Two Paradigms for Irregularly Sampled Time Series

2026.01.02.

Data Mining & Quality Analytics Lab.

조광은

발표자 소개

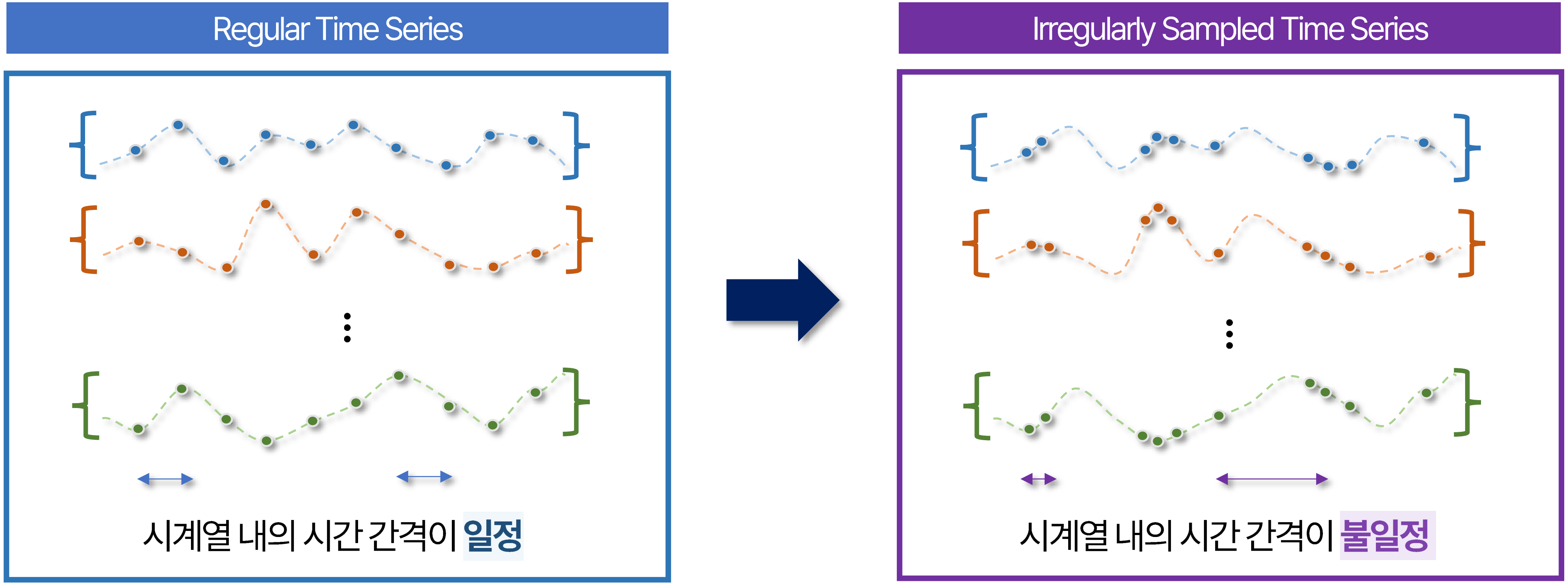
- **조광은 (Kwangeun Cho)**
  - 고려대학교 산업경영공학과 Data Mining & Quality Analytics Lab.
  - M.S. Student (2024.03 ~ Present)
  - 지도교수: 김성범 교수님
- **Research Interest**
  - Multivariate Time Series Forecasting
  - Multivariate Time Series Anomaly Detection
  - Irregular Time Series Modeling
- **Contact**
  - E-mail: chopol98@korea.ac.kr

Contents

Background
Irregularly Sampled Time Series

❖ **Irregularly sampled time series(ISTS)**

- **Irregularly sampled time series(ISTS)**는 **수집된 시간 간격이 불일정한 시계열**
- 따라서 ISTS의 **인접한 관측치의 시간 간격**은 **일정하지 않음**

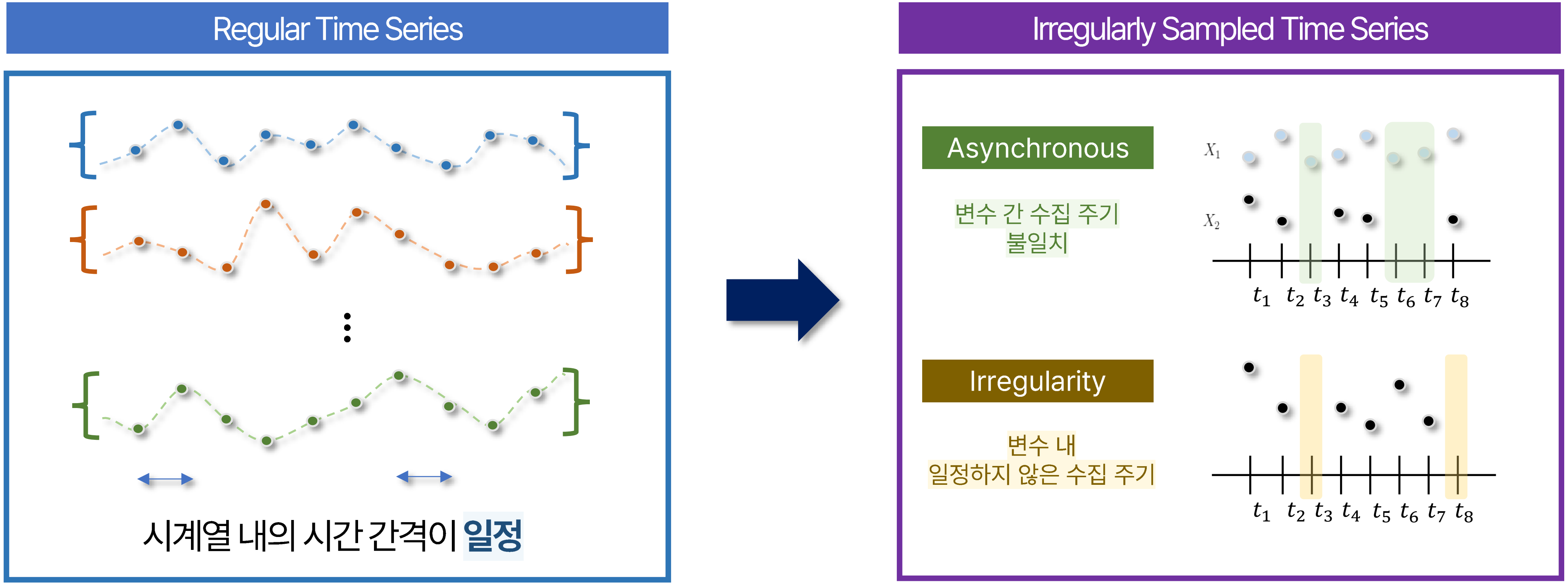


Background

Irregularly Sampled Time Series

❖ **Irregularly sampled time series(ISTS)**

- **Irregularity:** 변수 내 timestep들이 일정하지 않게 수집 / **Asynchronous:** 변수 간 timestep들이 불일치
- Irregularity나 asynchronous 상황이 발생하는 시계열 데이터를 통틀어 **Irregularly sampled time series (ISTS)** 라고 함



Background
Irregularly Sampled Time Series

❖ **왜 ISTS를 고려해야 하는가?**

- ISTS를 regular한 시계열로 가정하고 처리할 경우 수집되지 않은 구간은 결측으로 간주
- 수집되지 않은 값들을 탈락시킬 경우 관측되지 않은 구간의 정보는 무시됨

Why we should consider ISTS?

Irregular TS

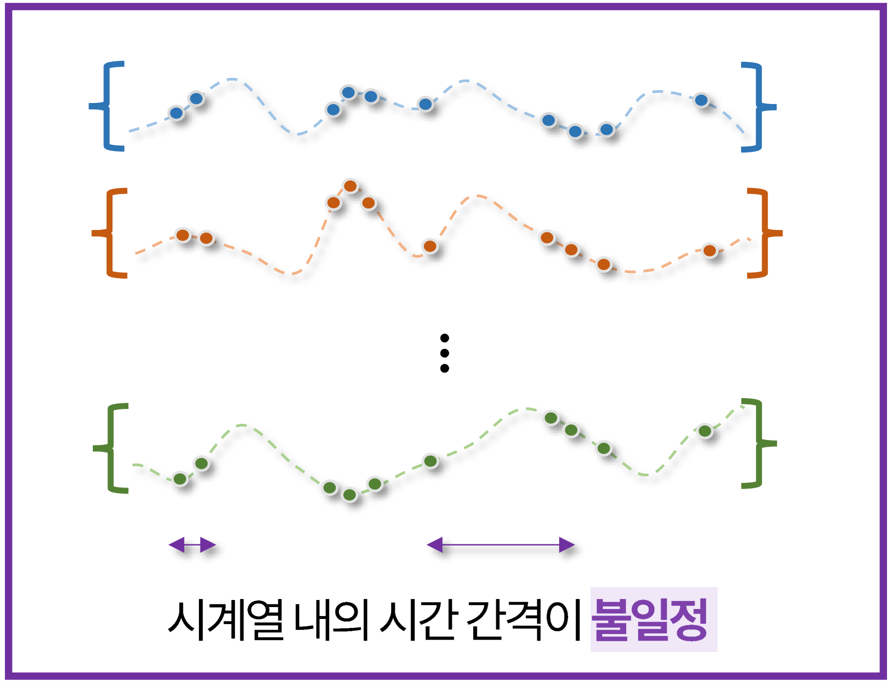
T	Value
0	22
1	23
2	26
6	40
7	40

Treat as Regular TS

T	Value
0	22
1	23
2	26
3	40
4	40

수집되지 않은 구간을 **결측**으로 간주
관측되지 못한 구간의 정보를 무시

Irregularly Sampled Time Series



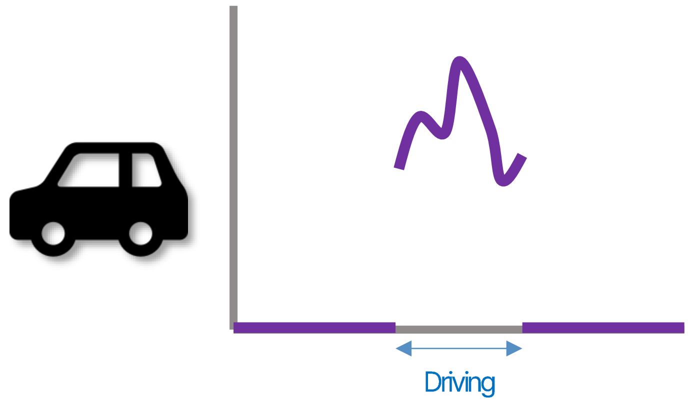
Background

Irregularly Sampled Time Series

❖ **왜 ISTS를 고려해야 하는가?**

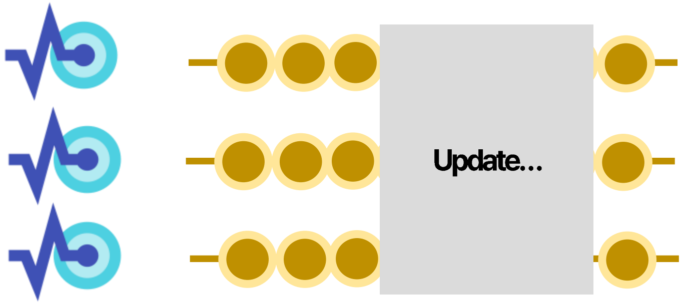
- 관측치가 **불규칙하게 수집된 상황**은 제조 현장 및 산업 데이터에서 자주 나타나는 현상
- 이러한 데이터를 regular time series를 가정하고 있는 시계열 모델에 투입할 경우 **optimal하지 않은 성능**

Case 1. 자동차 주행 데이터



자동차 주행의 경우 **휴지 구간** 존재
수집 주기가 불규칙할 수 있음

Case 2. 설비 데이터



센서 업데이트로 인한 휴지 구간 발생

Background

Irregularly Sampled Time Series

❖ **어떻게 ISTS를 고려하는가?**

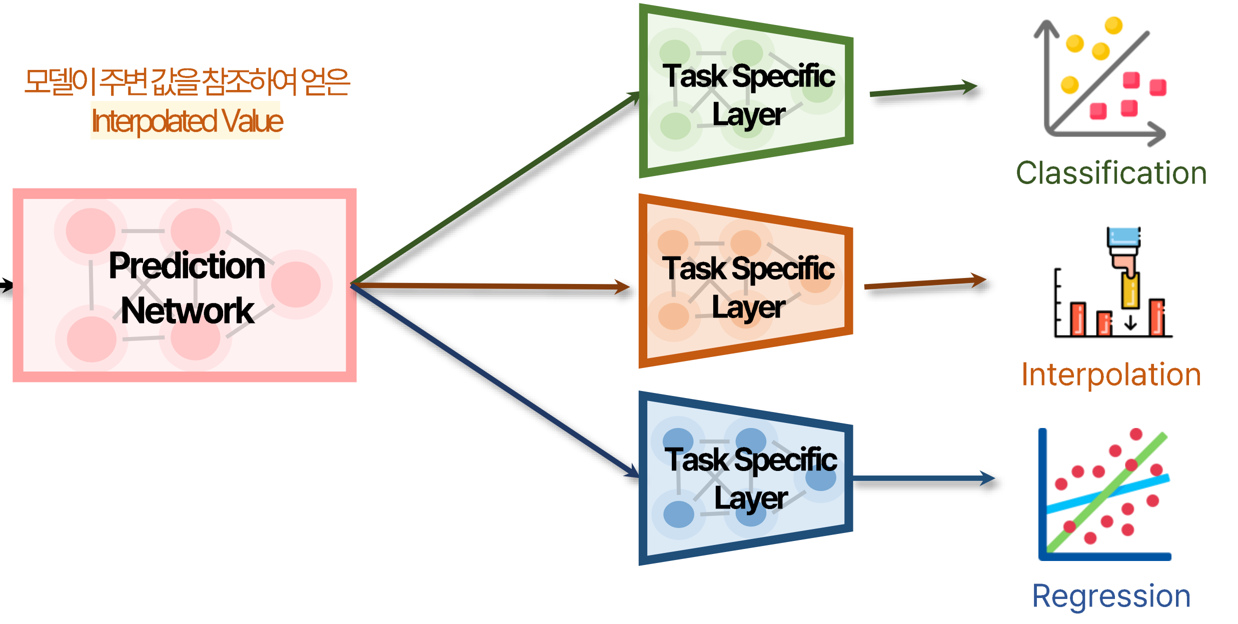
- Main Consideration: 어떻게 ISTS를 regular한 시계열로 다룰 것인가?
- 주변 값들을 이용하여 ISTS를 regular time series로 만들고, 이를 task에 적용하는 방법이 있음

Irregular TS

T	Value
0	22
1	23
2	26
6	40
7	40

Interpolated ISTS

T	Value
0	22
1	23
2	26
3	28
4	32
5	37
6	40
7	40



Background

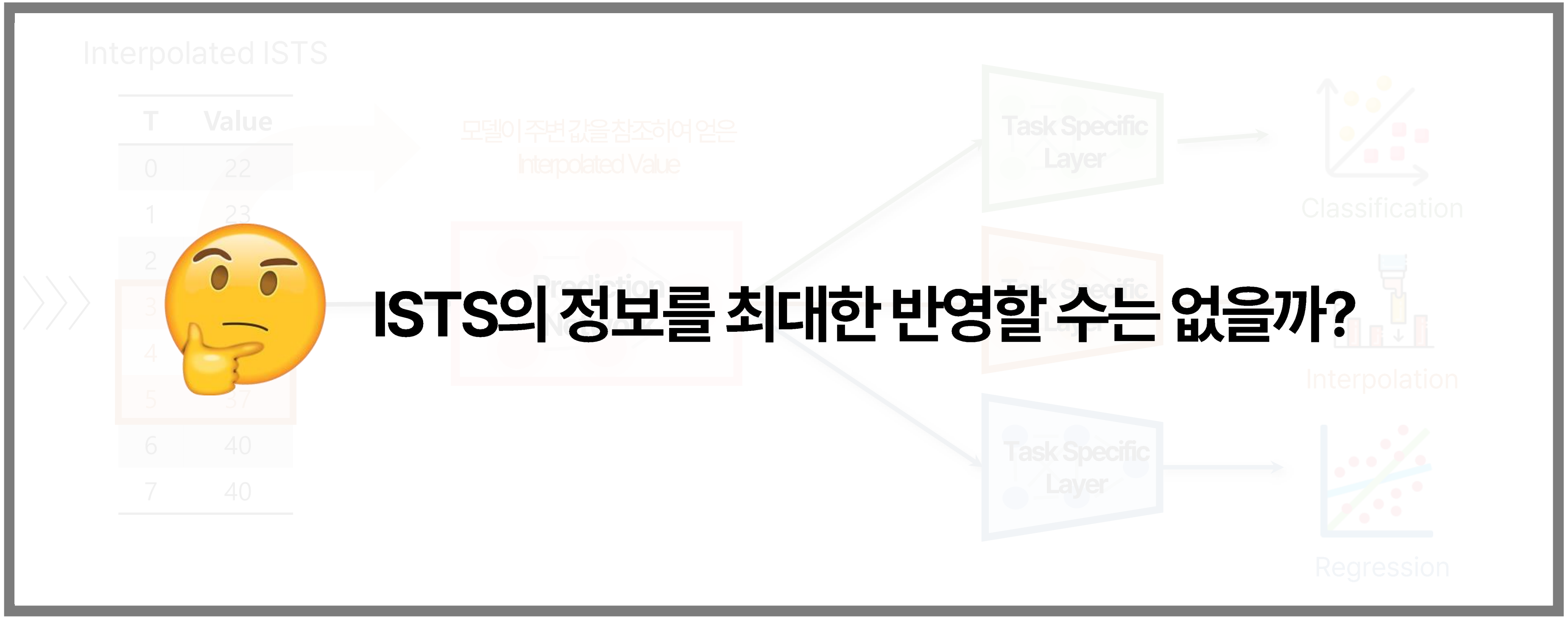
Irregularly Sampled Time Series

❖ **어떻게 ISTS를 고려하는가?**

- Main Consideration: 어떻게 ISTS를 regular한 시계열로 다룰 것인가?
- 주변 값들을 이용하여 ISTS를 regular time series로 만들고, 이를 task에 적용하는 방법이 있음

Irregular TS

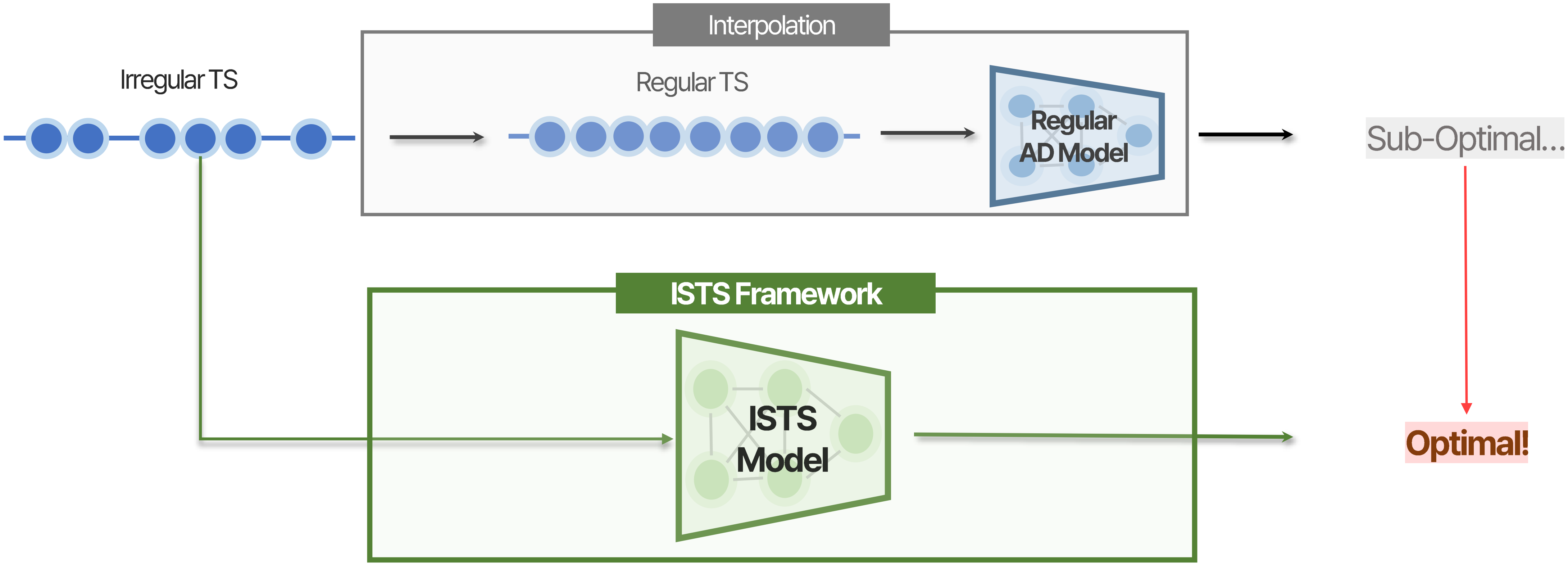
T	Value
0	22
1	23
2	26
6	40
7	40



Background
Research trend

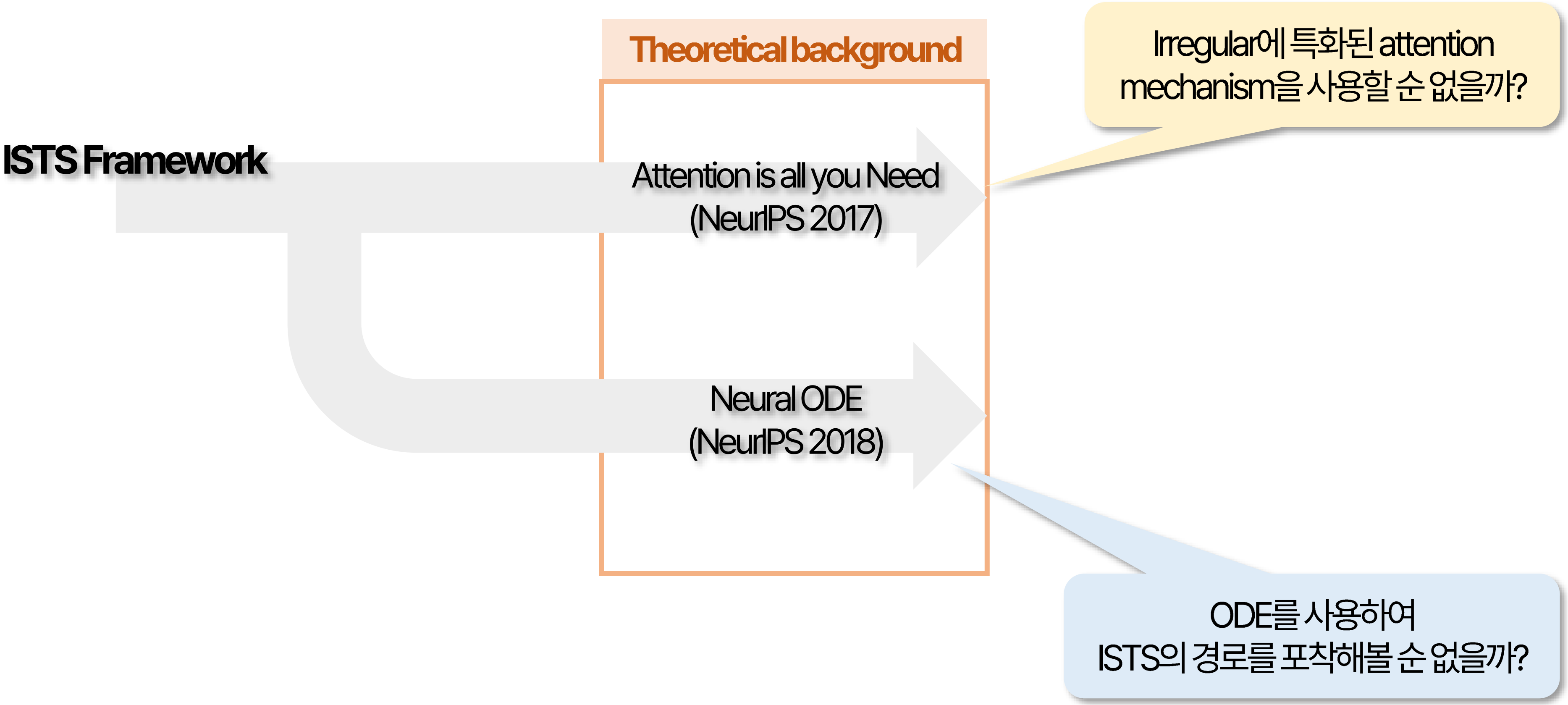
❖ **ISTS framework**

- **관측하지 못한 데이터가 가진 rich information을 활용하여 다양한 task에 적용해보자!**
- Forecasting, interpolation, classification 등을 ISTS에서 풀기 위한 모델 등장



Background
Research trend

❖ **Research trend**



Background

Research trend

❖ **Research trend**

ISTS Framework

Theoretical background

Attention is all you Need
(NeurIPS 2017)

Neural ODE
(NeurIPS 2018)

mTANs
(ICLR 2021)

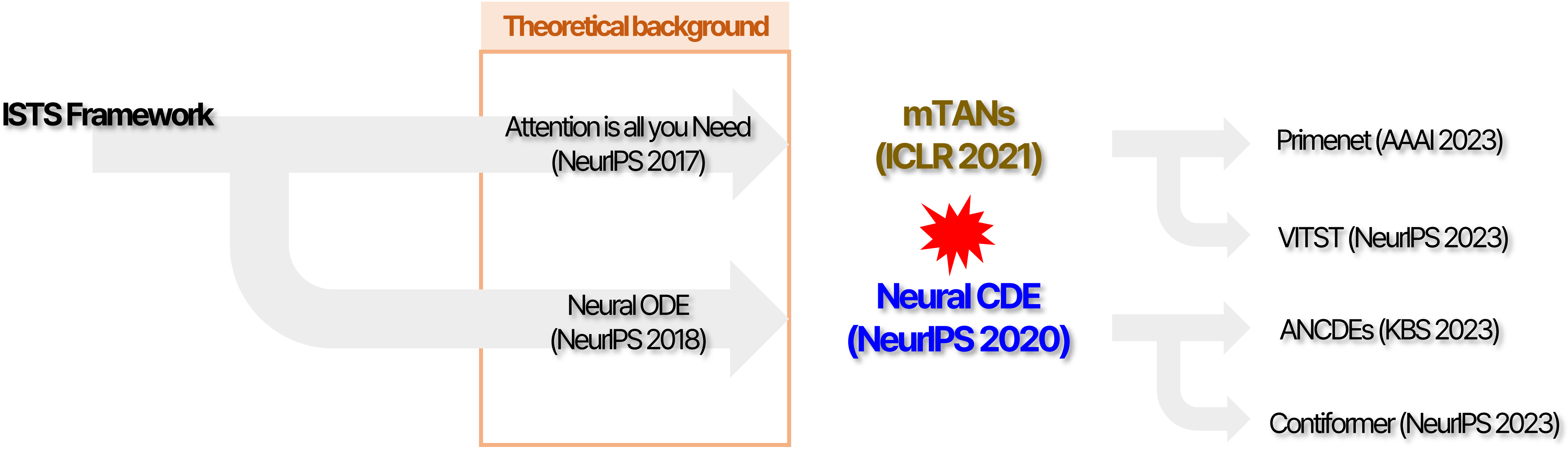
ISTS에 특화된 어텐션 메커니즘
Multi-time attention 사용

Neural CDE
(NeurIPS 2020)

ODE를 발전시킨 CDE를
Irregular time series에 적용

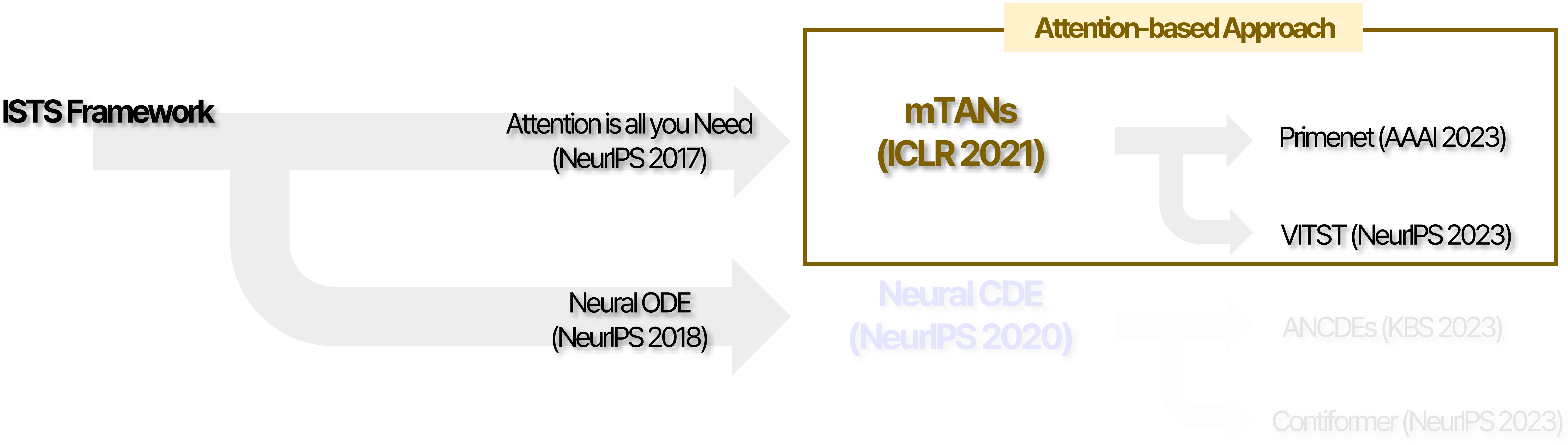
Background
Research trend

❖ **Research trend**



Background
Research trend

❖ **Research trend**



Methods

Attention-based Approach: Multi-Time Attention Networks for Irregularly Sampled Time Series

❖ **Multi-Time Attention Networks for Irregularly Sampled Time Series (ICLR 2021)**

- 2026년 1월 기준 338회 인용
- **Attention mechanism**을 ISTS modeling에 사용한 첫 번째 모델

MULTI-TIME ATTENTION NETWORKS FOR IRREGULARLY SAMPLED TIME SERIES

Satya Narayan Shukla & Benjamin M. Marlin
College of Information and Computer Sciences
University of Massachusetts Amherst
Amherst, MA 01003, USA
{snshukla,marlin}@cs.umass.edu

ABSTRACT

Irregular sampling occurs in many time series modeling applications where it presents a significant challenge to standard deep learning models. This work is motivated by the analysis of physiological time series data in electronic health records, which are sparse, irregularly sampled, and multivariate. In this paper, we propose a new deep learning framework for this setting that we call *Multi-Time Attention Networks*. Multi-Time Attention Networks learn an embedding of continuous time values and use an attention mechanism to produce a fixed-length representation of a time series containing a variable number of observations. We investigate the performance of this framework on interpolation and classification tasks using multiple datasets. Our results show that the proposed approach performs as well or better than a range of baseline and recently proposed models while offering significantly faster training times than current state-of-the-art methods.[1]

Background

Similarity 기반 self-attention이 여러 task를 잘하네?

mTAND

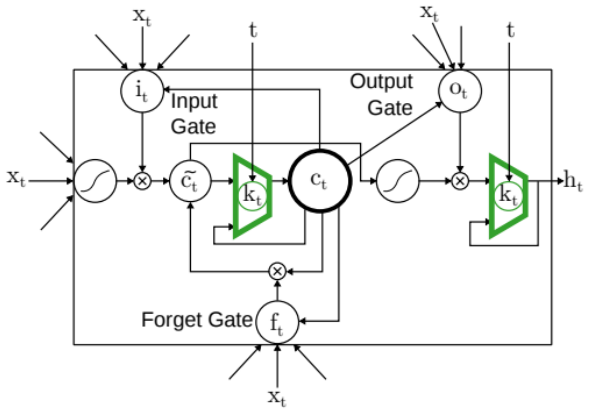
그렇다면 **ISTS의 수집되지 않은 구간**도 attention으로 모델링 가능하지 않을까?

Methods

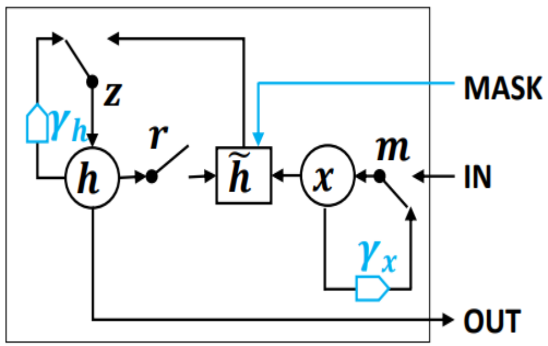
Attention-based Approach: Multi-Time Attention Networks for Irregularly Sampled Time Series

❖ **Previous work: model-based approach**

- Phased-LSTM, GRU-D 등 **모델 구조를 ISTS가 사용할 수 있도록** 변경하는 구조들이 연구
- 이러한 방법은 결측된 값을 missing value로 간주하고 **결측이 발생했을 때 모델의 업데이트**를 주로 다룸



Phased LSTM (2016)
게이트를 추가하여 결측과 상관없는 주기로
가중치를 업데이트



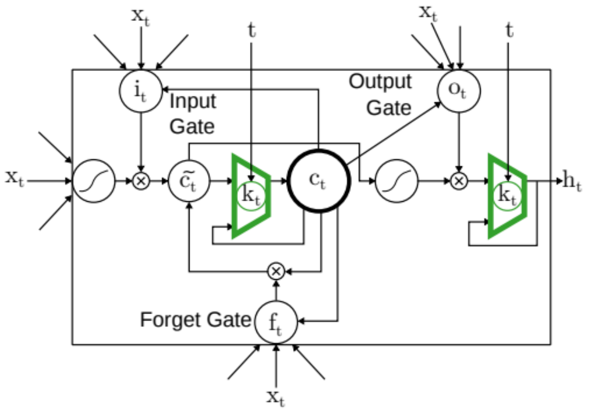
GRU-D (2018)
결측에 decay rate 부여해
이전 관측치의 값을 반영

Methods

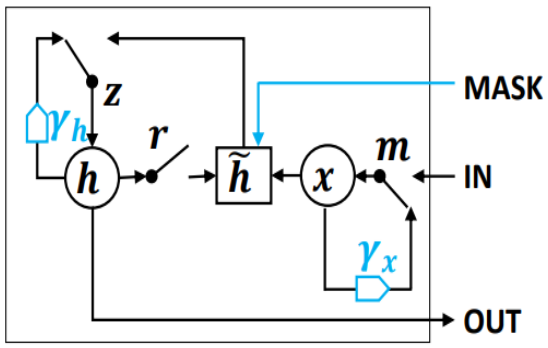
Attention-based Approach: Multi-Time Attention Networks for Irregularly Sampled Time Series

❖ **Previous work: model-based approach**

- **Irregularity 측면:** 결국 이들 모델은 **결측에 보정을 하는 구조,** ISTS의 표현을 효과적으로 학습하지 못한다
- **Asynchronous 측면:** 다변량 시계열이 정렬이 되지 않는 상황인 asynchronous 상황에서 효과적으로 작동하지 못한다



Phased LSTM (2016)
게이트를 추가하여 결측과 상관없는 주기로
가중치를 업데이트



GRU-D (2018)
결측에 decay rate 부여해
이전 관측치의 값을 반영

부분적으로 관찰된 Irreguarly sampled time series에서

1) 직접 표현을 학습하고
2) 샘플링 빈도까지 반영하는

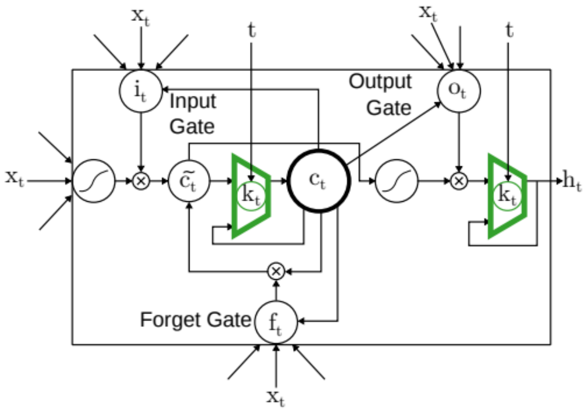
방법이 없을까?

Methods

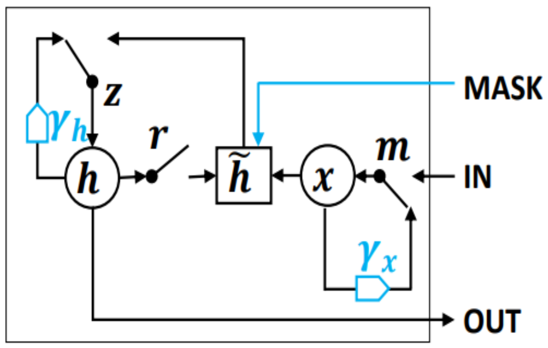
Attention-based Approach: Multi-Time Attention Networks for Irregularly Sampled Time Series

❖ **Previous work: model-based approach**

- **Irregularity 측면:** 결국 이들 모델은 **결측에 보정을 하는 구조,** ISTS의 표현을 효과적으로 학습하지 못한다
- **Asynchronous 측면:** 다변량 시계열이 정렬이 되지 않는 상황인 asynchronous 상황에서 효과적으로 작동하지 못한다



Phased LSTM (2016)
게이트를 추가하여 결측과 상관없는 주기로 가중치를 업데이트



GRU-D (2018)
결측에 decay rate 부여해 이전 관측치의 값을 반영

수집된 시간 간 유사도를 attention으로 구하고,
이를 기반으로 representation을 만들면 되겠구나!

부분적으로 관찰된 Ireguarly sampled time series에서

1) 직접 표현을 학습하고
2) 샘플링 빈도까지 반영하는

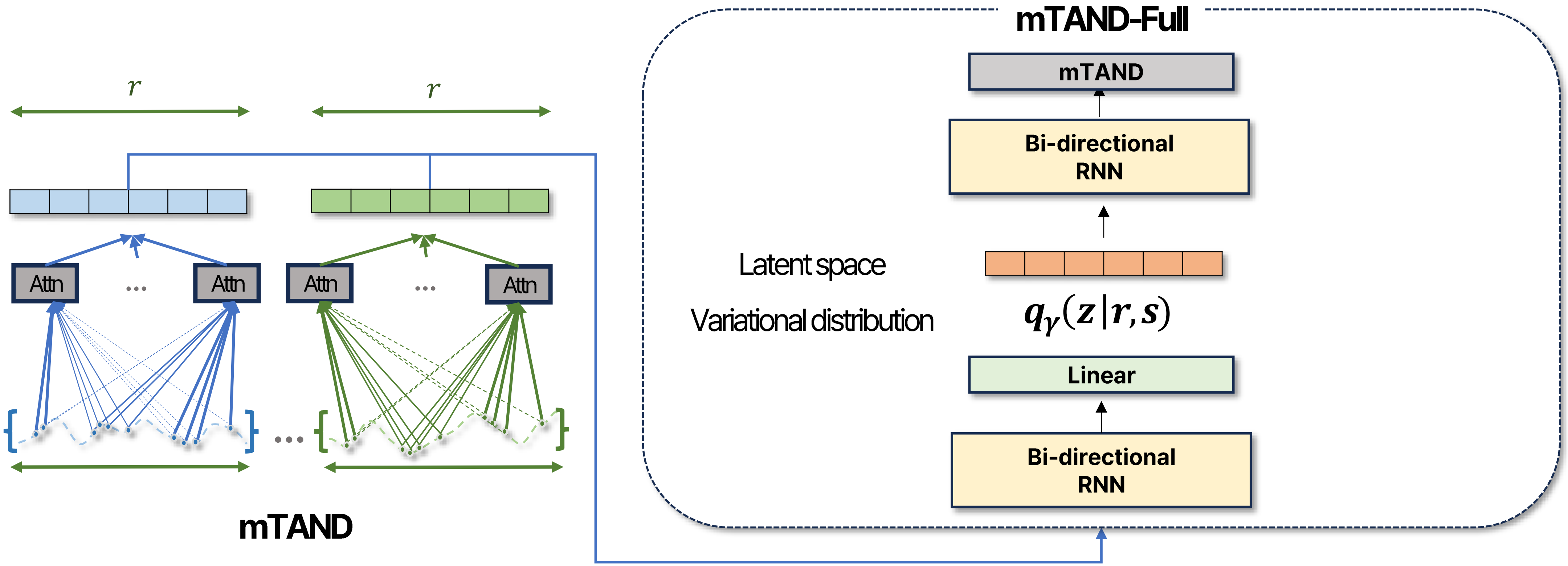
방법이 없을까?

Methods

Attention-based Approach: Multi-Time Attention Networks for Irregularly Sampled Time Series

❖ **Overall structure**

- mTAND: **Time embedding**과 **multi-time attention**을 이용하여 ISTS를 fixed representation으로 변환
- mTAND-Full: **VAE** 구조를 활용하여 encoder-decoder 구조로 irregular time series 처리

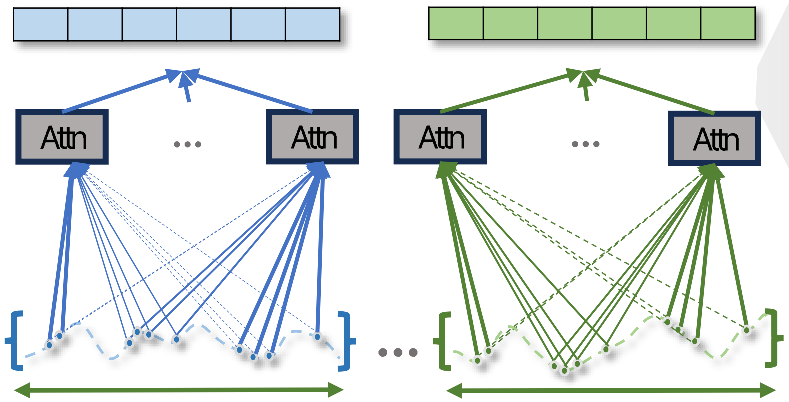


Methods

Attention-based Approach: Multi-Time Attention Networks for Irregularly Sampled Time Series

❖ **Overall structure**

- mTAND: **Time embedding**과 **multi-time attention**을 이용하여 ISTS를 fixed representation으로 변환
- mTAND-Full: **VAE** 구조를 활용하여 encoder-d



mTAND

Continuous time embedding

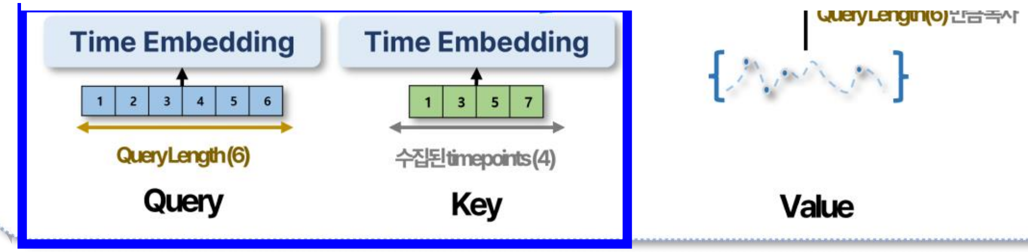
Linear term: linear pattern을 포착

$$\phi_h(t)[i] = \begin{cases} \omega_{0h} \cdot t + \alpha_{0h} & if\ i = 0 \\ \sin(\omega_{ih} \cdot t + \alpha_{ih}) & if\ 0 < i < d_r \end{cases}$$

Time embedding의 차원 수

Periodic terms: periodic pattern을 포착

Continuous한 time points를 vector space로 embedding



Methods

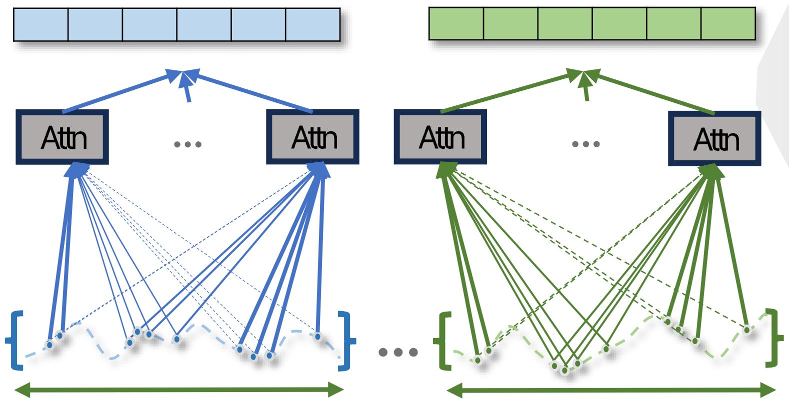
Attention-based Approach: Multi-Time Attention Networks for Irregularly Sampled Time Series

❖ **Overall structure**

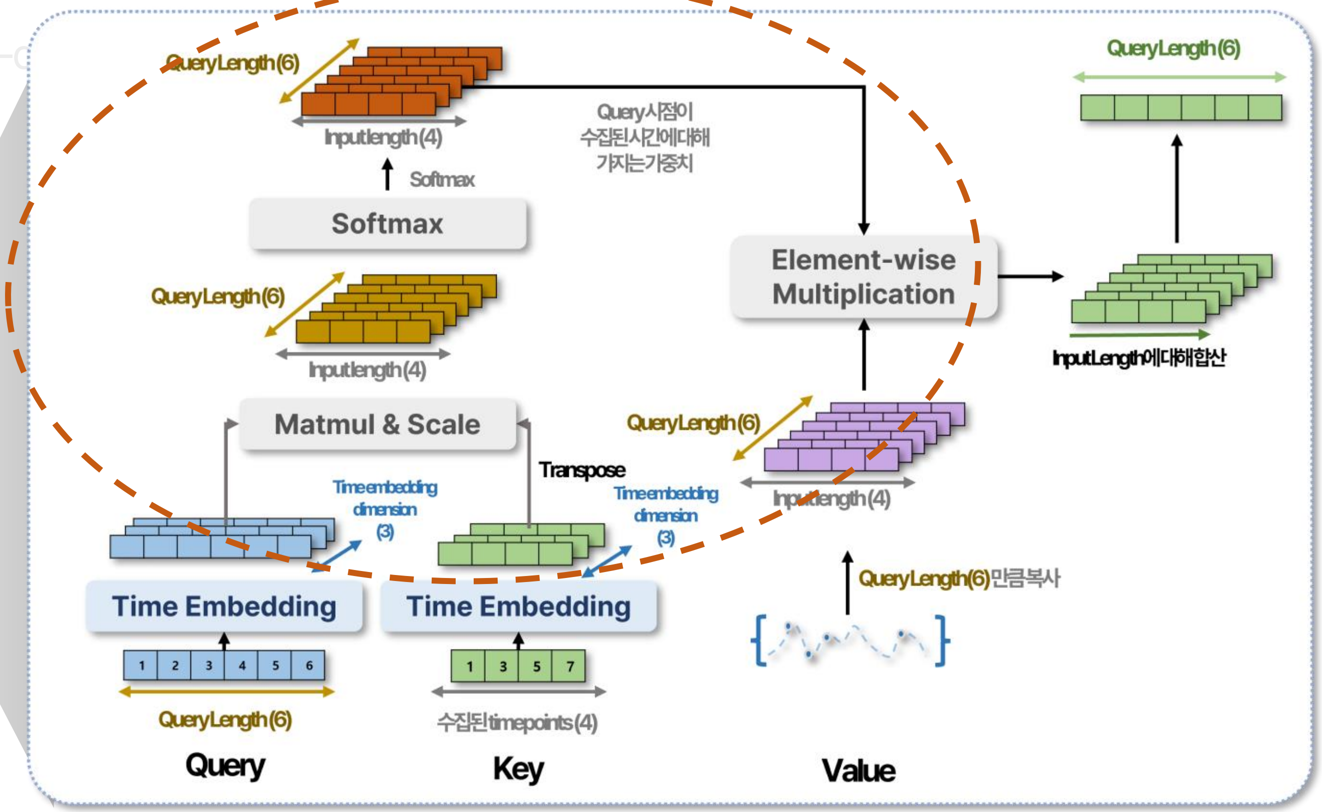
- mTAND: **Time embedding**과 **multi-time attention**을 이용하여 ISTS를 fixed representation으로 변환
- mTAND-Full: **VAE** 구조를 활용하여 encoder-c

수집된 시간 간 유사도를 attention으로 구하고,

이를 기반으로 representation을 만들면 되겠구나!



mTAND



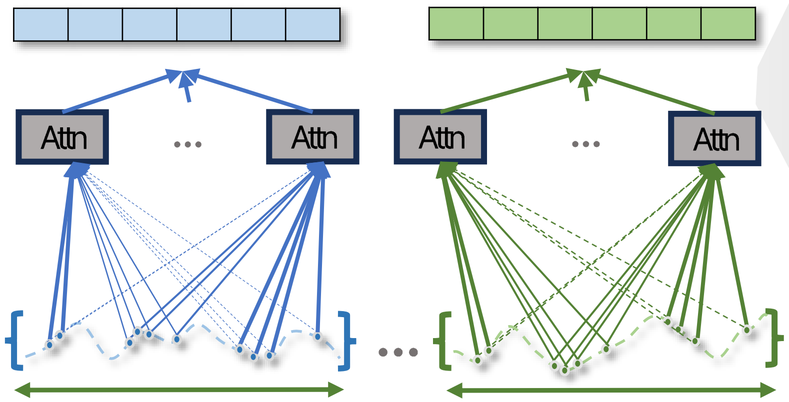
Methods

Attention-based Approach: Multi-Time Attention Networks for Irregularly Sampled Time Series

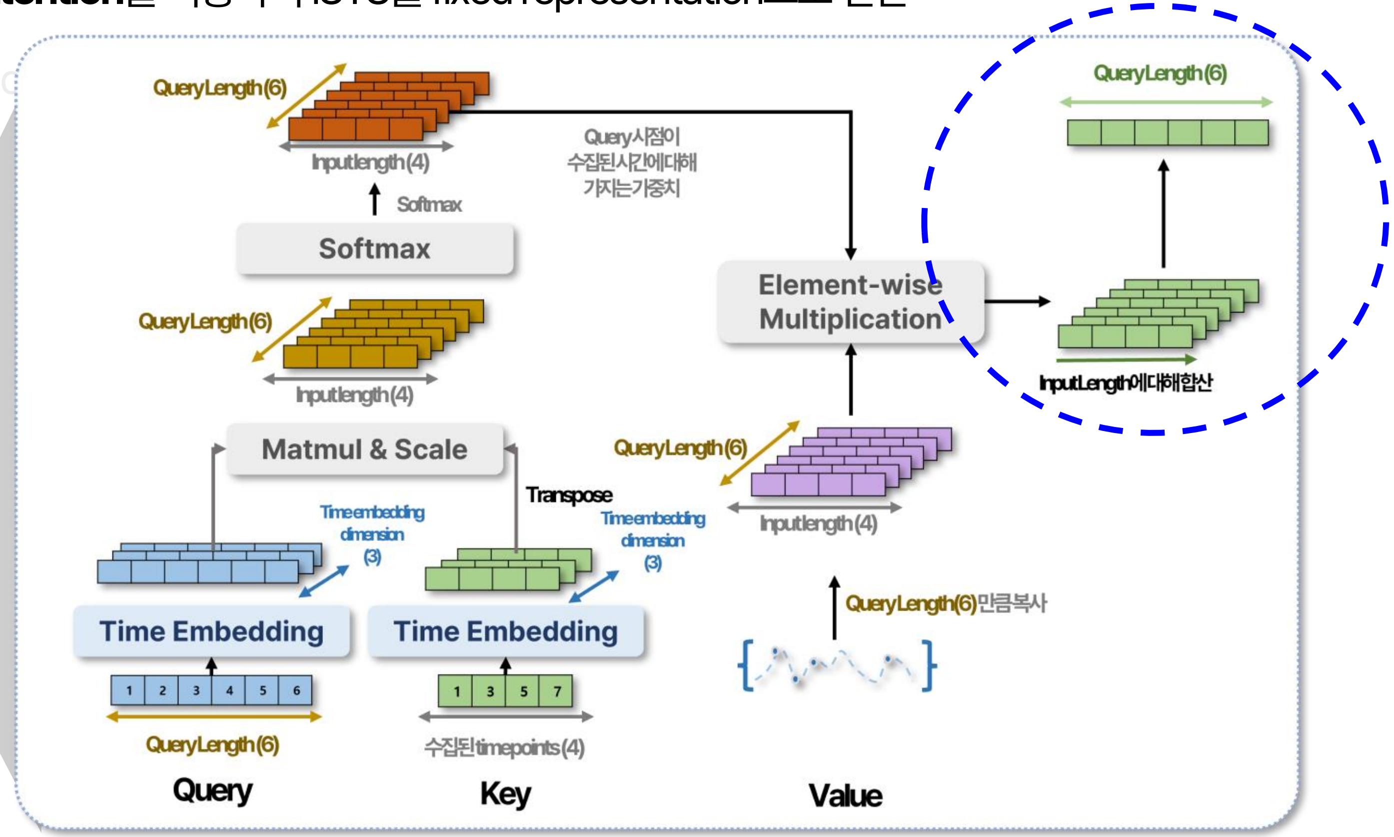
❖ **Overall structure**

- mTAND: **Time embedding**과 **multi-time attention**을 이용하여 ISTS를 fixed representation으로 변환
- mTAND-Full: **VAE** 구조를 활용하여 encoder-

수집된 시간 간 유사도를 attention으로 구하고,
이를 기반으로 **representation**을 만들면 되겠구나!



mTAND



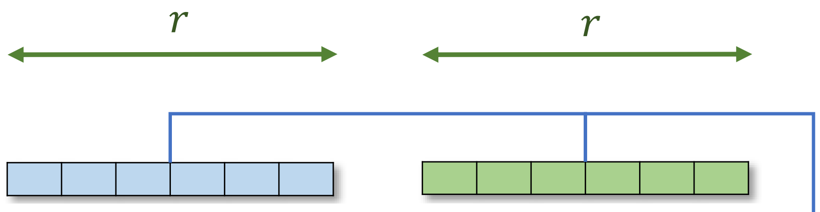
Methods

Attention-based Approach: Multi-Time Attention Networks for Irregularly Sampled Time Series

❖ **Overall structure**

- mTAND: **Time embedding**과 **multi-time attention**을 이용하여 ISTS를 fixed representation으로 변환
- mTAND-Full: **VAE** 구조를 활용하여 encoder-decoder 구조로 irregular time series 처리

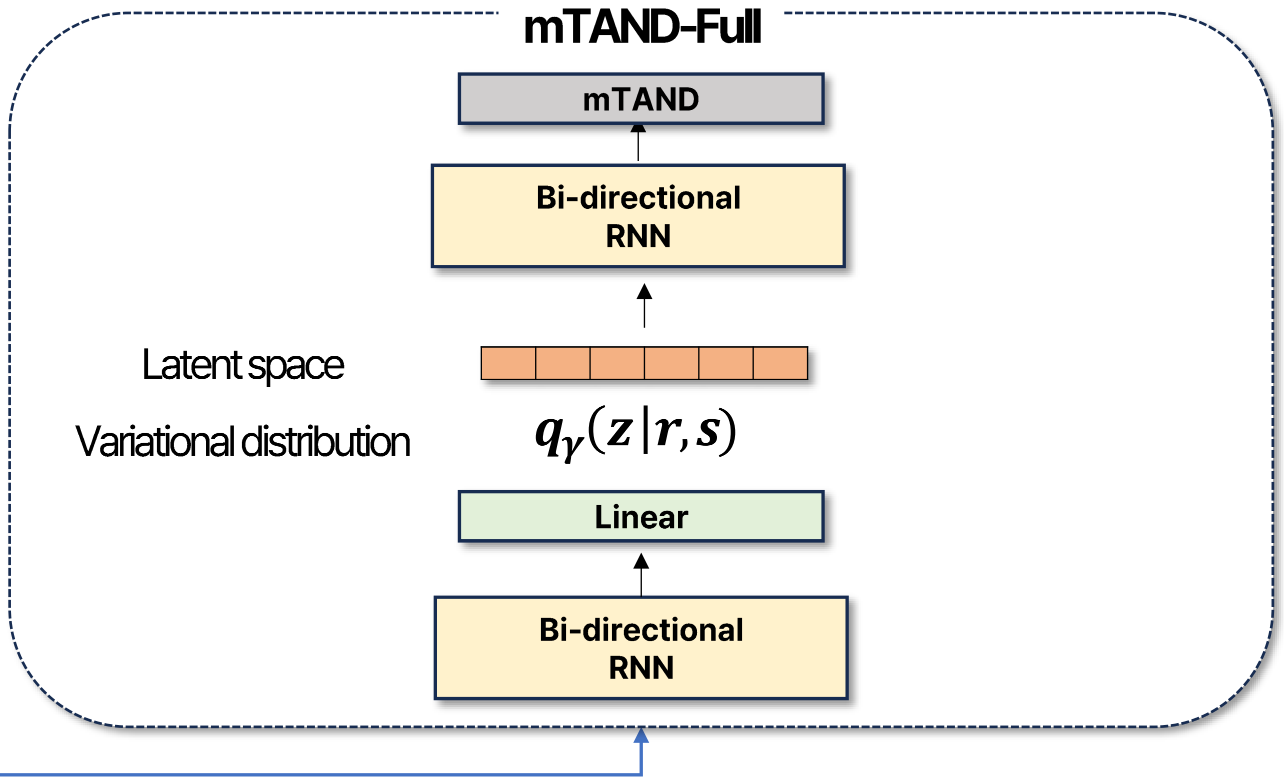
Fixed-length vector: 일반 모델에 투입 가능!



$$\mathcal{L}_{\text{NVAE}}(\theta,\gamma)=\sum_{n=1}^{N}\frac{1}{\sum_d L_{dn}}\Big(\mathbb{E}_{q_\gamma(\mathbf{z}|\mathbf{r},\mathbf{s}_n)}[\log p_\theta(\mathbf{x}_n|\mathbf{z},\mathbf{t}_n)]-D_{\text{KL}}(q_\gamma(\mathbf{z}|\mathbf{r},\mathbf{s}_n)||p(\mathbf{z}))\Big)$$

$$\mathcal{L}_{\text{supervised}}(\theta,\gamma,\delta)=\mathcal{L}_{\text{NVAE}}(\theta,\gamma)+\lambda\mathbb{E}_{q_\gamma(\mathbf{z}|\mathbf{r},\mathbf{s}_n)}\log p_\delta(y_n|\mathbf{z})$$

Unsupervised: VAE loss로만 학습
Supervised: task specific loss를 추가적으로 학습



Methods

Attention-based Approach: Multi-Time Attention Networks for Irregularly Sampled Time Series

❖ **Experimental results**

- **Interpolation:** ODE 계열 모델들에 비해 전체적으로 높은 성능
- **Classification:** 전체적으로 높은 성능, **타 irregular time series model에 비해 빠른 학습 속도**

Model	**Mean Squared Error** $(\times 10^{-3})$				
RNN-VAE	13.418 ± 0.008	12.594 ± 0.004	11.887 ± 0.005	11.133 ± 0.007	11.470 ± 0.006
L-ODE-RNN	8.132 ± 0.020	8.140 ± 0.018	8.171 ± 0.030	8.143 ± 0.025	8.402 ± 0.022
L-ODE-ODE	6.721 ± 0.109	6.816 ± 0.045	6.798 ± 0.143	6.850 ± 0.066	7.142 ± 0.066
mTAND-Full	$\mathbf{4.139 \pm 0.029}$	$\mathbf{4.018 \pm 0.048}$	$\mathbf{4.157 \pm 0.053}$	$\mathbf{4.410 \pm 0.149}$	$\mathbf{4.798 \pm 0.036}$
Observed %	50%	60%	70%	80%	90%

Methods

Attention-based Approach: Multi-Time Attention Networks for Irregularly Sampled Time Series

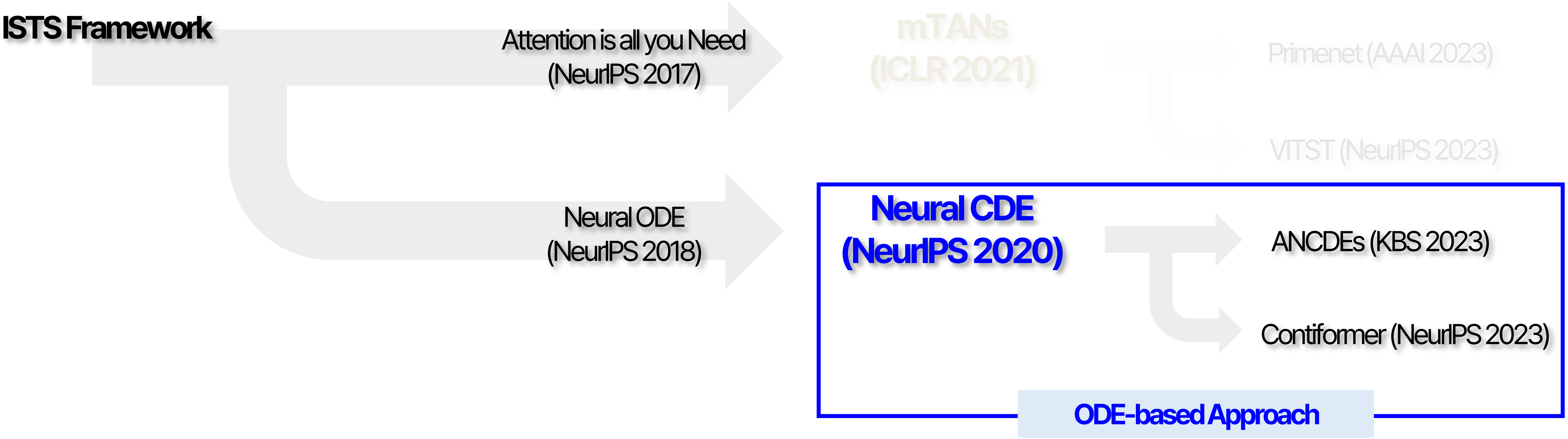
❖ **Experimental results**

- **Interpolation:** ODE 계열 모델들에 비해 전체적으로 높은 성능
- **Classification:** 전체적으로 높은 성능, **타 irregular time series model에 비해 빠른 학습 속도**

Model	AUC Score		Accuracy	time
	PhysioNet	MIMIC-III	Human Activity	per epoch
RNN-Impute	0.764 ± 0.016	0.8249 ± 0.0010	0.859 ± 0.004	0.5
RNN-Δ_t	0.787 ± 0.014	0.8364 ± 0.0011	0.857 ± 0.002	0.5
RNN-Decay	0.807 ± 0.003	0.8392 ± 0.0012	0.860 ± 0.005	0.7
RNN GRU-D	0.818 ± 0.008	0.8270 ± 0.0010	0.862 ± 0.005	0.7
Phased-LSTM	0.836 ± 0.003	0.8429 ± 0.0035	0.855 ± 0.005	0.3
IP-Nets	0.819 ± 0.006	0.8390 ± 0.0011	0.869 ± 0.007	1.3
SeFT	0.795 ± 0.015	0.8485 ± 0.0022	0.815 ± 0.002	0.5
RNN-VAE	0.515 ± 0.040	0.5175 ± 0.0312	0.343 ± 0.040	2.0
ODE-RNN	0.833 ± 0.009	$\mathbf{0.8561 \pm 0.0051}$	0.885 ± 0.008	16.5
L-ODE-RNN	0.781 ± 0.018	0.7734 ± 0.0030	0.838 ± 0.004	6.7
L-ODE-ODE	0.829 ± 0.004	$\mathbf{0.8559 \pm 0.0041}$	0.870 ± 0.028	22.0
mTAND-Enc	0.854 ± 0.001	0.8419 ± 0.0017	$\mathbf{0.907 \pm 0.002}$	**0.1**
mTAND-Full	$\mathbf{0.858 \pm 0.004}$	$\mathbf{0.8544 \pm 0.0024}$	$\mathbf{0.910 \pm 0.002}$	0.2

Background
Research trend

❖ **Research trend**



Methods

ODE-based Approach: Neural Controlled Differential Equations for Irregular Time Series

❖ **Neural Controlled Differential Equations for Irregular Time Series**

- 2026년 1월 기준 854회 인용
- Neural ordinary differential equation(Neural ODE)을 **ISTS에 특화된** 모델로 발전

Neural Controlled Differential Equations for Irregular Time Series

Patrick Kidger **James Morrill** **James Foster** **Terry Lyons**
Mathematical Institute, University of Oxford
The Alan Turing Institute, British Library
{kidger, morrill, foster, tlyons}@maths.ox.ac.uk

Abstract

Neural ordinary differential equations are an attractive option for modelling temporal dynamics. However, a fundamental issue is that the solution to an ordinary differential equation is determined by its initial condition, and there is no mechanism for adjusting the trajectory based on subsequent observations. Here, we demonstrate how this may be resolved through the well-understood mathematics of *controlled differential equations*. The resulting *neural controlled differential equation* model is directly applicable to the general setting of partially-observed irregularly-sampled multivariate time series, and (unlike previous work on this problem) it may utilise memory-efficient adjoint-based backpropagation even across observations. We demonstrate that our model achieves state-of-the-art performance against similar (ODE or RNN based) models in empirical studies on a range of datasets. Finally we provide theoretical results demonstrating universal approximation, and that our model subsumes alternative ODE models.

Methods

ODE-based Approach: Neural Controlled Differential Equations for Irregular Time Series

❖ **Neural Controlled Differential Equations for Irregular Time Series**

- 2026년 1월 기준 854회 인용
- Neural ordinary differential equation(Neural ODE)을 **ISTS에 특화된** 모델로 발전

Neural Controlled Differential Equations for Irregular Time Series

Patrick Kidger **James Morrill** **James Foster** **Terry Lyons**
Mathematical Institute, University of Oxford
The Alan Turing Institute, British Library
{kidger, morrill, foster, tlyons}@maths.ox.ac.uk

Abstract

Neural ordinary differential equations are an attractive option for modelling temporal dynamics. However, a fundamental issue is that the solution to an ordinary differential equation is determined by its initial condition, and there is no mechanism for adjusting the trajectory based on subsequent observations. Here, we demonstrate how this may be resolved through the well-understood mathematics of *controlled differential equations*. The resulting *neural controlled differential equation* model is directly applicable to the general setting of partially-observed irregularly-sampled multivariate time series, and (unlike previous work on this problem) it may utilise memory-efficient adjoint-based backpropagation even across observations. We demonstrate that our model achieves state-of-the-art performance against similar (ODE or RNN based) models in empirical studies on a range of datasets. Finally we provide theoretical results demonstrating universal approximation, and that our model subsumes alternative ODE models.

그래서 ODE가 뭔데?

Methods

ODE-based Approach: Neural Controlled Differential Equations for Irregular Time Series

❖ **Ordinary differential equation (ODE)**

- Differential equation: 종속 변수가 독립 변수의 변화에 따라 어떻게 변하는지를 설명하는 방정식
- **Ordinary differential equation: 단일 변수**에 대해 종속 변수의 변화율을 다루는 방정식

일반 방정식	**미분방정식**	**ODE**
$x + 3 = 5$	$\frac{\partial u}{\partial t} = \alpha \frac{\partial^2 u}{\partial x^2}$	$f'(x) = 2x$
미지수를 찾는 과정	**변화의 규칙을 만족하는 함수**를 찾는 과정	**독립변수가 하나**일 때의 미분방정식

Methods

ODE-based Approach: Neural Controlled Differential Equations for Irregular Time Series

❖ **Ordinary differential equation (ODE)**

- **ODE를 푼다 = 규칙을 만족하는 original function을 찾는다**
- Hidden layer로 함수를 근사하는 (universal approximation theorem) 딥러닝의 과정과 유사

ODE solve

$$f(x) = x^2 + C$$

Original function f를 적분을 통해서 풀이

ODE

$$f'(x) = 2x$$

독립변수가 하나일 때의 미분방정식

Methods

ODE-based Approach: Neural Controlled Differential Equations for Irregular Time Series

❖ **Ordinary differential equation (ODE)**

- **ODE를 푼다 = 규칙을 만족하는 original function을 찾는다**
- Hidden layer로 함수를 근사하는 (universal approximation theorem) 딥러닝의 과정과 유사

Problem?

쉬운 ODE는 간단한 적분으로 풀 수 있다
그런데 딥러닝 함수와 같은 복잡한 function은
어떻게 ODE를 풀어야 하는가?

→ ODE Solver!
(ex. **Euler method**, Runge-Kutta...)

ODE solve

$$f(x) = x^2 + C$$

Original function f를 적분을 통해서 풀이

ODE

$$f'(x) = 2x$$

독립변수가 하나일 때의 미분방정식

Methods

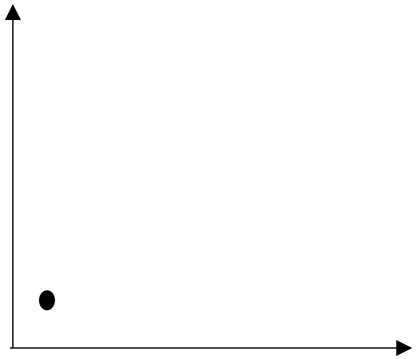
ODE-based Approach: Neural Controlled Differential Equations for Irregular Time Series

❖ **ODE Solver: Euler method**

- Idea: initial point와 도함수로 **original function을 근사**해보자
- 적분은 무수한 더하기이므로, initial point를 통해 함수를 근사하는 무수히 많은 점을 찾아낼 수 있음

$$f'(x) = 2x \qquad initial\ point = (2,1) \qquad y_n = y_{n-1} + h \cdot \frac{\partial y_{n-1}}{\partial x_{n-1}}$$

$$y_1 = 1 + h \cdot 4$$



$$slope\ of\ (2,1) = f'(2) = 4$$

Methods

ODE-based Approach: Neural Controlled Differential Equations for Irregular Time Series

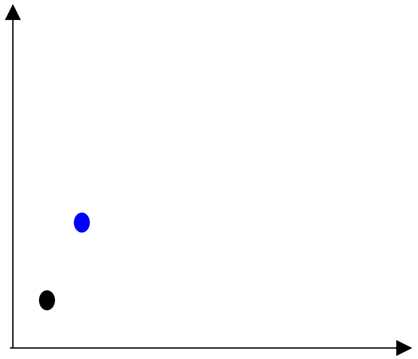
❖ **ODE Solver: Euler method**

- Idea: initial point와 도함수로 **original function을 근사**해보자
- 적분은 무수한 더하기이므로, initial point를 통해 함수를 근사하는 무수히 많은 점을 찾아낼 수 있음

$$f'(x) = 2x \qquad (x_1, y_1) = (4,9) \qquad y_n = y_{n-1} + h \cdot \frac{\partial y_{n-1}}{\partial x_{n-1}}$$

$$y_1 = 1 + h \cdot 4$$

$$y_2 = 9 + h \cdot 8$$



$$slope\ of\ (4,5) = f'(4) = 8$$

Methods

ODE-based Approach: Neural Controlled Differential Equations for Irregular Time Series

❖ **ODE Solver: Euler method**

- Idea: initial point와 도함수로 **original function을 근사**해보자
- 적분은 무수한 더하기이므로, initial point를 통해 함수를 근사하는 무수히 많은 점을 찾아낼 수 있음

$$f'(x) = 2x$$

$$(x_2, y_2) = (6, 25)$$

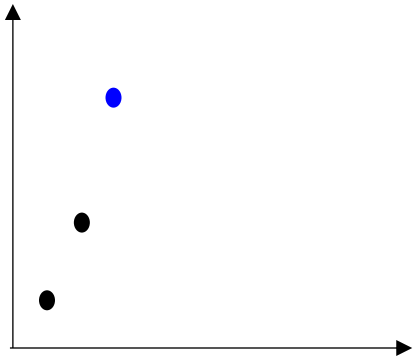
$$y_n = y_{n-1} + h \cdot \frac{\partial y_{n-1}}{\partial x_{n-1}}$$

$$y_1 = 1 + h \cdot 2$$

$$y_2 = 9 + h \cdot 8$$

$$y_3 = 25 + h \cdot 12$$

...



$$slope\ of\ (6,21) = f'(6) = 12$$

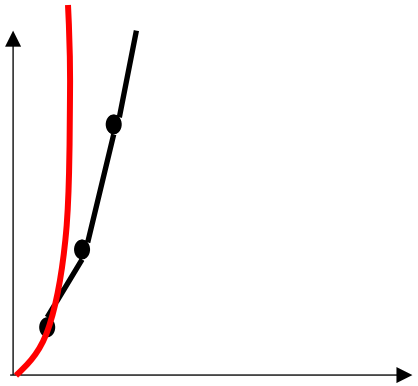
Methods

ODE-based Approach: Neural Controlled Differential Equations for Irregular Time Series

❖ ODE Solver: Euler method

- Idea: initial point와 도함수로 **original function을 근사**해보자
- 적분은 무수한 더하기이므로, initial point를 통해 함수를 근사하는 무수히 많은 점을 찾아낼 수 있음

$$f(x) = x^2 + C$$



$$y_n = y_{n-1} + h \cdot \frac{\partial y_{n-1}}{\partial x_{n-1}}$$

$$= y_0 + h \cdot \frac{\partial y_0}{\partial x_0} + h \cdot \frac{\partial y_1}{\partial x_1} + \cdots + h \cdot \frac{\partial y_{n-1}}{\partial x_{n-1}}$$

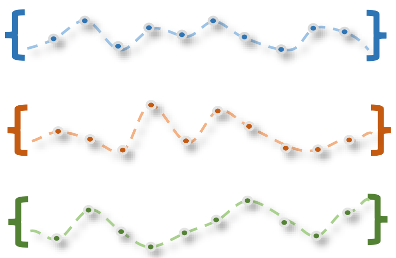
Original function을 근사하기 위해서는
무수히 많은 더하기가 필요하구나!

Methods

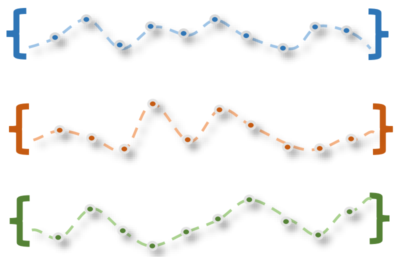
ODE-based Approach: Neural Controlled Differential Equations for Irregular Time Series

❖ **Neural ordinary differential equation (Neural ODE)**

- Idea: Neural network의 hidden state trajectory를 ODE solver로 풀어보자!



$$f(h(t))$$

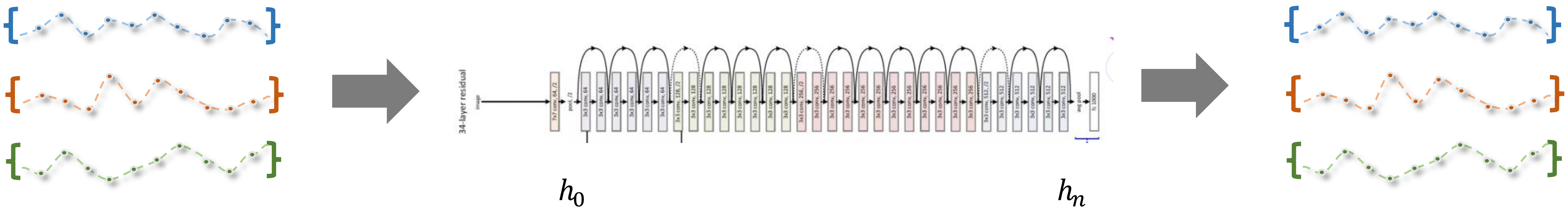


Methods

ODE-based Approach: Neural Controlled Differential Equations for Irregular Time Series

❖ **Neural ordinary differential equation (Neural ODE)**

- Idea: Neural network의 hidden state trajectory를 ODE solver로 풀어보자!
- 이는 **Resnet의 residual connection** 아이디어와 비슷



$$h_n = h_0 + f(h_1, \theta) + f(h_2, \theta) + f(h_3, \theta) + \cdots + f(h_{n-1}, \theta)$$

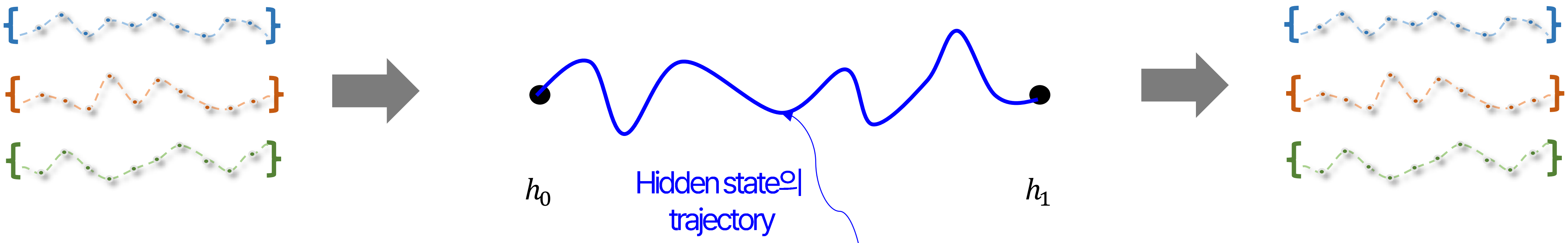
최종 hidden state의 계산은 **layer를 지나면서의 변화량의 합산으로 이루어진다!**

Methods

ODE-based Approach: Neural Controlled Differential Equations for Irregular Time Series

❖ **Neural ordinary differential equation (Neural ODE)**

- Idea: Neural network의 hidden state trajectory를 ODE solver로 풀어보자!
- 이는 **Resnet의 residual connection** 아이디어와 비슷 → **hidden state를 discrete가 아니라 continuous로 푼다는 것이 차이**



$$h_1 = h_0 + \int_0^1 f(h(t), t; \theta)\, dt$$

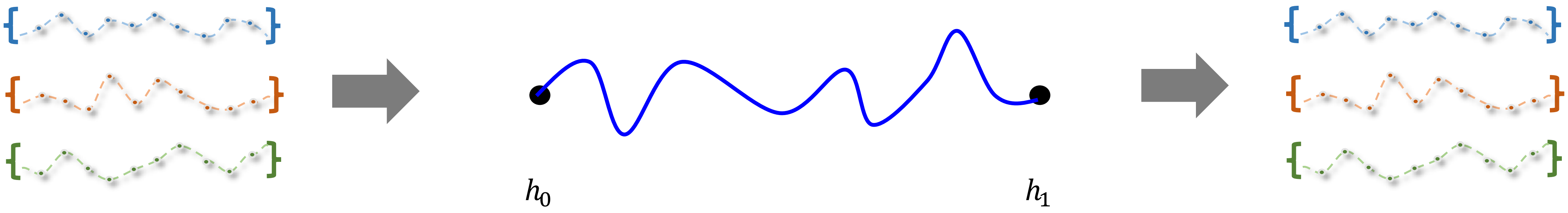
1부터 n까지의 discrete space가 아닌
Continuous한 trajectory로 가정

Methods

ODE-based Approach: Neural Controlled Differential Equations for Irregular Time Series

❖ **Neural ordinary differential equation (Neural ODE)**

- Forward: h_0을 initial state, h_1을 final state로 정의해 **Euler method**로 풀어주기



$$h_1 = h_0 + \int_0^1 f(h(t), t; \theta) dt$$

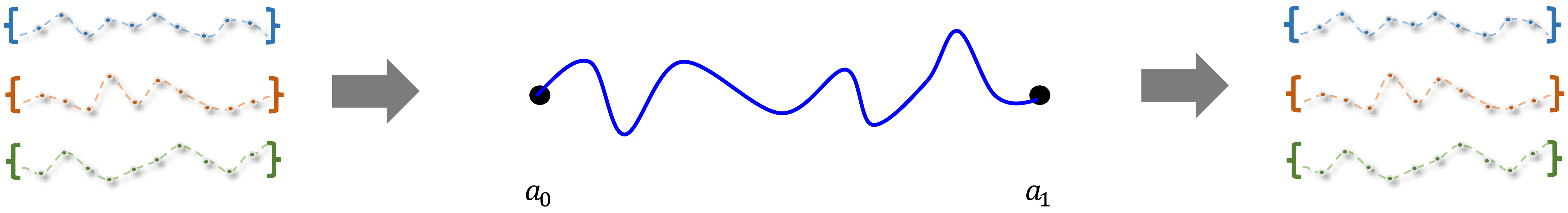
Forward → ODE solver (ex. Euler method)로 해결

Methods

ODE-based Approach: Neural Controlled Differential Equations for Irregular Time Series

❖ Neural ordinary differential equation (Neural ODE)

- Forward: h_0을 initial state, h_1을 final state로 정의해 **Euler method**로 풀어주기
- Backward: a_1을 initial state, a_0을 final state로 정의해 **Adjoint sensitivity method**로 풀어주기



$$a_0 = a_1 + \int_1^0 f(a(t), t; \theta) dt$$

$$a(t) = \frac{\partial L}{\partial h(t)}$$

Forward → ODE solver (ex. Euler method)로 해결

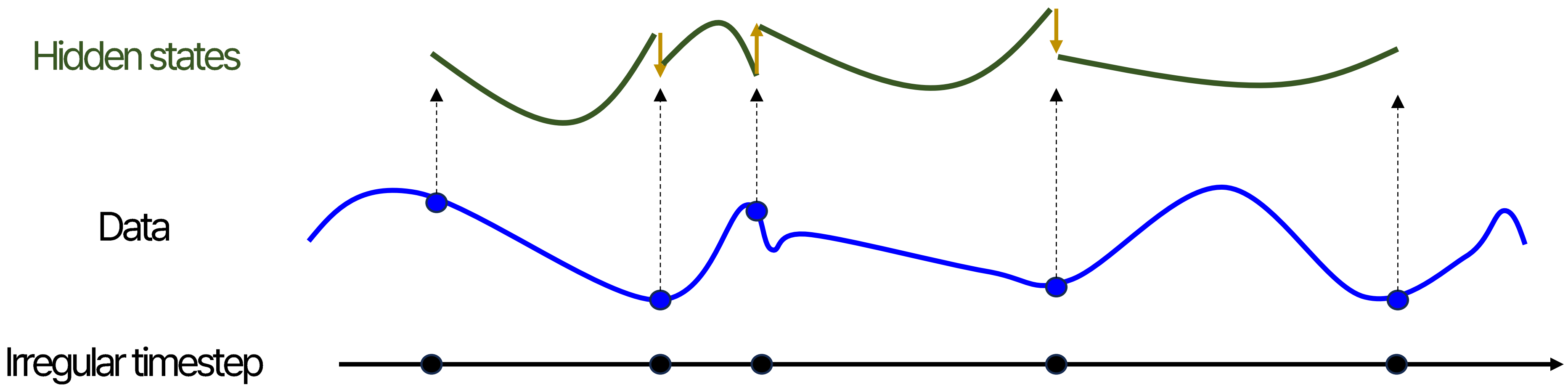
Backward ← ODE solver (ex. Euler method)로 해결

Methods

ODE-based Approach: Neural Controlled Differential Equations for Irregular Time Series

❖ **The problems of Neural ODE?**

- Really continuous?: Neural ODE는 hidden states의 continuous trajectory를 모사하지만, **데이터의 continuous path를 고려**하지는 않음
- Initial state?: Neural ODE는 **initial state에 의존,** time series의 순서에 따라 조정할 수 없음



What we solve?

$$\frac{dh}{dt}(t) = f_\theta(h(t))$$

Initial state?

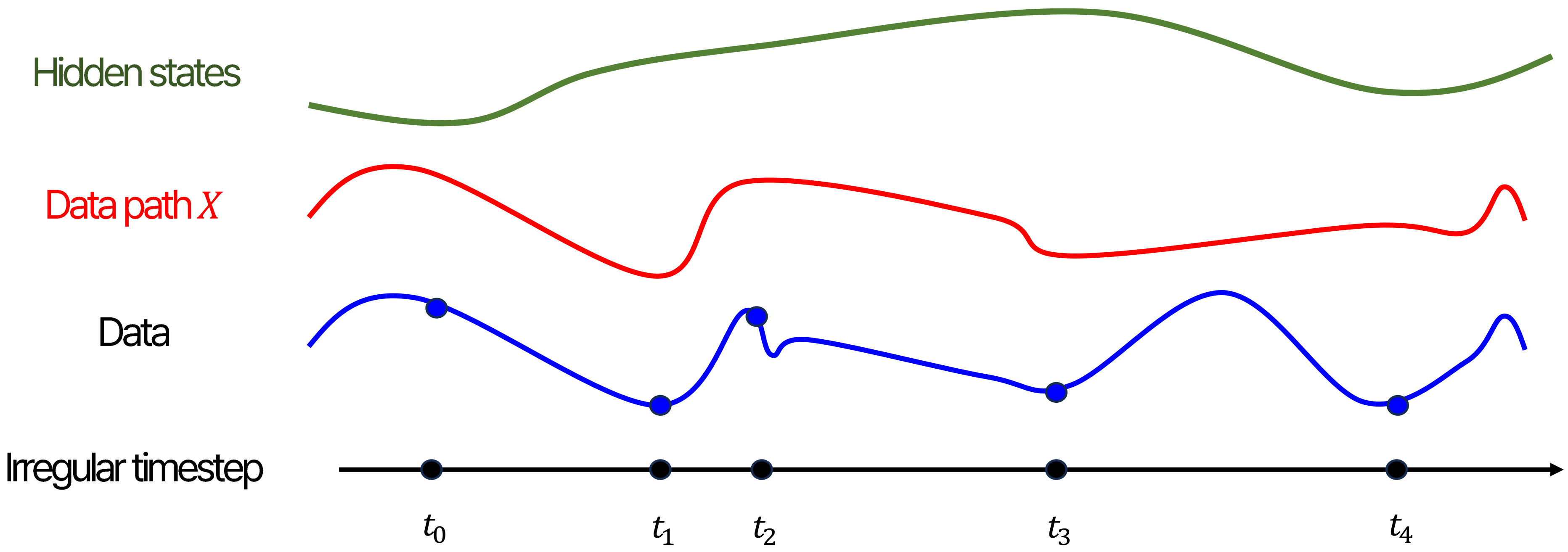
$$h(0) = l_\theta(x)$$

Methods

ODE-based Approach: Neural Controlled Differential Equations for Irregular Time Series

❖ **Neural controlled differential equations (Neural CDE)**

- Data path: Irregular한 data points를 cubic interpolation으로 보간 후 연속적인 데이터 경로 X를 생성
- Continuous한 path에 기반해 데이터에 내재된 변화를 연속적으로 modeling



What we solve?

$$\frac{dh}{dt}(t) = f_\theta(h(t))\frac{dX}{dt}(t)$$

Initial state?

$$h(0) = l_\theta(t_0, x_0)$$

Methods

ODE-based Approach: Neural Controlled Differential Equations for Irregular Time Series

❖ **Experimental results**

- Classification, prediction 등 여러 tasks에서 ODE 기반 모델들에 비해 좋은 성능 기록
- ODE의 강점인 cost efficiency를 유지하면서도 시계열 task에서 상당한 성능 개선

Table 1: Test accuracy (mean ± std, computed across five runs) and memory usage on CharacterTrajectories. Memory usage is independent of repeats and of amount of data dropped.

Model	Test Accuracy			Memory usage (MB)
	30% dropped	50% dropped	70% dropped	
GRU-ODE	92.6% ± 1.6%	86.7% ± 3.9%	89.9% ± 3.7%	1.5
GRU-Δt	93.6% ± 2.0%	91.3% ± 2.1%	90.4% ± 0.8%	15.8
GRU-D	94.2% ± 2.1%	90.2% ± 4.8%	91.9% ± 1.7%	17.0
ODE-RNN	95.4% ± 0.6%	96.0% ± 0.3%	95.3% ± 0.6%	14.8
Neural CDE (ours)	**98.7% ± 0.8%**	**98.8% ± 0.2%**	**98.6% ± 0.4%**	**1.3**

Methods

ODE-based Approach: Neural Controlled Differential Equations for Irregular Time Series

❖ **Experimental results**

- Classification, prediction 등 여러 tasks에서 ODE 기반 모델들에 비해 좋은 성능 기록
- ODE의 강점인 cost efficiency를 유지하면서도 시계열 task에서 상당한 성능 개선

Table 2: Test AUC (mean ± std, computed across five runs) and memory usage on PhysioNet sepsis prediction. 'OI' refers to the inclusion of observational intensity, 'No OI' means without it. Memory usage is independent of repeats.

Model	Test AUC		Memory usage (MB)	
	OI	No OI	OI	No OI
GRU-ODE	0.852 ± 0.010	0.771 ± 0.024	454	273
GRU-Δt	0.878 ± 0.006	0.840 ± 0.007	837	826
GRU-D	0.871 ± 0.022	**0.850 ± 0.013**	889	878
ODE-RNN	0.874 ± 0.016	0.833 ± 0.020	696	686
Neural CDE (ours)	**0.880 ± 0.006**	0.776 ± 0.009	**244**	**122**

Methods

ODE-based Approach: Neural Controlled Differential Equations for Irregular Time Series

❖ **Experimental results**

- Classification, prediction 등 여러 tasks에서 ODE 기반 모델들에 비해 좋은 성능 기록
- ODE의 강점인 cost efficiency를 유지하면서도 시계열 task에서 상당한 성능 개선

Table 3: Test Accuracy (mean ± std, computed across five runs) and memory usage on Speech Commands. Memory usage is independent of repeats.

Model	Test Accuracy	Memory usage (GB)
GRU-ODE	47.9% ± 2.9%	**0.164**
GRU-Δt	43.3% ± 33.9%	1.54
GRU-D	32.4% ± 34.8%	1.64
ODE-RNN	65.9% ± 35.6%	1.40
Neural CDE (ours)	**89.8% ± 2.5%**	0.167

Conclusion

❖ **Multi-Time Attention Networks for Irregularly Sampled Time Series**

- Transformer의 attention을 가공하여 **ISTS에 특화된 attention module** 제안
- Irreguarly sampled된 시간을 **유사도 기반**으로 고정 벡터로 변환하여 딥러닝 모듈에 적용
- VAE와 결합하여 ISTS task에서 우수한 성능 기록

❖ **Neural Controlled Differential Equations for Irregular Time Series**

- Neural ODE를 사용하여 hidden state의 **continuous한 trajectory를 모델링**
- Neural ODE가 간과하던 data path를 보간으로 생성하면서, ISTS에 특화된 **Neural ODE** 제안
- Complexity를 유지하면서 ODE 계열 모델 대비 성능 개선

고맙습니다