

German Credit Risk 데이터를 사용한 머신러닝 모델의 단계별 요소들이 AI 모델의 성능에 미치는 영향 분석

박 필 원*

Analysis of the Effect of Step-by-step Elements of Machine Learning Model on the Performance of AI Model using German Credit Risk Data

Pill-Won Park*

요 약

머신러닝은 다양한 분야에서 사용되고 있으며, 분석하고자 하는 데이터의 형태와 목적에 따라 다양한 알고리즘이 개발되었다. 머신러닝 모델의 성능은 동일한 알고리즘을 사용한다고 하더라도 단계별 설정에 따라 영향을 받으며 이에 관한 연구가 필요하다. 특정 절차 또는 특정 파라미터가 모델에 끼치는 영향들에 관한 연구는 진행되었으나, 이들을 종합적으로 분석한 연구는 미비하였다. 본 논문에서는 머신러닝 모델 개발에 필요한 처리 단계를 정리한 후, 각 단계들이 머신러닝 모델의 성능에 미치는 영향을 분석하였다. 처리 단계는 데이터 정제, 알고리즘 선정, 하이퍼 파라미터 조정, 검증 비율 조정의 단계로 나누었으며, 이를 kaggle의 German credit risk 데이터와 머신러닝 자동화 도구를 사용하여 실측하였다.

Abstract

Machine learning is used in various fields, and various algorithms have been developed according to the type and purpose of data. The performance of the machine learning model is affected by the step-by-step setting even if the same algorithm is used, and research on this is needed. However, studies on the effects of specific procedures or specific parameters on the model have been conducted, but studies that comprehensively analyze them have been insufficient. In this paper, after summarizing the processing steps required to develop the machine learning model, the effect of each step on the performance of the machine learning model was analyzed. Processing steps were divided into steps of data purification, algorithm selection, hyper-parameter adjustment, and verification ratio adjustment, which were measured using Kaggle's German credit risk data and machine learning automation tools.

Keywords

machine learning, performance of AI model, step of machine learning, performance analysis, auto ML

* ㈜토탈시스 연구소
- ORCID: <https://orcid.org/0000-0003-4693-6042>

· Received: Aug. 22, 2022, Revised: Nov. 13, 2022, Accepted: Nov. 16, 2020
· Corresponding Author: Pill-Won Park
Dept. of 9F 15, Sindorim-ro 17-gil, Guro-gu, Seoul, Republic of Korea
Tel.: +82-02-3667-2089, Email: pillwon79@gmail.com

1. 서론

최근 몇 년간 인공지능과 머신러닝이 사용되는 분야가 급증하고 있다[1][2]. 사용자가 얻고자 하는 결과물, 제공되는 데이터의 종류 및 활용 목적에 따라 다양한 알고리즘들이 개발 및 사용되고 있다.

머신러닝 모델의 성능은 다양한 방법이 있으나 정밀도와 재현율을 기준으로 평가하는 것이 일반적이다[3][4]. 모델의 성능은 머신러닝을 진행하며 거치게 되는 단계별 작업의 내용에 영향을 받는다. 기존의 머신러닝 모델의 성능 연구들이 특정 하이퍼 파라미터(파라미터) 또는 특정 단계에 관한 연구가 주를 이루어, 전체 단계에 대한 분석은 미비하였다.

본 논문에서는 머신러닝의 단계를 문제 인식, 데이터 수집, 데이터 정제, 알고리즘 선정, 파라미터 조정, 검증 비율 조정, 성능 분석 단계로 가정하고, 각각의 절차에서 진행되는 내용들이 모델의 성능에 영향을 끼치는 내용을 분석하였다.

단계별 요소들이 머신러닝 모델의 성능에 끼치는 영향을 분석하기 위한 도구로서 머신러닝 자동화 도구(Auto ML)를 선택하였다. 머신러닝 자동화 도구들은 기초적인 데이터 정제, 알고리즘의 적용, 파라미터 조정 및 시각화 등을 지원하고 있다[5][6]. 본 논문에서는 머신러닝 자동화 도구 중 하나인 WiseProphet(WP)를 활용하여 머신러닝 모델의 각각의 단계가 최종 결과에 끼치는 영향에 대해 분석을 진행하였다[7].

II. 머신러닝 프로세스의 단계별 성능 연관 요소 분석

2.1 머신러닝 프로세스

머신러닝 프로세스는 그림 1과 같다. 머신러닝 프로세스의 초기 단계는 문제 정의 및 문제 해결을 위한 데이터 수집 단계이지만 본 논문에서 분석하고자 하는 범위를 넘으므로 다루지 않는다.

데이터 정제 단계에서는 수집된 데이터를 정리하는 단계이다. 데이터의 변수를 머신러닝에 사용할지 판단해야 하며, 이상치와 결측치 처리, 스케일 적용 등을

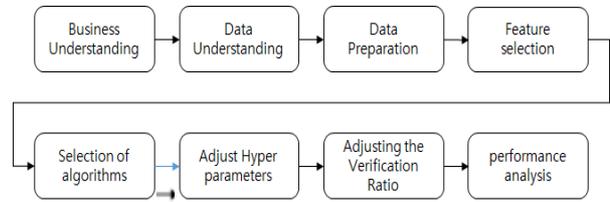


그림 1. 머신러닝 프로세스
Fig. 1. Machine learning process

알고리즘 선정 단계에서는 데이터의 타입과 목표 변수 설정들을 기반으로 사용할 알고리즘을 결정한다. 파라미터 조정 단계에서는 이전 단계에서 설정한 알고리즘에서 사용할 파라미터 및 옵션을 설정한다. 검증 비율 조정 단계에서는 학습에 사용할 데이터와 검증에 사용할 데이터의 비율을 설정한다. 또한 교차 검증을 위한 폴드의 사용 여부를 결정하게 된다. 성능 분석 단계에서는 머신러닝을 통해 나온 결과들을 분석하는 단계로서 해당 모델의 성능을 평가하는 단계이다.

2.1.1 데이터 정제 단계

데이터 정제 단계에서는 목표변수의 설정, 변수의 타입 설정, 데이터 분포 스케일 처리, 결측치 및 이상치 처리 등의 작업을 진행한다.

목표변수는 사용자가 머신러닝 모델 학습을 통해 예측하고자 하는 변수이다. 목표변수를 설정하지 않을 경우 군집 분석을 사용하는 경우가 많다. 변수 타입의 경우 분석에 사용하는 데이터의 유형을 의미하며, 범주형 데이터인 경우 분류모델을, 수치형인 경우 회귀모델을 사용하는 것이 일반적이다. 스케일 처리의 경우 동일한 중요도의 변수임에도 불구하고, 변수가 가지는 값의 범위가 매우 다르면 학습 결과에 악영향을 끼칠 수 있다. 이를 방지하기 위해 스케일 조정을 진행한다. 결측치 및 이상치의 경우 데이터의 결손이나, 비정상적 수치의 처리 여부이다. 결측치나 이상치 데이터를 삭제하거나 특정 값으로 설정한다.

특징 선택의 경우 목표변수와 다른 변수들 사이의 상관관계 분석을 통한 중요도를 계산하여, 몇 개의 변수를 머신러닝 모델에 포함 시킬지 여부를 결정한다.

2.1.2 알고리즘 선정 단계

머신러닝에 사용할 알고리즘을 선정하는 단계이다. 알고리즘 선택, 앙상블 적용 여부 등을 수행하게 된다.

알고리즘 선정 단계는 이전 단계인 데이터 정제 단계의 영향을 크게 받는다. 목표변수의 존재 여부, 데이터 타입 등으로 인해 사용할 수 있는 알고리즘이 한정되기 때문이다. 분류 알고리즘을 사용할 경우 Decision Tree, Random Forest, Gradient Boosting, XGBoost, Logistic Regression, CNN 기법을 사용할 수 있다. 군집 알고리즘을 사용할 경우 K-means, DBSCAN, Gaussian Mixture 등을 사용할 수 있다. 회귀 알고리즘을 사용할 경우 Decision Tree, Random Forest, Elastic Net, Gradient Boosting, XGBoost, RNN, CNN, LSTM 등을 사용할 수 있다.

앙상블은 다종의 모델들을 하나의 문제 해결을 위해 사용하는 것이다. 각각의 머신러닝 알고리즘은 독립적으로 활용이 가능한 알고리즘들이지만 초기 가중치 및 학습과정에서 편향된 학습을 진행하게 될 가능성이 있다. 이를 막기 위해 다양한 알고리즘을 통해 결론을 도출하고 이를 조합하는 것이 앙상블 기법이며, 해당 기법의 사용 여부를 이 단계에서 결정하게 된다.

2.1.3 하이퍼 파라미터 조정 단계

파라미터 조정 단계에서는 알고리즘의 파라미터를 조정하는 단계이다. 머신러닝 알고리즘은 다양한 파라미터를 설정해야 하며, 이 절차에서 알고리즘 성능 향상을 위해 반복작업이 필요할 수 있다.

CNN의 경우 filters, kernel_size, pool_size, optimizer, batch_size, epochs 등의 파라미터를 설정할 수 있다. DBSCAN의 경우 algorithm, eps, min_samples 등의 파라미터를 설정할 수 있다. Decision Tree의 경우, Criterion, Max_depth, min_samples_leaf 등의 파라미터를 설정할 수 있다. Elastic Net의 경우 alpha, L1_ratio, random_state 등의 파라미터를 설정할 수 있다. Gaussian Mixture의 경우 covariance_type, max_iter, n_components 등의

파라미터를 설정할 수 있다. Gradient Boosting의 경우 learning rate, max_depth, subsample 등의 파라미터를 설정할 수 있다. K-means의 경우 init(=initial), n_clusters, n_init 등의 파라미터를 설정할 수 있다. Logistic Regression의 경우 C(cost function), random_state 등의 파라미터를 설정할 수 있다. RNN과 LSTM의 경우 batch_size, epochs, loss, metrics, optimizer, window_size 등의 파라미터를 설정할 수 있다. Random Forest의 경우 Criterion, Max_depth, min_samples_leaf, n_estimators, random_state 등의 파라미터를 설정할 수 있다. XGBoost의 경우 eta, gamma, max_depth 등의 파라미터를 설정할 수 있다.

2.1.4 검증 비율 조정 단계

검증 비율 조정 단계에서는 학습데이터와 검증데이터의 비율을 설정한다. 또한 교차 검증을 위한 폴드 설정도 이 단계에서 진행한다.

학습 및 검증데이터 비율은 일반적으로 80(학습데이터) : 20(검증데이터)에서 70:30 로 나누어 평가한다. 이때 학습데이터는 검증에 전혀 사용되지 못한다. 즉 검증데이터 비율만큼의 데이터가 학습에 사용되지 못하는 상태가 된다.

폴드는 이러한 단점을 방지하기 위하여 전체 데이터를 n 개의 폴드로 나누어 하나의 폴드를 검증데이터로 사용하고 n-1개의 폴드를 학습데이터로 사용하는 모델이다. 이후 검증데이터로 사용하는 폴드를 바꾸어가며 총 n 개의 모델 테스트를 진행하여 평균값을 구하게 된다.

그림 2 는 전체 데이터를 8:2로 학습/검증데이터를 분할한 경우에 폴드 5 인 경우 전체를 5 개의 그룹으로 분할한 후 5 개의 그룹(폴드)로 전체 데이터를 나눈 후 학습 및 검증데이터로 사용하는 경우를 묘사한 그림이다. 폴드가 1 인 경우 모델이 1 번 동작 후 그 결과를 사용한다면, 폴드가 5인 경우 각각의 경우에 대하여 총 5 번 동작하게 되고 그 5 개의 결과물을 평균하여 최종 결과를 도출한다.

4 German credit risk 데이터를 사용한 머신러닝 모델의 단계별 요소들이 AI 모델의 성능에 미치는 영향 분석

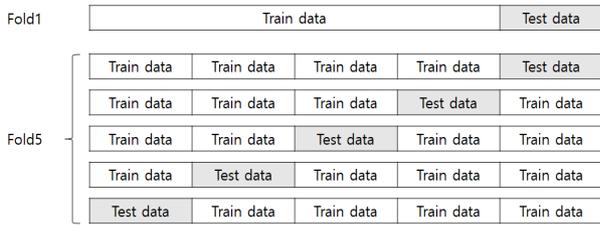


그림 2. 폴드 1과 폴드 5 인 경우의 학습/검증데이터 분할 샘플

Fig. 2. Sample train/test data segmentation for Fold 1 and Fold 5

III. 실험 및 고찰

3 장에서는 2 장에서 정의한 단계별 요소들을 조정하며 전체 모델의 성능에 미치는 영향을 분석한다.

3.1 실험 구성 및 방법

분석에 사용한 데이터는 kaggle에서 제공한 German credit risk 데이터이며[8], 독일인의 신용평가에 대한 데이터로서 금융 데이터를 기반으로 개인의 신용위험도(risk)를 good 과 bad 값으로 분석하기 위한 데이터이다. 1000 개의 데이터로 총 28 개의 변수로 이루어져 있고, Savingaccounts에 183개의 결측치를, Checkingaccount에 394개의 결측치를 가지고 있다. 비교 기준은 WP에서 제공하는 기본 알고리즘 중 분류 알고리즘인 Decision Tree를 기본 파라미터를 사용하였을 때로 잡았다. 각각의 단계마다 적용 가능한 옵션들을 적용하여 그 변화를 정리하였으며, 모델의 우수성은 정밀도와 재현율을 기반으로 판단하였다.

비교를 위해 설정한 기준값은 다음과 같다. 데이터 탐색 단계에서 결측치를 모두 0으로 설정하였고, 특징 선택 단계에서는 28개의 변수를 모두 머신러닝 모델에 포함하였다. risk를 목표변수로 설정하여 분류 알고리즘을 적용하였으며, 알고리즘 선택 단계에서는 Decision Tree를 선정하였고, 파라미터 설정 단계에서 criterion은 gini, max_depth는 5, min_samples_leaf는 1로 설정하였다. 검증데이터 비율 설정 단계에서는 검증데이터 분할 비율은 20%, 교차 검증을 위한 폴드 수는 5 로 설정하였다.

실험 시스템으로 사용한 WP의 프로세스는 그림 3과 같다.

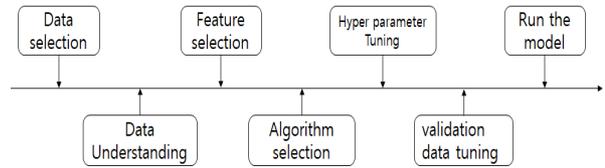


그림 3. WP 머신러닝 프로세스
Fig. 3. Machine learning process of WP

3.2 실험 결과 및 고찰

3.2.1 데이터 정제 단계

데이터 정제 단계에서의 성능에 영향을 주는 요소로는 결측치, 이상치, 변수 사용 여부가 있으며 그 결과는 표 1과 같다.

결측치를 0으로 대체 한 경우를 기준으로 잡았다. 일부 결측치 데이터만 삭제한 경우는 Saving accounts에 존재하던 결측치 183개를 삭제하고, Checking account에 있던 394개의 결측치는 0 으로 설정하였다. 1000개의 데이터 중 183개를 삭제한 817개의 데이터를 기반으로 머신러닝을 진행하였다. 모든 결측치 데이터를 삭제한 경우는 German credit risk에 존재하는 모든 결측치가 있는 데이터를 삭제한 경우이다.

표 1. 결측치 설정에 따른 모델 성능 변화

Table 1. Model performance change according to missing value setting

To set missing values	Target value (risk)	Performance	
		Precision	Recall
Replace missing values with 0 (default)	good	0.72	0.89
	bad	0.53	0.27
Delete some missing data	good	0.64	0.76
	bad	0.50	0.36
Delete all missing data	good	0.59	0.75
	bad	0.48	0.31
Do not use variables with missing values	good	0.60	0.82
	bad	0.52	0.27

Saving accounts와 Checking account의 결측치를 모두 삭제하여 606개의 데이터를 기반으로 머신러닝을 진행하였다. Saving accounts와 Checking account에 중복으로 결측 데이터가 포함된 경우가 있다. 결측치가 있는 변수 사용하지 않은 경우는 Saving accounts와 Checking account 변수를 머신러닝에 사용하지 않은 경우이다.

결측치가 있다고 하더라도 모델링에 사용한 데이터의 개수를 최대한 확보한 모델의 성능이 높게 측정되었으며, bad 값에 대한 모델의 정밀도와 재현율이 일관되게 낮게 나온 것으로 보아 데이터 부족 등의 외부 요인이 있던 것으로 판단된다.

추가적으로, 학습에 사용할 변수의 선정 역시 머신러닝 모델의 성능에 영향을 주며, 그 결과는 표 2와 같다.

모든 변수 사용한 경우를 기준으로 잡았다. 중요도 기준 상위 25개 변수 사용한 경우, 중요도 수치를 기준으로 0.01 이하의 변수를 삭제한 경우, 중요도 수치를 기준으로 0.05 이하의 변수를 삭제한 경우를 비교하였다.

초기값에 비해 너무 많은 변수를 삭제하거나, 모델링에 사용한 두 경우 모두 모델의 성능이 낮았으며, 적정 수준의 변수만을 사용하여 모델을 학습시킨 케이스가 성능이 가장 높게 측정되었다.

표 2. 특징 선택 방법에 따른 모델 성능 변화
Table 2. Model performance changes according to feature selection method

To select features	Target value (risk)	Performance	
		Precision	Recall
Use all variables (default)	good	0.72	0.89
	bad	0.53	0.27
Use the top 25 variables by importance	good	0.72	0.88
	bad	0.51	0.28
Delete variables below 0.01 by importance	good	0.70	0.93
	bad	0.51	0.28
Delete variables below 0.05 by importance	good	0.73	0.90
	bad	0.58	0.30

3.2.2 알고리즘 선정 단계

알고리즘 선정 단계에서의 성능에 영향을 주는 요소로는 알고리즘 과 앙상블 모델의 사용 여부 등이 있으며 그 결과는 표 3과 같다.

표 3에서는 WP에서 제공하는 Decision Tree, Random Forest, Gradient Boosting, XGBoost, Logistic Regression 알고리즘을 비교하였다. 앙상블 모델의 경우 위에서 언급한 5 개의 알고리즘을 동시에 적용하여 그 결과를 비교하였다.

Random Forest가 bad 상태의 정밀도가 급격히 상승하였으나 재현율이 매우 낮았다. 일전에 언급한 것과 같이 bad 데이터에 문제가 있는 것으로 판단된다. good 상태에 대해서는 5 가지 알고리즘의 정밀도는 유사하게 나타났으며, 재현율의 경우 Random Forest가 가장 높았다.

표 3. 알고리즘의 변화에 따른 모델 성능 변화
Table 3. Model performance changes according to algorithm changes

Algorithm	Target value (risk)	Performance	
		Precision	Recall
DECISION TREE (default)	good	0.72	0.89
	bad	0.53	0.27
RANDOM FOREST	good	0.71	0.99
	bad	0.80	0.13
GRADIENT BOOSTING	good	0.70	0.90
	bad	0.48	0.19
XGBOOST	good	0.72	0.84
	bad	0.48	0.31
LOGISTIC REGRESSION	good	0.72	0.95
	bad	0.68	0.23
Ensemble	good	0.70	0.96
	bad	0.60	0.14

3.2.3 하이퍼 파라미터 조정 단계

파라미터 조정 단계에서 성능에 영향을 주는 요소로는 파라미터의 값 등이 있으며 그 결과는 표 4와 같다. 표 4에서는 앞서 언급한 5 개 알고리즘의 파라미터들을 WP가 제공하는 최적화 기능을 사용하였을 경우를 비교하였으며, 조정된 파라미터는 다음과 같다.

Decision Tree의 경우 criterion, max_depth, min_samples_leaf. Random Forest의 경우 criterion, max_depth, min_samples_leaf, n_estimators. Gradient Boosting의 경우 learning_rate, max_depth, subsample. XGBoost의 경우 eta, gamma, max_depth. Logistic Regression의 경우, C, random_state 파라미터를 설정하였다. 최적화를 진행한 결과 정밀도는 약간 하락하거나 유지되었고, 재현율이 약간 상승하거나 유지된 것을 볼 수 있다. Random Forest의 경우 bad 정밀도에서 가장 큰 변화를 보였으며, XGBoost의 경우 재현율에서 가장 큰 상승률을 보였다.

표 4. 파라미터 수정에 따른 모델 성능 변화
Table 4. Model performance changes due to parameter modification

Algorithm	Target value (risk)	Performance	
		Precision	Recall
DECISION TREE (default)	good	0.70	0.93
	bad	0.47	0.14
RANDOM FOREST	good	0.69	0.97
	bad	0.56	0.08
GRADIENT BOOSTING	good	0.70	0.92
	bad	0.50	0.17
XGBOOST	good	0.71	0.93
	bad	0.55	0.19
LOGISTIC REGRESSION	good	0.72	0.95
	bad	0.67	0.22

3.2.4 검증 비율 설정 단계

검증 비율 설정 단계에서의 성능에 영향을 주는 요소로서 평가 데이터 분할 비율과 폴드 수 등이 있으며 그 결과는 표 5와 표 6과 같다.

표 5에서는 전체 데이터 중 평가 데이터 비율을 변경하며 그 결과를 측정하였다. 평가 데이터 비율이 증가함에 따라 정밀도가 증가하는 것을 볼 수 있었으며, 재현율은 평가데이터 비율을 40%로 설정한 경우 외에는 감소하는 것을 볼 수 있다.

표 6에서는 교차 검증을 위한 폴드 수를 변경하며 그 결과를 측정하였다. 폴드 수의 변화에 따라 good 의 정밀도와 재현율이 미세하게 증가하였다. bad의 경우 정밀도는 증가하였으나 재현율은 변동이 있었다.

표 5. 평가 데이터 분할 비율에 따른 모델 성능 변화
Table 5. Model performance changes according to evaluation data segmentation ratio

Data segmentation ratio	Target value (risk)	Performance	
		Precision	Recall
20% (default)	good	0.72	0.89
	bad	0.53	0.27
30%	good	0.70	0.84
	bad	0.30	0.08
40%	good	0.73	0.88
	bad	0.47	0.25
50%	good	0.76	0.78
	bad	0.44	0.41

표 6. 폴드 수에 따른 모델 성능 변화
Table 6. Model performance changes with number of Folds

Number of Folds	target variable (risk)	Performance	
		Precision	Recall
2	good	0.72	0.82
	bad	0.36	0.23
3	good	0.73	0.87
	bad	0.44	0.23
5 (default)	good	0.72	0.89
	bad	0.40	0.17
7	good	0.73	0.88
	bad	0.47	0.25
10	good	0.73	0.89
	bad	0.46	0.21

IV. 결론 및 향후 과제

본 논문에서는 머신러닝 모델 개발에 필요한 처리 단계를 정리한 후, 단계별로 진행하며 변경 가능한 변수들이 모델의 성능에 미치는 영향을 분석하였다. kaggle German credit risk 데이터를 기반으로 비교 분석하였으며, 데이터 정제, 알고리즘 선정, 알고리즘 변경, 하이퍼 파라미터 조정, 검증 비율 조정 등의 단계로 나누어서 비교하였다.

초기 세팅과 비교하였을 때 데이터 정제의 경우 -18%~1%, 알고리즘의 변경에 따라 -6%~8%, 파라미터 조정을 통해 -14%~10%, 검증 비율 조정을 통해 -6%~7%의 성능 향상을 확인하였다.

데이터 정제단계의 경우 최대한 자료를 많이 사용하는 경우 머신러닝 모델의 성능이 증가하였음을 알 수 있었으며, 그 외 요소들의 경우 알고리즘 선택 정보보다는 파라미터 선정에 따른 성능 변화가 큰 것을 알 수 있었다.

본 연구에 사용한 kaggle의 데이터를 보강하여 good과 bad의 비율을 조정할 경우 더 나은 머신러닝 성과를 보여 줄 수 있을 것으로 추측되며 향후 연구가 필요하다.

References

- [1] Databricks, <https://www.databricks.com/glossary/mlops>
- [2] Shin Won-Sub and Shin Dong-Hoon, "A Study on the Application of Artificial Intelligence in Elementary Science Education", Journal of Korean Elementary Science Education, Vol. 39, No. 1, pp. 117-132, Fed. 2020. <https://doi.org/10.15267/keses.2020.39.1.117>.
- [3] Stephen V. Stehman, "Selecting and interpreting measures of thematic classification accuracy", Remote Sensing of Environment, Vol. 62, No. 1, pp. 77-89, Oct. 1997. [https://doi.org/10.1016/S0034-4257\(97\)00083-7](https://doi.org/10.1016/S0034-4257(97)00083-7).
- [4] Charles E. Metz, "Basic principles of ROC analysis", Seminars in Nuclear Medicine, Vol. 8, No. 4, pp. 283-298, Oct. 1978. [https://doi.org/10.1016/S0001-2998\(78\)80014-2](https://doi.org/10.1016/S0001-2998(78)80014-2).
- [5] Chris Thornton, Frank Hutter, Holger H. Hoos, and Kevin Leyton-Brown, "Auto-WEKA: combined selection and hyperparameter optimization of classification algorithms", Proc. of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 847-855, Aug. 2013. <https://doi.org/10.1145/2487575.2487629>.
- [6] Gokhan Simsek(2019), Introduction to Automated Machine Learning(AutoML), SOFTWARE ENGINEERING DAILY(online web-page) [accessed: Dec. 20, 2022]
- [7] WiseProphet, <http://prophet.wise.co.kr> [accessed: Dec. 20, 2022]
- [8] Kaggle German Credit Risk Prediction, <https://www.kaggle.com/kabure/german-credit-data-with-risk> [accessed: Dec. 20, 2022]

저자소개

박 필 원 (Pill-Won Park)



2008년 2월 : 충남대학교
전기정보통신공학부(공학사)
2010년 2월 : 고려대학교
컴퓨터전파통신공학과(공학석사)
2017년 2월 : 고려대학교
컴퓨터전파통신공학과(공학박사)
2022년 10월 ~ 현재 : ㈜토탈시스

연구소장

관심분야 : 이동통신, AI, 머신러닝