

RNN을 활용한 도시철도 역사 부하 패턴 추정

Estimation of Electrical Loads Patterns by Usage in the Urban Railway Station by RNN

박 종 영*
(Jong-young Park)

Abstract - For effective electricity consumption in urban railway station such as peak load shaving, it is important to know each electrical load pattern by various usage. The total electricity consumption in the urban railway substation is already measured in Korea, but the electricity consumption for each usage is not measured. The author proposed the deep learning method to estimate the electrical load pattern for each usage in the urban railway substation with public data such as weather data. GRU (gated recurrent unit), a variation on the LSTM (long short-term memory), was used, which aims to solve the vanishing gradient problem of standard a RNN (recursive neural networks). The optimal model was found and the estimation results with that were assessed.

Key Words : Urban railway station, Load pattern, Deep learning, Recursive neural networks, Gated recurrent unit

1. 서 론

효율적인 에너지 소비의 필요성이 높아지면서, 용량이 큰 부하 중의 하나인 도시철도의 전력부하에 대한 관심 역시 높아지고 있다. 도시철도 전력부하는 크게 차량부하와 역사부하(고배용 부하)로 나눌 수 있다[1]. 이 중 차량부하는 차량의 운행일정에 따라 결정되기 때문에 조정하는 것이 어려우며, 역사부하에 대하여 부하 관리 기술이 연구되고 있다.

에너지저장장치를 통하여 첨두부하를 관리하는 기술이 연구되고 있는데, 도시철도 전력부하의 첨두 발생 시간대가 전력가격의 첨두 시간대와 다른 특성을 활용하여 전기요금을 절감하는 방법이 제안되었다[2]. 그리고 에너지저장장치를 첨두부하 관리 및 전력사용량 감소에 활용할 때 현재가치법(net present value, NPV)을 적용하여 경제성을 분석하였으며[3], 이렇게 확보한 전력수요자원으로 수요자원시장에 참여하였을 때 경제성을 분석하였다[4]. 이처럼 신재생에너지원, 에너지저장장치나 스마트그리드 기술 및 에너지관리시스템의 적용에 대한 연구 및 실증이 이루어지고 있다[5, 6].

이러한 연구들을 살펴보면 역사부하 중 그 사용량을 조절할 수 있는 부하의 비율을 파악하는 것이 필요하다. 예를 들어 전력수요자원 관리를 위해서는 필요시 차단이 가능한 부하량과 불가능한 부하량을 파악하는 것이 필요하다. 이를 위하여 계측 설비를 설치하여 각 용도별 소비전력을 직접 측정할 수도 있지만, 설치 비용이나 유지 비용이 발생하게 된다. 따라서 추가적인 계측

설비를 설치하지 않을 수 있는 방법이 바람직하다.

최근 들어 딥러닝(deep learning)과 같은 인공지능(artificial intelligence, AI) 기술이 발전하면서 전력계통 분야에서도 수요 예측 등 여러 가지 적용을 시도하고 있다. 딥러닝 중 RNN(recursive neural network, 순환신경망)을 적용하여 풍력발전량을 장기 예측하거나[7, 8], RNN 중 LSTM(long short-term memory, 장단기 기억 네트워크) 기법을 적용하여 가정용 전력부하를 예측한 연구가 수행되었다[9]. 전력수요량 예측에 있어서도 딥러닝의 입력 패턴에 따른 결과를 분석하거나[10], LSTM을 적용하여 전력부하를 예측하였다[11].

현재 기본적으로 측정되고 있는 데이터에는 도시철도 변전소의 전력소비량이 있으며, 그밖에 기온, 강수량, 미세먼지 등 기후 데이터 등이 공공데이터로서 공개되고 있다. 이러한 데이터들에 대해 빅데이터 기법으로 부하를 분석하고 다항회귀법을 적용하여 용도별 부하량을 추정하였다[12]. 분석 결과 역사부하 중에서 가장 큰 비중을 차지하는 부하는 조명, 공조, 상가(商街)부하였다. 이들 용도별 부하량에 대하여 다항회귀 기법을 적용하여 직접 측정하지 않고 용도별 부하량을 추정하는 방법을 제안하였다.

본 논문에서는 딥러닝 기법을 적용하여 도시철도 역사부하의 용도별 사용량을 추정하였다. 도시철도 변전소에서 측정된 실제 부하 데이터를 학습데이터로 사용하였으며, 그 외에 기온, 강수량, 미세먼지 등 기후 데이터와 요일, 특수일 여부 등 공공 데이터를 학습데이터로 사용하였다.

일반적인 부하예측은 특정 시점에서 과거의 데이터를 사용하여 목적하는 데이터인 부하량을 예측하게 된다. 본 연구에서 추정하고자 하는 데이터인 용도별 부하 사용량은 수집된 다른 데이터들의 과거 값을 가지고 특정 시점에서 목적 데이터인 용도별 부하 사용량을 추정하는 것이다. 따라서 부하예측과 마찬가지로

* Corresponding Author : Smart Electrical & Signaling Division,
Korea Railroad Research Institute, Korea.
E-mail : jypark@krii.re.kr

Received : September 28, 2018; Accepted : October 30, 2018

로 시계열 데이터를 활용하여 원하는 값을 추정하는 문제이다. 이에 적합한 딥러닝 모델로서 시계열 예측 등에 많이 쓰이는 RNN 기법을 사용하였다. 구체적으로는 RNN 기법 중 LSTM과 비슷한 성질을 가지지만 더 간단한 구조를 가지고 있는 GRU(gated recurrent unit) 기법을 적용하였다[13].

2. RNN과 LSTM, GRU 모델

RNN(recurrent neural network, 순환신경망)은 은닉계층의 출력을 다시 입력으로 순환시켜 순차적인 데이터를 학습하여 분류(classification)나 예측을 수행하는 기법이다. RNN은 음성 인식, 필기체 인식, 텍스트 인식이나 시계열 데이터 예측 등 시계열 분석 관련 문제에 활용되고 있다.

기존의 DNN(deep neural network, 심층신경망)은 각 계층(layer)마다 파라미터들이 독립적이나, RNN에서는 이를 공유하여 구성하게 된다. 이렇게 RNN에서는 루프(loop)를 통하여 과거의 데이터가 미래에 영향을 줄 수 있게 되며, 은닉계층(hidden layer)은 일종의 메모리 역할을 하게 된다. 그림 1은 RNN의 네트워크 구조와 루프를 풀어서 표현한 그림이다. 루프를 풀어서 나타난 오른쪽 부분을 보면 하나의 네트워크가 여러 개 복사된 형태임을 알 수 있으며, 각각의 네트워크는 다음 단계로 정보를 넘겨주게 된다.

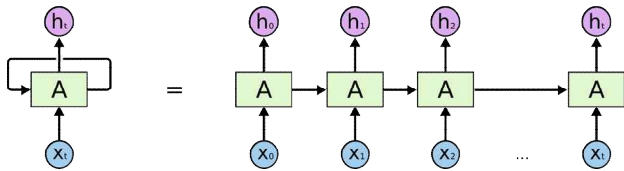


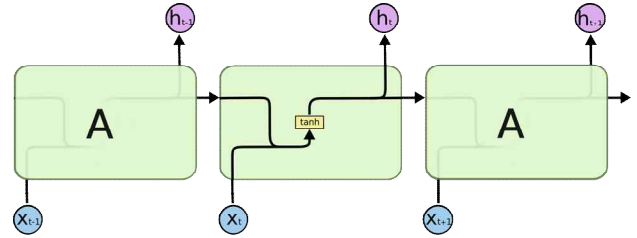
그림 1 RNN의 네트워크 구조 [14]

Fig. 1 Recurrent Neural Networks

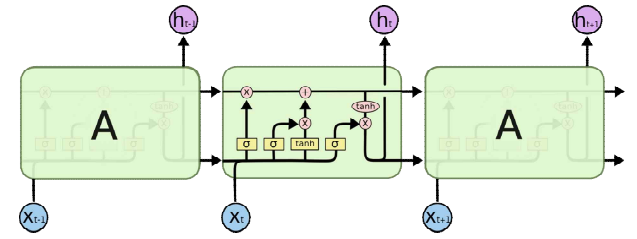
RNN은 이렇게 이전의 정보를 현재의 문제 해결에 활용할 수 있는 구조를 가지고 있다. 하지만 현재의 문제를 해결하는데 중요한 정보를 가진 데이터가 오랜 시간 전에 발생하였을 때, 이를 반영하기 힘든 문제가 있다. 이러한 문제를 장기 의존성(long-term dependency) 문제라고 한다.

이러한 문제를 해결하기 위해 LSTM(long short term memory, 장단기 기억 네트워크)가 제안되었다. 그림 2는 표준 RNN 모듈의 한 예와 LSTM의 모듈을 그림으로 나타낸 것이다. LSTM도 체인구조를 가지고 있으나 각 모듈은 다른 구조를 가지고 있다. 이러한 구조를 통하여 과거의 정보가 크게 변하지 않고 다음 단계로 전달될 수 있게 된다.

GRU(gated recurrent unit)는 LSTM의 장점을 유지하면서도 계산복잡성을 낮춘 셀 구조이다. 그림 3은 GRU의 네트워크 모델을 나타내고 있다. 이는 몇 가지 게이트를 합쳐서 보통의 LSTM 모델보다 단순하게 만들어, LSTM과 비슷한 성질을 가지지만 계산량을 감소시켰다. 이 모델은 점점 더 많이 활용되고 있으며, 모델에 대한 더 자세한 수학적 해석은 참고문헌 [13, 14]를 참고할 수 있다.



(a) 일반적인 RNN의 네트워크 구조



(b) LSTM의 네트워크 구조

그림 2 LSTM의 네트워크 구조 [14]

Fig. 2 Long Short Term Memory Networks

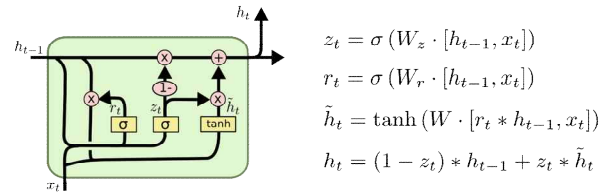


그림 3 GRU의 네트워크 구조 [14]

Fig. 3 The Gated Recurrent Unit Networks

3. 수요 추정 딥러닝 모델

서론에서 말한 바와 같이 딥러닝을 전력수요예측에 적용하는 연구들이 이루어지고 있다. 본 연구에서는 현재 측정되고 있는 도시철도 변전소의 전체 전력 사용량을 알고 있다는 가정 하에서, 딥러닝을 이용하여 역사부하 중 큰 비중을 차지하는 조명, 공조, 상가 부하의 부하량을 추정하였다. 이를 위하여 기온, 강수량, 미세먼지 등 기후데이터와 요일, 특수일 등의 공개되어 있는 공공데이터를 학습데이터로 입력하였다. 데이터 처리와 딥러닝 모델 구현은 파이썬(python) 언어 및 케라스(Keras) 모듈을 사용하였다.

3.1 데이터 생성

본 논문에서는 2016년 8월부터 2017년 7월까지 도시철도 변전소에서 측정한 전력부하 데이터에 대하여 학습을 수행하였다. 목표 데이터는 각 시간대별 조명, 공조, 상가 부하량이고 전체 전력 부하량은 학습 데이터로 정하였다. 전체 전력 부하량 외에 기온, 강수량, 미세먼지(PM10, PM2.5) 등 기후 데이터와 시간, 요

일, 특수일 여부 데이터를 학습 데이터로 추가하였다.

데이터는 학습(training), 검증(validation), 평가(test) 데이터로 구분했다. 학습데이터는 모델을 학습하는데 사용하는 데이터이고, 검증데이터는 학습이 진행되는 동안 성능을 검증하는데 사용하는 데이터이며, 평가데이터는 학습을 마치고 나서 모델의 성능을 평가하는데 사용하는 데이터이다. 일반적으로 검증데이터는 학습데이터에서 일정 비율로 추출하여 사용하며, 학습할 때마다 검증데이터에 대한 손실값을 평가함으로써 모델이 학습 데이터에 대해서만 과적합(overfitting)되는 것을 방지하는 역할을 한다.

신경망은 -1에서 1 사이의 값에서 가장 잘 작동하므로 신경망에 입력하기 전에 데이터 값의 범위를 조정해야 한다. 따라서 다음 식을 통하여 학습 데이터를 정규화하였다.

$$x_{new} = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (1)$$

그림 4는 정규화한 학습 데이터인 기온과 목표 데이터인 공조 부하 데이터의 일부를 그래프로 나타낸 그림이다.

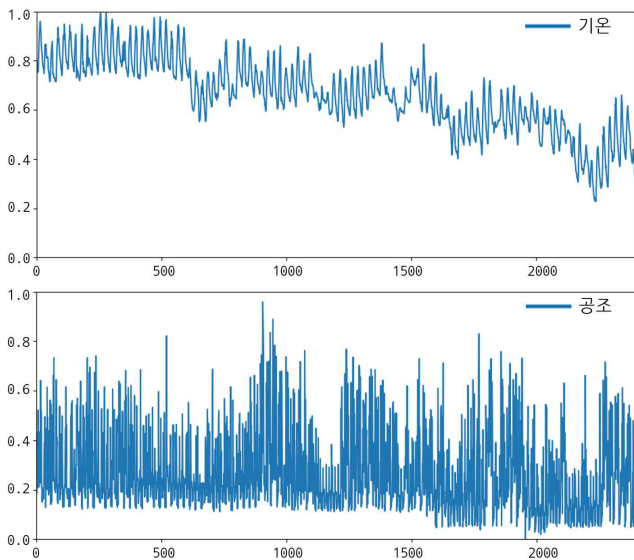


그림 4 정규화된 학습 데이터와 목표 데이터의 예
Fig. 4 Example of normalized training data

이렇게 구성된 학습데이터를 이용하여 학습을 수행하였는데, 학습하는 과정에서 1회 연산 시 적용하는 데이터의 범위에 따라 결과의 정확도가 달라질 수 있다. 너무 좁은 범위의 데이터를 사용하면 고려할 수 있는 특성을 놓칠 수 있고, 넓은 범위의 데이터를 사용하면 특성이 열어질 수 있다. 이 논문에서는 연산 시 1 주 전까지의 데이터를 사용하는 경우와 2 주 전까지의 데이터를 사용하는 경우를 비교하였다.

3.2 RNN 모델 구성 및 학습

본 논문에서는 RNN 기법 중 하나인 GRU를 사용하여 모델을

구성하였다. 입력계층에 GRU를 사용하였으며, 출력데이터는 0과 1 사이로 제한되어 있어, 출력계층은 시그모이드(sigmoid) 활성화 함수를 사용하였다. 이 논문에서는 GRU 계층의 개수에 따른 모델의 정확도를 비교하기 위해 1개 계층부터 최대 4개까지의 GRU 계층을 직렬로 연결한 모델까지 구성하여 결과를 비교하였다.

학습에서 사용하는 손실 함수(loss function)는 최소제곱오차(mean square error, MSE)를 최소화할 손실 함수로 사용하였다. 이를 통하여 모델의 출력값이 출력 신호와 얼마나 일치하는지 측정하였다. 그리고 RNN 모델의 특성상 학습 초기 시퀀스의 시작 부분에서는 입력 신호가 부족하여 부정확할 수 있다. 따라서 모델의 일정 시간 단계를 예열(warmup) 기간으로 생각하고 정확도를 신뢰하지 않음으로써 정확도 향상을 꾀했다.

그리고 학습을 한 번 할 때 모든 데이터를 계산에 넣을 수 없으므로 배치(batch)를 만들어 한 번 계산에 들어가는 데이터 수를 정하였다. 본 논문에서는 배치수를 64개와 128개로 하여 그 결과를 비교하였다.

신경망의 가중치를 조정하는 최적화 방법으로 경사하강법(gradient descent) 방법 중 하나인 RMSProp을 적용하였다. RMSProp은 Adagrad(adaptive gradient)의 단점을 해결하기 위해 제프리 힌튼(Geoffrey Hinton)이 제안한 방법이다. Adagrad는 변수들 중에서 업데이트 시 많이 변화하지 않아 그라디언트가 낮은 변수들은 스텝 사이즈를 크게 하고, 많이 변화해서 그라디언트가 큰 변수들은 스텝 사이즈를 작게 하여 최소값을 찾아가는 속도를 높이는 방법이다. RMSProp은 업데이트를 할 때 Adagrad와 같이 제곱 그라디언트의 평균을 이용하지 않고, 이동평균(moving average)를 이용함으로써 급진적이고 단조 감소하는 학습속도를 경감한다.

학습하는 동안 Keras의 콜백(call back) 기능을 활용하여 마지막 에포크(epoch) 이후 검증 데이터에 대한 손실값이 개선되지 않으면 학습률을 감소시키게 설정하였다. 이 때 최소 학습률은 1e-4로 하였다. 그리고 정해진 수의 에포크가 다 수행되지 않았더라도 5회 동안의 에포크 동안 손실값이 개선되지 않으면 학습을 중지하도록 하였다.

4. 결과 분석

4.1 모델 파라미터 결정

앞 장에서 언급한 바와 같이 최적의 학습 모델을 찾기 위하여 표 1의 조건들에 대하여 학습을 수행하고 그 결과를 비교하였다. GRU 모델 생성 및 학습은 파이썬 언어 환경에서 Keras 모듈을

표 1 GRU 모델 생성 파라미터의 조건들

Table 1 Parameters for GRU model

항목	조건	
데이터 당 시간 범위	1, 2 주	
학습 당 배치(batch) 수	64, 128 개	
GRU 모델	계층 수	1, 2, 3, 4 개
	노드 수	32, 64, 128 개

통하여 구현하였다.

파라미터들의 각 조합에 대해서 모델을 생성한 다음 학습을 수행하고, 시험 데이터를 통하여 생성된 모델을 평가하였다. 그림 5는 이렇게 구성된 GRU 모델의 구조를 나타낸 그림이며, 예시로 GRU 계층이 2개인 모델을 나타내고 있다. 계층 수에 따라서 GRU 계층의 수가 달라지게 되며, 노드 수에 따라 GRU 계층 내 노드의 개수가 달라진다. 출력계층에는 조명, 공조, 상가에 대한 3개의 노드가 있다.

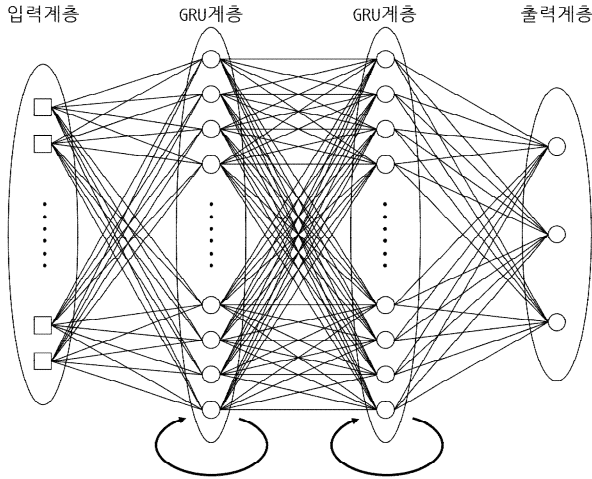


그림 5 GRU 모델의 구성
Fig. 5 Block diagram of GRU model

추정값이나 예측값과 실제 값 간 차이를 다루는 대표적인 방법에는 RMSE와 MAPE가 있다. RMSE(root mean square error, 평균제곱근오차)는 식 (2)를 통하여 구할 수 있으며 평가의 척도가 예측하려는 값의 크기에 의존적이라는 단점이 있다. MAPE(mean absolute percentage error, 평균절대오차백분율)은 식 (3)을 통하여 구할 수 있으며 RMSE의 단점을 보완하기 위한 방법이다. 이 논문에서는 MAPE를 통하여 모델에 의한 추정 결과를 평가하였다.

$$RMSE = \sqrt{\frac{1}{n} \sum_{t=1}^n (Y_t - F_t)^2} \quad (2)$$

$$MAPE = \frac{100}{n} \sum_{t=1}^n \left| \frac{Y_t - F_t}{Y_t} \right| \quad (3)$$

여기에서

- Y_t : 실제 값
- F_t : 모델에 의해 구한 추정값

평가를 위하여 세 가지 목적변수인 조명, 공조, 상가부하에 대하여 MAPE를 계산하여 비교하였다. 파라미터의 조합에 따른 모

델의 개수는 총 48가지이며, 각 모델에 대해서 세 가지 변수에 대한 MAPE와 그 평균값을 순서대로 정리했으며, 이 중 MAPE의 평균값이 작은 상위 10개의 모델을 표 2에 정리하였다. 여러 모델 중에서 MAPE의 평균값이 가장 작은 경우를 최적 모델로 정하였다.

표 2 파라미터에 따른 GRU 모델의 성능 비교

Table 2 Comparison of GRU models according to parameters

파라미터				MAPE			
배치	주	노드	계층	조명	공조	상가	평균
128	2	128	1	9.571	21.711	8.641	13.308
64	2	128	1	10.275	21.630	8.393	13.433
128	1	64	3	10.197	20.148	10.066	13.470
64	2	64	1	10.129	20.648	10.024	13.600
128	1	64	1	10.551	21.943	8.656	13.717
128	2	64	3	10.186	21.736	9.276	13.732
128	1	64	2	10.295	22.372	8.572	13.746
128	1	128	1	10.123	22.477	8.982	13.861
64	1	128	4	10.032	21.674	10.047	13.918
128	1	128	3	10.063	23.285	8.628	13.992

4.2 학습 과정 및 추정 결과

그림 6은 제안한 GRU 모델의 학습 곡선을 보여주고 있다. 학습곡선을 보면 학습 초기에는 학습데이터에 대한 손실함수의 값과 학습마다 검증데이터에 대한 손실함수의 값이 학습을 수행함에 따라 감소함을 볼 수 있다. 하지만 에포크가 17회 이상 넘어가면 학습데이터로 구한 손실은 계속 줄어들지만 검증데이터로 구한 손실은 줄어들지 않음을 볼 수 있다. 모델이 학습데이터에 대해 과적합되는 것을 막기 위하여 검증데이터에 대한 손실값이 줄어들지 않으면 학습을 조기 종료(early stopping)시켰다.

이렇게 학습된 GRU 모델로 목적변수인 조명, 공조, 상가의 값들을 추정하여 실제 값과 비교하였다. 그림 7은 평가데이터의 일

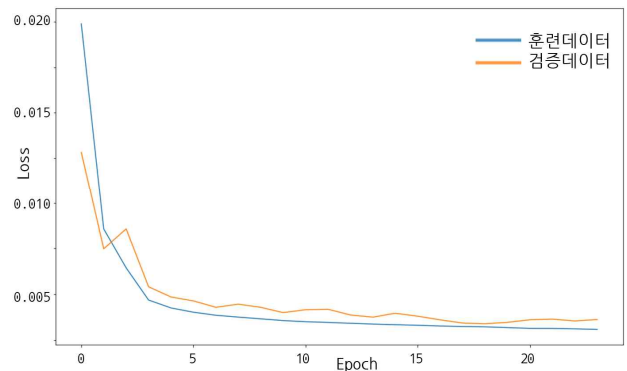


그림 6 제안한 GRU 모델의 학습 곡선
Fig. 6 Learning curve of the proposed GRU model

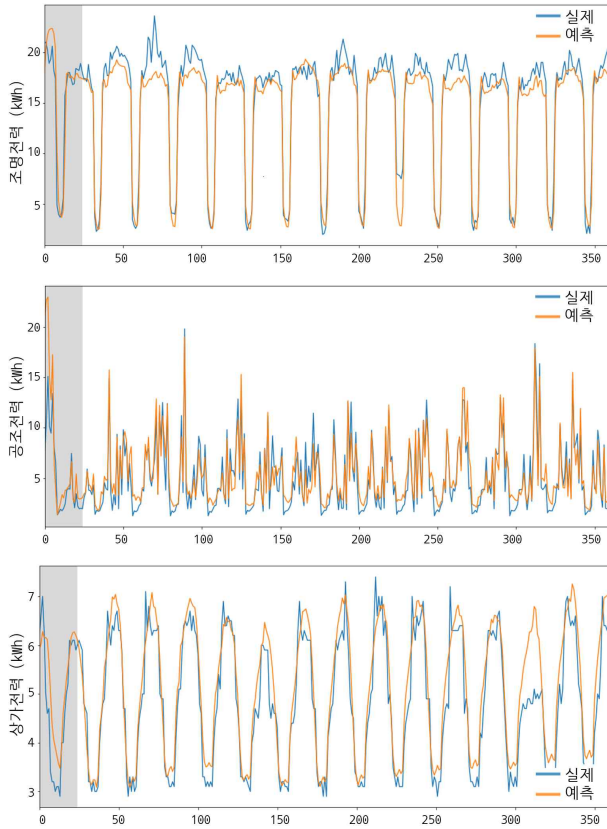


그림 7 딥러닝을 통하여 추정한 값과 실제 값의 비교 예
 Fig. 7 Example of estimate values of estimation by deep learning

부에 대하여 추정한 값과 실제 값을 그래프로 나타내고 있다. 실제 도시철도 변전소의 경우 동일한 노선에서는 거의 같은 용량과 구성을 가지게 되며, 도시철도 역사의 전력부하 구성 역시 지상역사끼리 또는 지하역사끼리는 크게 다르지 않은 부하 구성을 가진다. 따라서 이 연구에서 구성된 추정모델을 적용하면 같은 노선에 속한 각 변전소에서는 용도별 전력부하량을 추정 가능하다. 그리고 같은 도시철도 역사라도 설비의 증감 등 부하 특성이 변경되면 추정의 정확도가 떨어지게 될 것이다. 이를 보완하기 위해서 설비의 용량 등의 요소와 부하량을 추가로 고려한 모델을 구성하고 학습함으로써 더 범용인 부하추정모델을 만들 수 있을 것이다.

용도별 전력 부하량을 이렇게 딥러닝으로 구한 결과와 [12]에서 제안한 다항회귀를 통하여 구한 결과를 MAPE를 통하여 비교하면 표 3과 같다. 다항회귀는 3가지 목적변수 대해서 각각 상관관계가 높은 독립변수들에 대해 3차 또는 4차의 다항회귀분석을 수행한 결과이다. 상세한 다항회귀분석의 조건은 [12]에 기술되어 있다.

MAPE를 비교한 결과를 보면 3개의 목적변수 모두 딥러닝을 통한 추정의 결과가 더 좋아진 것을 알 수 있으며, 특히 조명부하에 대해서 더 많은 향상이 이루어졌다. 조명부하는 다른 두 부하에 비하여 규칙성이 강한 특성을 갖는데, 이 경우 딥러닝이 회

표 3 다항회귀분석과 딥러닝 기법에 의한 결과에 대한 MAPE 값 비교

Table 3 Comparison between results by polynomial regression and by deep learning

목적변수	다항회귀분석	딥러닝(RNN)
조명	17.380	9.571
공조	25.053	21.711
상가	9.958	8.641

귀분석에 비해 더 나은 결과를 보이는 것으로 보인다. 이와 같이 동일한 데이터에 대해 다항회귀분석과 딥러닝 방법 중 RNN을 통하여 부하를 추정해 본 결과 RNN이 더 좋은 성능을 나타내었으며, 데이터의 특성에 따라 그 향상 정도는 달랐다.

5. 결 론

도시철도에서 전력 소비의 효율을 높이기 위하여 여러 가지 부하관리 기술들이 적용되고 있으며, 이를 위해서는 용도별 전력 부하량을 파악하는 것이 필요하다. 이를 위하여 계측 설비를 설치하여 직접 측정할 수 있으나 설치나 유지관리 비용이 발생하게 된다. 이러한 계측 설비 추가 없이 현재 이미 측정하고 있는 변전소 전체 부하량과 공공 데이터인 기온, 강수량, 미세먼지량 등 기후 데이터를 이용하여 용도별 전력부하량을 추정하는 방법을 제안하였다.

기존 연구에서는 빅데이터 분석을 통한 다항회귀방법으로 용도별 전력부하량을 추정하였으며, 이 논문에서는 딥러닝 기법을 적용하였다. 딥러닝에는 여러 가지 기법이 있으며 시계열 데이터를 다루는 데에는 RNN(recursive neural network, 순환신경망)이 적당하다. RNN의 셀 가운데 하나인 GRU(gated recurrent unit)를 활용하였다. RNN은 과거 데이터의 반영이 어렵다는 단점이 있는데, 이 문제를 해결하기 위하여 LSTM(long short-term memory, 장단기 기억 네트워크)이 제안되었으며, GRU는 LSTM과 비슷한 특성을 가지면서 계산량을 줄인 방법이다.

제안하는 방법을 2016년 8월부터 2017년 7월까지 도시철도 변전소에서 측정된 전력부하 데이터에 적용하였다. 목적변수는 역사 부하에서 큰 비중을 차지하는 조명, 공조, 상가 부하량이며 학습 데이터는 전체 전력 부하량과 기온, 강수량, 미세먼지(PM10, PM2.5) 등 기후 데이터, 그리고 시간, 요일, 특수일 여부 데이터를 적용하였다. 데이터를 학습(training), 검증(validation), 평가(test) 데이터로 구분하여 학습을 진행하였다. 계층(layer)과 노드 수, 데이터의 학습 범위를 변경하면서 모델을 구성하고, 각 모델에 대하여 학습을 수행하여 모델의 최적 파라미터를 찾았다. 구성된 GRU 모델로 용도별 부하량을 추정된 결과 다항회귀분석을 통하여 추정된 결과보다 더 정확한 결과를 얻을 수 있었다.

지금까지 살펴본 바와 같이 이 논문에서는 전체부하량과 기타 공공데이터를 이용하여 원하는 용도별 부하량을 추정하였으며, 그 방법으로 딥러닝 중 RNN을 적용하였다. 일반적인 부하 예측을 하는 경우에는 특정 시점의 부하량을 목적변수로 하고 그 시

접에서 과거데이터를 일정 기간만큼 시프트 시킨 데이터를 학습 데이터로 이용하면 RNN을 적용하여 학습 및 예측이 가능하다.

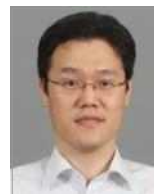
감사의 글

본 연구는 한국철도기술연구원 주요사업의 연구비 지원으로 수행되었습니다.

References

- [1] Han-Su Kim and Oh-Kyu Kwon, "Power demand forecasting in the DC urban railway substation", *Trans. of KIEE*, Vol. 63, No. 11, pp. 1608-1614, Nov. 2014.
- [2] Jae-Haeng Heo, Seungkwon Shin, Jong-young Park, and Hyeongig Kim, "Study on the optimal operation of ESS considering urban railway load characteristic", *Trans. of KIEE*, Vol. 64, No. 10, pp. 1508-1516, Oct. 2015.
- [3] Jong-young Park, Jae-Haeng Heo, Seungkwon Shin, and Hyungchul Kim, "Economic evaluation of ESS in urban railway substation for peak load shaving based on net present value", *Journ. of Electr. Eng. Technol.*, Vol. 12, No. 2, pp. 981-987, Mar. 2017.
- [4] Jong-young Park, Jae-Haeng Heo, Hyeongig Kim, Hyungchul Kim, and Seungkwon Shin, "Economic evaluation of ESS applying to demand response management in urban railway system", *Trans. of KIEE*, Vol. 66, No. 1, pp. 222-228, Jan. 2017.
- [5] Hosung Jung, Hyungchul Kim, Seungkwon Shin, Kiyong Yoon, Jae-moon Kim, and Yang-su Kim, "Installation of power monitoring system for load pattern analysis on DC urban transit system", *ISGC&E 2013*, July 2013.
- [6] Hansang Lee, Seungmin Jung, Hosung Jung, Hyungchul Kim, and Gilsoo Jang, "Power management for electric railway system to reduce the railway operating cost", *2012 KIEE fall conf.*, pp. 411-413, 2012.
- [7] Thanasis G. Barbounis, John B. Theocharis, Minas C. Alexiadis, and Petros S. Dokopoulos, "Long-term wind speed and power forecasting using local recurrent neural network models", *IEEE Trans. Energy Conver.*, Vol. 21, No. 1, pp. 273-284, March 2006.
- [8] Zhichao Shi, Hao Liang, and Venkata Dinavahi, "Direct interval forecast of uncertain wind power based on recurrent neural networks", *IEEE Trans. Sustain. Energy*, Vol. 9, No. 3, pp. 1177-1187, July 2018.
- [9] Weicong Kong, Zhao Yang Dong, David J. Hill, Fengji Luo, and Yan Xu, "Short-term residential load forecasting based on resident behaviour learning," *IEEE Trans. Power Syst.*, Vol. 33, No. 1, pp. 1087-1088, Jan. 2018.
- [10] Dong-Ha Shin and Chang-Bok Kim, "A study on deep learning input pattern for summer power demand prediction", *Journ. of KIIT*, Vol. 14, No. 11, pp. 127-134, Nov. 2016.
- [11] Jun-Ho Park, Dong-Ha Shin, and Chang-Bok Kim, "Deep learning model for electric power demand prediction using special day separation and prediction elements extension", *Journ. of Advanc. Navigat. Technol.*, Vol. 21, No. 4, pp. 365-370, Aug. 2017.
- [12] Jong-young Park, "Analysis of electrical loads in the urban railway station by big data analysis", *Trans. of KIEE*, Vol. 67, No. 3, pp. 460-466, Mar. 2018.
- [13] Kyunghyun Cho, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation", *Proc. of EMNLP 2014*.
- [14] "Understanding LSTM Networks", <http://colah.github.io/posts/2015-08-Understanding-LSTMs/> (accessed Sep. 11, 2018)

저 자 소 개



박종영 (Jong-young Park)

1976년 8월생. 1999년 서울대학교 전기공학부 졸업. 2007년 동 대학원 전기컴퓨터공학부 졸업(공학박사). 2007~2009년 한국전기연구원. 2009년~2013년 LS산전. 2013년~현재 한국철도기술연구원 연계환승연구팀 선임연구원.

Tel: 031-460-5731

Fax: 031-460-5749

E-mail : jypark@krri.re.kr