

A Proposal of Sensor-based Time Series Classification Model using Explainable Convolutional Neural Network

Youngjun Jang*, Jiho Kim*, Hongchul Lee*

*Student Researcher, Dept. of Industrial and Management Engineering, Korea University, Seoul, Korea

*Student Researcher, Dept. of Industrial and Management Engineering, Korea University, Seoul, Korea

*Professor, Dept. of Industrial and Management Engineering, Korea University, Seoul, Korea

[Abstract]

Sensor data can provide fault diagnosis for equipment. However, the cause analysis for fault results of equipment is not often provided. In this study, we propose an explainable convolutional neural network framework for the sensor-based time series classification model. We used sensor-based time series dataset, acquired from vehicles equipped with sensors, and the Wafer dataset, acquired from manufacturing process. Moreover, we used Cycle Signal dataset, acquired from real world mechanical equipment, and for Data augmentation methods, scaling and jittering were used to train our deep learning models. In addition, our proposed classification models are convolutional neural network based models, FCN, 1D-CNN, and ResNet, to compare evaluations for each model. Our experimental results show that the ResNet provides promising results in the context of time series classification with accuracy and F1 Score reaching 95%, improved by 3% compared to the previous study. Furthermore, we propose XAI methods, Class Activation Map and Layer Visualization, to interpret the experiment result. XAI methods can visualize the time series interval that shows important factors for sensor data classification.

▶ **Key words:** Sensor Data, Time Series Classification, Pattern Recognition, Deep Learning, eXplainable Artificial Intelligence(XAI)

[요 약]

센서 데이터를 활용하여 설비의 이상 진단이 가능해졌다. 하지만 설비 이상에 대한 원인 분석은 미비한 실정이다. 본 연구에서는 센서 기반 시계열 데이터 분류 모델을 위한 해석가능한 합성곱 신경망 프레임워크를 제안한다. 연구에서 사용된 센서 기반 시계열 데이터는 실제 차량에 부착된 센서를 통해 수집되었고, 반도체의 웨이퍼 데이터는 공정 과정에서 수집되었다. 추가로 실제 기계 설비에서 수집된 주기 신호 데이터를 이용 하였으며, 충분한 학습을 위해 Data augmentation 방법론인 Scaling과 Jittering을 적용하였다. 또한, 본 연구에서는 3가지 합성곱 신경망 기반 모델들을 제안하고 각각의 성능을 비교하였다. 본 연구에서는 ResNet에 Jittering을 적용한 결과 정확도 95%, F1 점수 95%로 가장 뛰어난 성능을 보였으며, 기존 연구 대비 3%의 성능 향상을 보였다. 더 나아가 결과의 해석을 위한 XAI 방법론으로 Class Activation Map과 Layer Visualization을 제안하였으며, 센서 데이터 분류에 중요 영향을 끼치는 시계열 구간을 시각적으로 확인하였다.

▶ **주제어:** 센서 데이터, 시계열 데이터 분류, 패턴 인식, 딥러닝, 설명가능한 인공지능

- First Author: Youngjun Jang, Corresponding Author: Hongchul Lee
- *Youngjun Jang (yjang11@korea.ac.kr), Dept. of Industrial and Management Engineering, Korea University
- *Jiho Kim (jihonav@korea.ac.kr), Dept. of Industrial and Management Engineering, Korea University
- *Hongchul Lee (hlee@korea.ac.kr), Dept. of Industrial and Management Engineering, Korea University
- Received: 2022. 03. 16, Revised: 2022. 04. 27, Accepted: 2022. 04. 28.

I. Introduction

1. Background

최근 제조 분야에서는 센서 장비를 통한 데이터의 수집 및 공유가 활발하게 이용되고 있다. 센서에서 수집된 데이터를 시간에 따라 축적하여 효율적으로 운영하는 것이 제조 분야에서의 핵심이다[1]. 설비에 부착된 센서에서 수집된 시계열 데이터를 통해 설비의 상태를 실시간으로 확인할 수 있도록 한다. 이를 통해 설비의 고장을 조기에 진단할 수 있으며, 이는 정비 시에 원인 파악 및 대응을 보다 효율적으로 운용할 수 있다[2]. 제조 산업 분야에서 활용되고 있는 스마트팩토리 시스템에서는 각 설비에 부착된 센서에서 수집된 데이터를 통해 공정의 불량 여부를 사전에 확인할 수 있다. 다양한 분야에서 시계열 데이터 분석이 요구됨에 따라 다수의 방법론이 제안되었다. 특히 컴퓨팅 기술의 발전으로 수집 및 저장할 수 있는 데이터의 양이 증가함에 따라 시계열 데이터 분석 알고리즘들이 제안되었다[3]. 기존의 시계열 데이터 분석을 위해 사용되었던 대표적인 방법은 거리기반 방법론과 특징 추출 방법론이 있다. Jeong 등[4]의 연구에서는 두 개의 시계열 데이터의 거리 및 유사도를 산출하여 최적의 index를 매칭 하여 추정된 후 이를 토대로 길이가 짧은 데이터에 거리기반 분석을 수행하였다. Ahn 등[5]의 연구는 시계열 간 관련성을 측정하기 위해 카이 제곱 통계량, t-검정 통계량, 왈드 통계량 등의 통계적 지표 등이 사용하였으며, 필터링(filtering) 방법을 이용하여 중복성이 적은 특징을 선택하고 시계열 데이터를 분류하였다.

시계열 데이터 분석에서 분류 및 예측은 여전히 통계적 방법론과 선형 모델과 같은 간단한 방법론들을 위주로 사용되고 있다. 최근에는 딥러닝 방법론을 활용하여 시계열 데이터 분류 및 예측을 시도하고 있다. Smirnov 등[6]의 연구에서는 Multi Layer Perceptron (MLP), Fully Convolutional Neural Network (FCN)을 계열 데이터 분류에 사용하였다. Wenninger 등[7]의 연구에서는 Residual Network (ResNet)를 사용하였고, Fawaz 등[8]의 연구에서는 MLP, FCN, ResNet 을 사용하여 센서 기반의 시계열 데이터를 분류하였다. 하지만 머신러닝과 딥러닝 모델은 높은 예측력을 가지나 설명력이 부족하다는 한계를 지닌다. 최근 머신러닝 및 딥러닝 모델들이 다양한 현실 상황에 대한 예측을 수행하고 있으나 이해 관계자들은 해석력을 더욱 요구하고 있다 [9]. 이는 사람들이 일반적으로 해석력이 부족한 기술 채택에 주저하기 때문이다 [10]. 이러한 일면에서, 시계열 데이터에 대한 딥러닝 분류 모델

을 구축하여도, 설명 가능한 근거를 제시하지 못한다면 무분별한 사용으로 인한 위험 요소가 크다고 볼 수 있다. 따라서, 본 연구에서는 딥러닝 모델을 사용하여 기존 분류 모델의 성능을 향상할 뿐 아니라 높은 분류 성능을 가진 이유를 설명 가능한 방법론을 사용하여 결과를 해석하였다.

XAI(eXplainable Artificial Intelligence)는 인공지능 모델이 의사결정을 수행할 시 결과에 대한 근거를 찾는 데 목적을 가지며, 모델의 학습되는 과정과 결과를 설명하는 기술이다 [10, 11]. XAI는 단순한 이진 예측보다 더 많은 정보를 추출하도록 한다 [12]. 본 연구에서는 XAI 방법론 중 이미지 분류 모델에서 해석 결과에 대한 근거를 시각화하여 탐색하게 하는 Class Activation Map(CAM)을 시계열 데이터 분석에 적용하였다 [13]. 더하여 딥러닝 모델 학습 과정에 대해 각각의 Layer를 시각화하여 설명하는 방법론을 제안하였으며 이를 Layer Visualization이라고 정의하였다. 본 연구에서는 클래스 분류에서는 영향을 미치는 중요변수 확인을 위하여 Layer Visualization과 CAM을 각각 적용하였다.

본 논문은 총 5장으로 이루어져 있다. 2장에서는 관련 연구를 소개하고, 3장에서는 본 연구에서 활용한 방법론, 4장에서는 시계열 데이터 분석을 위한 실험 세팅과 분류 모델 결과를 서술하였다. 마지막으로 5장에서 결론 및 시사점을 서술하였다.

II. Preliminaries

1. Related works

1.1 Deep learning based model

최근에 딥러닝 모델은 여러 시계열 데이터 분석 연구에 많이 활용되고 있다. 이미지 분류 및 예측에 자주 사용되는 합성곱 신경망 모델과 음성인식 및 텍스트 분석에 주로 활용하는 순환신경망 모델 또한 시계열 데이터 분류를 하는데 적용되고 있다.

본 연구에서는 UCR 2018 archive에 공개된 Ford A, Ford B 데이터와 반도체 공정에서 수집된 Wafer 데이터, 라미네이터 장비에 부착된 온도 센서에서 수집된 Cycle Signal 데이터를 사용하였으며, Ford A, Ford B와 Wafer 데이터에 딥러닝 모델을 적용한 관련 연구는 Smirnov 등[6] 연구에서는 MLP는 72%, FCN은 80%의 성능을 보였다. Wenninger 등[7] ResNet을 적용하였으며 93%, 82%의 성능을 보였다. Fawaz 등[8] 연구에서는 MLP, FCN과 ResNet을 적용하였으며 73%, 90%, 92%의

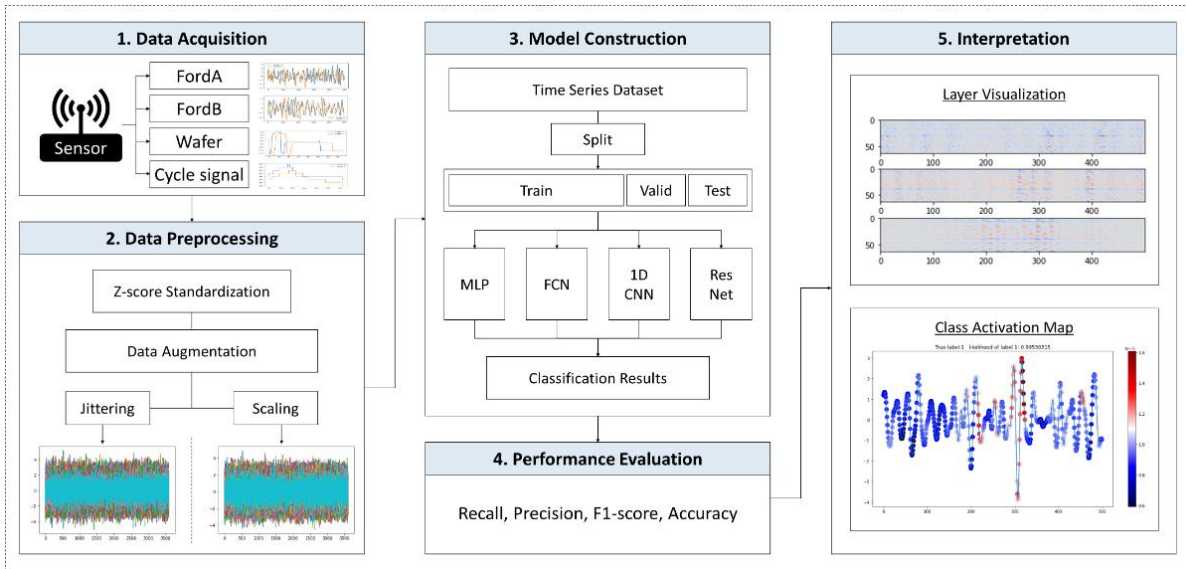


Fig. 1. Overall Process

성능을 보였다. Cycle signal 데이터는 실제 장비에서 수집되었으며 Kwon 등 [14]의 연구는 이상 탐지 및 분류를 수행하기 위해서 Long Short-Term Memory Autoencoder를 통하여 특징을 추출한 후 DBSCAN 모델로 군집화를 수행하여 Bidirectional LSTM 모델의 성능 평가에 사용하였다.

다양한 시계열 데이터를 활용한 Zhao 등[15]의 연구에서는 1차원 합성곱 신경망을 움직임 센서 데이터를 분류하였다. 컨볼루션 층을 순차적으로 늘려가며 분류 성능을 비교하였으며, 노이즈를 데이터에 추가하였을 경우 강건한지 또한 검증하였다. Kuang [16]의 연구에서는 합성곱 신경망을 UCI leaf 시계열 데이터에 적용하였다. 1차원으로 이루어져 있는 시계열 데이터를 합성곱 신경망과 Support Vector Machine Classifier를 합쳐서 분류 성능을 측정하였다. Smirnov 등[6] 연구에서는 장단기 메모리를 사용할 때 레이어를 많이 쌓는다고 하여도 성능이 올라가지 않는다는 것을 입증하였다. 순환신경망(RNN)과 장단기 메모리는 MLP와 FCN보다 성능이 좋지 않다는 것은 알 수 있었다.

1.2 XAI based method

XAI는 설명가능한 인공지능을 의미한다. 선행 연구에서는 딥러닝 방법론을 적용하여 예측 및 분류 성능을 높였으나, 딥러닝 방법론들은 복잡한 시계열 데이터의 해석을 명확하게 하지 못한다. 최근 결과에 대한 다양한 해석 방법론들의 적용되고 있다 [17].

Rojat 등[17]의 연구에서는 다양한 데이터 중에서도 대표적인 시계열 데이터의 형태를 가지고 있는 센서 데이터

를 대상으로 합성곱 신경망과 순환 신경망 모델을 사용하여 분류 성능을 측정 후 Layerwise Relevance Propagation, DeepLIFT, LIME, SHAP등을 시계열 데이터 분류를 하는데 적용하였다. 그 결과 전체 시계열 데이터에서 중요한 변수들을 확인하였다. Oviedo 등 [18]의 연구에서는 시계열 형태를 지닌 XRD(X-ray diffraction) 데이터에 대한 사후 분석 방법론으로 CAM을 사용하였으며, 시계열 분류에 적용된 합성곱 신경망에 높은 영향력을 가진 변수들을 강조하였다. Strodtthoff 등[19]의 연구에서는 PTB diagnostic 데이터셋을 대상으로 Gradient Input 방법론을 적용하여 학습이 많이 되는 노드를 추출하여 시각화하였다. Choi 등[20]의 연구에서는 합성곱신경망을 통하여 생체 신호 데이터의 특징을 추출한 후 장단기 메모리 모델로 학습을 진행하였고, Attention mechanism을 통하여 변수 간 해석력을 확인하였다. Ismail 등[21]의 연구에서는 다변량 형태의 움직임 모션 시계열 데이터에 장단기 메모리를 적용하였고 예측된 결과에 Deep SHAP을 사용하여 변수 간 중요도를 확인하였다.

III. The Proposed Scheme

본 연구에서 제안하는 전반적인 프로세스는 Fig 1과 같다. 딥러닝 모델을 사용하여 4가지 각각의 센서 데이터인 Ford A, Ford B, Wafer와 Cycle Signal을 분류하였다. 그리고 Layer Visualization과 CAM을 활용하여 해석 가능한 시계열 분석 딥러닝 프레임워크를 제안한다.

1. Data description

본 연구에서 사용한 데이터는 UCR 2018 archive에 공개된 벤치마크 데이터이다. 실제 자동차에 부착된 센서를 통하여 수집된 Ford A, Ford B 데이터셋은 자동차 엔진의 이상 유무에 따라 분류되었으며, 총 500개의 길이(feature)를 포함한다. Ford A의 훈련데이터와 평가데이터는 소음이 없는 환경에서 측정된 데이터를 사용하였으나 Ford B의 평가데이터는 소음이 심한 환경에서 측정된 데이터를 사용하였다[22]. 본 연구에서는 엔진 이상 여부를 기반으로 자동차의 특정 이상 여부를 확인하였다. Wafer 데이터셋은 인라인 프로세스 제어 공정 과정 간의 기록을 다양한 센서를 통하여 수집되었으며 Wafer 데이터 또한 공정 과정 간의 이상 유무에 따라 분류되었으며, 총 152개의 길이(feature)를 포함한다. 위에 언급된 데이터들은 분석을 위해 정규화가 적용되었다. 공개 데이터셋의 성능이 실제 데이터에 유효한지 확인을 위해 본 연구에서는 라미네이터 장비에 부착된 온도 센서에서 수집된 주기 신호(Cycle Signal) 간의 이상유무를 기반으로 분류하였으며, 총 368개의 길이(feature)를 포함시키고 있다. Fig 2 ~ 5는 각각 데이터셋들의 2개의 Class를 보여준다. Class 1은 정상을 의미하며 Class 0은 비정상을 의미한다. 본 연구에서는 분류만이 아니라 분류 결과에 대한 자세한 원인을 판별하기 위해서는 데이터 기반의 추가적인 해석이 필요하다고 판단되었다. 단순한 이진 분류 결과만이 아닌 자세한 해석 결과 및 분류의 근거를 판별하기 위해 본 연구에서는 XAI 기반 딥러닝 프레임워크를 제안하였다.

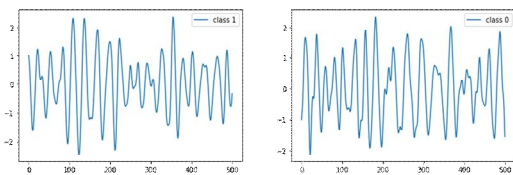


Fig. 2. Ford A

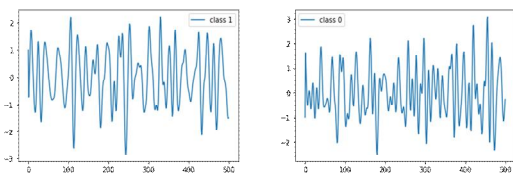


Fig. 3. Ford B

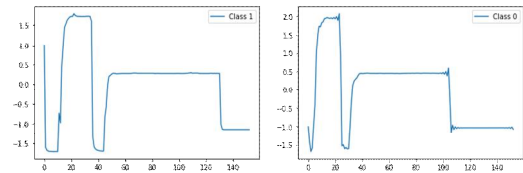


Fig. 4. Wafer

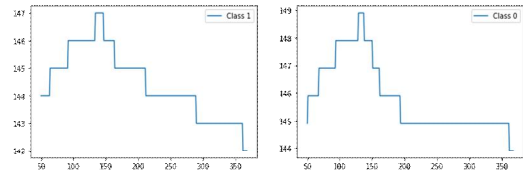


Fig. 5. Cycle Signal

2. Standardization

Z-score 표준화는 전체 데이터의 평균을 0, 표준편차를 1로 만들어 모든 변수의 범위를 조절하는 방법론이다. 이를 시계열 데이터 적용하였을 경우 이상치를 완화하고 특정 범위 안으로 데이터의 평균을 조정한다 [23]. 이는 벤치마크 데이터에 적용되었으므로 Cycle Signal 데이터도 Z-score로 표준화를 적용하였다. 아래 Z-Score 표준화 식에서 x 는 관측치, μ 는 sample의 평균, σ 는 sample의 표준편차를 의미한다.

$$Z_i = \frac{x_i - \mu}{\sigma}, \text{ where } i \in N \quad (1)$$

3. Data augmentation

Data augmentation 기법은 데이터의 양이 적을 때 유용하게 사용할 수 있다. 주로 소량의 시계열 데이터셋 일 때 Data augmentation을 진행하여 학습에 용이한 상태로 만든다 [24]. 학습 데이터의 양을 증강하기 위해 총 2가지 방법론인 Scaling과 Jittering을 적용하였다.

3.1 Scaling

Scaling은 시계열 데이터의 전역적인 규모도 혹은 강도를 조정한다. 이는 가우시안 분포 $\alpha \sim N(1, \sigma^2)$ 를 따르며, 임의의 스칼라 값을 데이터에 곱하는 Data augmentation 방법이다[25]. Ford A와 Ford B, Wafer, Cycle Signal 학습 데이터에 Scaling과 Jittering을 적용하였다.

3.2 Jittering

Jittering은 시계열 데이터에 노이즈를 추가하여 데이터를 증강하는 방법이다. 이는 $\epsilon \sim N(0, \sigma^2)$ 를 따르며 ϵ 는

원본 시계열 데이터에 추가되는 Gaussian 노이즈를 의미한다[26]. Ford A와 Ford B 데이터셋에 Jittering을 학습 데이터에 적용하였다.

4. Classification models

본 연구에서는 시계열 데이터 분류를 위해 MLP, FCN, 1D Convolutional Neural Network, ResNet등의 딥러닝 모델을 사용하였다. 구축한 모델에 대한 설명은 아래와 같다.

4.1 Multi Layer Perceptron (MLP)

다층 레이어 퍼셉트론(MLP)은 최초의 인공 신경망인 퍼셉트론을 기반으로 발전된 모델로 입력층, 은닉층과 출력층으로 이루어져 있다[27]. Wang 등[28]의 연구에서는 ReLU를 적용하여 MLP 모델을 구성하였으나 본 연구에서는 Leaky ReLU를 사용하여 기존에 쓰인 ReLU의 단점인 그래디언트의 해당 가중치가 학습이 안되는 점을 보완하였다. Fig. 6는 본 연구에서 제안한 MLP의 모델 구조이다.

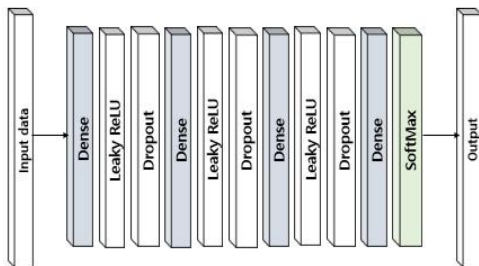


Fig. 6. Multi Layer Perceptron

4.2 Fully Convolutional Neural Network (FCN)

완전 합성곱 신경망 구조는 전결합 신경망이 없으며 합성곱 신경망으로만 구성된다 [28]. 본 연구에서는 시계열 데이터 분류를 위하여 1차원의 합성곱 신경망 연산을 이용하였다. 추가로 해석 가능한 시계열 분류 모델을 구축하기 위해 완전 합성곱 신경망의 마지막 레이어 전에 전역 평균 풀링(Global Average Pooling) 레이어를 사용하였다. Fig. 7는 본 연구에서 제안한 완전 합성곱 신경망의 모델 구조이다.

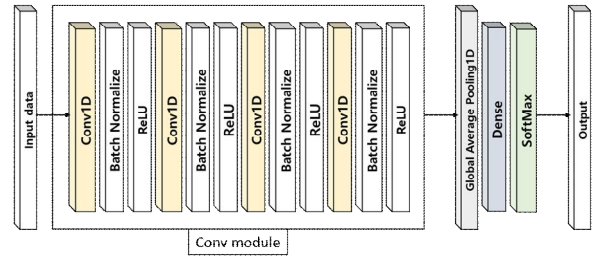


Fig. 7. Fully Convolutional Neural Network

4.3 1D Convolutional Neural Network (1D CNN)

1차원 합성곱 신경망은 필터가 한 방향으로 움직이는 합성곱 신경망이다 [29]. 1D CNN 또한 CAM을 적용하기 위하여 마지막 레이어 전에 Global Average Pooling을 사용하였다. 추가로 본 연구에서는 모델 학습의 과적합을 방지하기 위하여 각각 은닉 레이어에서의 연산 후에 drop out을 적용하였다. Fig. 8는 본 연구에서 제안한 1차원 합성곱 신경망의 모델 구조를 나타낸다.

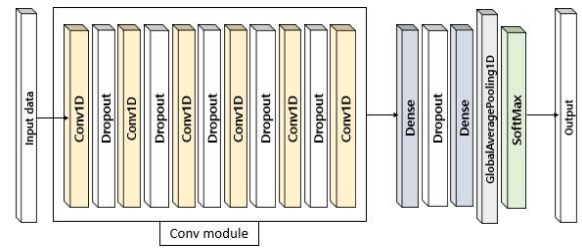


Fig. 8. 1D CNN

4.4 Residual Network (ResNet)

잔차 네트워크의 핵심은 오차 역전파 학습 시 신경망이 깊어짐에 따라 기울기가 소실된다는 단점을 보완하기 위해, 더 깊은 networks를 효과적으로 학습시키도록 만들었다[30]. 본 연구에서는 ResNet을 제안한 [8]의 연구보다 더 좋은 결과를 확인하였다. Fig. 9는 본 연구에서 제안한 ResNet의 모델 구조를 나타낸다.

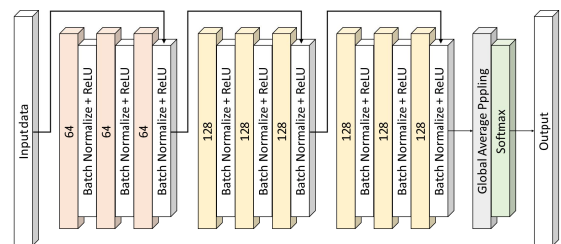


Fig. 9. ResNet

4.5 Class Activation Map (CAM)

CAM은 딥러닝 모델의 분류 결과에 대한 중요한 특징을 확인하는 방법이다. 컴퓨터 비전 분야에서의 CAM은 특정 클래스를 구분하는 이미지내의 정보를 파악한다. 본 연구에서는 시계열 데이터에 CAM을 적용하였으며, 과정은 다음과 같다. CAM은 마지막 레이어에 Global Average Pooling를 통해 가중치를 학습하고 각 feature map(채널)의 가중치 값들의 평균을 산출하게 된다. 그리고 분류 결과에 따라서 CAM에 활용되는 가중치 (weight)가 달라진다. 마지막 단계인 convolution feature map은 정보량이 많으므로 원인 분석을 용이하게 한다 [13].

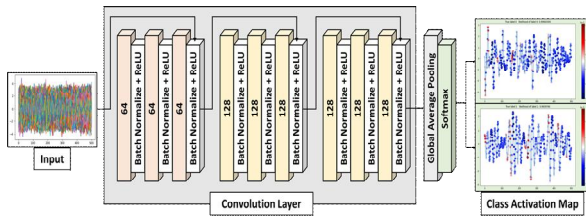


Fig. 10. Class Activation Map

본 연구에서는 CAM을 활용하여 센서 기반의 시계열 데이터 분류에 중요 영향을 끼치는 구간 변수를 확인하였다. Fig. 10은 본 연구에서 활용된 CAM의 예시를 보여주며 식 (2)를 이용하여 산출한다. 분석하고자 하는 시계열 데이터가 구축한 모델에 입력이 되었을 때 중요 구간을 시각적으로 구분하는 방법을 제안하였다. (2)번 수식에서 g_c 는 주어진 시계열 데이터를 보았을 때 Class를 찾는 것을 의미하며, $S_k(x)$ 는 마지막 Conv layer에서의 활성화되는 filter의 개수를, w_k^c 에서 w 는 마지막 softmax의 가중치를 의미하는데, 이 중에서 c 는 Class를 k 는 filter의 출력값이다.

$$g_c = \sum_k w_k^c \sum_x S_k(x) = \sum_k \sum_x w_k^c S_k(x) \quad (2)$$

IV. Experiments

1. Experiment setting

1.1 Experiment environment

본 연구에서는 Window 10 환경에 파이썬 3.6 텐서플로우 2.4.1과 케라스(keras) 2.4.3을 사용하였다. CPU는 Intel(R) Core(TM) i7-7700 (6-코어), GPU는 엔비디아의 RTX 3060, RAM은 64GB 환경에서 실험을 수행하였다.

1.2 Hyperparameter setting

본 연구의 실험에서 사용된 모든 딥러닝 모델에 Adam optimizer을 적용하였으며 손실함수는 Categorical cross entropy loss를 사용하였다. epoch은 500으로 설정하였고, batch size의 크기는 모델별로 최대 성능 산출하는 것에 비례하여 다르게 설정되었다.

1.3 Evaluation

본 연구에서 제안한 딥러닝 모델의 분류 성능 검증을 위하여 Table 1.에서 정의된 혼동행렬 지표를 활용하였다. 대표적인 머신러닝 및 딥러닝 분류 성능의 평가 지표인 정확도(Accuracy), 정밀도(Precision), 재현율(Recall), F1 점수(F1 score)는 식 (3) ~ (6)을 이용하여 계산한다.

Table 1. Confusion matrix

Confusion Matrix		Predicted Class	
		Positive	Negative
Actual Class	Positive	TP	FP
	Negative	FN	TN

$$Accuracy = \frac{TN + TP}{(TN + FP) + (FN + TP)} \quad (3)$$

$$Precision = \frac{TP}{TP + FP} \quad (4)$$

$$Recall = \frac{TP}{TP + FN} \quad (5)$$

$$F1 - Score = \frac{2 \times precision \times recall}{precision + recall} \quad (6)$$

1.4 Classification model setup

본 연구에서 MLP, FCN, 1D CNN과 ResNet등을 사용하여 분류 성능을 평가하였다. 분류 모형 훈련을 위해 전체 4921개의 Ford A 데이터 중 3601개를 훈련데이터로 1320개를 평가데이터로 사용하였다. 반면 전체 4446개의 Ford B 데이터는 3636개를 훈련데이터로 810개를 평가데이터로 사용하였다. 반도체 공정에서 사용된 Wafer 데이터는 전체 7164개 중 5731개를 훈련데이터로 1433개를 평가데이터로 구성하였으며, 전체 800개의 Cycle Signal 데이터는 400개를 훈련데이터로 400개를 평가데이터로 사용하였다. 추가로 Data augmentation을 적용하여 Ford A와 Ford B, Wafer와 Cycle Signal 데이터는 학습 데이

터를 2배로 증강하였으며, Ford A는 7202개, Ford B는 7272개, Wafer는 11462개, Cycle Signal는 800개의 데이터를 훈련데이터로 사용하였다. K-fold 교차 검증을 사용하였고 K는 5로 선정하여 모델의 성능을 검증하였다.

Table 2. Data description

Data	Augmentation	Train	Test	Total
Ford A	Before	3601	1320	4921
	After	<u>7202</u>	1320	8522
Ford B	Before	3636	810	4446
	After	<u>7272</u>	810	8082
Wafer	Before	5731	1433	7164
	After	<u>11462</u>	1433	12895
Cycle signal	Before	400	400	800
	After	<u>800</u>	400	1200

2. Experiment results

2.1 Comparative results

Table 3. Ford A

Ford A (Accuracy)	MLP	FCN	1D CNN	Res Net
Smirnov, 2018 [8]	0.72	0.80	-	-
Wenninger, 2019 [9]	-	-	-	0.93
Fawaz, 2019 [10]	0.73	0.90	-	0.92
Our method	<u>0.84</u>	<u>0.90</u>	<u>0.93</u>	<u>0.93</u>

Table 4. Ford B

Ford B (Accuracy)	MLP	FCN	1D CNN	Res Net
Smirnov, 2018 [6]	0.72	0.79	-	-
Wenninger, 2019 [7]	-	-	-	0.82
Fawaz, 2019 [8]	0.60	0.88	-	<u>0.91</u>
Our method	<u>0.74</u>	<u>0.91</u>	<u>0.80</u>	0.83

Table 6. Comparison between Original Dataset and Augmented Dataset experiment results (Ford A)

Ford A		MLP	FCN	1DCNN	ResNet
Original method	Accuracy	0.842	0.904	0.937	0.936
	Precision	0.835	0.902	0.924	0.935
	Recall	0.842	0.904	0.937	0.923
	F1 Score	0.835	0.902	0.924	0.925
Scaling (Augmentation)	Accuracy	0.865	0.915	0.942	0.944
	Precision	0.843	0.913	0.925	0.935
	Recall	0.865	0.915	0.943	0.944
	F1 Score	0.854	0.913	0.935	0.923
Jittering (Augmentation)	Accuracy	0.862	0.914	0.945	<u>0.953</u>
	Precision	0.863	0.911	0.944	<u>0.945</u>
	Recall	0.855	0.914	0.931	<u>0.953</u>
	F1 Score	0.854	0.912	0.933	<u>0.952</u>

Table 5. Wafer

Wafer (Accuracy)	MLP	FCN	1D CNN	Res Net
Smirnov, 2018 [6]	0.97	0.96	-	-
Wenninger, 2019 [7]	-	-	-	0.99
Fawaz, 2019 [8]	0.99	0.99	-	<u>0.99</u>
Our method	0.99	0.99	0.99	1.00

Table 3, Table 4와 Table 5는 본 연구에서 제안하는 기본 모델들과 선행 연구에서 제안하는 모델들의 성능 및 결과 비교를 하였다. 본 연구에서 제안한 모델들은 선행 연구에서 제안하는 모델들보다 분류 성능이 향상됨을 확인하였다. Ford A에서는 정확도를 기준으로 선행 연구와 비교하면 MLP는 약 11~12%, 본 연구에서 쓰인 중점모델인 ResNet은 약 1~2% 증가 향상되었다. Ford B에서 MLP는 14%, FCN은 11% 향상되었다. Wafer에서는 ResNet은 약 1% 증가한 것을 확인하였다. 반면 Ford B의 ResNet에서는 Wenninger [7]에 비해 높은 성능을 얻었지만 Fawaz [8]보다는 낮은 성능을 보였다. 결과를 분석하기 위하여 본 연구에서는 CAM을 적용하여 시각적으로 해석하였다.

2.2 Data augmentation evaluation and results

Ford A와 Cycle Signal의 실험 결과는 Table 6와 Table 7과 같으며 Ford B와 Wafer의 실험 결과는 Table 8과 9에서 확인할 수 있다. Data augmentation의 적용 이전과 이후의 성능을 비교하였다. 본 연구에서 제안하는 모델과 Data augmentation을 적용한 모델들의 우수한 성능을 확인할 수 있었다. Ford A 데이터에 Scaling을 적용한 1D CNN과 ResNet의 전체적인 성능은 1% 가량 향상되었으나 Jittering을 적용하였을 때 2~3% 성능이 향상됨을 확인하였다.

Table 7. Comparison between Original Dataset and Augmented Dataset experiment results (Cycle Signal)

Cycle Signal		MLP	FCN	1DCNN	ResNet
Original method	Accuracy	0.985	0.987	0.995	0.996
	Precision	0.995	0.997	0.995	0.996
	Recall	0.952	0.884	0.887	0.995
	F1 Score	0.997	0.931	0.935	0.995
Scaling (Augmentation)	Accuracy	0.994	0.996	0.996	0.998
	Precision	0.997	0.996	0.994	0.998
	Recall	0.998	1.000	1.000	0.997
	F1 Score	0.998	0.995	0.994	0.997
Jittering (Augmentation)	Accuracy	0.999	0.999	0.999	0.999
	Precision	0.996	0.994	0.992	0.999
	Recall	1.000	1.000	1.000	1.000
	F1 Score	0.992	0.994	0.993	0.998

Table 8. Comparison between Original Dataset and Augmented Dataset experiment results (Ford B)

Ford B		MLP	FCN	1DCNN	ResNet
Original method	Accuracy	0.734	0.912	0.805	0.836
	Precision	0.725	0.913	0.792	0.835
	Recall	0.734	0.912	0.805	0.824
	F1 Score	0.725	0.913	0.792	0.825
Scaling (Augmentation)	Accuracy	0.745	0.924	0.816	0.844
	Precision	0.723	0.925	0.811	0.835
	Recall	0.741	0.924	0.816	0.844
	F1 Score	0.733	0.925	0.802	0.835
Jittering (Augmentation)	Accuracy	0.747	0.927	0.835	0.846
	Precision	0.745	0.925	0.832	0.833
	Recall	0.736	0.927	0.825	0.846
	F1 Score	0.732	0.925	0.824	0.842

Table 9. Comparison between Original Dataset and Augmented Dataset experiment results (Wafer)

Wafer		MLP	FCN	1DCNN	ResNet
Original method	Accuracy	0.995	0.997	0.998	0.999
	Precision	0.995	0.997	0.996	0.999
	Recall	0.985	0.985	0.988	0.987
	F1 Score	0.995	0.997	0.996	0.998
Scaling (Augmentation)	Accuracy	0.996	0.997	0.998	1.000
	Precision	0.994	0.998	0.996	1.000
	Recall	0.985	0.984	0.998	1.000
	F1 Score	0.986	0.982	0.996	1.000
Jittering (Augmentation)	Accuracy	0.997	0.998	0.999	1.000
	Precision	0.994	0.996	0.994	1.000
	Recall	0.985	0.987	0.999	1.000
	F1 Score	0.995	0.997	0.994	1.000

Ford A 데이터는 ResNet 모델은 다른 모델과 비교하였을 때 Jittering의 성능이 정확도 95%, 정밀도 94%, 재현율 95%, F1 점수 95%로 Jittering을 적용하였을 때 정확도는 2% 증가하여 높은 성능을 확인하였다. Cycle Signal 데이터는 ResNet의 모델이 다른 모델과 비교하였을 때 Jittering과 scaling에서 모두 성능이 높았으며 정확도 99%, 정밀도 99%, 재현율 99%, F1 점수 99%로 Jittering을 적용하였을 때 재현율은 1% 증가하였다.

2.3 XAI results

Fig. 11은 딥러닝 모델들을 적용하여 클래스 분류 결과 및 중요변수를 CAM을 통하여 시각화한 결과이다. 색의 명암에 따라서 변수 중요도를 확인할 수 있었다. 진한 빨간색을 지니고 있을수록 그 시점의 중요도가 더 높음을 의미하며 클래스 분류에 중요 영향을 미치는 특징점임을 의미한다. 본 연구에서는 합성곱 신경망을 통과하여 각각은닉층에서 학습되는 과정을 레이어 시각화(Layer Visualization)를 통하여 확인하였고, CAM에 결과를 확인하였다. Layer 단계별 학습되는 과정을 확인할 수 있었다.

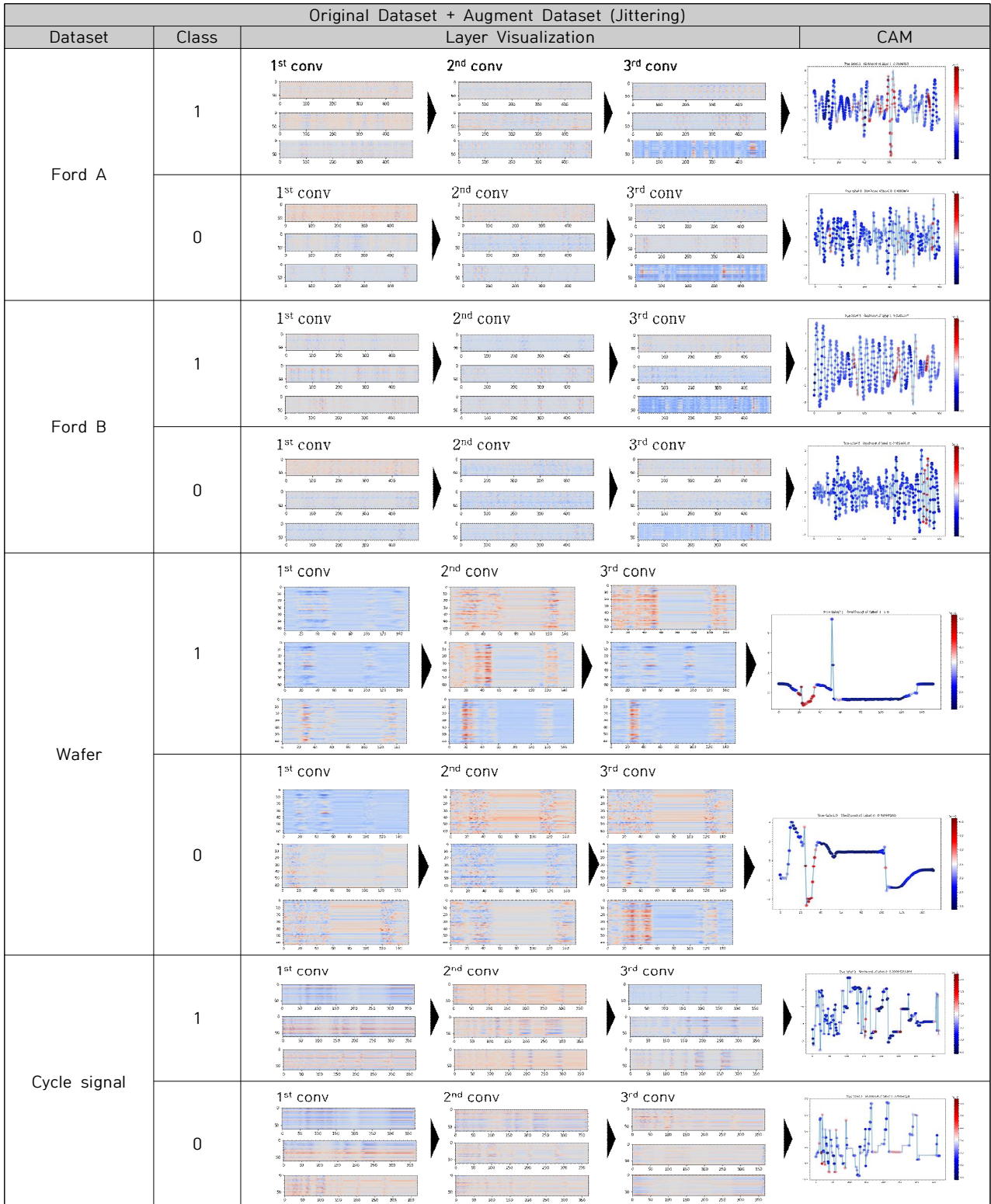


Fig. 11. XAI result

Fig. 11에서는 Layer Visualization이 ResNet의 3rd Conv layer를 기준으로 학습 완료된 후에 CAM을 통한 분류한 결과 및 중요변수 추출을 시각적으로 표기하였다. Layer Visualization을 통해 합성곱 신경망을 통과할수록

중요 구간이 선명해짐을 확인하였고, 이러한 결과를 바탕으로 CAM에서 최종적인 중요 구간을 시각화하였다. Data augmentation 방법론인 Scaling과 jittering으로 학습 데이터의 양을 2배로 증가시킨 후 Layer Visualization을 적

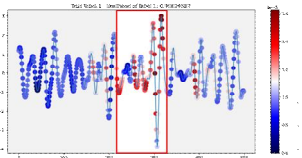
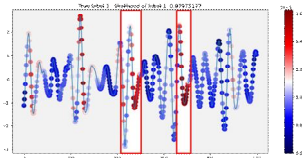
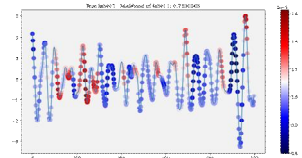
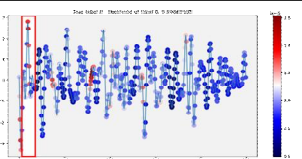
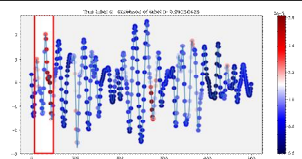
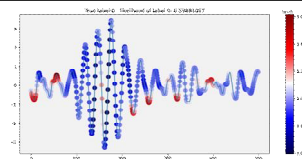
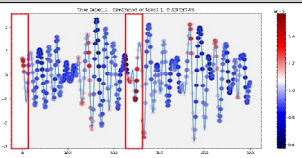
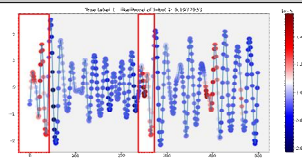
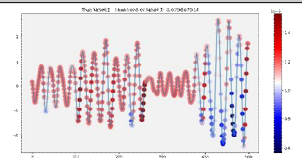
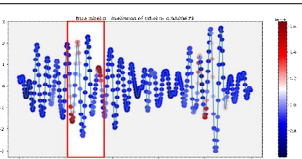
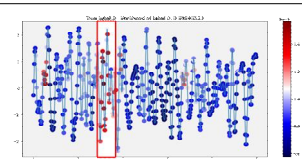
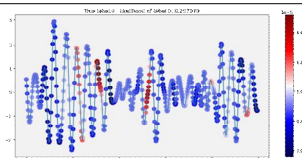
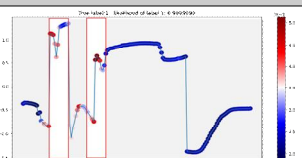
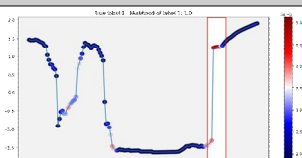
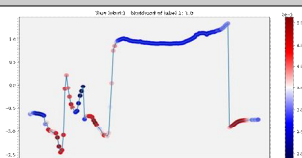
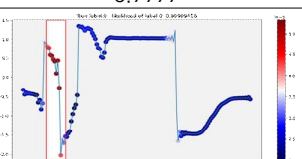
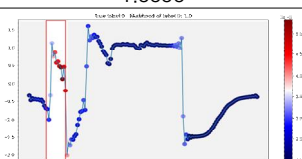
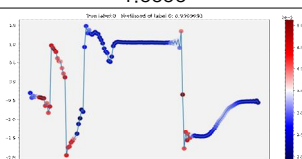
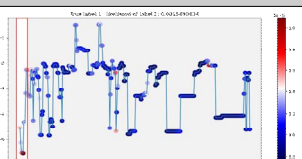
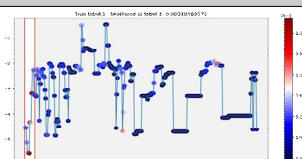
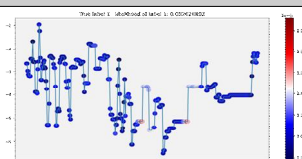
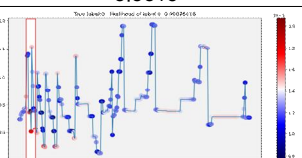
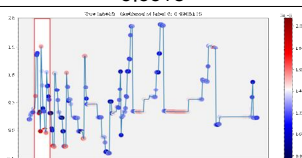
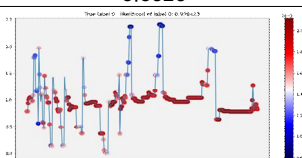
Class	Well Classified		Poorly Classified
	Ford A		
1			
Likelihood	0.9963	0.9798	0.7381
0			
Likelihood	0.9966	0.9926	0.5989
Ford B			
1			
Likelihood	0.9939	0.9978	0.0799
0			
Likelihood	0.9811	0.9999	0.2579
Wafer			
1			
Likelihood	0.9999	1.0000	1.0000
0			
Likelihood	0.9994	1.0000	0.9993
Cycle Signal			
1			
Likelihood	0.0015	0.0018	0.0026
0			
Likelihood	0.9987	0.9986	0.9984

Fig. 12. Classification interpretation

용하였다. 데이터에 Jittering을 적용하였을 때 중요변수 추출이 Original과 Scaling에 비하여 좋은 것을 확인하였다.

Ford A의 Jittering에서 정상 클래스를 의미하는 Class 1은 Layer Visualization을 적용하였을 때 전체 시계열에 길이 중 460에 해당하는 구간이 가장 중요한 변수인 것을 확인하였다. 반면에 Class 0을 비정상 클래스를 분류하는데 전체 시계열 길이 중 50에 해당하는 구간이 가장 중요한 변수로 확인되었고 CAM에서 동일하였다.

Ford B의 Jittering에서 정상 클래스를 의미하는 Class 1은 Layer Visualization을 적용하였을 때 전체 시계열에 길이 중 440~450에 해당하는 구간이 가장 중요한 변수로 CAM에서 또한 중요한 구간인 것을 확인했으며 비정상 클래스인 Class 0을 분류하는 구간은 440에 해당하는 구간이 정상과 비정상 여부를 분류하는데 중요한 변수인 것을 확인하였다.

Wafer의 Jittering에서 정상 클래스를 의미하는 Class 1과 비정상 클래스인 Class 0에 Layer Visualization을 적용하였을 때 전체 시계열에 길이 중 모두 20~35에 해당하는 구간이 가장 중요한 변수로 CAM에서 또한 중요한 구간인 것을 확인하였다.

Cycle Signal의 Jittering에서 정상 클래스를 의미하는 Class 1은 Layer Visualization을 적용하였을 때 전체 시계열에 길이 중 150, 250에 해당하는 구간이 가장 중요한 변수로 CAM에서 또한 중요한 구간인 것을 확인했으며 비정상 클래스인 Class 0을 분류하는 구간은 20~50에 해당하는 구간이 정상과 비정상 여부를 분류하는데 중요한 변수인 것을 확인하였다.

Likelihood는 가능도를 의미하며, 이는 클래스가 관측되었을 때, 클래스가 어떤 확률분포를 기준으로 분류 성능을 가져왔을지에 대한 확률을 의미한다. 분류가 잘된 결과들의 기준은 Likelihood가 높은 것에 근거하였다.

2개의 정상, 비정상 클래스를 분류하는데 Likelihood값은 모두 90%을 능가하므로 클래스 분류하는데 좋은 성능을 보이는 것을 확인했으며, 정확한 클래스 분류하는데 중요한 변수가 속한 시계열 구간을 통해 Class 1은 정상을 의미하며 Class 0은 비정상을 구분하는 요인을 Layer Visualization과 CAM을 통하여 해석하였다.

정상과 비정상 클래스 분류는 Layer Visualization에서 모델이 학습되는 과정을 기반으로 클래스 분류에 기준이 되는 시계열 구간인 중요변수를 확인하였다. 본 연구에서는 4개의 시계열 데이터에 대한 딥러닝 분류 모델을 구축하고 CAM과 Layer Visualization의 결과를 종합하여 정상과 비정상 대한 자세한 분류 근거를 확인하였다.

2.4 Classification interpretation

CAM을 통한 분류 성능 결과는 Likelihood를 통하여 확인할 수 있는데, Likelihood값이 높을수록 분류 결과를 더욱 신뢰할 수 있다. Fig. 12는 정상 및 비정상인 클래스를 layer visualization을 통하여 분류가 전체 시계열 길이 중 중요한 구간을 잘된 사례와 안된 사례를 나타내고 있다. 기존 연구는 성능에 대해 초점을 두었지만, 본 연구는 높은 성능뿐 만이 아닌 결과에 대한 시각적 해석을 제시하는 차별성을 두었다. 이러한 제시된 방법으로 시계열 데이터에 대한 원인 분석이 가능하며 추후 전문가의 판단을 복합적으로 고려하여 모델의 가용성을 검증할 수 있을 것이라 기대된다. Fig. 12에서 Ford A 데이터가 잘 분류된 Class 1과 0을 기준으로 Likelihood 값은 모두 97% 이상이었으며, Class 1과 0이 제대로 분류되지 않은 것들은 73%, 59%인 것을 확인하였다. Ford B 데이터의 Likelihood 값을 확인하였을 때 Class 1과 0이 잘 분류된 사례들은 모두 98% 이상인 것을 확인하였으며, Class 1과 0 모두 제대로 분류되지 않은 것들은 25% 이하로 Ford A 데이터와 비교하였을 때 제대로 Class를 분류하지 못한 사례들의 Likelihood값이 큰 차이를 보이는 것을 확인하였다.

Wafer 데이터가 잘 분류된 Class 1과 0을 기준으로 Likelihood 값은 모두 99% 이상이었으며, Class 1과 0이 제대로 분류되지 않은 것들은 특징 구간을 제대로 나타내지 못하는 것을 확인하였다. 반면에 Cycle Signal 데이터는 Class 0의 likelihood 값은 99% 이상이었으나 Class 1의 likelihood 값을 분류하지 못하는 것을 확인할 수 있었다. Ford A 데이터의 분류 결과에 대해 높은 Likelihood를 가지는 CAM 결과들을 확인하였다. Class 1은 220~310 시계열 구간, Class 0은 10~40 시계열 구간이 클래스를 구분하기 위한 중요 영향을 미치는 변수이다. Ford B는 같은 방법으로 Class 1은 0~20, 230~260의 시계열 구간, Class 0은 100~200 시계열 구간이 클래스 분류에 중요 영향을 미치는 것으로 확인하였다. Wafer에서의 Class 1은 30~50, 120의 시계열 구간, Class 0은 20의 시계열 구간만이 클래스 분류에 중요 영향을 지니는 변수로 확인했다. 반면 실제 장비에서 수집된 Cycle Signal 데이터에서의 Class 1과 Class 0은 10~30 시계열 구간이 유사하게 중요 영향을 보여주는 변수임을 확인하였다. Ford A, Ford B, Wafer와 Cycle Signal 모두 분류가 잘 되지 않은 결과들은 잘 분류된 구간들과 비교해 보았을 때 공통된 특징 구간이 없으며 Likelihood의 값이 낮은 것을 확인할 수 있다.

V. Conclusion

본 논문에서는 차량에 부착된 센서 데이터, 반도체 공정 데이터 및 주기 신호 데이터들을 기반으로 딥러닝 모델을 이용하여 시계열 분류 성능 측정 후 해석 가능한 방법론을 제시하였다. 딥러닝 모델은 MLP, FCN, 1D CNN, ResNet 등 4개의 모델을 사용하였으며 ResNet 모델이 가장 우수한 성능을 보임을 확인하였다. 적은 양의 데이터 불균형 문제를 해결하기 위하여 Data augmentation을 진행하여 학습 데이터 크기를 2배로 늘렸으며, 모델의 성능이 향상됨을 확인하였다. 해석 가능 방법론은 Layer Visualization을 통하여 Convolutional Layer에서 모델이 학습되는 과정을 확인하고 CAM을 통하여 센서 데이터 중 중요한 특징 구간을 시각적으로 확인하였다.

본 연구에서 제안하는 설명 가능한 딥러닝 기반 프레임워크는 센서 데이터의 특징을 내포하는 중요한 구간을 시각적으로 확인할 수 있도록 하며, 해석 결과를 종합하여 각 데이터의 전반적인 특징 구간들을 확인할 수 있었다. 이는 단순한 분류 및 예측뿐만이 아닌 결과에 대한 다양한 해석을 요구하는 데이터 기반 센서 데이터 관리에 적용할 수 있으며, 스마트팩토리 분야에서 수집되는 센서 데이터 뿐만 아니라 다양한 도메인에서 적용할 수 있을 것으로 기대된다. 더하여 복잡한 시계열 데이터 분석에 대한 해석 기법의 부재에 관한 연구 공백을 채울 것으로 기대된다.

ACKNOWLEDGEMENT

This work was supported by Brain Korea 21.

REFERENCES

- [1] C. Oh, S. Han and J. Jeong, "Time-series data augmentation based on interpolation." *Procedia Computer Science* 175, pp. 64-71, 2020.
- [2] C. L. Yang, Z. X. Chen and C. Y. Yang, "Sensor classification using convolutional neural network by encoding multivariate time series as two-dimensional colored images." *Sensors* 20.1, pp. 168, 2019.
- [3] H. Ismail Fawaz et al., "Inceptiontime: Finding Alexnet for Time Series Classification." *Data Mining and Knowledge Discovery*, vol. 34, no. 6, Springer, 2020, pp. 1936-62.
- [4] Y. S. Jeong, M. K. Jeong and O. A. Omitaomu, "Weighted dynamic time warping for time series classification." *Pattern recognition* 44.9, 2011, pp. 2231-2240.
- [5] G. S. Ahn, H. C. Lee and S. Hur, "Feature Selection Method for Multivariate Time Series Data Classification." *Journal of the Korean Institute of Industrial Engineers* 43.6, 2017, pp. 413-421.
- [6] D. Smirnov and E. M. Nguifo, "Time Series Classification with Recurrent Neural Networks." *Advanced Analytics and Learning on Temporal Data*, vol. 8, 2018.
- [7] M. Wenninger et al., "Timage-A Robust Time Series Classification Pipeline." *International Conference on Artificial Neural Networks*, Springer, 2019, pp. 450-61.
- [8] H. Ismail Fawaz et al., "Deep Learning for Time Series Classification: A Review." *Data Mining and Knowledge Discovery*, vol. 33, no. 4, Springer, 2019, pp. 917-63.
- [9] A. Preece et al., "Stakeholders in Explainable AI." *ArXiv Preprint ArXiv:1810.00184*, 2018.
- [10] J. Zhu et al., "Explainable AI for Designers: A Human-Centered Perspective on Mixed-Initiative Co-Creation." *2018 IEEE Conference on Computational Intelligence and Games (CIG)*, IEEE, 2018, pp. 1-8.
- [11] M. Van Lent, W. Fisher and M. Mancuso, "An explainable artificial intelligence system for small-unit tactical behavior." *Proceedings of the national conference on artificial intelligence*. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2004.
- [12] D. Gunning and D. Aha, "DARPA's Explainable Artificial Intelligence (XAI) Program." *AI Magazine*, vol. 40, no. 2, 2019, pp. 44-58.
- [13] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva and A. Torralba, "Learning Deep Features for Discriminative Localization." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 2921-29.
- [14] S. H. Kwon, M. J. An, H. C. Lee, "Fault Detection and Classification of Process Cycle Signals Using Density-based Clustering and Deep Learning." *Journal of the Korean Institute of Industrial Engineers* 44.6, 2018, pp. 475-482.
- [15] B. Zhao et al., "Convolutional neural networks for time series classification." *Journal of Systems Engineering and Electronics* 28.1, 2017, pp. 162-169.
- [16] D. Kuang, "A 1d convolutional network for leaf and time series classification." *arXiv preprint arXiv:1907.00069*, 2019.
- [17] T. Rojat, R. Puget, D. Filliat, J. Del Ser, R. Gelin and N. Díaz-Rodríguez, "Explainable artificial intelligence (xai) on timeseries data: A survey." *arXiv preprint arXiv:2104.00950*, 2021.
- [18] F. Oviedo et al., "Fast and Interpretable Classification of Small X-Ray Diffraction Datasets Using Data Augmentation and Deep Neural Networks." *Npj Computational Materials*, vol. 5, no. 1, Nature Publishing Group, 2019, pp. 1-9
- [19] N. Strodthoff et al. "Deep Learning for ECG Analysis:

- Benchmarks and Insights from PTB-XL.” IEEE Journal of Biomedical and Health Informatics, vol. 25, no. 5, IEEE, 2020, pp. 1519–28.
- [20] K. S. Choi et al., “Prediction of IDH Genotype in Gliomas with Dynamic Susceptibility Contrast Perfusion MR Imaging Using an Explainable Recurrent Neural Network.” *Neuro-Oncology*, vol. 21, no. 9, Oxford University Press US, 2019, pp. 1197–209.
- [21] A. A. Ismail, M. Gunady, H. Corrada Bravo and S. Feizi, “Benchmarking Deep Learning Interpretability in Time Series Predictions.” *ArXiv Preprint ArXiv:2010.13924*, 2020.
- [22] G. Forestier, F. Petitjean, H. A. Dau, G. I. Webb, and E. Keogh, “Generating Synthetic Time Series to Augment Sparse Datasets.” 2017 IEEE International Conference on Data Mining (ICDM), IEEE, 2017, pp. 865–70.
- [23] D. Q. Goldin, and C. P. Kanellakis, “On Similarity Queries for Time-Series Data: Constraint Specification and Implementation.” *International Conference on Principles and Practice of Constraint Programming*, Springer, 1995, pp. 137–53.
- [24] T. Um, F. M. Pfister, D. Pichler, S. Endo, M. Lang and D. Kulić, “Data Augmentation of Wearable Sensor Data for Parkinson’s Disease Monitoring Using Convolutional Neural Networks.” *Proceedings of the 19th ACM International Conference on Multimodal Interaction*, 2017, pp. 216–20.
- [25] Q. Wen, L. Sun, F. Yang, X. Song, J. Gao, X. Wang and H. Xu, “Time Series Data Augmentation for Deep Learning: A Survey.” *ArXiv Preprint ArXiv:2002.12478*, 2020.
- [26] B. K. Iwana and S. Uchida, “An Empirical Survey of Data Augmentation for Time Series Classification with Neural Networks.” *Plos One*, vol. 16, no. 7, Public Library of Science San Francisco, CA USA, 2021, p. e0254841.
- [27] F. Rosenblatt, *Principles of Neurodynamics. Perceptrons and the Theory of Brain Mechanisms*. Cornell Aeronautical Lab Inc Buffalo NY, 1961.
- [28] Z. Wang, W. Yan and T. Oates, “Time Series Classification from Scratch with Deep Neural Networks: A Strong Baseline.” 2017 International Joint Conference on Neural Networks (IJCNN), IEEE, 2017, pp. 1578–85.
- [29] W. Tang, G. Long, L. Liu, T. Zhou, J. Jiang and M. Blumenstein, “Rethinking 1d-Cnn for Time Series Classification: A Stronger Baseline.” *ArXiv Preprint ArXiv:2002.10061*, 2020.
- [30] K. He, X. Zhang, S. Ren and J. Sun, “Deep Residual Learning for Image Recognition.” *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–78.

Authors



Youngjun Jang received the B.S. degrees in Finance from University of Connecticut, U.S., in 2017, and M.S. in Industrial Management Engineering from Korea University, Korea in 2022 respectively.

Youngjun Jang is currently a student in the Department of Industrial Management Engineering, Korea University. He is interested in Time Series Classification, Deep learning, and XAI.



Jiho Kim received the B.S. degree in Industrial and Information Systems Engineering from Seoul National University of Science and Technology, Korea, in 2015. He is currently a Integrated M.S/Ph.D

Student in the department of Industrial Management Engineering at Korea University. He is interested in Data Mining, Artificial Intelligence and Text Mining.



Hongchul Lee received the B.S. degree in Industrial Engineering from Korea University, Korea, in 1983, M.S. degree in Industrial Engineering from Texas Arlington University, U.S. in 1988 and he received Ph.D. degree in

Industrial Engineering from Texas A&M University, U.S. respectively. Dr. Lee joined the faculty of the Department of Industrial Management Engineering at Korea University, Seoul, Korea, in 1996. He is currently a Professor in the Department of Industrial Management Engineering, Korea University. He is interested in Artificial Intelligence, Manufacturing Engineering System, and Simulation.