

인공지능을 기반으로 한 에너지저장장치 최적운영

이지현¹ · 김우현^{2*} · 강태영³ · 박태준⁴

¹전남대학교 기계공학과, 석사과정 & (주)비에이에너지, 선임연구원

²전남대학교 기계공학과, 교수

³(주)비에이에너지, 대표이사

⁴(주)비에이에너지, PM

Optimal Operation of Energy Storage Devices based on Artificial Intelligence

Lee Ji-Hyoun¹ · Kim Woo-Hyun^{2*} · Kang Tae-Young³ · Park Tae-Jun⁴

¹MS Candidate, Department of Mechanical Engineering, Chonnam National University & Junior Researcher, BAEnergy

²Professor, Department of Mechanical Engineering, Chonnam National University

³CEO, BAEnergy

⁴Project Manager, BAEnergy

†Corresponding author: whkim@jnu.ac.kr

Abstract

Energy storage system (ESS) is a power storage device to increase power utilization efficiency by storing generated electricity and supplying power, when needed. As the number of ESS explosion accidents has been rapidly increasing over the past two to three years, a failure diagnosis study on ESS fire safety should be preceded for its continuous use. When the ESS is operated at high temperature for a long time, aging is caused owing to deterioration, and hence, stable operation is required. Among the recent 28 fires, 15 of them occurred during charging and resting. Because rapid temperature and humidity changes occur during charging and discharging, it is important to predict the trend of change. This study predicts ESS operation patterns through an artificial intelligence-based prediction algorithm, and also diagnoses ESS failures through a supervised learning-based confusion matrix. When a failure pattern is found in the ESS, a method to reduce damage through air conditioner control is studied.

Keywords: 에너지저장장치(Energy Storage System), 최적화(Optimization), 딥러닝(Deep Learning), 인공지능(Artificial Intelligence)

1. 서론

최근 친환경 에너지에 대한 관심이 높아지면서 태양광, 풍력 등의 신재생에너지에 의한 발전량이 증가하고 있다. 신재생 에너지는 제한없이 사용할 수 있는 에너지원이라는 이점이 있지만, 간헐적으로 전력을 생산하기 때문에 발전량의 예측 및 제어가 힘들고 일기 및 지형지물에 의해 직접적인 영향을 받는다. 이러한 영향으로 전자기기의 고장 문제가 발생할 수 있으며, 에너지 품질이 낮아 산업체의 가동 중단과 같은 문제가 발생할 수 있다. 이

 OPEN ACCESS



Journal of the Korean Solar Energy Society
Vol.42, No.1, pp.155-175, February 2022
<https://doi.org/10.7836/kSES.2022.42.1.155>

pISSN: 1598-6411

eISSN: 2508-3562

Received: 12 January 2022

Revised: 4 February 2022

Accepted: 5 February 2022

Copyright © Korean Solar Energy Society

This is an Open-Access article distributed under the terms of the Creative Commons Attribution NonCommercial License which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

를 극복하기 위해 에너지를 미리 저장하여 적정 사용 환경에서 사용할 수 있는 에너지 저장 시스템(Energy Storage System, ESS)의 수요가 전 세계적으로 증가하고 있다¹⁾.

앞선 이유로 수요가 증가함에 따라 ESS 내부 상태와 주변 설비 간의 상호 관계, 주변 환경에 의해 화재와 같은 위험이 발생하고 있으며, 화재가 발생하기 전 고장과 같은 사전 증상이 나타난다. 배터리를 적절하지 않은 온습도 환경에서 보관하면 전기화학적 특성의 열화 현상에 영향을 미치므로 고장을 예방하기 위해 ESS룸 내부의 온도와 습도를 적정하게 유지하는 것은 중요하며, 배터리 제조사 기준에 의해 ESS 실내 온도는 $23 \pm 5^\circ\text{C}$ 로 관리하고, 습도는 80% 이하를 유지하는 것이 권고된다²⁾. 산간지방이나 해안가에 설치되는 ESS의 경우 일교차가 커서 결로와 많은 먼지에 노출되어 배터리 모듈에서 결로 현상이 반복적으로 발생하고 건조되면서 셀과 모듈 사이의 접지 부분에서 절연이 파괴되어 화재가 발생할 수 있다^{3,4)}.

작은 ESS 화재로 인해 정부는 ESS 운영 중단을 권고하였고 1622개 사업장 중 32%인 522개 사업장이 ESS 운영을 중단했으며, 최근 2, 3년 사이에 ESS 폭발 사고가 급증하여 지속적인 이용을 위해서는 ESS 내부 온습도 최적화 연구가 선행되어야 한다. 지속적으로 발생한 화재 사고를 방지하기 위해 ESS 운영 최적화 및 고장진단 연구는 필수적이며, ESS 운영 최적화에 적절한 제어방식으로 다양한 종류의 딥러닝 모델을 통해 태양열 전력 예측, 비용 절감을 예측, 배터리 온도 예측을 하여 최적화하는 모델 기반 제어기술이 활용되고 있다. 최근에는 시계열 기반 예측 모델인 RNN (Recurrent Neural Networks), LSTM (Long Short Term Memories) 이외에도 이미지 처리에 강점이 있는 CNN (Convolutional Neural Networks)와 레이어를 혼합한 하이브리드 모델(CNN-LSTM, CNN-GRU, LSTM-GRU)과 예측 성능을 비교한 연구도 진행되고 있다. 1D CNN은 시계열 데이터의 특성을 효과적이고 효율적으로 추출할 수 있으며, LSTM은 시계열 데이터에 기반한 잔존 수명 예측이 가능하도록 한다.

Wu et al. (2011)⁵⁾은 배터리 온도에 의한 배터리 성능에 관한 연구로, 리튬 이온 배터리의 비에너지, 에너지 밀도, 비출력, 충방전 효율, 가격당 에너지 자가 방전량, 내구성정도, 충방전 시 온도 특성을 통해 셀의 온도가 65도를 초과하거나 0도 이하인 경우 리튬 이온 전지의 성능은 급격히 낮아지는 것을 확인하였다.

Ziyu et al. (2015)⁶⁾은 영하 조건에서의 하이브리드 에너지 저장 시스템(Hybrid Energy Storage System, HESS) 열분석에 관한 연구로, 영하의 온도 환경에서 전기차가 배터리 에너지 및 전력 용량 감소와 배터리 성능 저하로 인해 시스템 운영 비용이 증가하여 넓은 온도 범위에서 배터리의 새로운 열화 모델을 바탕으로 운영 비용을 최소화하는 방법을 제시하였다.

Mohammad et al. (2015)⁷⁾은 TES (Thermal Energy Storage) 운영 시스템 최적화에 관한 연구로, 최대 열 에너지 한도를 초과하는 열은 태양광 패널에서 TES로 전달되지 않고 최소 허용 열 에너지 수준 아래로 떨어지는 경우 열이 TES 시스템에서 계통으로 방출되지 않는 시스템을 구축하였다.

Joao et al. (2018)⁸⁾은 에너지 시계열 데이터의 이상 감지를 위해 오토 인코더를 사용하여 비정상사이트를 감지했다. 딥러닝 모델은 양방향 장기 단기 메모리 네트워크인 양방향 재귀 신경 네트워크를 사용했으며 인공신경망으로 인코더와 디코더를 적용한 Seq2Seq 프레임 워크를 적용했다. 태양광 발전량의 패턴을 차원을 축소시

키는 주성분 분석 및 t-분산 확률 분석 기법을 통해 분석하였다.

Taewook et al. (2019)⁹⁾은 동일한 데이터의 서로 다른 표현에서 학습한 특징을 결합하여 주가를 예측하는 LSTM-CNN 모델을 제안하였다. 시간적 특징을 추출하는 데 활용되는 LSTM 모델과 영상의 특징을 추출하는 데 활용되는 CNN 모델을 결합하여 동일한 데이터의 시간적 특징과 영상 특징을 함께 사용함으로써 예측 효과를 효율적으로 감소시켰다.

Loannis et al. (2020)¹⁰⁾은 시계열 데이터의 움직임을 정확하게 예측하기 위한 CNN-LSTM 모델을 제안하였다. 유용한 지식을 추출하고 시계열 데이터의 내부 표현을 학습하기 위한 CNN 계층의 기능과 단기 및 장기 종속성을 식별하기 위한 LSTM 계층의 효율성을 활용하여 예측 성능을 크게 향상시켰다.

Jin et al. (2020)¹¹⁾은 문장 내의 지역적인 정보와 문장 간의 장거리 종속성을 기반으로 이전 연구에서 제시된 회귀 및 기존 신경망 방법을 능가하는 CNN-LSTM 모델을 제시하였다. 기존 CNN과 달리 제안된 CNN-LSTM 모델은 개별 문장을 영역으로 사용하여 입력 텍스트를 여러 영역으로 분할하여 각 영역의 유용한 감성 정보를 추출하고 가중치를 부여할 수 있다. 따라서 기존 연구에서 제안한 어휘 기반, 회귀 기반, 신경망 기반 방법보다 우수한 성능을 보였다.

기존 ESS의 EMS (Energy Management System)은 에너지 소비 현황에 대한 모니터링 위주로 구성되어 있으며, IoT 기기를 기반으로 저장 시스템 내 시스템의 운전 및 성능 정보, 에너지사용량 등을 데이터로 수집한 후 단순한 규칙기반 제어를 적용하는 기술이 부분적으로 이루어지고 있으나 배터리 셀의 온도만을 추정하는 등 실질적인 에너지 최적화 제어가 체계적으로 되고 있지 않다. 따라서 ESS 온습도예측을 위한 상태 정보의 수집, 처리 기술, 분석 및 상황판단 기술이 필요하며, 최종 판단된 각 배터리셀들의 상태와 목표에 따른 최적 제어 방향을 결정하는 기술이 필요하다. 따라서 본 연구에서는 딥러닝 예측모델을 통해 ESS의 온도를 예측하고, 하이퍼파라미터 최적화를 통해 충방전 동안에 변화하는 온도를 높은 정확도로 예측하는 모델을 제작하여 ESS에서 발생할 수 있는 위험요소를 사전에 대응하고자 한다.

2. ESS 개요 및 실증 테스트베드 구성

본 장에서는 ESS의 개요와 실증 테스트베드에서 얻을 수 있는 데이터 개요를 설명한다.

2.1 ESS 개요

ESS는 심야시간의 잉여 전력을 저장하여 피크시간대 또는 정전 시 비상전원으로 활용하여 전력 부하를 조절하기 위한 장치이다. 태양광이나 풍력 등의 신재생에너지는 날씨에 따라 꾸준하게 전력을 얻을 수 없다는 단점이 있기 때문에 이를 해결하기 위해 에너지를 저장했다가 피크시간대에 방전하여 발전 설비에 대한 과잉 투자를 막을 수 있다. ESS는 지속적이고 유연한 전기 공급으로 인해 전력 공급이 부족할 때 전력을 공급하기 위한 수단으로 가능하며, 비상발전기에 비해 유지 및 관리가 쉽고 저장시설이 전력계통과 연결되어 있어 유사시 활용

성이 뛰어나다. 또한 전력망 연결이 어려운 지역에 신재생 에너지 시스템과 연결한 에너지 저장장치를 이용하여 전력을 공급할 수 있게 된다.

2.2 실증 테스트베드 개요

실증 사이트 건물의 개요는 Fig. 1과 같다. 실증 건물은 제주시 한림읍에 위치하고 있으며, 배터리 사양은 626 kWh이다. 본 사이트에는 총 4개의 랙과 두개의 공조장치, 팬 하나가 장착되어 있으며 139 kW로 전력을 송신한다. BMS에서 확인할 수 있는 데이터는 Table 1과 같다. 배터리의 발전량, 충전시 사용 전력, SOC 및 배터리의 전압 전류를 확인할 수 있다. BMS (Battery Management System) 데이터는 15분마다 측정되며 예측하고자 하는 ESS 랙의 온습도 데이터를 취득하기 위해 무선 온습도 센서를 랙의 상, 하단에 Fig. 2와 같이 각각 부착하였다.

본 건물에 부착된 센서는 Radionode 사에서 구매한 USB Temp/RH Sensor를 사용하였으며, 온도센서의 측정범위는 $-40 \sim 80^{\circ}\text{C}$, 습도센서의 측정범위는 $5 \sim 80\%$ 이다. 온도센서의 정확도는 $\pm 0.3^{\circ}\text{C}$ 이며 습도센서의 정확도는 $\pm 2.0\%$ 이다. 각 센서마다 트랜스미터 및 데이터로거를 통해 데이터를 수집 및 전송하는 기능이 있다. RS485통신으로 Power of Ethernet (PoE)를 사용하여 실시간으로 충전을 하며, 모니터링 장치가 구축되어 센서 정보를 10초마다 게더링 할 수 있다. 온습도 센서의 부착 위치 개요는 Fig. 2와 같다.

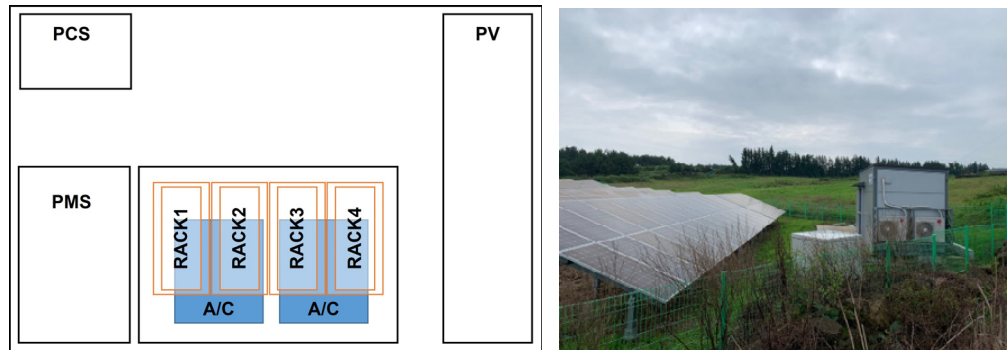


Fig. 1 Test bed overview

Table 1 BMS Data

Battery Management System Data		
1	Power	Power
2	Cum Power	Accumulated Power
3	Charge Power	Charge Power
4	Discharge Power	Discharge Power
5	Cum Charge Power	Accumulated Charge Power
6	Cum Discharge Power	Accumulated Discharge Power
7	SoC	State Of Charge
8	V_bat	Battery Voltage
9	I_bat	Battery Current

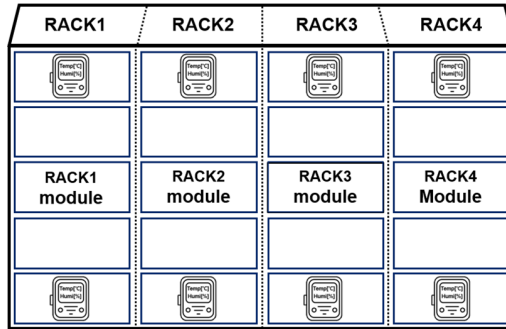


Fig. 2 Test bed Sensor overview

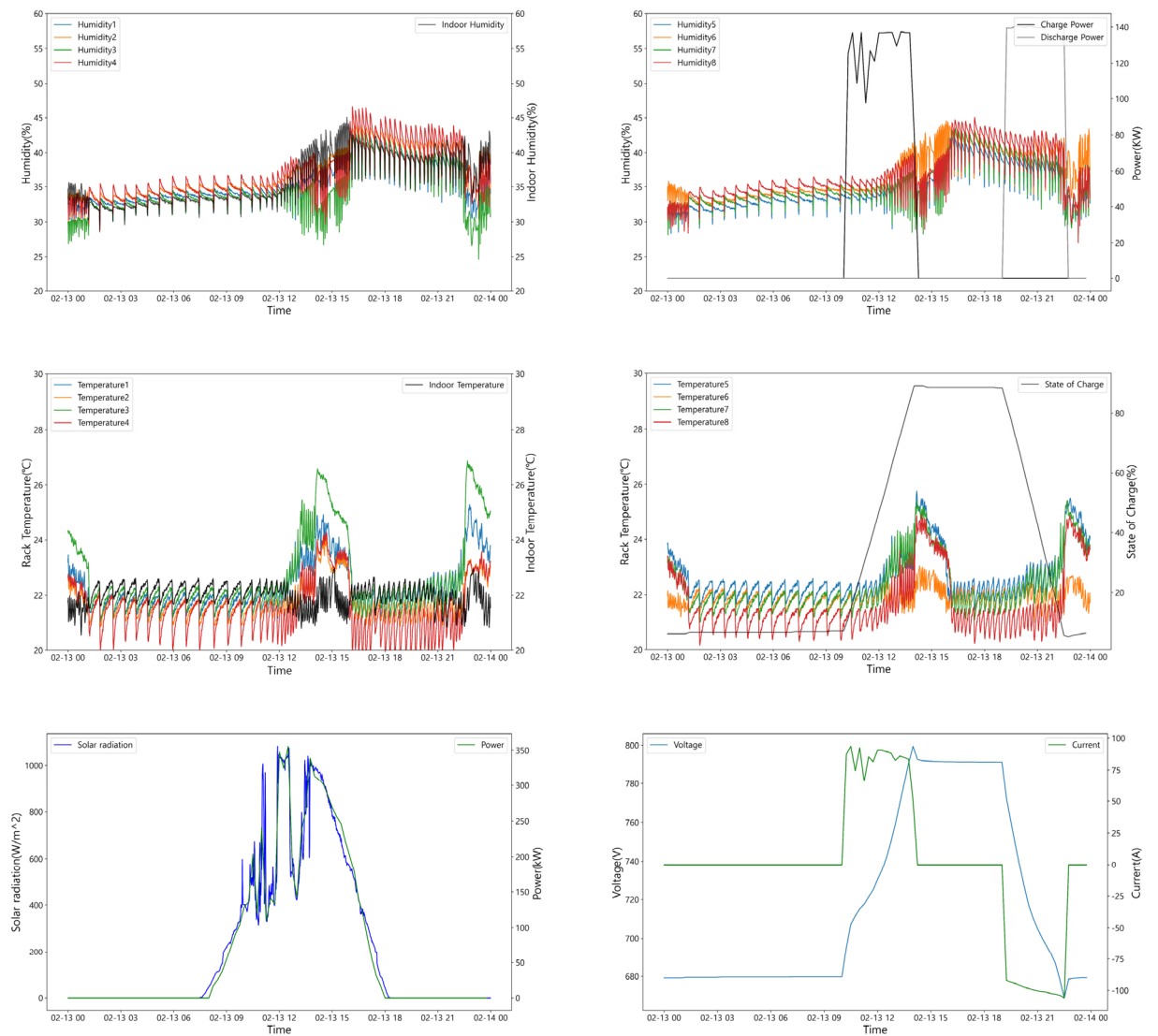


Fig. 3 Jeju data Plot

Fig. 3은 제주 실증사이트의 2021년 2월 13일의 전압, 전류, 일사량, 전압 값을 충방전에 따라 나타낸 그래프이다. 충전시간은 오전 10시 15분부터 오후 14시까지이며, 방전시간은 오후 19시 15분부터 23시까지이다. ESS가 휴지 중일 때는 실내 22 ~ 25°C로 맞추어진 실내 조건에 따라 온도와 습도 값이 평이하게 움직이며, 충방전 직후에 온도 값이 크게 상승하는 양상을 보인다. 최고온도는 26.86°C이며, 최고습도는 46.03%이다. 상대적으로 상단부에 위치한 온도 센서의 값이 더 높게 찍히는 것을 확인할 수 있다.

3. 딥러닝 기반 ESS 운영 최적화

본 장에서는 데이터 전처리 및 딥러닝 모델 구성에 대해 설명한다. Python의 Pandas library를 사용하여 데이터 전처리를 진행하였다.

3.1 데이터 전처리 개요

데이터 전처리의 과정은 Fig. 4와 같이 크게 데이터 통합, 데이터 정제, 데이터 레이블링의 순서로 이루어진다. 먼저, 데이터 통합(Data Integration) 과정에서는 각각의 사이트에서 측정된 데이터를 필요한 형태의 데이터로 파싱 및 추출, 통합하고 데이터의 중복, 단위 등을 확인 및 변환하여 하나의 빅데이터로 통합한다. 그 다음, 데이터 정제(Data Cleansing) 과정에서는 데이터의 오측 값, 결측값, 이상치 등을 제거 및 보간을 통한 재구성 과정을 통해 데이터를 정제한다. 마지막으로, 데이터 레이블링(Data Labeling) 과정에서는 Train data, Validation data, Test data의 기간 설정 후에 예측을 위해 필요한 데이터를 추출한다. 위의 과정은 프로그래밍 언어인 Python을 사용하였고, Pandas, Numpy 와 같은 Python library를 활용하여 진행했다.

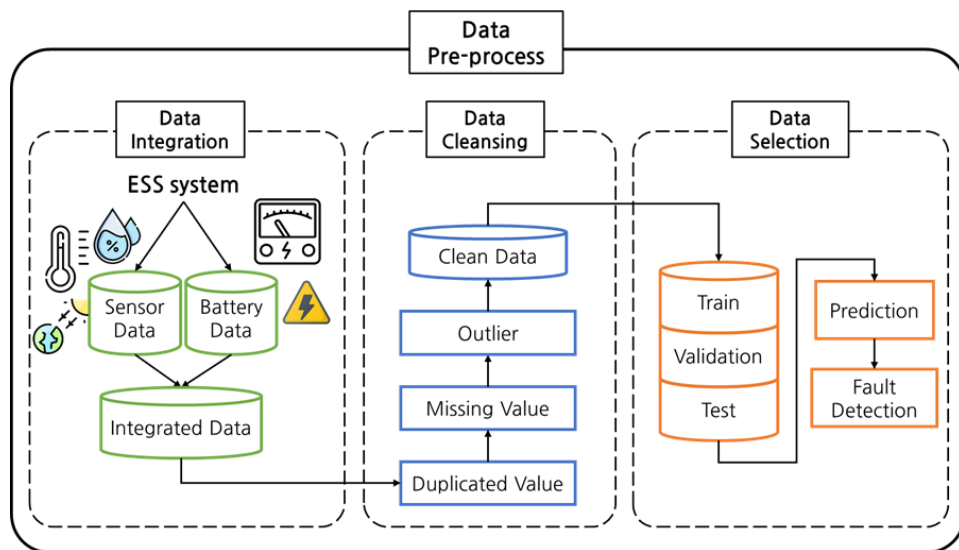


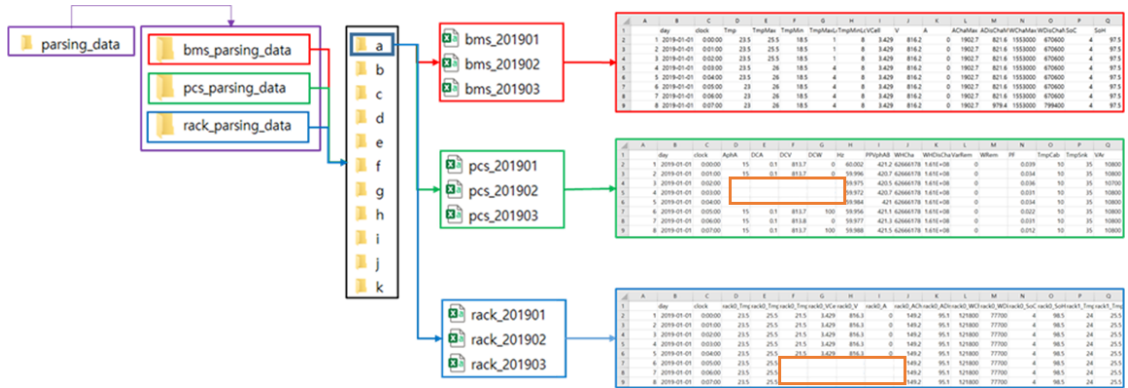
Fig. 4 Data Preprocessing Flow Chart

데이터 통합 단계에서 대상 ESS 데이터는 1개 사이트의 데이터를 사용하였으며, PCS 139 kW, Battery 626 kW 용량을 가진 ESS 데이터이다. 총 데이터는 2020년 8월부터 2021년 9월까지의 데이터이며, 온습도 센서 데이터는 5 ~ 10초 간격으로 측정되었고, 전기적 특성을 가진 배터리 데이터는 15분 간격으로 측정되었다. 파일의 형태는 db 형식으로 되어 있으며, 딥러닝 모델 입력 데이터로 활용하기 위해 csv 파일 형식으로 변환하였다. 데이터는 센서 데이터와 배터리 데이터로 분할 추출하였으며, 센서 데이터와 배터리 데이터는 통합 후에 분석하였다. 센서 데이터는 온도, 습도, 일사량 등 실증사이트를 구축할 때 외부센서를 부착하여 취득한 데이터이며, 배터리 데이터는 BMS (Battery Management System)상에서 읽어올 수 있는 배터리 전압, 배터리 전류, SOC 등 전기적인 특성을 갖는 데이터이다. 자체 구축된 서버를 통해서 취득하기 때문에 센서마다 측정 주기가 달라 데이터 전처리 과정에서 일괄적으로 10초 주기로 통합한 후에 배터리 데이터와 통합 시에 15분 주기로 맞춰 주었다.

데이터 정제 단계에서 통합된 ESS데이터의 결측값을 제거 또는 대체하거나, 이상 값을 감지하여 처리하는 등 일련의 과정을 통하여 데이터의 신뢰도를 향상시킨다. 결측값이란 데이터 수집 단계에서 점검, 고장 등의 이유로 데이터가 수집되지 않은 값을 의미한다. 결측값이 있는 상태로 딥러닝 모델에 적용하게 될 경우 변수간의 관계가 왜곡될 수 있기 때문에 모델의 정확성이 떨어지고 결과가 왜곡될 수 있다. 결측값 처리의 방법에는 결측값을 제거하는 방법, 다른 값으로 결측값을 대체하는 방법, 예측 값을 삽입하는 방법 등이 있다. 결측값을 제거하는 방법은 결측값이 발생한 시간의 전체 데이터를 삭제 또는 결측이 발생한 측정값을 모두 삭제하는 방법이다. 결측값이 발생한 시간의 전체 데이터를 제거했을 경우, 시계열 데이터 활용에 있어 주기성과 계절성 등의 추세를 이용하는데 제한사항이 있을 수 있다. 결측이 발생한 측정값을 모두 삭제했을 때, ESS 안전진단을 위한 특성 예측에 영향을 주는 데이터일 경우 예측의 정확도가 떨어지거나 왜곡된 모델이 생성될 수 있다. 결측값 대체의 방법은 결측값이 발생한 데이터를 전체 데이터의 평균값, 최빈값, 중간값과 같은 통계적 수치 또는 결측이 일어난 시간의 이전 혹은 이후 시간의 데이터로 대체하는 방법이다. 모든 결측값을 평균값 등으로 대체하는 일괄 대체 방법과 범주형 데이터를 이용해 유사한 유형의 평균값 등으로 대체하는 유사 유형 대체 방법이 있다. 데이터 정제하는 과정은 Fig. 5와 같이 각각의 데이터를 통합하면서 결측값을 대체하는 방법으로 진행하였으며, 한 시간 이상 데이터가 취득되지 않은 기간은 제외하였다.

Fig. 6에서 A와 B는 화재가 발생하지 않은 ESS 현장의 데이터이며, F는 화재가 발생한 ESS 현장의 데이터이다. 데이터 정제 전에는 Fig. 6의 왼쪽 그림처럼 결측값이나 중복값으로 인해 화재 발생 여부에 따라 데이터를 구분하기 어려우나, 데이터 정제 후에는 Fig. 6의 오른쪽 그림처럼 F에서 3월 14일 경 비정상적으로 온도가 내려감을 알 수 있다.

이상치란 기존데이터의 분포에 크게 벗어나는 데이터 또는 비현실적인 데이터로 모델을 왜곡할 가능성이 많은 데이터이다. 비현실적인 데이터는 물리적으로 설명할 수 없는 데이터를 지칭하는 말로 예를 들어, 전력사용량이 음수로 계측된 경우와 같은 데이터를 지칭하는 것이다. 온도나 습도의 값이 문자열로 취득되거나, 여름철 데이터에서 음수가 측정되는 등 비현실적인 데이터를 제거하였다.



Frame_a.csv

[Time]	[BMS]						[PCS]					[RACK]				
	A	B	C	D	E	F	P	Q	R	S	T	AC	AD	AE	AF	AG
1	Time	Tmp	TmpMax	TmpMin	TmpMaxL	TmpMinL	AphA	DCA	DCV	DCW	Hz	rack0_Tmp	rack0_TmpC	rack0_Tmp	rack0_Vc	rack0_V
541839	2019-01-12 6:38	22	24.5	17	3	8	15.1	0.1	814	100	59.983	22	23	18.5	3.431	816.6
541840	2019-01-12 6:39	22	24.5	17	3	8	15.1	0.1	814	100	59.999	22	23	18.5	3.431	816.6
541841	2019-01-12 6:40	22	25	17	3	8	15.2	0.1	814	100	60.015	22	23	18.5	3.431	816.6
541842	2019-01-12 6:41	22	25	17	3	8	15.1	0.1	814	100	60.026	22	23	18.5	3.431	816.6
541843	2019-01-12 6:42	22	25	17	3	8	15.1	0.1	814	100	59.942	22	23	18.5	3.431	816.6
541844	2019-01-12 6:43	22	25	17	3	8	15.1	0.1	814	100	59.992	22	23	18.5	3.431	816.6
541845	2019-01-12 6:44	22	25	17	3	8	15.1	0.1	814	100	60.03	22	23	18.5	3.431	816.6
541846	2019-01-12 6:45	22	25	17	3	8	15.1	0.1	814	100	59.998	22	23	18.5	3.431	816.5
541847	2019-01-12 6:46	22	25	17	3	8	15.1	0.1	814	100	59.962	22	23	18.5	3.431	816.6
541848	2019-01-12 6:47	22	25	17	3	8	15.2	0.1	814	100	60.013	22	23	18.5	3.431	816.5
541849	2019-01-12 6:48	22	25	17	3	8	15.1	0.1	814	100	59.988	22	23	18.5	3.431	816.5
541850	2019-01-12 6:49	22	25	17	3	8	15	0.1	814	100	59.997	22	23	18.5	3.43	816.5
541851	2019-01-12 6:50	22	25	17	3	8	14.9	0.1	814	100	60.018	22	23	18.5	3.431	816.6
541852	2019-01-12 6:51	22	25	17	3	8	15.1	0.1	814	100	59.95	22	23	18.5	3.431	816.5
541853	2019-01-12 6:52	22	25	17	3	8	15.1	0.1	814	100	60.055	22	23	18.5	3.431	816.6
541854	2019-01-12 6:53	22	25	17	3	8	15.1	0.1	814	100	60.023	22	23	18.5	3.431	816.5
541855	2019-01-12 6:54	22	25	17	3	8	15.1	0.1	814	100	60.013	22	23	18.5	3.431	816.6
541856	2019-01-12 6:55	22	25	17	3	8	15.1	0.1	814	100	60.011	22	23	18.5	3.43	816.5
541857	2019-01-12 6:56	21.5	25	17	3	8	15.1	0.1	814	100	59.965	22	23	18.5	3.431	816.5
541858	2019-01-12 6:57	21.5	25	17	3	8	15.2	0.1	814	100	59.985	22	23	18.5	3.431	816.5
541859	2019-01-12 6:58	21.5	25	17	3	8	15.1	0.1	813.9	100	59.951	22	23	18.5	3.431	816.5

Fig. 5 Data Cleansing Process

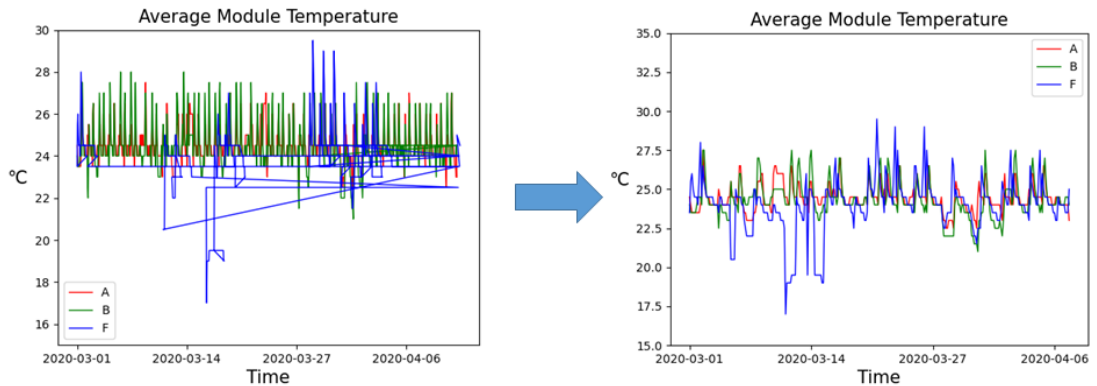


Fig. 6 Before Data Cleansing (Left), After Data Cleansing (Right)

3.2 딥러닝 개요

딥러닝은 머신러닝의 한 부류로 인공지능망 구조를 활용해 주어진 입력데이터를 받아 정해진 수식을 반복 계산하여 출력데이터를 출력하는 기술이며, 딥러닝의 핵심은 인공지능망이다. Fig. 7은 간단한 딥러닝 구조로 뉴런(노드)를 여러 개 결합하여 신경망을 구성하며, 입력층(Input layer), 은닉층(Hidden layer), 출력층(Output

layer)의 3층 계층구조를 가진다. 모델변수 행렬인 는 난수로 초기화되고, 입력이 최종 출력까지 전달되는 전방 입력과정을 거쳐 비용 함수(cost function)로 오류와 실제 출력, 기대출력과 차를 구한다. 역전파(back propagation)과정에서 출력층에서 구한 기대출력과 실제 출력간의 차를 반대방향으로 전파시키면서 뉴런(노드)의 가중치와 편차를 반복 갱신하는 작업을 통해 최적화된 결과를 도출해낸다.

활성화 함수는 입력을 받아 활성화, 비활성을 결정하는데 사용되는 함수이며, 신호를 입력받아 적절한 형태로 출력해주는 함수로 입력과 가중치의 곱의 합인 가중합이 임계치보다 큰 경우, 출력을 활성화하고, 그 반대의 경우 비활성화하는 형태의 함수이다. 활성화 함수를 통해 딥러닝 모델은 필요한 정보는 기억하고 필요 없는 정보는 잊어버리는 학습을 할 수 있게 된다. 대표적인 활성화 함수로는 시그모이드(sigmoid)함수, 쌍곡탄젠트(tanh) 함수, 정류 선형 유닛(ReLU)함수 등이 있다. 제한변수는 과적합을 피하게 하며, 학습 과정 동안 예측 값과 목적 값의 오차를 줄이는 방향으로 가중치가 지속적으로 변경되며 주어진 과정 동안 반복을 통해 인공신경망의 가중치와 편차가 안정화된다.

$$a^2 = g(\theta^1 x) \tag{1}$$

$$h_\theta(x) = a^3 = g(\theta^2 a^2) \tag{2}$$

$$J(\theta) = \frac{1}{2m} \left[\sum_{i=1}^m (h_\theta(x^i) - y^i)^2 + \lambda \sum_{i=1}^{L-1} \sum_{j=1}^n \theta_{ij}^2 \right] \tag{3}$$

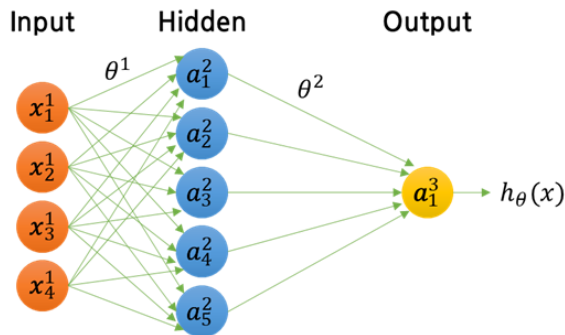


Fig. 7 Deep Learning Structure

3.3 특징 추출 개요

본 절에서는 데이터의 각 항목들이 어떤 인과관계를 가지고 있는지 분석하여 모델의 성능 향상을 도모한다. 항목이 너무 많으면 모델의 과적합 위험과, 모델이 이상적인 방향으로 학습을 하지 않을 위험이 있으므로 특징 추출을 진행한다. 세 가지 상관분석 기법 중 가장 적합한 기법을 적용하여 진행하였다. 피어슨 상관분석은 가장

대표적으로 사용되는 특징 추출로, 공분산을 각 확률변수의 표준편차의 곱으로 나눠준 값이다. $-1 \sim 1$ 사이의 값을 가지며 절댓값 1에 가까워질수록 두 변수의 값들은 직선 가까이에 위치하여 선형적인 상관성이 커지고, 0 일 때는 선형적인 상관성이 없다. 변수간의 정규분포가 선형성을 가지는 정규성 가정을 만족해야 적용할 수 있다는 특징이 있다. 켄달 상관분석은 두 변수들간의 순위를 비교하여 연관성을 계산하는 방법이며, 샘플사이즈가 적거나 데이터의 동률이 많을 때 유용하다. 스피어만 상관분석은 값에 순위를 매겨 그 순위에 대한 상관계수를 구하는 방법으로, 켄달 상관분석과 유사하지만 데이터 내 편차와 에러에 민감하다는 특징이 있다. 피어슨 상관분석 사용에 앞서 정규성 가정이 만족되는지를 확인하기 위해 Q-Q Plot을 그려 시각적으로 확인해 보았다.

Fig. 8은 Q-Q Plot의 결과이다. 대각선 참조선을 따라서 값들이 분포하게 되면 정규성을 만족한다고 볼 수 있으며, 한쪽으로 치우치는 모습이라면 정규성 가정에 위배되었다고 볼 수 있다. 온도-온도와 습도-습도 그래프는 대각선 참조선을 따라 분포하는 반면, 전기적인 데이터와 혼합되어 있는 온도-SOC, 온도-전력과 같은 데이터는 왼쪽 상단으로 값들이 치우쳐져 있다. 따라서 피어슨 상관분석은 사용할 수 없다고 판단하였고, 켄달 상관분석은 값들이 적을 때 사용하는 방법이기 때문에 특징 추출 방법으로는 스피어만 상관분석을 선정하였다.

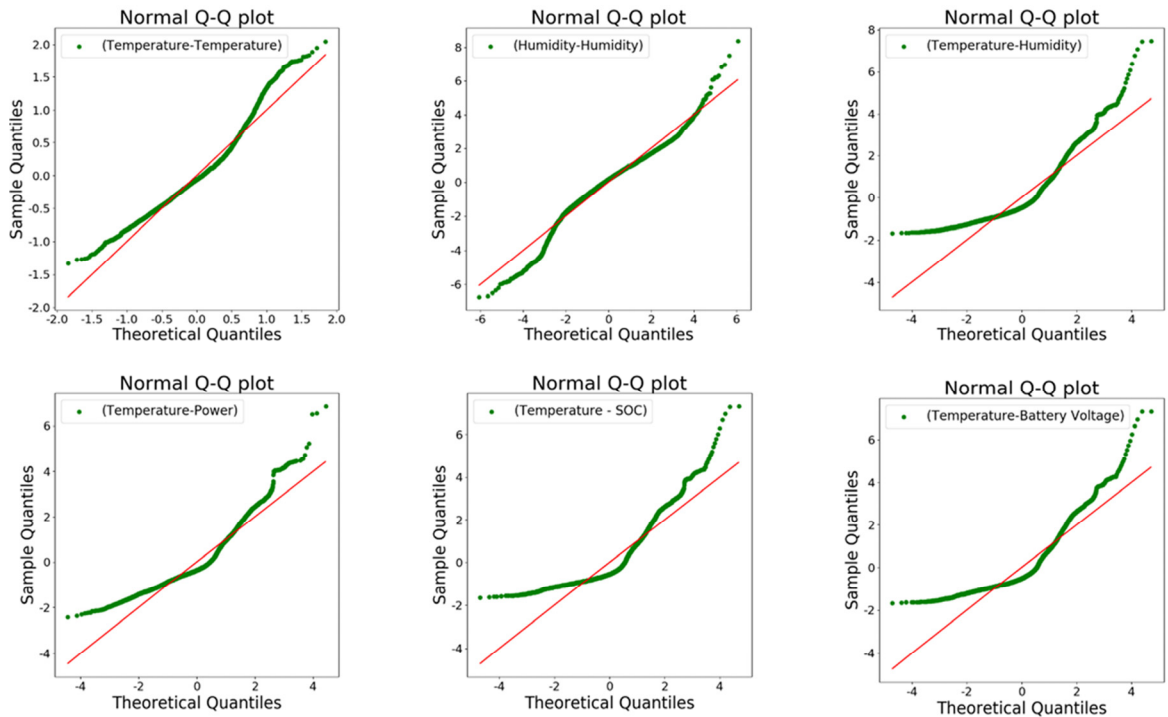


Fig. 8 Normality Test

스피어만 상관분석의 결과는 데이터들의 배열을 가시화하기 위해 Heatmap을 사용해 Fig. 9와 같이 시각화하였으며 상관계수는 Table 2에 정리하였다. Fig. 9에서 빨간색으로 표시된 부분은 온도(Tmp), 습도(Hmd), 배터리 데이터들을 1대1로 상관분석한 부분이다. 초록색에 가까울수록 양의 상관관계가 높으며, 분홍색에 가까울

수록 음의 상관관계가 높다. 온도, 습도, 배터리 데이터들은 각자의 카테고리 내에서 상관관계가 높음을 확인할 수 있으며, 온습도에 영향을 미치는 전체적인 인자를 확인해 보았을 때 배터리 인자보다 온습도 센서 자체의 값들이 주요한 인자로 작용함을 확인할 수 있다. 본 연구에서는 제주 실증사이트에서 예측할 주요 파라미터인 온도와 습도를 주요 인자로 두고 진행하였다.

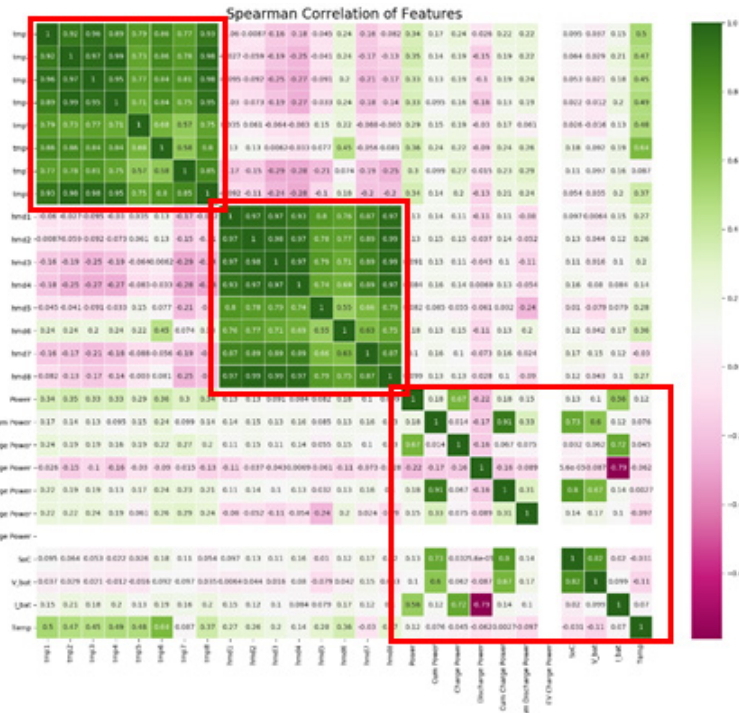


Fig. 9 Heat map for the Pearson correlation

Table 2 Coefficient of the Pearson correlation

No	Parameters	Coefficient	No	Parameters	Coefficient
1	Power	0.34	14	Hmd4	0.18
2	Cum Power	0.17	15	Hmd5	0.045
3	Charge Power	0.24	16	Hmd6	0.24
4	DischargePower	-0.026	17	Hmd7	0.316
5	Cum Charge Power	0.22	18	Hmd8	0.082
6	Cum Discharge Power	0.22	19	Tmp1	1.00
7	SoC	0.095	20	Tmp2	0.92
8	V_bat	0.037	21	Tmp3	0.96
9	I_bat	0.15	22	Tmp4	0.89
10	Temperature	0.5	23	Tmp5	0.79
11	Hmd1	0.06	24	Tmp6	0.86
12	Hmd2	0.0087	25	Tmp7	0.77
13	Hmd3	0.16	26	Tmp8	0.93

3.4 주기성

본 연구에서는 배터리 특성상 충방전의 영향을 많이 받기 때문에 시간에 따른 특징 추출이 필수적이다. 일반적인 ESS는 10시 ~ 15시에 충전을 하고 18시 ~ 23시에 방전을 한다. 이 때 온도, 전압, 충전량 등 BMS 데이터에 변화가 일어난다.

본 연구에서 사용한 데이터에는 각 관측치에 대한 날짜 및 시간 정보가 Datetime 형태로 포함되어 있지만 데이터를 정렬하기 위한 인덱스로만 사용된다. 따라서 요일(월요일 ~ 일요일)과 시간대(23시)에 대한 정보를 데이터에 별도의 입력 특징들로 추가하였다. 시간은 연속적이며 일정한 주기를 가지고 있는 데이터이기 때문에 $\frac{2\pi}{x}$ 에 x 를 month는 12, week은 52, hour은 24, minute은 60으로 표기한다. Table 3은 삼각함수를 사용하여 시간을 전처리한 예시이다.

Table 3 Time processing using trigonometric functions

Timestamp	week_sin	week_cos	hour_sin	hour_cos	minute_sin	minute_cos
2021-02-01 01:00:00	-0.46	0.89	0.26	0.97	0	1
2021-02-01 02:10:00	-0.46	0.89	0.5	0.87	0.87	0.5
2021-02-01 03:20:00	-0.46	0.89	0.71	0.71	0.87	-0.5

4. 딥러닝 예측 모델 비교

본 장에서는 각 Basic model인 RNN (Recurrent Neural Networks), LSTM (Long Short Term Memory), CNN (Convolutional Neural Networks), GRU (Gated Recurrent Unit) 모델과 Hybrid model인 CNN-GRU, CNN-LSTM, LSTM-CNN 예측 모델 성능을 비교하고, 각 결과에 대해 설명한다. Python의 Keras 모듈 기반으로 제작한 모델을 사용하여 딥러닝 기반 온습도 예측을 진행하였다.

$$y_t = W_{hy}h_t + b_y \tag{4}$$

$$h_t = \tanh(W_{hh}h_{t-1} + W_{xh}x_t + b_h) \tag{5}$$

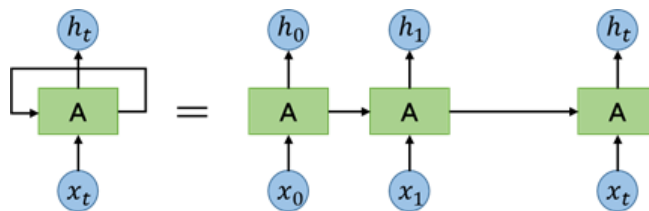


Fig. 10 RNN Model Structure

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{6}$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{7}$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \tag{8}$$

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \tag{9}$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{10}$$

$$h_t = o_t * \tanh(C_t) \tag{11}$$

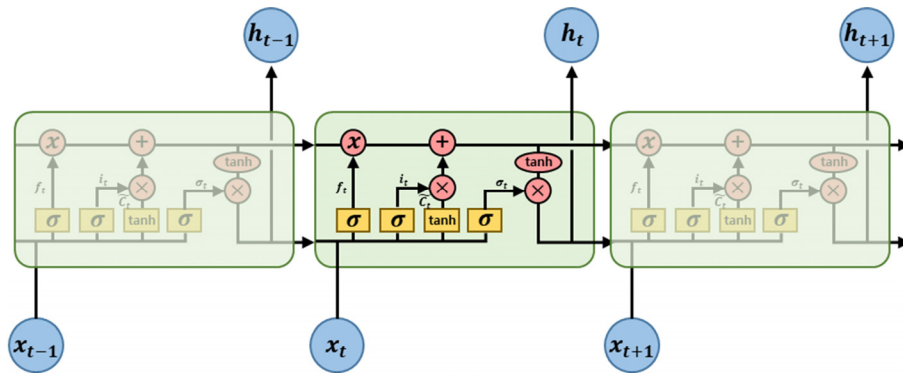


Fig. 11 LSTM Model Structure

$$y[i, j] = \sum_{n=-INF}^{\infty} \sum_{m=-INF}^{\infty} x[m-i, n-j]w[m, n] \tag{12}$$

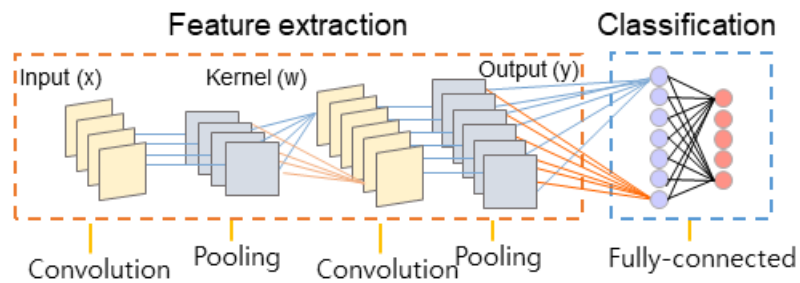


Fig. 12 CNN Model Structure

$$r_t = \sigma(W_{xr}x_t + W_{hr}h_{t-1} + b_r) \tag{13}$$

$$z_t = \sigma(W_{xz}x_t + W_{hz}h_{t-1} + b_z) \tag{14}$$

$$g_t = \tanh(W_{hg}(r_t \circ h_{t-1}) + W_{xg}x_t + b_g) \tag{15}$$

$$h_t = (1 - z_t) \circ g_t + z_t \circ h_{t-1} \tag{16}$$

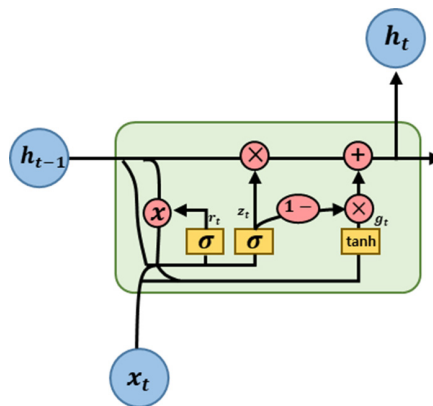


Fig. 13 GRU Model Structure

Figs. 10 ~ 13은 각각의 딥러닝 모델 구조를 도식화한 그림이다. RNN 모델은 시계열 데이터에 적합한 모델로 모든 노드가 방향을 가진 엣지로 연결되어 순환구조를 이루는 인공신경망이다. 이전 시간에 따른 가중치를 부여받아 Hidden State인 h 에 따라 출력값을 예측한다. LSTM 모델은 Cell state와 Hidden state를 통해 어느 시점에서 정보를 버리거나 유지하여 선택적으로 흘러갈 수 있도록 거시적으로 과거 데이터를 고려해서 출력값을 예측하는 모델로, RNN 모델의 Vanishing Gradient 문제를 해결하기 위해 고안되었다. CNN 모델은 이미지 데이터에 특화된 모델로, 이미지의 공간 정보를 유지하면서 인접 이미지와의 특징을 효과적으로 인식할 수 있다. 이미지의 특징을 추출하는 부분과 클래스를 분류하는 부분으로 나뉜다. GRU 모델은 기존 LSTM 모델의 Cell State를 Hidden State에 합친 모델로, LSTM의 게이트를 단순화하여 구현하였기 때문에 LSTM보다 연산이 약간 더 빠를 수 있다는 장점이 있다. 각각의 모델 구조는 Table 4와 같다. Basic Model은 각각의 Layer를 여러 개 쌓아서 구현하였으며, Hybrid Model은 Basic Model의 Layer를 순차적으로 쌓아서 구현하였다. Batch size와 Dropout은 범위 설정 후에 Grid Search로 최적의 Hyperparameter를 선정하였다.

Table 4 Deep Learning Model Structures

Model	CNN	GRU	LSTM	RNN	CNN-GRU	CNN-LSTM	LSTM-CNN
Batch size (64 ~ 512)	128	64	64	128	512	128	64
# of layers	2 Conv layers	3 GRU layers	3 LSTM layers	3 RNN layers	1 Conv layers	2 Conv layers	2 LSTM layers
# of cells	128 / First layer 64 Second layer	64 / Single layer	64 / Single layer	64 / Single layer	20 / Single layer	128 / First layer 64 / Second layer	64 / Single layer
# of layers	2 Dense layers	-	-	-	2 GRU layers	2 LSTM layers	2 CNN layers
# of cells	100 / First layer 50 / Second layer	-	-	-	20 / Single layer	64 / Single layer	128 / First layer 64 / Second layer
# of layers	1 Dense layer	1 Dense layer	1 Dense layers	1 Dense layer	1 Dense layer	1 Dense layer	1 Dense layer
# of cells	1 / Single layer	1 / Single layer	1 / Single layer	1 / Single layer	1 / Single layer	1 / Single layer	1 / Single layer
Dropout (0.0 ~ 0.9)	0.3	0.3	0.1	0.0	0.1	0.1	0.0

4.1 온도 예측 모델

온도 모델의 Input length는 1시간으로 4개이며 Output length는 1시간으로 4개이며, 2021년 5월 1일부터 2021년 6월 30일의 데이터를 사용하였다. Train date 31일, Validation date 15일, Test date 15일이다. 입력 데이터로는 Power (발전량), hr_sin (시간 사인 변환), hr_cos (시간 코사인 변환), tmp2 (Rack1 하단온도), tmp4 (Rack2 하단온도), tmp8 (Rack4 하단온도)을 사용하였다. Grid Search를 통해 Table 5의 Hyperparameters 중 최적의 조건을 탐색하였다.

Table 5 Hyperparameters and Values

Hyperparameters	Values
Activation function	relu, elu, tanh
Epoch	1000
Batch size	32, 64, 128, 256, 512
Learning rate	0.001, 0.005, 0.01, 0.05
Kernel initializer	glorot uniform, he uniform
Dropout	0 ~ 1
Optimizer	Adam

Fig. 14는 2021년 6월 16일부터 6월 30일까지의 전체적인 예측 결과이며, Table 6은 각각의 모델의 성능을 비교한 결과이다. 전체 예측의 경우 각 모델별로 성능 차이는 크게 발생하기 않았다. 6월 21일 오후 13시 30분

경 온도가 29.576도까지 상승하였으며, 이는 배터리 제조사에서 권장하는 온도 범위를 벗어나는 범위이기에, 해당 피크점을 예측하는 것은 중요하다. Table 7은 피크점에서의 모델별 예측 성능 비교 테이블이다. Fig. 15는 피크 발생일의 Basic Model과 Hybrid Model의 결과이다. Basic Model에서는 GRU Model이 가장 좋은 성능을 보였으며, Hybrid Model에서는 LSTM-CNN 모델이 가장 좋은 성능을 보였다. GRU 모델과 LSTM-CNN 모델의 Hyperparameter는 Table 8과 같다. 단일 모델보다 Hybrid Model이 예측 성능이 우수하였다. 데이터 내의 이미지 정보와 장거리 종속성을 기반으로 시간 및 이미지 특성의 조합을 사용하여 기존에 제시된 Basic Model의 성능보다 뛰어난 성능을 보였다. Fig. 16은 피크 발생일의 GRU와 LSTM-CNN 모델의 결과 그래프이며, GRU는 예측값이 피크점에 못미치는 반면, LSTM-CNN은 피크 구간에도 실제값을 따라갔다. 이는 LSTM-CNN 예측 모델을 통해 ESS 실내 온도가 급격히 상승할 때 위험 관리가 가능함을 나타낸다.

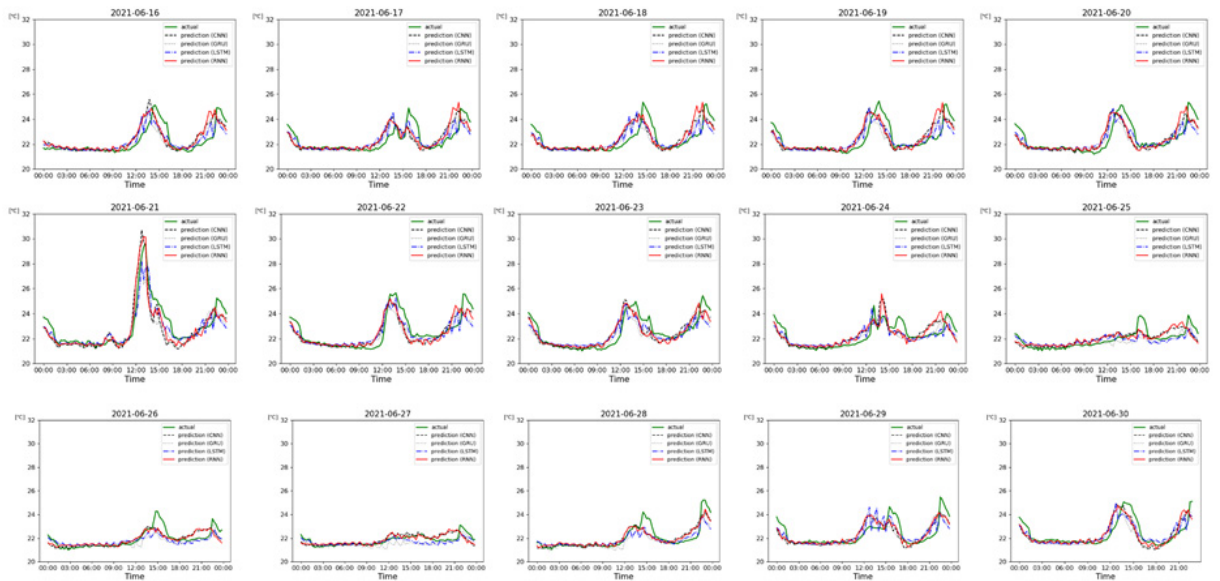


Fig. 14 Temperature Predict Model Results (Daily)

Table 6 Temperature Model Predict Results

Model	RMSE	MAE	Cv (RMSE)
CNN	0.41	0.46	1.84
GRU	0.39	0.418	1.75
LSTM	0.401	0.401	1.8
RNN	0.39	0.418	1.75
CNN-GRU	0.468	0.45	2.1
CNN-LSTM	0.43	0.47	1.93
LSTM-CNN	0.391	0.403	1.755

Table 7 Temperature Model Peak Predict Results

Model	RMSE	MAE	Cv (RMSE)
CNN	0.821	0.634	3.571
GRU	0.74	0.561	3.219
LSTM	0.977	0.616	4.25
RNN	0.788	0.622	3.428
CNN-GRU	1.275	0.779	5.546
CNN-LSTM	0.926	0.635	4.028
LSTM-CNN	0.708	0.582	3.08

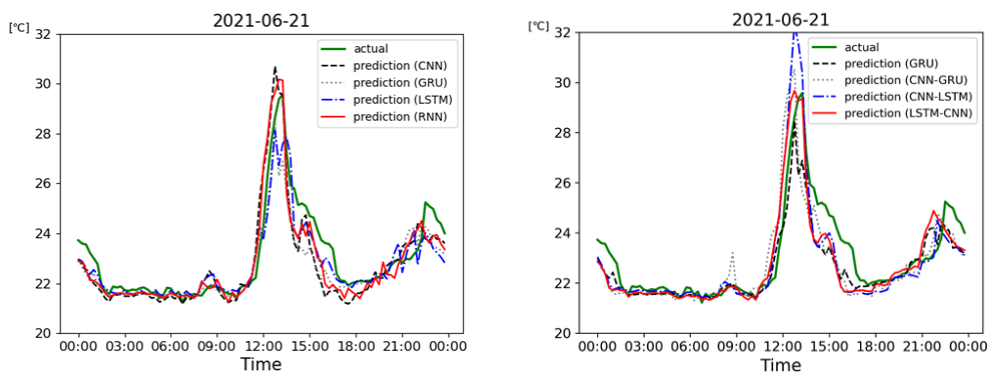


Fig. 15 Temperature Model Peak Predict Results (All Models)

Table 8 Hyperparameters for the Best Temperature Model

Models	Activation function	Epoch	Batch size	Learning rate	Dropout	Optimizer
GRU	elu	1000	128	0.001	0.2	Adam
LSTM-CNN	elu	1000	512	0.05	0.3	Adam

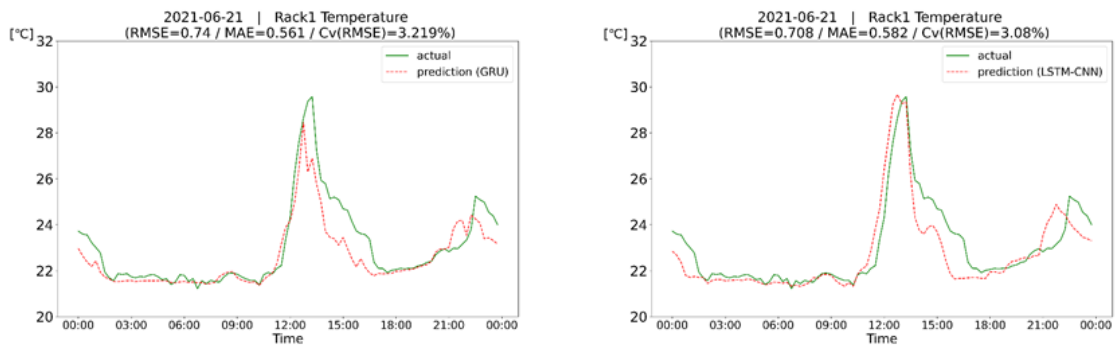


Fig. 16 Temperature Model Peak Predict Results (Best Models)

4.2 습도 예측 모델

습도 모델의 Input length는 1시간으로 4개이며 Output length는 1시간으로 4개이며, 2021년 5월 1일부터 2021년 6월 30일의 데이터를 사용하였다. Train date 31일, Validation date 15일, Test date 15일이다. 입력 데이터로는 Power (발전량), hr_sin (시간 사인 변환), hr_cos (시간 코사인 변환), hmd2 (Rack1 하단습도), hmd4 (Rack2 하단습도), hmd8 (Rack4 하단습도)을 사용하였다. Grid Search를 통해 Table 5의 Hyperparameters 중 최적의 조건을 탐색하였다.

Fig. 17은 2021년 6월 16일부터 6월 30일까지의 전체적인 예측 결과이며, Table 9는 각각의 모델의 성능을 비교한 결과이다. 전체 예측의 경우 CNN-GRU 모델을 제외하고는 각 모델별로 성능 차이는 크게 발생하지 않았다. 월 21일 오전 10시 30분 ~ 오후 18시경 습도가 급격히 변화하였으며, 이는 배터리 제조사에서 권장하는 습도 범위 이내이나, 변동성이 크기에 80%를 넘을 우려가 있어 해당 피크점을 예측하는 것은 중요하다. Table 10는 피크점에서의 모델별 예측 성능 비교 테이블이다. Fig. 18은 피크 발생일의 Basic Model과 Hybrid Model의 결과이다. Basic Model에서는 LSTM Model이 가장 좋은 성능을 보였으며, Hybrid Model에서는 CNN-LSTM 모델이 가장 좋은 성능을 보였다. LSTM 모델과 CNN-LSTM 모델의 Hyperparameter는 Table 11과 같다. 시계열 데이터의 유용한 특징을 추출하고 내부 표현을 학습하는 과정에 강점이 있는 CNN과 단기 및 장기 종속성을 식별하기 위한 LSTM 모델을 혼합하여 예측 성능이 뛰어났다. Fig. 19는 피크 발생일의 LSTM와 CNN-LSTM 모델의 결과 그래프이며, LSTM은 작은 피크점들을 예측을 잘 못하는 반면, CNN-LSTM은 전체적인 예측률이 좋았다. 이는 LSTM-CNN 예측 모델을 통해 ESS 실내 습도가 변화하는 지점에서도 위험 관리가 가능함을 나타낸다.

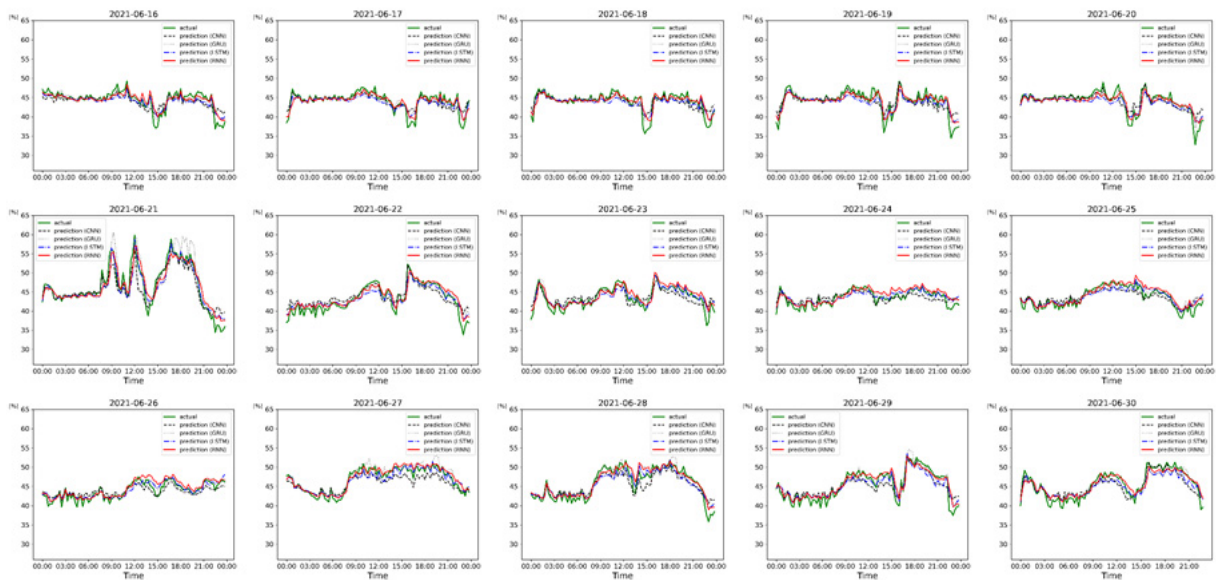


Fig. 17 Humidity Predict Model Results (Daily)

Table 9 Humidity Model Predict Results

Model	RMSE	MAE	Cv (RMSE)
CNN	1.702	1.001	3.821
GRU	1.537	0.958	3.45
LSTM	1.573	0.95	3.531
RNN	1.556	0.933	3.493
CNN-GRU	3.698	1.39	8.302
CNN-LSTM	1.391	0.952	3.078
LSTM-CNN	1.739	1.032	3.904

Table 10 Humidity Model Peak Predict Results

Model	RMSE	MAE	Cv (RMSE)
CNN	4.415	1.444	9.444
GRU	6.519	1.888	13.945
LSTM	3.491	1.302	7.468
RNN	4.142	1.519	8.86
CNN-GRU	10.773	2.456	23.045
CNN-LSTM	1.663	1.02	3.557
LSTM-CNN	10.43	2.3	22.311

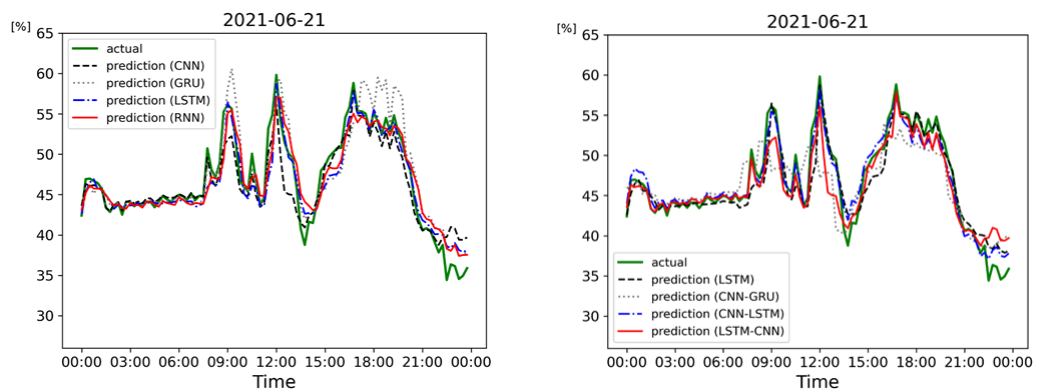


Fig. 18 Humidity Model Peak Predict Results (All Models)

Table 11 Hyperparameters for the Best Humidity Model

Models	Activation function	Epoch	Batch size	Learning rate	Dropout	Optimizer
LSTM	elu	1000	128	0.001	0.1	Adam
CNN-LSTM	relu	1000	512	0.05	0.3	Adam

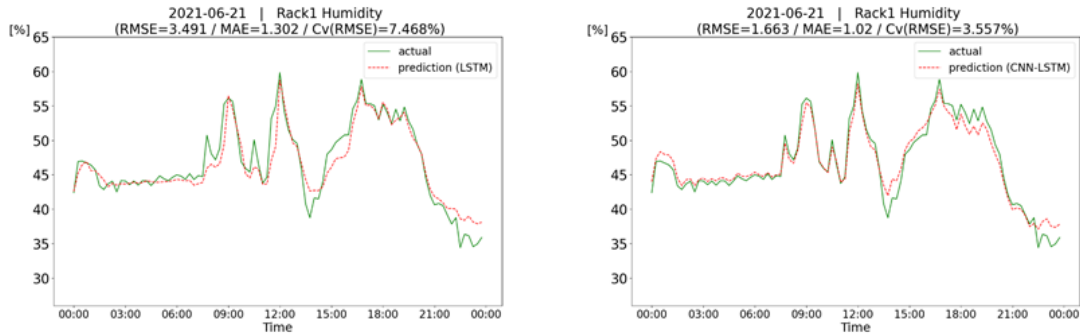


Fig. 19 Humidity Model Peak Predict Results (Best Models)

5. 결론

본 논문에서는 ESS 최적운동을 위해 온습도 센서 데이터 및 배터리 데이터를 이용해 배터리 충방전 시 온습도의 증감패턴을 고려하여 설계된 딥러닝 기반 학습 모델을 설계 및 개발하였다. 본 연구의 결과는 피크지점에서의 예측 모델을 비교했을 때, 온도 모델은 Hybrid Model인 LSTM-CNN 모델의 성능이 가장 우수하였으며, RMSE 0.708, Cv (RMSE) 3.08%의 성능을 보였다. 습도 모델은 Hybrid Model인 CNN-LSTM 모델의 성능이 가장 우수하였으며, RMSE 1.663, Cv (RMSE) 3.557%의 성능을 보였다. Hybrid 모델은 데이터에서 유용한 특징을 뽑아내는 CNN 레이어와 시계열 데이터의 장기 종속성을 식별하는 데 강점이 있는 LSTM 레이어를 혼합하여 동일한 데이터의 특징을 함께 사용함으로써 예측 성능을 효율적으로 증가시켰다. 현재는 간단한 구조의 네트워크로 구성되어 있기 때문에 향후 성능이 높은 학습모델로 발전할 가능성이 높다. 본 논문에서 개발한 딥러닝 모델은 실시간 센서 데이터만 받아오는 기존의 시스템과 달리 향후 1시간의 온도와 습도를 예측함으로써 ESS룸 내부에 온도편차가 생기는 구역을 최소화하고 여름철 높은 습도로 인해 발생하는 결로를 예방할 수 있다. 공조장치와 연계하여 산간지방이나 해안가에 설치된 ESS 내부 룸 온도와 습도를 적절히 관리하고, 에너지 최적화 제어를 통해 배터리의 폭발 위험을 줄일 수 있다.

후기

이 논문은 2021년도 중소벤처기업부의 재원으로 한국산업기술진흥원의 지역특화산업육성+(R&D) 지역스 타기업육성사업(No. S3192961)과 2021년도 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(No.2021R1F1A1063161)의 지원을 받아 수행된 연구임.

REFERENCES

1. Ministry of Trade, Industry and Energy (Korea), Power System Reliability and Electricity Quality Maintenance Standard, Notification No. 2018-104, 2018.

2. Chen, Z., Wang, J., Huang, J., Fu, T., Sun, G., Lai, S., and Zhao, J., The High-temperature and High-humidity Storage Behaviors and Electrochemical Degradation Mechanism of $\text{LiNi}_{0.6}\text{Co}_{0.2}\text{Mn}_{0.2}\text{O}_2$ Cathode Material for Lithium Ion Batteries, *Journal of Power Sources*, Vol. 363, pp. 168-176, 2017.
3. National Fire Research Institute, Basic Research for Analysis of Main Ignition Causes of ESS, 2019.
4. Mongird, K., Viswanathan, V., Alam, J., Vartanian, C., Sprenkle, V., and Baxter, R., 2020 Grid Energy Storage Technology Cost and Performance Assessment, *Energy*, 2020.
5. Wu, X., Mi, L., Tan, W., Qin, J. L., and Zhao, M. N., State of Charge (SOC) Estimation of Ni-MH Battery based on Least Square Support Vector Machines, In *Advanced Materials Research*, Trans Tech Publications Ltd., Vol. 211, pp. 1204-1209, 2011.
6. Song, Z., Hofmann, H., Li, J., Hou, J., Zhang, X., and Ouyang, M., The Optimization of a Hybrid Energy Storage System at Subzero Temperatures: Energy Management Strategy Design and Battery Heating Requirement Analysis, *Applied Energy*, Vol. 159, pp. 576-588, 2015.
7. Abutayeh, M., Alazzam, A., and El-Khasawneh, B., Optimizing Thermal Energy Storage Operation, *Solar Energy*, Vol. 120, pp. 318-329, 2015.
8. Pereira, J. and Silveira, M., Unsupervised Anomaly Detection in Energy Time Series Data Using Variational Recurrent Autoencoders with Attention, In *2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pp. 1275-1282, 2018.
9. Kim, T., and Kim, H. Y., Forecasting Stock Prices with a Feature Fusion LSTM-CNN Model Using Different Representations of the Same Data, *PloS one*, Vol. 14, e0212320, 2019.
10. Livieris, I. E., Pintelas, E., and Pintelas, P., A CNN – LSTM Model for Gold Price Time-series Forecasting, *Neural Computing and Applications*, Vol. 32, pp. 17351-17360, 2020.
11. Wang, J., Yu, L. C., Lai, K. R., and Zhang, X., Dimensional Sentiment Analysis Using a Regional CNN-LSTM Model, In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*, Vol. 2, pp. 225-230, 2016.