

# 트랜스포머 모델을 활용한 전기차 배터리 SoH 예측 연구

전 준 식<sup>1)</sup> · 임 완 수<sup>\*2,3)</sup>

국립금오공과대학교 전자공학부<sup>1)</sup> · 국립금오공과대학교 전자공학과(항공기계융합전공)<sup>2)</sup> · 성균관대학교 전자전기공학부<sup>3)</sup>

## Novel SoH Estimation of Electric Vehicle Battery Using Transformer Model

Junsik Jeon<sup>1)</sup> · Wansu Lim<sup>\*2,3)</sup>

<sup>1)</sup>School of Electronic Engineering, Kumoh National Institute of Technology, Gumi 39177, Korea

<sup>2)</sup>Department of Aeronautics, Mechanical and Electronic Convergence Engineering, Kumoh National Institute of Technology, Gumi 39177, Korea

<sup>3)</sup>School of Electronic and Electrical Engineering, Sungkyunkwan University, Gyeonggi 16419, Korea

(Received 19 December 2023 / Revised 22 January 2024 / Accepted 24 January 2024)

**Abstract** : This paper presents a Transformer-based State of Health(SoH) estimation algorithm that is applied to publicly available NASA data. Data preprocessing involves the application of a moving average to reduce noise and utilize wavelet transformers to extract features. The preprocessed data serves as input to the Transformer model in estimating SoH. Next, a comparative analysis with an LSTM-based model is conducted by using the Root Mean Square Error(RMSE) metric. The proposed model demonstrates a superior SoH estimation accuracy, surpassing the LSTM-based model by up to 39.58 %.

**Key words** : Electric vehicle(전기 자동차), Lithium-ion battery(리튬 이온 배터리), State of health estimation(건강상태 예측), Transformer(트랜스포머 모델), Long short-term memory(장단기 기억 모델)

### 1. 서론

전기차 배터리의 관리와 유지보수는 전기차의 효율적이고 안전한 운영을 위해 매우 중요하여 이에 관한 다양한 연구가 진행되고 있다.<sup>1-3)</sup> 배터리의 상태를 나타내는 여러 지표 중 배터리 건강 상태(SoH, State of Health)는 배터리 수명을 나타내고 SoH를 정확히 예측하면 전기차의 성능과 안전을 높일 수 있다.<sup>4)</sup> 따라서, 전기차 배터리 SoH 예측을 위한 많은 연구가 진행되고 있으며, 주요 연구 결과는 다음과 같다.

Li 등<sup>5)</sup>은 휴먼 상태 전압(OCV, Open Circuit Voltage)과 등가 회로 모델(ECM, Equivalent Circuit Model)을 결합하여 SoH를 예측하였다. OCV를 이용하여 배터리 정보를 수집하고, ECM은 배터리의 전기적 특징을 수학적으로 모델링하여, 배터리의 동적 동작을 시뮬레이션한다. 그러나, OCV-ECM 방법은 실시간 측정에는 부적합하고, 환경 변화나 온도와 같은 외부 요인에 의해 결과가 왜곡될 수 있다. 이러한 이유로 OCV-ECM 방법은 주로 정적이

거나 느린 동적 조건에서 효과적이다. Xia와 Abu Qahouq<sup>6)</sup>은 ANN(Artificial Neural Network)을 이용하여 배터리 SoH를 예측하였다. ANN은 배터리의 전기적 특징과 SoH 간의 복잡한 상관관계를 학습하여 배터리의 건강 상태를 실시간으로 모니터링할 수 있다. 그러나 ANN의 성능은 학습 데이터양에 크게 의존하므로 데이터가 충분하지 않으면 SoH 예측 결과가 정확하지 않은 단점이 있다.

Wong 등<sup>7)</sup>은 시계열 데이터 분석에 많이 사용하는 LSTM(Long Short-Term Memory)을 이용하여 SoH를 예측하였다. Wang 등<sup>8)</sup>은 CNN 기반 모델을 이용하여 배터리 데이터의 잡음 처리와 특징 추출을 한 후, LSTM을 이용하여 SoH 학습을 진행하였다.

Zhang 등<sup>9)</sup>은 일반화 능력이 뛰어난 RBFNN(Radial Basis Function Neural Network)을 사용하여 모델을 구축해 SoH를 추정한다. Zhang 등<sup>10)</sup>은 여러 기계학습 알고리즘을 활용해서 SoH를 추정한 후 칼만 필터를 이용하여

\*Corresponding author, E-mail: [wansu.lim@skku.edu](mailto:wansu.lim@skku.edu)

<sup>†</sup>This is an Open-Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License(<http://creativecommons.org/licenses/by-nc/3.0>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium provided the original work is properly cited.

추정치들을 결합한 후 최종 SoH를 제공하는 방법을 제안했다.

연구 동향을 분석한 결과, 시계열 데이터 학습에 효과적인 딥러닝 모델을 활용하면 타 모델 대비 정확한 배터리 상태 예측 결과를 도출할 수 있었다. 이는 배터리 상태를 확인할 수 있는 중요한 정보인 전류와 전압이 시계열 데이터이기 때문이다. 따라서, 본 논문은 시계열 데이터 처리에 뛰어난 트랜스포머(Transformer) 모델을 사용하여 배터리 SoH를 예측하는 알고리즘을 제안한다. 또한, 예측 정확도를 높이기 위하여 이동 평균(Moving average)과 웨이블릿 변환(Wavelet transform)을 활용하여 데이터의 잡음 제거와 특징 추출을 수행한다. 성능 검증을 위해 NASA의 공개 배터리 데이터를 활용하여 이 트랜스포머 모델을 학습하고, SoH 예측 결과를 기존 논문에서 제시한 LSTM과 비교하여 제안한 알고리즘의 성능을 평가한다.

## 2. 트랜스포머 기반 SoH 예측

Fig. 1은 본 논문에서 제안한 연구 흐름도이다. 제안한 SoH 예측 알고리즘은 4단계로 진행된다. 데이터 전처리(Data preprocessing) 단계는 이동 평균을 사용하여 배터리 데이터를 평활화하고 잡음을 제거한다. 이동 평균은 데이터의 잡음이나 일시적인 변동을 제거하고 장기적인 패턴이나 추세를 더 명확하게 드러내는 것을 도와주고, 분석에 유용한 정제된 데이터를 생성한다. 데이터 특징 추출(Feature extraction) 단계는 웨이블릿 변환을 이용하여 데이터에서 중요한 특징을 추출한다. 웨이블릿 변환은 시간-주파수 영역에서 신호를 분석하여 배터리의 동작 특징을 더욱 명확하게 드러내게 도와준다. 추출한 특징으로 생성한 데이터를 트랜스포머 모델에 입력으로 사용하여 SoH를 예측하기 위한 학습을 진행한다.

마지막으로, 성능 평가(Evaluate performance)단계는 제안한 모델을 학습한 후 SoH를 예측하여 기존 SoH 예측 모델과 비교한다. 제안한 기법과 벤치마크 모델의 오

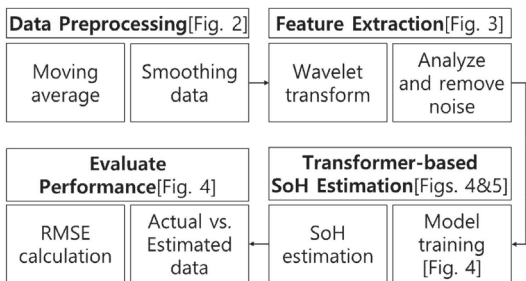


Fig. 1 Flowchart of Transformer-based SoH estimation

차를 비교하기 위해 RMSE(Root Mean Squardd Error)를 사용한다.

### 2.1 데이터 전처리(Data Preprocessing)

SoH는 전압과 전류에 영향을 받으므로 센서의 잡음이나 환경적 요인으로 인해 오차가 생길 수 있다. 이러한 문제를 해결하기 위해 데이터 전처리 단계에서 이동평균기법을 이용한다.

이동평균기법 작동 메커니즘은 두 가지 단계로 나뉜다. 첫 번째 단계에서는 입력 데이터의 변동성을 고려하며 윈도우 크기를 결정한다. 윈도우 크기는 평균을 계산할 때 고려되는 연속적인 데이터 포인트의 수를 정의하며, 데이터의 평활화 정도와 비례한다. 두 번째 단계에서는 설정된 윈도우 크기에 따라 각 데이터 포인트에 대한 이동 평균을 계산한다. 데이터 포인트를 주변 데이터 포인트들과 함께 평균화하여 계산함으로써 이동 평균은 데이터의 변동성과 잡음을 감소시키고, 데이터의 이상치를 완화하는 데 있어 장점을 가진다.

Fig. 2는 Fig. 1의 Data preprocessing을 구현한 의사코드이다. 전처리를 위해 이동평균기법을 사용하고, Load data and variable initialization, calculate moving average 등 2단계로 구성한다. Load data and initialization은 배터리의 전류, 전압, 그리고 SoH 데이터를 알고리즘으로 호출하고 리스트에 저장하는 단계이다. 호출한 데이터는 전처리를 위해 리스트에 저장되며, 각 데이터의 이동 평균 계산을 위한 ‘Window\_size’ 설정한다. ‘Window\_size’는 연속된 데이터 포인트 개수이다. Calculate moving

Data Preprocessing	
Input : cuurent, voltage, SoH	
Output : smoothed_data	
<b>Load Data and Variable Initialization</b>	
1	dataset = Load the current and voltage, SoH data
2	smoothed_data=Initialize empty lists for smoothed
3	current, voltage, and SoH
4	window_size = Set the window size for moving
5	average
<b>Calculate Moving Average</b>	
6	for sample in dataset :
7	Extract data from each sample data
8	smoothed_data = Calculate moving average
9	Append smoothed data to list
10	Return the smoothed_data

Fig. 2 Pseudo code of data preprocessing

average는 이동평균기법을 이용해서 데이터의 변동성을 줄이는 단계이다. 이 단계는 각 데이터 포인트의 이동 평균을 계산하고 그 결과를 리스트에 저장한다. ‘Smoothed\_data’는 이동평균기법을 적용한 데이터이고 원본 데이터에 비해 잡음이 적고 변화 추세가 명확한 데이터이다. 따라서, 배터리 상태를 더욱 정확하게 예측할 수 있다. 마지막으로 ‘Smoothed\_data’를 반환하여 SoH를 예측하는 트랜스포머 모델 학습에 사용한다.

**2.2 특징 추출(Feature Extraction)**

특징 추출은 데이터 구조에서 중요한 정보를 식별하는 핵심 과정이다. 웨이블릿 변환은 다양한 데이터 타입에서 동적 특성을 동시에 분석할 수 있는 유연성을 가지고 있어 다양한 형태의 데이터에도 적용할 수 있다.

웨이블릿 변환의 작동 메커니즘은 두 가지 단계로 나뉜다. 첫 번째 단계에서는 분석하고자 하는 데이터 특징 분석에 최적화된 Wavelet 함수를 사용한다. 두 번째 단계는 선택한 Wavelet 함수를 사용하여 원래의 데이터를 여러 주파수 구성요소로 분해한 후 주파수 변화를 포착한다. 이 과정은 데이터의 특정 구간에서 미세한 변화까지 확인할 수 있어 데이터 패턴을 세밀하게 분석하는데 장점이 있다.

Fig. 3은 Fig. 1의 Feature extraction을 구현한 의사코드이다. 특징 추출을 위해 웨이블릿 변환을 사용하고, Load

Feature Extraction	
Input :	smoothed_data
Output :	vector_data
Load Data and Variable Initialization	
1	smoothed_data = Load smoothed current, voltage and SoH (Output of Fig. 2)
2	
3	features_data = Initialize empty list for
4	features_current, features_voltage, features_SoH
Feature Extraction	
8	for sample in smoothed_data :
9	Extract features of current, voltage, and SoH from
10	smoothed_data
11	Save features to features_data
Vectorization for Transformer Model	
12	Create vectors representing three features (current,
13	voltage, SoH) for Transformer’s input
14	vector_current, vector_voltage, vector_SoH
15	Save vector_data

Fig. 3 Pseudo code of feature extraction

data and variable initialization, Feature extraction, Vectorization for transformer model 등 3단계로 구성한다. 첫 번째 단계인 Load data and variable initialization은 이동평균기법을 수행한 Fig. 2의 결과인 ‘Smoothed\_data’를 호출하고 다양한 변수를 초기화하는 단계이다. Feature extraction 단계는 ‘Smoothed\_data’에 웨이블릿 변환을 적용하여 특징을 추출한 후, 추출한 특징을 ‘Features\_data’에 저장한다. Vectorization for transformer model 단계는 추출한 특징을 트랜스포머의 입력으로 사용할 수 있도록 ‘Features\_data’를 ‘Vector\_data’로 변환하여 저장한다.

**2.3 트랜스포머 모델 기반 SoH 예측**

트랜스포머 모델은 ‘Attention’ 메커니즘을 중심으로, 시계열 데이터의 복잡한 데이터 간 의존성을 모델링 하는데 효과적이다. 트랜스포머는 인코더와 디코더 두 부분으로 구성되며, 각 부분은 다수의 층으로 되어 있어 복잡한 데이터 패턴인식에 유리하다. 이러한 구조는 BERT, GPT, T5 등 현대 NLP 모델에게도 적용되어 있으며, RNN이나 LSTM보다 빠르고 효율적인 학습이 가능하다.

트랜스포머의 또 다른 장점은 다양한 분야에서 높은 성능과 유연성을 보인다는 것이다. 텍스트 데이터뿐 아니라 이미지 분류, 객체 탐지, 음성 인식 등 다양한 분야

Train Model	
Input :	Load vector_data
Output :	trained_model
Load Data	
1	Load vector_data
Configure Model Architecture	
2	Stack multiple transformer encoder layers
3	Add an output layer for SoH estimation
Set Training Parameter	
4	Apply the Adam optimization algorithm
5	Set the batch size and number of epoch
Train Model	
6	Split the data in to training data and test data
7	Train the model with the training data
8	Implement early stopping if validation loss does not improve
Evaluate Performance	
9	Evaluate the model performance using test data
10	Compute evaluation metrics such as RMSE
11	Save the trained_model

Fig. 4 Pseudo code of model training

에서 그 효용성이 입증되고 있다. 또한, Self-Attention 메커니즘을 활용하여 다변량 데이터를 정교하게 파악한다. 이는 순차적 데이터 처리방식과 달리 높은 병렬 처리 능력을 제공하며, 학습 과정을 빠르게 진행할 수 있는 장점이 있다. 본 논문은 이러한 트랜스포머 모델의 장점을 SoH 예측에 활용한다.

Fig. 4는 Fig. 1의 Transformer-based SoH estimation와 Evaluate performance을 구현한 의사코드이고, 트랜스포머 모델을 이용하여 배터리 SoH를 예측한다. 의사코드는 Load data, Configure model architecture, Set training parameter, Train the model, Evaluate performance 등 5단계로 구성한다. Load data 단계는 Fig. 3의 출력인 ‘Vector\_data’를 호출하여 트랜스포머 모델 입력으로 사용한다. Configure model architecture 단계는 트랜스포머 모델의 구조를 설정한다. 여러 개 ‘Transformer encoder layers’를 쌓고, 배터리 SoH를 예측하기 위한 ‘Output layer’를 추가한다. 이러한 구조는 시계열 데이터를 효과적으로 처리하고, 정확한 예측을 가능하게 한다. Set training parameter 단계는 학습 과정을 최적화하기 위해 Adam optimization algorithm(Adam)을 적용한다. Adam은 학습률을 최적으로 조절하여 빠른 수렴을 도모하고, 효율적인 학습을 지원한다. 또한, 학습에 사용할 ‘Batch size’와 ‘Number of epochs’ 등 하이퍼파라미터를 설정한다. 이러한 설정은 모델의 학습 속도와 성능에 직접적인 영향을 미치며, 과적합을 방지하고 일반화 능력을 향상하는 역할을 한다. Train model 단계는 데이터를 ‘Training data’와 ‘Test data’로 분리하여 모델을 학습하고 검증한다. 이 과정에서 ‘Early stopping’ 기법이 적용되어, 검증 데이터에 대한 손실이 개선되지 않으면 학습을 조기에 종료한다. 마지막으로 Evaluate performance 단계는 ‘Test data’를 사용하여 모델 성능을 평가한다. RMSE를 평가 지표로 사용하여 모델의 예측 정확도를 측정하고, 평가 결과를 저장한다.

Fig. 5는 Fig. 1에서 Transformer-based SoH estimation에서 사용한 모델의 구조이다. 트랜스포머 모델은 Embedding, Multi-head self-attention, Position-wise FFNN, Add&Norm dense, Softmax 등 5개 모듈로 구성한다. Embedding은 입력 데이터를 모델이 효율적으로 학습할 수 있도록 형태를 변경하는 모듈이다. 제안한 모델은 위치 임베딩(Position embedding)을 사용하여 입력 데이터의 순서 정보를 제공한다. 트랜스포머 모델이 입력 데이터의 각 요소 위치를 파악할 수 있으므로 학습 정확도를 높일 수 있다. Multi-head self-attention 모듈은 복수의 Attention head로 구성된 메커니즘이다. 각 Attention head는 입력 데이터의 벡터 간의 유사도를 계산하여 이를 기

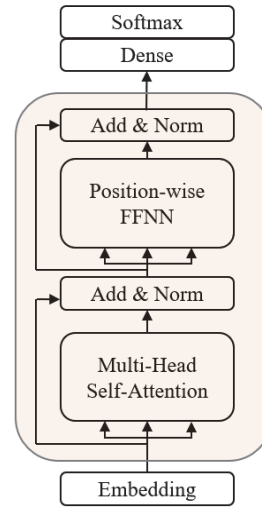


Fig. 5 Transformer model used for SoH estimation

반으로 가중치를 할당한다. 높은 가중치를 가진 벡터들은 출력에서 주요하게 반영되며, 이를 통해 모델이 데이터 내의 패턴을 효율적으로 학습하도록 도와준다. 가중치가 높은 벡터들의 영향은 모델의 후속 처리 과정에서도 지속되어 모델의 전체적인 성능과 정확도를 높인다. ADD&Norm 모듈은 잔차 연결과 레이어 정규화를 통해 모델의 안정적인 학습을 지원한다. Position-wise Feed-Forward Neural Network(FFNN) 모듈은 벡터를 고차원 화하여 정확도를 높여주는 역할을 한다. 병렬로 데이터가 처리되며, 이 과정에서 선형 레이어와 활성화 함수를 사용한다. 선형 레이어는 데이터를 높은 차원으로 확장하여 더 많은 정보를 포함하고, 활성화 함수는 비선형 변환을 제공한다. 선형 레이어와 활성화 함수의 조합은 데이터의 각 부분에 숨겨진 패턴과 자세한 구조를 파악하고 추출하는 역할을 하여 모델의 학습을 효과적으로 만든다. 마지막으로 Dense 레이어를 통해 데이터의 다양한 특징들을 통합하여 비선형 활성화 함수인 Softmax가 있는 출력 레이어에서 결과를 생성한다.

### 3. 시뮬레이션 결과

#### 3.1 시뮬레이션 설정

NASA 공개 배터리 데이터 5, 7, 18번 셀의 전류, 전압, SoH를 시뮬레이션에 이용하였다. 데이터 전처리를 위해 이동 평균과 웨이블릿 변환을 사용하였고, 학습 데이터 양을 30 %, 50 %, 70 % 비율로 늘려가며 시뮬레이션을 진행하였다. 학습이 끝난 후 검증 데이터로 예측 정확도를 도출하였고, 예측 정확도는 RMSE를 이용하였다. (1)은 RMSE 수식이며,  $n$ 은 전체 데이터 포인트 개수이고  $y_i$

와  $\hat{y}_i$ 는  $i$ 번째 실제 SoH와 예측 SoH를 각각 의미한다.

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{n}} \quad (1)$$

### 3.2 SoH 예측 결과 비교

Table 1은 각각 30 %, 50 %, 70 % 등으로 변화시켜 시뮬레이션을 수행하여 나온 RMSE이다. 학습 데이터를 전체 데이터에서 30 % 사용했을 때 Cell#05, Cell#07, Cell#18의 RMSE는 각각 0.0047, 0.0049, 0.0065로 평균은 0.0054이고, 학습 데이터를 50 % 사용하면 셀마다 RMSE는 0.0039, 0.0034, 0.0057로 평균은 0.0043이다. 학습 데이터를 70 %로 하면 셀마다 RMSE는 0.0029, 0.0028, 0.0037이고 평균은 0.0031로 RMSE는 30 %와 50 % 대비 각각 42.59 %와 27.91 % 향상하였다.

Table 2는 제안한 모델과 기존 LSTM 모델 간 RMSE 비교 결과이다. 비교에 사용한 셀은 5, 7, 18번 셀이고, LSTM 모델의 RMSE는 각각 0.0048, 0.0042, 0.0059이다. 트랜스포머 기반 제안한 모델의 RMSE는 각각 0.0029, 0.0028, 0.0037로 LSTM 대비 각각 39.58 %, 33.33 %, 37.29 % 향상하였다.

Fig. 6은 NASA 5번 셀 데이터를 이용하여 제안한 트랜스포머 기반 모델과 벤치마크로 사용한 LSTM 모델의 예측값과 실제값을 비교한 그래프이다. 실선은 실제 데이터를 의미하고 파선은 제안한 모델의 예측값이고 등근 점선은 LSTM의 예측값을 의미한다. 학습에 사용한 데이터는 전체 데이터의 70 %이며 검증은 나머지 30 %

Table 1 RMSE comparison according to different volume of training data, 30 %, 50 %, and 70 %

	Model trained with 30 %	Model trained with 50 %	Model trained with 70 %
Cell #05	0.0047	0.0039	0.0029
Cell #07	0.0049	0.0034	0.0028
Cell #18	0.0065	0.0057	0.0037
Average	0.0054	0.0043	0.0031

Table 2 RMSE comparison between proposed and LSTM

	Cell #05	Cell #07	Cell #18
Proposed (Model trained with 70 %)	0.0029	0.0028	0.0037
LSTM [7]	0.0048	0.0042	0.0059
Improvement	39.58 %	33.33 %	37.29 %

를 사용하였다. 그래프에서 학습과 검증은 각각 1~119번과 120~168번 Discharge cycle이다. 확대한 부분에서 125번 Discharge cycle을 살펴보면 LSTM 기반 모델 대비 제안한 모델이 최대 79.51 % 향상된 것을 확인할 수 있다.

Fig. 7은 NASA 7번 셀 데이터를 이용하여 제안한 트랜스포머 기반 모델과 벤치마크로 사용한 LSTM 모델의 예측값과 실제값을 비교한 그래프이다. 학습에 사용한

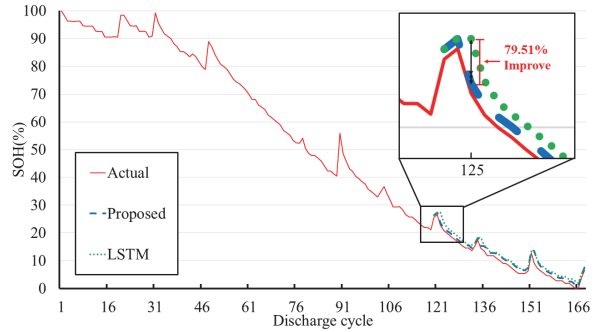


Fig. 6 Comparison of SoH estimation between proposed method and benchmark LSTM with Cell #05

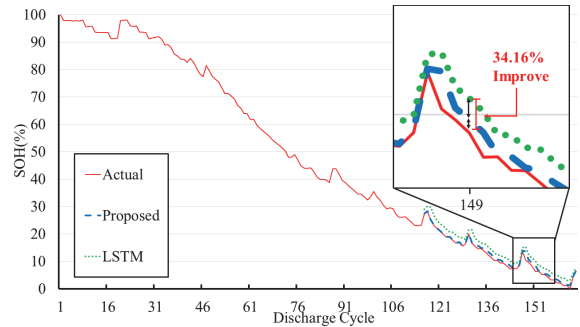


Fig. 7 Comparison of SoH estimation between proposed method and benchmark LSTM with Cell #07

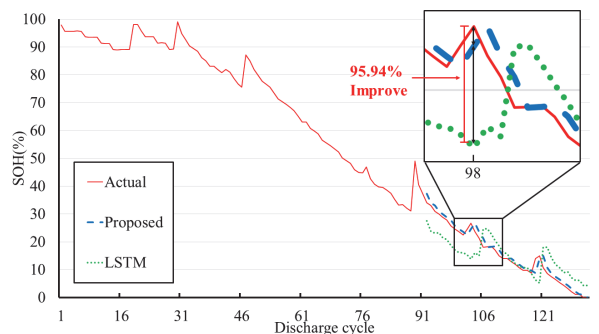


Fig. 8 Comparison of SoH estimation between proposed method and benchmark LSTM with Cell #18

데이터는 전체 데이터의 70 %이며 검증은 나머지 30 %를 사용하였다. 그래프에서 학습과 검증은 각각 1~119번과 120~168번 Discharge cycle이다. 확대한 부분에서 153번 Discharge cycle을 살펴보면 LSTM 기반 모델 대비 제안한 모델이 최대 34.16 % 향상된 것을 확인할 수 있다.

Fig. 8은 NASA 18번 셀 데이터를 이용하여 제안한 트랜스포머 기반 모델과 벤치마크로 사용한 LSTM 모델의 예측값과 실제값을 비교한 그래프이다. 학습에 사용한 데이터는 전체 데이터의 70 %이며 검증은 나머지 30 %를 사용하였다. 그래프에서 학습과 검증은 각각 1~91번과 92~132번 Discharge cycle이다. 확대한 부분에서 98번 Discharge cycle을 살펴보면 LSTM 기반 모델 대비 제안한 모델이 최대 95.94 % 향상된 것을 확인할 수 있다.

#### 4. 결론

본 논문은 트랜스포머 기반 SoH 예측 알고리즘을 제안하였다. 시뮬레이션은 NASA 공개데이터를 사용하였다. 데이터 전처리에는 이동 평균을 사용하여 잡음을 제거하였고, 웨이블릿 변환을 이용하여 데이터 특징을 추출하였다. 트랜스포머 모델은 전처리한 데이터로 학습하였다. 다양한 시뮬레이션 분석을 위해 학습 데이터양을 30 %, 50 %, 70 %로 변화시키며 시뮬레이션을 수행하였고 70 %의 데이터로 학습시켰을 때 정확도가 가장 높았다. 또한, SoH 예측 정확도는 RMSE를 이용해 표현했으며, 비교 모델로 LSTM을 이용하였다. 제안한 모델의 SoH 예측 정확도는 LSTM 예측보다 최대 39.58 % 향상되었다.

#### 후 기

본 논문의 일부 내용은 한국자동차공학회 2023 추계 학술대회의 포스터 세션에 제출되었음.

This work was supported by the National Research Foundation of Korea(2021R111A3056900).

#### References

- 1) C. Zheng, Y. Park, W. Lim and S. Cha, "A Study on Battery SOC Estimation by Regenerative Braking in Electric Vehicles," Transactions of KSAE, Vol.20, No.1, pp.119-123, 2012.
- 2) J. Gim, W. Choi and C. Ahn, "Design of Unscented Kalaman Filter with Gated Recurrent Units-based Battery Model for SOC Estimation," Transactions of KSAE, Vol.30, No.1, pp.61-68, 2022.
- 3) J. Kim, H. Chun, J. Yu, S. Han, J. Lee and T. Lee, "SoH Estimation of Li-ion Batteries based on Multilayer Perceptron," KSAE Spring Conference Proceedings, pp.1129-1130, 2019.
- 4) J. Kim, C. Park and T. Kwon, "Development of SOC Algorithm," KSAE Spring Conference Proceedings, pp.1527-1532, 2012.
- 5) S. Li, K. Li, E. Xiao and C. Wong, "Joint SoC and SoH Estimation for Zinc-Nickel Single-Flow Batteries," IEEE Transactions on Industrial Electronics, Vol.67, pp.8484-8494, 2020.
- 6) Z. Xia and J. Abu Qahouq, "State-of-Charge Balancing of Lithium-Ion Batteries With State-of-Health Awareness Capability," IEEE Transactions on Industrial Applications, Vol.57, pp.673-684, 2020.
- 7) K. Wong, M. Bosello, R. Tse, C. Falcomer, C. Rossi and G. Pau, "Li-Ion Batteries State-of-Charge Estimation Using Deep LSTM at Various Battery Specifications and Discharge Cycles," GoodIT Conference Proceedings, pp.85-90, 2021.
- 8) X. Wang, Z. Hao, Z. Chen and J. Zhang, "Joint Prediction of Li-ion Battery State of Charge and State of Health Based on the DRSN-CW-LSTM Model," IEEE Access, Vol.11, pp.70263-70273, 2023.
- 9) D. Zhang, Z. Wang, L. Peng, Z. Qin, Q. Wang, C. She and P. Bauer, "Multi-Step Fast Charging Based State of Health Estimation of Lithium-Ion Batteries," IEEE Transactions on Industrial Electrification, Early Access, 2023.
- 10) Y. Zhang, T. Wik, J. Bergstrom and C. Zou, "State of Health Estimation for Lithium-Ion Batteries under Arbitrary Usage Using Data-Driven Multi-Model Fusion," IEEE Transactions on Industrial Electrification, Vol.10, No.1, pp.1494-1507, 2023.