

머신러닝을 이용한 태양광 발전량 예측 및 태양 추적 패널을 이용한 발전량 연구

기준홍* · 백승준* · 소재영* · 엄한결* · 신정헌**†

*, **, 홍익대학교 기계시스템디자인공학과

Solar Power Generation Prediction Using Machine Learning and Study of Power Generation Using Solar Tracking Panels

Jun Hong Ki*, Seung-jun Baek*, Jae-young So*, Han-gyeol Eom* and Jeong-Heon Shin**

*Under grad., Dept. of Mechanical System and Design Engineering, Hongik Univ.

** Professor, Dept. of Mechanical System and Design Engineering, Hongik Univ.

(Received January 1, 2018 ; Revised January 2, 2018 ; Accepted January 1, 2018)

Key Words: Machine Learning(기계학습), Recurrent Neural Network(순환신경망), Pearson Correlation (피어슨 상관관계), Ensemble(앙상블), Photovoltaics(태양광 발전), Field Experiments (실험), Sun tracking system (태양추적시스템), Computation of sun position (태양위치계산)

초록: 전 세계적으로 신재생 에너지의 중요성이 부각됨에 따라 대한민국은 2030년까지 신재생에너지 발전비중을 30.2%까지 늘리는 것을 목표로 하고 있다. 태양광 발전은 국내 신재생 에너지 발전량 중 80% 이상을 차지하는 중요한 역할을 한다. 하지만 태양광 발전은 기상요소에 따라 발전량이 크게 변동하므로 안정적인 에너지 확보를 위해서 태양광 발전의 불안정성에 대한 대응과 발전량 증대가 필요하다. 따라서 본 논문에서는 기상데이터를 기반으로 머신러닝 모델을 학습시켜 태양광 발전량을 예측하고, 계절 및 시간에 따라 변화하는 태양 고도와 방위각을 계산하고, 태양을 추적하도록 발전 패널 제어하여 태양광 발전량을 증가시킨다. 실험 결과 0.208의 제곱평균오차 이내로 발전량을 예측했고, 태양 추적 전후 발전량의 차이가 23.4%임을 확인하였다.

Abstract: As the importance of renewable energy is growing worldwide, Republic of Korea aims to increase the proportion of renewable energy generation to 30.2% by 2030. Solar power plays an important role, accounting for more than 80% of domestic renewable energy generation. However, since the amount of solar power generation varies greatly depending on weather factors, it is necessary to respond to the instability of solar power generation and increase the generation amount in order to secure stable energy. Therefore, in this paper, we learn a machine learning model based on weather data to predict the amount of solar power generation, calculate the solar altitude and azimuth that change according to seasons and time, and increase the amount of solar power generation by controlling the solar power panel. As a result of the experiment, the power generation was predicted to be within 0.208 mean square error, and it was confirmed that the difference in power generation before and after sun tracking was 23.4%.

1. 서 론

기후변화협약 및 유가 불안정 상황을 대비한 신재생 에너지의 관심과 중요성은 상승하고 있다. 여러 신재생 에너지 중 태양광 발전은 국내 신재생

에너지 발전량 중 80 % 이상으로서 중요한 에너지 원이지만, 일사량, 기온 등의 기상 요서에 따라 민감하게 변동하기 때문에 일정한 에너지 수요를 충족시키지 못하는 경우가 발생한다. 한편, 국내 에너지 전환을 위한 구체적 로드맵인 3020 이행 계획은 재생에너지의 발전량 비중을 20 %, 누적 설비용량을 64 GW 보급, 신규 설비용량의 95 %는 태양광 및 풍력 등의 청정에너지로 공급하는 것이다. 즉, 태양광 발전은 충분한 에너지 원으로서

† Corresponding Author, jshin@hongik.ac.kr

© 0000 The Korean Society of Mechanical Engineers

인정을 받지만, 불안정적인 에너지 공급은 불안요소이다. 따라서 태양광 발전량 예측을 통해 발전량 증대에 대한 연구는 필요하다.

한편, 태양광 발전량 예측을 위한 머신러닝을 이용한 연구는 최근 발표되고 있다. 태양광 발전에 머신러닝을 이용하는 방법은 태양광 발전량에 영향을 미치는 기상데이터 중 주요 기상 변수들을 바탕으로 머신러닝 모델을 학습시킨 뒤, 태양광 발전량을 예측하는 것이다. 정밀한 발전량 예측을 하게 되면 발전량과 수요량의 차이로 인한 전력공백 등의 사고의 예방 및 비용을 절감할 수 있게 된다. 또한, 발전량 증대를 위해서 고정된 패널의 사용 대신, 태양을 추적하는 패널을 사용할 수 있다. 이는 태양의 고도와 방위각을 매 순간 변화하여 조사되는 태양광이 패널과 수직이 되도록 하게 함으로써 태양광을 적극적으로 흡수하는 방법이다. 이를 수행하기 위해서는 태양광 패널이 위치한 곳의 위도를 고려하여 태양의 고도 및 방위각을 계산하고, 계산된 결과를 패널에 적용하여 태양을 추적하도록 해야 한다.

본 연구에서는 태양광 발전량에 영향을 미치는 주요변수를 고려하고, 기상청 및 공공데이터 포털에서 제공하는 기상 데이터를 이용하여 머신러닝을 수행하였다. 한편으로는 태양의 고도를 추적하도록 하는 태양광 패널 장치를 설계하여 제작함으로써 태양광 발전량을 측정하고, 고정패널과의 차이를 분석하여 태양광 발전량 예측에 대한 가능성을 확인하였다.

2. 데이터 확보와 전처리

2.1 데이터 확보와 데이터 처리

본 연구의 수행한 장소인 서울시 마포구에서의 태양광 예측을 하기 위하여 서울시의 과거 기상정보와 태양광 발전량 데이터를 이용하였다. 기상정보는 기상자료개방포털 (kma.go.kr), 태양광 발전량 데이터는 공공데이터 포털(data.go.kr)에서 제공하는 한국전력거래소 자료를 사용하였다. 기상정보는 과거부터 최근의 데이터까지 폭넓게 제공하고 있었으나, 발전량 데이터가 2018 ~ 2020 년까지만 제공되어서 총 3 년간의 데이터를 사용하였다. 기상정보를 포함하는 파일에는 일부 측정값이 누락되어 있었고, 선형 변화율을 적용해 누락된 측정값을 대체하였다. 이 과정에서는 Python의 Pandas Library를 사용하였고, 선형 변화율에 대한 식은 다음 식과 같다.

$$NaN = a + \frac{(b-a)}{(i+1)-(i-1)}[i - (i-1)] = \frac{(a+b)}{2} \quad (1)$$

[여기서, i 는 데이터 순서, a 와 b 는 각각 $i-1$ 과 $i+1$ 에서의 data 값을 나타낸다.]

또한, 일조시간과 일사량의 경우 해가 떠있지 않은 시간대의 기록되지 않은 측정값은 0으로 대체하였다.

2.2 주요 Feature 선정

우수한 예측성능을 보이는 기계학습 모델을 만들기 위해서는 연관성이 높은 변수로만 학습을 진행하는 것이 굉장히 중요하고, 연관성이 없는 변수가 학습에 반영될 경우, 예측 성능을 감소시키는 요인이 된다. 따라서, 태양광 발전량과 높은 연관성을 가지는 기상 변수를 선별하기 위해 Pearson 상관관계 분석을 진행하였다. Numpy Library의 “corrcoef()” 메서드를 사용해 변수들 간의 상관관계를 분석하였다. 상관계수가 양수일 때는 양의 상관관계를, 음수일 때 음의 상관관계를 나타낸다. 또한 0에 가까울수록 상관관계가 없는 것이다. 피어슨 (Pearson) 상관관계 분석 결과와 관련 연구[5]에서 진행된 실험사례에서도 적용하였듯이 시간, 일조, 일사, 전운량, 중하층운량, 지면온도, 시정 변수들을 채택하였다. Table 1에 최종적으로 채택된 변수와 target 간의 상관계수를 나타냈다. 이 기상 변수들과 태양광 발전량을 Pandas Library의 “concat” 메서드로 병합하고 “to_csv” 메서드를 통해 csv 파일로 변환해 데이터를 저장하여 기계학습을 진행하였다. Pandas Library의 “.read_csv()” 메서드를 활용하여 앞서 정리한 csv 파일을 읽었고, 기상변수들의 단위가 서로 다르기 때문에 학습과정에서 특정 변수에 대한 편향이 발생할 수 있기 때문에 이러한 현상을 방지하기 위해서 Scikit-Learn Library의 MinMaxScaling 메서드를 이용해 모든 수치 데이터의 범위를 0~1 사이로 조정하는 scaling을 하였다. Scaling을 한 후, 18년 1월부터 20년 11월까지의 학습용 데이터로 지정하고 학습과정에서 모델 개선을 위해 사용되는 검증용 데이터의 비율을 20%로 구분하여 설정하였다. 20년 12월 데이터는 학습이 완료된 모델의 성능 시험을 위한 예측용 데이터로 지정하였다. 데이터의 양이 3년치로 한정되어있기 때문에 학습용 데이터셋의 비중을 크게 선정하였다.

Table. 1 Correlation coefficient of selected weather variables

기상변수	시간	일조 (hr)	일사 (MJ/m ²)	전운량	중하층운량	지면온도 (°C)	전력거래량 (MWh)
시간	1	0.15	0.15	-0.026	-0.049	0.14	0.065
일조 (hr)	0.15	1	0.82	-0.38	-0.35	0.29	0.53
일사 (MJ/m ²)	0.15	0.82	1	-0.17	-0.18	0.56	0.72
전운량	-0.026	-0.38	-0.17	1	0.78	0.25	0.054
중하층운량	-0.049	-0.36	-0.18	-0.78	1	0.16	0.018
지면온도 (°C)	0.14	0.29	0.56	0.25	0.16	1	0.40
전력거래량 (MWh)	0.065	0.53	0.72	0.054	0.018	0.40	1

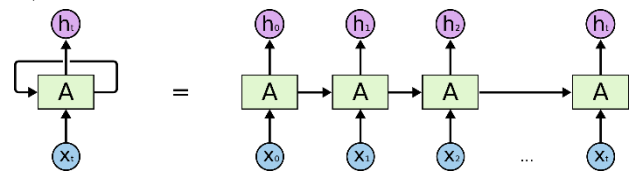
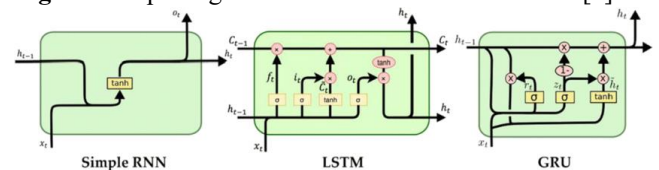
Tensorflow 의 Prefetch 자료구조 데이터셋으로도 학습이 가능함을 확인하였다. Prefetch 자료구조는 3 차원 Tensor 자료구조를 이용해 한 batch data 가 학습되는 동안 다음 학습에 투입될 data 를 미리 준비한다는 특징을 갖는다. 따라서, Numpy Array 데이터셋에 비해 학습속도가 빠르지만, 본 논문에서는 Ensemble 기법까지 활용하는 방식을 제안하기 때문에 호환성이 좋은 Numpy Array 자료구조로 데이터셋을 통일하여 진행하였다.

3. 순환신경망 머신러닝 모델 구축

3.1 단일모델 구성

앞에서 학습된 기록을 다음 단계의 학습에도 반영하는 구조를 순환신경망이라고 일컬으며 도식은 Fig.1 과 같이 학습을 진행하였다. 순환신경망은 시간의 흐름이 변수에 영향을 미치는 시계열 데이터 기반 학습 방식에 유리하다는 특징을 갖는다. 순환신경망으로는 Fig.2 와 같고, 대표적인 RNN, LSTM, GRU 모델이 있으며, 본 논문에서는 각각의 단일모델 예측 성능을 비교해보았다. 따라서, 각종 Parameter 들을 통일시켰으며 해당 내용은 Table 2 와 같고, 그것을 반영하여 RNN 모델을 구성하고, summary 를 출력한 결과는 Fig. 3 과 같다. LSTM 과 GRU 역시 동일한 조건으로 모델을 구성했으며,

반복학습 횟수를 의미하는 epoch 을 변화시켜가면서 예측성능이 어떻게 변하는지 실험을 진행하였다.

**Fig. 1** Example Figure of Recurrent Neural Network [5]**Fig. 2** Various Types of Neural Network models [5]**Table. 2** Main Parameters for Comparing with RNN, LSTM, GRU each single models

Activation Function	tanh
AI Layers	4
Perceptrons for each layer	32, 64, 64, 1
Loss Function	Mean Squared Error
Optimizer	Adam Optimizer
Metrics	Mean Absolute Error
Dropout Rate	20%

Layer (type)	Output Shape	Param #
simple_rnn_3 (SimpleRNN)	(None, 5, 32)	1088
dropout_12 (Dropout)	(None, 5, 32)	0
simple_rnn_4 (SimpleRNN)	(None, 5, 64)	6208
dropout_13 (Dropout)	(None, 5, 64)	0
simple_rnn_5 (SimpleRNN)	(None, 64)	8256
dropout_14 (Dropout)	(None, 64)	0
dense_4 (Dense)	(None, 1)	65
Total params: 15,617		
Trainable params: 15,617		
Non-trainable params: 0		

Fig. 3 Summary of Machine Learning Model: RNN

3.2 Ensemble 모델 구성

학습데이터에 편향되어 예측 성능이 저하되는 overfitting 현상을 막기 위해 parameter 선정시 학습 데이터의 20 %를 검증용으로 설정하고, dropout rate 도 20 %로 설정하였다. 하지만, 단일모델 만으로 진행되는 학습과 예측의 한계를 극복하기 위해 Ensemble 기법을 사용하였다. Ensemble 기법은 여러 모델을 생성하고 각각의 모델들의 예측 결과를 바탕으로 확률이 높다고 판단되는 결과값이 최종적으로 출력하는 방법이다. 그 중에서 Boosting 기법과 Voting 기법이 대표적으로 사용되는데, 각 기법을 간단하게 Fig. 4(a)와 Fig. 4(b)와 같다.

Boosting 기법을 사용할 경우, 하나의 전체 학습용 데이터 셋을 여러 개의 서로 다른 구성을 가진 샘플 데이터 셋으로 분할하고, 이렇게 나누어진 샘플 데이터 셋들을 하나의 머신러닝 모델로 학습하고 예측하게 된다. 단일모델들의 성능을 우선적으로 비교한 뒤, 가장 좋은 모델을 선정하여 Boosting 기법의 estimator 로 사용한다. Epochs 역시

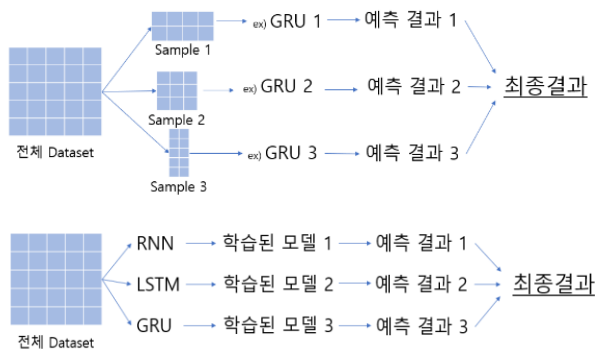


Fig. 4 Schematic of Process of (a) Boosting Ensemble (b) Voting Regressor Ensemble

단일모델 비교에서 가장 좋은 성능을 보였던 횃수를 사용하며, 총 estimator 의 수는 10 개로 제한하였다.

반면, Voting 기법은 같은 학습용 데이터로 서로 다른 머신러닝 모델에 학습시켜 예측을 진행한 뒤 가장 확률이 높다고 판단되는 결과값 하나만을 출력하는 방식이다. 단일모델 비교 방식과 마찬가지로, epoch 조절에 따라 성능이 어떻게 변하는지 확인하였다. 또한 Voting 기법에서는 각각의 모델에서 서로 다른 epoch 을 설정할 수 있으므로, 단일모델 비교에서 각각의 모델이 최적의 성능을 보였던 epochs 를 따로 적용해 예측하는 방식도 진행하였다.

4. 실험

4.1. 태양 추적 임베디드시스템 설계

4.1.1 태양 위치 계산

태양의 위치를 계산하는 대다수의 식들은 황도좌표계를 기준으로 하지만, 태양을 추적하기 위해선 지평좌표계를 기준으로 한 값이 필요하게 된다.. 본 연구에서는 서울시 마포구에 위치한 홍익대학교 (위도: 37.478°N, 경도: 127.148°E)에서 시간에 따른 태양의 (적위, 시간각)을 계산한 후 좌표 변환식을 통해 (방위각, 고도)를 구하였다. [1] 한편, 하루의 길이는 지구 공전 궤도의 이심률과 지축 경사에 의해 하루의 길이는 매일 조금씩 바뀐다. 대한민국은 동경의 경도를 기준으로 한 시간대를 사용한다. 따라서 정밀한 태양의 추적을 위해서는 지역표준시(Local Standard Time)을 평균태양시(Mean Solar Time)로 변환 후 진태양시(Local Solar Time)로 변환해야 한다. 진태양시(Local Solar Time)와 평균태양시(Mean Solar Time)의 차이를 균시차(ET, Equation of time)라고 한다. 이를 구하기 위한 수식은 수식(2)와 같다.

$$ET = 229.2 \times (0.000075 + 0.001868 \times \cos B - 0.032077 \times \sin B - 0.014615 \times \cos 2B - 0.04089 \times \sin 2B) \quad (2)$$

[여기서, $B = (n - 1) \frac{360}{365}$, ($n = \text{Day of year}$)]

시간각 (hour angle)이란 천구상에서 어떤 천체와 하늘의 극을 잇는 대원인 시권과 자오선이 이루는 각을 말하며 이를 구하는 식은 다음과 같다.

$$\omega = \frac{LT \times 60 + 4 \times (LL - LST) + ET}{60} \times 15 - 180 [^\circ] \quad (3)$$

[여기서 LT 는 지역표준시(Local Standard Time), LL 은 지역 경도 (Local Longitude) 그리고, LST 는 표준 자오선 경도(Standard Longitude)이다.]

적위(declination)는 천구상의 천체 위치를 나타내는 적도좌표에서의 지구상의 위도에 해당하는 좌표 값을 뜻하며 이를 구하는 식은 다음과 같다.

$$\delta = 23.45 \times \sin \left[\frac{360}{365} \times (284 + \text{Day of year}) \right] [^\circ] \quad (4)$$

이제 시간각과 적위라는 변수를 대입해 태양조도 측정을 위해서는 태양의 방위각과 고도가 필요하고, 시간각과 적위 변수를 다음의 식에 대입하여 구하게 된다.

$$\cos \varphi = \frac{\sin \beta \sin L - \sin \delta}{\cos \beta \cos L} \quad (5)$$

[여기서 β 는 태양고도, L 은 지역 경도, 그리고, δ 는 적위이다.]

태양의 고도, β 를 구하는 수식은 다음과 같다.

$$\sin \beta = \cos L \cos \delta \cos \omega + \sin L \sin \delta \quad (6)$$

[여기서 ω 는 hour angle 이다.]

앞에서 구한 태양의 방위각과 고도를 구하는 수식을 아두이노 코드로 구현하고 그래프를 구현했다. 이를 한국천문연구원의 데이터 (<https://astro.kasi.re.kr/life/pageView/10>) 와 비교하여 정확성을 검증했으며 그 결과는 Fig. 5 와 같다. 수식을 이용한 태양 고도 계산 값과 천문 연구원의 데이터의 차이는 최대 1°의 차이를 보이는 높은 일치도를 보였다.

4.2 실험준비

실험은 태양 추적형 시스템(Fig. 6(a))과 고정형 시

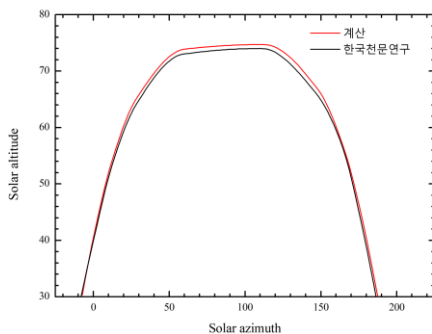


Fig. 5 A comparison between the calculation and the real data

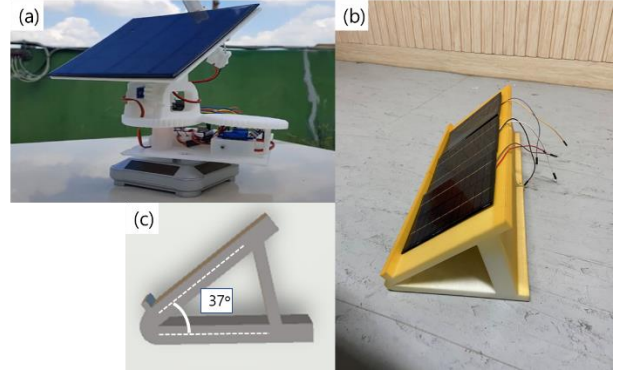


Fig 6. (a) Tracking system with solar panel (b) fixed system with solar panel (c) the frame used for the fixed system

스템 및 프레임 (Fig. 6(b) & (c))을 이용한 고정 시스템 두 가지로 준비하였다. 태양 추적형 시스템의 경우, 아두이노 태양광 자동추적기(EDUIot)를 이용하였고, 기존 조도 센서가 위치한 곳에 추가로 태양광 패널을 부착해 각 키트 당 3 개의 패널을 설치하고, 2 개의 키트를 통해 실험을 진행하였다. 고정형 시스템의 경우, 바닥과 패널이 놓인 판의 각도는 춘추분의 태양의 고도인 37° 로 설정하여 제작하였다. 구성된 장치는 서울 마포구에 위치한 홍익대학교 제 1 공학관 옥상에 위치하였다. 태양광 발전을 구현하기 위한 태양광 패널(DM528)은 출력전압 12V, 출력전류 160 mA, 출력전력은 1.92 W 의 사양을 갖고 있고, 크기는 13.5 cm × 11 cm 이었다. 태양광 패널에서 전환되어 출력되는 전류 측정을 위해 직류 전류센서 모듈(INA219, Texas Instruments)를 사용하였고, 0.08mA의 분해능과 1%의 오차율로 미세한 전류 측정이 가능하였다. 이를 통해 패널에 흐르는 전류(I)와 전압(V)을 계산한 뒤 5 분마다 발전량(mW)를 계산하여 데이터를 저장하였다. 패널의 각도와 태양의 움직임은 밀접한 관계를 갖게 되고, 시간의 측정과 동시간의 데이터의 관계가 중요하기 때문에, 서로 시간을 측정하기 위해 RTC 모듈인 DS1302를 사용하였다. 패널의 움직임은 Servo모터 (OEM, SG-90, 1.8kg·cm)를 통해 제어했다.

4.3 실험 과정

추적형 시스템의 경우, 앞서 구한 태양 고도 계산 값을 바탕으로 두 개의 모터는 5 분 간격으로 작동하여 태양광 패널이 태양의 고도가 최대가 되도록 작동하였다. 고정형 시스템의 경우, 실험을 진행한 기간이 6 월초 인 것을 감안하여 춘. 추분 시기의 태양의 고도가 대한민국의 위도 (37°N)와 같

은 37° 의 각도로 설정하였고, 방위각은 180° 즉, 태양의 패널이 정남방향을 향하도록 설치하여 진행하였다. 실험은 오전 9시부터 오후 5시까지 그림자가 생기지 않는 지점에서 고정형과 추적형 시스템에서 동시에 두 종류의 데이터를 저장했다. 발전량은 매 5 분마다 측정하였고, 아두이노 내 EEPROM 에 해당 값을 저장하였고, 수치적분을 진행하여 총 발전량을 구한 뒤 고정형과 추적형을 비교하여 추적형의 발전량 증가량을 계산하였다.

6. 실험 결과 및 고찰

6.1 단일모델 결과 비교

Table 2 에서 제시한 통일된 Parameters 로 모델을 구성하고 실험을 진행하였다. 반복 학습 횟수 (epochs)에 따른 예측 값과 실제 값 사이의 차이를 평균절대오차(MAE)로 나타낼 수 있었고 다음의 식(8)을 사용하였다.

$$MAE = \frac{1}{N} \sum_{i=1}^N | (predict_i - real_i) | \quad (8)$$

실험 결과는 Table 3 과 같다. RNN 모델의 경우 200 epochs 에서 0.253 로 가장 작은 오차를 보였고, LSTM 은 250 epochs 에서 0.237 로 가장 낮았으며, GRU 는 150 epoch 에서 0.229 로 가장 낮았다. 각각의 모델을 서로 비교한다면 150 epochs 에서의 GRU 모델이 단일 모델들 중에서는 가장 좋은 성능을 보였다.

6.2 Ensemble 기법 모델 결과 비교

단일 모델끼리의 성능을 비교한 결과 GRU 가 가장 좋았기 때문에, Boosting 기법에 GRU 모델을 사용하였다. 즉, 전체 학습용 데이터셋을 서로 다르게 분할하고, 각각의 분할된 데이터 셋을 GRU 모델 방식으로 학습시킨 뒤 예측을 진행하였다. Epochs 에 따른 MAE 는 Table 4 와 같다. 가장 좋은

Table.3 Comparison of MAE when using RNN, LSTM GRU single models

Epochs	RNN[MAE]	LSTM[MAE]	GRU[MAE]
100	0.288	0.325	0.253
150	0.373	0.265	0.239
200	0.263	0.259	0.252

250	0.308	0.247	0.265
-----	-------	-------	-------

Table.4 MAE result of AdaBoost – GRU model

Epochs	AdaBoost – GRU
150 (total: 1500)	0.336
200 (total: 2000)	0.341

성능을 보였던 단일 모델 GRU 에 Boosting 기법을 사용하면 더 좋은 예측 성능을 보일 것이라는 기대와 달리, 오히려 더 MAE 가 늘어나는 결과를 보였다. 이는 학습 및 검증 데이터 수의 한계로 판단된다. 따라서, Ensemble 기법의 또 다른 방식인 Voting Regressor 를 적용해보았다. 해당 모델에는 RNN, LSTM, GRU 방식이 모두 사용되며 각각의 모델이 예측한 결과 중 가장 정확도가 높을 것으로 판단되는 예측 결과값 하나만을 종합하여 출력하게 되는 원리이다. 또한 Voting 기법은 각 모델의 epochs 를 다르게 설정할 수 있다. Epochs 별 학습에 따른 성능 결과는 다음 Table 5 와 같다. 결과적으로 단일모델 RNN, LSTM, GRU 각각의 최적 epochs 를 반영하여 Voting 기법을 적용한 방식이 가장 좋은 예측 성능을 보여주었다. 이를 통해, 데이터가 충분치 않은 상태에서는 Voting Regressor 를 이용한 기계학습 예측 성능이 극대화될 수 있음을 확인하였다.

6.3 추적 유무에 따른 결과 비교

태양 추적 유무에 따른 발전량 차이는 Fig. 7 과 같이 나타났다. 앞서 설명하였듯이 데이터는 5 분 간격으로 측정되었고, 시시각각으로 변하는 기상 변수들로 인해 불규칙한 변화에 대해 1 시간동안의 데이터를 산술평균으로 대표 값을 이용하였다. 추적형과 고정 패널 모두 오전에서 정오로 시간이 변하면서 발전량이 증가하고, 이후 발전량이 감소하면서, 동시에 추적형 패널의 발전량이 고정형 패널에 비해 모든 시간대에서 높음을 확인할 수 있다. 이는 정오를 전후로 발전량이 많을 것이라는 예상과 다름없다. 이후 고정 패널의 발전량의 감소폭이 추적형 패널의 발전량의 감소폭에 비해 크음을 보여준다. 두 패널이 정오때의 발전량에 비해 오후가 되면서 추적형 패널이 상대적인 발전량의 절대량이 우월함을 보여준다. 시간과 발전량을 적분하여 발전 총량을 계산한 결과, 고정형 패널

이 251.7kJ, 추적형 패널이 310.6kJ로 총 발전량이 23.4% 증가함을 확인했다. 이를 통해 전체 발전량의 20% 이하의 전력을 추적형 패널의 동력 사용한다면, 추적형 패널의 효율이 더 높다고 판단할 수 있다.

6.4 실제 측정값과 예측값 결과 비교

6.2에서 제시한 Voting 기법 모델을 사용한 예측결과와 실험을 통한 측정 결과와의 비교를 진행했다. 2022년 6월 4일 기상예보자료를 input data로 사용하여 예측 값을 출력하였으며, 해당 날짜에 고정형 패널과 추적형 패널로 실제 태양광 발전량 측정 실험을 진행하였다. 결과는 Fig. 8 및 Table 5의 마지막 행의 결과와 같다. 기존 MWh 단위였던 예측 발전량 기준을 mW 단위의 실제 발전량으로 선형적 스케일링을 적용했다. 실제 발전량 그래프와의 경향성은 일치하는 반면, MAE는 4.340으로

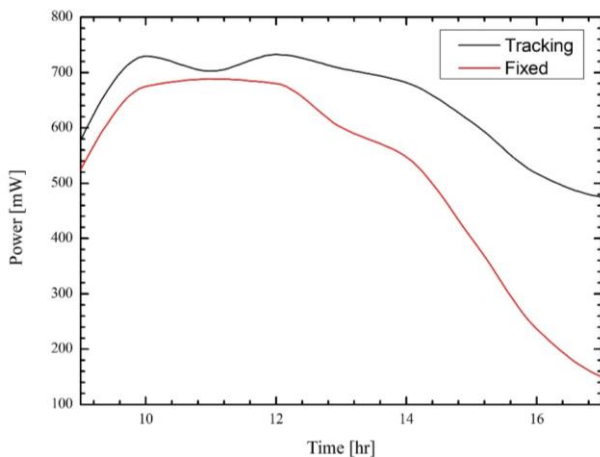


Fig.7 Difference in power generation with or without solar tracking

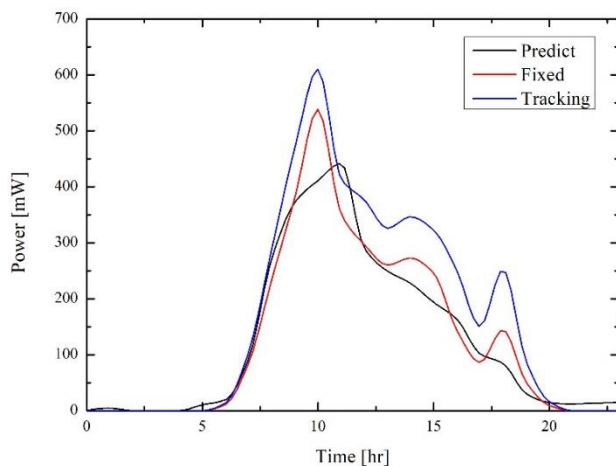


Fig.8 Predicted and Experimental Power variation on 4th June 2022

Table.5 Result of Voting Regressor model

Epochs	AdaBoost – GRU, MAE
RNN 150 LSTM 150 GRU 150 (total: 450)	0.243
RNN 200 LSTM 200 GRU 200 (total: 600)	0.233
RNN 200 LSTM 250 GRU 150 (total: 600)	0.209
[4th of June] RNN 200 LSTM 250 GRU 150 (total:600)	4.340

다소 큰 오차를 보였다. 원인 중 하나는 설치 규모와 실험 장치와의 규모 차이이다. 즉, 머신러닝 모델이 학습한 데이터는 대규모로 운용되는 서울시 태양광 발전량으로서 단위가 MWh 인 반면, 본 실험에서 구축한 소형패널의 경우는 mWh 단위의 발전량이고, 크기와 발전량이 선형적이지 않고, 그 관계는 추가 연구를 통해 확인해야 할 것으로 판단한다.

7. 결론 및 토론

태양광 패널을 이용한 발전 시스템에 대한 실험과 머신러닝을 이용한 발전량 예측 시스템에 대한 연구를 진행하였다.

1. 머신러닝 모델을 구축하는 과정에서 신경망인 RNN, LSTM, GRU 모델을 사용하여 기존 데이터를 예측하였을 때, 150 epochs의 GRU 모델이 가장 좋은 결과를 나타내었다.
2. 단일 모델 대신 Ensemble 기법 중 voting 기법을 적용하였을 때, 좋은 예측 성능을 보여주었다. 충분하지 않은 데이터로는 Voting Regressor를 이용한 예측이 가장 좋은 성능이 나타남을 확인하였다.
3. 태양 추적 시스템과 고정 태양광 패널 시스템에 대한 모형 실험을 진행하였다. 태양의 고도를 고려한 추적 시스템을 통한 태양광 발전은 고정 태양광 패널의 발전에 비해 23.4 % 증가

함을 확인하였다.

4. 실험을 통한 결과와 머신러닝 기법을 사용한 예측을 비교하였다. 전체적인 경향성은 예측하였지만, 그 예측의 정확도가 높지 않았다. 이는 충분한 데이터 축적이 되지 않았고, 머신러닝에 사용된 데이터의 스케일링에 대한 오차로 판단된다.

머신러닝 기법을 사용하여 모델을 구축하고, 태양 추적 시스템을 이용한 실험 결과와 비교하는 연구를 진행하였다. 머신러닝을 이용하여 태양광 발전을 예측하고, 추적형과 고정형 패널의 발전량의 차이를 확인하여 효과적인 태양광 발전에 대한 가능성을 보고자 하는 연구였기에 소규모의 태양광 패널을 사용하였고, 내구성의 문제로 긴 기간의 데이터 축적은 불가능하였다. 따라서, 예측 모델과 결과와의 차이가 존재하였다. 하지만, 태양 추적 패널의 높은 발전량, 그리고 머신러닝 기법을 이용하여 예측한 결과의 경향성이 크게 벗어나지 않음을 확인하였다. 즉, 수십년의 태양 조사량의 데이터를 통한 머신러닝과 함께, 실제 사용되었던 크기의 대형 패널의 사용 및 내구성을 지닌 실험을 통한 데이터 축적을 통한 결과를 획득하면 더 향상된 결과를 기대할 수 있다고 판단한다.

후 기

이 논문은 2021년도 정부(교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임 (No. 2021R1F1A1061409)

참고문헌 (Reference)

- (1) J. Meeus, 1991, *Astronomical Algorithms*, Willmann-Beel Inc.1991
- (2) 신희성, 2010, “태양광발전 시스템의 효율 향상을 위한 태양추적 장치의 원리 및 종류”, RESEAT, pp.4
- (3) Nae Won Kwak, Dong Hoon Lim, 2020, “Financial time series forecasting using AdaBoost-GRU ensemble model”, *The Korean Data Information Science Society*, pp. 268~279.
- (4) Seungmin Lee, Woo Jin Lee, 2016, “Development of a System for Predicting Photovoltaic Power Generation and Detecting Defects Using Machine Learning”, *Korea Information Processing Society*, pp. 353~359.
- (5) Tserenpurev Chuluunsaikhan el, 2021, “Predicting the Power Output of Solar Panels based on Weather and Air

Pollution Features using Machine Learning”, *Journal of Korea Multimedia Society*, pp. 222-232.

- (6) Kim Jeongwon, 2019, “A Solar Power Prediction Scheme Based on Machine Learning Algorithm from Weather Forecasts”, *JKIIT Vol.17 No.9*, pp. 83~88