Time Series

권지혜 윤빈나 윤재경
전혜민 정재원 최성웅

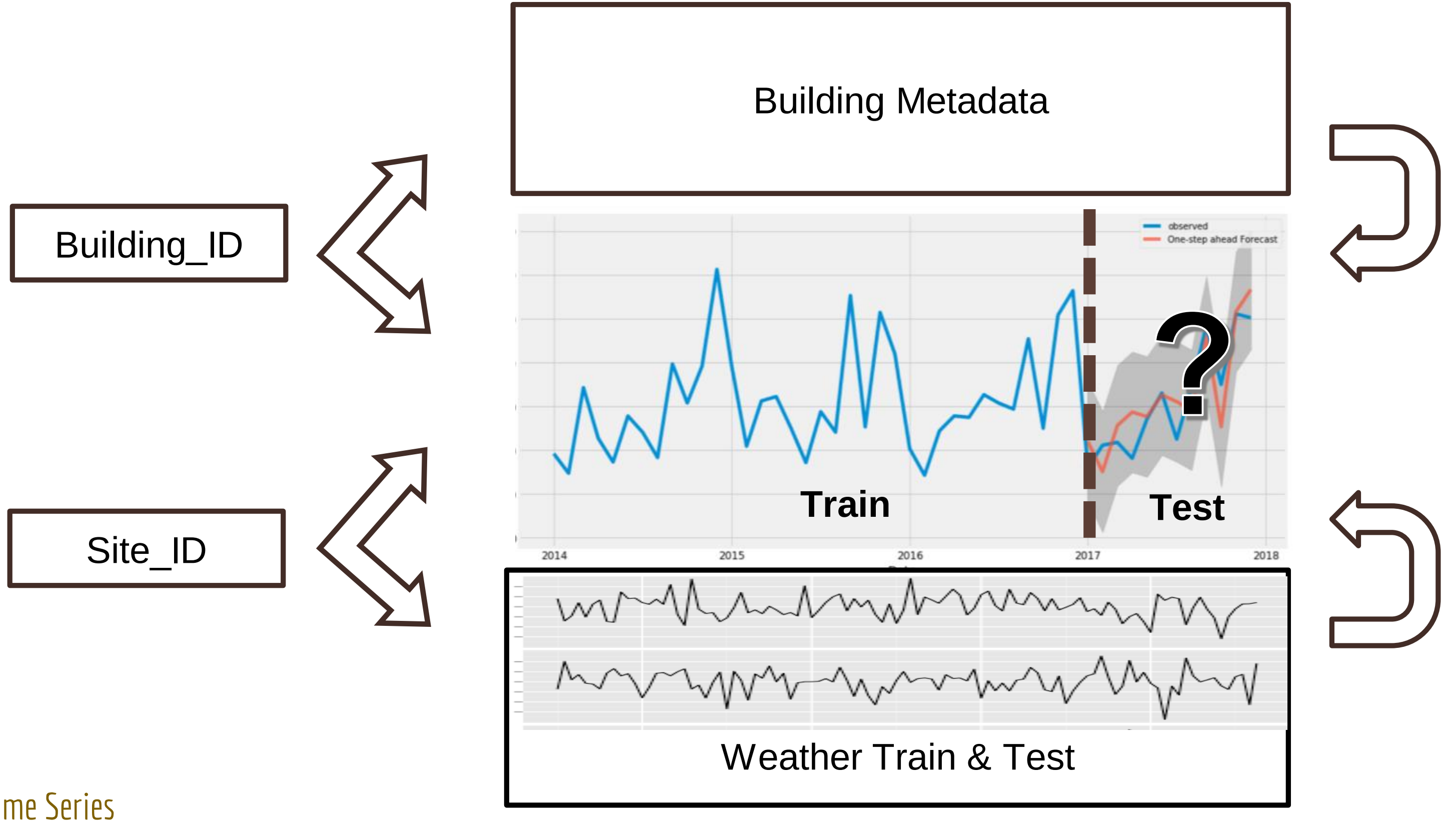
KU-BIG

INDEX

Topic Description

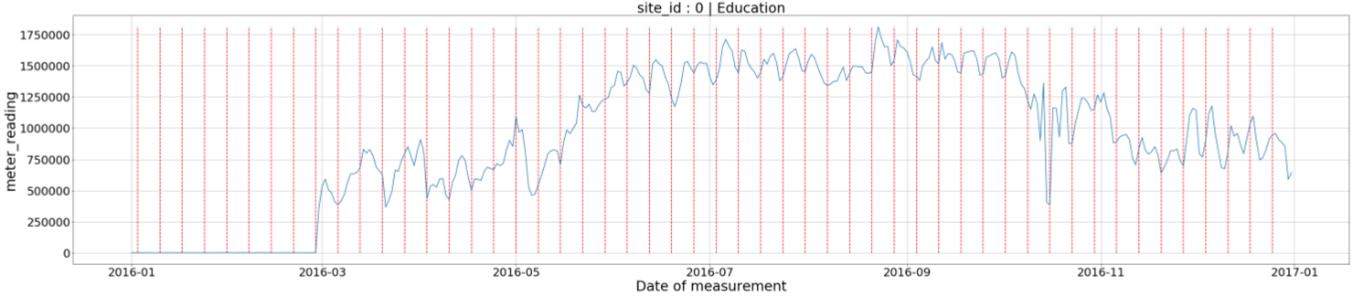
https://www.kaggle.com/c/ashrae-energy-prediction/overview

Data Description

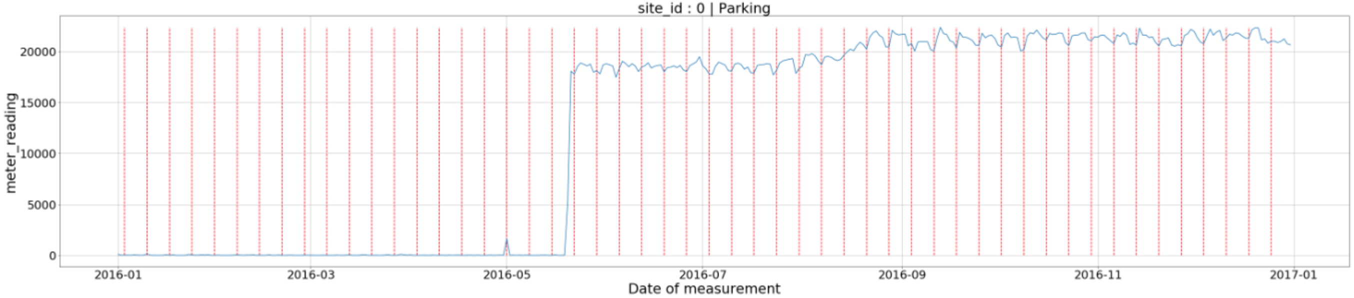


Data Description - Visualization (Daily Sum by Site&Category)

Education

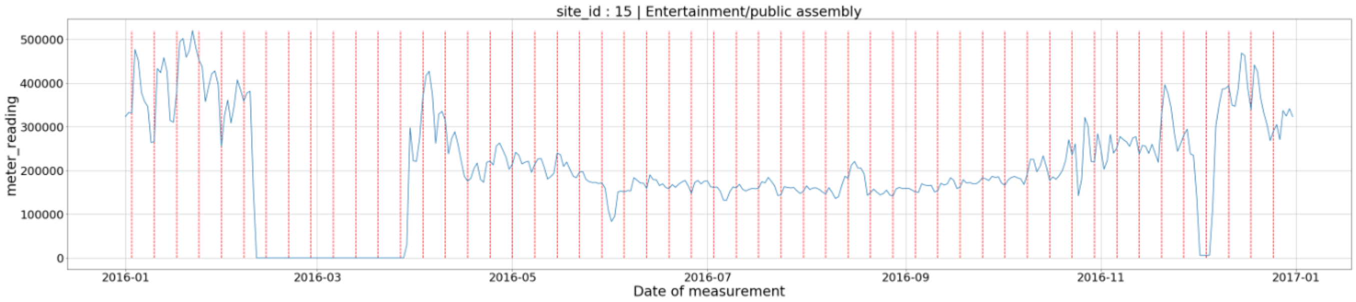


Parking

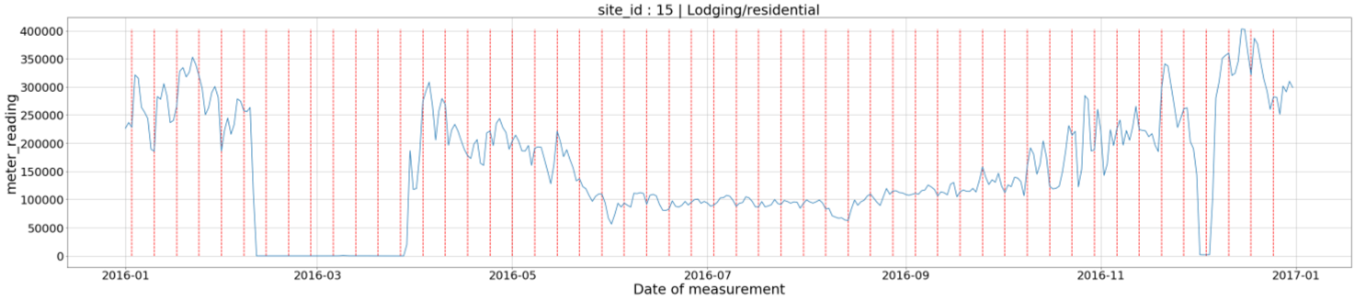


Data Description - Visualization (Daily Sum by Site&Category)

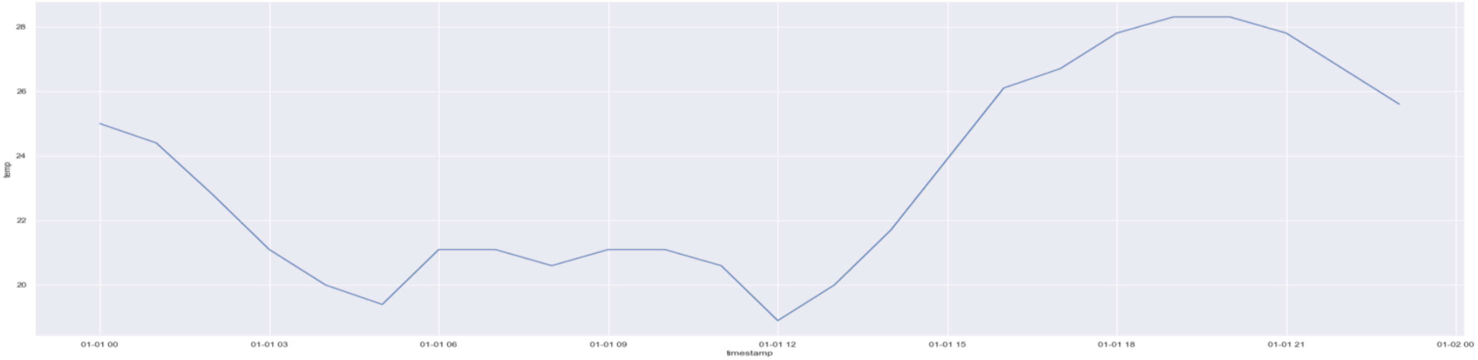
Entertainment/public assembly

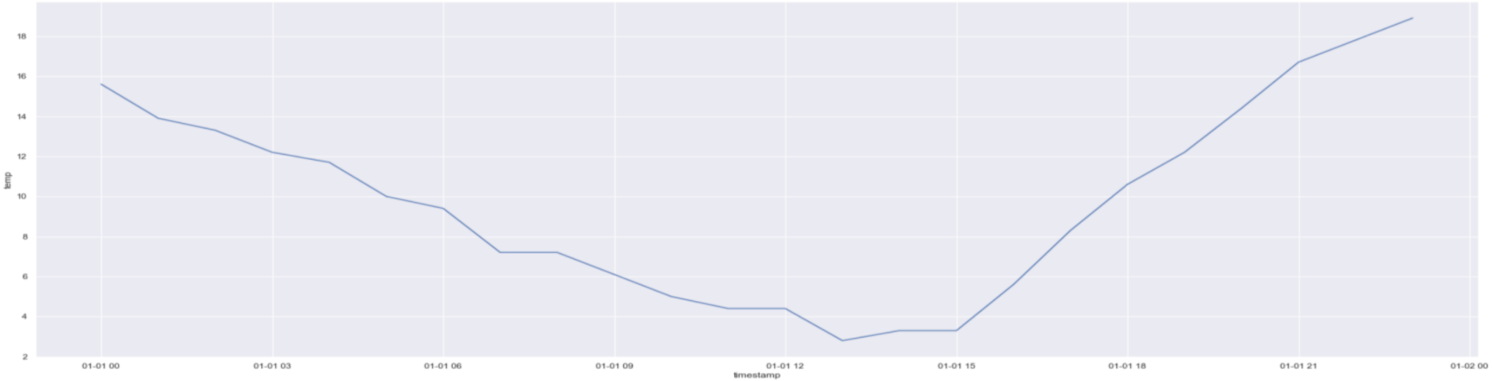


Lodging/residential

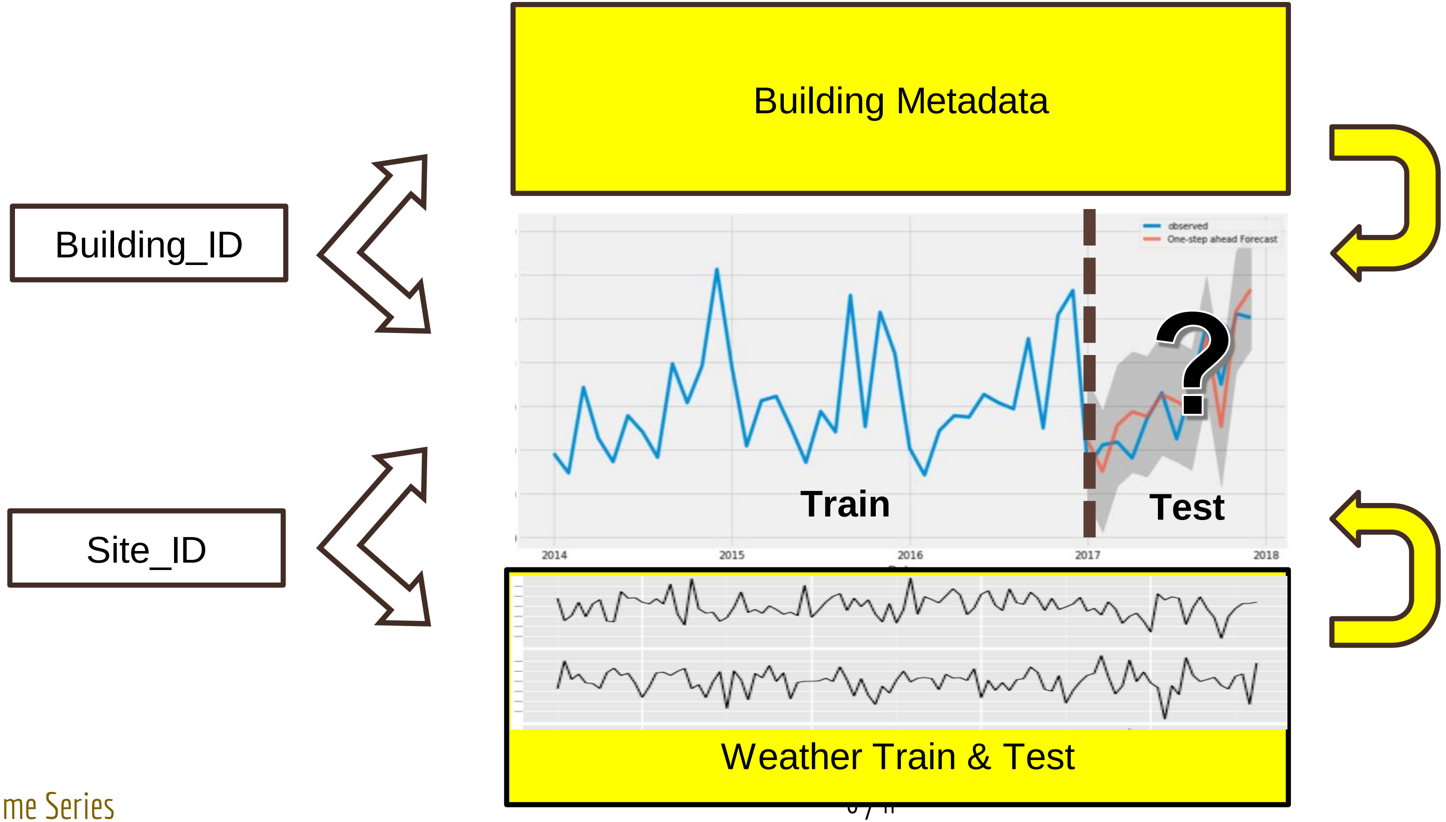


Data Description - Visualization(Temperature in 2016.01.01)





Issue> Exogenous Variables(Weather, Building Info)



Model Introduction

1. Statistical Model – ARIMA & Dynamic Regression

(1) 후방이동 기호

$$By_t = y_{t-1} .$$

$$y'_t = y_t - y_{t-1} = y_t - By_t = (1-B)y_t .$$

(2) AR(p) ; Auto-Regressive Model

$$y_t = c + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \cdots + \phi_p y_{t-p} + \varepsilon_t,$$

(3) MA(q) Model ; Moving-Average Model

$$y_t = c + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \cdots + \theta_q \varepsilon_{t-q},$$

(4) ARIMA(p,d,q) Model

$$y'_t = c + \phi_1 y'_{t-1} + \cdots + \phi_p y'_{t-p} + \theta_1 \varepsilon_{t-1} + \cdots + \theta_q \varepsilon_{t-q} + \varepsilon_t,$$

$$\underset{\substack{\uparrow \\ \text{AR}(p)}}{(1-\phi_1 B - \cdots - \phi_p B^p)} \quad \underset{\substack{\uparrow \\ d \text{ differences}}}{(1-B)^d y_t} \quad = \quad \underset{\substack{\uparrow \\ \text{MA}(q)}}{c + (1 + \theta_1 B + \cdots + \theta_q B^q)\varepsilon_t}$$

출처 https://otexts.com/fppkr/dynamic.html

Model Introduction

1. Statistical Model – ARIMA & **Dynamic Regression**

<Dynamic Regression. 동적 회귀>

- 이전 관측값으로부터 얻은 정보 이외의 정보는 들어가지 않는 ARIMA 모델을 확장시켜, 시계열 자료에 영향을 줄 수 있는 외부적 요인(시장 shock, 휴일 효과, 날씨)을 반영할 수 있는 모델

- 외생변수로 회귀식을 적합한 후, 그 잔차에 ARIMA 모델을 적용시키는 방법.

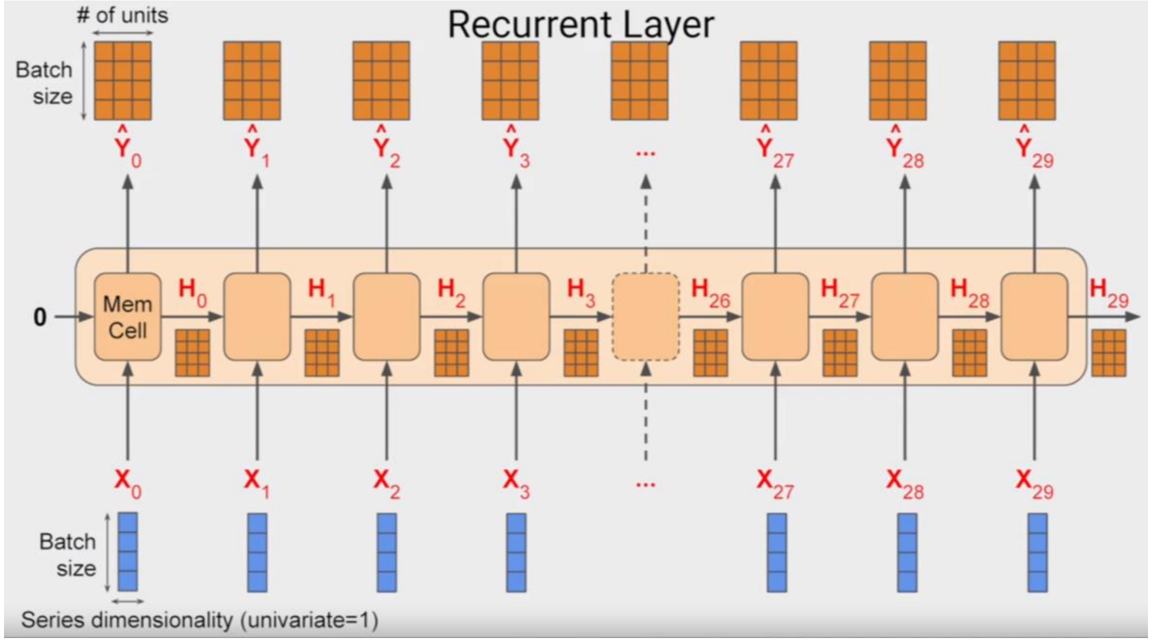
$$y_t = \beta_0 + \beta_1 x_{1,t} + \cdots + \beta_k x_{k,t} + e_t,$$

$$y_t = \beta_0 + \beta_1 x_{1,t} + \cdots + \beta_k x_{k,t} + n_t,$$
$$(1 - \phi_1 B)(1 - B)n_t = (1 + \theta_1 B)e_t,$$

출처 https://otexts.com/fppkr/dynamic.html

Model Introduction

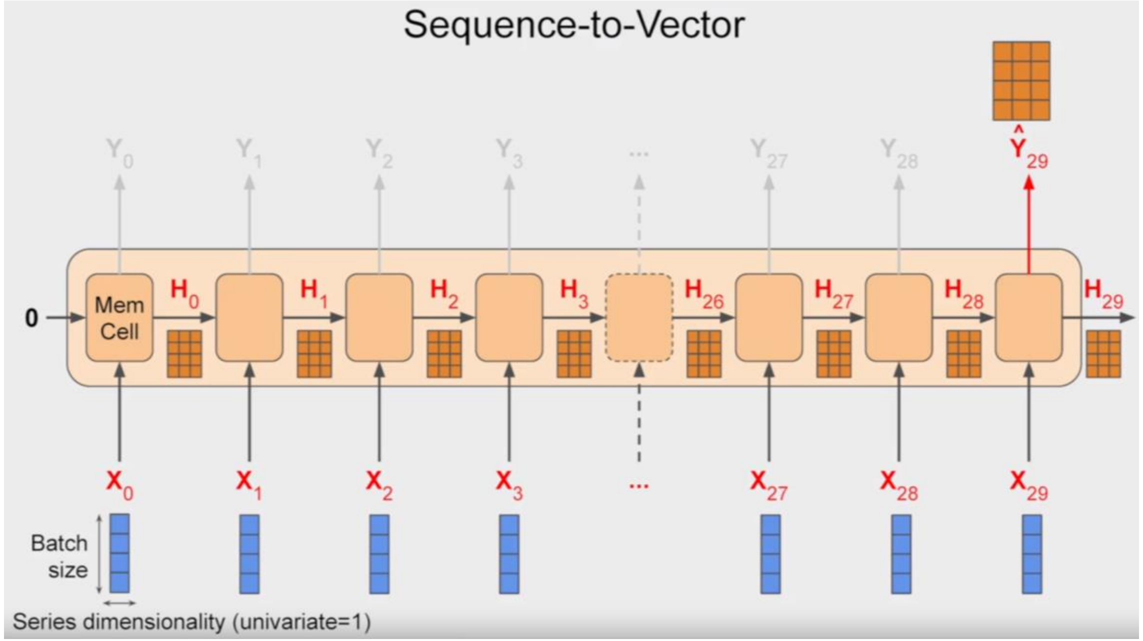
2. RNN Based Model & LSTM



출처 https://www.coursera.org/learn/tensorflow-sequences-time-series-and-prediction/home/welcome

Model Introduction

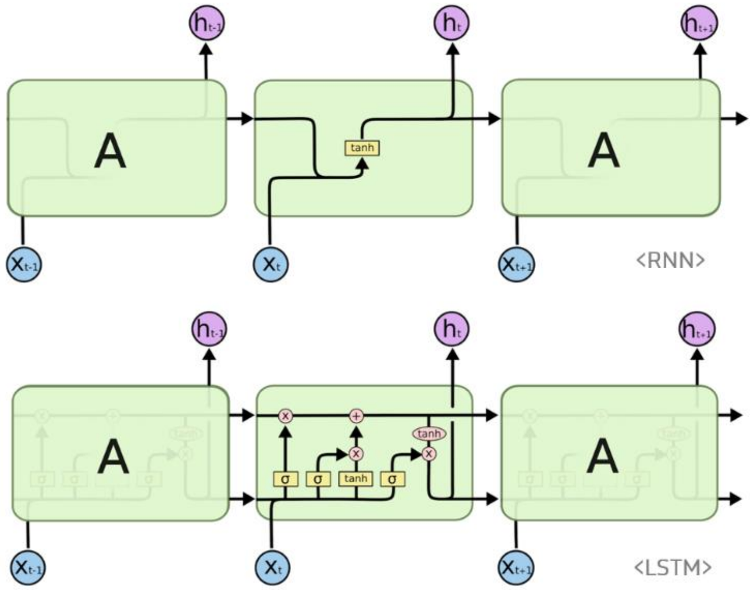
2. RNN Based Model & LSTM

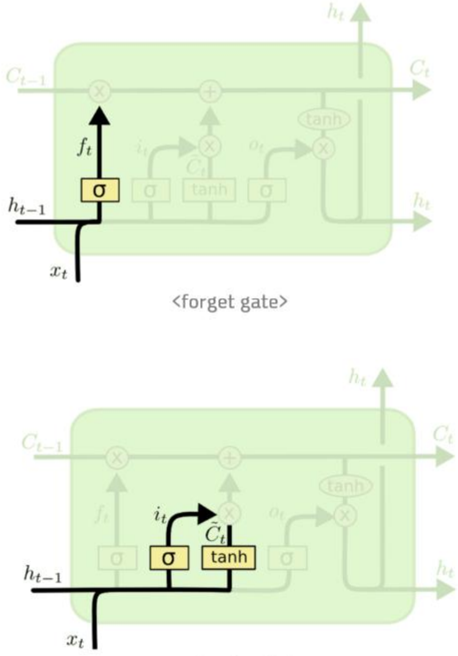


출처 https://www.coursera.org/learn/tensorflow-sequences-time-series-and-prediction/home/welcome

Model Introduction

2. RNN Based Model & **LSTM**





출처 https://ratsgo.github.io/natural%20language%20processing/2017/03/09/rnnlstm/

What we've done

1. ARIMA & Dynamic Regression

```
model1.summary()
```

Statespace Model Results

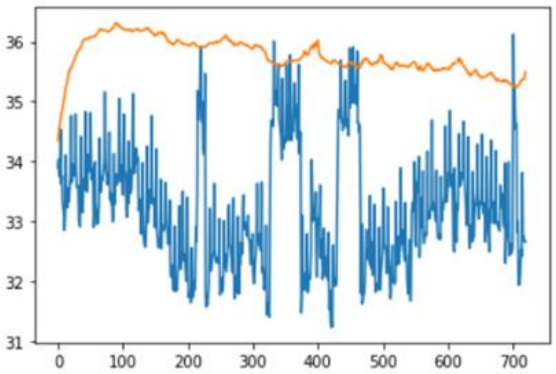
Dep. Variable:	y	**No. Observations:**	8064
Model:	SARIMAX(2, 1, 3)	**Log Likelihood**	-9287.607
Date:	Thu, 14 Nov 2019	**AIC**	18597.213
Time:	13:09:46	**BIC**	18674.159
Sample:	0	**HQIC**	18623.540
	- 8064		
Covariance Type:	opg		

	coef	**std err**	**z**	**P>|z|**	**[0.025**	**0.975]**
intercept	-0.0001	0.000	-0.345	0.730	-0.001	0.001
x1	0.0159	0.010	1.606	0.108	-0.004	0.035
x2	0.0024	0.011	0.228	0.820	-0.018	0.023
x3	0.0001	0.002	0.077	0.939	-0.003	0.004
x4	-0.0039	0.008	-0.502	0.616	-0.019	0.011
ar.L1	0.1013	0.197	0.515	0.606	-0.284	0.487
ar.L2	0.8132	0.184	4.419	0.000	0.452	1.174
ma.L1	-0.2545	0.197	-1.290	0.197	-0.641	0.132
ma.L2	-0.8281	0.213	-3.892	0.000	-1.245	-0.411
ma.L3	0.1150	0.023	4.967	0.000	0.070	0.160
sigma2	0.5862	0.001	561.157	0.000	0.584	0.588

Ljung-Box (Q):	306.10	**Jarque-Bera (JB):**	96628979.67
Prob(Q):	0.00	**Prob(JB):**	0.00
Heteroskedasticity (H):	0.65	**Skew:**	0.12
Prob(H) (two-sided):	0.00	**Kurtosis:**	539.30

```
plt.plot(mer1_meter[-720:].reset_index(drop = True))
plt.plot(model1.predict(n_periods = 720, exogenous=mer1_weather.iloc[-720:,0:]))
```

```
[<matplotlib.lines.Line2D at 0x7f7bc80f5eb8>]
```



What we've done

1. ARIMA & Dynamic Regression

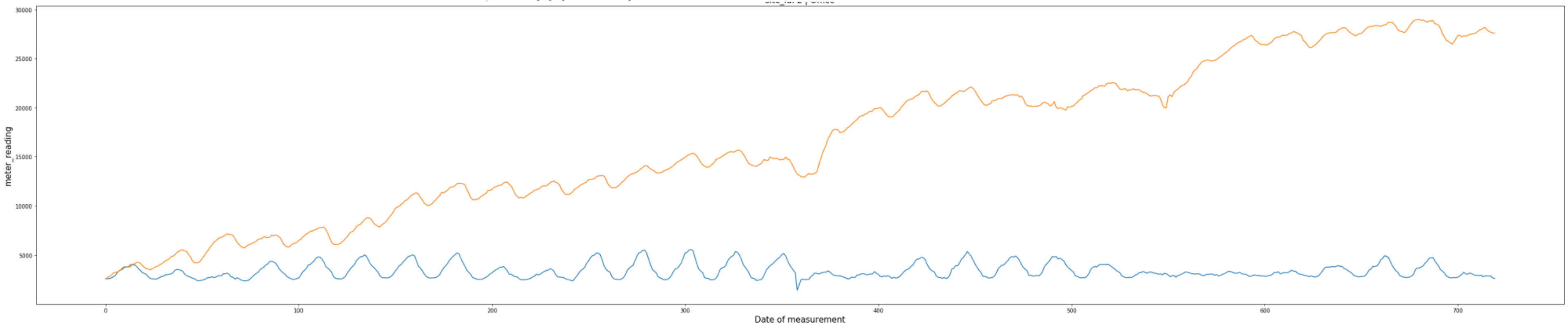
```
[69] model_1.summary()
```

Statespace Model Results

Dep. Variable:	y	**No. Observations:**	8064
Model:	SARIMAX(2, 1, 2)	**Log Likelihood**	-59258.456
Date:	Fri, 15 Nov 2019	**AIC**	118538.913
Time:	07:37:02	**BIC**	118615.858
Sample:	0	**HQIC**	118565.239
	- 8064		

	coef	std err	z	P>|z|	[0.025	0.975]
intercept	2.4199	0.379	6.381	0.000	1.677	3.163
x1	-45.6527	4.656	-9.805	0.000	-54.778	-36.527
x2	-49.9881	4.453	-11.226	0.000	-58.715	-41.261
x3	8.2310	0.439	18.735	0.000	7.370	9.092
x4	225.6792	9.320	24.215	0.000	207.413	243.946
x5	14.8525	2.776	5.350	0.000	9.412	20.293
ar.L1	1.9289	0.001	3676.473	0.000	1.928	1.930
ar.L2	-0.9969	0.000	-2041.748	0.000	-0.998	-0.996
ma.L1	-1.9178	0.002	-1240.059	0.000	-1.921	-1.915
ma.L2	0.9860	0.001	667.366	0.000	0.983	0.989
sigma2	1.745e+05	1118.327	156.076	0.000	1.72e+05	1.77e+05

Ljung-Box (Q):	2741.24	**Jarque-Bera (JB):**	163152.44
Prob(Q):	0.00	**Prob(JB):**	0.00
Heteroskedasticity (H):	1.16	**Skew:**	-0.04
Prob(H) (two-sided):	0.00	**Kurtosis:**	25.04

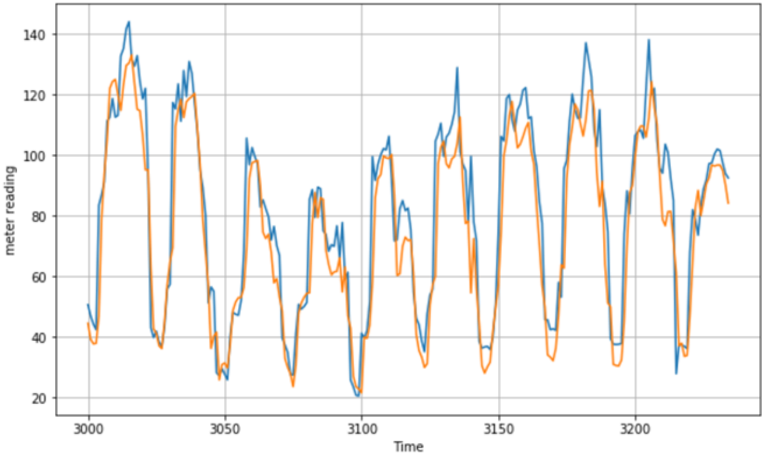


What we've done

2. RNN Based model & LSTM

In [72]:
```
rnn_forecast = model_forecast(model, series[..., np.newaxis], window_size)
rnn_forecast = rnn_forecast[split_time - window_size:-1, -1, 0]
```

In [73]:
```
plt.figure(figsize=(10, 6))
plot_series(time_valid, x_valid)
plot_series(time_valid, rnn_forecast)
```



- Site02 – sum(Office) hourly forecasting
- Training Set 200, Validation Set 135
- Window Size = 30   Batch Size = 32   Epochs = 30

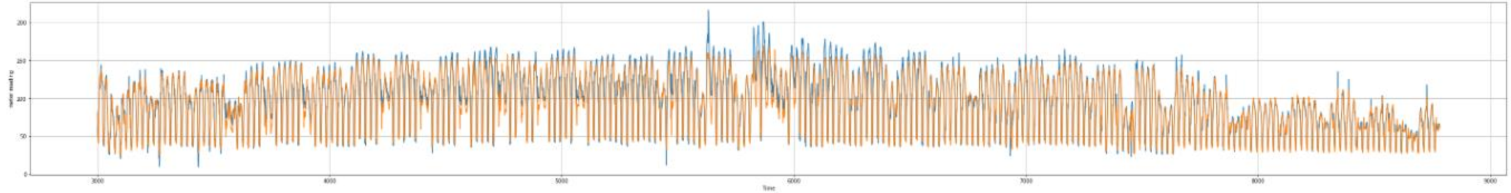
```
loss: 5.6806 - mae: 6.1491
```

What we've done

2. RNN Based model & LSTM

- Site02 – sum(Office) hourly forecasting
- Training Set 3000, Validation Set 나머지
- Window Size = 60 Batch Size = 100 Epochs = 30

In [14]:
```
plt.figure(figsize=(50, 6))
plot_series(time_valid, x_valid)
plot_series(time_valid, rnn_forecast)
```



In [15]:
```
tf.keras.metrics.mean_absolute_error(x_valid, rnn_forecast).numpy()
```

Out[15]: 9.561941

What to do next

1. **ARIMA -> SARIMA** (auto arima 함수 X)

2. **RNN&LSTM 모델에 외생변수 적용하기**
   좋은 방법이 있을까요?

3. **잔차 검정**
   ACF/PACF/Normality 검정
   포트만토 검정 – 잔차의 자기상관
   Box Ljung 검정 – 잔차의 독립성

4. **너무 많은 빌딩 수를 어떻게 감당해낼 것인가**


Do you
have any
question?
Thank you
for your attention.