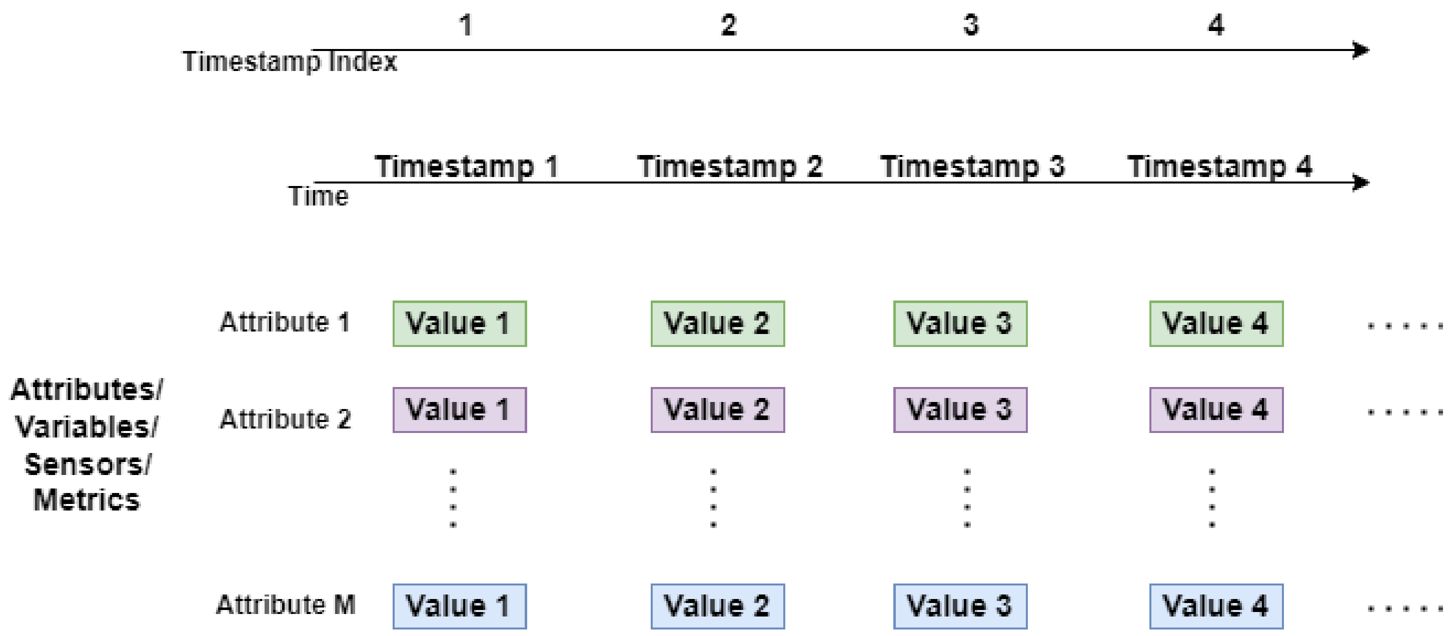
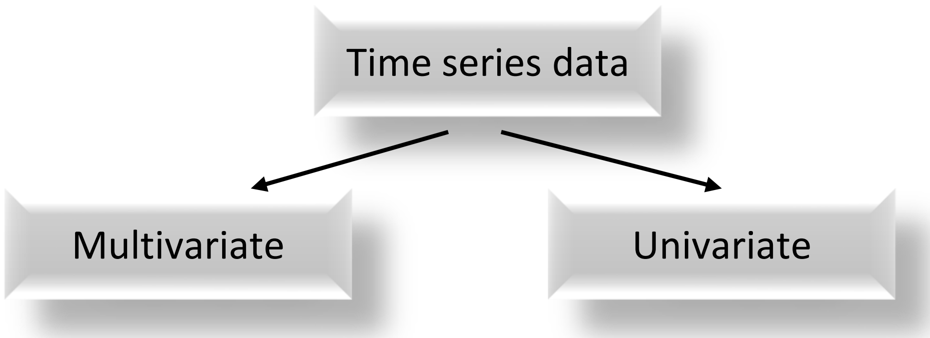
Introduction
to
Time Series

Introduction

- Time series is quantitative observations recorded over time in chronological order

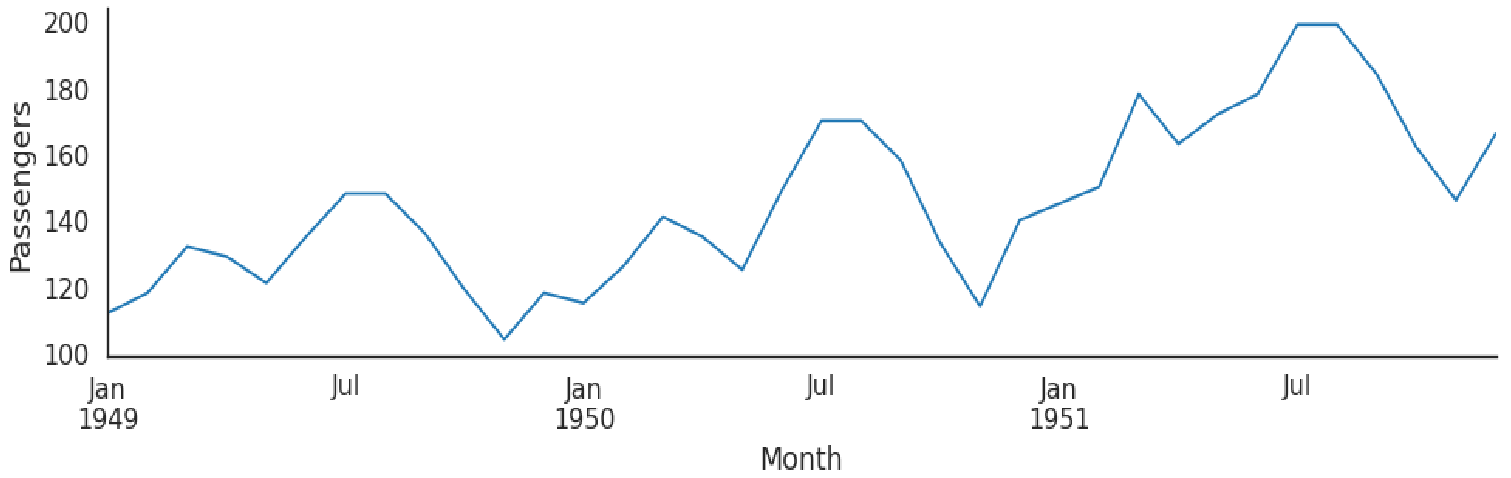




Introduction

- **A univariate times series**

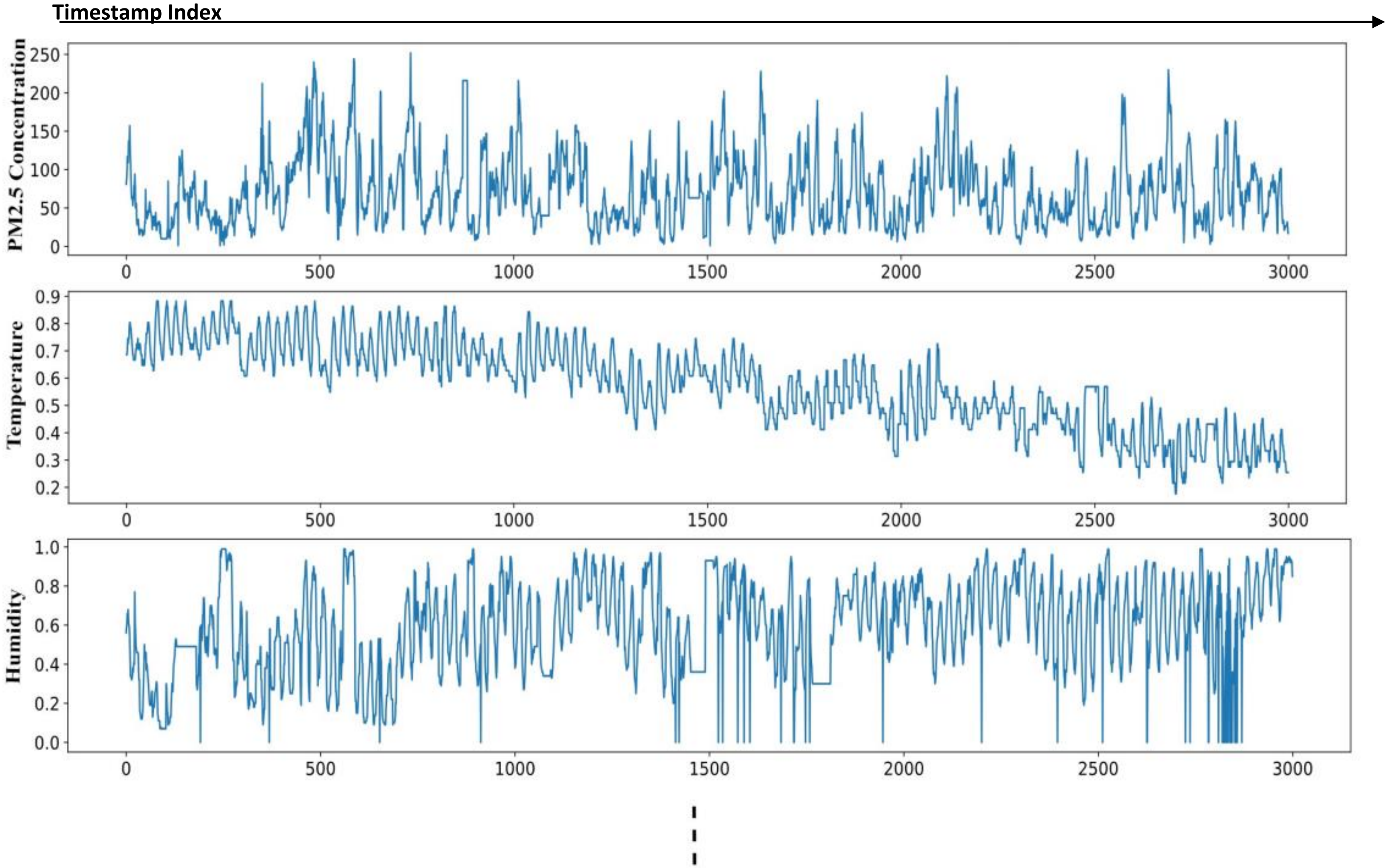
	Month	#Passengers
0	1949-01	112
1	1949-02	118
2	1949-03	132
3	1949-04	129
4	1949-05	121
5	1949-06	135
6	1949-07	148
7	1949-08	148
8	1949-09	136
9	1949-10	119
10	1949-11	104
11	1949-12	118
12	1950-01	115



Air Passenger Dataset

Introduction

- **A multivariate times series**

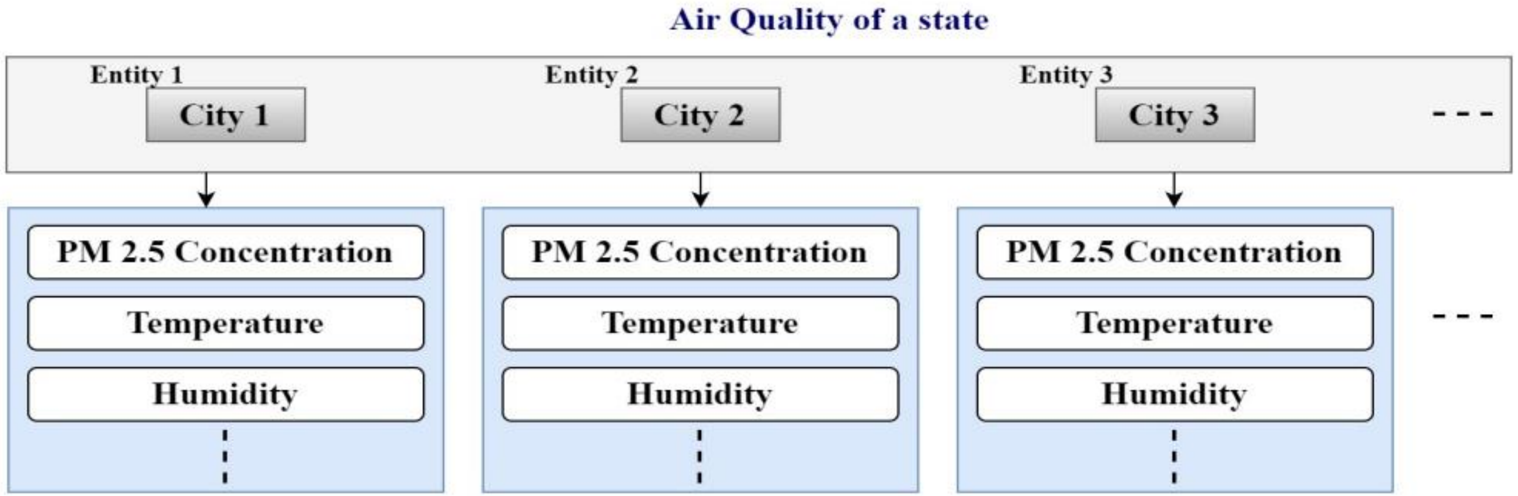


Air Quality Dataset

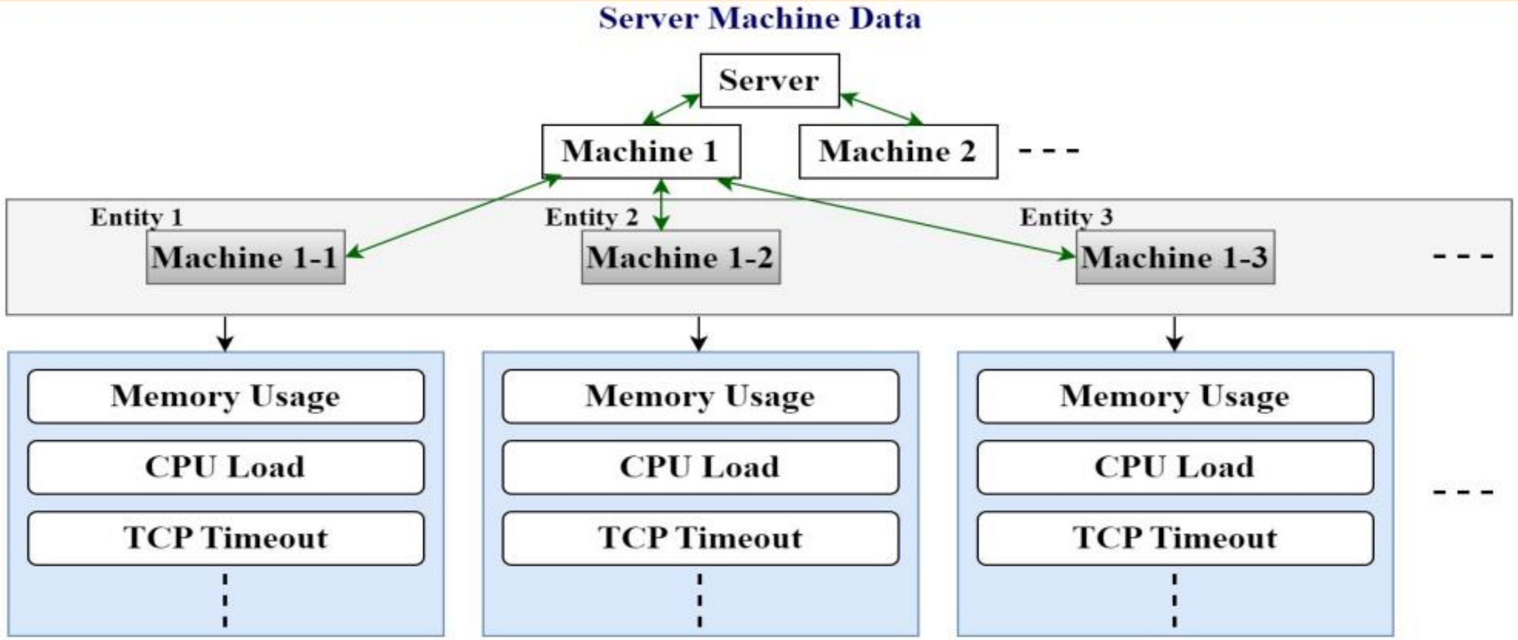
Introduction

- **Multi-entity environment**

Example 1 :



Example 2 :



Introduction

- **A Time Series Data**

$$
\begin{array}{ccccc}
T_1 & T_2 & T_3 & \cdots & T_N \\
\hline
X_1^1 & X_2^1 & X_3^1 & \cdots & X_N^1 \\
X_1^2 & X_2^2 & X_3^2 & \cdots & X_N^2 \\
\vdots & \vdots & \vdots & & \vdots \\
X_1^M & X_2^M & X_3^M & \cdots & X_N^M
\end{array}
$$

- N : Number of timestamps
- M : Number of attributes
- X_t^m : Value of an attribute m at timestamp t (m ϵ M , t ϵ T)

- Time series data is analysed for three broad categories of tasks: Forecasting, classification, and anomaly detection

Forecasting in multivariate time series

- **Single step forecasting**

Input

$$\begin{matrix} X^1_{t-w-1} & X^1_{t-w} & X^1_{t-w+1} & \cdots & X^1_t \\ X^2_{t-w-1} & X^2_{t-w} & X^2_{t-w+1} & \cdots & X^2_t \\ \vdots & \vdots & \vdots & & \vdots \\ X^M_{t-w-1} & X^M_{t-w} & X^M_{t-w+1} & \cdots & X^M_t \end{matrix}$$

Output

$$\begin{matrix} X^1_{t+1} \\ X^2_{t+1} \\ \\ X^M_{t+1} \end{matrix}$$

- **Multi step forecasting**

Input

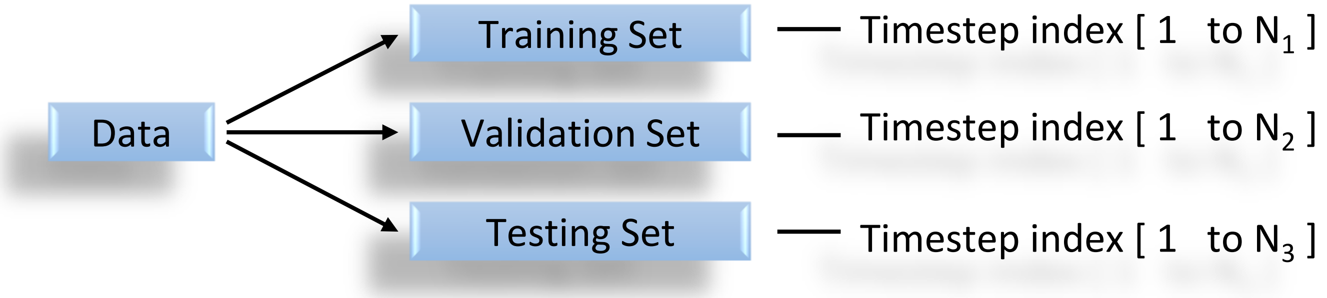
$$\begin{matrix} X^1_{t-w-1} & X^1_{t-w} & X^1_{t-w+1} & \cdots & X^1_t \\ X^2_{t-w-1} & X^2_{t-w} & X^2_{t-w+1} & \cdots & X^2_t \\ \vdots & \vdots & \vdots & & \vdots \\ X^M_{t-w-1} & X^M_{t-w} & X^M_{t-w+1} & \cdots & X^M_t \end{matrix}$$

Output

$$\begin{matrix} X^1_{t+1} & X^1_{t+2} & \cdots & X^1_{t+s} \\ X^2_{t+1} & X^2_{t+2} & \cdots & X^2_{t+s} \\ \\ X^M_{t+1} & X^M_{t+2} & \cdots & X^M_{t+s} \end{matrix}$$

- t : Timestamp index
- w: Window size / number of time stamps of historical observations
- M: Number of attributes

Pre-processing for forecasting



- **Input-Output samples for training**

Sample 1	T_1	T_2	T_3	- - -	T_w	T_{w+1}
Sample 2	T_2	T_3	T_4	- - -	T_{w+1}	T_{w+2}
	⋮	⋮	⋮		⋮	
Sample N_1-w	T_{N_1-w}	T_{N_1-w+1}	T_{N_1-w+2}	- - -	T_{N_1-1}	T_{N_1}

Pre-processing for forecasting

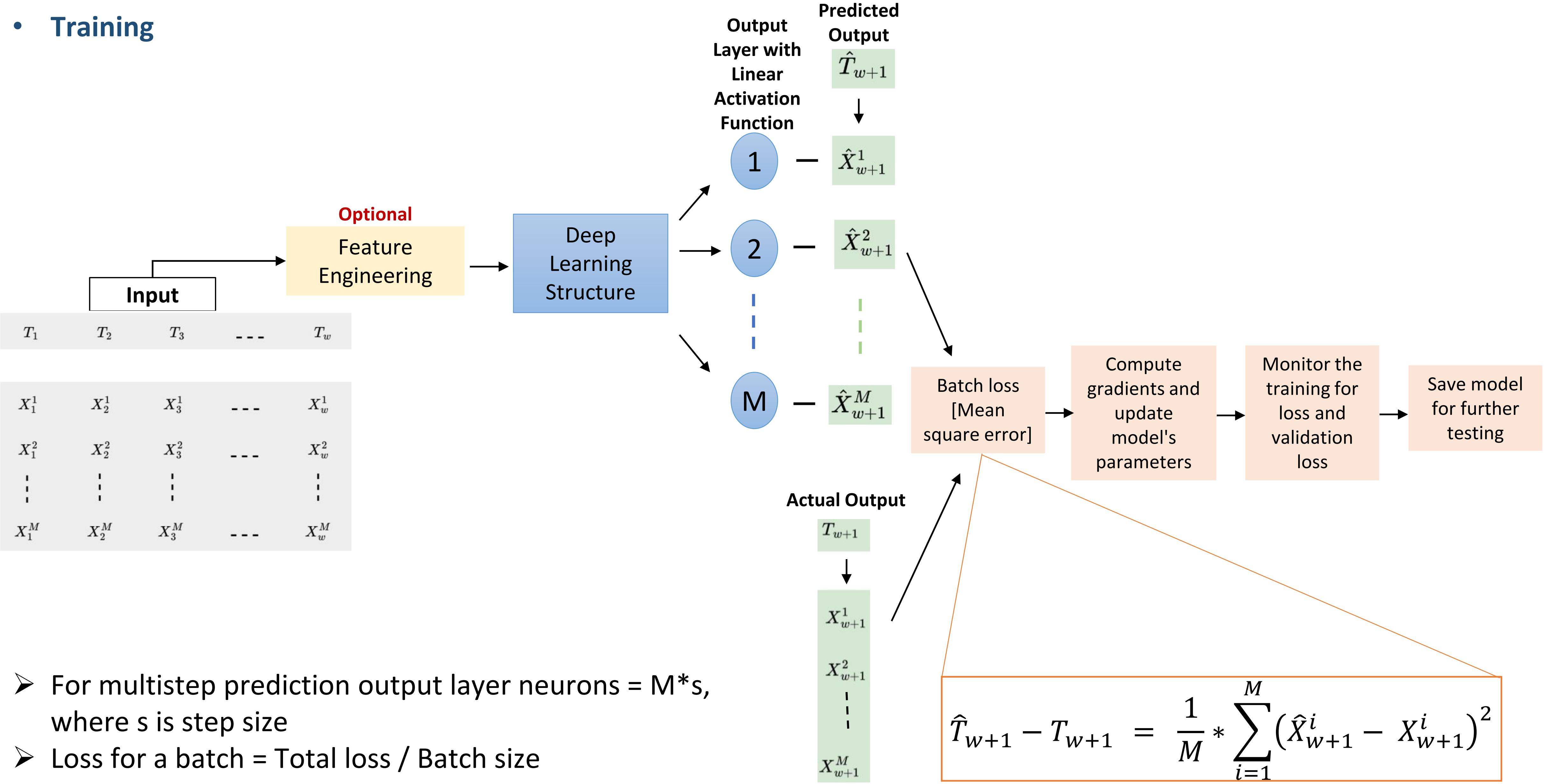
Sample 1	T_1	T_2	T_3	---	T_w	T_{w+1}

$\downarrow$

X_1^1	X_2^1	X_3^1	---	X_w^1	X_{w+1}^1
X_1^2	X_2^2	X_3^2	---	X_w^2	X_{w+1}^2
$\vdots$	$\vdots$	$\vdots$		$\vdots$	
X_1^M	X_2^M	X_3^M	---	X_w^M	X_{w+1}^M

A complete pipeline

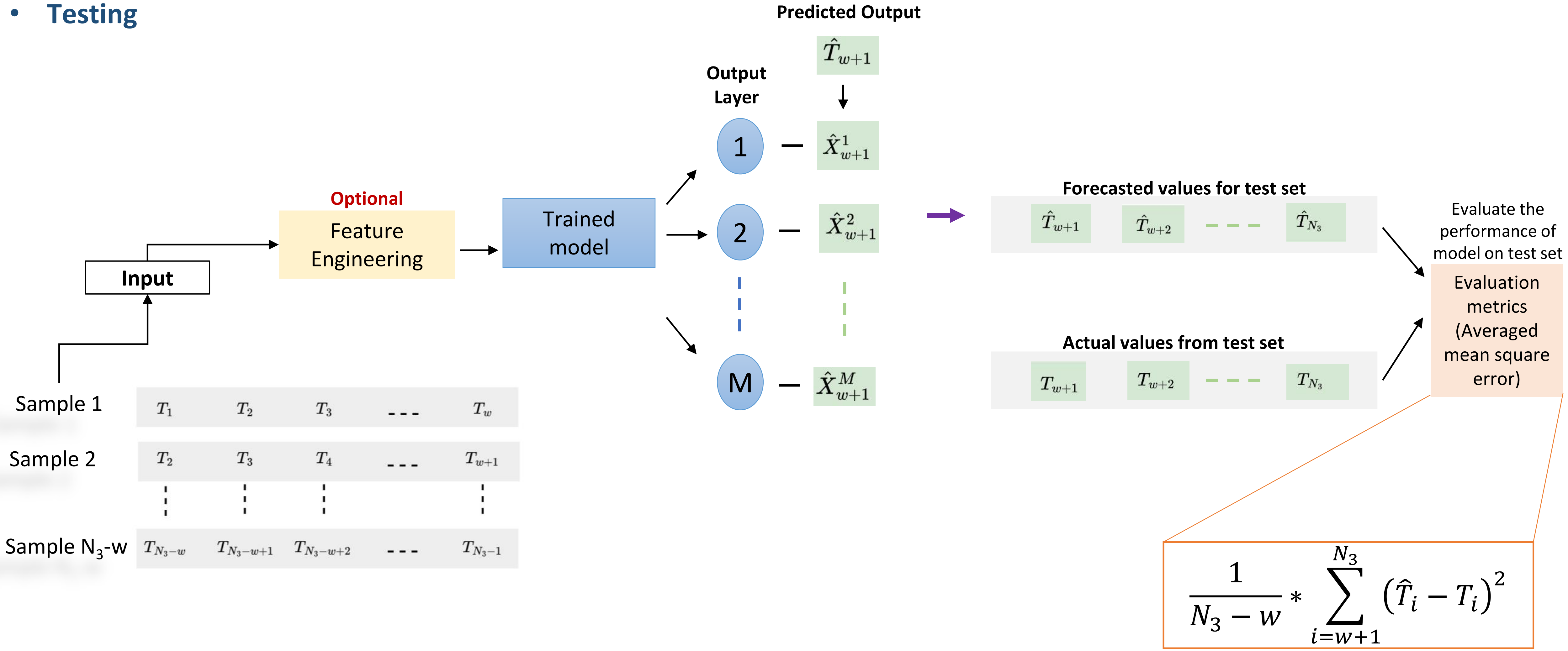
- **Training**



- For multistep prediction output layer neurons = M*s, where s is step size
- Loss for a batch = Total loss / Batch size

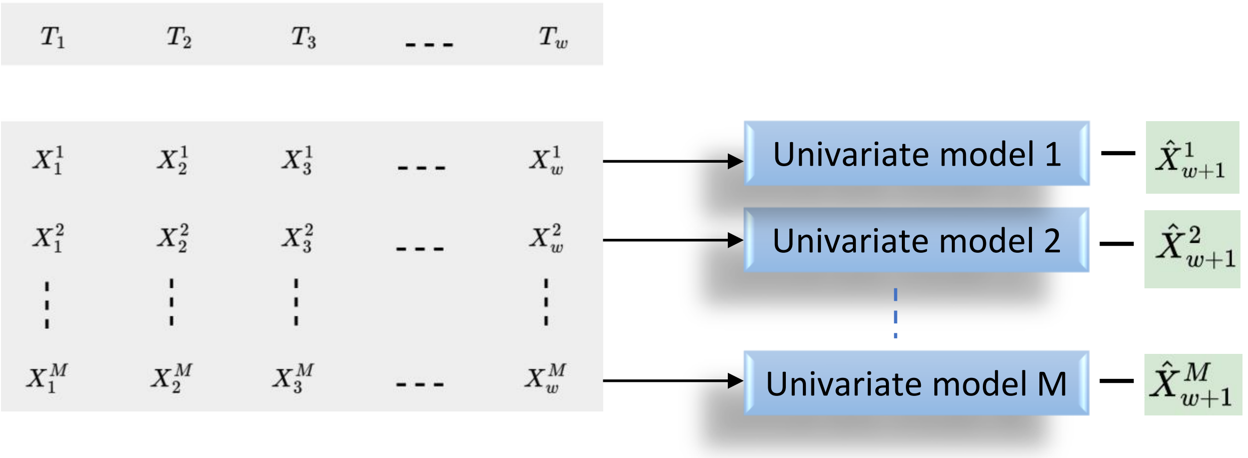
A complete pipeline

- Testing



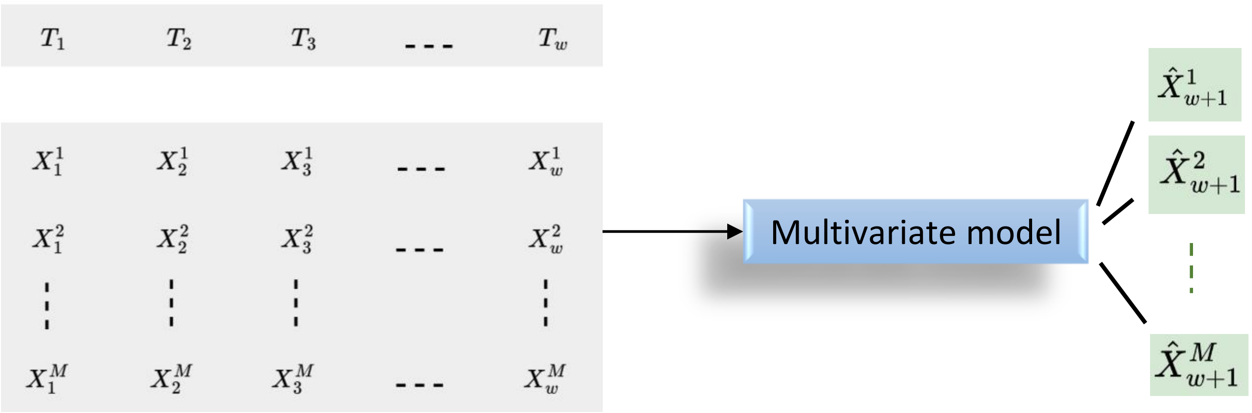
Forecasting methods

- **Based on univariate models**



- Auto Regression (AR)
- Moving Average (MA)
- ARMA
- ARIMA
- Seasonal ARIMA
- Profet model by facebook
- Feed forward network
- N-Beats

- **Based on Multivariate models**



- RNN / GRU / LSTM
- CNN

Forecasting methods

➢ Univariate models

- **AR(h) :** $\hat{T}_i = \beta_0 + \beta_1 \hat{T}_{i-1} + \beta_2 \hat{T}_{i-2} + \dots + \beta_h \hat{T}_{i-h} + error_i$

- **MA(h) :** $\hat{T}_i = \beta_0 + \beta_1 e_{i-1} + \beta_2 e_{i-2} + \dots + \beta_h e_{i-h} + error_i$

- **ARMA :** AR(1) + MA(1) = [$\hat{T}_i = C + \beta_1 \hat{T}_{i-1} + \beta_1 e_{i-1} + error_i$] or ARIMA(1,0,1)

- **ARIMA :**
  - AR + MA + Lag differencing
  - ARIMA(1,1,1) = [$\hat{T}_i = C + \beta_1 \widehat{(T_{i-1}} - \hat{T}_{i-2}) + \beta_1 e_{i-1} + error_i$]

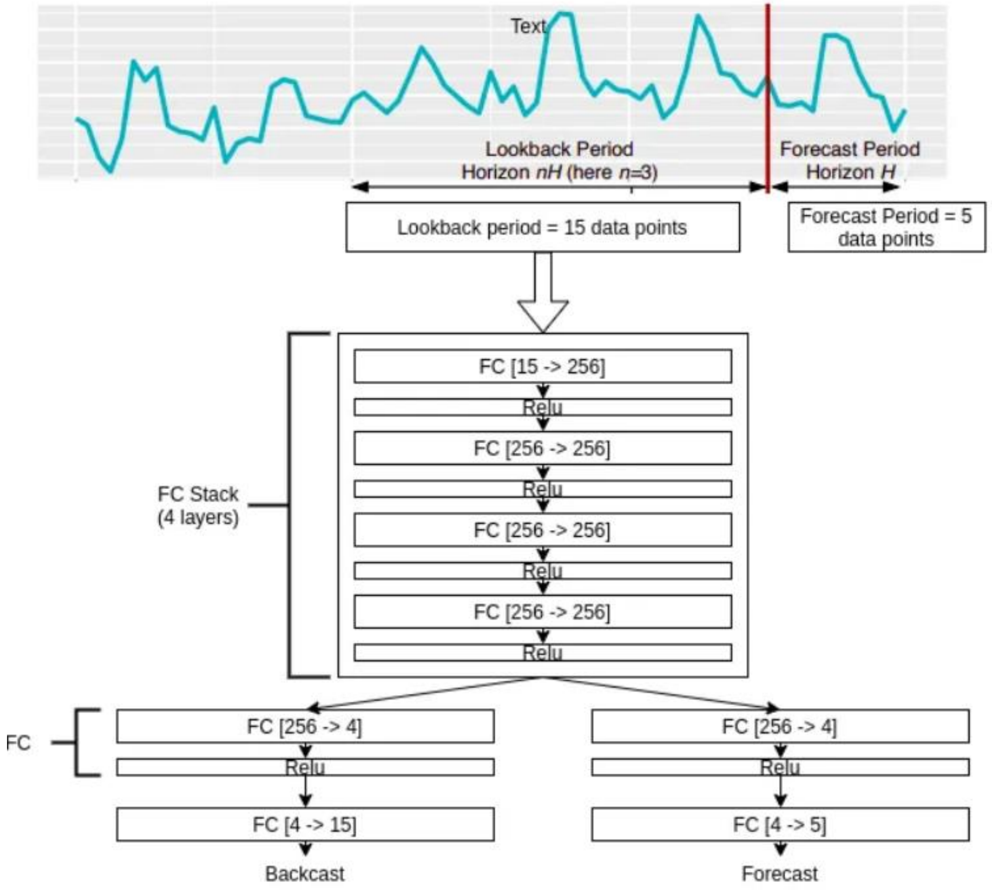
- **SARIMA :** AR + MA + Lag differencing + Seasonal differencing

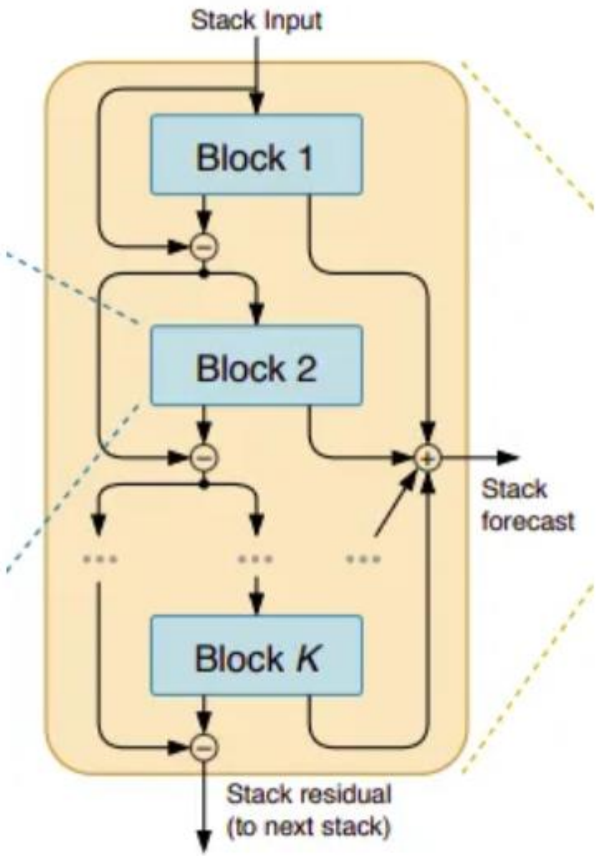
- **Prophet :**
  - Introduced by Facebook
  - Trend + Seasonality + Holiday + Error

- **Feed forward Network**

- **N-Beats :**
  - A deep neural architecture based on backward and forward residual links

Forecasting methods

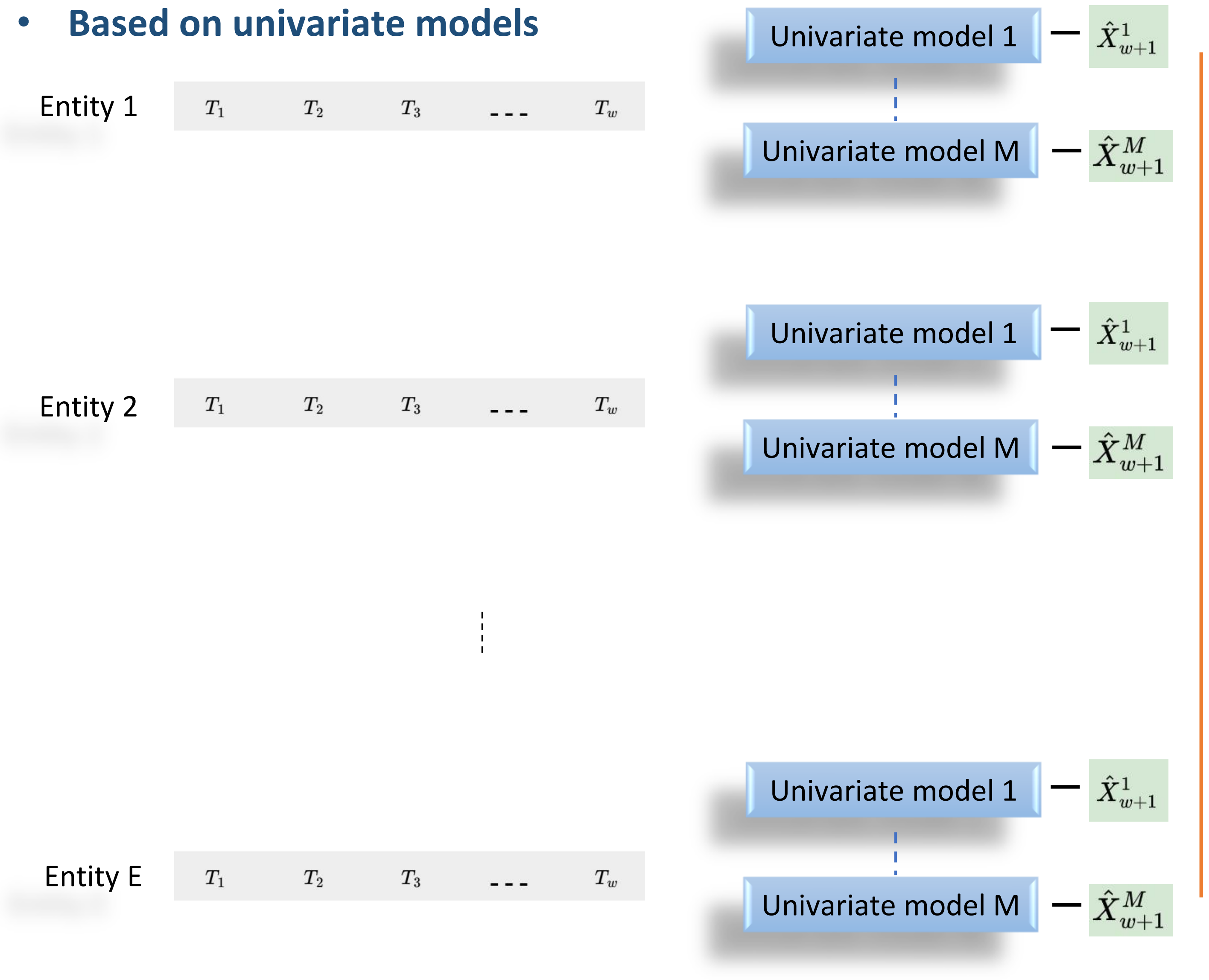




- Univariate models
- Computationally expensive for multivariate series and multi-entity environment
- Not able to capture relationship between attributes

Forecasting in multi-entity environment

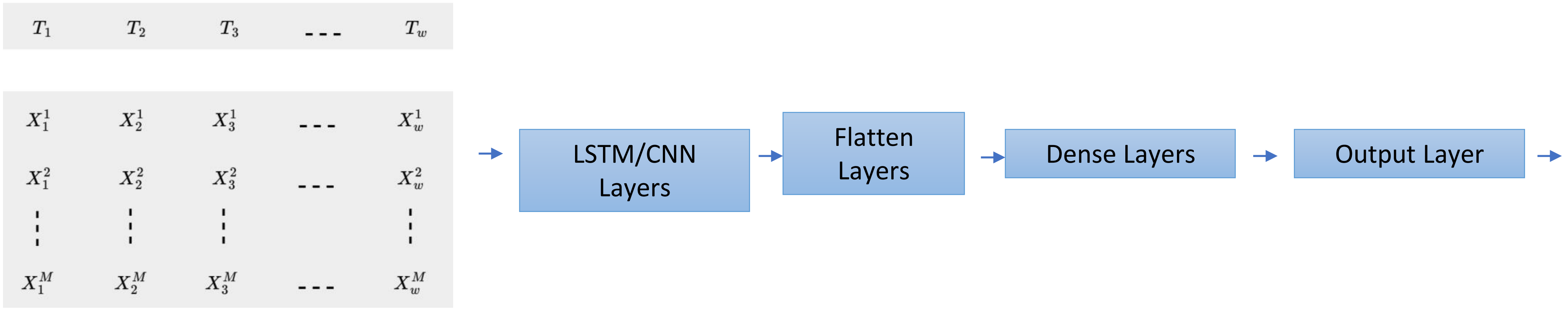
- **Based on univariate models**



$$\text{Error} = \frac{1}{N*M*E} * \sum_{e=1}^{E} \sum_{m=1}^{M} \sum_{i=1}^{N} \left(\hat{X}_i^{m,e} - X_i^{m,e}\right)^2$$

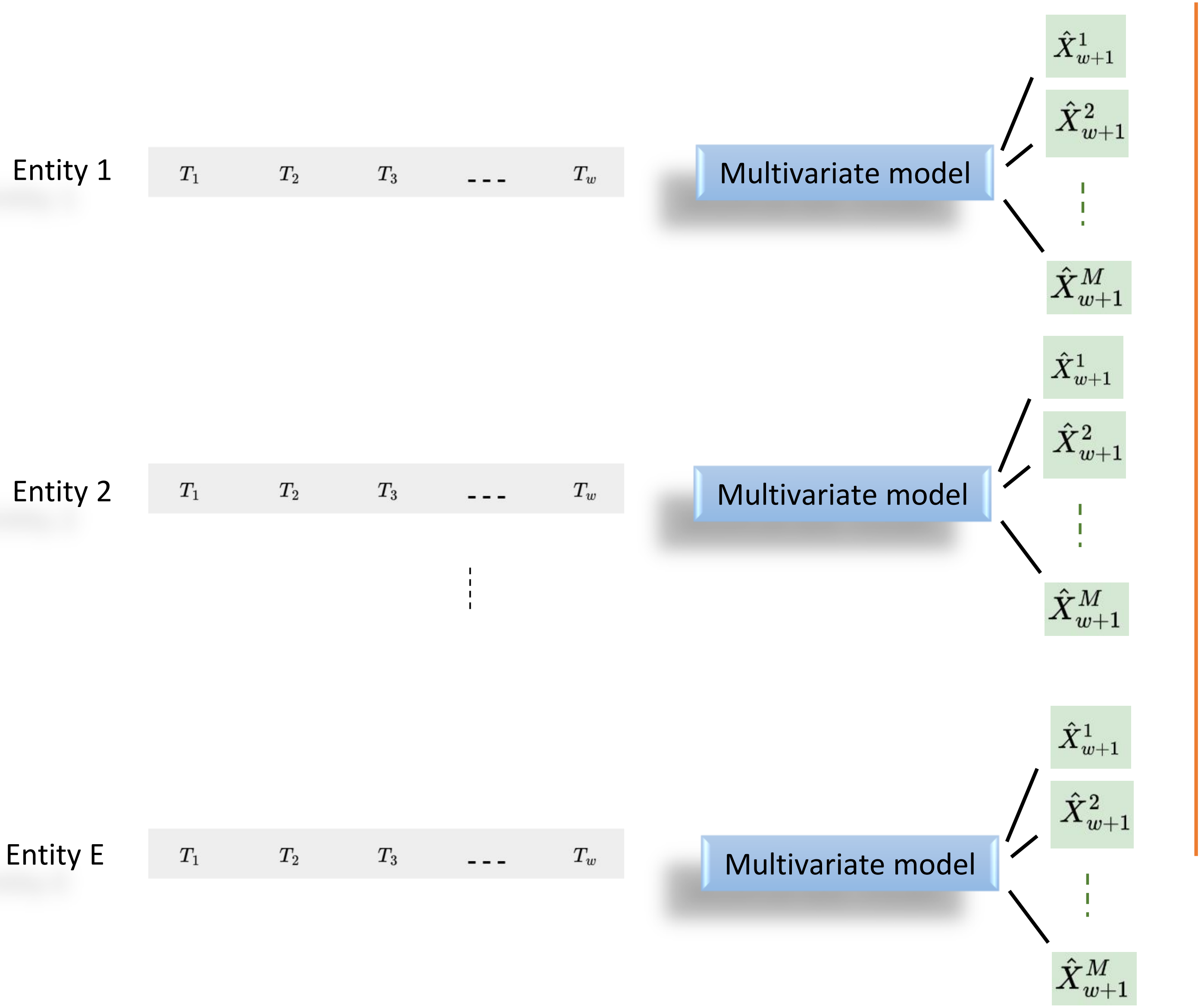
Forecasting methods

Multivariate models



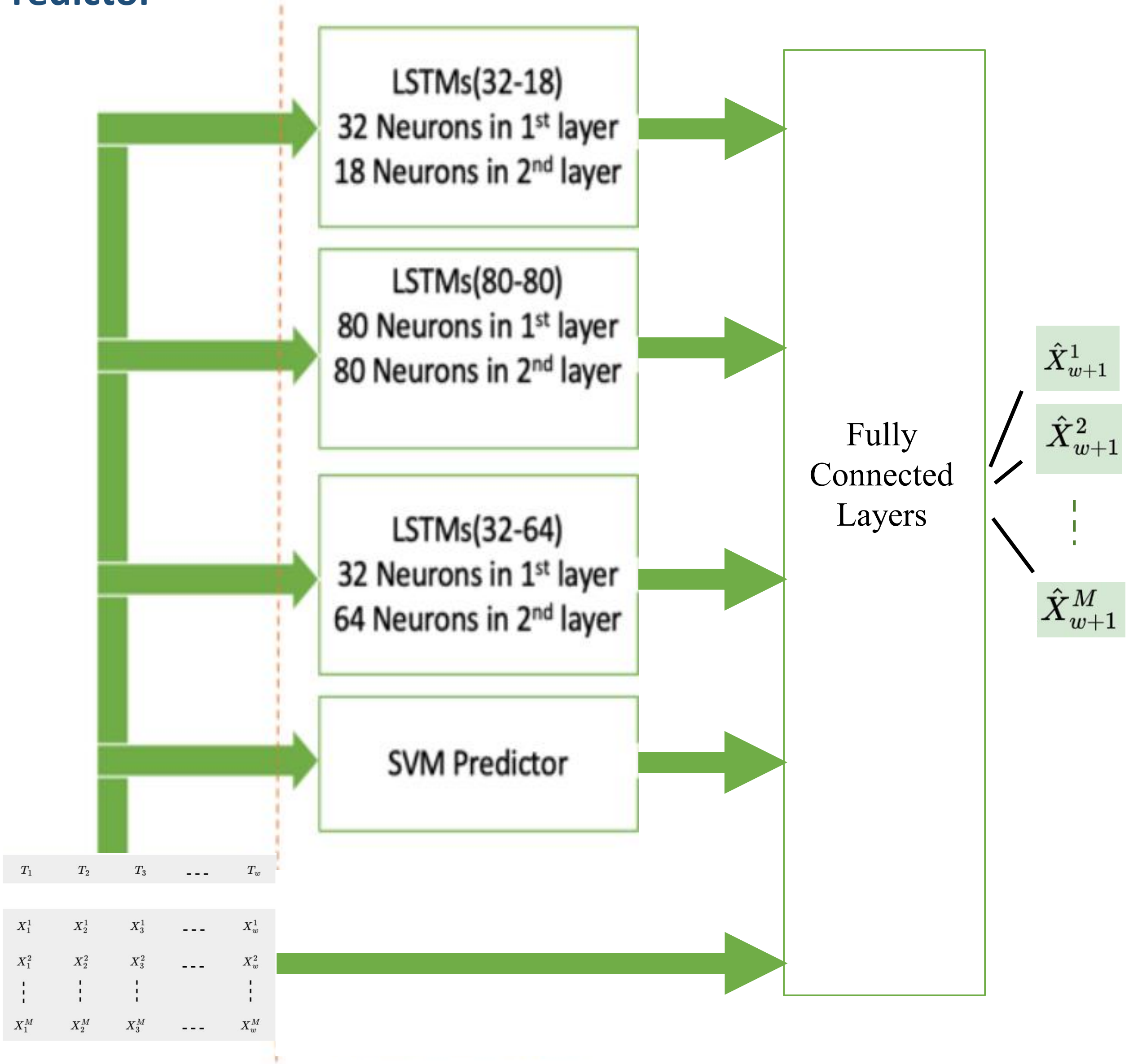
Forecasting in multi-entity environment

- **Based on Multivariate models**



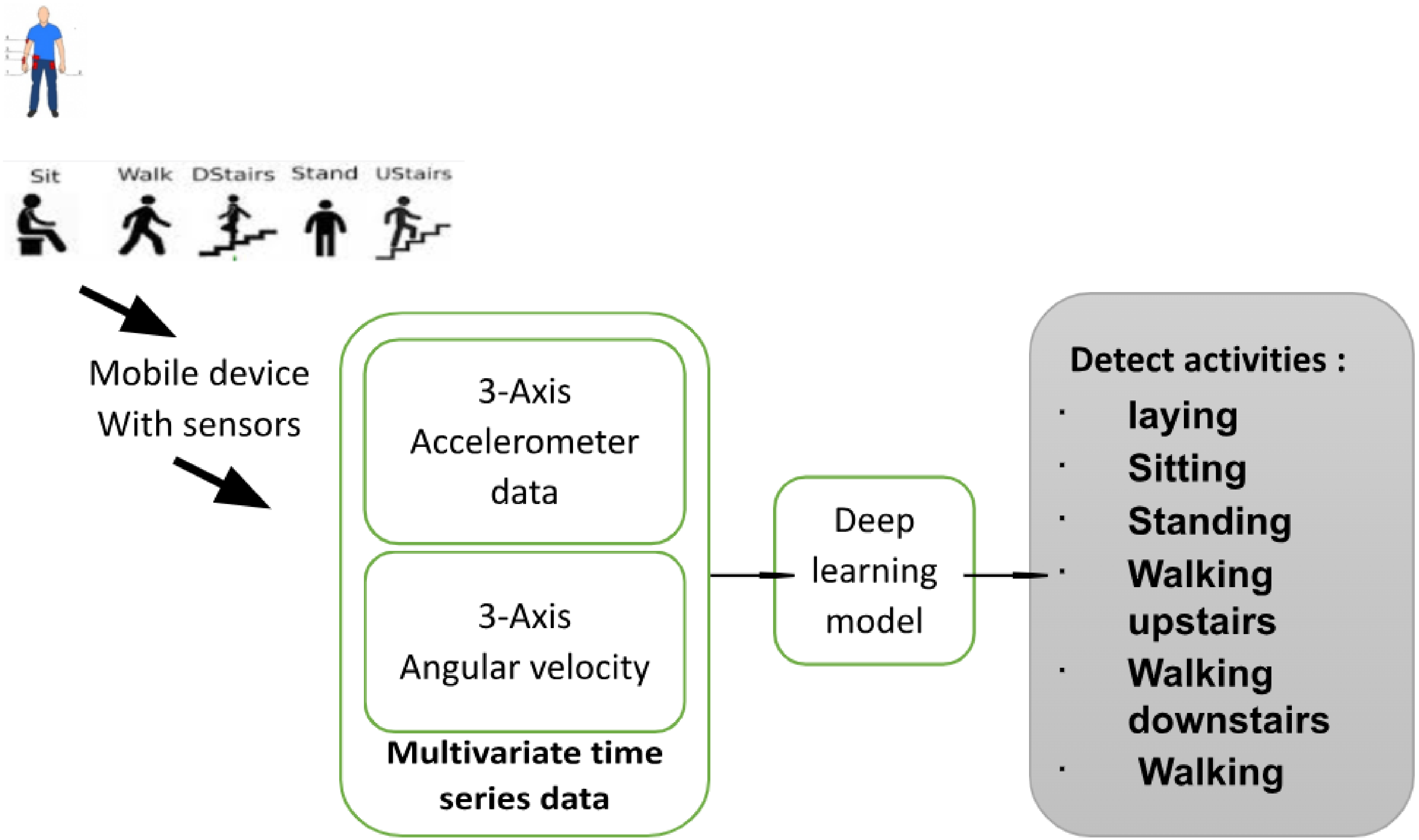
Forecasting method

- **A Stacked Predictor**



Classification in multivariate time series

- **Example**



Classification in multivariate time series

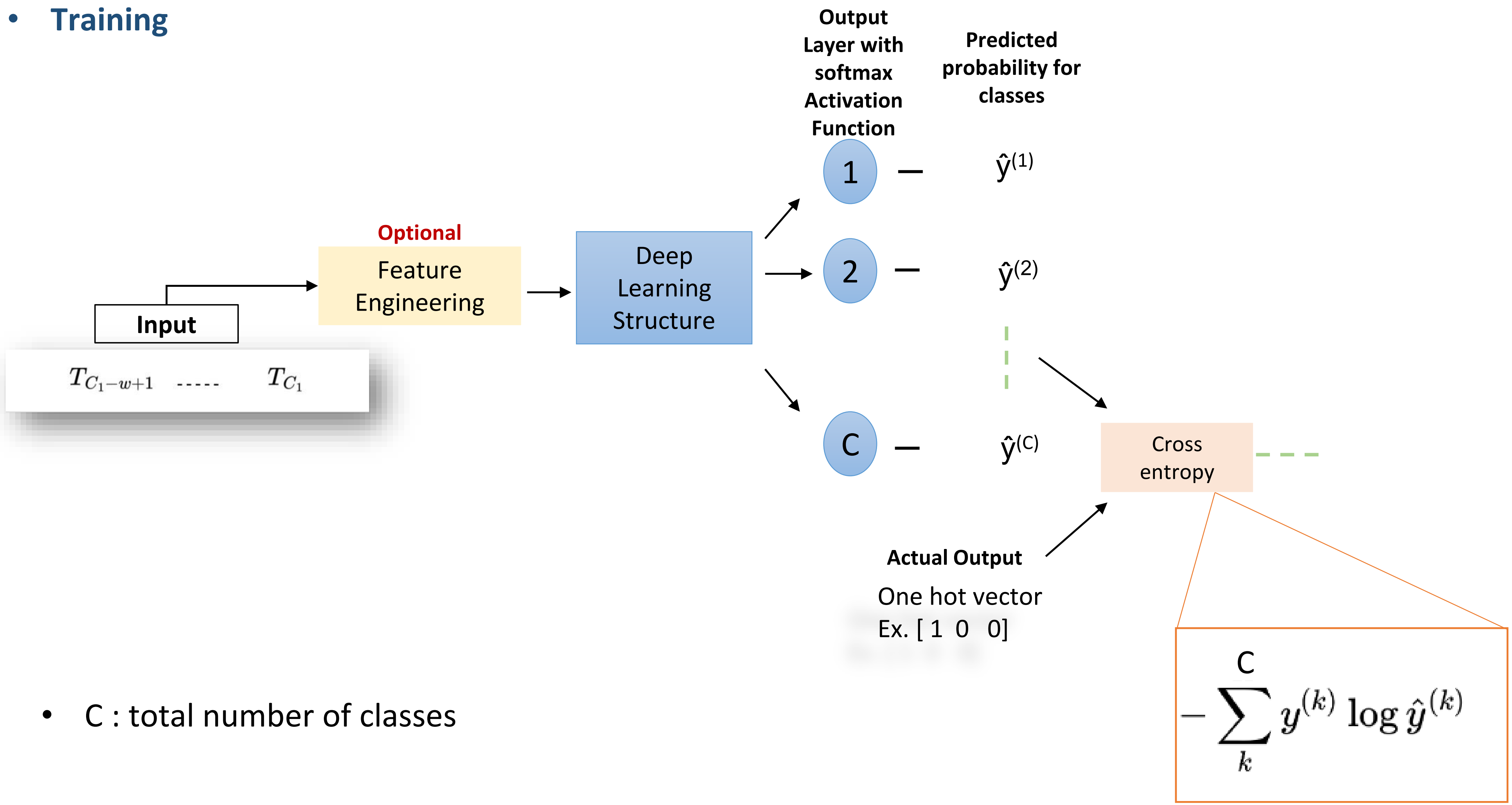
- **Pre-process Dataset**

	Input			Output
	T_1	-----	T_w	[1 0 0]
Event/Class 1 : T_1 T_2 T_3 --- $T_{c_1} \rightarrow$	T_2	-----	T_{w+1}	[1 0 0]
	T_{C_1-w+1}	-----	T_{C_1}	[1 0 0]
	T_1	-----	T_w	[0 1 0]
Event/Class 2 : T_1 T_2 T_3 --- $T_{c_2} \rightarrow$	T_2	-----	T_{w+1}	[0 1 0]
	T_{C_2-w+1}	-----	T_{C_2}	[0 1 0]
	T_1	-----	T_w	[0 0 1]
Event/Class 3 : T_1 T_2 T_3 --- $T_{c_3} \rightarrow$	T_2	-----	T_{w+1}	[0 0 1]
	T_{C_3-w+1}	-----	T_{C_3}	[0 0 1]

Sample

A complete pipeline

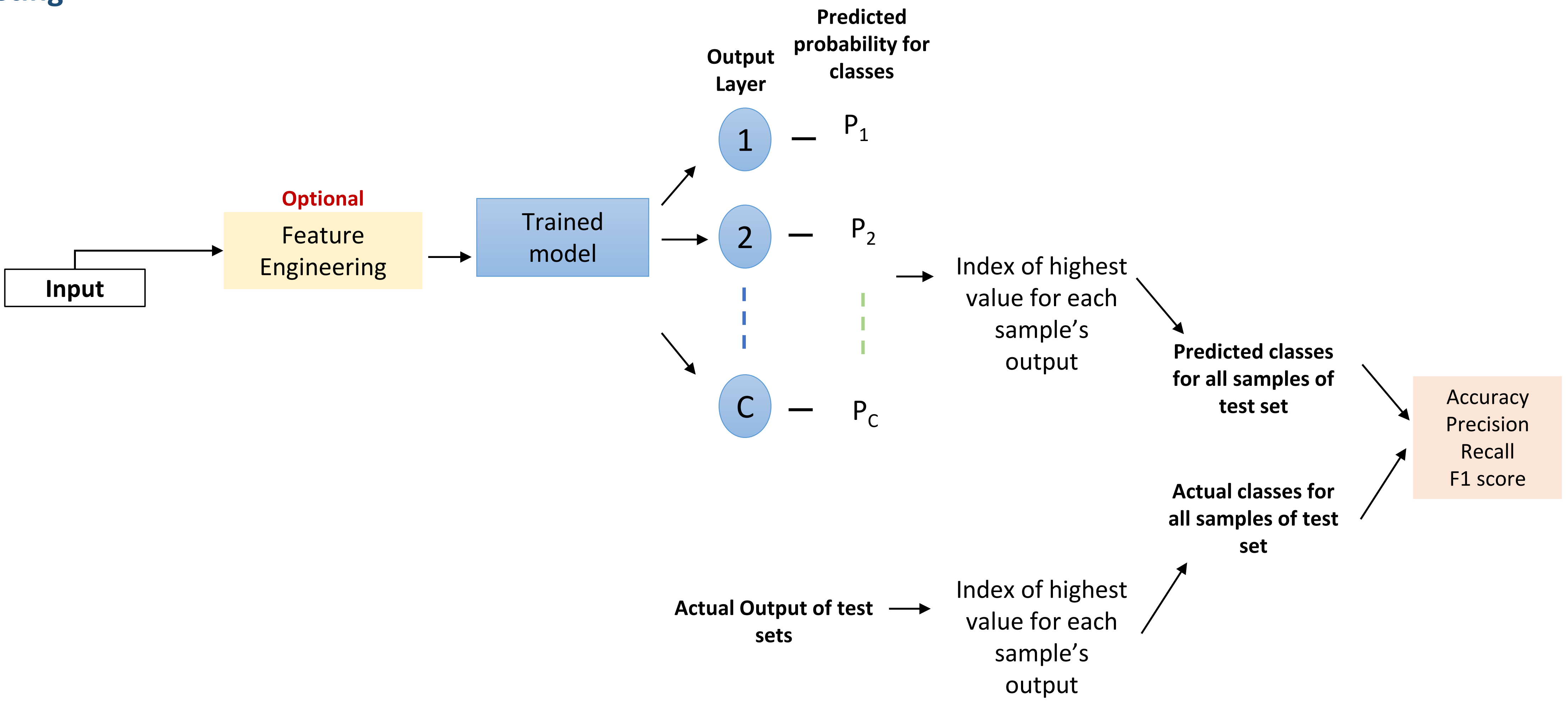
- **Training**



- C : total number of classes

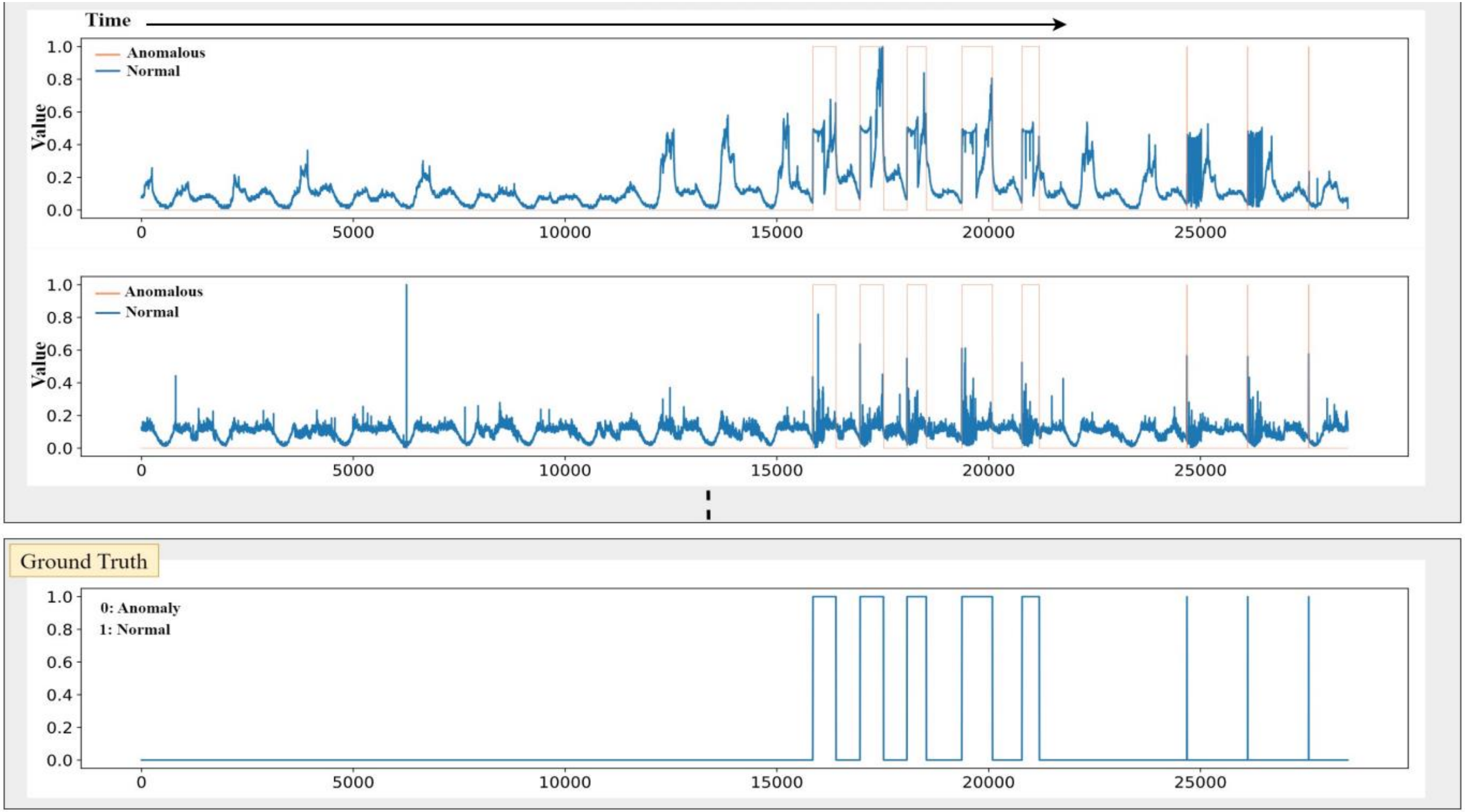
A complete pipeline

- **Testing**

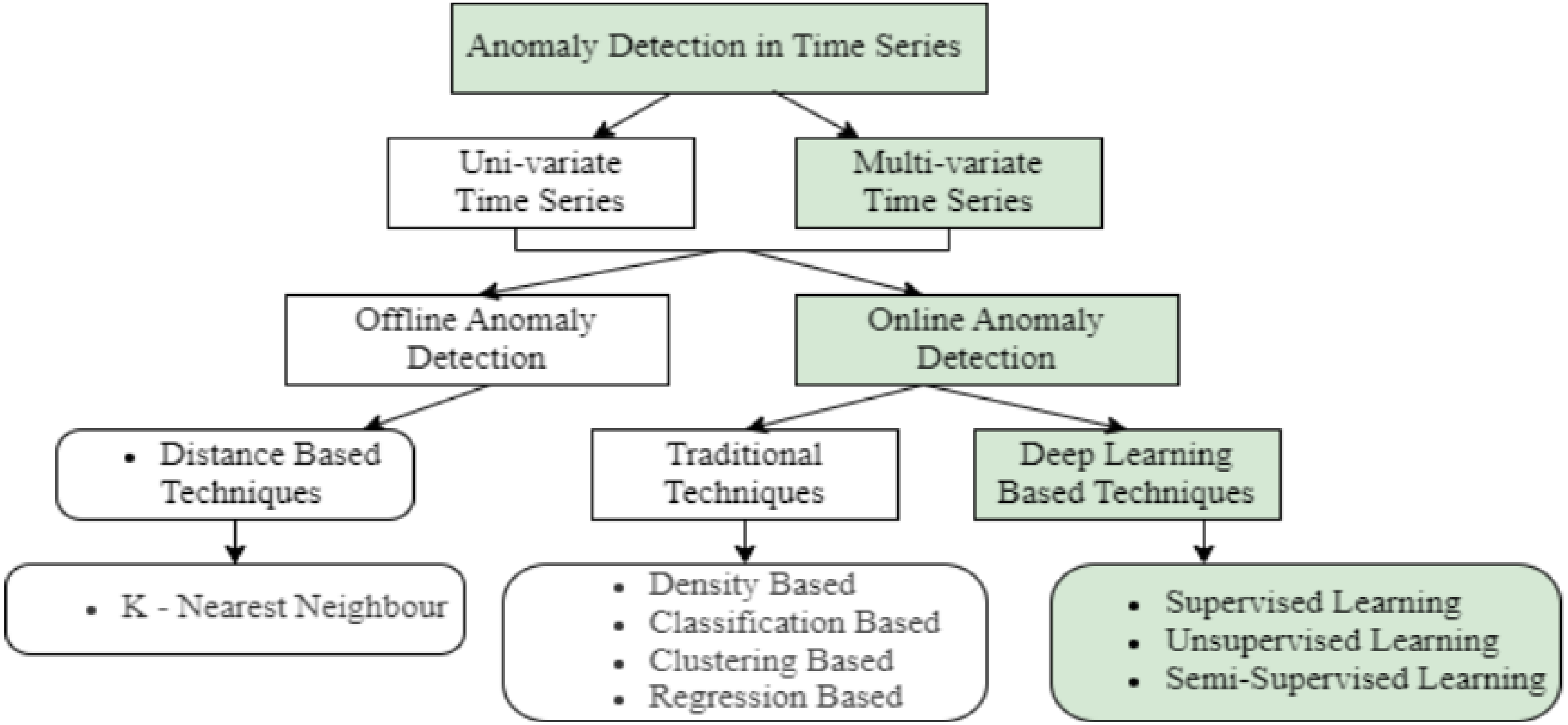


Anomaly detection in multivariate time series

- **Anomaly in a multivariate time series**



Anomaly detection in multivariate time series



Anomaly detection in multivariate time series

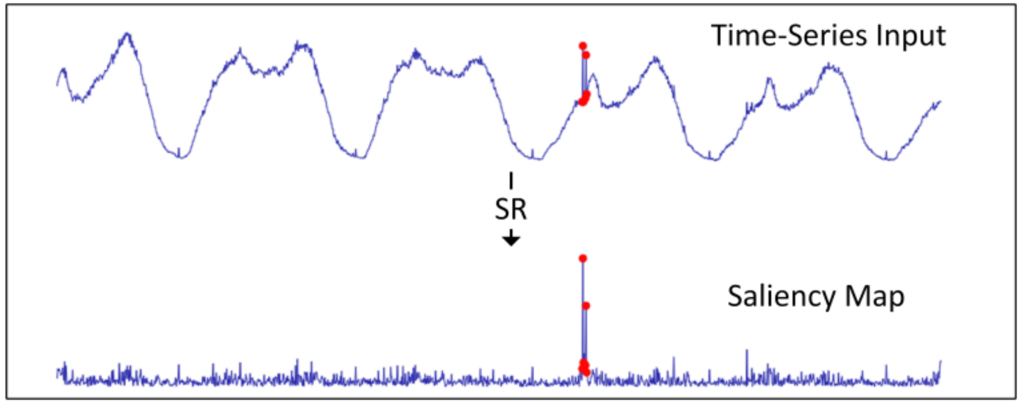
- Supervised learning methods
  - Solve problem using classification approach
  - Require labelling
  - Future anomalies may not be similar to the labelled anomalies
  - Problem of class imbalance

- Unsupervised learning methods
  - Solve problem by learning the property of the data
  - Poor performance in noisy data
  - Assume that the frequency of occurring abnormal instances is significantly less in the training dataset

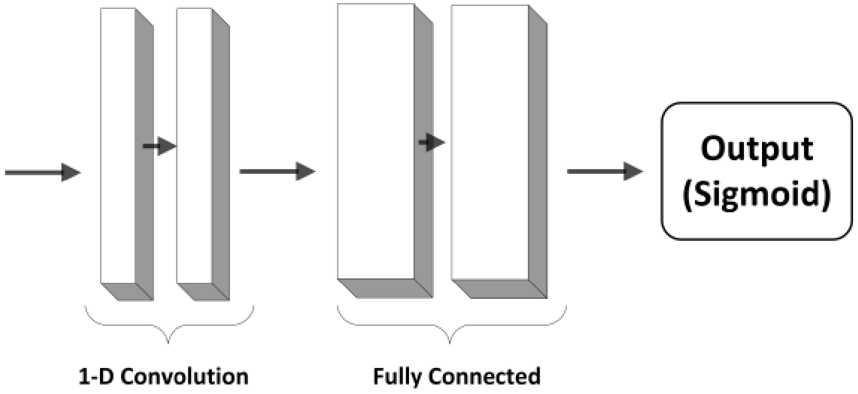
- Semi-supervised learning methods
  - Reconstruction based approach
  - Prediction based approach

Unsupervised learning methods

- Identify the anomaly from training set using traditional methods

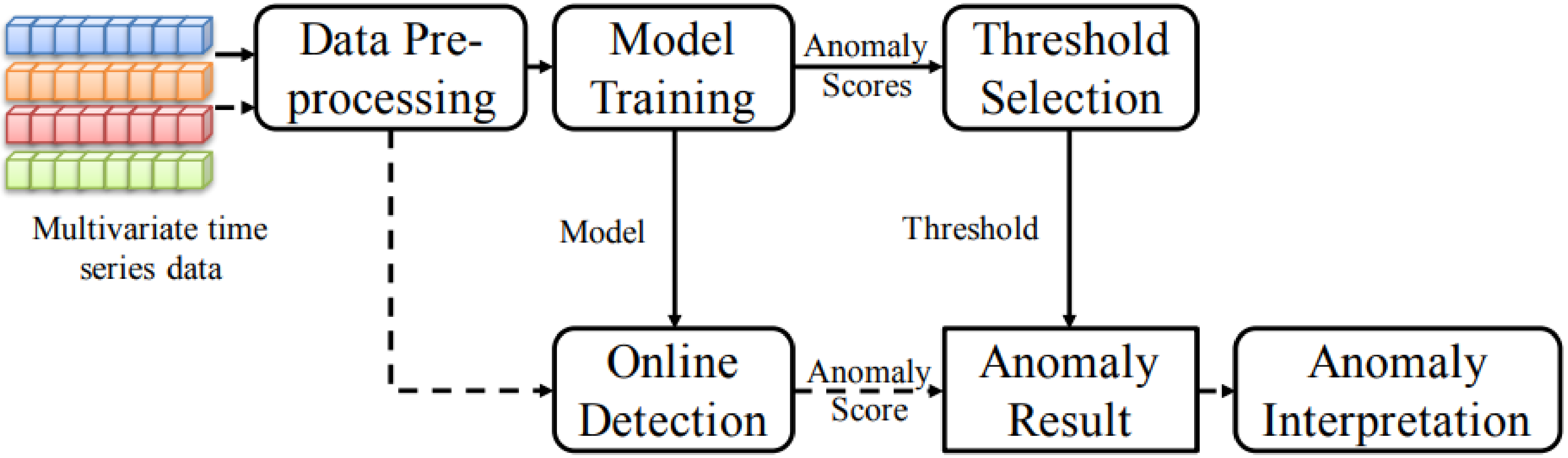


- Train a model

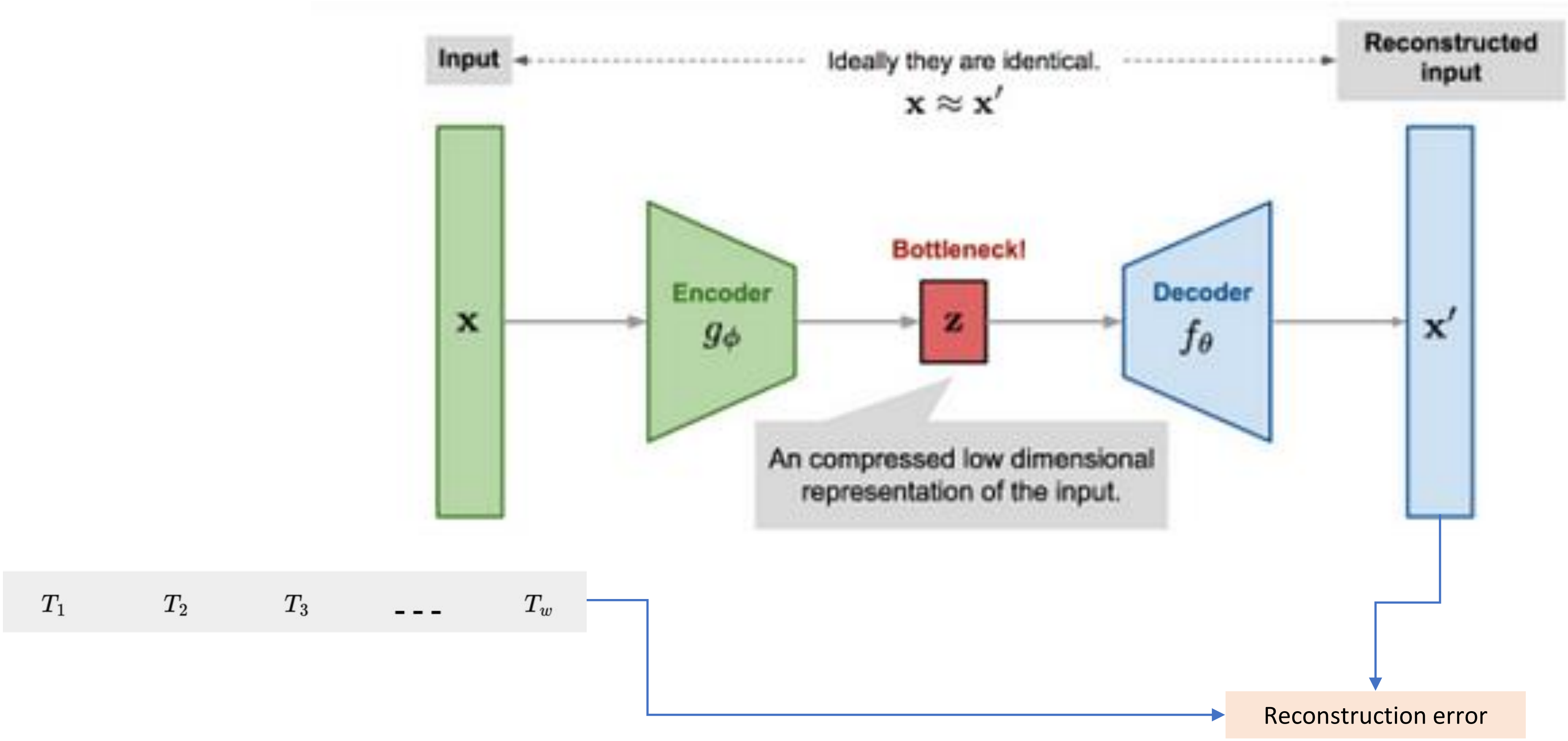


-Time-Series Anomaly Detection Service at Microsoft[2019]

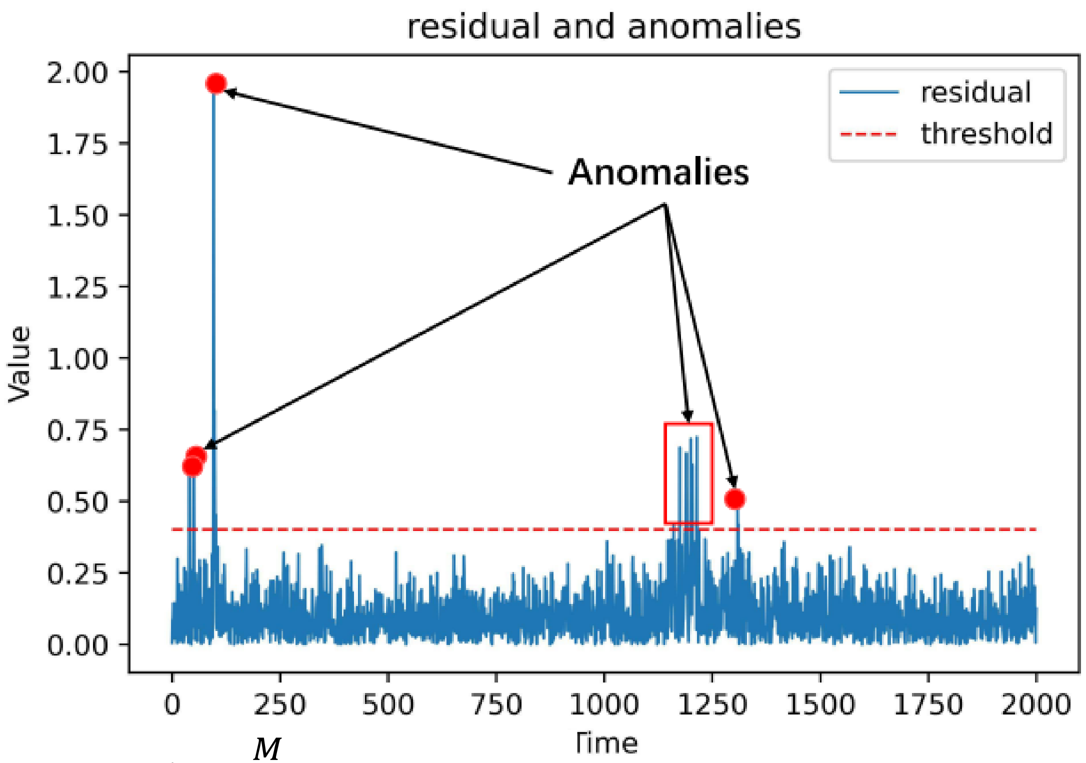
Semi-supervised learning methods



Reconstruction based approach



Reconstruction/ Prediction error



- Value at a timestamp index t = $\frac{1}{M} * \sum_{i=1}^{M} \left(\hat{X}_t^i - X_t^i\right)^2$

➢ Thresholding
  - Static thresholding
  - Dynamic thresholding

Feature engineering on input sample

- Dimensionality reduction
  - Principle component analysis
  - Singular value decomposition

- Feature extraction
  - Time domain features
  - Frequency domain features

- Noise removal methods
  - Moving average
  - Exponential smoothing
  - Fourier transform based methods

- Data augmentation
  - Jittering
  - Scaling

- Time series to image conversion
  - Recurrence plot
  - Spectrogram images