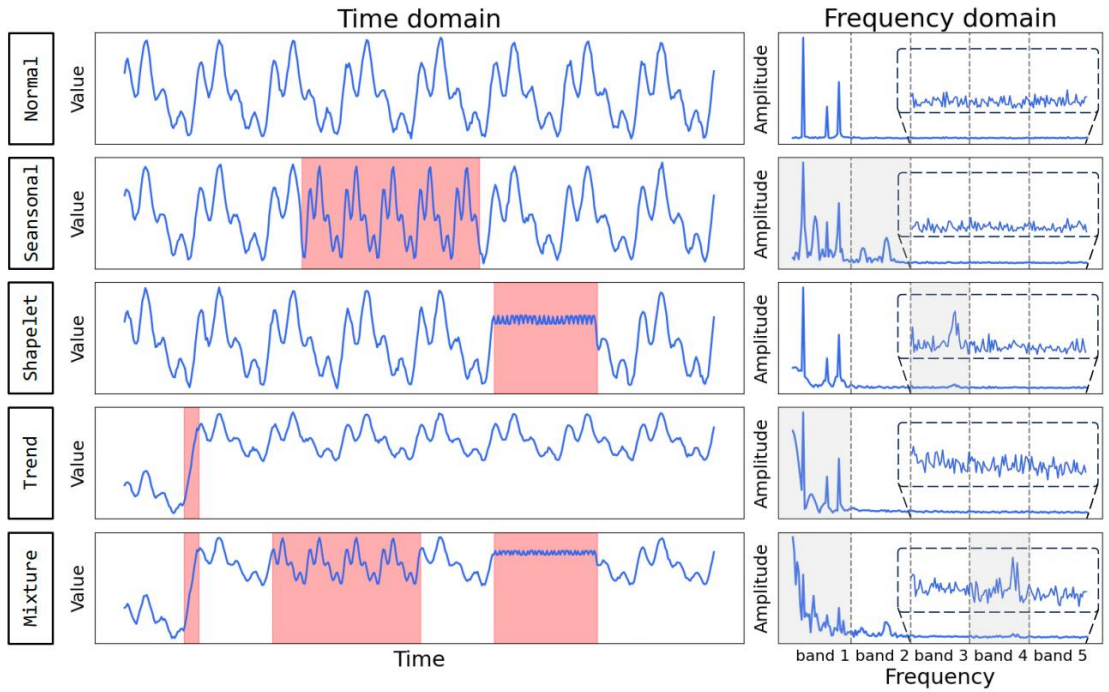

Global Point
Contextual Point
Seasonal
Trend
Shapelet
Input Time series
DualTF
ModernTCN
iTransformer
Anomaly Transformer
CATCH
Time
Input Time Series
Ground Truth
Final anomaly score

问题与贡献点

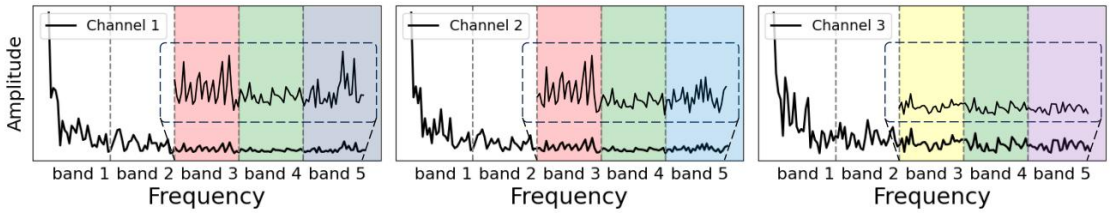
- 问题
  - 如何建模子序列异常？ ---------> **频域建模**
  - 如何同时建模子序列以及点异常？ ---------> **时频对齐**
  - 如何克服基于重建的方法倾向于建模低频成分？ ---------> **频域划分Patch**

- 贡献点
  - 我们提出了一个名为 CATCH 的通用框架，该框架通过**频域补丁学习**，实现了对异质点异常和子序列异常的同时检测。该框架通过频域补丁增强了子序列异常检测能力，并在各频带间集成了细粒度的自适应通道相关性。
  - 我们设计了 CFM（通道相关性模块）以**充分利用细粒度的通道相关性**。通过双层多目标优化算法驱动，CFM 能够迭代发现适当的通道相关性，隔离无关通道并聚合相关通道，从而提供更高的能力和鲁棒性
  - 我们在 23 个多变量数据集上进行了广泛的实验。结果表明，CATCH 的**性能优于当前最先进的基线方法**

Motivation



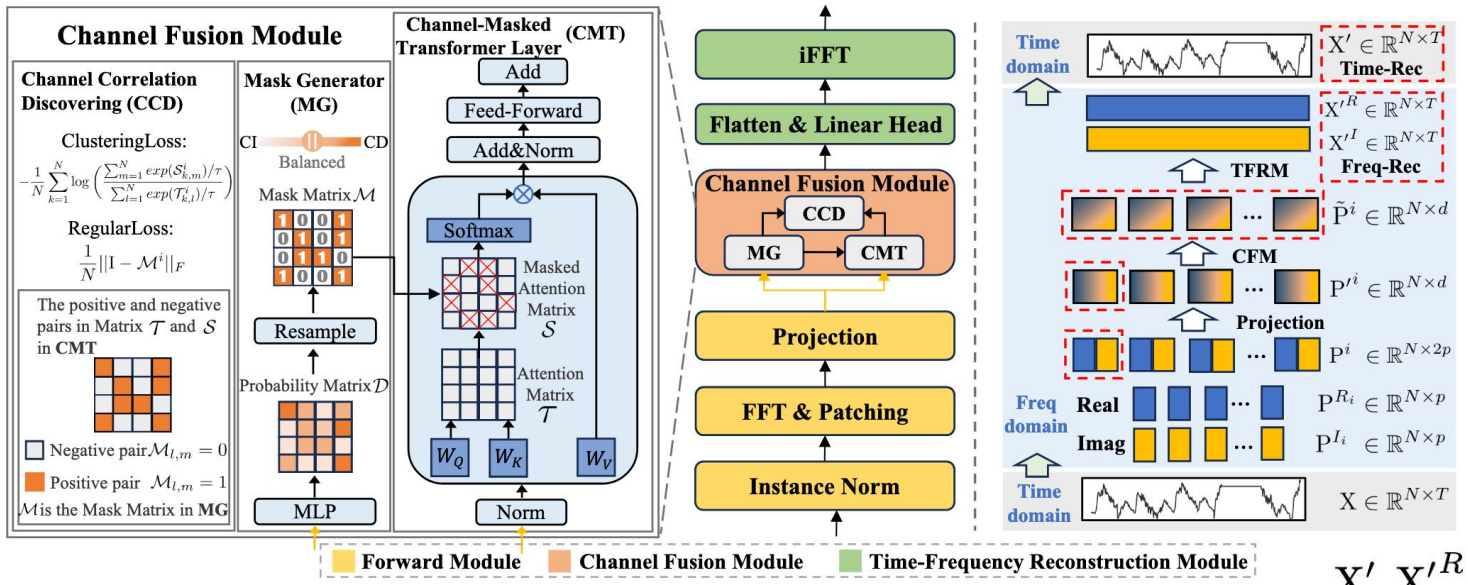
(a) Different subsequence anomalies.



(b) Varying correlations among channels.

- 频域被划分为五个频带，并对高频带进行放大以便更清晰地观察。
- （a）展示了**一个正常时间序列以及在注入季节性、形状子序列、趋势和混合子序列异常后的变化**。在时间域中，异常部分用红色标注，而在频域中受影响的频带用灰色突出显示。
- （b）展示了一个具有三个通道的多变量时间序列的频带。这些通道在**不同频带上表现出不同的相关性**
  - 通道 1 和通道 2 在第三频带中表现出相似的行为，因此用红色标记，而通道 3 表现出不同的特性，用黄色标记。
  - 在第四频带中，所有通道表现出相似的行为，用绿色标记，
  - 在第五频带中，它们表现出不同的特性，因此用不同的颜色标记。

代码架构——CATCH



$$X^R, X^I = \text{FFT}(X)$$

$$\{P^{R_1}, P^{R_2}, \cdots, P^{R_L}\} = \text{Patching}(X^R)$$

$$\{P^{I_1}, P^{I_2}, \cdots, P^{I_L}\} = \text{Patching}(X^I)$$

concat each pair of P^{R_i} and P^{I_i} into $P^i \in \mathbb{R}^{N\times 2p}$

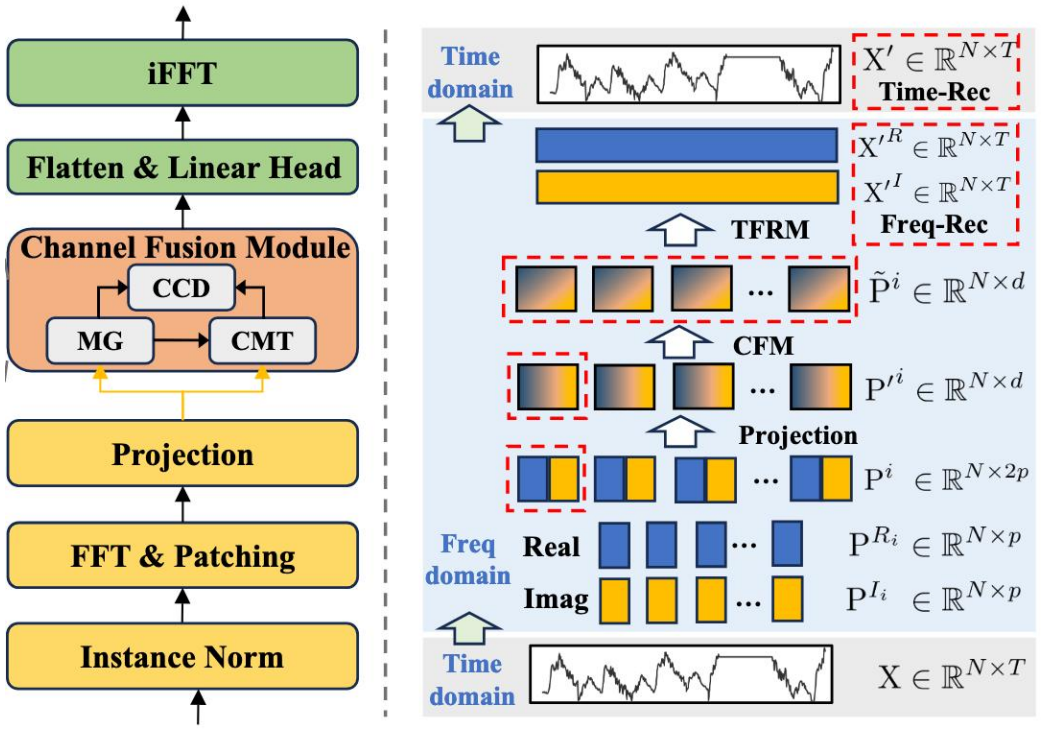
$$P'^i = \text{Projection}(P^i)$$

$$\{\tilde{P}^1, \tilde{P}^2, \cdots, \tilde{P}^L\} = \text{CFM}(\{P'^1, P'^2, \cdots, P'^L\})$$

$$X', X'^R, X'^I = \text{TFRM}(\{\tilde{P}^{R_1}, \tilde{P}^{R_2}, \cdots, \tilde{P}^{R_L}\}, \{\tilde{P}^{I_1}, \tilde{P}^{I_2}, \cdots, \tilde{P}^{I_L}\})$$

Figure 2: CATCH architecture. (1) Forward Module normalizes the input data, patchifies the frequency domain, and then projects it into the hidden space. (2) Channel Fusion Module captures channel interrelationships in each frequency band with a Channel-Masked Transformer (CMT) Layer, where the mask matrix (channel correlation) is generated by Mask Generator (MG). During training, MG and CMT are optimized by Channel Correlation Discovering (CCD) for more appropriate channel correlations. (3) Time-Frequency Reconstruction Module obtains the frequency reconstruction through Flatten & Linear Head Layer, and obtains the time reconstruction after iFFT.

代码架构——Forward Module



$$X^R, X^I = \mathrm{FFT}(X)$$

$$\{P^{R_1}, P^{R_2}, \cdots, P^{R_L}\} = \mathrm{Patching}(X^R)$$

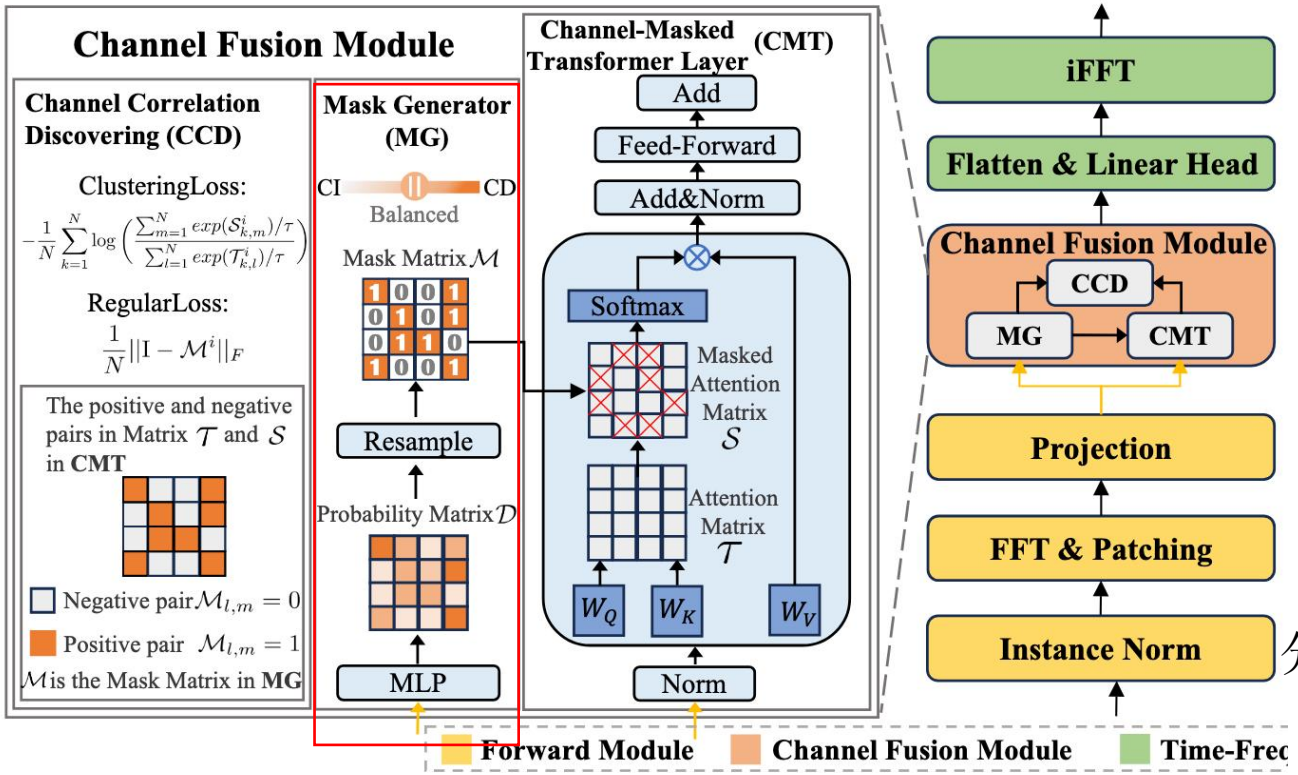
$$\{P^{I_1}, P^{I_2}, \cdots, P^{I_L}\} = \mathrm{Patching}(X^I)$$

concat each pair of P^{R_i} and P^{I_i} into $P^i \in \mathbb{R}^{N \times 2p}$

$$P'^i = \mathrm{Projection}(P^i)$$

$L = [T - p]/s + 1$ is the total patch number, where p is the patch size and s is the patch stride

代码架构——Channel Fusion Module



$$\mathrm{P}'^{i} \in \mathbb{R}^{N\times d}$$

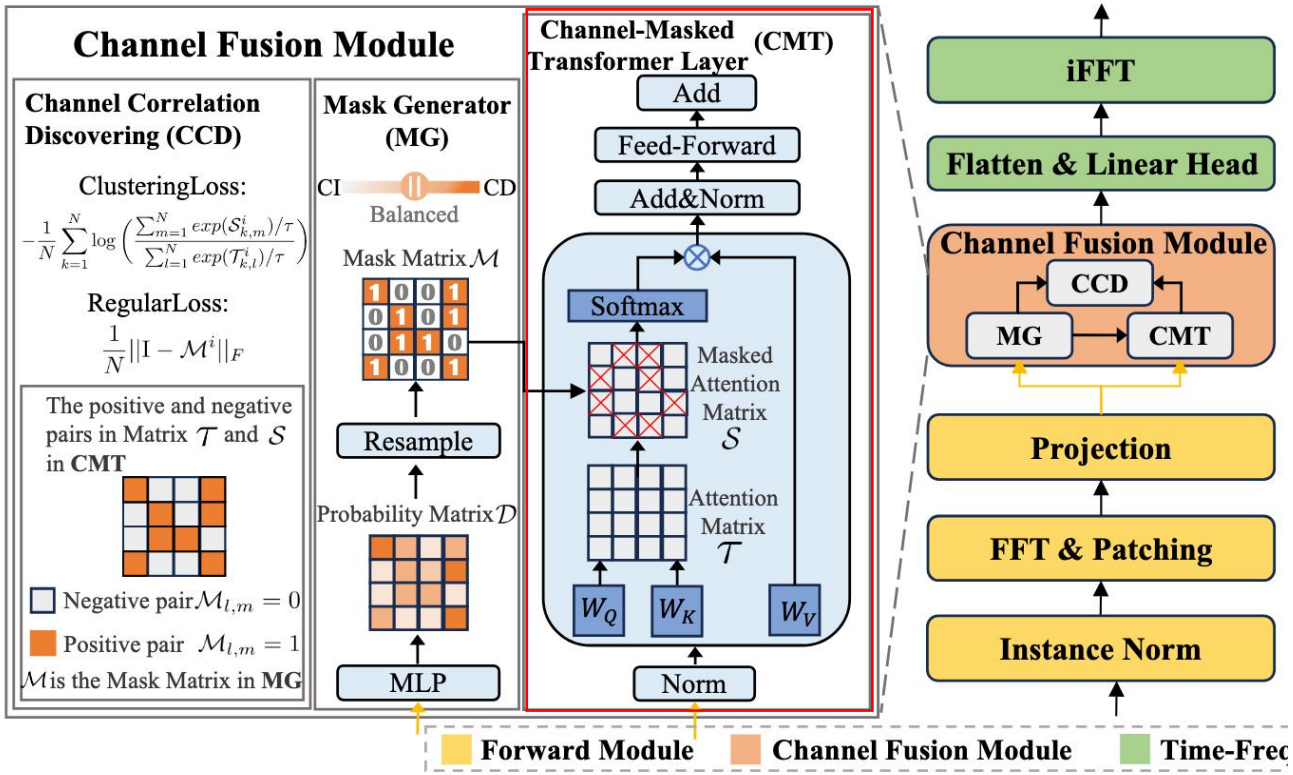
Mask Generator

$$\mathcal{D}^i = \sigma(\mathrm{Linear}(\mathrm{P}'^{i})), \mathcal{M}^i = \mathrm{Resample}(\mathcal{D}^i),$$

where $\mathrm{P}'^{i} \in \mathbb{R}^{N\times d}$, $\mathrm{D}^i \in \mathbb{R}^{N\times N}$, and $\mathcal{M}^i \in \mathbb{R}^{N\times N}$

分别是第i个 patch 的隐藏表示、概率矩阵和二值掩码矩阵

代码架构——Channel Fusion Module



$$P'^{i} \in \mathbb{R}^{N \times d}$$

Channel-Masked Transformer Layer

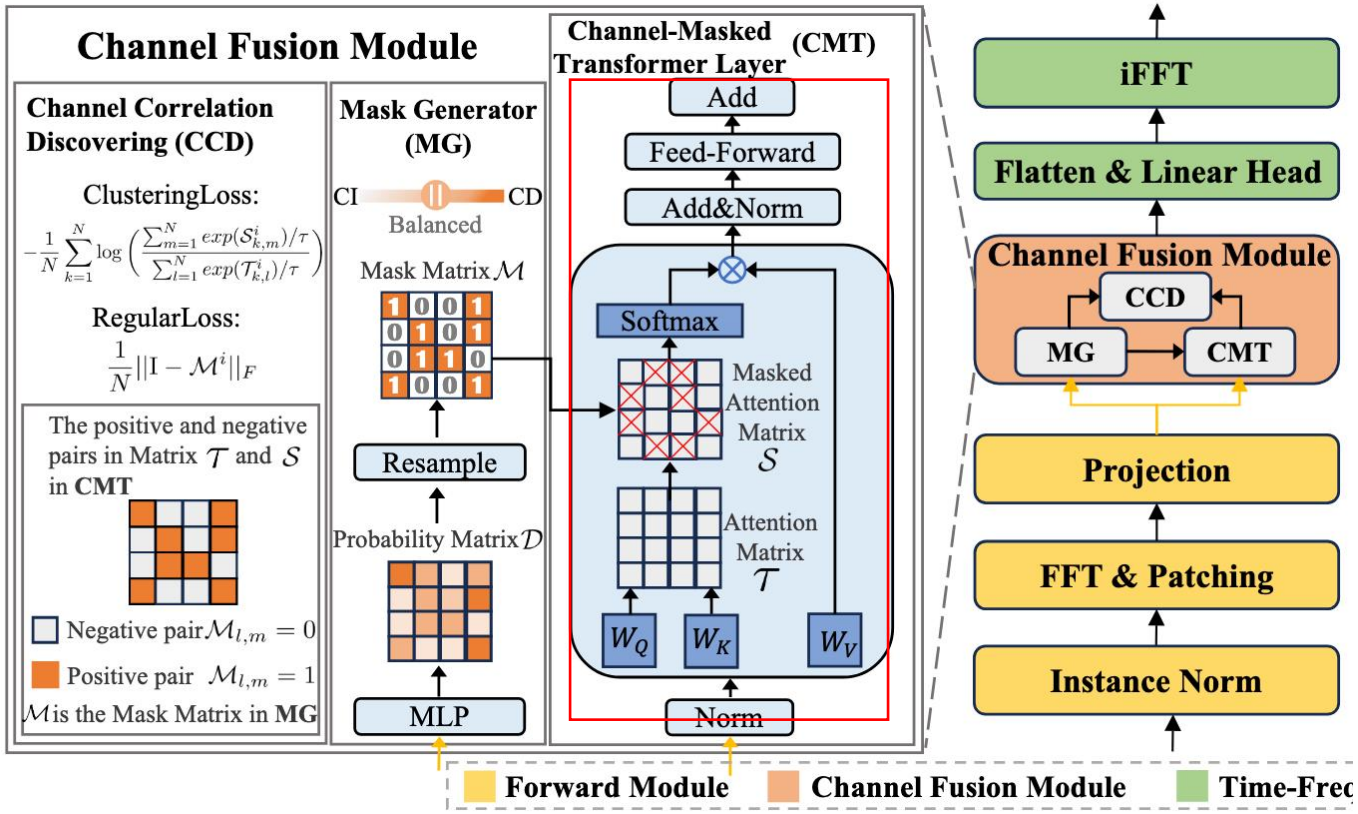
$$P^{*i} = \text{LayerNorm}(P'^{i}) = (P'^{i} - \text{Mean}_{n=1}^{N}(P'^{i}_{n,:}))/\sqrt{\text{Var}_{n=1}^{N}(P'^{i}_{n,:})},$$

$$Q^{i} = P^{*i} \cdot W^{Q}, K^{i} = P^{*i} \cdot W^{K}, V^{i} = P^{*i} \cdot W^{V},$$

$$\mathcal{T}^{i} = Q^{i} \cdot (K^{i})^{T}, \mathcal{S}^{i} = \mathcal{T}^{i} \odot \mathcal{M}^{i} + (1 - \mathcal{M}^{i}) \odot (-\infty),$$

$$\text{MaskedScores}^{i} = \mathcal{S}^{i}/\sqrt{d}, \tilde{P}^{i} = \text{Softmax}(\text{MaskedScores}^{i}) \cdot V^{i},$$

代码架构——Channel Fusion Module



Channel Correlation Discovering

$$\text{ClusteringLoss} = -\frac{1}{N}\sum_{k=1}^{N}\log\left(\frac{\sum_{m=1}^{N} exp(\mathcal{S}_{k,m}^{i})/\tau}{\sum_{l=1}^{N} exp(\mathcal{T}_{k,l}^{i})/\tau}\right)$$

$$\text{RegularLoss} = \frac{1}{N}||\mathrm{I} - \mathcal{M}^{i}||_{F}$$

代码架构——Time-Frequency Reconstruction Module

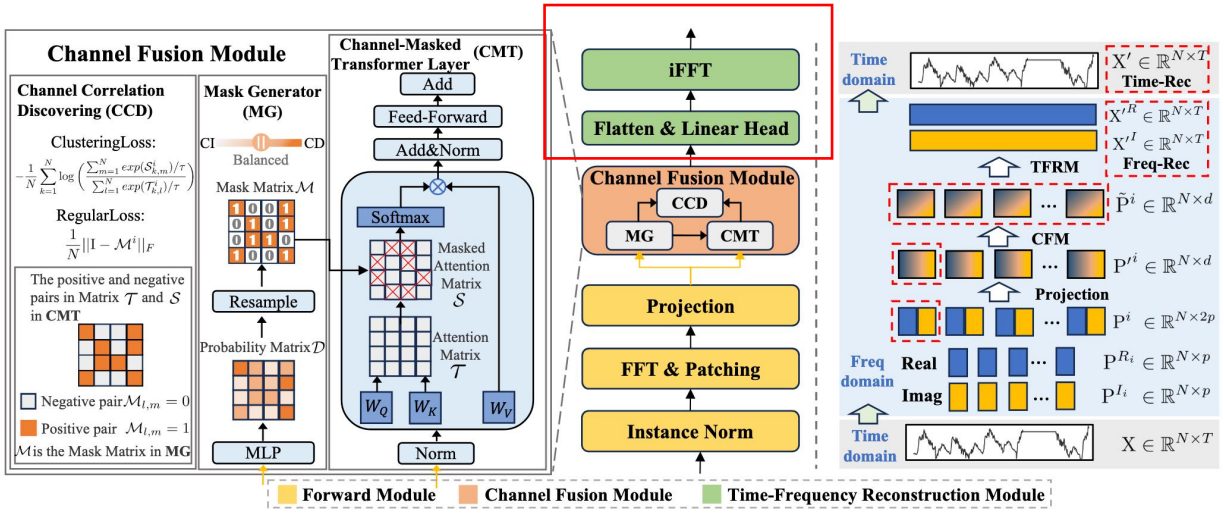


Figure 2: CATCH architecture. (1) Forward Module normalizes the input data, patchifies the frequency domain, and then projects it into the hidden space. (2) Channel Fusion Module captures channel interrelationships in each frequency band with a Channel-Masked Transformer (CMT) Layer, where the mask matrix (channel correlation) is generated by Mask Generator (MG). During training, MG and CMT are optimized by Channel Correlation Discovering (CCD) for more appropriate channel correlations. (3) Time-Frequency Reconstruction Module obtains the frequency reconstruction through Flatten & Linear Head Layer, and obtains the time reconstruction after iFFT.

$$\mathrm{X}'^R = \mathrm{Projection}_R(\mathrm{FlattenHead}(\{\tilde{\mathrm{P}}^{R_1}, \tilde{\mathrm{P}}^{R_2}, \cdots, \tilde{\mathrm{P}}^{R_L}\})),$$
$$\mathrm{X}'^I = \mathrm{Projection}_I(\mathrm{FlattenHead}(\{\tilde{\mathrm{P}}^{I_1}, \tilde{\mathrm{P}}^{I_2}, \cdots, \tilde{\mathrm{P}}^{I_L}\})),$$
$$\mathrm{X}' = \mathrm{iFFT}(\mathrm{X}'^R, \mathrm{X}'^I)$$

$$\mathrm{RecLoss}^{time} = ||\mathrm{X} - \mathrm{X}'||_F^2$$

$$\mathrm{RecLoss}^{freq} = ||\mathrm{X}^R - \mathrm{X}'^R||_1 + ||\mathrm{X}^I - \mathrm{X}'^I||_1$$

联合双重优化

$$\mathcal{L} = \text{RecLoss}^{time} + \lambda_1 \cdot \text{RecLoss}^{freq} + \lambda_2 \cdot \text{ClusteringLoss} + \lambda_3 \cdot \text{RegularLoss},$$

Algorithm 1 Bi-level Gradient Descent Optimization

Input: Model parameters θ_{model}, θ_{mask}, learning rate η_{model}, η_{mask}, number of iterations $\mathcal{N}_O$, $\mathcal{N}_I$, loss function $\mathcal{L} = \text{RecLoss}^{time} + \lambda_1 \cdot \text{RecLoss}^{freq} + \lambda_2 \cdot \text{ClusteringLoss} + \lambda_3 \cdot \text{RegularLoss}$
Initialize: $\theta_{\text{model}} \leftarrow$ initial value, $\theta_{\text{mask}} \leftarrow$ initial value
For $i = 1$ to $\mathcal{N}_O$
Outer Loop: Update the mask generator parameters
    $\theta_{\text{mask}} \leftarrow \theta_{\text{mask}} - \eta_{\text{mask}} \cdot \nabla_{\theta_{\text{mask}}} \mathcal{L}(\theta_{\text{model}}, \theta_{\text{mask}})$ ▷ Update the Mask Generator
    For $j = 1$ to $\mathcal{N}_I$
    Inner Loop: Update the model parameters
        $\theta_{\text{model}} \leftarrow \theta_{\text{model}} - \eta_{\text{model}} \cdot \nabla_{\theta_{\text{model}}} \mathcal{L}(\theta_{\text{model}}, \theta_{\text{mask}})$ ▷ Update the model
    EndFor
EndFor
Output: Optimized parameters θ_{model}, θ_{mask}

异常评分

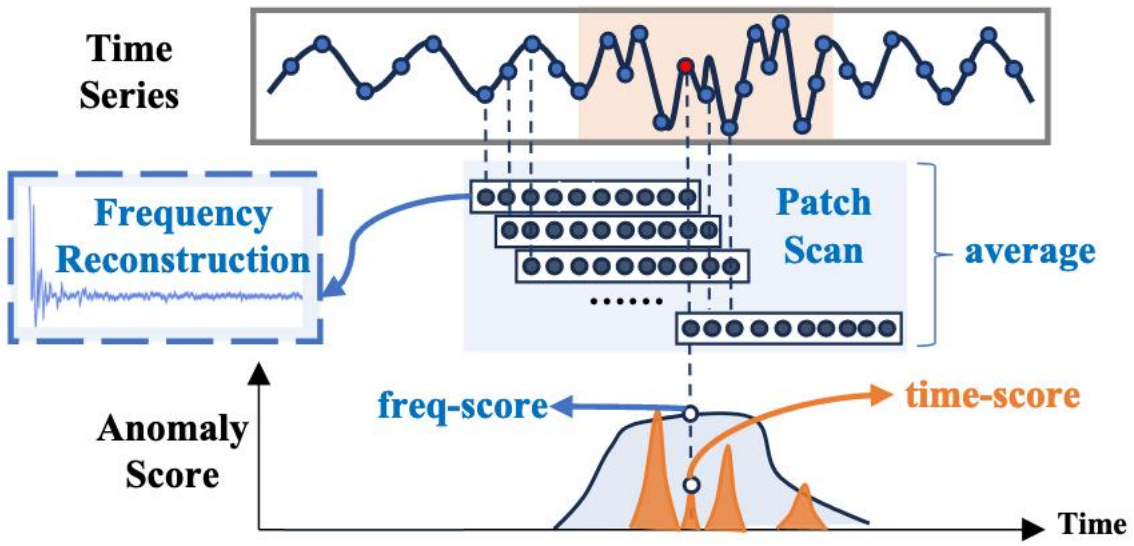


Figure 3: Proposed Anomaly Scoring.

- 以时间序列中红色时间点的异常分数计算为例：
  - 首先计算其在时间域的重构值与真实值之间的时间域重构误差，并将结果记为时间域异常分数（time-score）。
  - 与此同时，我们收集所有包含该红色时间点的片段，将这些片段转换到频域，并计算重构值与真实频域值之间的频域重构误差。然后，将所有片段的频域重构误差取平均值，作为该红色时间点的频域异常分数（freq-score），因为研究表明，平均值相比于最小值或最大值表现更优。
  - 最后，对时间域异常分数（time-score）和频域异常分数（freq-score）逐点加权求和，得到该时间序列的最终异常分数。

$$\text{AnomalyScore} = \text{time-score} + \lambda_{score} \cdot \text{freq-score}$$

实验结果

Table 2: Average A-R (AUC-ROC) and Aff-F (Affiliated-F1) accuracy measures for 11 real-world datasets and 6 synthetic datasets of different types of anomalies. The best results are highlighted in bold, and the second-best results are underlined.

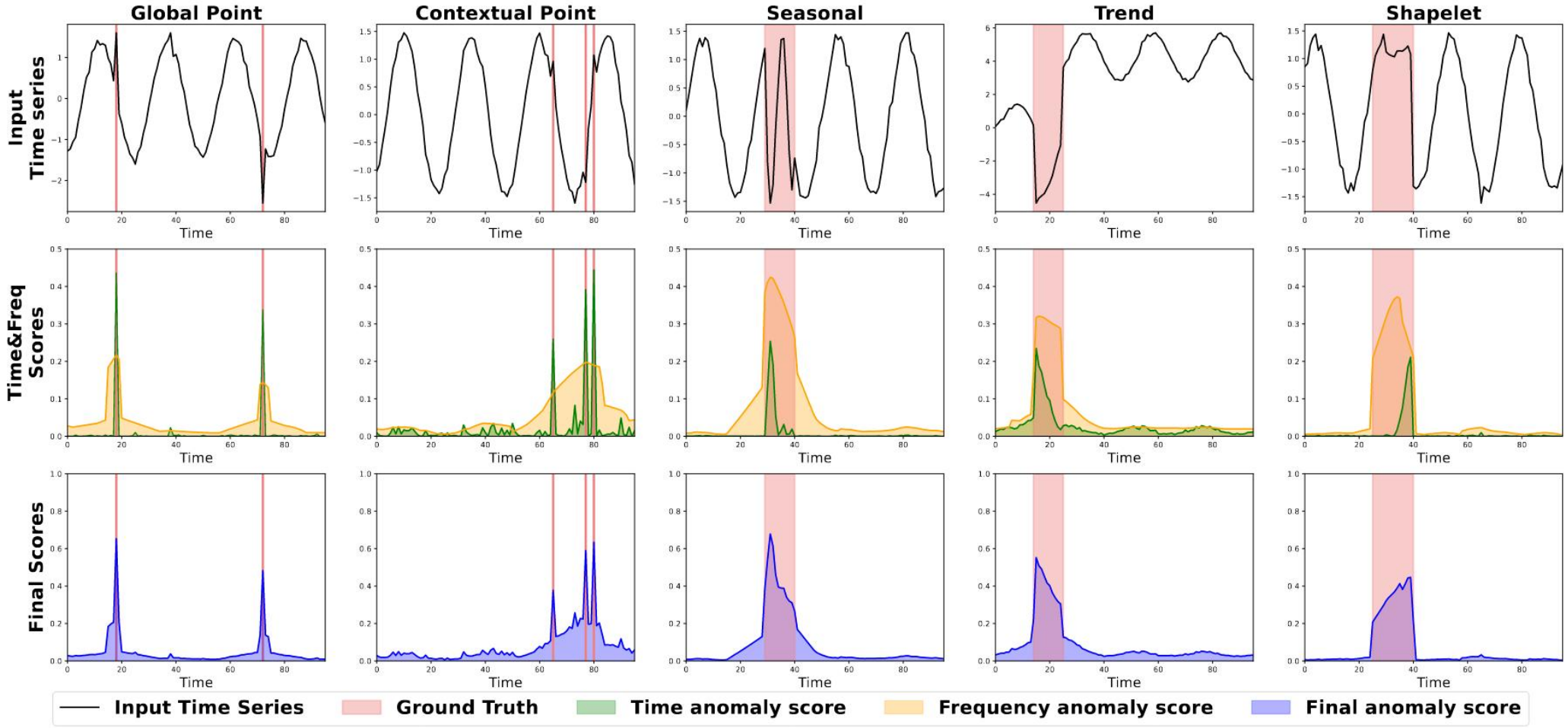
Dataset	Metric	CATCH	Modern	iTrans	DualTF	ATrans	DC	TsNet	Patch	DLin	NLin	AE	Ocsvm	IF	PCA	HBOS	TFAD
CICIDS	Aff-F	**0.787**	0.654	0.708	0.692	0.560	0.664	0.657	0.660	0.669	0.669	0.243	0.693	0.604	0.619	0.542	0.579
	A-R	**0.795**	0.697	0.692	0.603	0.528	0.638	0.732	0.716	0.751	0.691	0.629	0.537	0.787	0.601	0.760	0.504
CalIt2	Aff-F	**0.835**	0.780	0.812	0.751	0.729	0.697	0.794	0.793	0.793	0.757	0.587	0.783	0.402	0.768	0.756	0.744
	A-R	**0.838**	0.676	0.791	0.574	0.533	0.527	0.771	0.808	0.752	0.695	0.767	0.804	0.775	0.790	0.798	0.504
Credit	Aff-F	**0.750**	0.744	0.713	0.663	0.650	0.632	0.744	0.746	0.738	0.742	0.561	0.714	0.634	0.710	0.695	0.600
	A-R	**0.958**	0.957	0.934	0.703	0.552	0.504	0.957	0.957	0.954	0.948	0.909	0.953	0.860	0.871	0.951	0.500
GECCO	Aff-F	**0.908**	0.893	0.839	0.701	0.782	0.687	0.894	0.906	0.893	0.882	0.823	0.666	0.424	0.785	0.708	0.627
	A-R	**0.970**	0.952	0.795	0.714	0.516	0.555	0.954	0.949	0.947	0.936	0.769	0.804	0.619	0.711	0.557	0.499
Genesis	Aff-F	**0.896**	0.833	0.891	0.810	0.856	0.776	0.864	0.856	0.856	0.829	0.854	0.677	0.788	0.814	0.721	0.535
	A-R	**0.974**	0.676	0.690	0.937	0.947	0.659	0.913	0.685	0.696	0.755	0.931	0.733	0.549	0.815	0.897	0.497
MSL	Aff-F	**0.740**	0.726	0.710	0.588	0.692	0.694	0.734	0.724	0.725	0.723	0.625	0.641	0.584	0.678	0.680	0.665
	A-R	**0.664**	0.633	0.611	0.576	0.508	0.507	0.613	0.637	0.624	0.592	0.562	0.524	0.524	0.552	0.574	0.500
NYC	Aff-F	**0.994**	0.769	0.684	0.708	0.853	0.862	0.794	0.776	0.828	0.819	0.689	0.667	0.648	0.680	0.675	0.689
	A-R	**0.816**	0.466	0.640	0.633	0.671	0.549	0.791	0.709	0.768	0.671	0.504	0.456	0.475	0.666	0.446	0.502
PSM	Aff-F	**0.859**	0.825	0.854	0.725	0.710	0.682	0.842	0.831	0.831	0.843	0.707	0.531	0.620	0.702	0.658	0.628
	A-R	**0.652**	0.593	0.592	0.600	0.514	0.501	0.592	0.586	0.580	0.585	0.650	0.619	0.542	0.648	0.620	0.500
SMD	Aff-F	**0.847**	0.840	0.827	0.679	0.724	0.675	0.831	0.845	0.841	0.844	0.439	0.742	0.626	0.738	0.629	0.660
	A-R	**0.811**	0.722	0.745	0.631	0.508	0.502	0.727	0.736	0.728	0.738	0.774	0.602	0.664	0.679	0.626	0.500
SWAT	Aff-F	**0.755**	0.728	0.718	0.695	0.573	0.567	0.720	0.730	0.725	0.729	0.737	0.691	0.586	0.678	0.673	0.686
	A-R	0.545	0.244	0.242	**0.567**	0.488	0.534	0.506	0.482	0.471	0.500	0.497	0.529	0.410	0.496	0.521	0.500
ASD	Aff-F	**0.804**	0.782	0.780	0.605	0.674	0.702	0.800	0.777	0.782	0.766	0.731	0.617	0.781	0.656	0.669	0.630
	A-R	**0.824**	0.692	0.759	0.579	0.506	0.520	0.805	0.760	0.739	0.690	0.704	0.588	0.618	0.656	0.603	0.502
Contextual	Aff-F	**0.823**	0.619	0.802	0.635	0.601	0.597	0.666	0.766	0.780	0.700	0.755	0.696	0.679	0.475	0.481	0.569
	A-R	**0.910**	0.562	0.905	0.598	0.546	0.525	0.908	0.854	0.700	0.530	0.896	0.711	0.821	0.538	0.464	0.504
Global	Aff-F	**0.949**	0.748	0.922	0.649	0.656	0.567	0.910	0.940	0.928	0.808	0.919	0.849	0.912	0.704	0.528	0.566
	A-R	**0.997**	0.873	0.976	0.595	0.564	0.514	0.989	0.992	0.979	0.675	0.996	0.996	0.938	0.758	0.608	0.500
Seasonal	Aff-F	**0.997**	0.681	0.992	0.776	0.788	0.859	0.992	0.989	0.993	0.951	0.927	0.805	0.938	0.637	0.673	0.686
	A-R	**0.998**	0.512	0.946	0.701	0.584	0.644	0.958	0.922	0.823	0.623	0.949	0.829	0.918	0.437	0.516	0.502
Shapelet	Aff-F	**0.985**	0.675	0.961	0.692	0.699	0.737	0.941	0.933	0.961	0.759	0.871	0.771	0.887	0.683	0.640	0.684
	A-R	**0.970**	0.522	0.864	0.573	0.519	0.597	0.877	0.818	0.684	0.563	0.865	0.655	0.748	0.517	0.337	0.503
Trend	Aff-F	**0.916**	0.734	0.901	0.677	0.584	0.765	0.897	0.888	0.721	0.830	0.699	0.691	0.914	0.693	0.669	0.642
	A-R	**0.892**	0.612	0.847	0.524	0.500	0.569	0.858	0.835	0.671	0.642	0.482	0.471	0.878	0.484	0.468	0.502
Mixture	Aff-F	**0.892**	0.856	0.862	0.652	0.641	0.709	0.863	0.879	0.727	0.839	0.673	0.676	0.881	0.676	0.667	0.710
	A-R	**0.931**	0.763	0.854	0.570	0.522	0.516	0.861	0.863	0.767	0.749	0.493	0.475	0.911	0.517	0.531	0.501

实验结果

Table 3: Multi-metrics results on three real-world multivariate datasets. The best ones are in Bold.

Dataset	Method	Acc	P	R	F1	R-P	R-R	R-F	Aff-P	Aff-R	Aff-F	A-R	A-P	R-A-R	R-A-P	V-ROC	V-PR
GECCO	TimesNet	0.984	0.379	0.804	0.516	0.053	0.782	0.099	0.810	0.997	0.894	0.954	0.410	0.977	0.428	0.974	0.429
	ModernTCN	0.984	0.373	0.779	0.504	**0.086**	0.644	**0.152**	0.808	0.998	0.893	0.952	**0.447**	0.978	0.459	0.975	0.461
	CATCH (ours)	**0.984**	**0.380**	**0.818**	**0.518**	0.065	**0.795**	0.119	**0.832**	**0.998**	**0.908**	**0.970**	0.418	**0.990**	**0.473**	**0.987**	**0.465**
MSL	TimesNet	0.855	0.166	0.093	0.119	0.130	0.224	0.164	0.589	0.973	0.734	0.613	0.146	0.701	0.231	0.692	0.227
	ModernTCN	**0.857**	0.166	0.090	0.117	0.129	0.194	0.155	0.578	**0.975**	0.726	0.633	0.146	0.708	0.224	0.701	0.220
	CATCH (ours)	0.853	**0.185**	**0.117**	**0.143**	**0.150**	**0.241**	**0.185**	**0.599**	0.966	**0.740**	**0.664**	**0.167**	**0.747**	**0.260**	**0.735**	**0.256**
SMD	TimesNet	0.931	0.176	0.181	0.178	**0.110**	0.385	**0.171**	0.745	0.938	0.831	0.727	0.141	0.747	0.140	0.746	0.140
	ModernTCN	**0.931**	0.151	0.145	0.148	0.092	0.378	0.148	0.755	**0.948**	0.840	0.722	0.130	0.743	0.130	0.742	0.130
	CATCH (ours)	0.918	**0.194**	**0.305**	**0.237**	0.095	**0.478**	0.158	**0.773**	0.938	**0.847**	**0.811**	**0.172**	**0.800**	**0.159**	**0.797**	**0.159**

实验结果


Global Point
Contextual Point
Seasonal
Trend
Shapelet
Input Time series
Time&Freq Scores
Final Scores
Time
Input Time Series
Ground Truth
Time anomaly score
Frequency anomaly score
Final anomaly score

论文 & 代码

华東師範大學 EAST CHINA NORMAL UNIVERSITY

论文

Published as a conference paper at ICLR 2025

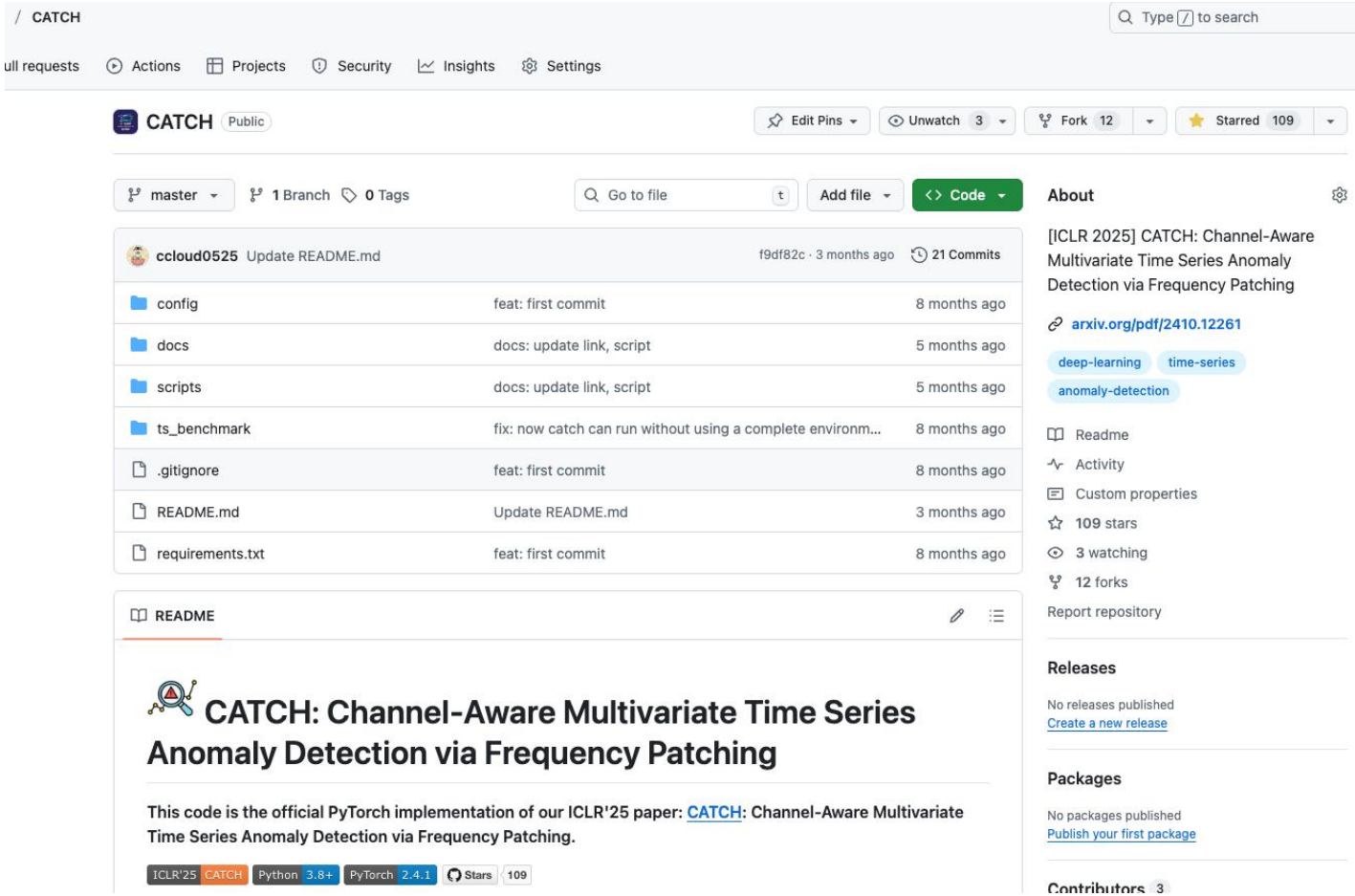
CATCH: CHANNEL-AWARE MULTIVARIATE TIME SERIES ANOMALY DETECTION VIA FREQUENCY PATCHING

Xingjian Wu[1], Xiangfei Qiu[1], Zhengyu Li[1], Yihang Wang[1], Jilin Hu[1], Chenjuan Guo[1], Hui Xiong[2], Bin Yang[1]*
[1]East China Normal University
[2]The Hong Kong University of Science and Technology (Guangzhou)
{xjwu,xfqiu,lizhengyu,yhwang}@stu.ecnu.edu.cn,
{jlhu,cjguo,byang}@dase.ecnu.edu.cn, xionghui@ust.hk

ABSTRACT

Anomaly detection in multivariate time series is challenging as heterogeneous subsequence anomalies may occur. Reconstruction-based methods, which focus on learning normal patterns in the frequency domain to detect diverse abnormal subsequences, achieve promising results, while still falling short on capturing fine-grained frequency characteristics and channel correlations. To contend with the limitations, we introduce CATCH, a framework based on frequency patching. We propose to patchify the frequency domain into frequency bands, which enhances its ability to capture fine-grained frequency characteristics. To perceive appropriate channel correlations, we propose a Channel Fusion Module (CFM), which features a patch-wise mask generator and a masked-attention mechanism. Driven by a bi-level multi-objective optimization algorithm, the CFM is encouraged to iteratively discover appropriate patch-wise channel correlations, and to cluster relevant channels while isolating adverse effects from irrelevant channels. Extensive experiments on 12 real-world datasets and 12 synthetic datasets demonstrate that CATCH achieves state-of-the-art performance. We make our code and datasets available at https://github.com/decisionintelligence/CATCH.

代码



感谢您的垂听