

---

# Time Series Representation Learning

---

2022-01-14

**Data Mining and Quality Analytics Lab**

조억

# 발표자 소개

## ◆ 조억

- E-mail : ukjo@korea.ac.kr
- Data Mining & Quality Analytics Lab(김성범 교수님)
- 석박사 통합과정 학생(2019.8 ~)



**종료** Reinforcement Learning for Generalization

2021-03-12  
Data Mining and Quality Analytics Lab  
조억

Reinforcement Learning for Generalization

발표자: 조억

📅 2021년 3월 12일  
🕒 오후 1시 ~  
📺 온라인 비디오 시청 (YouTube)

세미나 정보 보기 →

**종료** RL Fundamental + Cooperative Multi-Agent RL Framework for Scalping Trading

발표자 : 조억

Cooperative Multi-Agent Reinforcement Learning

발표자: 조억

📅 2020년 1월 17일  
🕒 오후 1시 ~  
📍 고려대학교 신공학관 218호

**종료** Metric Studio "Metric is all you need for Model Evaluation"

Data Mining & Quality Analytics Lab  
2020년 8월 28일  
발표자 : 조억

Metrics is all you need for model evaluation

발표자: 조억

📅 2020년 8월 28일  
🕒 오후 1시 ~  
📺 온라인 비디오 시청 (YouTube)

- ❖ 관심분야
- ✓ Reinforcement Learning
  - ✓ Time Series Data Analysis

# 목차

---

1. Time Series Representation Learning
2. Time Series Augmentations
3. Representative Concrete Methods
4. Conclusion

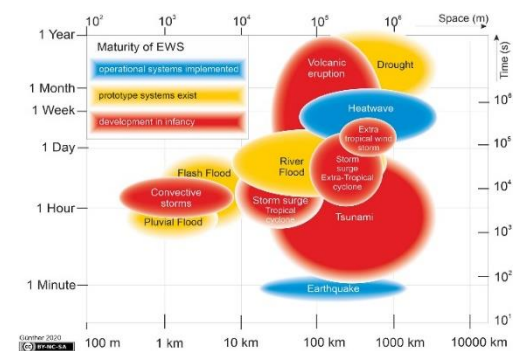
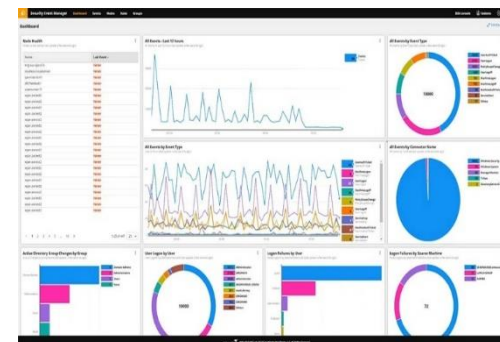
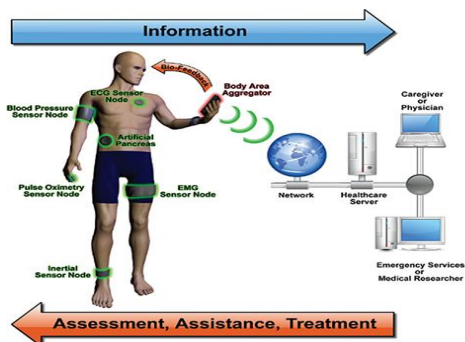
---

# **Time Series Representation Learning**

---

# 시계열 데이터

- ❖ 빅데이터, IoT, SNS 등 다양한 시계열 데이터들이 아주 빠른 속도로 생산되고 있음
- ❖ 헬스케어, 장비 이상탐지, 보안, 금융, 게임, 재난 예측 등 다양한 어플리케이션으로 중요성 증대



# 시계열 데이터 특성

---

## ❖ 시계열 데이터는 다음의 특성 가지고 있다

- ✓ 추세(Trend) : 세부적인 데이터를 다 빼고 전체적으로 보았을 때 큰 흐름을 보이는 경향 정보
- ✓ 계절성(Seasonality) : 특정한 기간마다 어떤 패턴을 가지고 반복하는 특성
- ✓ 랜덤(Random) : 추세, 계절성 등으로 설명되지 않은 노이즈(noise) 데이터 의미

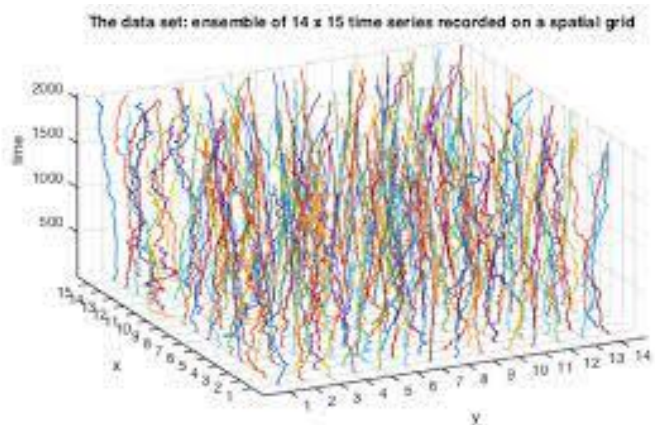
## ❖ 정상성 : 시계열의 분포가 시간에 따라 불변하다고 가정하고 시계열 분석 진행

## ❖ 일반 데이터와 다르게 $x, y$ 의 상관관계가 있다면 시계열 데이터는 이전 시점과 이후 시점과의 상관관계가 있다.

## ❖ 다변량 데이터의 경우는 단변량 데이터보다 더 복잡해지고 어려워짐

# 시계열 데이터 특징

- ❖ 시계열 데이터는 생성은 쉬운 반면, 도메인 전문가의 오랜 시간과 노력이 필요하여 라벨링 정보는 적고, 비용은 Image, Text 데이터보다 크다
- ❖ 라벨링 회사에도 이미지/텍스트에 집중하며 고차원 시계열 데이터는 지원 어려움



고차원의 시계열 데이터에서 라벨링은 쉽지 않으며, 전문가의 도움이 필요



HOW IT WORKS SERVICES INDUSTRIES ABOUT US CASE STUDIE

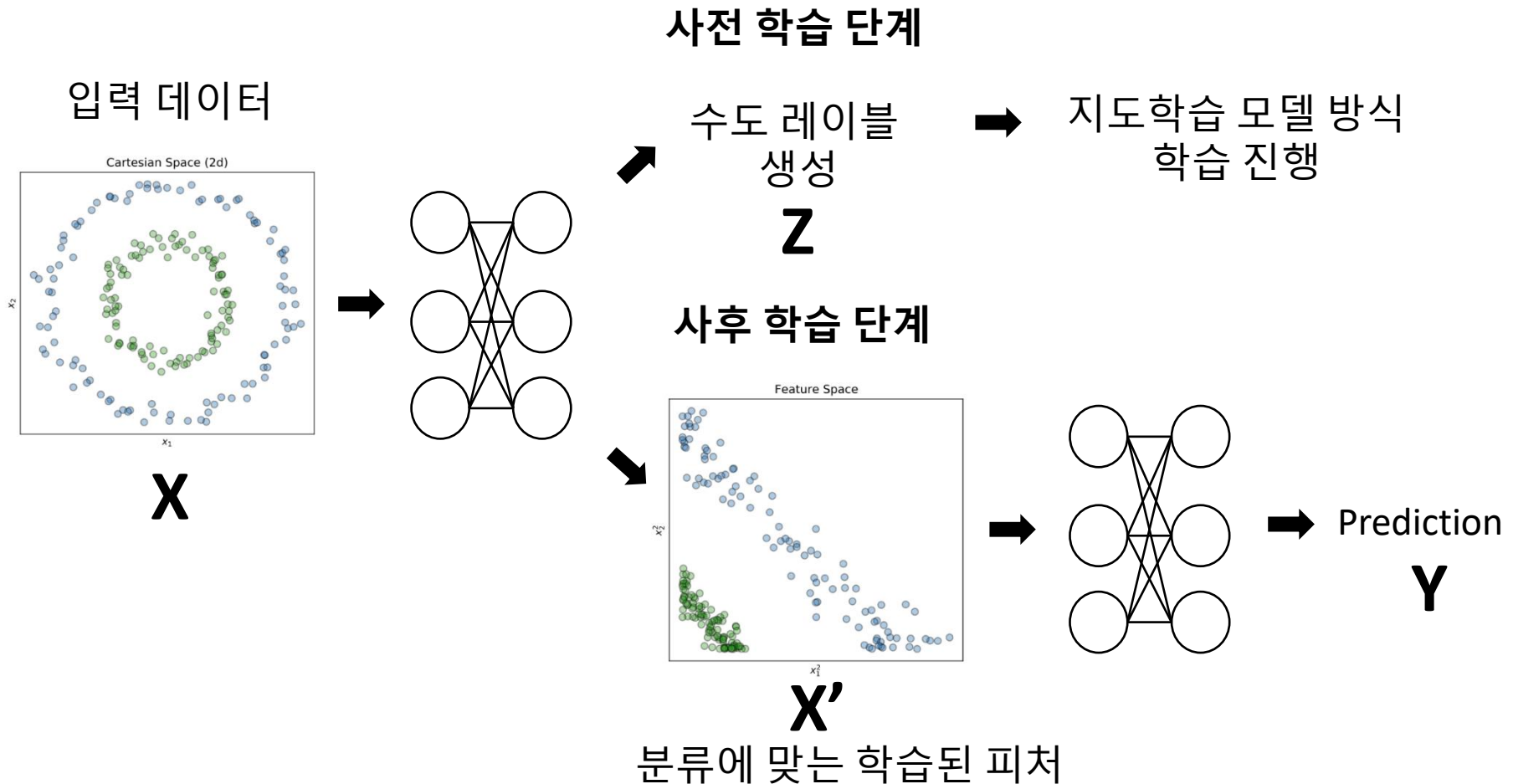
## Data Annotation Services

Artificial Intelligence (AI) and Machine Learning (ML) dictate a new approach to business – one that requires plenty of data. Pick the best data annotation outsourcing service for computer vision while saving money and time!

라벨링 서비스에 고차원의 시계열 데이터 라벨링 분류는 존재하지 않음

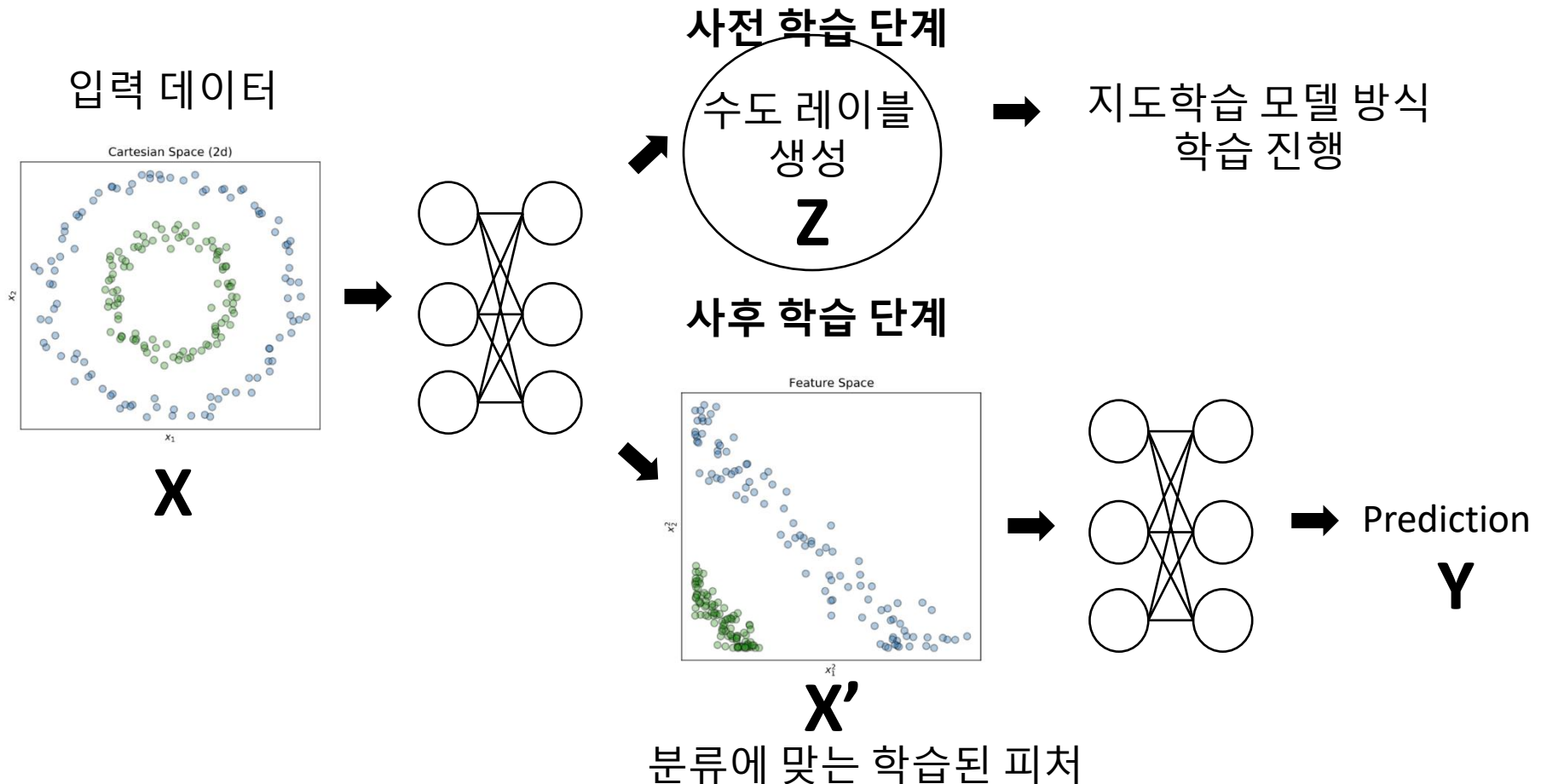
# Representation Learning

- ❖ 주어진 입력 데이터만을 이용하여 레이블 정보 없이 원래 학습하고자 하는 목표에 맞는 좋은 피쳐 정보를 추출하기 위한 사전 학습 방법



# Representation Learning(표현학습)

- ❖ 사전 학습 단계의 **수도 레이블**은 **Pretext**을 정의하면 알아서 만들어지게 됨
- ❖ 사후 학습 목적을 위한 좋은 피쳐( $X'$ )를 생성하기 위해 **사전 정의한 문제**이며, 언레이블 데이터에 표현 학습을 위한 레이블( $Z$ ) 부여

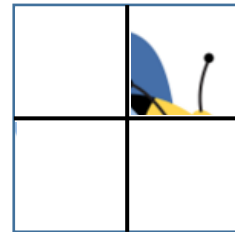


# Representation Learning

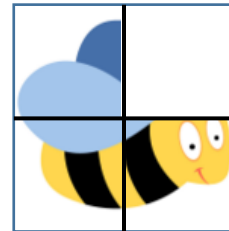
- ❖ Pretext 정의는 **데이터 특성과 사후 학습 목적에 따라 달라질 수 있음**
- ❖ 연구가 많이 이루어진 이미지 분야의 예시로 Pretext 종류 설명
  - ✓ **Masked Prediction**
  - ✓ Transformation Prediction
  - ✓ Instance (Contrastive) Discrimination



레이블이 “없는” 데이터



z



x

레이블이 “있는” 데이터

# Representation Learning

- ❖ Pretext 정의는 **데이터 특성과 사후 학습 목적에 따라 달라질 수 있음**
- ❖ 연구가 많이 이루어진 이미지 분야의 예시로 Pretext 종류 설명
  - ✓ Masked Prediction
  - ✓ **Transformation Prediction**
  - ✓ Instance (Contrastive) Discrimination



X

레이블이 “없는” 데이터



90도

Z

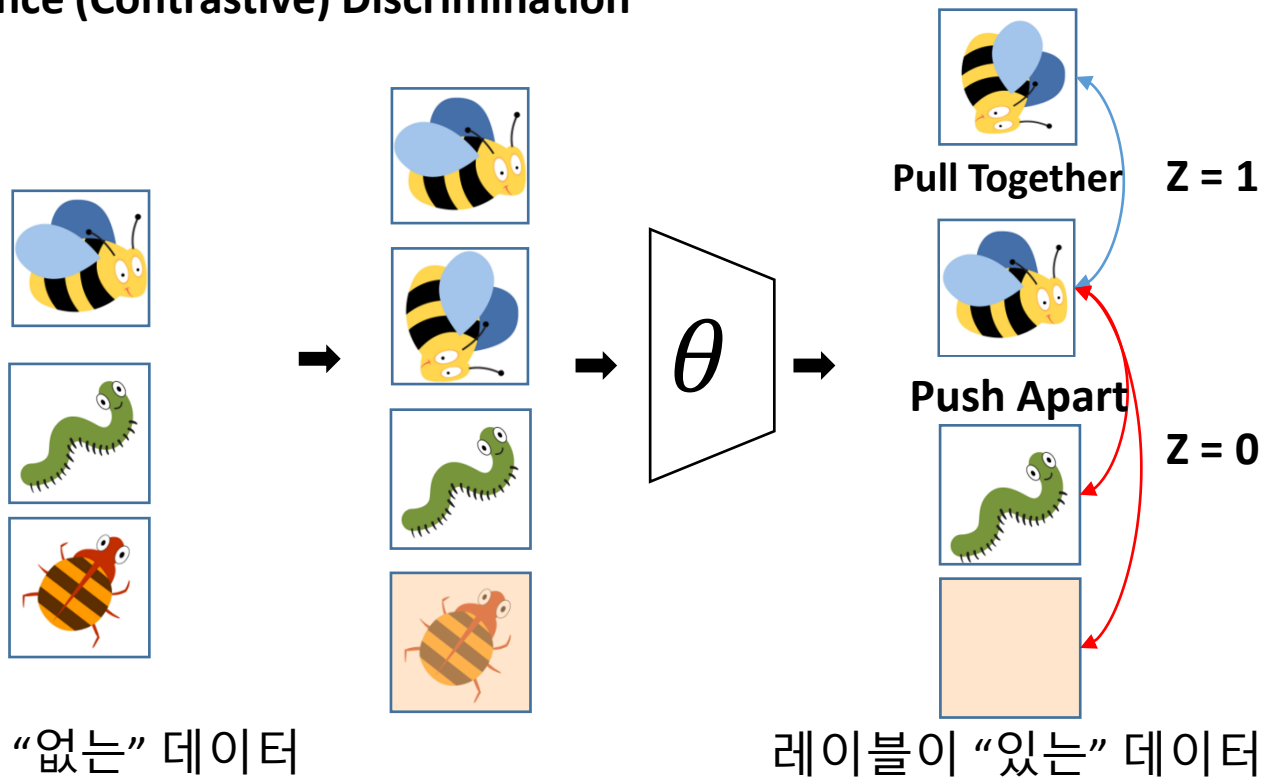


X

레이블이 “있는” 데이터

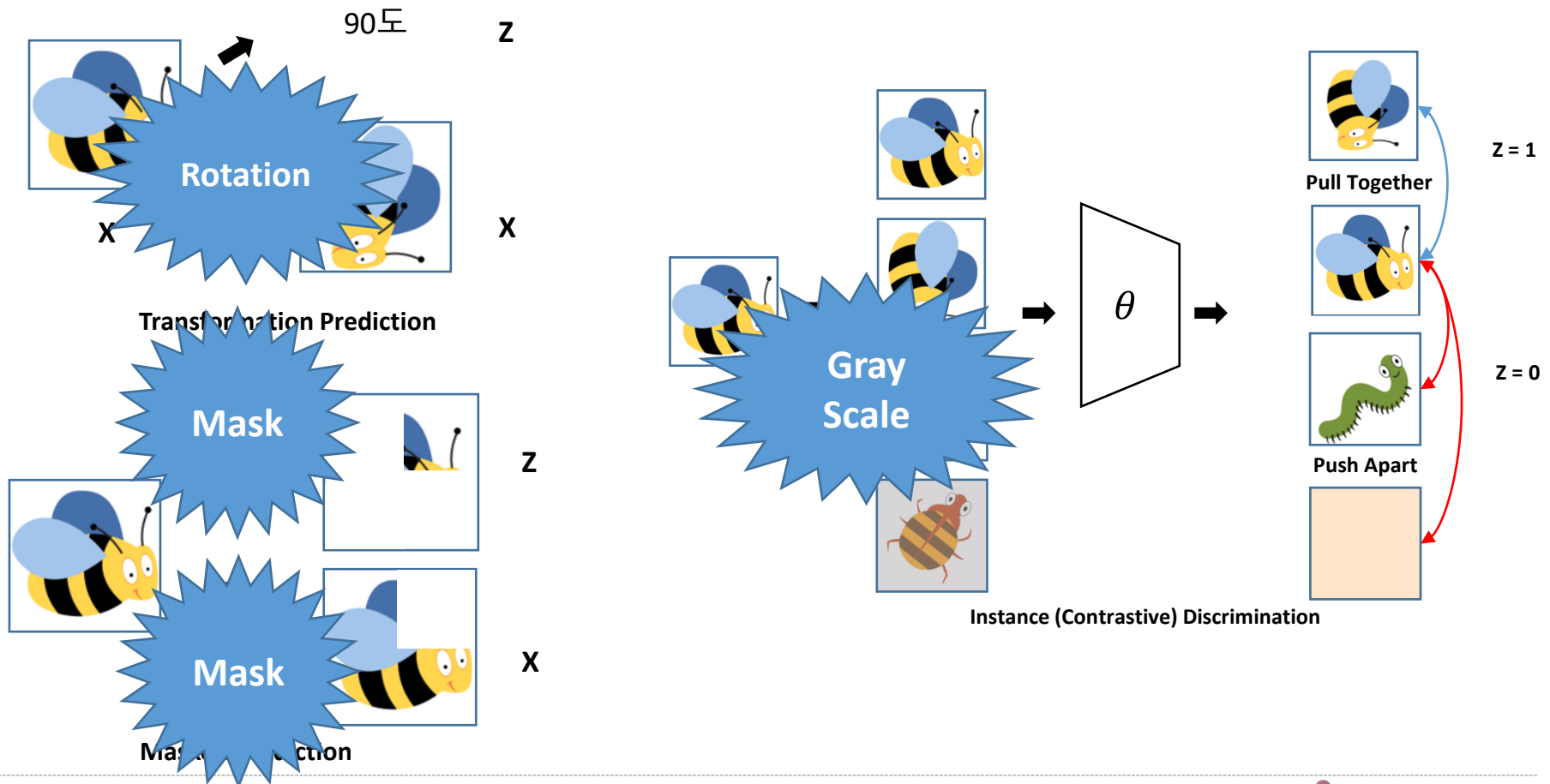
# Representation Learning

- ❖ Pretext 정의는 **데이터 특성과 사후 학습 목적에 따라 달라질 수 있음**
- ❖ 연구가 많이 이루어진 이미지 분야의 예시로 Pretext 종류 설명
  - ✓ Masked Prediction
  - ✓ Transformation Prediction
  - ✓ Instance (Contrastive) Discrimination



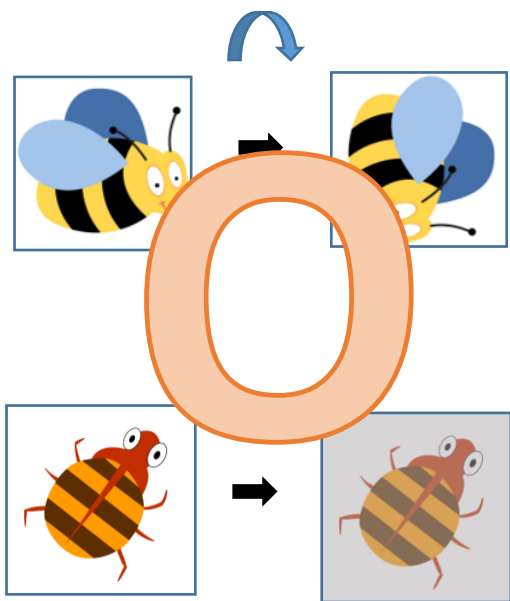
# Time Series Augmentations

- ❖ 이미지 분야에 다양한 Pretext 연구가 많이 이뤄지는 이유는 다양한 Augmentation 기법들이 존재하고 그 또한 직관적인 이해가 쉽고, 데이터 특성이 뒷받침 된다.



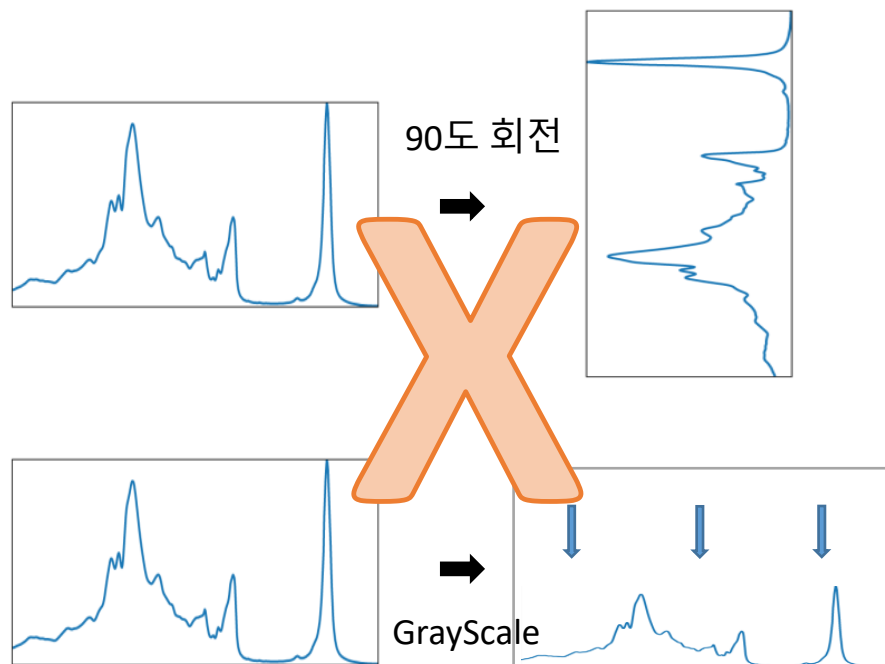
# Time Series Augmentations

- ❖ Q: 이미지/텍스트에서 사용하는 Augmentation 기법들을 시계열에 적용한다면?
- ❖ A: 어렵다. 시계열 데이터 특성상 시간에 따라 분포나 특성이 바뀐다.



이미지 데이터

VS



시계열 데이터

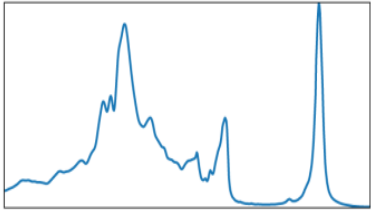
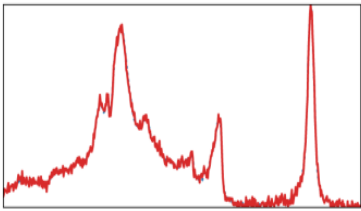
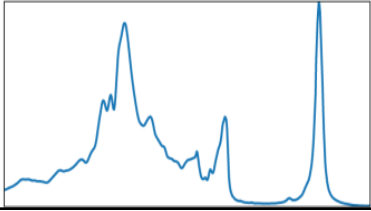
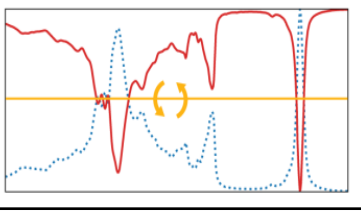
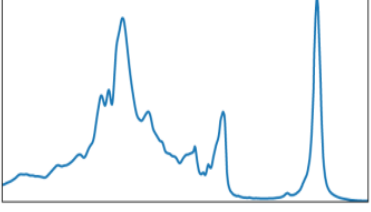
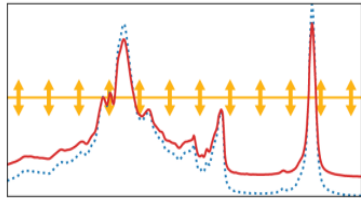
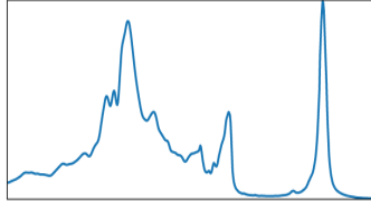
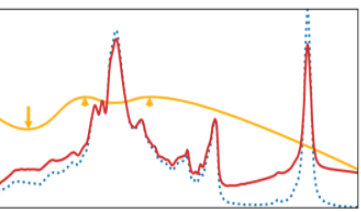
---

# **Time Series Augmentations**

---

# Time Series Augmentations

## ❖ 시계열 데이터 크기를 바꾸는 가공 방법(Y축 기준)

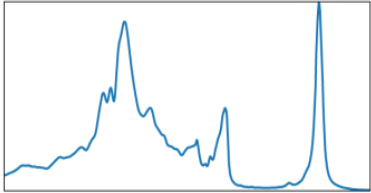
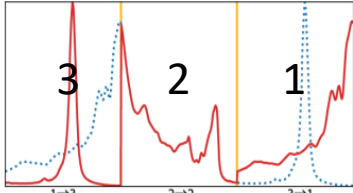
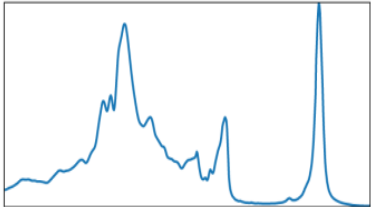
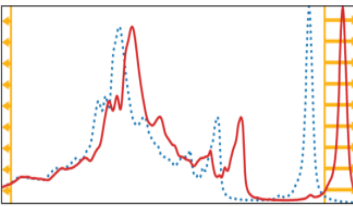
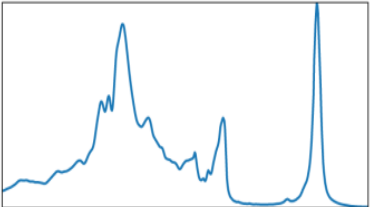
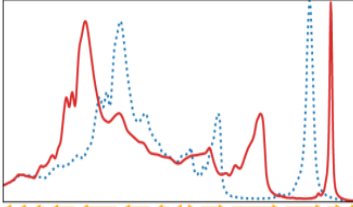
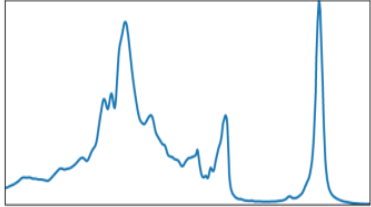
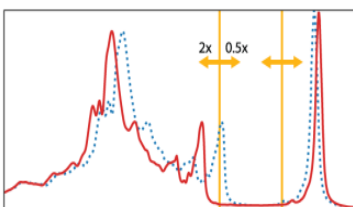
| 가공 방법             | 원래 데이터                                                                              | 가공 데이터                                                                               | 특징                                                                                            |
|-------------------|-------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------|
| Jitter            |    |    | 각 타임스탬프에 노이즈 적용<br>노이즈는 $\mathcal{N}(0, \sigma^2)$ 에서 샘플링<br>자주 쓰이나, 데이터가 노이즈를 포함하는 가정이 성립 필요 |
| Flipping          |    |    | 이상탐지, 분류에 자주 사용<br>위아래가 바뀔 수 있는 상황의 데이터                                                       |
| Scaling           |   |   | 시간 축이 아닌 데이터의 크기 또는 강도로 스케일 처리하며 그 비율은 $\mathcal{N}(1, \sigma^2)$ 에서 샘플링                      |
| Magnitude Warping |  |  | 변화가 적은 구간에 상승/하락을 만들어냄<br>하이퍼파라미터가 2개, 가공 데이터가 문제에 현실적이어야 함                                   |

[2] Wen, Q., Sun, L., Yang, F., Song, X., Gao, J., Wang, X., & Xu, H. (2020). Time series data augmentation for deep learning: A survey.

[3] Iwana, B. K., & Uchida, S. (2021). An empirical survey of data augmentation for time series classification with neural networks. Plos one, 16(7), e0254841.

# Time Series Augmentations

## ❖ 시계열 데이터의 시간을 가공 방법(x축 기준)

| 가공 방법          | 원래 데이터                                                                              | 가공 데이터                                                                               | 특징                                                                      |
|----------------|-------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------|-------------------------------------------------------------------------|
| Permutation    |    |    | 시간 의존성을 보존하지 않음<br>센서 데이터 분류에 사용된 적이 있음                                 |
| Window Slicing |    |    | 이미지에서 Cropping과 동일한 기법<br>윈도우 사이즈만큼 샘플을 잘라냄                             |
| Time Warping   |   |   | 시간축의 데이터를 왜곡을 시키는 기법                                                    |
| Window Warping |  |  | Time Warping의 대표적인 방법중의 하나, 랜덤하게 특정 윈도우 구간을 2배로 스트레칭을 하고 어떤 구간은 1/2배 처리 |

[2] Wen, Q., Sun, L., Yang, F., Song, X., Gao, J., Wang, X., & Xu, H. (2020). Time series data augmentation for deep learning: A survey.

[3] Iwana, B. K., & Uchida, S. (2021). An empirical survey of data augmentation for time series classification with neural networks. Plos one, 16(7), e0254841.

# Time Series Augmentations

- ❖ 시계열 데이터 특성에 적합한 Augmentation 기법을 선택시 세심한 주의가 필요
  - ✓ 정상성
  - ✓ 계절성, 트렌드
  - ✓ 시간에 따라 데이터의 특성, 분포가 바뀜
- ❖ 시계열 분류 모델에 가공 데이터를 사용할 경우 정확도가 떨어짐을 볼 수 있음

| Model    | None<br>Ave. (%) | Jittering<br>Ave. (%) | <i>t</i> | Rotation<br>Ave. (%) | <i>t</i> | Scaling<br>Ave. (%) | <i>t</i> | Magnitude Warping<br>Ave. (%) | <i>t</i> |
|----------|------------------|-----------------------|----------|----------------------|----------|---------------------|----------|-------------------------------|----------|
| MLP      | 70.23±21.91      | 70.52±21.67           | 1.55     | 69.13±21.70          | -1.95*   | 70.24±21.97         | 0.05     | 69.43±22.50                   | -2.91*** |
| VGG      | 73.02±23.05      | 74.21±22.57           | 0.91     | 71.87±22.14          | -0.84    | 73.69±23.86         | 0.53     | 74.42±22.63                   | 1.45     |
| ResNet   | 81.39±17.19      | 80.87±18.08           | -0.87    | 78.01±19.93          | -3.31*** | 81.92±16.81         | 1.05     | 81.33±17.15                   | -0.08    |
| LSTM     | 53.46±28.10      | 54.72±27.10           | 1.09     | 49.64±26.93          | -2.95*** | 53.69±26.94         | 0.21     | 54.35±27.29                   | 0.93     |
| BLSTM    | 62.51±24.74      | 60.23±24.87           | -2.01**  | 57.76±24.76          | -3.51*** | 62.13±25.89         | -0.37    | 62.00±26.51                   | -0.44    |
| LSTM-FCN | 81.54±17.53      | 79.18±19.87           | -4.89*** | 78.87±19.37          | -3.66*** | 80.73±18.37         | -1.74*   | 79.46±19.68                   | -3.15*** |

\*  $p < 0.1$ , \*\*  $p < 0.05$ , \*\*\*  $p < 0.01$

| Model    | Permutation<br>Ave. (%) | <i>t</i> | Slicing<br>Ave. (%) | <i>t</i> | Time Warping<br>Ave. (%) | <i>t</i> | Window Warping<br>Ave. (%) | <i>t</i> |
|----------|-------------------------|----------|---------------------|----------|--------------------------|----------|----------------------------|----------|
| MLP      | 68.17±22.17             | -3.89*** | 70.00±22.51         | -0.49    | 67.20±22.31              | -5.64*** | 69.73±22.50                | -0.87    |
| VGG      | 73.75±22.00             | 0.81     | 76.33±24.47         | 2.68**   | 75.50±20.82              | 1.86*    | 75.88±21.63                | 2.57**   |
| ResNet   | 80.67±18.31             | -1.16    | 81.51±17.64         | 0.24     | 79.62±19.20              | -2.25**  | 82.32±16.93                | 1.99**   |
| LSTM     | 49.65±26.90             | -3.56*** | 52.44±27.81         | -0.96    | 51.09±27.14              | -2.28**  | 54.41±28.23                | 1.73*    |
| BLSTM    | 60.89±24.73             | -1.38    | 64.60±24.01         | 1.93*    | 61.42±25.22              | -0.97    | 62.63±24.40                | 0.10     |
| LSTM-FCN | 81.25±17.22             | -0.74    | 80.30±19.27         | -1.67*   | 78.82±18.69              | -4.66*** | 79.73±19.35                | -1.94*   |

\*  $p < 0.1$ , \*\*  $p < 0.05$ , \*\*\*  $p < 0.01$

[2] Wen, Q., Sun, L., Yang, F., Song, X., Gao, J., Wang, X., & Xu, H. (2020). Time series data augmentation for deep learning: A survey.

[3] Iwana, B. K., & Uchida, S. (2021). An empirical survey of data augmentation for time series classification with neural networks. *PLoS one*, 16(7), e0254841.

---

# **Concrete Representative Methods**

---

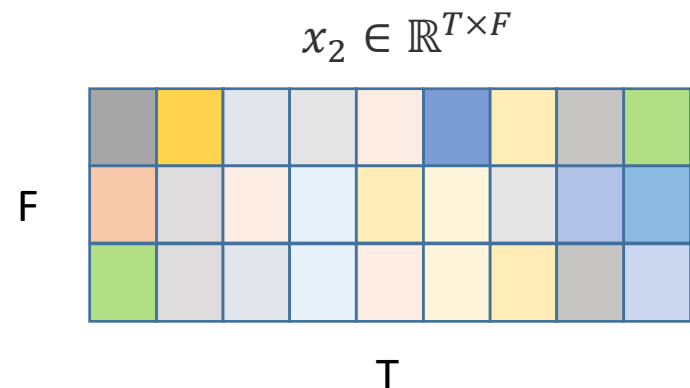
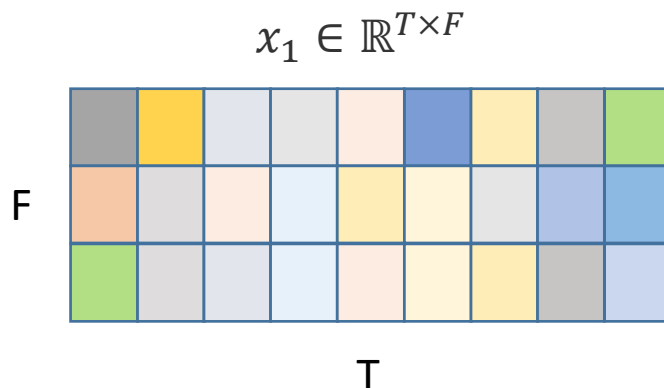
# Time Series Pretexts 종류

---

- ❖ Masked Prediction : 이미지 분야와 마찬가지로 결측치를 찾는 Pretext 정의
  - ✓ 마스킹 전략을 데이터의 특성에 따라 잘 수립할 필요
    - 얼마큼, 언제, 어디에, 마스크 예측을 위한 입력 데이터)
    - 너무 많은 마스킹은 추론이 불가능하고, 너무 적은 마스킹은 쉬운 문제로 전략
  - ✓ A Transformer-based Framework for Multivariate Time Series Representation Learning(TST)
  
- ❖ Transformation Prediction : 데이터 특성을 이용하여 Pretext 정의
  - ✓ 시계열 데이터의 순차성 정보를 활용하여 문제 정의를 할 수 있음
    - 예를 들면, 순서를 맞추는 문제 같은 Pretext가 가능
  - ✓ Self-supervised representation learning from electroencephalography signal

# Time Series Pretexts 종류

- ❖ Instance (Contrastive) Discrimination : 인스턴스 간 대조 학습을 통하여 인스턴스 차이를 학습, 시계열에서는 이 인스턴스의 개념이 여러 레벨로 나뉜다
  - ✓ 샘플 전체를 하나의 단위로 보는 경우
    - : Contrastive Representation Learning for Electroencephalogram Classification(SeqCLR)
  - ✓ 샘플 내의 일정 구간을 하나의 단위로 바라보는 경우
    - : Time-Series Representation Learning via Temporal and Contextual Contrasting(TS-TCC)
  - ✓ 타임스탬프 단위로 바라보는 경우
    - : TS2Vec : Towards Universal Representation of Time Series(TS2Vec)



# Time Series Pretexts 종류

❖ Instance (Contrastive) Discrimination : 인스턴스 간 대조 학습을 통하여 인스턴스 차이를 학습, 시계열에서는 이 인스턴스의 개념이 여러 레벨로 나뉜다

✓ 샘플 전체를 하나의 단위로 보는 경우

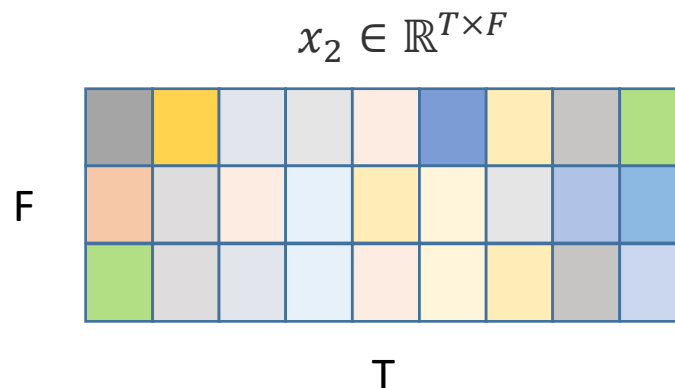
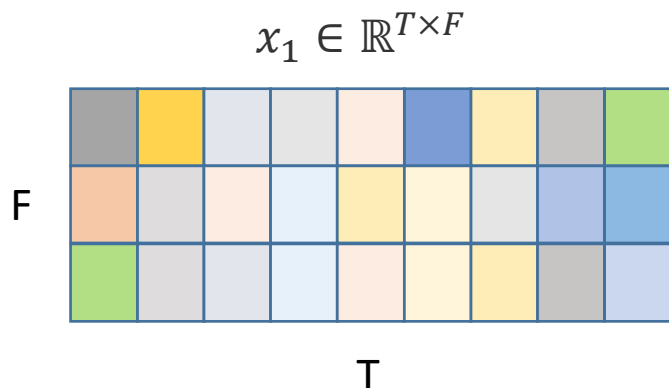
: **Contrastive Representation Learning for Electroencephalogram Classification(SeqCLR)**

✓ 샘플 내의 일정 구간을 하나의 단위로 바라보는 경우

: Time-Series Representation Learning via Temporal and Contextual Contrasting(TS-TCC)

✓ 타임스탬프 단위로 바라보는 경우

: TS2Vec : Towards Universal Representation of Time Series(TS2Vec)



# Time Series Pretexts 종류

❖ Instance (Contrastive) Discrimination : 인스턴스 간 대조 학습을 통하여 인스턴스 차이를 학습, 시계열에서는 이 인스턴스의 개념이 여러 레벨로 나뉜다

✓ 샘플 전체를 하나의 단위로 보는 경우

: **Contrastive Representation Learning for Electroencephalogram Classification(SeqCLR)**

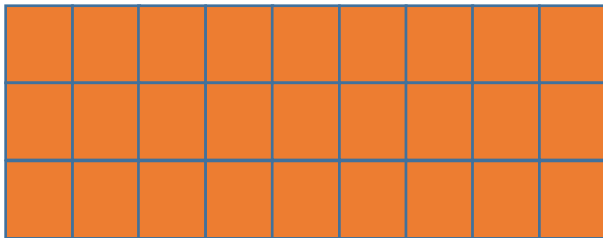
✓ 샘플 내의 일정 구간을 하나의 단위로 바라보는 경우

: Time-Series Representation Learning via Temporal and Contextual Contrasting(TS-TCC)

✓ 타임스탬프 단위로 바라보는 경우

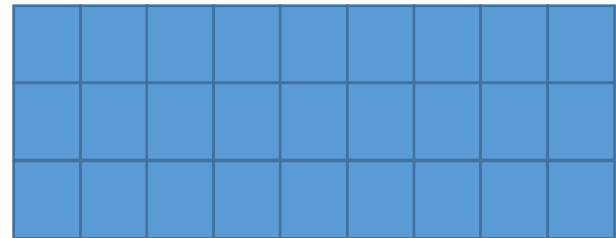
: TS2Vec : Towards Universal Representation of Time Series(TS2Vec)

$$x_1 \in \mathbb{R}^{T \times F}$$



**Positive**

$$x_2 \in \mathbb{R}^{T \times F}$$



# Time Series Pretexts 종류

❖ Instance (Contrastive) Discrimination : 인스턴스 간 대조 학습을 통하여 인스턴스 차이를 학습, 시계열에서는 이 인스턴스의 개념이 여러 레벨로 나뉜다

✓ 샘플 전체를 하나의 단위로 보는 경우

: **Contrastive Representation Learning for Electroencephalogram Classification(SeqCLR)**

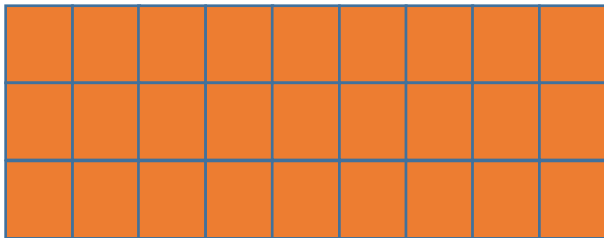
✓ 샘플 내의 일정 구간을 하나의 단위로 바라보는 경우

: Time-Series Representation Learning via Temporal and Contextual Contrasting(TS-TCC)

✓ 타임스탬프 단위로 바라보는 경우

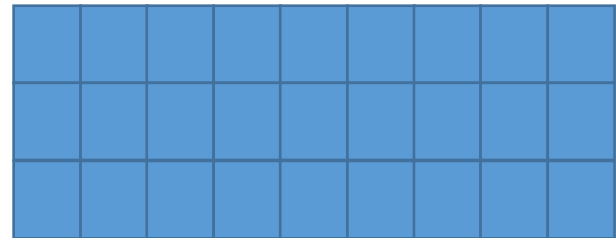
: TS2Vec : Towards Universal Representation of Time Series(TS2Vec)

$$x_1 \in \mathbb{R}^{T \times F}$$



**Negative**

$$x_2 \in \mathbb{R}^{T \times F}$$



# Time Series Pretexts 종류

❖ Instance (Contrastive) Discrimination : 인스턴스 간 대조 학습을 통하여 인스턴스 차이를 학습, 시계열에서는 이 인스턴스의 개념이 여러 레벨로 나뉜다

✓ 샘플 전체를 하나의 단위로 보는 경우

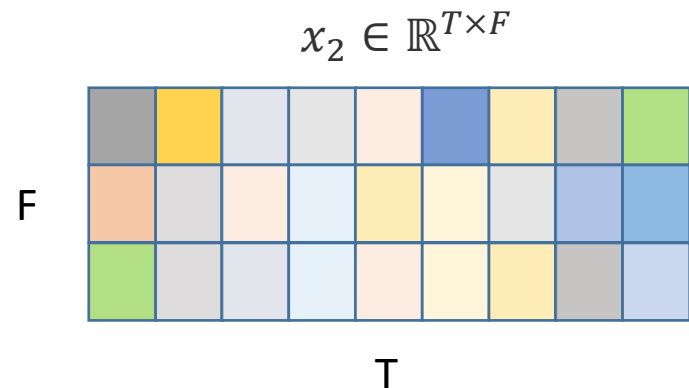
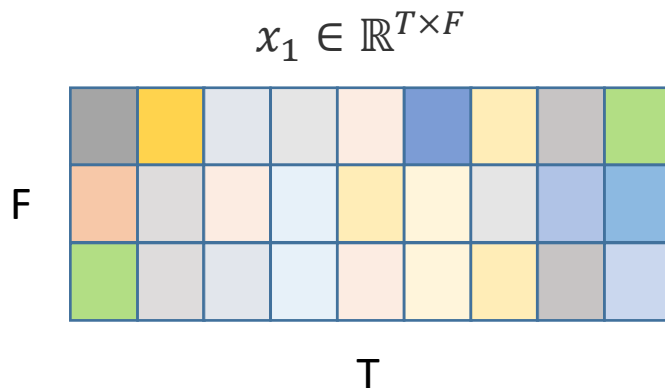
: Contrastive Representation Learning for Electroencephalogram Classification(SeqCLR)

✓ 샘플 내의 일정 구간을 하나의 단위로 바라보는 경우

: Time-Series Representation Learning via Temporal and Contextual Contrasting(TS-TCC)

✓ 타임스탬프 단위로 바라보는 경우

: TS2Vec : Towards Universal Representation of Time Series(TS2Vec)



# Time Series Pretexts 종류

❖ Instance (Contrastive) Discrimination : 인스턴스 간 대조 학습을 통하여 인스턴스 차이를 학습, 시계열에서는 이 인스턴스의 개념이 여러 레벨로 나뉜다

✓ 샘플 전체를 하나의 단위로 보는 경우

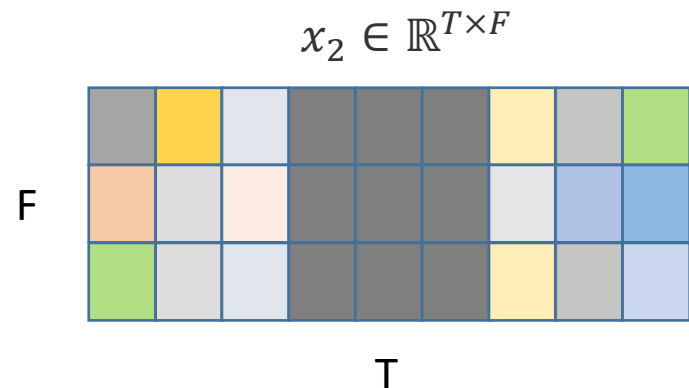
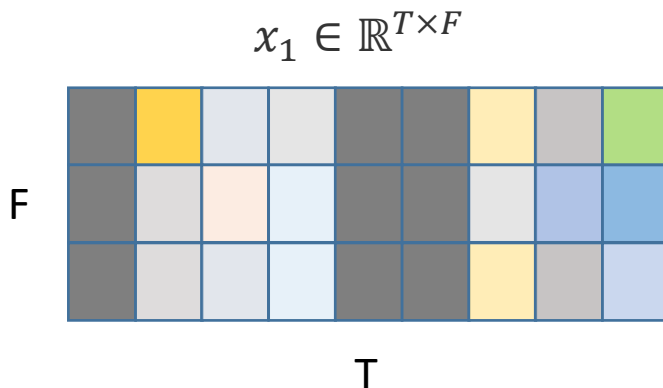
: Contrastive Representation Learning for Electroencephalogram Classification(SeqCLR)

✓ 샘플 내의 일정 구간을 하나의 단위로 바라보는 경우

: Time-Series Representation Learning via Temporal and Contextual Contrasting(TS-TCC)

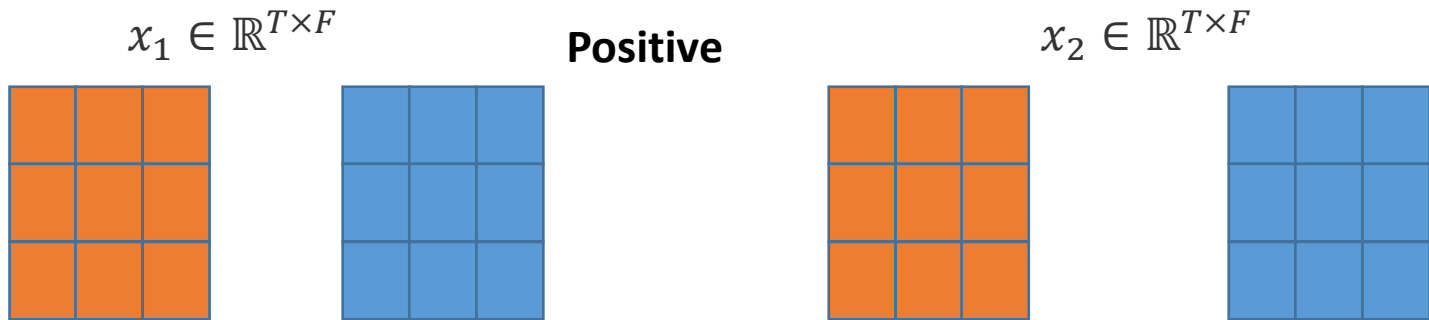
✓ 타임스탬프 단위로 바라보는 경우

: TS2Vec : Towards Universal Representation of Time Series(TS2Vec)



# Time Series Pretexts 종류

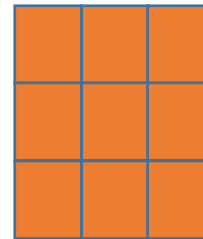
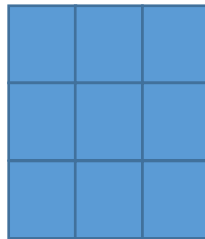
- ❖ Instance (Contrastive) Discrimination : 인스턴스 간 대조 학습을 통하여 인스턴스 차이를 학습, 시계열에서는 이 인스턴스의 개념이 여러 레벨로 나뉜다
  - ✓ 샘플 전체를 하나의 단위로 보는 경우
    - : Contrastive Representation Learning for Electroencephalogram Classification(SeqCLR)
  - ✓ 샘플 내의 일정 구간을 하나의 단위로 바라보는 경우
    - : **Time-Series Representation Learning via Temporal and Contextual Contrasting(TS-TCC)**
  - ✓ 타임스탬프 단위로 바라보는 경우
    - : TS2Vec : Towards Universal Representation of Time Series(TS2Vec)



# Time Series Pretexts 종류

- ❖ Instance (Contrastive) Discrimination : 인스턴스 간 대조 학습을 통하여 인스턴스 차이를 학습, 시계열에서는 이 인스턴스의 개념이 여러 레벨로 나뉜다
  - ✓ 샘플 전체를 하나의 단위로 보는 경우
    - : Contrastive Representation Learning for Electroencephalogram Classification(SeqCLR)
  - ✓ 샘플 내의 일정 구간을 하나의 단위로 바라보는 경우
    - : **Time-Series Representation Learning via Temporal and Contextual Contrasting(TS-TCC)**
  - ✓ 타임스탬프 단위로 바라보는 경우
    - : TS2Vec : Towards Universal Representation of Time Series(TS2Vec)

**Negative**



# Time Series Pretexts 종류

❖ Instance (Contrastive) Discrimination : 인스턴스 간 대조 학습을 통하여 인스턴스 차이를 학습, 시계열에서는 이 인스턴스의 개념이 여러 레벨로 나뉜다

✓ 샘플 전체를 하나의 단위로 보는 경우

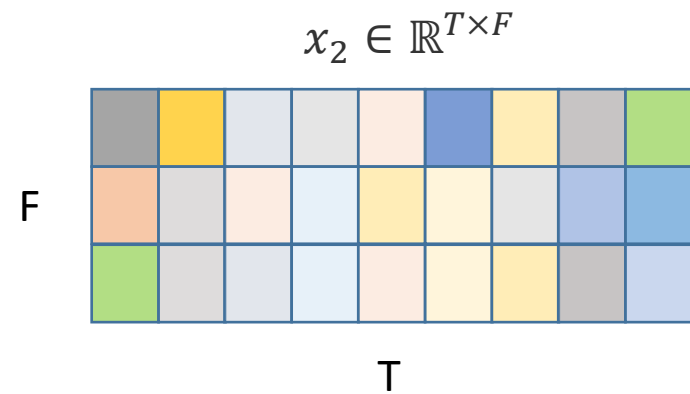
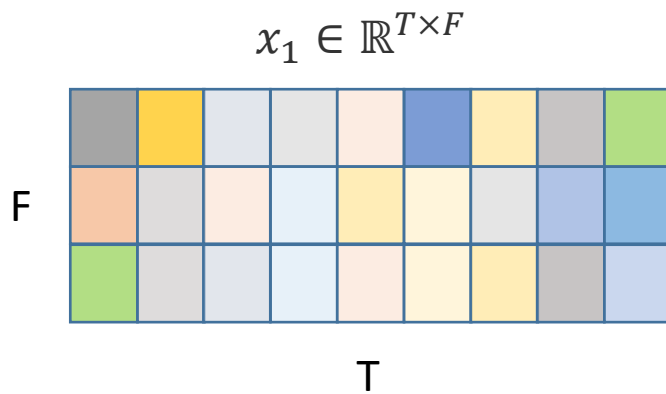
: Contrastive Representation Learning for Electroencephalogram Classification(SeqCLR)

✓ 샘플 내의 일정 구간을 하나의 단위로 바라보는 경우

: Time-Series Representation Learning via Temporal and Contextual Contrasting(TS-TCC)

✓ 타임스탬프 단위로 바라보는 경우

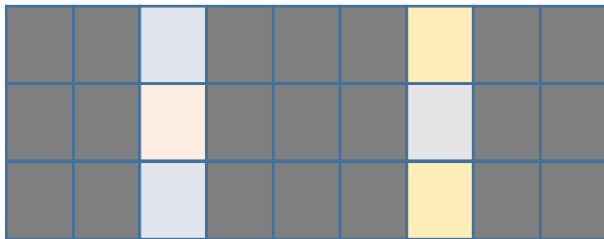
: TS2Vec : Towards Universal Representation of Time Series(TS2Vec)



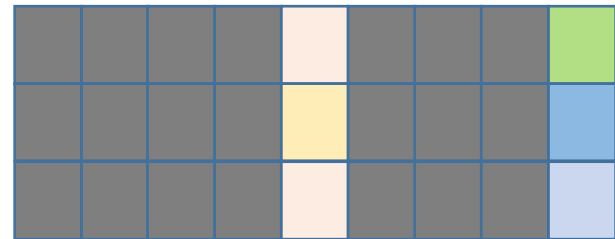
# Time Series Pretexts 종류

- ❖ Instance (Contrastive) Discrimination : 인스턴스 간 대조 학습을 통하여 인스턴스 차이를 학습, 시계열에서는 이 인스턴스의 개념이 여러 레벨로 나뉜다
  - ✓ 샘플 전체를 하나의 단위로 보는 경우
    - : Contrastive Representation Learning for Electroencephalogram Classification(SeqCLR)
  - ✓ 샘플 내의 일정 구간을 하나의 단위로 바라보는 경우
    - : Time-Series Representation Learning via Temporal and Contextual Contrasting(TS-TCC)
  - ✓ 타임스탬프 단위로 바라보는 경우
    - : **TS2Vec : Towards Universal Representation of Time Series(TS2Vec)**

$$x_1 \in \mathbb{R}^{T \times F}$$

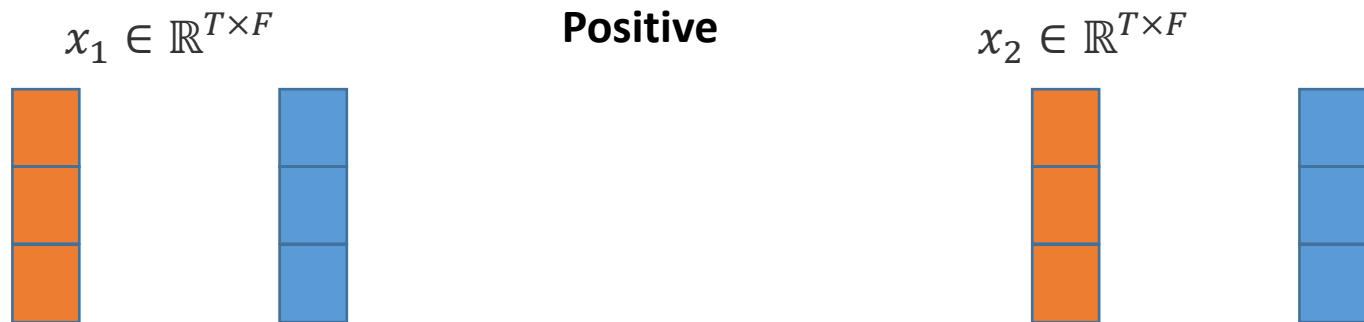


$$x_2 \in \mathbb{R}^{T \times F}$$



# Time Series Pretexts 종류

- ❖ Instance (Contrastive) Discrimination : 인스턴스 간 대조 학습을 통하여 인스턴스 차이를 학습, 시계열에서는 이 인스턴스의 개념이 여러 레벨로 나뉜다
  - ✓ 샘플 전체를 하나의 단위로 보는 경우
    - : Contrastive Representation Learning for Electroencephalogram Classification(SeqCLR)
  - ✓ 샘플 내의 일정 구간을 하나의 단위로 바라보는 경우
    - : Time-Series Representation Learning via Temporal and Contextual Contrasting(TS-TCC)
  - ✓ 타임스탬프 단위로 바라보는 경우
    - : **TS2Vec : Towards Universal Representation of Time Series(TS2Vec)**



# Time Series Pretexts 종류

- ❖ Instance (Contrastive) Discrimination : 인스턴스 간 대조 학습을 통하여 인스턴스 차이를 학습, 시계열에서는 이 인스턴스의 개념이 여러 레벨로 나뉜다
  - ✓ 샘플 전체를 하나의 단위로 보는 경우
    - : Contrastive Representation Learning for Electroencephalogram Classification(SeqCLR)
  - ✓ 샘플 내의 일정 구간을 하나의 단위로 바라보는 경우
    - : Time-Series Representation Learning via Temporal and Contextual Contrasting(TS-TCC)
  - ✓ 타임스탬프 단위로 바라보는 경우
    - : **TS2Vec : Towards Universal Representation of Time Series(TS2Vec)**

$$x_1 \in \mathbb{R}^{T \times F}$$



**Negative**

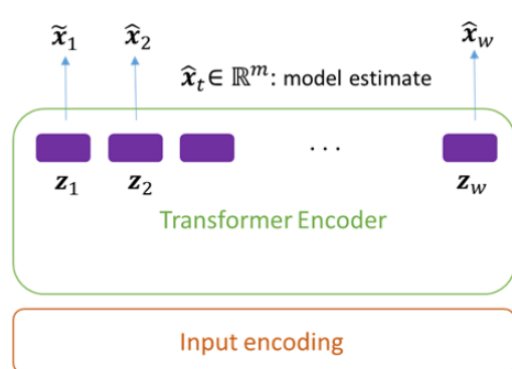
$$x_2 \in \mathbb{R}^{T \times F}$$



# Masked Prediction Pretext

## A Transformer-based Framework for Multivariate Time Series Representation Learning(TST)

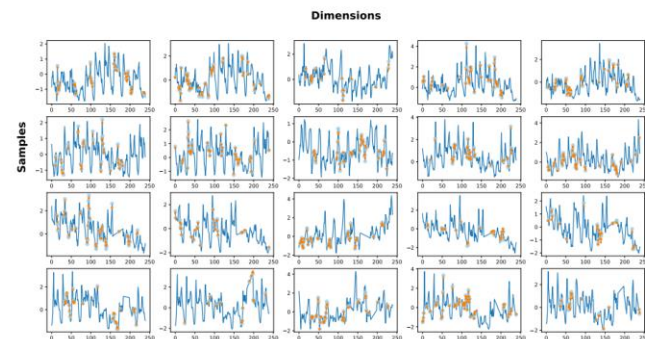
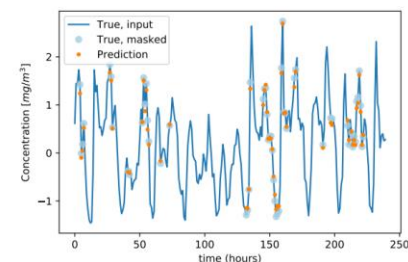
- ❖ 트랜스포머 기반 인코더로 사용하여 다변량 시계열 데이터에 결측치를 임의로 만들고 이를 예측하는 Pretext 기법 제안(Mask Prediction)
- ❖ Markov chain의 상태 변환 확률에 따르도록 마스킹 전략을 취함
- ❖ 유일하게 사후학습 작업으로 결측치 Imputation 또한 제시함



$$\mathcal{L}_{MSE} = \frac{1}{|M|} \sum_{(t,i) \in M} (\hat{x}(t,i) - x(t,i))^2$$

트랜스포머 구조와 동일

마스킹 전략에 따라  
채널별 별도 마스킹 처리



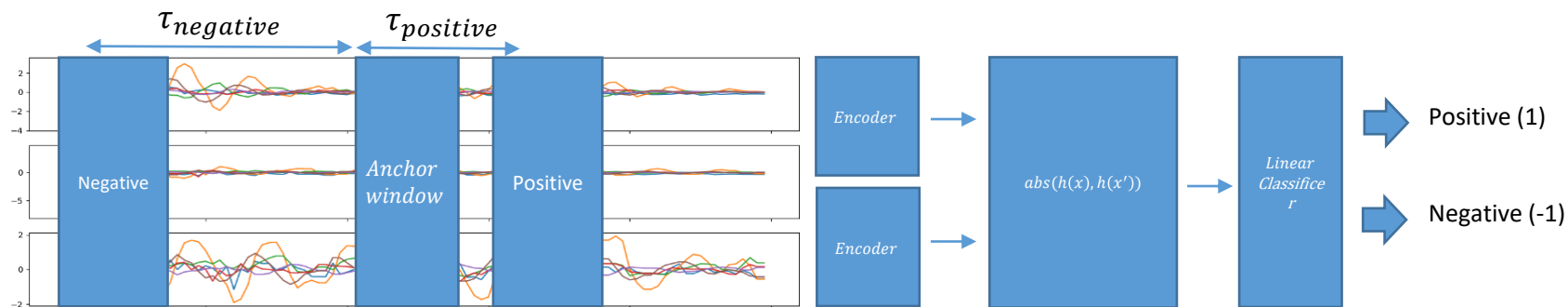
[5] Zerveas, G., Jayaraman, S., Patel, D., Bhamidipaty, A., & Eickhoff, C. (2021, August). A transformer-based framework for multivariate time series representation learning. In Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining (pp. 2114-2124).

# Transformation Prediction Pretext

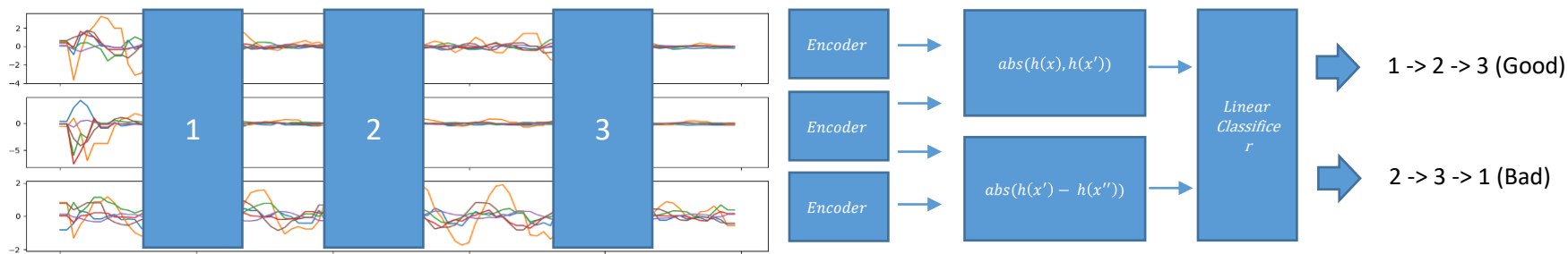
## Self-Supervised Representation Learning from Electroencephalography Signals

- ❖ 시간의 순차성 정보를 활용한 Pre-Text 기법 2가지를 제안
- ❖ EEG 데이터셋에 최초로 적용

### ① Relative Positioning (RP)



### ② Temporal Shuffling (TS)



[7] Banville, H., Albuquerque, I., Hyvärinen, A., Moffat, G., Engemann, D. A., & Gramfort, A. (2019, October). Self-supervised representation learning from electroencephalography signals. In 2019 IEEE 29th International Workshop on Machine Learning for Signal Processing (MLSP) (pp. 1-6). IEEE.

# Instance Contrastive Discrimination Pretext

## Contrastive Representation Learning for Electroencephalogram Classification (SeqCLR)

❖ 사용한 데이터 : 다변량 시계열 데이터인 EEG

❖ 데이터는 많지만 노이즈가 많고 해석이 어렵고 머신러닝에 적합하지 않음

❖ 공개된 데이터셋은 많지만 각각 채널 수, 샘플링 비율, 센서 종류가 다름

❖ Pretext

❖ Instance (Contrastive) Discrimination 통한 기반 대조 학습(SimCLR)

❖ 샘플 전체에 대한 차이를 학습하므로 분류에 적합

❖ 특징

❖ EEG 전문가의 도움으로 전처리 및 EEG에 적합한 Augmentation 제안

❖ 채널 차분 조합으로  $N \times N - 1$ 의 개수로 늘리며 샘플 효율성 증대

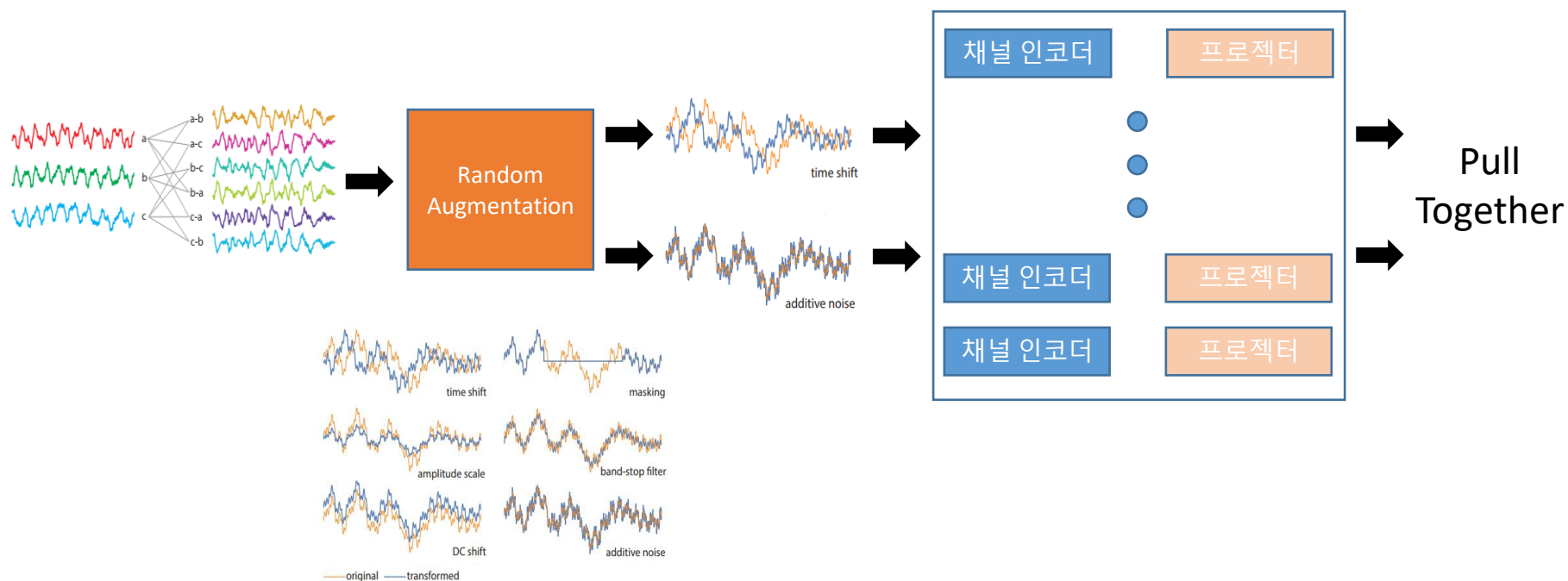
❖ 3개 서로 다른 태스크를 위한 사후 학습에서 성능 검증

❖ 출처가 다른 6개 데이터셋을 하나의 모델로 사전 학습하여 성능 증가

# Instance Contrastive Discrimination Pretext

## Contrastive Representation Learning for Electroencephalogram Classification (SeqCLR)

- ❖ 채널 간 차분 데이터를 활용, 채널 별 Encoder 별도 인코딩 학습
- ❖ 서로 다른 출처 여섯 종류의 EEG 데이터셋을 한꺼번에 사전 학습에 활용
- ❖ 시계열 데이터에 최초로 SimCLR 적용하여 사전학습 진행

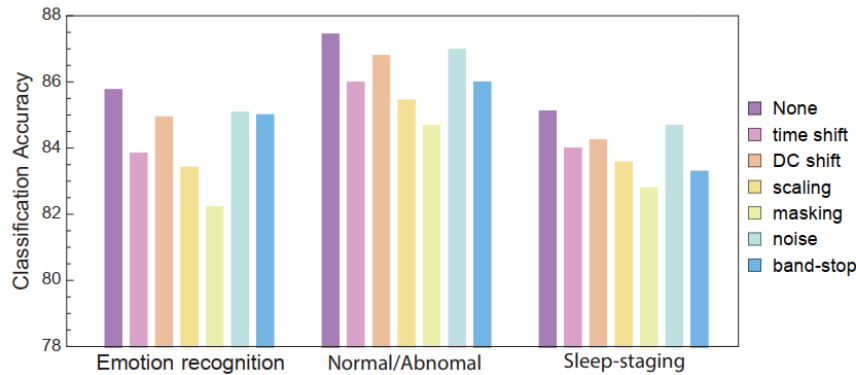


[8] Mohsenvand, M. N., Izadi, M. R., & Maes, P. (2020, November). Contrastive representation learning for electroencephalogram classification. In Machine Learning for Health (pp. 238-253). PMLR.

# Instance Contrastive Discrimination Pretext

## Contrastive Representation Learning for Electroencephalogram Classification (SeqCLR)

- ❖ 여러 데이터셋을 한꺼번에 사용하고, 채널 간 차분 데이터를 사용한 것이 효과적
- ❖ 6개의 가공 방법을 모두 사용하였을 경우 가장 높은 성능을 기록



| Channel recombination | Dataset fusion | SEED         | TUH          | SleepEDF     |
|-----------------------|----------------|--------------|--------------|--------------|
| ✗                     | ✗              | 78.93        | 79.12        | 77.72        |
| ✓                     | ✗              | 83.01        | 83.78        | 81.10        |
| ✗                     | ✓              | 80.23        | 83.44        | 79.59        |
| ✓                     | ✓              | <b>84.11</b> | <b>86.27</b> | <b>83.05</b> |

| Model                 | Accuracy     |              |              |              |
|-----------------------|--------------|--------------|--------------|--------------|
| Percentage of labels  | 1%           | 10%          | 50%          | 100%         |
| RGNN                  | -            | -            | -            | 85.30        |
| BiHDM                 | -            | -            | -            | 85.40        |
| CPC                   | 69.17        | 76.33        | 79.98        | 81.12        |
| RP                    | 67.76        | 74.29        | 77.95        | 80.39        |
| TS                    | 69.73        | 78.27        | 81.66        | 82.10        |
| SeqCLR - C            | 77.09        | 81.01        | 83.73        | 84.11        |
| SeqCLR - R            | 76.52        | 79.04        | 81.45        | 83.78        |
| fine-tuned SeqCLR - C | <b>79.04</b> | <b>83.12</b> | <b>85.21</b> | <b>85.77</b> |
| fine-tuned SeqCLR - R | 78.18        | 82.93        | 84.00        | 85.25        |

[8] Mohsenvand, M. N., Izadi, M. R., & Maes, P. (2020, November). Contrastive representation learning for electroencephalogram classification. In Machine Learning for Health (pp. 238-253). PMLR.

# Instance Contrastive Discrimination Pretext

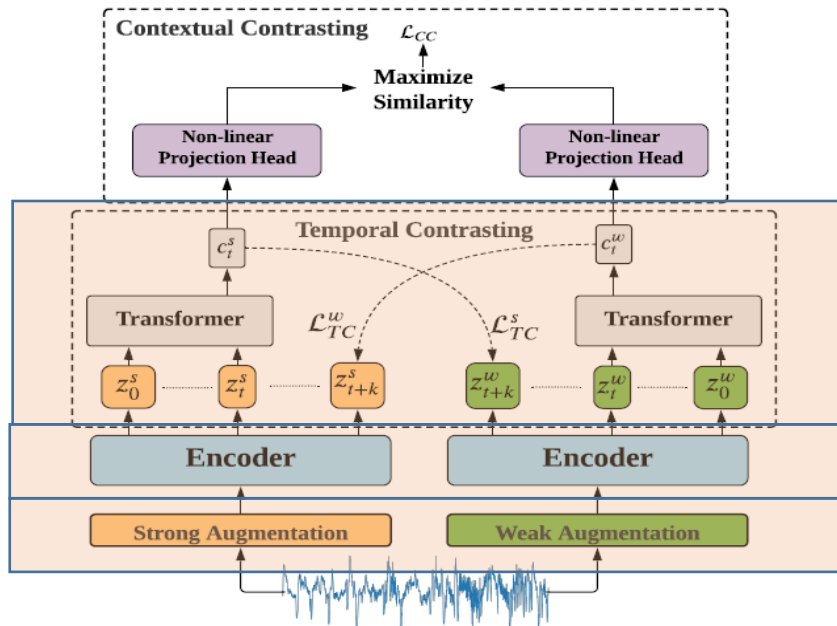
## Time-Series Representation Learning via Temporal and Contextual Contrasting (TS-TCC)

- ❖ EEG 에 국한되지 않는 일반 시계열 데이터에 적합한 프레임워크 제시
- ❖ 순차성과 의미 관점의 대조학습을 동시 진행
- ❖ 시계열 대조학습을 위한 간단하고 쉽게 적용할 수 있는 가공 방법 제시
- ❖ 자기지도학습 성능 중에 지도학습 대비 좋은 성능 보임 (Human Action Recognition, Epsilepsy)
- ❖ 시계열 데이터 특성 상, 같은 데이터에서 2개의 다른 가공 방법으로 생성되는 샘플이 과연 같은 샘플로 바라볼 수 있는가에 의문
  - ✓ 여러 Augmentation 기법 중 랜덤하게 선택하는게 아닌 Positive 샘플 쌍을 위한 2개의 Augmentation 기법으로 고정
    - Weak : Jitter + Scaling
    - Strong : Permutation Jitter
  - ✓ 데이터 가공을 위한 파라미터는 데이터에 따라 신중히 결정

# Instance Contrastive Discrimination Pretext

## Time-Series Representation Learning via Temporal and Contextual Contrasting (TS-TCC)

- ❖ Temporal Dependency에 대한 Robust representation을 위해 Tough Cross-View prediction Task를 제안
- ❖ Weak Augmentation의 지난 Latent 벡터를 이용하여 다른 View(Strong Augmentation)의 미래 시점 벡터 예측
- ❖ Robustness를 위하여 다른 종류의 Augmentation을 사용하지 않음.



$$L_{TC}^s = -\frac{1}{K} \sum_{k=1}^K \log \frac{\exp((W_k(c_t^w))^T z_{t+k}^s)}{\sum_{n \in \mathcal{N}_{t,k}} \exp((W_k(c_t^s))^T z_n^w)}$$

$$L_{TC}^w = -\frac{1}{K} \sum_{k=1}^K \log \frac{\exp((W_k(c_t^w))^T z_{t+k}^s)}{\sum_{n \in \mathcal{N}_{t,k}} \exp((W_k(c_t^w))^T z_n^s)}$$

(Conv1D + BatchNorm1D + MaxPool + ReLU) \* 3

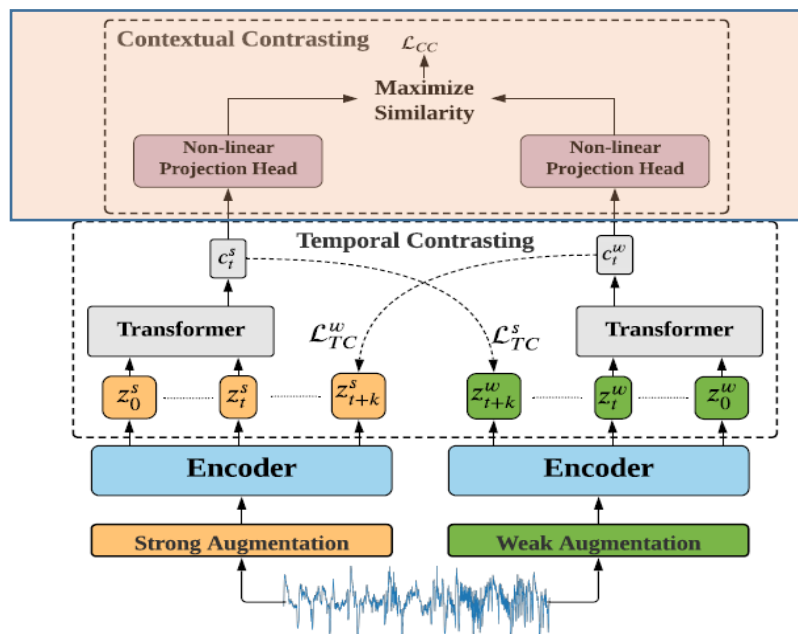
Weak Augmentation : Jittering + Scale

Strong Augmentation : Permutation + Jittering

# Instance Contrastive Discrimination Pretext

## Time-Series Representation Learning via Temporal and Contextual Contrasting (TS-TCC)

- ❖ 1단계 대조 학습 이후의 Context Vector를 다시 한번 대조학습
- ❖ 배치 내에서 Original Sample이 같은 건은 Positive Sample, 다른 모든 샘플들은 Negative Sample이라고 정의



N개 배치에서 Strong, Weak Augmentation을 하므로 2N 크기에 데이터에 대해서 추가 대조 학습 진행

Context vector  $c_t^i$ 와  $c_t^{i+}$ 를 Positive 로 보고 나머지 2N-2를 Negative sample로 바라봄.

$$L_{CC} = - \sum_{i=1}^K \log \frac{\exp(\text{sim}(c_t^i, c_t^{i+})/\tau)}{\sum_{m=1}^{2N} \mathbb{I}_{[m \neq i]} \exp(\text{sim}(c_t^i, c_t^m)/\tau)}$$

# Instance Contrastive Discrimination Pretext

## Time-Series Representation Learning via Temporal and Contextual Contrasting (TS-TCC)

- ❖ 샘플이 적은 상황에서도 모델이 강건함을 보임
- ❖ Ablation Study를 통해서 Temporal Contrasting module과 Contextual Contrasting module의 효과성 입증

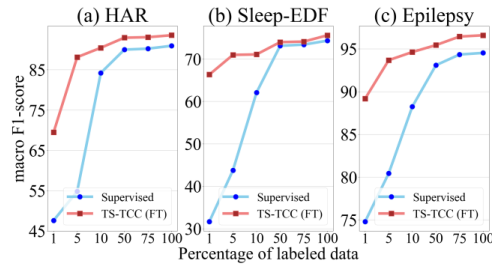


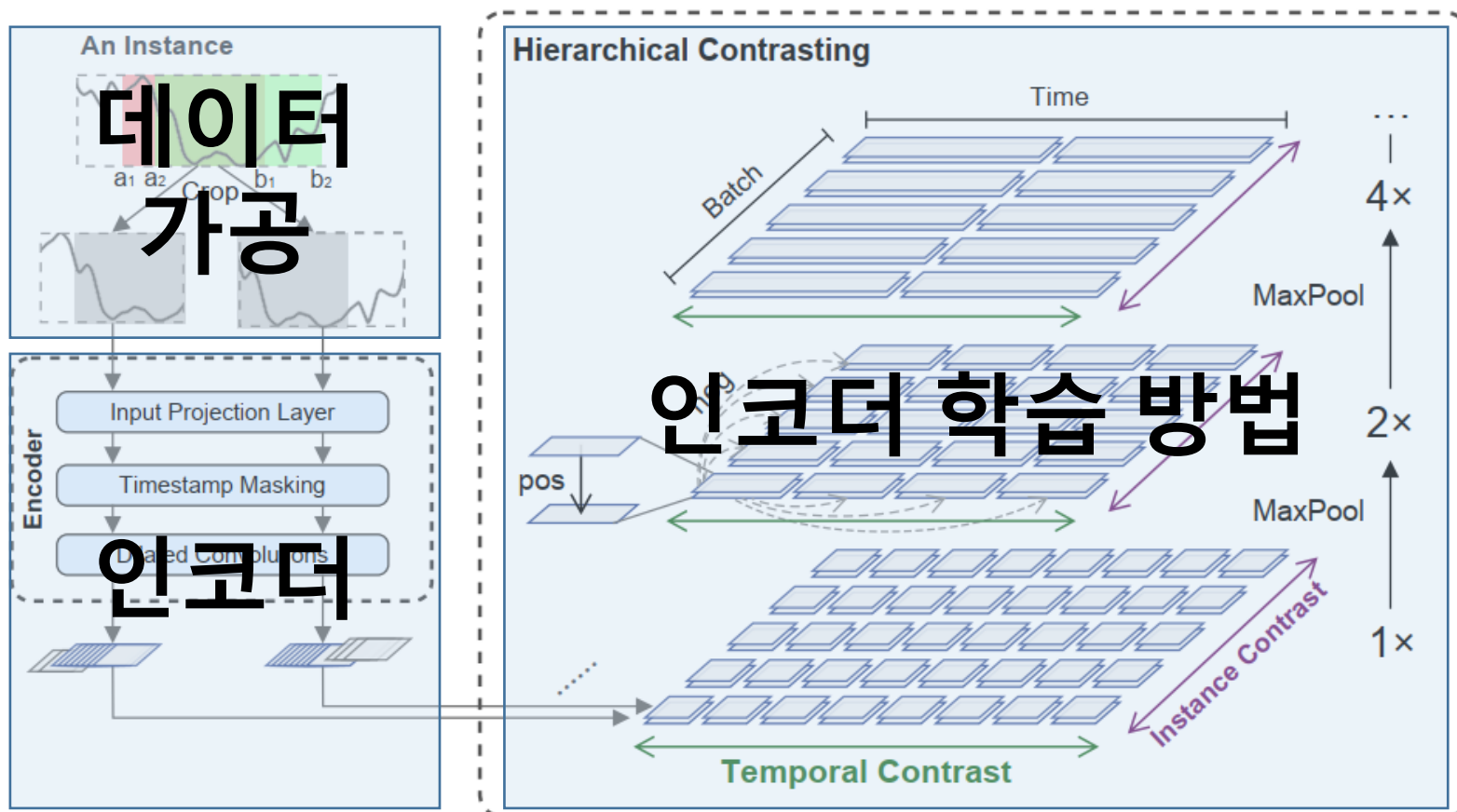
Figure 3: Comparison between supervised training vs. TS-TCC fine-tuning for different few-labeled data scenarios in terms of MF1.

|                          | HAR               |                   | Sleep-EDF         |                   | Epilepsy          |                   |
|--------------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|
| Component                | ACC               | MF1               | ACC               | MF1               | ACC               | MF1               |
| TC only                  | 82.76±1.50        | 82.17±1.64        | 80.55±0.39        | 70.99±0.86        | 94.39±1.19        | 90.93±1.41        |
| TC + X-Aug               | 87.86±1.33        | 87.91±1.09        | 81.58±1.70        | 71.88±1.71        | 95.56±0.24        | 92.57±0.29        |
| TS-TCC (TC + X-Aug + CC) | <b>90.37±0.34</b> | <b>90.38±0.39</b> | <b>83.00±0.71</b> | <b>73.57±0.74</b> | <b>97.23±0.10</b> | <b>95.54±0.08</b> |
| TS-TCC (Weak only)       | 76.55±3.59        | 75.14±4.66        | 80.90±1.87        | 72.51±1.74        | 97.18±0.17        | 95.47±0.31        |
| TS-TCC (Strong only)     | 60.23±3.31        | 56.15±4.14        | 78.55±2.94        | 68.05±1.87        | 97.14±0.23        | 95.39±0.29        |

# Instance Contrastive Discrimination Pretext

## TS2Vec : Towards Universal Representation of Time Series

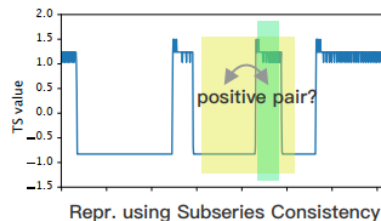
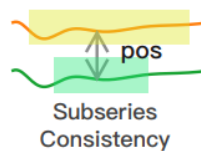
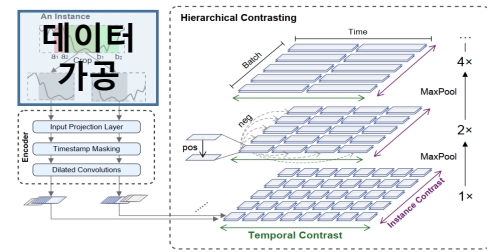
- ❖ 전체적인 모델 아키텍처는 데이터 가공 방법, 인코더 구조, 학습 방법으로 소개한다.



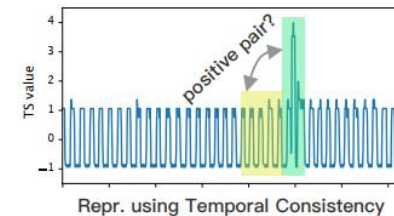
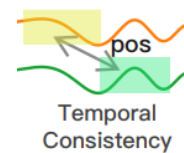
# Instance Contrastive Discrimination Pretext

## TS2Vec : Towards Universal Representation of Time Series

- ❖ Self Contrastive Learning에서 Positive Sample을 정의는 매우 중요
- ❖ 기존의 연구의 문제점을 제기함
  - ✓ Subseries consistency (Franceschi, Dieuleveut, and Jaggi 2019)
    - 한 샘플에서 부분 샘플링된 세그먼트들은 서로 컨텍스트가 일치
  - ✓ Temporal consistency (Tonekaboni, Eytan, and Goldenberg 2021) :
    - 인접한 세그먼트들을 서로 컨텍스트가 일치
  - ✓ Transformation consistency (Eldele et al. 2021)
    - 스케일, 순열등 다양한 변형을 통해도 컨텍스트가 유지한다는 강한 가정으로 시계열 데이터에 대해 적합하지 않음



Subseries consistency 사례



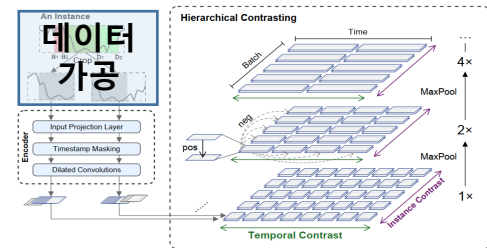
Temporal consistency 사례

# Instance Contrastive Discrimination Pretext

## TS2Vec : Towards Universal Representation of Time Series

### ❖ Contextual Consistency 개념 제안

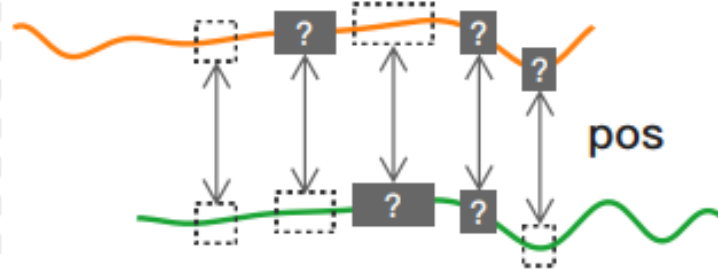
- ✓ 시계열 데이터 특성의 중요한 점인 magnitude가 변하지 않으며,
- ✓ 각각의 타임스탬프마다 강건한 학습된 피처를 향상시키도록 유도 가능하며, position-agnostic한 피처를 배우고 피처가 붕괴되는것을 돕는다고 함
- ✓ Timestamp Masking과 Random Cropping 둘다 학습단계에서만 적용
- ✓ 같은 타임스탬프의 임베딩 벡터를 각각 augmentation을 취해 Positive 샘플 구성하여 타임스탬프 레벨 마스킹 및 입력 데이터에 Cropping 적용하여 새로운 컨텍스트 생성



### ❖ 데이터 가공

- ✓ Timestamp Masking : Input Projection Layer를 통과한 latent vector  $z_i$  에 마스크 값  $m \in \{0,1\}^T$  을 적용하며, 베르누이 분포( $p = 0.5$ )로 매번 샘플링하여 마스킹
- ✓ Random Cropping : Cropping을 하되 2개 샘플이 겹치도록 샘플링하며, 겹치는 부분을 새로 만들 컨텍스트 벡터가 유사하도록 대조 학습.

### Contextual Consistency (ours)

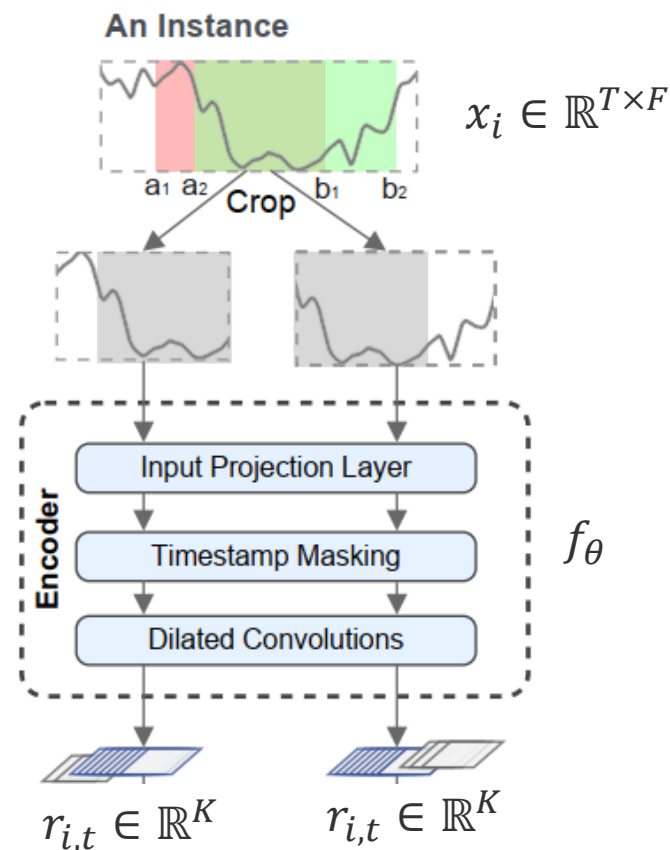


# Instance Contrastive Discrimination Pretext

인코더 학습 방법

## TS2Vec : Towards Universal Representation of Time Series

- ❖ 인코더는 3개의 모듈로 구성
  - ✓ Input Projection Layer
  - ✓ Timestamp Masking Module
  - ✓ Dilated Convolutions
- ❖ Input projection layer : 하나의 FC Layer로 구성되며, 잠재 벡터 생성
- ❖ Timestamp masking module : 랜덤 선택된 타임스탬프들 구간 마스킹 처리(원래 데이터에서 처리 안하는 이유는 단순히 마스킹 토큰 포함이 될 수 있어서라고 함)
- ❖ Dilated Convolutions
  - ✓ 10개 Residual 블록 통하여 각 타임스탬프의 컨텍스트 피처를 얻기 위해 사용
  - ✓ 각 블록은 2개의 1-D convolutional layers를 dilation parameter ( $2^l$  for  $l$ -th block)을 설정하여 넓은 receptive field를 가능하게 함

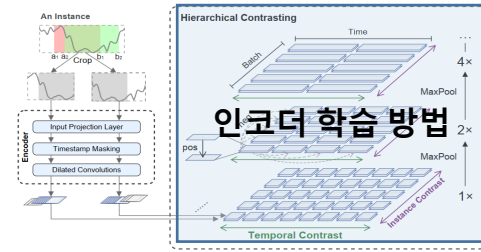


# Instance Contrastive Discrimination Pretext

## TS2Vec : Towards Universal Representation of Time Series

### ❖ hierarchical contrastive loss 제안

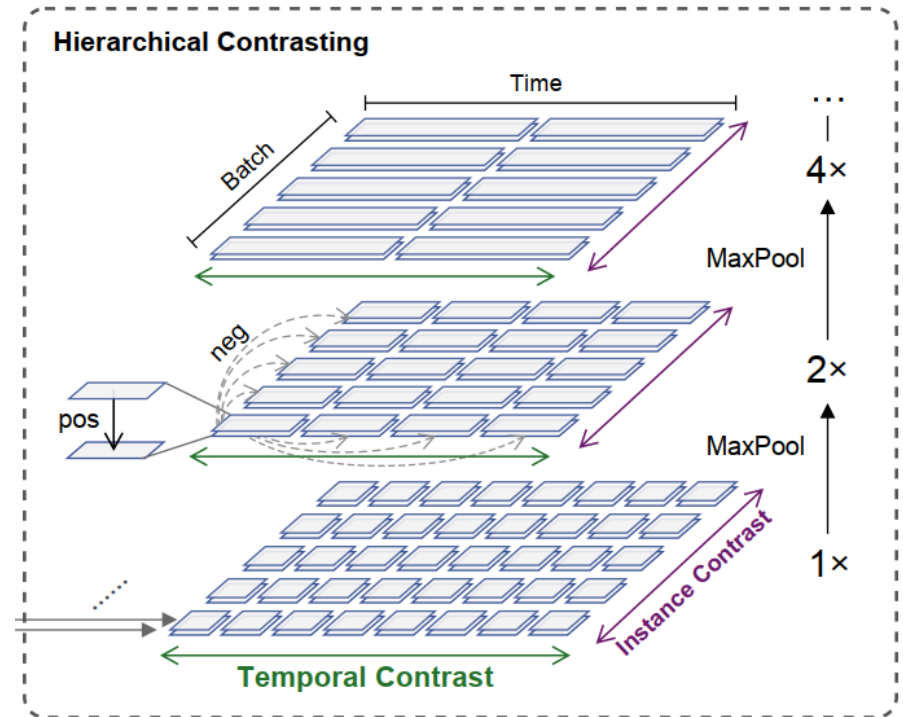
- ❖ 다양한 스케일에서 좋은 피처를 학습하기 위해 시간 차원에 따라 학습된 피처에 Max Pooling을 2배씩 늘려가며 매번 시간과 샘플차이의 대조 학습 손실함 수 반복 적용
- ❖ 타임스탬프레벨에서 전체 길이의 컨텍스트 학습 가능하여 포괄적인 피처 학습 가능
- ❖ Hierarchical Contrasting Module에서 모든 Granularity 레벨이 적용됨



Algorithm 1: Calculating the hierarchical contrastive loss

```

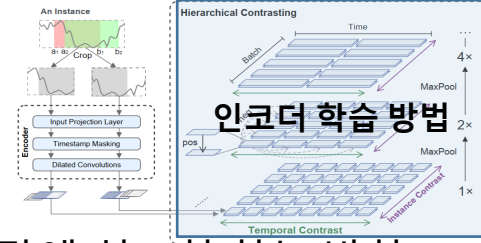
1: procedure HIERLOSS( $r, r'$ )
2:    $\mathcal{L}_{hier} \leftarrow \mathcal{L}_{dual}(r, r')$ ;
3:    $d \leftarrow 1$ ;
4:   while time_length( $r$ ) > 1 do
5:     // The maxpool1d operates along the time axis.
6:      $r \leftarrow \text{maxpool1d}(r, \text{kernel\_size} = 2)$ ;
7:      $r' \leftarrow \text{maxpool1d}(r', \text{kernel\_size} = 2)$ ;
8:      $\mathcal{L}_{hier} \leftarrow \mathcal{L}_{hier} + \mathcal{L}_{dual}(r, r')$ ;
9:      $d \leftarrow d + 1$ ;
10:  end while
11:   $\mathcal{L}_{hier} \leftarrow \mathcal{L}_{hier} / d$ ;
12:  return  $\mathcal{L}_{hier}$ 
13: end procedure
    
```



# Instance Contrastive Discrimination Pretext

## TS2Vec : Towards Universal Representation of Time Series

- ❖ 계층적으로 Temporal Contrastive Loss와 Instance-wise Contrastive Loss 값에 최소화 하는 방향으로 학습
- ❖ 계층 구조에 따라 다양한 스케일의 대조 학습 손실 값을 합산하여 계산되어짐



Time Contrastive Loss

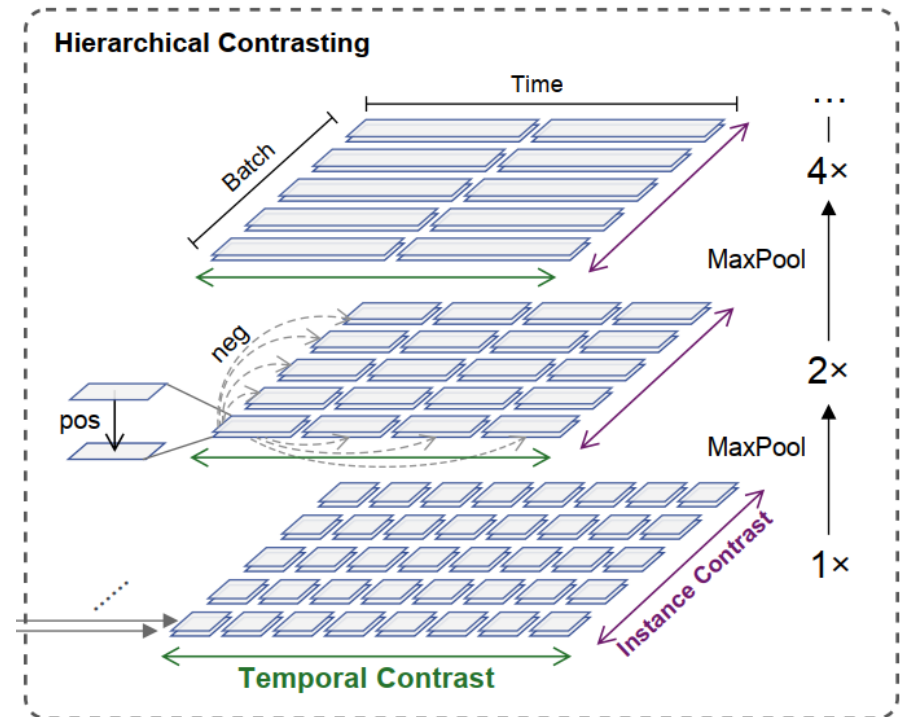
$$l_{temp}^{(i,t)} = -\log \frac{\exp(r_{i,t} \cdot r'_{i,t})}{\sum_{t' \in \Omega} (\exp(r_{i,t} \cdot r'_{i,t'}) + \mathbb{I}_{[i \neq j]} \exp(r_{i,t} \cdot r_{j,t}))}$$

Instance-wise Contrastive Loss

$$l_{inst}^{(i,t)} = -\log \frac{\exp(r_{i,t} \cdot r'_{i,t})}{\sum_{j=1}^B (\exp(r_{i,t} \cdot r'_{j,t}) + \mathbb{I}_{[i \neq j]} \exp(r_{i,t} \cdot r_{j,t}))}$$

Total Loss

$$\mathcal{L}_{dual} = \frac{1}{NT} \sum_i \sum_t (l_{temp}^{(i,t)} + l_{inst}^{(i,t)})$$



# Instance Contrastive Discrimination Pretext

## TS2Vec : Towards Universal Representation of Time Series

- ❖ 사후 학습으로 분류 태스크에 적용한 실험 결과
- ❖ UCR datasets(125개) 평균 2.4%, UEA datasets(29개) 평균 3.0% 성능 향상

| Method | 125 UCR datasets     |             |                       | 29 UEA datasets      |             |                       |
|--------|----------------------|-------------|-----------------------|----------------------|-------------|-----------------------|
|        | Avg. Acc.            | Avg. Rank   | Training Time (hours) | Avg. Acc.            | Avg. Rank   | Training Time (hours) |
| DTW    | 0.727                | 4.33        | –                     | 0.650                | 3.74        | –                     |
| TNC    | 0.761                | 3.52        | 228.4                 | 0.677                | 3.84        | 91.2                  |
| TST    | 0.641                | 5.23        | 17.1                  | 0.635                | 4.36        | 28.6                  |
| TS-TCC | 0.757                | 3.38        | 1.1                   | 0.682                | 3.53        | 3.6                   |
| T-Loss | 0.806                | 2.73        | 38.0                  | 0.675                | 3.12        | 15.1                  |
| TS2Vec | <b>0.830 (+2.4%)</b> | <b>1.82</b> | <b>0.9</b>            | <b>0.712 (+3.0%)</b> | <b>2.40</b> | <b>0.6</b>            |

Table 1: Time series classification results compared to other time series representation methods. The representation dimensions of TS2Vec, T-Loss, TS-TCC, TST and TNC are all set to 320 and under SVM evaluation protocol for fair comparison.

# Instance Contrastive Discrimination Pretext

## TS2Vec : Towards Universal Representation of Time Series

- ❖ Ablation Study를 통하여 각 모듈의 필요성 입증
- ❖ 타임스탬프 레벨의 피쳐 정보까지 잘 생성하고 계절 + 트렌드성을 잘 포착함

|                                | Avg. Accuracy  |
|--------------------------------|----------------|
| <b>TS2Vec</b>                  | <b>0.829</b>   |
| w/o Temporal Contrast          | 0.819 (-1.0%)  |
| w/o Instance Contrast          | 0.824 (-0.5%)  |
| w/o Hierarchical Contrast      | 0.812 (-1.7%)  |
| w/o Random Cropping            | 0.808 (-2.1%)  |
| w/o Timestamp Masking          | 0.820 (-0.9%)  |
| w/o Input Projection Layer     | 0.817 (-1.2%)  |
| <i>Positive Pair Selection</i> |                |
| Contextual Consistency         |                |
| → Temporal Consistency         | 0.807 (-2.2%)  |
| → Subseries Consistency        | 0.780 (-4.9%)  |
| <i>Augmentations</i>           |                |
| + Jitter                       | 0.814 (-1.5%)  |
| + Scaling                      | 0.814 (-1.5%)  |
| + Permutation                  | 0.796 (-3.3%)  |
| <i>Backbone Architectures</i>  |                |
| Dilated CNN                    |                |
| → LSTM                         | 0.779 (-5.0%)  |
| → Transformer                  | 0.647 (-18.2%) |

Table 5: Ablation results on 128 UCR datasets.

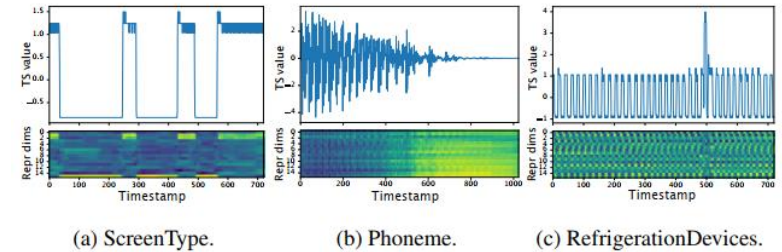


Figure 7: The heatmap visualization of the learned representations of TS2Vec over time.

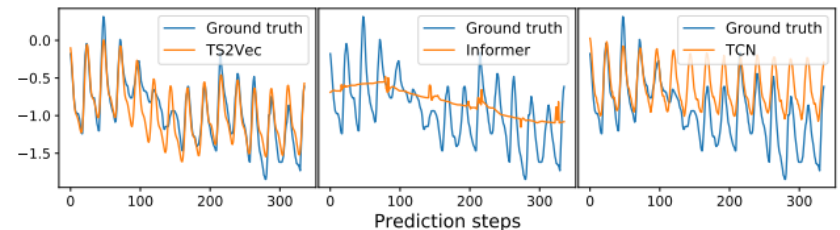


Figure 5: A prediction slice ( $H=336$ ) of TS2Vec, Informer and TCN on the test set of ETTh<sub>2</sub>.

# Instance Contrastive Discrimination Pretext

## TS2Vec : Towards Universal Representation of Time Series

❖ 단변량과 다변량 데이터 모두에서 Informer, N-BEATS 같은 지도학습 모델보다도 좋은 성능을 내고 있음

| Dataset           | H   | TS2Vec       |              | Informer     |              | LogTrans |       | N-BEATS |       | TCN   |       | LSTnet |       |
|-------------------|-----|--------------|--------------|--------------|--------------|----------|-------|---------|-------|-------|-------|--------|-------|
|                   |     | MSE          | MAE          | MSE          | MAE          | MSE      | MAE   | MSE     | MAE   | MSE   | MAE   | MSE    | MAE   |
| ETTh <sub>1</sub> | 24  | <b>0.039</b> | <b>0.152</b> | 0.098        | 0.247        | 0.103    | 0.259 | 0.094   | 0.238 | 0.075 | 0.210 | 0.108  | 0.284 |
|                   | 48  | <b>0.062</b> | <b>0.191</b> | 0.158        | 0.319        | 0.167    | 0.328 | 0.210   | 0.367 | 0.227 | 0.402 | 0.175  | 0.424 |
|                   | 168 | <b>0.134</b> | <b>0.282</b> | 0.183        | 0.346        | 0.207    | 0.375 | 0.232   | 0.391 | 0.316 | 0.493 | 0.396  | 0.504 |
|                   | 336 | <b>0.154</b> | <b>0.310</b> | 0.222        | 0.387        | 0.230    | 0.398 | 0.232   | 0.388 | 0.306 | 0.495 | 0.468  | 0.593 |
|                   | 720 | <b>0.163</b> | <b>0.327</b> | 0.269        | 0.435        | 0.273    | 0.463 | 0.322   | 0.490 | 0.390 | 0.557 | 0.659  | 0.766 |
| ETTh <sub>2</sub> | 24  | <b>0.090</b> | <b>0.229</b> | 0.093        | 0.240        | 0.102    | 0.255 | 0.198   | 0.345 | 0.103 | 0.249 | 3.554  | 0.445 |
|                   | 48  | <b>0.124</b> | <b>0.273</b> | 0.155        | 0.314        | 0.169    | 0.348 | 0.234   | 0.386 | 0.142 | 0.290 | 3.190  | 0.474 |
|                   | 168 | <b>0.208</b> | <b>0.360</b> | 0.232        | 0.389        | 0.246    | 0.422 | 0.331   | 0.453 | 0.227 | 0.376 | 2.800  | 0.595 |
|                   | 336 | <b>0.213</b> | <b>0.369</b> | 0.263        | 0.417        | 0.267    | 0.437 | 0.431   | 0.508 | 0.296 | 0.430 | 2.753  | 0.738 |
|                   | 720 | <b>0.214</b> | <b>0.374</b> | 0.277        | 0.431        | 0.303    | 0.493 | 0.437   | 0.517 | 0.325 | 0.463 | 2.878  | 1.044 |
| ETTm <sub>1</sub> | 24  | <b>0.015</b> | <b>0.092</b> | 0.030        | 0.137        | 0.065    | 0.202 | 0.054   | 0.184 | 0.041 | 0.157 | 0.090  | 0.206 |
|                   | 48  | <b>0.027</b> | <b>0.126</b> | 0.069        | 0.203        | 0.078    | 0.220 | 0.190   | 0.361 | 0.101 | 0.257 | 0.179  | 0.306 |
|                   | 96  | <b>0.044</b> | <b>0.161</b> | 0.194        | 0.372        | 0.199    | 0.386 | 0.183   | 0.353 | 0.142 | 0.311 | 0.272  | 0.399 |
|                   | 288 | <b>0.103</b> | <b>0.246</b> | 0.401        | 0.554        | 0.411    | 0.572 | 0.186   | 0.362 | 0.318 | 0.472 | 0.462  | 0.558 |
|                   | 672 | <b>0.156</b> | <b>0.307</b> | 0.512        | 0.644        | 0.598    | 0.702 | 0.197   | 0.368 | 0.397 | 0.547 | 0.639  | 0.697 |
| Electricity       | 24  | 0.260        | 0.288        | <b>0.251</b> | <b>0.275</b> | 0.528    | 0.447 | 0.427   | 0.330 | 0.263 | 0.279 | 0.281  | 0.287 |
|                   | 48  | <b>0.319</b> | <b>0.324</b> | 0.346        | 0.339        | 0.409    | 0.414 | 0.551   | 0.392 | 0.373 | 0.344 | 0.381  | 0.366 |
|                   | 168 | <b>0.427</b> | <b>0.394</b> | 0.544        | 0.424        | 0.959    | 0.612 | 0.893   | 0.538 | 0.609 | 0.462 | 0.599  | 0.500 |
|                   | 336 | <b>0.565</b> | <b>0.474</b> | 0.713        | 0.512        | 1.079    | 0.639 | 1.035   | 0.669 | 0.855 | 0.606 | 0.823  | 0.624 |
|                   | 720 | <b>0.861</b> | <b>0.643</b> | 1.182        | 0.806        | 1.001    | 0.714 | 1.548   | 0.881 | 1.263 | 0.858 | 1.278  | 0.906 |
| Avg.              |     | <b>0.209</b> | <b>0.296</b> | 0.310        | 0.390        | 0.370    | 0.434 | 0.399   | 0.426 | 0.338 | 0.413 | 1.099  | 0.536 |

Table 7: Univariate time series forecasting results.

| Dataset           | H   | TS2Vec       |              | Informer     |              | StemGNN      |              | TCN          |              | LogTrans     |              | LSTnet |              |
|-------------------|-----|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------|--------------|
|                   |     | MSE          | MAE          | MSE          | MAE          | MSE          | MAE          | MSE          | MAE          | MSE          | MAE          | MSE    | MAE          |
| ETTh <sub>1</sub> | 24  | 0.599        | <b>0.534</b> | <b>0.577</b> | 0.549        | 0.614        | 0.571        | 0.767        | 0.612        | 0.686        | 0.604        | 1.293  | 0.901        |
|                   | 48  | <b>0.629</b> | <b>0.555</b> | 0.685        | 0.625        | 0.748        | 0.618        | 0.713        | 0.617        | 0.766        | 0.757        | 1.456  | 0.960        |
|                   | 168 | 0.755        | 0.636        | 0.931        | 0.752        | <b>0.663</b> | <b>0.608</b> | 0.995        | 0.738        | 1.002        | 0.846        | 1.997  | 1.214        |
|                   | 336 | <b>0.907</b> | <b>0.717</b> | 1.128        | 0.873        | 0.927        | 0.730        | 1.175        | 0.800        | 1.362        | 0.952        | 2.655  | 1.369        |
|                   | 720 | <b>1.048</b> | <b>0.790</b> | 1.215        | 0.896        | —*           | —            | 1.453        | 1.311        | 1.397        | 1.291        | 2.143  | 1.380        |
| ETTh <sub>2</sub> | 24  | <b>0.398</b> | <b>0.461</b> | 0.720        | 0.665        | 1.292        | 0.883        | 1.365        | 0.888        | 0.828        | 0.750        | 2.742  | 1.457        |
|                   | 48  | <b>0.580</b> | <b>0.573</b> | 1.457        | 1.001        | 1.099        | 0.847        | 1.395        | 0.960        | 1.806        | 1.034        | 3.567  | 1.687        |
|                   | 168 | <b>1.901</b> | <b>1.065</b> | 3.489        | 1.515        | 2.282        | 1.228        | 3.166        | 1.407        | 4.070        | 1.681        | 3.242  | 2.513        |
|                   | 336 | <b>2.304</b> | <b>1.215</b> | 2.723        | 1.340        | 3.086        | 1.351        | 3.256        | 1.481        | 3.875        | 1.763        | 2.544  | 2.591        |
|                   | 720 | <b>2.650</b> | <b>1.373</b> | 3.467        | 1.473        | —            | —            | 3.690        | 1.588        | 3.913        | 1.552        | 4.625  | 3.709        |
| ETTm <sub>1</sub> | 24  | 0.443        | 0.436        | <b>0.323</b> | <b>0.369</b> | 0.620        | 0.570        | 0.324        | 0.374        | 0.419        | 0.412        | 1.968  | 1.170        |
|                   | 48  | 0.582        | 0.515        | 0.494        | 0.503        | 0.744        | 0.628        | <b>0.477</b> | <b>0.450</b> | 0.507        | 0.583        | 1.999  | 1.215        |
|                   | 96  | <b>0.622</b> | <b>0.549</b> | 0.678        | 0.614        | 0.709        | 0.624        | 0.636        | 0.602        | 0.768        | 0.792        | 2.762  | 1.542        |
|                   | 288 | <b>0.709</b> | <b>0.609</b> | 1.056        | 0.786        | 0.843        | 0.683        | 1.270        | 1.351        | 1.462        | 1.320        | 1.257  | 2.076        |
|                   | 672 | <b>0.786</b> | <b>0.655</b> | 1.192        | 0.926        | —            | —            | 1.381        | 1.467        | 1.669        | 1.461        | 1.917  | 2.941        |
| Electricity       | 24  | <b>0.287</b> | <b>0.374</b> | 0.312        | 0.387        | 0.439        | 0.388        | 0.305        | 0.384        | 0.297        | <b>0.374</b> | 0.356  | 0.419        |
|                   | 48  | <b>0.307</b> | <b>0.388</b> | 0.392        | 0.431        | 0.413        | 0.455        | 0.317        | 0.392        | 0.316        | 0.389        | 0.429  | 0.456        |
|                   | 168 | <b>0.332</b> | <b>0.407</b> | 0.515        | 0.509        | 0.506        | 0.518        | 0.358        | 0.423        | 0.426        | 0.466        | 0.372  | 0.425        |
|                   | 336 | <b>0.349</b> | 0.420        | 0.759        | 0.625        | 0.647        | 0.596        | <b>0.349</b> | 0.416        | 0.365        | 0.417        | 0.352  | <b>0.409</b> |
|                   | 720 | 0.375        | 0.438        | 0.969        | 0.788        | —            | —            | 0.447        | 0.486        | <b>0.344</b> | <b>0.403</b> | 0.380  | 0.443        |
| Avg.              |     | <b>0.828</b> | <b>0.636</b> | 1.154        | 0.781        | —            | —            | 1.192        | 0.837        | 1.314        | 0.892        | 1.903  | 1.444        |

\* All  $H \geq 672$  cases of StemGNN fail for the out-of-memory (24GB) even when  $batch\_size = 1$ .

Table 8: Multivariate time series forecasting results.

# Instance Contrastive Discrimination Pretext

## TS2Vec : Towards Universal Representation of Time Series

- ❖ 어떤 도메인이나 특성에 상관없고 가변 길이의 시계열 데이터의 특징 추출하는 보편적인 프레임워크 제안
- ❖ 타임스탬프 단위의 인스턴스로 샘플과 시간 단위 대조 학습하여 특정 타임스탬프 레벨의 특징까지 잘 추출함
  - ❖ 국소적인 부분부터 전역적인 특징을 타임스탬프 단위의 특징 벡터를 통해 잘 파악
  - ❖ 이상 탐지에도 사용 가능하며, 분류, 예측 등 다양한 사후학습에서 좋은 성능을 보임
  - ❖ 샘플 간 대조학습의 경우는 분류 외에 예측이나 이상탐지에는 사용 불가
- ❖ 시계열 데이터의 특성을 위반할 수 있는 방법들을 사용하지 않음으로써 강건한 컨텍스트 정보를 얻을 수 있으며, 미싱 데이터가 있음에도 강건한 성능을 보임

---

# Conclusion

---

# Conclusion

---

- ❖ CV, NLP에서만 국한되어오던 표현 학습들이 시계열 데이터에 적용도 많이 이뤄지고 있음.
- ❖ 시계열 데이터 데이터 특성이 까다롭고 복잡하여 쉬운 적용이나 범용적인 프레임워크는 아쉬움이 있으나 앞으로 많은 연구들이 나올 것으로 기대함
- ❖ 최근에는 타임스탬프 시점까지의 특징을 찾아내는 표현학습 모델이 등장하고 좋은 성능을 내고 있고 성능 또한 좋은 점이 신선

# Reference

---

- [1] Yue, Z., Wang, Y., Duan, J., Yang, T., Huang, C., Tong, Y., & Xu, B. (2021). TS2Vec: Towards Universal Representation of Time Series. arXiv preprint arXiv:2106.10466.
- [2] Wen, Q., Sun, L., Yang, F., Song, X., Gao, J., Wang, X., & Xu, H. (2020). Time series data augmentation for deep learning: A survey. arXiv preprint arXiv:2002.12478.
- [3] Iwana, B. K., & Uchida, S. (2021). An empirical survey of data augmentation for time series classification with neural networks. Plos one, 16(7), e0254841.
- [4] Ericsson, L., Gouk, H., Loy, C. C., & Hospedales, T. M. (2021). Self-Supervised Representation Learning: Introduction, Advances and Challenges. arXiv preprint arXiv:2110.09327.
- [5] Zerveas, G., Jayaraman, S., Patel, D., Bhamidipaty, A., & Eickhoff, C. (2021, August). A transformer-based framework for multivariate time series representation learning. In Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining (pp. 2114-2124).
- [6] Eldele, E., Ragab, M., Chen, Z., Wu, M., Kwok, C. K., Li, X., & Guan, C. (2021). Time-series representation learning via temporal and contextual contrasting. arXiv preprint arXiv:2106.14112.
- [7] Banville, H., Albuquerque, I., Hyvärinen, A., Moffat, G., Engemann, D. A., & Gramfort, A. (2019, October). Self-supervised representation learning from electroencephalography signals. In 2019 IEEE 29th International Workshop on Machine Learning for Signal Processing (MLSP) (pp. 1-6). IEEE.
- [8] Mohsenvand, M. N., Izadi, M. R., & Maes, P. (2020, November). Contrastive representation learning for electroencephalogram classification. In Machine Learning for Health (pp. 238-253). PMLR.