

Explainable Hybrid LSTM-resCNN for Forecasting Multiple Time Series Data

Hyojung Choi, Il-Youp Kwak

Department of Applied Statistics, Chung-Ang University, Seoul, Republic of Korea



Background

오늘날 딥러닝(Deep Learning)은 여러 분야에서 적용되고 있으며, 활용범위 또한 다양해지고 있다. 하지만 딥러닝 모델은 왜 그런 결정을 했는지에 대한 근거와 타당성을 파악하기 어려워 소위 블랙박스(Black Box)라 부르기도 한다.

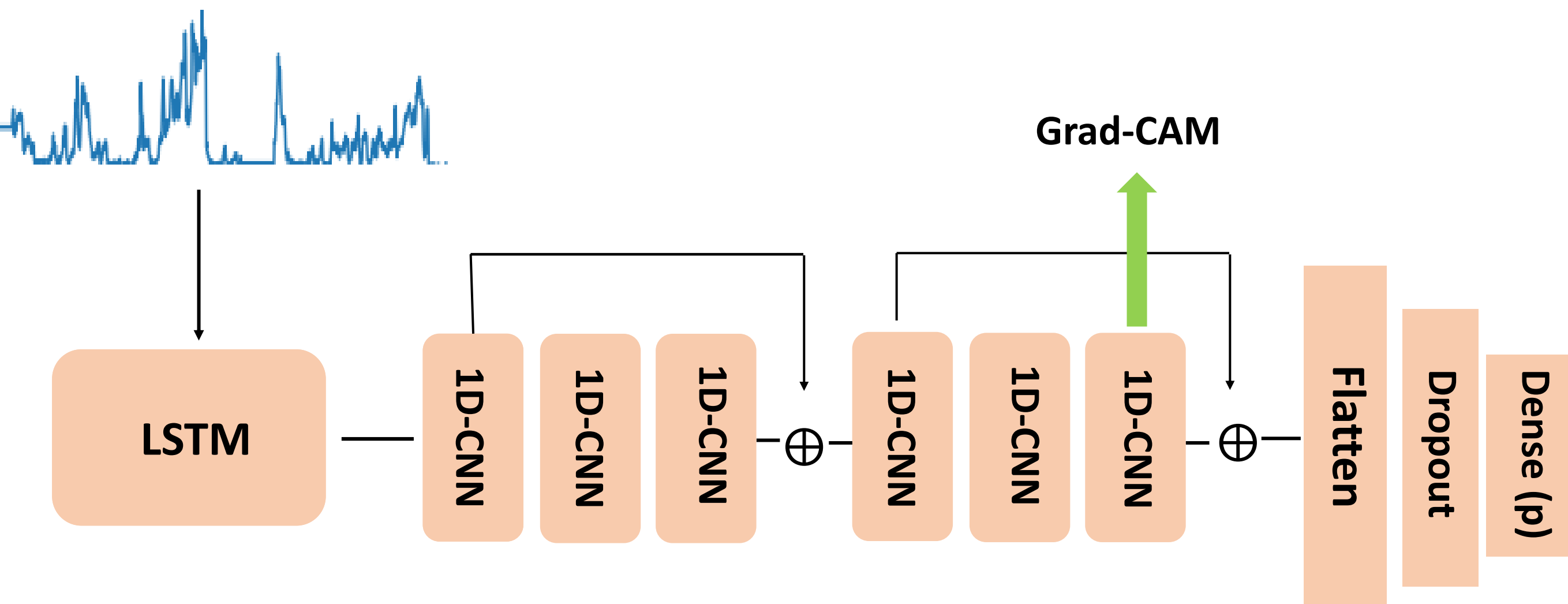
본 연구에서는 딥러닝을 이용한 시계열 예측 모델에 설명력을 추가하였다. 데이터는 미세먼지(PM 2.5), 서울 자전거 대여량 총 2가지를 이용해 예측했으며, 기존 시계열 예측 문제에서 많이 쓰였던 LSTM과 1D-CNN의 하이브리드(hybrid) 모델에 잔차 연결(Residual-Connection)을 추가하면서 예측성능을 향상시켰다. 또한 CNN(Convolutional Neural Network) 기반 모델에서 클래스(class) 결정을 시각적으로 설명할 수 있는 그라드캠(Grad-CAM)의 원리를 활용하여 예측에 가장 큰 영향을 준 변수와 시점을 확인하였다

Model Architecture

- RNN(Recurrent Neural Network)의 **기울기 소실**으로 인한 성능저하를 극복하기 위해 고안된 **LSTM**.
- **1D-CNN**은 자연어 처리와 시계열에서 사용되며 커널을 시간 방향으로 움직이면서 피쳐를 뽑음.
- **CNN-LSTM**은 단일 LSTM, 1D-CNN보다 좋은 성능을 보인다는 선행 연구가 많이 있음.

본 연구에서는 **LSTM-CNN** 구조에 **Residual Connection**을 추가하면서 기존의 CNN-LSTM 보다 좋은 예측력을 보였다. Residual Connection은 기울기 소실/ 발산에 의한 낮은 성능을 개선시키고, 유연성과 네트워크 수렴을 향상시킬 수 있는 방법으로 ResNet(2015)에서 처음 고안되었다. 아래의 수식처럼 단순히 input x를 더하면서 기존에 학습한 정보를 보존하고 새로 학습해야 할 정보량을 축소시켜 주는 역할을 한다.

$$Y = f(x) + x$$



Data preprocessing

- 스케일의 차이가 크면 잘 작동되지 않기 때문에 범위를 일정하게 맞춰주는 작업 필요
- (0,1) 사이로 데이터 정규화

$$\text{Normalization : } x = \frac{x - \min}{\max - \min}$$

Prediction Method

- **Many – to – One** : 이전 24시간 데이터로 향후 1(p=1)시간 예측
- **Many – to – Many** : 이전 168시간 (일주일) 데이터로 향후 24(p=24)시간 예측
- **Data Split** : Train:Val:Test = 8:1:1

Evaluation Metric

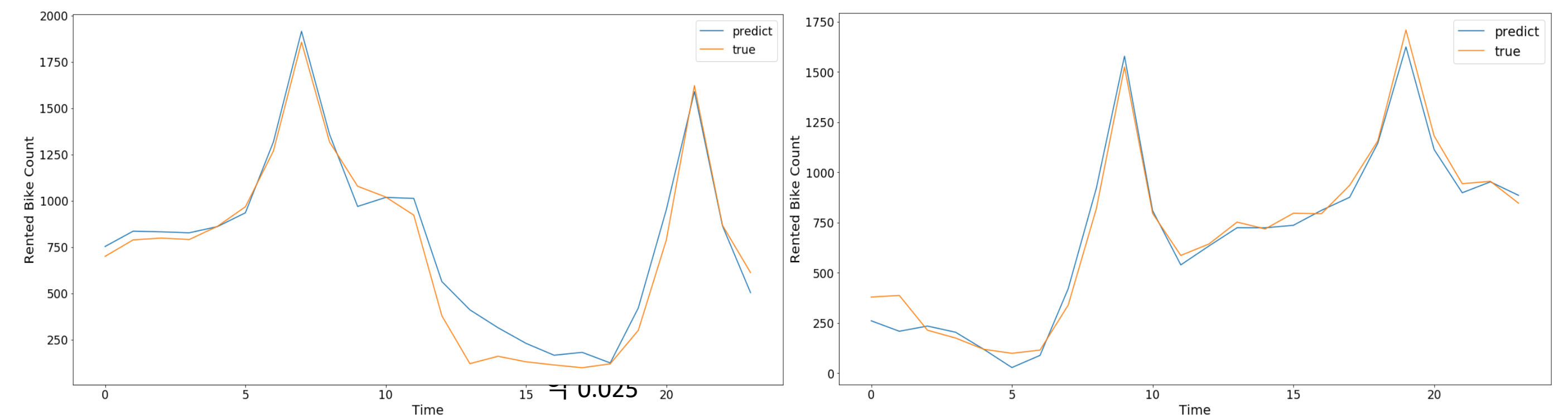
- 작을수록 좋은 예측력을 의미

$$\text{RMSE} = \sqrt{\frac{1}{N} \sum_{i=1}^N (y_i - \hat{y})^2}$$

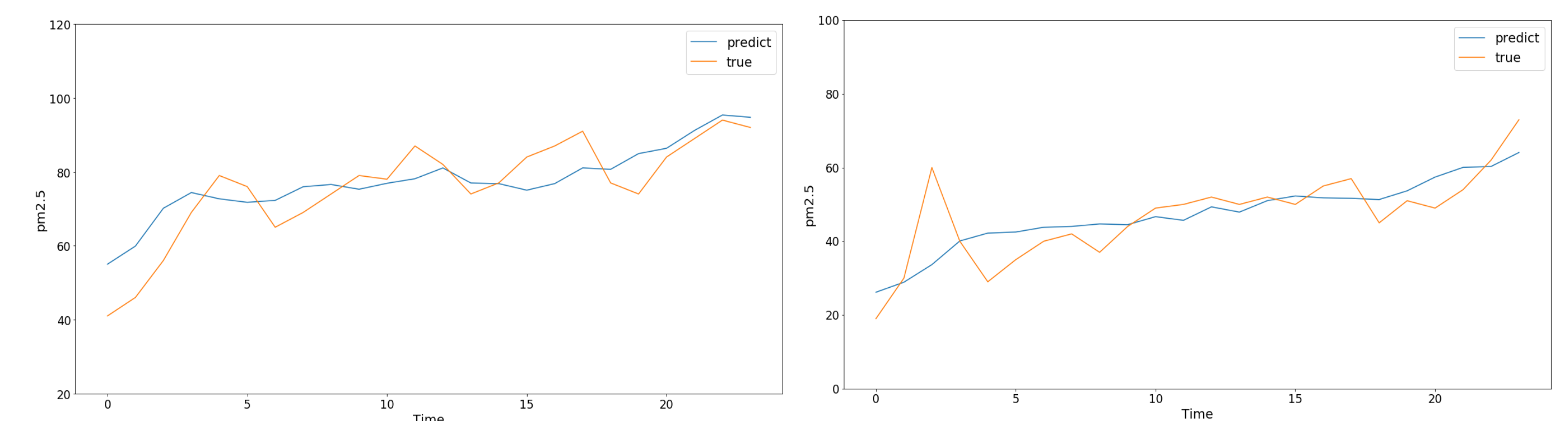
$$\text{MAE} = \frac{1}{N} \sum_{i=1}^N |y_i - \hat{y}|$$

<Table 1> Onestep-ahead performance

Model	pm2.5 (RMSE)	PM 2.5 (MAE)	Bike (RMSE)	Bike (MAE)	Parameters PM2.5 / Bike
1D-CNN	0.01665	0.01034	0.03083	0.02078	(65,321)/(77,477)
BiLSTM	0.01634	0.00970	0.04231	0.02559	(66,131)/(72,181)
LSTM-CNN	0.01676	0.01037	0.03192	0.02170	(62,337)/(70,017)
CNN-LSTM	0.01658	0.00972	0.03124	0.02144	(67,649) /(72,065)
LSTM-resCNN	0.01629	0.00961	0.02902	0.01964	(62,337) /(70,017)

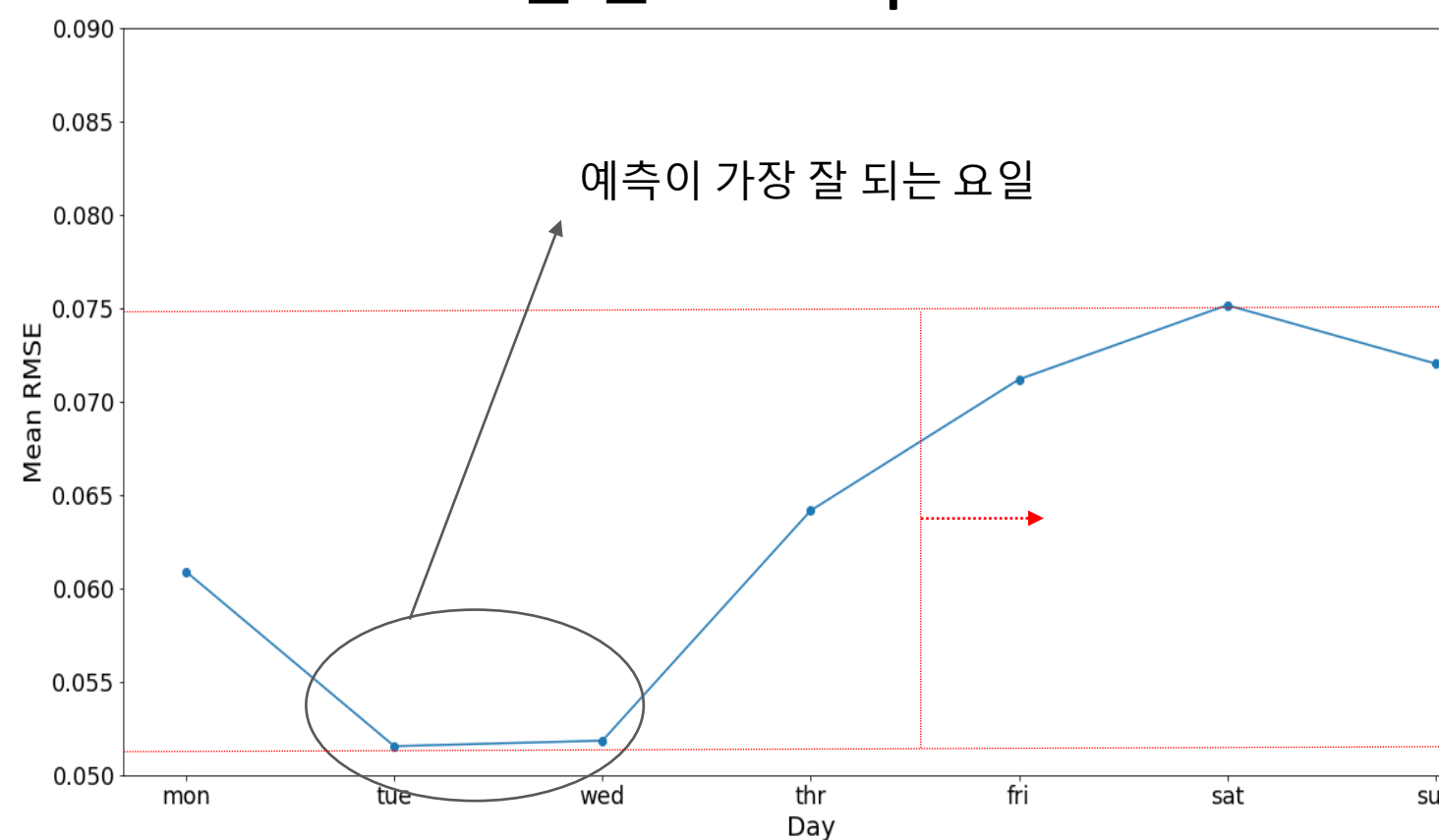


c) Seoul Bike sharing multistep-ahead (168 -> 24)

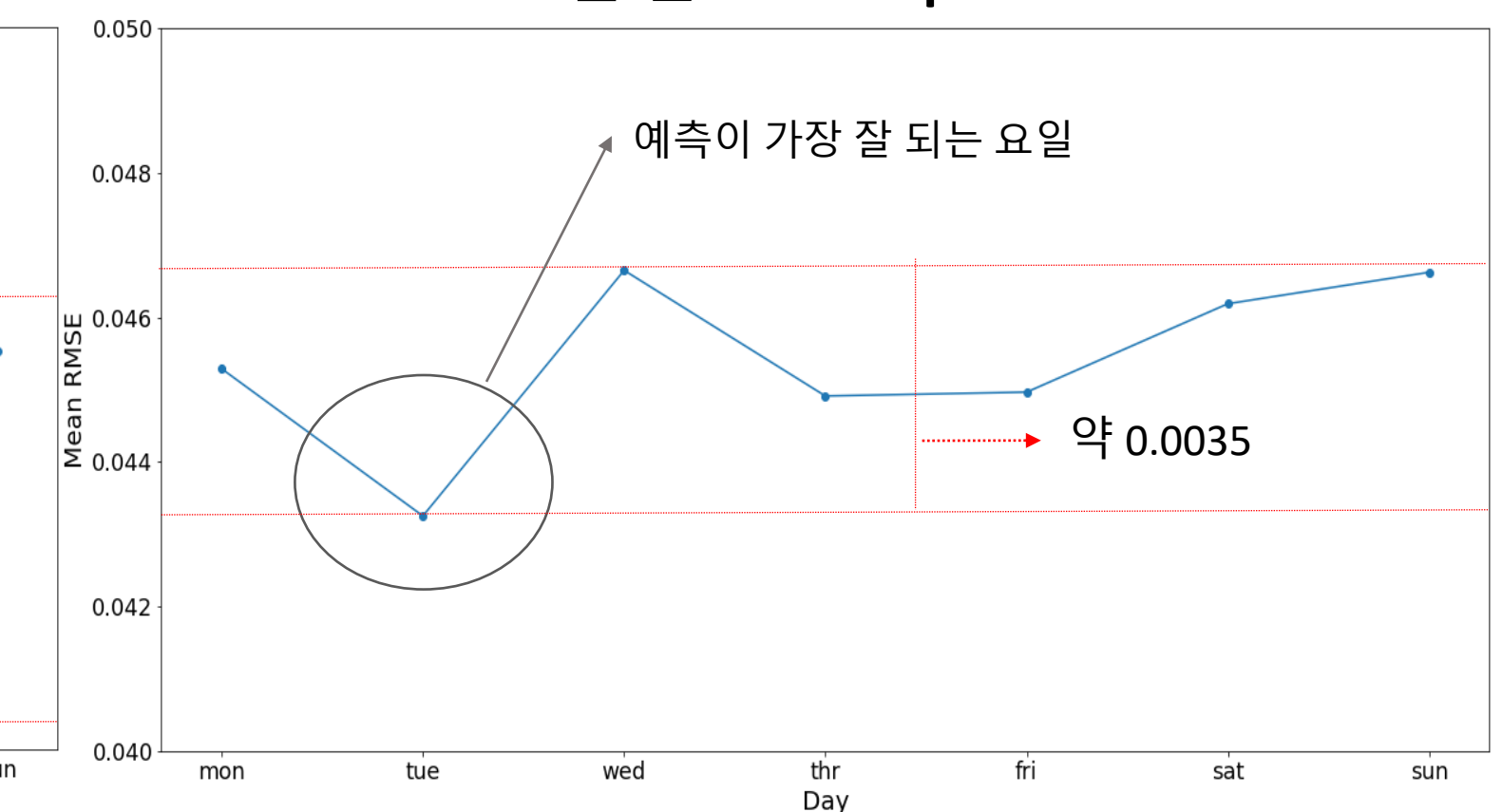


d) PM2.5 in Beijing multistep-ahead (168 -> 24)

<Bike 요일 별 Multistep-ahead RMSE>



<PM2.5 요일 별 Multistep-ahead RMSE>

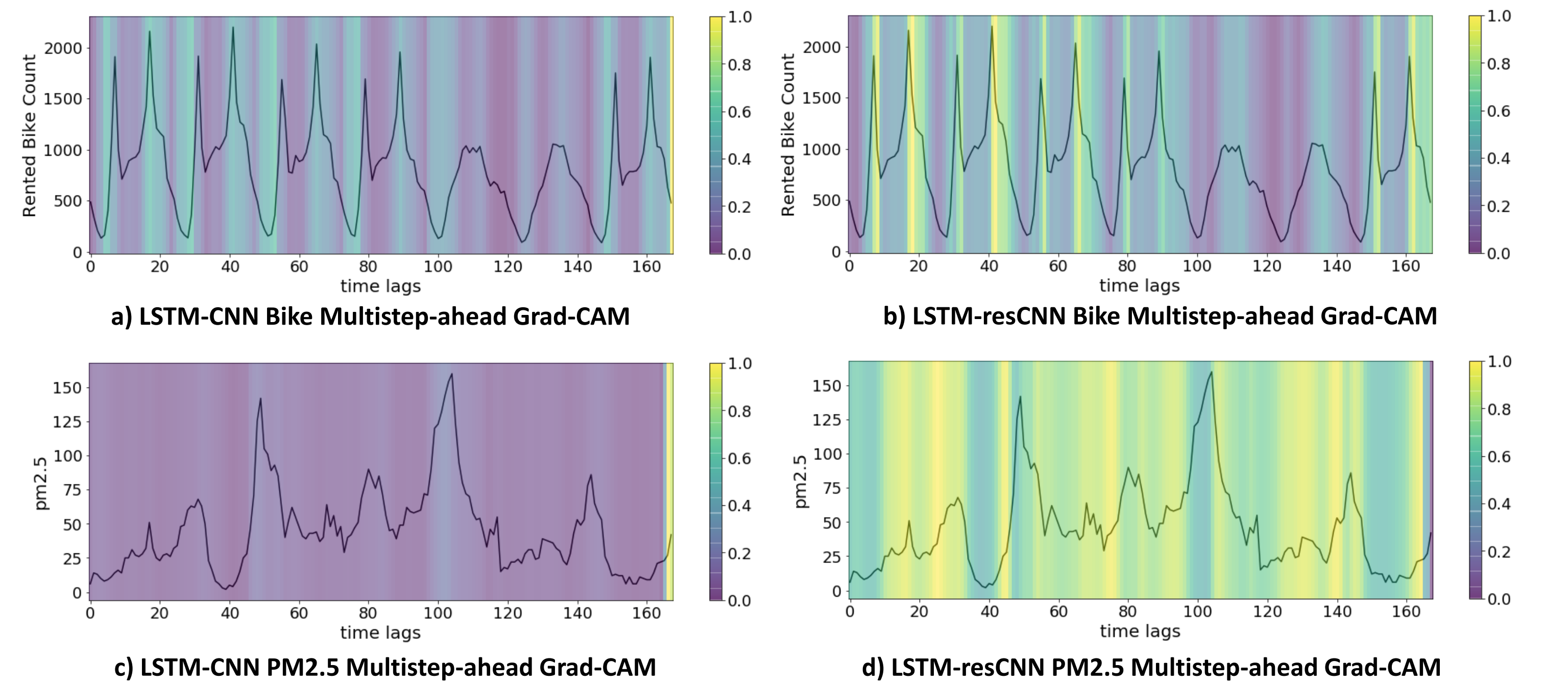


Visual Explanation : Grad-CAM

- CNN 모델이 Input 데이터의 어떤 부분을 더 많이 보고 예측했는지를 시각적으로 확인할 수 있는 획기적인 기법
- 데이터를 인식하는데 있어 온전히 +의 영향을 주는 곳만 표시하기 위해 ReLU 함수 이용

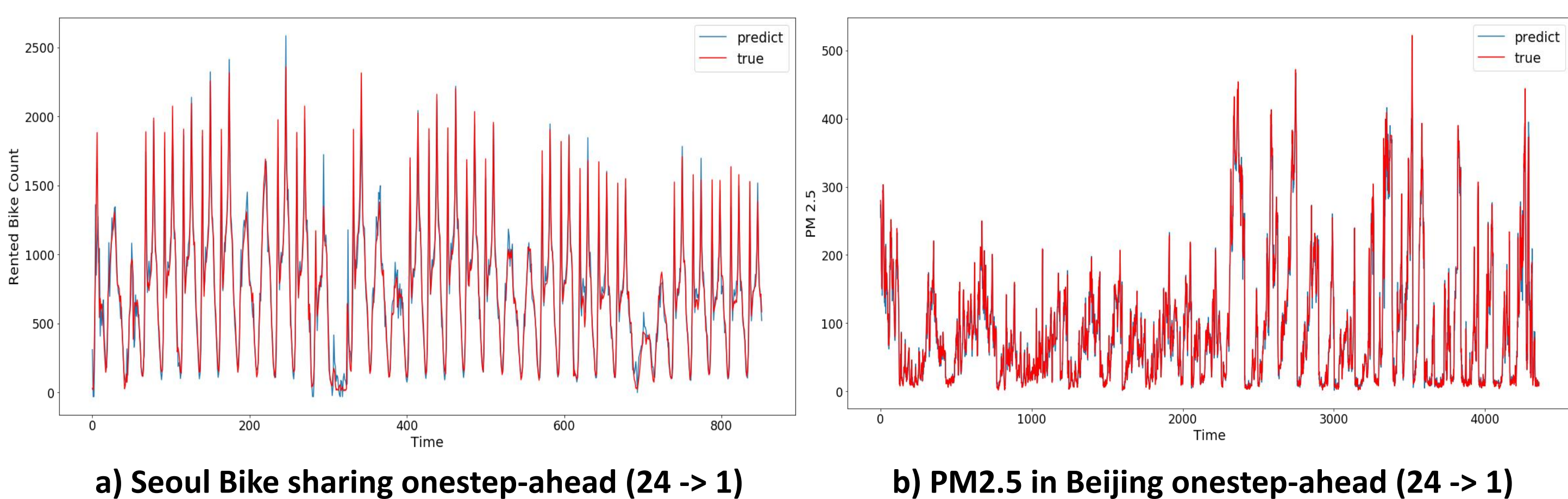
$$L_{\text{Grad-CAM}}^c = \text{ReLU} \left(\sum_k a_k^c A^k \right)$$

아래의 예시는 이전 168시간의 데이터로 향후 24시간을 예측함에 있어 이전 168시간 중 어떤 시점을 가장 많이 봤는가를 시각적으로 확인할 수 있다. 1에 가까울수록 밝은 색을 띄고 있으며 더 많이 본 시점이라는 것을 의미한다.



Conclusion

Residual Connection을 추가한 LSTM-resCNN은 일반적으로 성능이 가장 좋은 것으로 알려진 CNN-LSTM 보다 좋은 예측력을 보여주었다. 또한 CNN layer에서 Grad-CAM을 이용하여 모델 예측에 대한 설명도 가능하다. Residual Connection을 추가하기 전에는 오래된 과거 데이터보다는 예측하고자 하는 시점 바로 이전을 더 많이 보고, Residual Connection을 추가한 LSTM-resCNN은 바로 이전보다는 오래된 시점의 데이터까지 넓은 범위의 시점을 보고 예측하는 것을 확인 할 수 있었다. 또한 일주일의 데이터로 하루를 예측하는 경우 Bike와 같이 요일 별로 다른 패턴을 보이는 경우와 PM2.5 처럼 요일과 상관없는 패턴을 보이는 데이터들의 요일 별 예측 성능의 차이를 확인 할 수 있었다.



a) Seoul Bike sharing onestep-ahead (24 -> 1)

b) PM2.5 in Beijing onestep-ahead (24 -> 1)