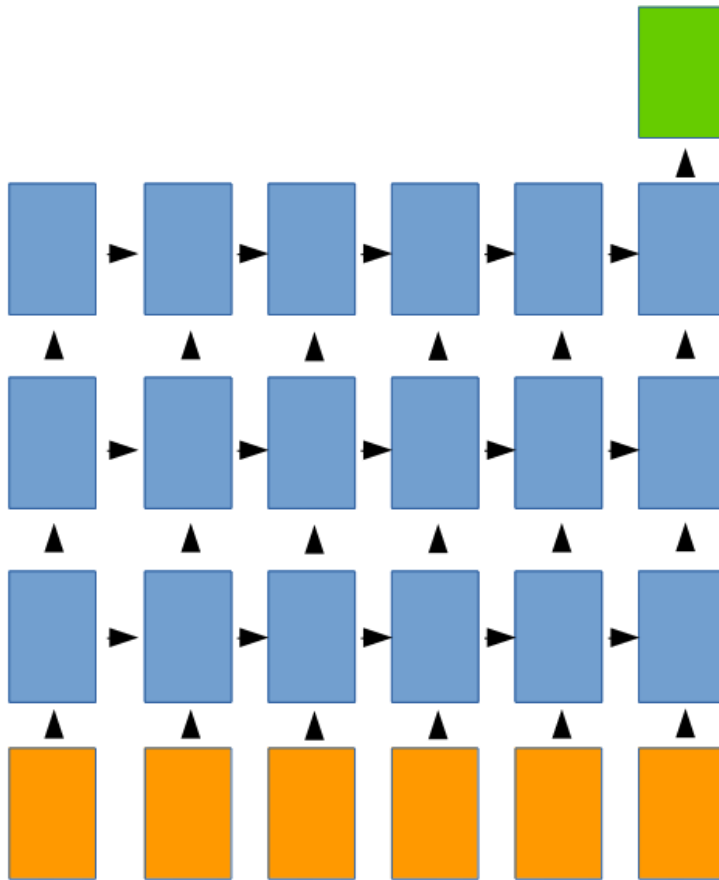


1. RNN 네트워크 구조



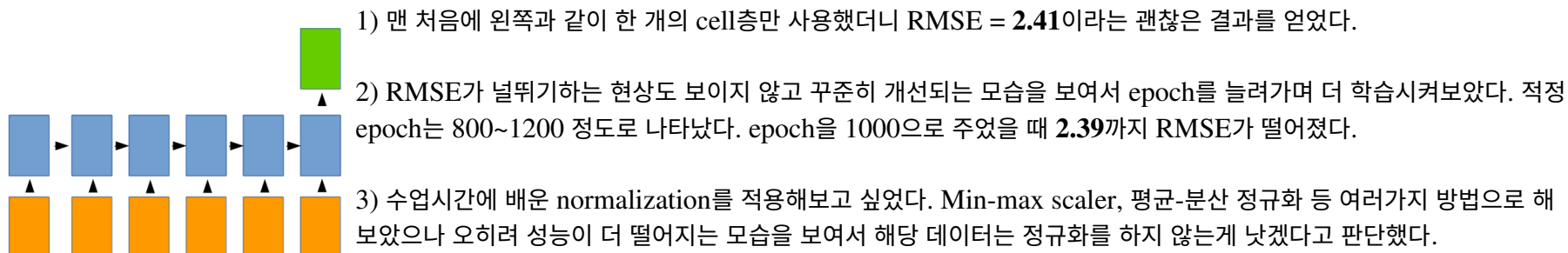
- 6일 간의 기온이 input으로 주어지고, 7일째의 기온 예측값이 output으로 출력된다. 따라서 다음과 같은 many-to-one 구조를 사용했다.

- RMSE 방법으로 성능을 측정하며 하이퍼파라미터를 튜닝한 결과, cell을 3층으로 쌓았을 때 가장 적절한 결과를 보였다.

- 더욱 자세한 설명은 2번 항목에서 하도록 하겠다.

2. 인자 탐색 과정 및 결과

cell층 갯수, learning rate, drop out, normalization 여부, layer tuning 조정, epoch 수 등을 바꿔가며 hyper parameter 탐색 과정을 거쳤다.



4) drop out & more layes: RNN에는 일반적으로 지금까지 우리가 해왔던 방식의 Dropout이 잘 동작하지 않는다는 것을 알게 되었다. dropout이 지워버리면 안되는 과거 information까지 전부 지워버리기 때문이라고 알려져있다.¹ 때문에 RNN에 Dropout을 적용하기 위해서는 recurrent connection이 아닌 connection 들에 대해서만 Dropout을 적용해야한다고 논문에서는 주장하고 있다. 따라서 cell layer를 더 쌓으면서 그것들 간에만 dropout을 적용하는 방식을 채택했다.

5) Learning rate: 0.001은 학습이 너무 늦어졌고, 0.01, 0.05정도로 주었을 때는 RMSE가 4.3근처에서 널뛰기 하는 현상을 보였다. 적정 learning rate은 0.005로 판단했다.

Summary) 3개의 cell층, non-recurrent connection간 dropout, learning rate = 0.005, no normalization, epoch 1000을 적용한 결과 2.13라는 RMSE를 달성했다.

3. 데이터 조작 과정 및 결과

데이터는 별다른 조작을 가하지 않기로 결정했다. 보유한 데이터를 정규화하기 위해 여러가지 방식을 취했으나 번번히 더 큰 RMSE를 보였기 때문이다. 시도 해본 방법으로는 min-max scaler, 평균과 분산을 이용한 정규화 등이 있다.

¹<http://sanghyukchun.github.io/89/>