

플라스틱 원료 혼합기의 자동 보정 소프트웨어를 위한 머신러닝 예측 모델

황 희 정*

Machine Learning Predictive Model for Automatic Calibration Software in Plastic Raw Material Mixer

Heejoung Hwang*

요 약

플라스틱 사출 성형 공정에 있어 원료 혼합은 완제품의 품질을 좌우하고 원가절감에 영향을 미치는 중요한 과정이다. 특히 원료 크기 및 형태, 무게 등 다양한 조건에 따라 정밀하게 설정된 원료들을 투입하기 위해서는 여러 변수들과 혼합기내 잔류량, 초과 투입량, 파워, 모터 등의 동작 상태에 따라 자동으로 보정하는 기술이 필요하다. 본 연구에서는 기존과 같이 정해진 조건에 따른 프로그램 구현이 아닌 머신러닝을 이용해 혼합기 동작에 영향을 미치는 다양한 변수 값들을 분석해 자동으로 보정이 필요한지 판단하는 인공지능 예측 모델을 분석하고 설계했다. 실제 공장 환경에서 실험을 통해 평균 98.3%의 정확도를 확인할 수 있었으며 처리 속도에서도 데이터 1건당 평균 0.093초의 처리 속도를 달성해 실제 공정 운영에 충분한 성능을 보여줄 수 있었다.

Abstract

In the plastic injection molding process, raw material mixing is an important process that determines the quality of the finished product and affects cost reduction. In particular, in order to precisely inject raw materials set according to various conditions such as the size, shape, and weight of the raw materials, a technology that automatically compensates for various variables and operating conditions such as residual amount, overfeeding, power, and motor in the mixer is required. In this study, rather than implementing a program according to set conditions as before, we used machine learning to analyze and design an artificial intelligence prediction model that analyzes the values of various variables that affect the operation of the mixer to determine whether automatic correction is necessary. Through experiments in a real factory environment, we were able to confirm an average accuracy of 98.3%, and in terms of processing speed, we achieved an average processing speed of 0.093 seconds per data, showing sufficient performance for actual process operation.

Keywords

smart factory, automatic calibration, plastic mixer, prediction model, artificial intelligence

* 가천대학교 IT융합대학 컴퓨터공학부 교수
- ORCID: <https://orcid.org/0000-0001-5462-0983>

· Received: Jun. 28 2023, Revised: Jul. 11, 2023, Accepted: Jul. 14, 2023
· Corresponding Author: Heejoung Hwang
Dept. of Computer Engineering, Gachon Univ. Korea
Tel.: +82-31-750-4758, Email: hwanghj@gachon.ac.kr

I. 서 론

플라스틱 사출 성형에 있어 원료 혼합은 완제품의 품질을 좌우하고 원가절감에 영향을 미치는 중요한 공정이다. 스마트 팩토리는 대규모 생산 시설뿐만 아니라 중소 규모의 생산환경에도 도입이 절실하나 현실적으로 초기 도입 비용이나 유지에 따른 비용 및 기술전문가 확보가 쉽지 않은 중소기업의 경우 도입에 많은 어려움이 있다[1][2]. 특히 소프트웨어 인력이 절대적으로 부족해 다양한 생산 설비를 탄력적으로 운영해야 하는 중소기업의 경우 개별 환경에 대한 소프트웨어 개발은 불가한 상황이다.

일반적인 플라스틱 사출성형 공정은 원료투입 -> 원료 혼합 -> 원료 제습/건조 -> 사출/성형의 단계로 진행되며 각 공정은 개별 기기들을 파이프라인으로 연결한 형태가 된다. 특히 원료 혼합 공정은 제품의 품질을 향상 시키고 원재료 절감과도 밀접한 영향이 있기 때문에 가장 중요한 공정이라고도 할 수 있다. 원료 혼합은 작업 단위(배치)에 따라 2~6개의 호퍼별로 원료 투입량을 각각 설정하는 방식으로 예를 들어 1번 호퍼에 10kg 투입을 설정 하면 개폐기 조작을 통해 로드셀 계량기로 투입하고 측정값에 기반해 개폐기를 조정하는 과정으로 진행된다. 이때, 플라스틱 원료(Pallet)의 크기 및 형태, 단위 무게가 다양하고 개폐기 역시 게이트형과 스크류형 등으로 다양해 단순히 투입량을 로드셀 계량기에서 측정하는 것만으로 정확한 원료 투입은 어렵다. 문제 해결을 위해서는 생산과정에서 발생하는 원료공급 데이터를 기반으로 생산이 진행되는 과정에서 투입 설정과 실제 투입량의 오차범위를 줄이기 위한 자동화된 기술이 필요하나 일부 외산 기기에서는 고정된 프로그램 방식으로 모듈이 제공되거나 기기 모델이 달라지는 경우 프로그램 수정에 따른 추가 비용들이 발생하기 때문에 중소기업 혹은 다양한 모델을 운용하는 경우 도입하기 어렵다는 문제가 있다[3].

본 논문에서는 이러한 문제점을 해결하고자

첫째, 다양한 변수 및 환경 요인에 따라 인공지능 기술을 통해 자동으로 보정하는 방법 제안

둘째, 실제 환경에서 발생한 데이터를 기반으로 여러 학습모델을 비교 실험하고 정확도 98% 이상

의 최적 학습모델을 도출하고자 한다.

연구 결과를 활용하면 다양한 원료, 혼합기 구성에 따라 매번 프로그램을 개발하지 않고 기기 운영을 통해 자동으로 보정할 수 있는 플라스틱 원료 혼합기 개발이 가능해질 것으로 기대 한다.

본 논문의 구성은 다음과 같다. 2장은 관련연구로 스마트 팩토리의 필요성과 중요성에 대하여 논하고 학습 모델들에 대한 최근 연구 동향을 살펴본다. 3장에서는 본 논문에서 제안하는 투입량 자동 보정을 위한 인공지능 학습 모델에 대해 자세하게 기술한다. 4장에서는 실제 공정에서 수집된 데이터에 기반 한 실험 환경을 구현하고 다섯 가지 학습 모델을 적용해 성능을 비교 평가 하고 이를 기반으로 실시한 성능 평가를 진행한다. 마지막으로 5장에서는 결론에 대해 기술하고 본 연구가 갖는 한계점 및 향후 연구에 대하여 기술하고자 한다.

II. 관련 연구

2.1 스마트 팩토리

스마트 팩토리는 기존의 제조 공장을 IT기술과 융합하여 고객 맞춤형 제품을 최소 비용과 시간으로 생산하는 혁신적인 공장 시스템을 말한다. 이는 제조 분야에서 경기불황과 고용 문제를 극복하고 생산성과 효율성을 극대화하기 위한 방법으로 주목 받고 있다. 스마트 팩토리는 산업 기기와 생산 전 과정이 네트워크로 연결되며 고객 요구에 유연하게 대응할 수 있는 체계를 가진다. 스마트 팩토리의 핵심은 생산 공정, 인터페이스, 운영의 최적화 기술, 다품종 복합생산, 조달 및 물류혁신, 장비와 인간의 협업을 가능하게 하는 것으로 제조 기술과 센서, 클라우드, 빅데이터 등 다양한 ICT 기술과 결합한 형태로 실현되고 있다[1][4].

플라스틱 사출성형 분야에서 스마트 팩토리를 실현하기 위한 노력으로는 공정 자동화, 센서를 활용한 실시간 모니터링, 원격제어 등이 주된 관심사이며 원료 혼합기의 경우 다양한 원료 형태와 기기 동작에 따른 오차율을 지속적으로 보정해야 생산품의 품질을 일정하게 유지할 수 있고 원료 및 원가 절감이 가능하다.

현재 설정에 따라 원료 투입을 정확하게 유지하기 위한 자동 보정 기술은 주로 정해진 값과 조건에 따라 처리하는 프로그램을 탑재해 처리하는 방식이며 고정된 원료와 단일 제품 생산 공정의 경우 효과적으로 운용이 가능하다[5][6]. 그림 1은 정해진 파라미터에 의해 자동보정을 수행하는 일반적인 처리 방식의 처리 구조 이다.

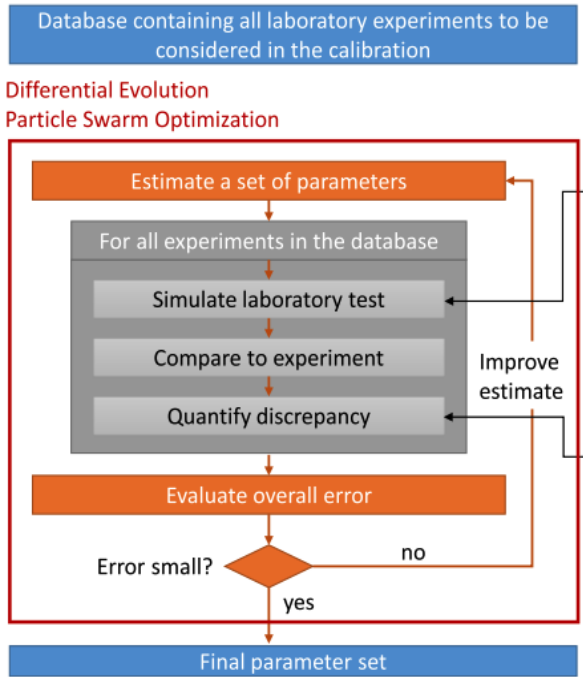


그림 1. 자동 보정 방식 예시
Fig. 1. Example of automatic calibration method

그러나 원료의 형태가 특성이 자주 변경되고 다 품종 소량 생산의 형태에서는 정확도가 떨어지고 새로운 환경을 위해 프로그램을 새로 개발해야 하는 문제가 있어 중소 플라스틱 제조 공장에서 도입하기 어려운 문제가 있다[7][9].

2.2 머신러닝 분류 기술

일반적으로 머신러닝의 세 가지 학습종류는 지도 학습, 비지도 학습, 강화 학습을 들 수 있다. 지도 학습은 레이블 된 데이터 즉, 특정 데이터셋을 특징할 수 있는 결과 값이 과거의 경험이나 누군가에 의해 부여되어 있어야 한다. 지도 학습의 주요 목적은 레이블 된 훈련 데이터에서 모델을 학습해 새로운 미래 데이터에 대한 예측을 만드는 것이라 볼 수 있

다. 정상과 비정상 등을 감지하는 것과 같은 이진 분류와 손글씨 이미지 분석과 같은 다중 분류 등 분류 기술은 가장 손쉽게 접근할 수 있으면서도 적용 분야에 따라 우수한 결과를 얻을 수 있는 기법이기도 하다.

대표적인 분류 기법으로는 로지스틱 회귀, 선형 판별 분석, KNN(K-Nearest Neighbors), 트리, 신경망, SVM(Supported Vector Machine)이 대표적이다[8]. 이 중 KNN은 다른 방법과 달리 데이터 학습에 추가적인 시간이 소요되지 않으면서도 실시간 예측이 가능한 구조로 학습 데이터셋이 자주 변경되거나 저 사양의 컴퓨팅 유닛을 사용해야 하는 환경등에서 유리한 부분이 있다.

III. 원료 혼합기 머신러닝 자동 보정 모델

기존 방식들은 사전에 정의된 알고리즘, 즉 특정 변수 값들이 어떤 범위에 있거나 특정 상황에서 조정이 필요하다는 것을 경험에 의해 프로그램하는 방식으로 변수 값들이 변하는 상황 즉, 혼합기 모델 변경에 따라 구성이 달라지거나 투입되는 원재료의 추가적인 유형이 도입되는 경우 매번 새로운 프로그램 개발이 필요한 방식이다.

본 연구에서는 이러한 문제점을 해결하고자 다양한 변수 값들을 사용해 머신 러닝 기법으로 스스로 학습을 통해 조정이 필요한 상황을 예측하고 그에 따라 자동 보정이 이루어질 수 있는 인공지능 기반 기술을 제안한다. 연구의 목표는 사람의 개입에 따른 문제를 최소화 하고 시간의 흐름에 따라 투입 오차율을 줄일 수 있는 자율 계측 보정 원료 혼합기를 만들기 위해 98% 이상의 정확도를 가지는 머신러닝 기반 자동 보정 모델을 만드는 것으로 다음과 같은 기존 문제를 해결하기 위한 것이다.

- 원료의 형태, 크기, 무게, 개폐기 종류 등에 따라 투입량을 설정해도 실제 투입에 오차가 발생
- 현재 숙련된 작업자가 작업 환경에 맞게 작동 범위와 기기 설정을 변경해 조정해서 운영
- 초기 설정 값이 운영 중 발생하는 오차에 대응하지 못해 시간이 지남에 따라 오차율이 커져 불량 및 원재료 낭비의 원인이 됨.

제안된 인공지능 학습 모델 개발을 위해 그림 2와 같은 절차로 연구를 진행한다. 그림 2에서 파란색이 설정된 값이지만 실제 운영은 붉은색과 같은 데이터 조정에 의해 정상 운영이 진행된다. 실험 과정에서 학습을 통해 시간 흐름에 따라 자동적으로 오차 값이 보정되어 설정 값에 최대한 근접한 결과를 내도록 하는 것이 목표이다.

3.1 기초 데이터 수집

기초 데이터는 혼합부 제어에 사용되는 측정 및 제어값을 기준으로 선정했으며 측정 범위는 현재 판매 중인 원료 혼합기 제품 중 사이드 믹서 방식의 ASM-24 모델을 기준으로 선정했다. 측정 범위는 제조회사 기기 마다 차이가 발생할 수 있으나 추후 미세 튜닝을 통한 조정으로 해결이 가능하다.

데이터 항목 중 지도 학습을 위한 레이블에 해당하는 것은 adjust이며 특정 동작 상황에서 혼합기 설정 값 조정이 필요 했는지에 대한 결과 이다. 이는 실제 기기를 생산 공정에서 사용하면서 기존과 같은 방법으로 관리자에 의해 조정된 결과 값이다. 그림 3은 수집된 데이터 분포 결과를 보여준다.

표 1. 실험 데이터 세트
Table 1. Experimental data sets

Data	Unit	Description
id	sequence	sequential number of data
work_cnt	count	weekly working count
day_w_cnt	count	daily working count
residual	mg	residual amount in mixer
entry_unit_g	g	input amount of 1 unit
mix_cnt	count	number of mixed materials
color_g	g	amount of color materials
material_shape	type(1~3)	main material shape
add_g	g	extra add amount
micro_log	bool	micro adjustment status
motor_log	bool	motor event occurrence
power_log	bool	power event occurrence
material_type	type(0,1)	type(0:new, 1:recycled)
adjust	bool	adjustment, 0: YES, 1: NO

3.2 머신러닝 학습 모델 설계

플라스틱 사출성형 공정은 기본적인 구성요소가 동일하며 혼합기의 운영을 통해 입력 값들과 결과에 대한 데이터 수집이 가능하다.

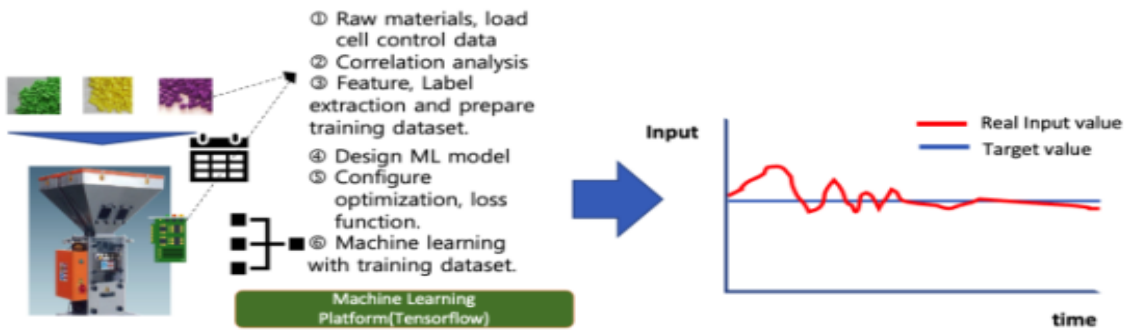


그림 2. 연구 절차 및 목표
Fig. 2. Research process and good

	id	work_cnt	day_w_cnt	residual	entry_unit_g	mix_cnt	color_g	material_shape	add_g	adjust	micro_log	motor_log	power_log	material_type
count	5000.000000	5000.000000	5000.000000	5000.000000	5000.000000	5000.000000	5000.000000	5000.000000	5000.000000	5000.000000	5000.000000	5000.000000	5000.000000	5000.000000
mean	2500.500000	45.338400	20.104600	73.774200	93152.503000	2.396400	1.937913	1.881000	56.498800	0.096000	0.104400	0.060400	0.596800	0.294000
std	1443.520003	11.463166	11.467954	46.033729	2121.852197	1.147663	1.747666	0.839869	101.713802	0.294621	0.305809	0.23825	0.490589	0.455637
min	1.000000	23.000000	-3.000000	8.000000	9307.000000	1.000000	0.000000	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
25%	1250.750000	35.000000	10.000000	39.000000	91911.000000	1.000000	0.700000	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
50%	2500.500000	45.000000	20.000000	64.000000	93437.000000	2.000000	1.500000	2.000000	0.000000	0.000000	0.000000	0.000000	1.000000	0.000000
75%	3750.250000	55.000000	30.000000	98.000000	94608.000000	3.000000	2.500000	3.000000	101.000000	0.000000	0.000000	0.000000	1.000000	1.000000
max	5000.000000	67.000000	43.000000	224.000000	96651.000000	4.000000	10.000000	3.000000	635.000000	1.000000	1.000000	1.000000	1.000000	1.000000

그림 3. 데이터 분포 결과
Fig. 3. Data distribution

도메인 특성상 지도학습을 통해 모델을 생성하고 생성된 모델을 통해 실시간으로 입력되는 혼합기 데이터를 기반으로 예측과 보정을 진행하게 된다. 그림 4의 상관 관계 분석을 통해 레이블에 해당하는 컬럼인 adjust와 관련성이 낮은 데이터들을 선별해 최종적으로 work_cnt, day_w_cnt, micro_log, power_log 컬럼은 제거해 관련성이 높은 데이터 항목을 중심으로 최적화된 분석이 이루어질 수 있도록 했다.

다음 단계는 최적화된 학습 데이터를 기반으로 머신러닝 모델을 만드는 과정이다. 본 연구 대상의 데이터는 분류 문제에 해당하는 것으로 5천건의 데이터는 80:20으로 분할해 훈련과 검증용으로 사용하고 해당 데이터 조건으로 공정이 진행되었을 때 관리자에 의한 조정이 발생했는지 유무를 다섯가지 기계학습 모델을 통해 구성한다. 모델 구성은 scikit-learn 과 XGClassifier 라이브러리를 사용하고 파라미터와 설정 값을 본 연구에 맞게 구성 하였다.

다음은 모델 설계 과정으로 scikit-learn 으로 구성된 각 모델 구성과 주요 파라미터에 대한 선정 기

준과 설정 값을 중심으로 살펴본다. 각 모델에 대한 주요 파라미터는 표 2~표 6에서 확인할 수 있다.

3.2.1 KNN 모델 설계

metric은 KNN에서 인접값들의 거리를 구하는데 사용하는 방법에 해당하는 것으로 본 논문에서는 manhattan distance를 사용했다. n_neighbors 는 인접 이웃의 수를 지정하는 것으로 일반적으로는 1~10 이내가 가장 정확한 것으로 알려져 있으며 데이터의 특성에 따라 차이가 발생할 수 있다. 따라서 본 연구에서는 k 값을 변화시켜 가장 높은 정확도를 보여준 2로 설정했다.

표 2. KNN 모델 구성
Table 2. KNN model configuration

```
knn = KNeighborsClassifier(algorithm='auto',
                           leaf_size=30, metric='manhattan',
                           metric_params=None,
                           n_neighbors = 2, weights='distance')
```

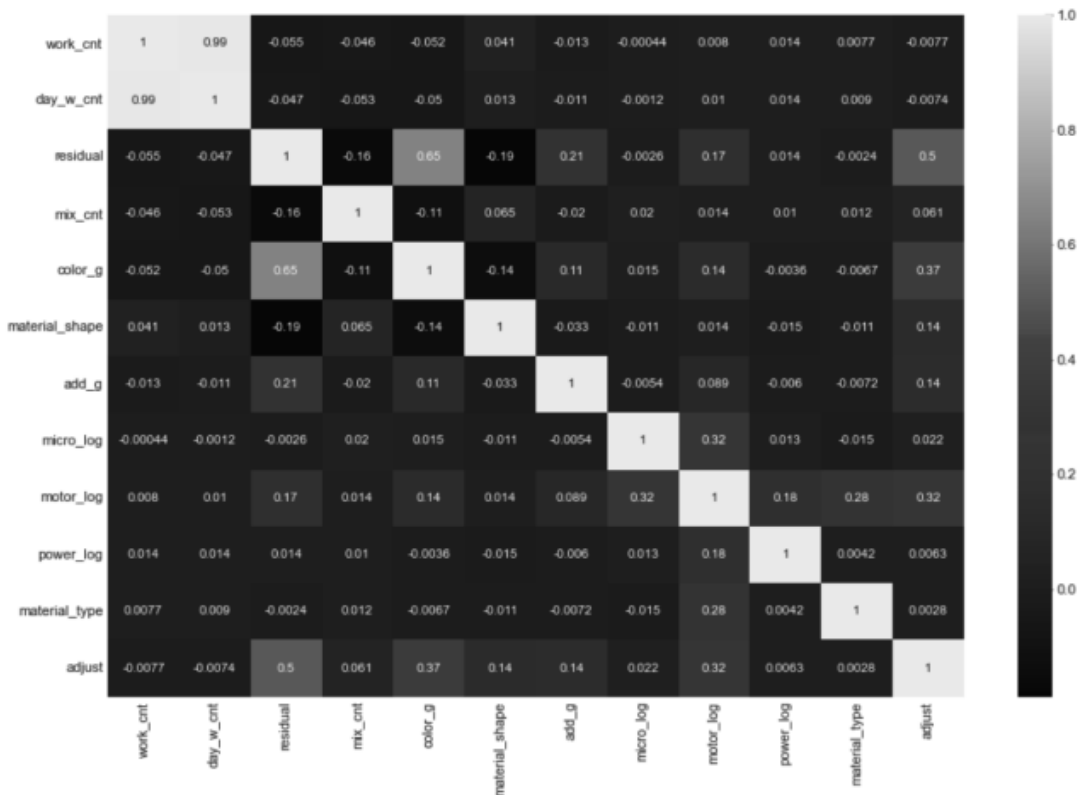


그림 4. 데이터 상관 관계 분석
Fig. 4. Data correlation analysis

3.2.2 Logistic regression

`fit_intercept`는 모형에 상수항이 있는가 없는가를 결정하는 인수로 본 데이터셋에는 상수항이 존재하므로 `True`로 설정한다. `penalty`는 L1, L2 regularization에 관한 것으로 L1은 LASSO model로 feature selection의 역할을 수행하기 때문에 설정했으며 `solver`는 회귀 분석 방법을 지정하는 것으로 선형모형을 선택하였다.

표 3. Logistic regression 모델 구성
Table 3. Logistic regression model configuration

```
lr = LogisticRegression(C=0.4,
    class_weight=None, dual=False,
    fit_intercept=True, intercept_scaling=1,
    penalty="l1", solver='liblinear')
```

3.2.3 Random forest

`n_estimators`는 랜덤 포레스트에서 생성할 트리의 개수이다. 전체 데이터 규모로 최적화 테스트를 거쳐 100으로 설정했으며 `max_features`는 `auto`로 설정해 크기에 따른 오차를 줄이도록 하였다.

표 4. Random forest 모델 구성
Table 4. Random forest model configuration

```
rfc = RandomForestClassifier(n_estimators = 100,
    max_depth = 5, criterion = "entropy",
    min_samples_split=2, min_samples_leaf=1,
    min_weight_fraction_leaf=0.0,
    max_features='auto',
    bootstrap=True, oob_score=False,
    n_jobs=None, random_state=None,
    verbose=0, warm_start=False,
    class_weight=None, ccp_alpha=0.0,
    max_samples=None)
```

3.2.4 SVM

SVM에서 커널은 기저 함수를 하나씩 정의하는 수고를 덜고 계산량을 줄이는데 도움이 된다. 대표적으로 `softmax`, `RBF`가 있는데 `RBF`는 가우시안 커널이라고도 하며 본 연구의 특성에는 차수가 무한대인 다항커널 구성인 `RBF`를 선택하였다.

표 5. SVM 모델 구성
Table 5. SVM model configuration

```
svc = SVC(C=1.0, kernel='rbf', degree= 2,
    gamma=1.0, random_state=None,
    coef0=0.0, shrinking=True,
    probability=True,tol=0.001,
    cache_size=200, class_weight=None,
    verbose=False,max_iter= -1)
```

3.2.5 XGBoost

익스트림 그레디언트 부스팅이라고 하며 정확도와 속도를 달성하기 위해 계산량을 극대화하는 방법이다. 랜덤 포레스트와 유사한 결정트리의 앙상블 기법이다.

표 6. XGBoost 모델 구성
Table 6. XGBoost model configuration

```
xgc = XGBClassifier(base_score=0.5,
    booster='gbtree', colsample_bylevel=1,
    colsample_bytrees=1, gamma=0,
    learning_rate=0.1, max_delta_step=0,
    max_depth = 3, min_child_weight=1,
    missing=None, n_estimators=100,
    objective='binary:logistic', random_state=0,
    reg_alpha=0, reg_lambda=1,
    scale_pos_weight=1, subsample=1)
```

`booster`는 XGBoost의 부스팅 기법을 지정하는 파라미터로 에이다부스트, 그레디언트 부스트 등이 있으며 예측오차를 기반으로 새로운 트리를 학습시키는 그레디언트 부스트(`Gbtree`)를 선택하였다.

3.3 예측 함수 구현

예측 함수는 앞에서 설계한 다섯 유형의 분류 모델들을 이용해 예측을 수행하는 함수이다. 데이터에 대한 정보와 예측 모델을 인자로 예측을 수행하고 결과를 리턴한다. 함수의 인자는 다음과 같이 정의 하였다.

- name: 예측 모델 이름(출력용)
- algo: 예측 알고리즘
- training_x: 훈련 데이터
- testing_x: 검증 데이터

- trainint_y: 훈련 데이터
- testing_y: 검증 데이터
- plot: 시각화를 위한 plot 객체

함수의 처리 과정은 표 7과 같다.

표 7. 예측함수 처리 의사코드
Table 7. Pseudo code for prediction function

```

for each model:
  algorithm fitting with trainint_x, training_y;
  predictions := predict using testing_x;
  get confusion matrix(testing_y, predictions);
  calculate roc_auc_score(testing_y, predictions);
  print result;
end for;
save top result model to reusable model file;
    
```

IV. 실험

4.1 실험 환경

실험은 클라우드 기반의 컴퓨팅 유닛을 사용했으며 Google의 Colab에서 Python 3.10과 GPU 백엔드를 사용해 XGBoost, SVM, Random Forest, KNN, Logistic Regression 등 총 다섯 가지 분류 기법으로 각각 학습 시키고 학습된 예측 모델을 다시 검증용 데이터셋으로 검증 후 AUC(Area Under Curve)를 산출해 예측 성능을 비교 실험한다.

표 8. 실험 환경
Table 8. Experimental environment

Item	Specification
Development workstation	Intel core i7, 16G ram, 1TB SSD
Cloud computing unit	Google Colab, GPU option
Language	Python 3.10
Libraries	Scikitlearn 1.2.2, XGBClassifier 1.7.6

그림 5는 실험 과정을 보여준다. 개발 환경에 따라 실험은 테스트 데이터 로드 -> 보정값 예측 요청 -> 학습 모델별 예측 결과 -> 결과 요약으로 진행 되었다.

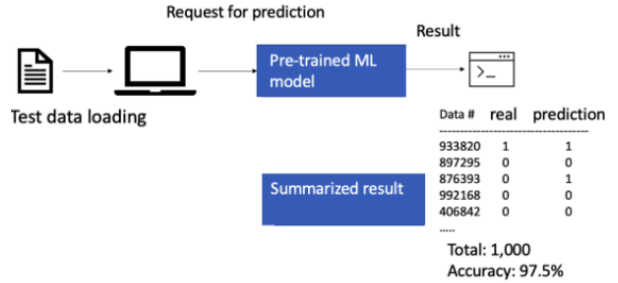


그림 5. 실험 과정
Fig. 5. Experimental process

4.2 실험 결과

3장에서 설계한 다섯 방식의 학습모델을 구현한 예측함수를 통해 실험한 결과는 표 9와 같다. 표에서 보여주는 것처럼 KNN모델의 AUC값이 가장 높게 계산 되었다.

표 9. 모델별 실행 결과
Table 9. Result of each model

Model	AUC
Logistic regression	0.8164644852354349
KNN	0.975073157754722
Random forest	0.9380952380952381
SVM	0.9666666666666667
XGBoost	0.9655493482309125

실험 결과 가장 정확도가 높은 학습 모델은 AUC 0.975를 보인 KNN Classifier이며 예측 결과 precision은 각각 0.99, 0.98이며 recall은 1.0, 0.95, f1-score는 1.0, 0.97이다. 이를 기반으로 산출된 accuracy는 0.99이며 AUC는 0.975로 다른 방법보다 높은 정확도를 보였다. 실험 수행 결과 리포트는 그림 6과 같다.

Prediction report:

	precision	recall	f1-score	support
0	0.99	1.00	1.00	895
1	0.98	0.95	0.97	105
accuracy			0.99	1000
macro avg	0.99	0.98	0.98	1000
weighted avg	0.99	0.99	0.99	1000

Area under the curve: 0.975073157754722

그림 6. KNN 실험결과
Fig. 6. Result of KNN method

성능 결과 도출을 위한 confusion matrix는 표 10과 같은 2x2 테이블로 표현된다.

표 10. KNN confusion matrix
Table 10. Confusion matrix of KNN model

TP(True Positive) 100	FN(False Negative) 5
FP(False Positive) 2	TN(True Negative) 893

결과는 TP, FN 이 각각 100, 80 으로 모델의 예측이 오차가 적음을 보여주고 있으며 이를 기반으로 accuracy, precision, recall 등이 앞에서 설명한 것 같이 산출 되었다.

그림 7은 ROC curve, AUC를 보여주는 차트이다. 임계값에 따라 FPR을 0~1 까지 변화시켜가며 TPR의 변화를 기록한 ROC curve 는 그림과 같이 가장 이상적인 형태로 모델이 우수함을 보여주고 있다. AUC는 ROC의 아랫부분 면적으로 0~1 사이의 값을 가지게 되는데 실험 결과는 0.97로 모델이 잘 설계되었음을 보여준다.

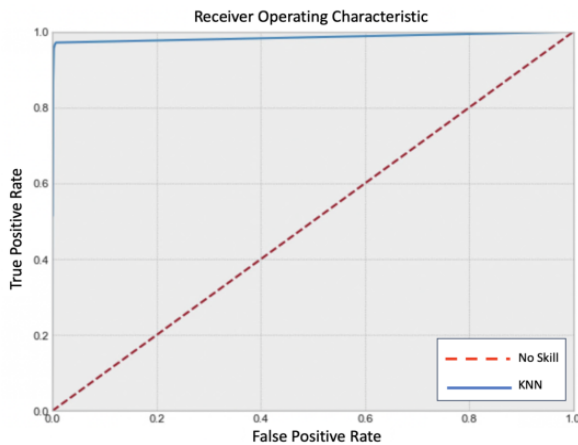


그림 7. KNN 실험결과
Fig. 7. Result of KNN method

4.3 학습 모델 성능 검증

실험을 통해 다섯 가지 분류 기법 중 가장 높은 정확도를 보인 KNN 학습모델을 실제 공정에서 취득된 5천 건의 데이터를 사용해 실효성 있는 예측

결과가 나오는지 검증 실험을 진행 하였다. 또한 처리 성능이 실제 기기 운영에 영향을 미치지 않기 위해서는 연산 속도도 중요하기 때문에 연산 속도를 포함해 성능 검증을 진행 했다.

실험 시나리오는 다음과 같다.

- 실제 공정 운영에서 취득된 5천건의 데이터
- 1셋트당 100개의 데이터를 임의 추출해서 예측하고 실제 결과 값과 비교
- 총 100셋트 수행
- 총 1000개의 데이터에 대해 수행하게 되면 각 셋트의 평균 성공률을 합산해 10셋트에 대한 평균 성공률을 구함.
- 평균 성공률 97% 이상, 처리속도 1초 미만.

표 11. KNN 모델로부터 정확도 실험 로그
Table 11. Log data for KNN model accuracy test

```

>>>>> start time: 2023-06-18 13:58:35.028927
#####
#           Experiment Result           #
#####
* Total set: 10, sample data per set: 100,
* Total test data: 1000

[Set-1] Start ->
* 1/100 => prediction value: 0 , real value: 0
residual      44.0
mix_cnt       4.0
color_g       0.8
material_shape 1.0
add_g         0.0
motor_log     0.0
adjust        0.0
Name: 872, dtype: float64

-----
...
* total: 100, success: 98, fail: 2, success rate: 98.0%
....

=====
* Total: 1000, Average success rate: 98.3%
=====

>>>>> end time: 2023-06-18 14:00:24.661910
    
```

실험결과 총 1,000개의 데이터에 대해 평균 성공률 98.3%를 달성해 실험 목표치인 97%를 상회한 결과를 얻을 수 있었다.

연산속도는 인공지능 학습 모듈의 처리 속도로 1건의 데이터를 처리하는데 소요되는 시간으로 정의하고 앞에서 진행한 실험과정에 예측 함수 호출 전/후 시간을 계산하고 총 10세트(1000건)에 대한 데이터에 대해 평균 처리 속도를 구한 결과 약 0.093초로 인공지능 학습모델을 사용해 실시간 데이터를 처리하는데 있어 공정상 문제가 되지 않을 수준의 성능이 확보 되었다고 볼 수 있다.

V. 결론 및 향후 과제

스마트 팩토리는 시대의 흐름이고 중소기업 공장에서도 도입이 증가한다. 다양한 스마트 팩토리 기술과 솔루션들이 있지만 플라스틱 사출성형과 같은 특정 산업분야의 솔루션은 부족한 현실이다.

본 논문에서는 플라스틱 사출 성형 공정 중 원료 혼합기의 자동 보정을 위해 인공지능 학습 모델을 사용한 방법을 제안 했으며 널리 알려진 다섯 개의 인공지능 분류 기법을 동일하게 학습시켜 최적의 결과를 보여준 KNN 기반 학습모델 기반으로 실제 공장 환경에서 실험을 통해 평균 98.3%의 정확도를 확인할 수 있었으며 처리 속도에서도 1000건 처리 속도 데이터 1건당 평균 0.093초의 처리 속도를 달성해 실제 공정운영에 충분한 성능을 보여줄 수 있었다.

본 논문에서는 혼합기 자동 보정 분야에서 이전에 시도된 바 없었던 인공지능 모델에 기반 한 투입량 자동 보정 기술을 제시하였고 실험을 통해 실제 적용 가능함을 보였다. 다만 제품화를 위해서는 예측에 따라 혼합기의 제어를 통해 혼합기를 컨트롤 하는 모듈이 추가되어야 하는 점 등에서 본 연구의 한계를 찾을 수 있다. 따라서 향후 연구과제에서는 본 연구 결과물을 통한 예측 결과에 따라 실제 투입량을 조절할 수 있는 모듈을 설계하고 개발하고자 한다.

References

- [1] S. J. Kim, "Technology Research Trends of Smart Factory through the Keyword Network Analysis", *Journal of the Korea Academia-industrial cooperation Society*, Vol. 23, No. 5, pp. 17-23, May 2022. <http://dx.doi.org/http://dx.doi.org/10.5762/KAIS.2022.23.5.17>.
- [2] J. Lee, S. Kim, and H. Kim, "A Study on Smart Factory Construction Plan of Small and Medium-Sized Enterprises in Korea", *Yonsei Business Review*, Vol. 59, No. 2, pp. 101-126, Dec. 2022. <http://dx.doi.org/10.55125/YBR.2022.12.59.2.101>.
- [3] Y. Lee, W. Lee, and S. Lee, "The Development of Protocol for Construction of Smart Factory", *Journal of IKEEE*, Vol. 23, No. 3, pp. 1096-1099, Sep. 2019.
- [4] S. Cha, "Smart Factory Standard Trend and System Architecture", *The Magazine of the IEEE*, Vol. 43, No. 6, pp. 31-37, Jun. 2016.
- [5] J. Jung and D. An, "Development of Remote Control System based on CNC Cutting Machine for Gradual Construction of Smart Factory Environment", *KIPS Transactions on Computer and Communication Systems*, Vol. 8 No. 12, pp. 297-304, Aug. 2019. <http://dx.doi.org/10.3745/KTCCS.2019.8.12.297>.
- [6] J. Lim, D. Jo, S. Lee, and H. Park, "A Case Study for the Smart Factory Application in the Manufacturing Industry", *Korean Journal of Business Administration*, Vol. 30, No. 9, pp. 1609-1630, Sep. 2017.
- [7] J. Machacek, P. Staubach, C. E. G. Tavera, T. Wichtmann, and H. Zachert, "On the automatic parameter calibration of a hypoplastic soil model", *Acta Geotech*, Vol. 17, pp. 5253-5273, Dec. 2022. <https://doi.org/10.1007/s11440-022-01669-4>.
- [8] S. Park, D. Seong, and Y. Byun, "A Hybrid Collaborative Filtering based on Online Shopping Patterns using XGBoost and Word2Vec", *Journal of KIIT*, Vol. 18, No. 9, pp. 1-8, Sep. 2020. <http://doi.org/10.14801/jkiit.2020.18.9.1>.
- [9] D. Jeon, B. Lee, and H. Hwang, "Design and

Implement of Smart Gateway Interface API for Real-time Monitoring in Smart Factory", Journal of Korea Multimedia Society, Vol. 22, No. 5, pp. 601-612, May 2019. <http://doi.org/10.9717/kmms.2019.22.5.601>.

저자소개

황 희 정 (Heejoung Hwang)



2008년 2월 : 인천대학교

컴퓨터공학과(공학박사)

2000년 9월 ~ 현재 : 가천대학교

컴퓨터공학과 교수

관심분야 : 소프트웨어 공학,

스마트 팩토리, 헬스케어, 유헬스,

소프트웨어 아키텍처