

실시간 ESS 화재 예측을 위한 Convolutional Autoencoder

이연서, 김현정, 김파란하늘, 현장훈*

국립한밭대학교

20231068@edu.hanbat.ac.kr, *jhhyeon@hanbat.ac.kr

Real-Time Fire Prediction in ESS via Convolutional Autoencoder

Yeonseo Lee, Hyeonjeong Kim, Paranhaneul Kim, Janghun Hyeon*

Hanbat National University

요약

최근 ESS의 사용이 증가하며 ESS 화재 또한 급증하고 있다. 리튬이온 ESS 화재의 대표적인 원인은 리튬이온 배터리의 열폭주이며, 열폭주로 인한 화재는 인접한 배터리로 빠르게 전파될 수 있기 때문에 사전에 예측해야 한다. 대표적인 화재 예측 알고리즘인 임계값 기반 알고리즘은 측정된 값이 사전에 정의된 임계값을 초과하면 화재라고 감지하기 때문에 노이즈에 민감하다는 문제점이 존재한다. 따라서, 본 논문에서는 노이즈 민감도를 줄이기 위해 Sliding Window와 Spectrogram을 사용하여 데이터를 재구성한 후, Convolutional Autoencoder를 이용하여 실시간 화재 예측을 수행하는 방법을 제안한다. 제안한 알고리즘은 본 연구에서 구축한 Test 데이터에 대해서 Accuracy 1.0, Recall 1.0, Precision 1.0을 보여주며 화재 예측을 정확하게 수행하였다.

I. Introduction

최근 전력을 저장해두었다가 필요한 시기에 전력을 공급하는 시스템인 ESS(Energy Storage System)에 대한 관심이 증가하며 친환경 에너지 분야에서의 ESS 사용이 증가하고 있다. ESS 사용이 증가함에 따라 ESS 화재도 빈번히 발생하고 있다. 리튬이온 배터리를 사용하는 ESS 화재는 주로 리튬이온배터리의 열폭주로 인해 발생하며, 열폭주로 인한 화재는 인접한 배터리로 화재가 전파될 수 있기 때문에 미리 대비하기 위해 예측을 하여야 한다.

화재를 예측하기 위하여 주로 사용되는 임계값 기반의 화재 예측 알고리즘은 특정 상황에 대한 지표값이 사전에 정의된 임계값을 초과하면 화재가 감지되는 간단한 방식을 사용하여 노이즈가 존재할 때 높은 오경보율을 나타낼 수 있다[1].

특정 상황에 대한 임계값만을 사용하여 노이즈에 민감하다는 기존 알고리즘의 문제점을 보완하기 위해 본 논문에서는 Spectrogram을 사용하여 데이터를 재구성하고, Convolutional Autoencoder를 수행하는 알고리즘을 제안한다.

II. Proposed Method

1) 전체 알고리즘

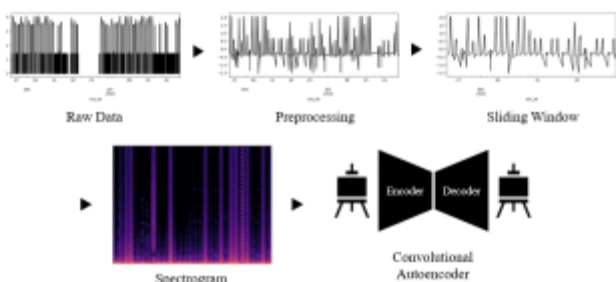


그림 1. 전체 알고리즘

본 연구에서 제안한 알고리즘은 [그림 1]과 같다. ESS 배터리의 Cell에서 1분 단위로 측정된 전압값을 불러온 후 보간, 스케일링과 같은 전처리를 진행한다. 데이터를 실시간으로 처리하는 것과 비슷한 형태로 데이터를 재구성하기 위해 Sliding Window를 사용하여 전체 날짜로 구성된 데이터를 일정한 구간으로 나눈다. 시간 정보와 주파수 성분의 정보를 모두 표현할 수 있는 Short-Time Fourier Transform을 수행한 뒤 시간, 주파수, 해당 시간대의 주파수 강도를 모두 나타낼 수 있는 Spectrogram으로 이미지화한다. 이미지화된 데이터를 Convolutional Autoencoder의 입력으로 넣어 이상탐지를 수행하여 화재를 예측한다.

2) 데이터셋 구성

ESS 배터리는 Rack, Module, Cell로 구성된다. 연구에 사용한 데이터는 Cell에서 1분 단위로 측정된 전압값이며, 화재가 난 데이터와 화재가 나지 않은 데이터로 구분된다.

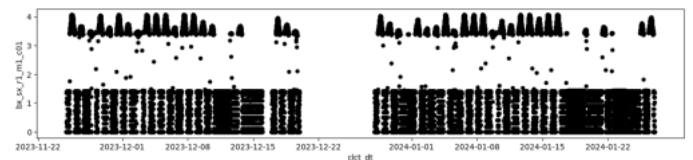


그림 2. 결측된 데이터 시각화

각 데이터의 측정날짜의 길이는 31일부터 182일까지 다양하게 존재하며 하드웨어의 결함이나 네트워크 문제와 같은 불안정한 센싱 환경에 노출되는 경우가 많기에 [그림 2]와 같이 결측된 날짜가 존재한다[2].

위와 같은 이유로 측정이 되지 않을 경우, 결측치는 일 단위로 존재한다.

3) 전처리

ESS 배터리에 따라 Module 안에 존재하는 Cell의 개수가 달라진다. Module을 구성하는 Cell의 개수가 다를 경우 STFT를 수행하는 길이가 달라지기 때문에 Spectrogram의 시간 축의 길이가 달라질 수 있다. 따라서, Module을 구성하는 각 Cell들의 STFT 수행 결과를 모두 더하여 Module

별로 하나의 Spectrogram으로 이미지화한다.

결측치 이전의 값과 결측치 모두 화재가 난 데이터의 패턴이 관찰된다면 화재라고 예측할 것이고, 두 값 모두 화재가 나지 않은 데이터의 패턴이 관찰된다면 화재가 나지 않았다고 예측할 것이다.

따라서 상술한 결측치를 처리하기 위해 결측된 날짜가 하루일 경우 결측된 날짜의 바로 이전 날짜의 데이터로 복사하고 연속 2일 이상일 경우 결측된 날짜로부터 결측된 길이만큼의 이전 데이터를 복사한다.

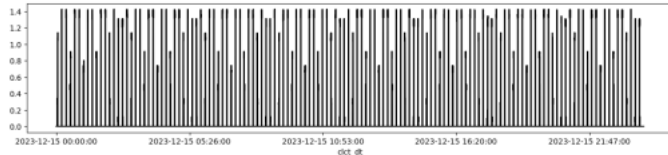


그림 3. 이상치 데이터 그래프

[그림 3]과 같이 결측이 되지 않았어도 센서 오류 등의 이유로 인해 제대로 측정되지 않은 데이터가 존재할 수 있다. 값을 복사하여 그대로 사용하는 방법에서 제대로 측정되지 않은 값을 사용하는 것은 노이즈로 취급될 수 있다. 이를 해결하기 위해 하루의 데이터에서 방전종지전압인 3.3V 이하 수치가 일정 비율 이상일 경우 해당 날짜도 결측치로 처리하고 위와 같은 방법으로 보간을 진행하였다.

4) Convolutional Autoencoder

Convolutional Autoencoder[3]은 1차원 데이터를 입력으로 하는 Autoencoder와 동일한 구조를 가지지만 Convolutional layer를 사용하여 Autoencoder보다 2차원 데이터의 특징을 추출하는 데 더 적합하다.

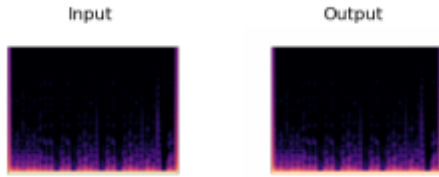


그림 4. 원본 데이터와 Convolutional Autoencoder의 출력

본 연구에서는 [그림 4]와 같이 Spectrogram으로 이미지화한 데이터를 Encoder의 입력으로 하며, Decoder를 거쳐 다시 생성된 Spectrogram을 출력으로 한다.

Autoencoder를 사용하여 이상탐지를 진행하는 방식은 정상 데이터를 사용하여 Train을 진행하고 모든 데이터에 대해서 Test를 진행하면서 재건 오차를 비교한다. 위의 방법을 사용하여 본 연구에서는 화재가 나지 않은 데이터로만 Train을 진행하며, 화재가 나지 않은 데이터와 화재가 난 데이터 모두 사용하여 Test를 진행한다.

5) 실험 과정

	Date Length	Rack * Module	Total
Train	33	840	27720
Validation	33	146	4845
Test	35	128	4471

표 1. 데이터셋 구성

Train 데이터는 ESS 배터리 4개, Validation 데이터는 화재가 난 ESS 배터리 1개와 화재가 나지 않은 ESS 배터리 1개, Test 데이터도 화재가 난 ESS 배터리와 화재가 나지 않은 ESS 배터리 각각 1개씩 사용하였다. [표 1]은 본 연구에서 사용한 데이터 날짜의 평균, Rack*Module의 수, 총 개수를 나타낸 것이다.

Sliding Window 기법을 사용하여 시계열 데이터를 30일 단위로 재구성

하여 전처리를 진행한다. ESS 배터리의 각 Cell 별로 STFT를 수행한 후 각 시간대에 해당하는 각 Cell의 STFT 수행 결과를 더하여 각 Module을 표현할 수 있는 하나의 Spectrogram으로 이미지화한다. Spectrogram을 Convolutional Autoencoder의 입력으로 사용한다.

Convolutional Autoencoder를 사용하여 화재가 난 데이터와 화재가 나지 않은 데이터의 재건오차를 구하여 임의의 임계값 이상이 되면 화재로 예측한다. 이때의 임계값은 Validation으로 사용된 화재가 난 데이터와 화재가 나지 않은 데이터의 재건오차 평균인 0.000107을 사용한다.

6) 결과

Accuracy	Recall	Precision
1.0	1.0	1.0

표 2. Test 데이터 성능

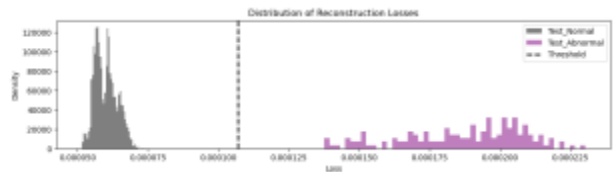


그림 5. Test 데이터 재건오차 분포

[그림 5]는 Test 데이터의 재건오차 분포를 시각화한 것이다. x축은 재건 오차, y축은 해당 재건오차의 빈도수를 나타낸다. 화재가 난 데이터와 화재가 나지 않은 데이터의 재건오차 분포를 보면 임계값을 기준으로 명확하게 분류할 수 있음을 알 수 있다.

이는 [표 2]와 같이 모든 Test 데이터에 대해서 완벽하게 분류할 수 있다는 것을 의미한다. 특히, Recall이 1.0이라는 수치는 본 논문에서 제안한 알고리즘이 화재를 조기에 예측해야하는 목표를 달성했음을 의미한다.

III. Conclusion

본 논문에서는 임계값만을 사용하여 화재를 예측한다는 기존 알고리즘의 문제점을 해결하기 위해 Spectrogram으로 이미지화한 데이터로 Convolutional Autoencoder를 수행하는 알고리즘을 제안한다.

Validation 데이터 재건오차의 평균을 임계값으로 설정하여 Test 데이터의 재건오차를 기준으로 분류한 결과 Accuracy 1.0, Recall 1.0, Precision 1.0으로 모든 데이터를 제대로 분류하는 것을 알 수 있다.

다만 Test에 사용한 데이터는 한 제조사에서 생성한 ESS 배터리의 데이터이므로 추후에 다양한 제조사의 데이터에서도 활용할 수 있도록 연구할 예정이다.

References

- [1] Y. Heo, S. Seo, W. Shim, and J. Kang, "A Study on the Real-time Fire Classification Algorithm Based on Stacked LSTM," J. Korean Soc. Hazard Mitig., vol. 21, no. 3, pp. 93-104, Jun. 2021. doi: 10.9798/KO SHAM.2021.21.3.93
- [2] 정재익, 구태연, and 박완기, "협업필터링 오토인코더 기반 센서 데이터 결측치 처리," in 한국통신학회 학술대회논문집, 2022, pp. 80-81.
- [3] X.-J. Mao, C. Shen, and Y.-B. Yang, "Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections," in Proc. Advances in Neural Information Processing Systems (NeurIPS), 2016, pp. 2802-2810.