
Predictive maintenance system for production lines in manufacturing: A machine learning approach using IoT data in real-time

Ayvaz.S, Alpay.K

Expert Systems With Applications (2021)

vol. 173, ISSN : 0957-4174, DOI: 10.1016/j.eswa.2021.114598

Smart Factory 논문리뷰
신승준 교수님
한양대학교 산업 데이터 엔지니어링학과
석사과정 강병모



한양대학교

Background & Contributions

인용 네트워크

Web of Science 핵심 컬렉션

75

인용

 인용 알림 만들기




75

인용 횟수: 모든 데이터 베이스

62

인용 문헌

[관련 레코드 보기](#)

 2022년 November/December부로, 이 **Highly Cited Paper**의 인용 횟수가 분야와 출판 연도에 대해 인용 빈도가 높은 임계값을 기반으로 **Engineering** 학술 분야에서 **상위 1%**에 올랐습니다.

➤ Why is the paper highly cited?

1. 처음으로 실제 공장에서 AI 기반 예측 유지 보수 시스템 구현

→이전 연구에서 제안된 많은 접근 방식이 있었지만, 예기치 못한 문제로 인해 실패

→효과적이고 고차원의 데이터를 실시간으로 빠르게 확장

2. End to End 방식 적용

→이전 연구/방법은 구성요인 별 고장 예측 진행

→문제점 : 전체 시스템을 고려하지 않은 유지보수로 인한 재고장으로 인해 추가 비용 발생

→모든 구성 요소를 통합한 End to End 방식을 통해 재고장으로 인한 라인 가동 중단을 예방

3. 제조 장비의 IoT 센서로 수집된 데이터 사용

→생산 과정을 디지털화하고 전체 라인의 성능 개선을 위한 데이터 분석을 활용할 수 있게 함

Background & Contributions

Motivation

➤ 제조 시스템에서 생산 중단을 방지하기 위해서는 어떻게 해야 할까?

✓ 제조 산업에서의 생산 중단

-장비 고장, 유지 보수 필요로 생산 라인이 중단 될 수 있음

→생산 중단 1. 원인 파악 및 수리를 위한 다운 타임 발생

2. 다운타임으로 인한 생산 손실

3. 시스템 복구까지의 시간 손실

→시간 및 비용 발생

→생산 중단 전 **예측**해서 장비를 **유지보수** 하자!

<하이테크 물류로봇: Digit>



➤ 예지보전 시스템(Predictive maintenance: PdM)

✓ 목적 : 부품 및 장비의 장애 발생전 오류를 예측하여 생산 중단을 방지하는 것

✓ 과거의 방법 : 부품 고장 발생 후 또는 부품 교체 주기 이후 교체

→다운타임 발생으로 인한 시간 및 비용 발생

- 기대효과 1. 예기치 못한 장애 방지로 다운타임 최소화

2. 제품/부품의 유효 수명 극대화

→ 다운 타임 최소화 및 유효 수명 극대화를 통한 시간 및 비용 최소화

Data from IoT in Order to PdM

➤ IoT으로 수집한 데이터를 통한 예지보전

✓ 사물인터넷(Internet of Things : IoT) : 장치 및 시스템, 사람을 연결하는 기술

-Industry 4.0: IoT는 생산 시스템 간의 연결과 데이터 교환을 통해 제조 생산성 향상

→현대 제조 공정은 사물인터넷을 통해 구성되어 있음

→로봇 및 자율 장비 IoT를 통해 얻은 데이터를 기반으로 최근 몇 년 동안 전 세계적으로 제조 시스템의 효율성이 크게 향상됨



<CJ 냉동식품 컨테이너 냉각기 이상 감지 IoT>



<테슬라 부품 이송 로봇 이상 감지 IoT>

Purpose of Paper

- 머신러닝을 사용하여 제조 생산 라인의 잠재적 고장이 발생하기 전에 현실적으로 예측하는 예지 보전 시스템을 개발
- IoT 센서로 부터 얻은 데이터를 이용하여 시스템의 효과와 예측력을 테스트함
 - ➔ 알고리즘들을 비교하면서 예측력이 좋은 머신러닝 알고리즘 제안
 - ✓ 사용된 알고리즘 : 4가지 앙상블 알고리즘-Random Forest, Boosting(XGBoost, Gradient Boosting, AdaBoost)
2가지 알고리즘-Multi Layer Perceptron, Support Vector
 - ✓ 앙상블 알고리즘과 별개의 알고리즘을 비교하였을 때, 앙상블 알고리즘이 전반적으로 우수한 성능

Why Do They Use ML Instead of DL

➤ 과거 많은 연구들이 딥러닝 기법을 사용한 예지보전 시스템 구축 시도

✓ RNN(Recurrent Neural Network) : 순환신경망

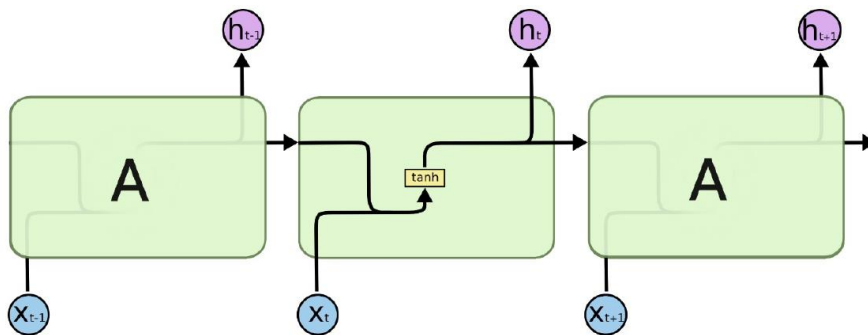
-신경망 모듈이 반복되는 형태

→출력 벡터가 다시 입력되어 어떠한 sequential 데이터도 처리할 수 있음

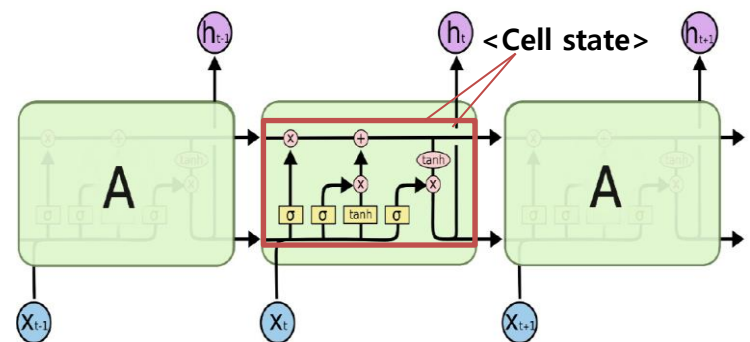
-정보가 많아지면, 뒤쪽 정보에 의존하게 되는 단점을 가지고 있음(Long Term Dependency)

✓ LSTM(Long Short Term Memory)

-RNN의 단점인 Long Term Dependency를 개선하기 위해 cell state를 추가하여 불필요한 정보는 삭제하고, 필요한 정보는 업데이트하면서 long term dependency를 해결함



<RNN>



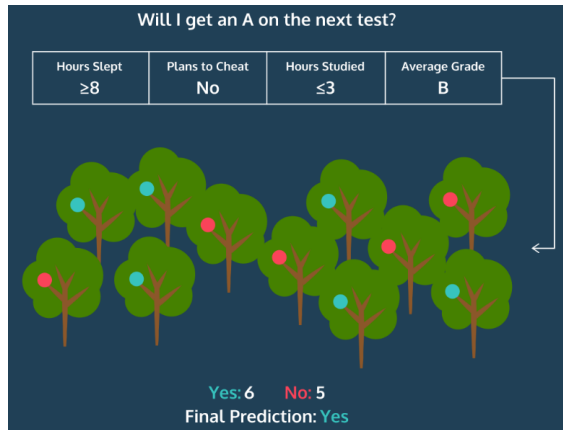
<LSTM>

Why Do They Use ML Instead of DL

- 과거 많은 연구들이 딥러닝 기법을 사용한 예지보전 시스템 구축 시도
 - ✓ NASA의 항공기 엔진 성능 저하 모니터링 및 엔진 유효 수명 예측(Zhang et al., 2018)
 - 6개의 항공기 엔진 센서를 통해 데이터 획득
 - LSTM을 사용하여 항공기 엔진 유효 수명 예측
 - ✓ IoT 데이터 획득을 통한 예지보전(Song et al., 2016, Xie et al., 2017)
 - IoT 센서에서 획득한 데이터의 종류 -시계열 데이터
 - RNN 보다 LSTM이 보다 나은 예측 결과
 - ✓ IoT 디바이스의 예지보전에 사용되는 딥러닝 방법 검토 및 분류(Rieger et al., 2019)
 - 시계열 데이터 유형으로 인해 처음에 RNN 제시
 - 실시간으로 많은 데이터가 쌓이기 때문에 Long Term Dependency 발생 파악
→단독 RNN보다 RNN과 LSTM을 조합 알고리즘이 더 나은 결과
 - 딥러닝 기반 모델의 문제점 : 1. 긴 처리 시간
 - 2. 과도한 전력 소비
 - 3. 메모리 성능에 미치는 영향
- ➔실제 산업에 적합하지 않음

Why Do They Use ML Instead of DL

- **Ensemble Methods** : 여러 개의 분류기를 통해 정확한 예측에 도움을 주는 방법
 - ✓ **Random Forest** : 훈련을 통해 구성해 놓은 다수의 의사 결정 나무로부터 분류 결과를 취합해 예측하는 방법
 - ➔ 많은 요인(Feature)을 고려하여, 여러가지의 의사결정 나무를 통해 의사결정하는 알고리즘
 - 장점 1. 과적합 방지 : 모든 요인을 고려하였기 때문에 과적합 예방
 - 2. 정확도 높음: 과적합 방지를 통해 정확도 향상
 - 3. 결측에 대해 강건함
 - ✓ **Boosting**: 이전 모델의 좋지 않은 결과를 가중치를 통해 성능을 높이는 방식
 - ➔ 오답에 집중할수 있지만, 이상치(Outlier)에 취약함
 - AdaBoost: 분류하기 어려운 모델에 가중치를 부여함으로써 예측하는 알고리즘
 - ➔ 기본 부스팅은 단계마다 같은 가중치를 주었지만, AdaBoost는 가중치를 다르게 줌
 - Gradient Boosting(GB): AdaBoost와 같이 단계마다 다른 가중치를 주지만, 경사하강법을 사용하여 오차 최소화
 - XGBoost: GB와 같이 경사하강법을 사용하여 가중치를 업데이트하면 과적합 및 느린 속도의 문제가 있음
 - ➔ 과적합 방지를 위해 GB에 파라미터(λ, γ)를 추가한 알고리즘



<Random Forest>



<Boosting>

Why Do They Use ML Instead of DL

➤ Multilayer Perceptron Regression

- ✓ 여러 개의 입력들이 하나의 은닉층을 거쳐 결과를 출력하는 알고리즘
 - 은닉층(Hidden Layer)-활성화 함수(Sigmoid 함수)를 통해 비선형성으로 분포하는 데이터들을 학습 가능

➔은닉층을 통해 비선형 데이터를 학습하고 XOR 분류를 할 수 있게 함

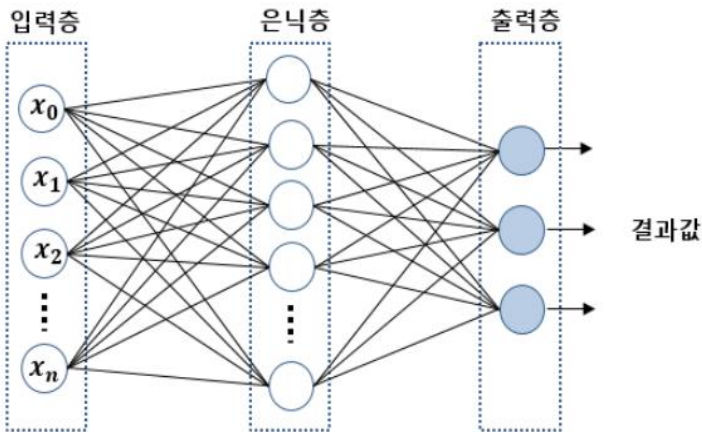
➤ Support Vector Machine

- ✓ 결정 경계 -분류를 위한 기준 선 :데이터 속성 2개 분류→ 선, 3개 분류→ 초평면
- ✓ 마진(Margin) : 결정 경계와 서포트 벡터 사이의 거리

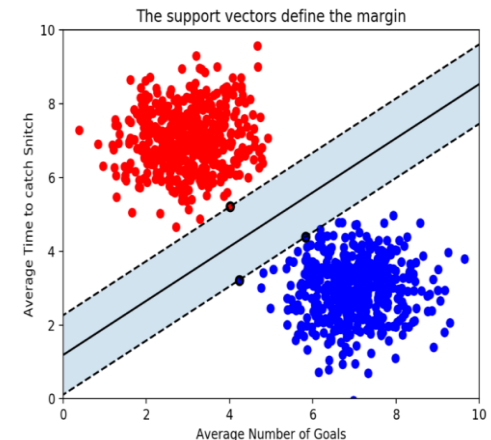
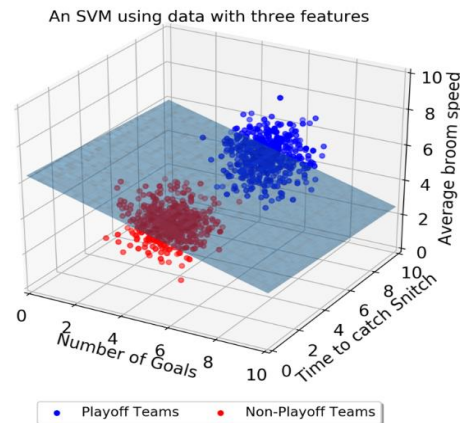
➔마진 최대화를 통해 분류, 예측 성능 향상

장점 : 마진 경계에 있는 서포트 벡터 추출→나머지 데이터 무시가능

➔속도가 매우 빠름



<Multilayer Perceptron Regression>



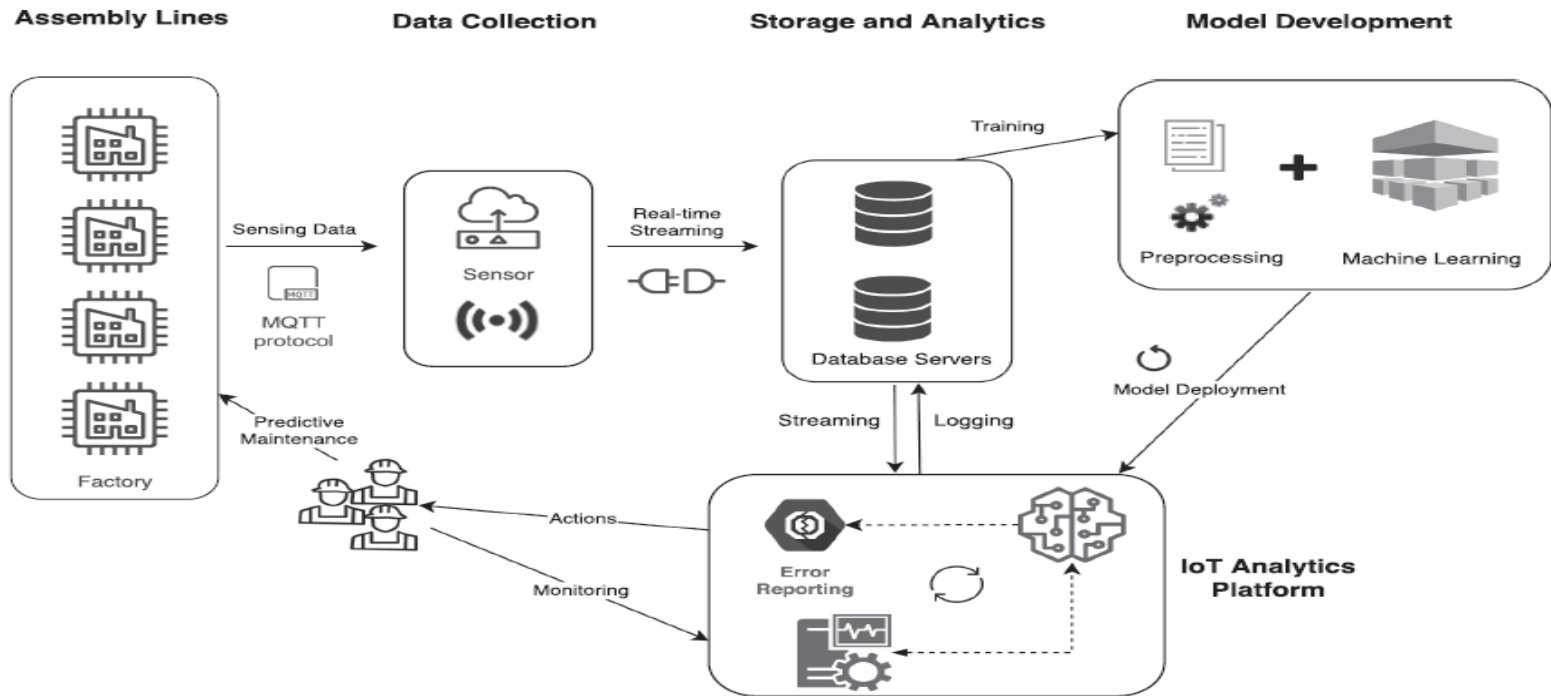
<SVM>

Architectural Overview

➤ 제안된 6개의 알고리즘을 평가하기위해 실제 생산라인(개인 위생용품)에 적용

✓ 제안된 알고리즘 측정 방법

1. IoT에서 데이터 수집
2. 데이터 처리 -불균형 데이터 처리, 상관관계(요인축소), 요인선택
3. 제안된 알고리즘 모델의 적용 및 평가



<예지보전 시스템의 전체 구조>

Data Collection & Data Processing

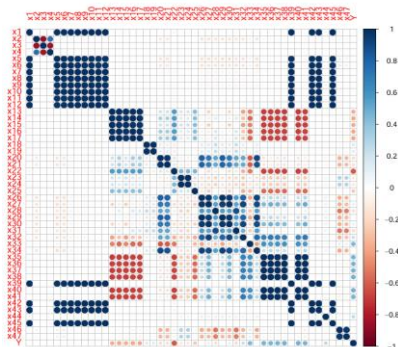
➤ 데이터 수집

✓ 각 장치마다 부착된 IoT를 통해 데이터를 수집

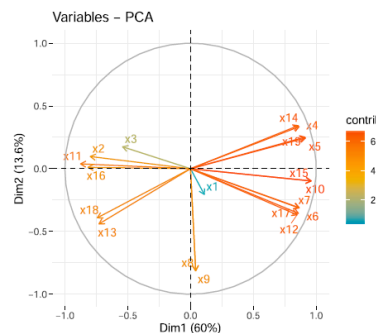
- MQTT(Message Queuing Telemetry Transport): 데이터를 단일 유형으로 변환시키는 장치
 - ➔ 가벼움(최소한의 리소스), 구현하기 쉬움, 보안이 좋음, 지원가능성 높음
 - ➔ IoT 데이터를 효율적으로 전달가능하기 때문에 사용됨

➤ 데이터 처리

1. 불균형 데이터 처리 : 생산라인 데이터는 오류에 대한 정보가 부족하기 때문에 불균형
→ 불균형으로 인한 언더 샘플링, 오버 샘플링 발생 방지 위해 배깅과 부스팅을 통해 균형 맞춤
2. 상관관계: 요인들간의 연관성을 보여줌으로써 상관성이 높은 요인들을 묶어 축소
→ 주성분 분석(Principal Component Analysis : PCA)를 통해 차원 축소
3. 요인선택: 고차원 데이터는 과적합, 계산 복잡성, 다중공선성, 해석 가능성 저하 야기
→ **주성분분석의 기여율**과 도메인 전문 지식을 고려하여 101개의 요인에서 18개의 독립 변수와 1개의 종속변수(부품의 남은 유효 수명: RUL)를 선택함



<PCA>



<PCA Variable Variance Contributions>

$$\text{※주성분분석의 기여율} = \frac{\text{주성분의 분산값}}{\text{데이터 전체의 분산값}}$$

→ 주성분의 분산이 데이터 전체의 분산과 비슷하다면 좋은 요인

➔ 1에 가까울수록 좋음, 0에 가까울수록 설명불가

Algorithm Evaluation

➤ 알고리즘 평가 지표

- ✓ R^2 (설명력/결정계수) : 변수의 총 변동성에서 모델이 줄일 수 있었던 변동성의 비율
➔ 독립변수가 종속변수에 대해 얼마만큼 설명력을 가지는가(클수록 좋음)

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} = 1 - \frac{SSR = \text{설명안된 변동}}{SST = \text{총변동성}}$$

- ✓ 평균절대오차(Mean Absolute Error : MAE): 모든 절대 오차(Error)의 평균

-오차 : 예측한 값과 실제 정답간의 차이 $\rightarrow |y_i - \hat{y}_i|$

➔ 실제 제품 수명과 모델이 예측한 제품 수명의 차이 간의 평균을 절대값으로 측정(작을수록 좋음)

$$MAE = \frac{1}{n} \sum_{i=1}^n \frac{|y_i - \hat{y}_i|}{y_i}$$

- ✓ 평균절대백분율오차(Mean Absolute Percentage Error : MAPE)

-MAE는 Outlier에 취약함 \rightarrow 백분율로 변동을 측정

-MAPE는 0값에 취약함 \rightarrow MAE와 같이 비교하여 사용

$$MAPE = \frac{100}{n} \sum_{i=1}^n \frac{|y_i - \hat{y}_i|}{y_i}$$

- ✓ 평균 제곱근 편차(Root Mean Square Deviation: RMSE)

-평균제곱오차 \rightarrow 실제값과 예측값의 오차제곱의 평균을 통해 측정 \rightarrow 값이 커지면 연산속도가 느려짐

➔ 루트를 통해 연산속도를 향상(작을수록 좋음)

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

y_i = 테스트 데이터의 실제 값

\hat{y}_i = 알고리즘에 의해 예측된 남은 제품 수명

Experimental Results

➤ 실험 결과

➤ 앙상블 알고리즘이 MLP, SVR 알고리즘보다 더 성능이 좋게 나옴(AdaBoost 제외)

✓ 앙상블 알고리즘이 결과가 더 좋은 이유

1.과적합 방지 : Random Forest와 XGBoost는 과적합 방지에 장점이 있음

→MLP나 SVR은 비교적 과적합에 취약함

2.노이즈 데이터에 대한 견고성 :앙상블 알고리즘은 여러 모델의 예측을 집계하기
때문에 노이즈 데이터에 대해 비교적 견고함

3.Hyper-parameter의 설정: MLP나 SVR은 성능 최적화하기 위해 hyper-parameter의
많은 조정이 필요함

→향상된 모델을 위해 Hyper-parameter의 설정 필요

Evaluation results for accuracy of ML algorithms on test dataset.

Algorithm	R^2	MAE	MAPE	RSME
Random forest	0.982	51.97	3.27	147.19
XGBoost	0.979	82.09	5.16	157.28
Gradient boosting	0.776	394.52	24.79	523.91
MLP regressor	0.675	466.32	29.30	632.69
SVR	0.347	682.43	42.88	896.07
AdaBoost	0.338	752.95	47.32	902.85

<Experimental Results>

Conclusion

- 전통적인 유지보수 : 고장 발생 후 또는 제품 교체 주기 이후 교체
→다운타임 발생으로 인한 시간 및 비용 발생
➔머신러닝 기반 예지보전 시스템 개발을 통해 시간 및 비용 절약
- 과거 예지보전 시스템: 전체적인 시스템 대신 부분적 시스템만 적용
→전체 시스템을 고려하지 않은 유지보수로 재고장 발생
➔모든 구성 요소를 통합한 End to End 방식으로 재고장 예방
- 실제 생산라인 제조 장비의 IoT 센서로 부터 수집된 데이터 사용
➔생산 과정을 디지털화하고 전체 라인의 성능 개선을 위한 데이터 분석을 활용할 수 있게 함
- 6개의 알고리즘 알고리즘을 비교분석 한 결과 Random Forest와 앙상블 기법이 좋은 성능을 나타냄

Further Study

- 터키에 있는 하나의 개인 위생용품 공장에서 데이터를 수집하였기 때문에 수집된 공장에서만 유효한 성능
 - 일반화 불가 – 같은 IoT 장치를 사용하는 공장에서는 사용할 수 있지만 다른 IoT 장치를 사용한 고장에서 사용하기 위해서는 여러 IoT 장치를 통해 얻은 데이터로 실험이 필요함
- Hyper-parameter의 조정
 - MLP과 SVR의 낮은 성능의 원인은 hyper-parameter의 조정 실패
 - 발전된 최신 연구를 통해 hyper-parameter 조정을 한다면, random forest/앙상블 기법보다 더 좋은 성능을 낼 수도 있음
- 빈도가 적은 오류 제외
 - 공장에서 발생하는 오류들 중에도 발생빈도 차이가 존재함
 - 본 연구에는 빈번한 장애 데이터만 포함
 - 빈도가 적은 오류도 고려하여 후속 연구 수행 필요