2025 한양대학교

SW/ICT/AI 종합학술대회

[HYAI : PAPER TO CODE CHALLENGE]

논문기반 Anomaly Transformer의 재현 및 검증

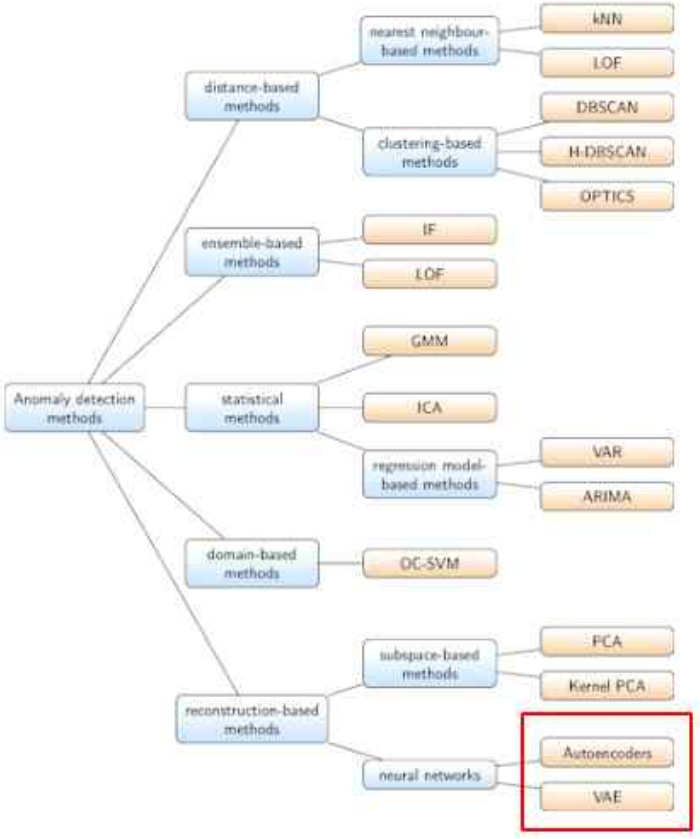
김예준(2025091621), 이라온(2025078622)

2025. 11. 28

한양대학교 ERICA 소프트웨어융합대학 / 한양대학교 ERICA SW중심대학사업단

ANOMALY TRANSFORMER: TIME SERIES ANOMALY DETECTION WITH ASSOCIATION DISCREPANCY

- 비지도 학습 anomaly detection 의 하나로 가장 많이 사용되는 방법 중 하나
- 시계열 데이터를 입력받아 latent space(잠재 공간)을 만들어 입력 데이터를 그대로 복원시키는 방법
- 이때 원본 데이터와의 차이(에러)를 기준으로 정상 / 비정상 데이터를 구분
- PCA, AE, VAE 등이 대표적인 알고리즘
  ,해당 모델은 트랜스포머를 사용하여 연관성을 학습한 모델
- OC-SVM, LOF, SVDD와 같은 anomaly detection의 classic한 방법들은
  temporal한 정보들을 고려하지 않고 실제 시나리오의 실제 데에터에 일반화되기 어렵다는 문제점들이 있음
- 기존의 트랜스포머 모델을 사용하여 시계열 데이터 내에서 포인트들간의 연관성을 학습할 수 있도록 변형



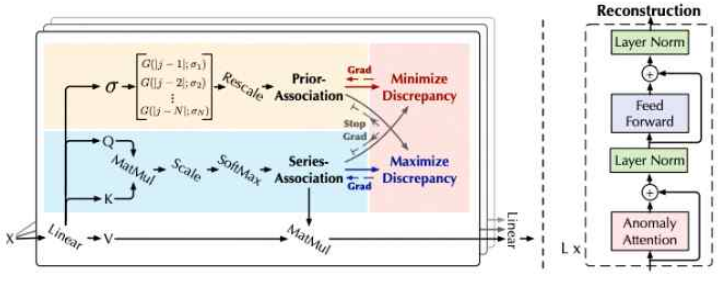
문제 정의

왜 기존 Transformer는 이상 탐지(Anomaly Detection)에서 한계가 있는가?

- Point-wise vs Pattern-wise: 시계열 데이터의 이상치는 단일 점뿐만 아니라 패턴의 변화로 나타남.
- Transformer의 딜레마: Transformer는 전역적(Global)인 연관성을 찾는 능력이 탁월하여,
  이상한 패턴마저도 주변 문맥을 통해 '잘' 복원해버림 (Over-generalization).
- 정상 데이터뿐만 아니라 이상 데이터도 재구성 오차(Reconstruction Error)가
  낮게 나와 탐지가 어려움.
- "Anomaly Transformer"는 전체적인 연관성과 지역적인 연관성의 차이(Association Discrepancy)를
  이용해 이 문제를 해결함.

모델 구조 및 Minimax 전략, Anomaly Attention

- 어노말 트랜스포머는 시계열 데이터 내에서 prior-association, series-association 이라는 2가지 연관성을 학습
- Prior-association(P) : 인접한 데이터들간의 연관성, 시간 차이를 기준으로 가우시안 분포로 모델링
- Series-association(S) : 시계열 전체에서 데이터들간의 연관성, self-attention에 의해 모델링



$$\mathcal{Z}^l = \text{Layer-Norm}\Big(\text{Anomaly-Attention}(\mathcal{X}^{l-1}) + \mathcal{X}^{l-1}\Big)$$
$$\mathcal{X}^l = \text{Layer-Norm}\Big(\text{Feed-Forward}(\mathcal{Z}^l) + \mathcal{Z}^l\Big),$$

I 번째 layer에서의 anomaly attention은 아래와 같이 계산됨

$$\text{Initialization: } \mathcal{Q}, \mathcal{K}, \mathcal{V}, \sigma = \mathcal{X}^{l-1}W_{\mathcal{Q}}^l, \mathcal{X}^{l-1}W_{\mathcal{K}}^l, \mathcal{X}^{l-1}W_{\mathcal{V}}^l, \mathcal{X}^{l-1}W_{\sigma}^l$$
$$\text{Prior-Association: } \mathcal{P}^l = \text{Rescale}\left(\left[\frac{1}{\sqrt{2\pi}\sigma_i}\exp\left(-\frac{|j-i|^2}{2\sigma_i^2}\right)\right]_{i,j\in\{1,\cdots,N\}}\right)$$
$$\text{Series-Association: } \mathcal{S}^l = \text{Softmax}\left(\frac{\mathcal{Q}\mathcal{K}^{\mathrm{T}}}{\sqrt{d_{\text{model}}}}\right)$$
$$\text{Reconstruction: } \widehat{\mathcal{Z}}^l = \mathcal{S}^l\mathcal{V},$$

Association Discrepancy

prior 과 series association의 대칭화된 KL divergence가 association discrepancy임이는 두개의 distribution에 관한 information gain을 나타냄

$$\text{AssDis}(\mathcal{P}, \mathcal{S}; \mathcal{X}) = \left[\frac{1}{L}\sum_{l=1}^{L}\Big(\text{KL}(\mathcal{P}_{i,:}^l \| \mathcal{S}_{i,:}^l) + \text{KL}(\mathcal{S}_{i,:}^l \| \mathcal{P}_{i,:}^l)\Big)\right]_{i=1,\cdots,N} \quad (3)$$

Minimax association learning

- 비지도 task에서는 model의 optimizing을 위해서 reconstruction loss를 사용함
- reconstruction loss는 series association이 가장 중요한 association을 찾을 수 있도록 도와줌
- noraml과 abnormal time point간의 difference를 증폭시키기 위해서 additional loss를 사용해서 association discrepancy를 증가시켜 주었음

$$\mathcal{L}_{\text{Total}}(\widehat{\mathcal{X}}, \mathcal{P}, \mathcal{S}, \lambda; \mathcal{X}) = \|\mathcal{X} - \widehat{\mathcal{X}}\|_{\text{F}}^2 - \lambda \times \|\text{AssDis}(\mathcal{P}, \mathcal{S}; \mathcal{X})\|_1 \qquad (4)$$

- association discrepancy를 직접적으로 maximize시키면,
  가우시안 커널의 파라미터 스케일을 엄청나게 감소시킬 거고 prior association을 의미없게 만들것임
- association learning을 보다 효과적으로 control하기 위해서 본 연구에서는 minimax strategy를 사용

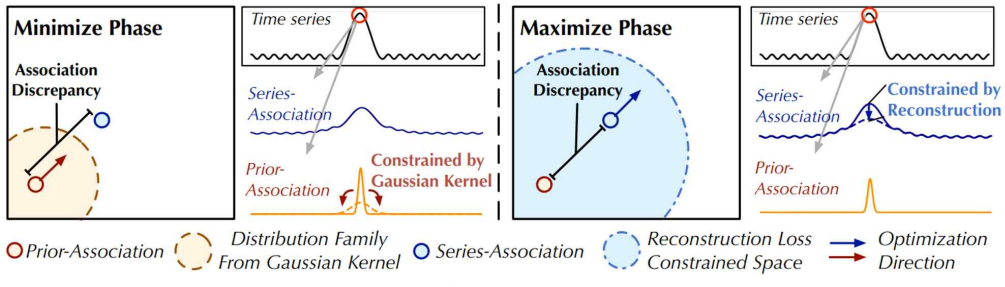


Figure 2: Minimax association learning. At the minimize phase, the prior-association minimizes the Association Discrepancy within the distribution family derived by Gaussian kernel. At the maximize phase, the series-association maximizes the Association Discrepancy under the reconstruction loss.

$$\text{AnomalyScore}(\mathcal{X}) = \text{Softmax}\Big(-\text{AssDis}(\mathcal{P}, \mathcal{S}; \mathcal{X})\Big) \odot \Big[\|\mathcal{X}_{i,:} - \widehat{\mathcal{X}}_{i,:}\|_2^2\Big]_{i=1,\cdots,N} \qquad (6)$$

핵심내용 구현

Reconstruction & Association 수집 논문에서는 모델이
재구성(Reconstruction) 결과와 연관성(Association) 정보를 동시에 내뱉어야 한다고 말함.

```
def forward(self, x):
    enc_out = self.embedding(x)
    enc_out, series, prior, sigmas = self.encoder(enc_out)
    enc_out = self.projection(enc_out)
```

논문의 핵심은 각 레이어에서 전역적 관계(Series)와 지역적 관계(Prior)를 동시에 계산하는 것.
EncoderLayer가 이 두 가지 Attention을 분리해서 처리함

```
def forward(self, x, attn_mask=None):
    new_x, attn, mask, sigma = self.attention(
        x, x, x,
        attn_mask=attn_mask
    )
    x = x + self.dropout(new_x)
    y = x = self.norm1(x)
    y = self.dropout(self.activation(self.conv1(y.transpose(-1, 1))))
    y = self.dropout(self.conv2(y).transpose(-1, 1))

    return self.norm2(x + y), attn, mask, sigma
```

각 레이어에서 전역적 관계(Series)와 지역적 관계(Prior)를 동시에 계산
EncoderLayer가 이 두 가지 Attention을 분리해서 처리

핵심내용 구현,

모든 레이어에 대해 Association Discrepancy를 계산하여 합침
Encoder 클래스는 이를 위해 각 레이어의 출력을 리스트로 모음

```python
def forward(self, x, attn_mask=None):
    # x [B, L, D]
    series_list = []
    prior_list = []
    sigma_list = []
    for attn_layer in self.attn_layers:
        x, series, prior, sigma = attn_layer(x, attn_mask=attn_mask)
        series_list.append(series)
        prior_list.append(prior)
        sigma_list.append(sigma)

    if self.norm is not None:
        x = self.norm(x)

    return x, series_list, prior_list, sigma_list
```

