

배깅 및 스택킹 기반 앙상블 기계학습법을 이용한 고성능 콘크리트 압축강도 예측모델 개발

곽 윤 지¹·고 채 연¹·곽 신 영^{2†}·임 승 현³

¹국립 한밭대학교 건설환경공학과 석사과정, ²국립 한밭대학교 건설환경공학과 부교수,
³경북대학교 융복합시스템공학과 플랜트시스템전공 조교수

Development of a High-Performance Concrete Compressive-Strength Prediction Model Using an Ensemble Machine-Learning Method Based on Bagging and Stacking

Yun-Ji Kwak¹, Chaeyeon Go¹, Shinyoung Kwag^{2†} and Seunghyun Eem³

¹Graduate Student, Department of Civil & Environmental Engineering, Hanbat National University, Daejeon, 34158, Korea

²Associate Professor, Department of Civil & Environmental Engineering, Hanbat National University, Daejeon, 34158, Korea

³Assistant Professor, Department of Convergence and Fusion System Engineering, Kyungpook National University, Sangju, 37224, Korea

Abstract

Predicting the compressive strength of high-performance concrete (HPC) is challenging because of the use of additional cementitious materials; thus, the development of improved predictive models is essential. The purpose of this study was to develop an HPC compressive-strength prediction model using an ensemble machine-learning method of combined bagging and stacking techniques. The result is a new ensemble technique that integrates the existing ensemble methods of bagging and stacking to solve the problems of a single machine-learning model and improve the prediction performance of the model. The nonlinear regression, support vector machine, artificial neural network, and Gaussian process regression approaches were used as single machine-learning methods and bagging and stacking techniques as ensemble machine-learning methods. As a result, the model of the proposed method showed improved accuracy results compared with single machine-learning models, an individual bagging technique model, and a stacking technique model. This was confirmed through a comparison of four representative performance indicators, verifying the effectiveness of the method.

Keywords : high-performance concrete, compression strength, machine-learning method, ensemble technique, bagging, stacking

1. 서론

고성능 콘크리트(High Performance Concrete: HPC)는 전 세계적으로 건설 산업에서 자주 사용되는 건설 재료 중 하나이다. HPC는 일반적으로 시멘트, 물, 잔골재와 굵은 골재의 4가지 기본 콘크리트 구성요소에 2종의 추가 시멘트질 재료인 플라이애시, 고로슬래그 그리고 1종의 화학 혼합물인 가소제가 추가되어 제작된다(Yeh, 1998). 여기서, 2종의 추가 재료인 플라이애시 및 고로슬래그는 기존 시멘트를 대체하는 재료로서

대체 비율이 10~60%까지 사용되고 있고(Gartner, 2004; Juenger et al., 2011; Lam et al., 1998), 이는 시멘트의 사용량을 낮추어 시멘트 생산을 위한 CO₂ 저감을 위해 적절한 재료로 확인되고 있다(Asteris et al., 2016; Rutkowska et al., 2020).

그러나 이러한 2종의 시멘트질 재료 사용은 기존 방법으로는 콘크리트 압축강도(Compressive Strength: CS) 예측을 어렵게 하여 좀 더 향상된 예측모델 개발이 필요하다(Asteris and Mokos, 2020; Sun et al., 2021). 이러한 맥락에서 HPC 압축강도 예측모델 개발 연구는 크게 통계적 접근 방법 사용, 혹은 기

†Corresponding author:

Tel: +82-42-821-1102; E-mail: skwag@hanbat.ac.kr
Received August 11 2022; Revised December 6 2022;
Accepted December 27 2022

© 2023 by Computational Structural Engineering Institute of Korea

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.

계학습방법 사용의 두 가지 주요 흐름에서 수행되고 있다.

통계적 접근 방법 사용 연구는 실험 데이터와 미리 정의된 방정식을 기반으로 다중 선형 또는 비선형 회귀분석을 수행하여 HPC 압축강도 예측모델을 개발하였다(Ahmad and Alghamdi, 2014; Atici, 2011; Jerath 1983; Ozbay *et al.*, 2011; Yeh, 1998; Zain and Abd, 2009). 이러한 연구는 기본적으로 모델 방정식의 잘 정의된 기본 형식을 제시하지 않고는 본질적으로 예측모델의 오류를 줄일 수 없다는 한계를 가지고 있다. 또한, HPC 압축강도 응답은 여러 입력 변수의 비선형성이 매우 높은 함수 형태이기 때문에 방정식의 기본 형식을 찾는 데에는 어려움이 있다.

따라서, 이러한 통계적 접근 연구의 문제점을 해결하기 위하여 여러 가지 기계학습 방법이 HPC 압축강도 예측모델 개발 연구에 적용된 바 있다. 예측모델 개발 연구를 위하여 사용된 방법은 인공신경망 기법, 서포트 벡터 머신 기법, 가우시안 프로세스 회귀 등 다양한 형태로 적용되었다(Atici, 2011; Cheng *et al.*, 2012; Chou *et al.*, 2011; Kasperkiewicz *et al.*, 1995; Neshat *et al.*, 2012; Öztaş, 2006; Yeh, 1998; Zhang *et al.*, 2020). 이러한 연구들은 단일 기계학습 예측모델 사용으로 응답을 예측하는데 있어 모델 과적합 문제, 코컬해 수렴 문제 등으로 상황에 따라 서로 다른 성능을 보여준다. 따라서, HPC 압축강도 예측모델 개발에 있어 복합 기계학습모델 적용에 대한 필요성이 논의된 바 있다(Apostolopoulou *et al.*, 2019). 또한, 이러한 단일 기계학습모델 사용이 동일한 조건에서 어떠한 성능 차이를 보이는지에 대한 비교 연구가 필요한 상황이다.

이러한 배경 아래, 본 연구에서는 배깅 및 스택킹 기법의 앙상블 기계학습법을 이용하여 HPC 압축강도 예측모델을 개발하는 것을 목적으로 한다. 이는 기존의 앙상블 방법인 배깅 기법과 스택킹 기법을 통합하여 새로운 앙상블 기법을 제시하고, 이를 통하여 단일 기계학습모델의 문제점을 해결하여 예측 성능을 높이고자 하였다. 이 논문의 핵심적인 기여는 기존의 배깅 앙상블 기법 혹은 스택킹 앙상블 기법을 개별적으로만 활용하지 않고(Asteris *et al.*, 2021), 이 두 기법을 동시에 활용하여 모델의 예측성능을 향상시키는 데에 있다. 스택킹 기법 내 배깅 기법 모델의 활용은 모델 예측치의 불확실성에 대한 강인성을 증가시킴으로써 예측성능을 향상시킬 것으로 보인다. 또한, 본 연구에서는 비선형 회귀분석, 인공신경망(ANN), 가우시안 회귀분석(GPR) 및 서포트 벡터 머신(SVM)을 통한 단일 기계학습방법 예측모델을 제시하고 이에 대한 성능을 분석하였다.

2. 실험 데이터 및 방법

2.1 HPC 압축강도 실험 결과

본 연구에서 사용한 HPC 압축강도 실험 결과는 총 1030 세

Table 1 Descriptive statistics of the input and output parameters used for predicting HPC CS

Variable	Statistics			
	Min	Average	Max	STDEV
Cement(kg/m ³)	102.0	281.2	540.0	104.5
Blast Furnace Slag(kg/m ³)	0.0	73.9	359.4	86.3
Fly Ash(kg/m ³)	0.0	54.3	200.1	64.0
Water(kg/m ³)	121.8	181.6	247.0	21.4
Superplasticizer(kg/m ³)	0.0	6.2	32.2	6.0
Coarse Aggregate(kg/m ³)	801.0	972.9	1145.0	77.8
Fine Aggregate(kg/m ³)	594.0	773.6	992.6	80.2
Age(days)	1.0	45.7	365.0	63.2
HPC Compressive Strength(MPa)	2.3	35.8	82.6	16.8

트로 구성되었으며, 이는 Yeh(1998)에 의하여 제공되었고, UCI (University of California, Irvine)의 Machine learning 데이터 저장소(<http://archive.ics.uci.edu/ml/datasets/Concrete+Compressive+Strength>)에 제시되어 있다. 이 실험 결과의 하나의 데이터 세트는 (1) 시멘트(Cement), (2) 고로슬래그(Blast Furnace Slag), (3) 플라이애시(Fly Ash), (4) 물(Water), (5) 고성능 감수제(Superplasticizer), (6) 굵은 골재(Coarse Aggregate), (7) 잔골재(Fine Aggregate), (8) 재령(Age)의 8개의 입력 변수값과 1개의 HPC 압축강도의 출력 변수값으로 이루어진다. Table 1은 모든 실험 데이터에 대한 입력 변수값과 출력 변수값의 통계적인 정보를 나타낸다.

2.2 방법

2.2.1 단일 모델 기계학습 방법

본 연구에서는 단일 모델 기계학습 방법으로서 각각 비선형 회귀분석(Nonlinear Regression analysis: NR), 서포트 벡터 머신(Support Vector Machine: SVM), 인공신경망(Artificial Neural Network: ANN) 및 가우시안 프로세스 회귀(Gaussian Process Regression: GPR)를 사용하였다.

본 연구에서 사용한 NR 방법은 입력 변수(x_i)와 출력 변수(y) 사이의 관계를 아래와 같은 2차 다항식의 함수 형태로 가정하였다. 따라서, NR 모델은 위에서 제시된 실험 데이터를 기반으로 정의된 2차 다항식 함수 모델의 예측값과 관련 실험 데이터 출력값의 오차가 가장 적은 다항식 함수 모델의 계수 값(α)을 결정하는 방식으로 수행되었다(Kwag *et al.*, 2018).

$$y = \alpha_0 + \sum_{i=1}^n \alpha_i \cdot x_i + \sum_{i=1}^n \alpha_{ii} \cdot x_i^2 + \sum_{i < j} \alpha_{ij} \cdot x_i \cdot x_j \quad (1)$$

SVM 방법은 데이터 분류 혹은 회귀분석에 널리 사용되고 있고, 이는 비모수 기법으로서 커널 함수에 의존하여 실험 데이터를 잘 묘사하는 함수를 찾는 것을 목적으로 한다. 비선형성이 큰 데이터를 선형 함수 형식으로 표현하여 예측값(y)을 산출하기 위해 입력 변수(\mathbf{x})는 mapping 기법을 활용하여 비선형 고차원의 공간으로 변형시킨다(식 (2) 참조). 여기서, \mathbf{W} 는 가중치 벡터이고, $\varphi(\mathbf{x})$ 는 비선형 mapping을 위한 기능(feature) 함수이고, “ \cdot ”은 벡터 내적을 의미하며, b 는 상수를 나타낸다. 사용하는 함수에 대한 적합도는 손실함수(즉, 예측값과 실험값의 차이)로 확인이 가능하며, 이러한 손실함수 값을 최소화하는 최적화된 모델을 찾는 것으로 관련 SVM 회귀 모델을 만들 수 있다(Burges, 1998).

$$y = \mathbf{W} \cdot \varphi(\mathbf{x}) + b \quad (2)$$

ANN 방법은 뇌 신경계의 정보 처리 구조를 모방하여 만든 알고리즘이다. 신경망은 입력층(Input Layer), 은닉층(Hidden Layer)과 출력층(Output Layer)으로 구분된다. 입력층은 시스템 외부로부터 입력자료를 받아들여 시스템으로 전송하며, 이를 구성하는 뉴런의 개수는 입력 변수의 개수와 같다. 은닉층은 시스템 안쪽에 자리 잡고 입력값을 넘겨받아 이러한 값들을 처리한다. 출력층은 은닉층으로부터 받은 정보를 기반으로 입력 변수에 대한 결과값을 출력한다. 각 층에는 하나 이상의 뉴런이 존재하며, 모든 뉴런은 가중치(Weights)와 편향(Bias)을 통하여 서로 연결되어 있다. 구체적으로 은닉층 및 출력층에 존재하는 각각의 뉴런은 입력되는 값에 가중치를 곱하고, 편향값을 더하여 모두 합한 후, 활성화 함수(Activation Function)를 통하여 각 층에서의 결과값을 도출한다. 이러한 과정을 거쳐 출력층을 통하여 최종적으로 예측된 값의 오차를 최소화하기 위해 역방향으로 오차를 전파시키면서 각 층의 가중치를 업데이트하고, 최적의 학습 결과를 찾아가는 역전파 알고리즘을 사용한다(McClelland *et al.*, 1986). 결론적으로 인공신경망의

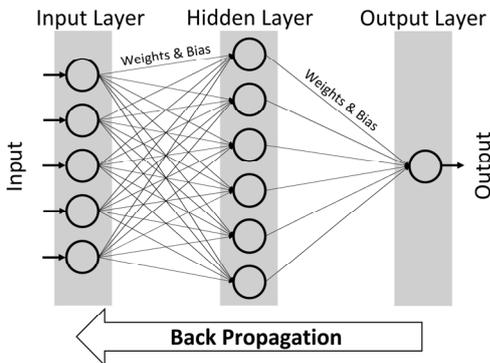


Fig. 1 Artificial neural network model architecture

학습은 이러한 오차를 최소화하는 가중치를 찾는 목적으로 순전파와 역전파를 반복하여 최종적인 ANN 예측모델을 만들게 된다. Fig. 1은 ANN의 계층구조를 개념적으로 보여주고 있다.

GPR 방법은 가우시안 분포(Gaussian Distribution)와 확률과정(Random Process)의 개념을 하나로 통합하여 회귀분석을 수행한다(Alpaydin, 2020). 이는 비모수 회귀 모델 중의 하나로서 커널 함수 기반 확률론적 모델을 사용한다. GPR 방법은 구체적으로 (1) 입력 변수 데이터(\mathbf{x})를 다차원의 공간으로 변환하고, 변환된 함수($\mathbf{h}(\mathbf{x})$)에 가중 계수(β)를 곱한 후, (2) 평균이 0이고 공분산이 커널 함수($k(\mathbf{x}, \mathbf{x}; \theta)$)로 이루어진 가우시안 분포($\mathbf{f}(\mathbf{x})$)를 더하여 (3) 최종적인 확률적 응답(y)을 찾게 된다(식 (3)~(4) 참조). 여기서, 가중계수(β), 커널 함수 매개변수(θ)와 예측값(y)의 분산 값은 실험 데이터를 기반으로 최우도법을 사용하여 최적의 파라미터 값을 산출하게 된다. 결과적으로 산출된 파라미터 값을 기반으로 GPR 방법의 예측모델을 최종적으로 구성하게 되고, 이를 통하여 응답 값을 예측하게 된다.

$$y = \mathbf{h}(\mathbf{x})^T \beta + \mathbf{f}(\mathbf{x}) \quad (3)$$

$$\mathbf{f}(\mathbf{x}) \sim N(0, k(\mathbf{x}, \mathbf{x}; \theta)) \quad (4)$$

2.2.2 앙상블모델 기계학습 방법

앙상블 기계학습법은 주어진 데이터로부터 여러 단일 기계학습법 모델을 생성하고, 이들을 적절히 조합하여 보다 정확도가 높은 모델을 생성하는 것을 목적으로 한다. 앙상블 기법은 크게 배깅, 부스팅 및 스택킹 등의 3가지 기법으로 구분된다.

배깅(Bagging)은 Bootstrap Aggregating을 의미하고, 기계학습 알고리즘의 안정성과 정확도를 향상시키기 위해 고안되었다(Breiman, 1996). Fig. 2는 배깅 방법을 개념적으로 보여주고 있다. 구체적으로 배깅 방법은 주어진 학습 데이터를 무작위 복원 추출하여 동일한 크기를 가진 여러 개의 학습 데이터

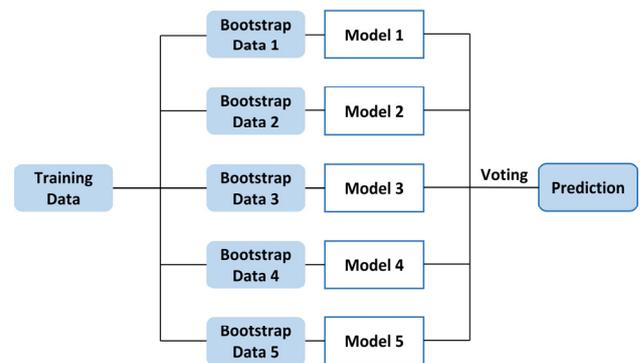


Fig. 2 Bagging architecture

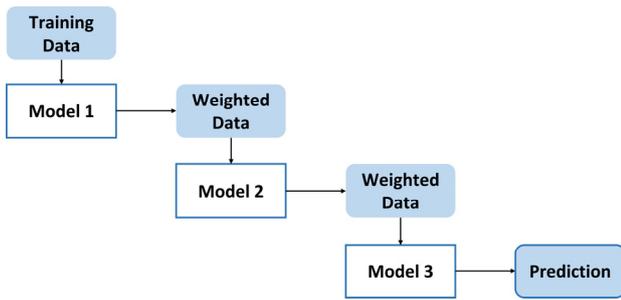


Fig. 3 Boosting architecture

(Bootstrap Data 1-5)를 도출하고, 이러한 개별적인 모델들을 통하여 얻어진 예측값을 바탕으로 Voting(본 연구에서는 평균값을 이용함)을 통하여 최종 예측값을 도출하게 된다. 이러한 배깅 과정은 개별 모델들의 분산을 줄일 수 있고, 과적합을 피할 수 있게 한다.

부스팅(Boosting) 기법은 독립적인 배깅 기법과는 달리 학습 데이터를 기반으로 순차적으로 모델을 생성하는 방법이다 (Fig. 3 참조)(Schapire *et al.*, 1998). 구체적으로 각 단계별로 생성된 모델 예측 결과의 정확도에 따라 단계별 학습 데이터에 가중치를 부여하고, 부여된 가중치가 적용된 학습 데이터를 기반으로 다음 모델을 생성하여 예측 성능을 높게 된다. 이러한 과정에서 기본적으로 예측이 잘 된 데이터는 가중치를 적게 주고, 잘못 예측된 데이터는 가중치를 높게 주는 방식을 활용한다. 부스팅 기법은 최종적으로 이러한 방식을 연속적으로 적용하여 잘못 예측된 데이터에 대한 정확도가 높아지도록 새로운 예측 규칙을 만드는 것으로서 보다 정확한 모델을 개발하는 것에 초점을 둔다. 이러한 부스팅 과정은 순차적으로 편향(bias)을 조정함으로써 모델의 성능을 높이고, 배깅에 비하여 오차가 적어질 수 있으나 속도가 느리고 과적합될 가능성이 있다.

스택킹(Stacking) 기법은 개별적인 단일 모델을 서로 결합해 예측 결과를 얻는다는 점에서 배깅 및 부스팅 기법과 공통점이 있으나 개별 모델(Model 1-5)로 예측한 데이터를 기반으로 다시 예측 모델(Meta Model)을 생성한다는 점에 있어서 큰 차이점을 가지게 된다(Fig. 4 참조)(Syarif *et al.*, 2012). 즉, 이 기법은 단일 모델들을 통하여 예측된 결과를 산출하고, 이러한 산출된 예측 결과와 기존 학습 데이터 실제 결과를 서로 결합하여 새로운 학습 데이터를 생성한다. 이를 기반으로 다시 기계학습 방법을 통하여 최종적인 메타 모델을 생성하고, 최종 응답값을 예측하는 방식을 갖는다. 스택킹 기법은 일반적으로 각 개별 모델이 독립적으로 생성되기 때문에 이상치(outlier) 값에 대한 대응력이 높아 단일 모델의 오차보다 작은 값을 갖게 되어 성능이 우수하게 된다.

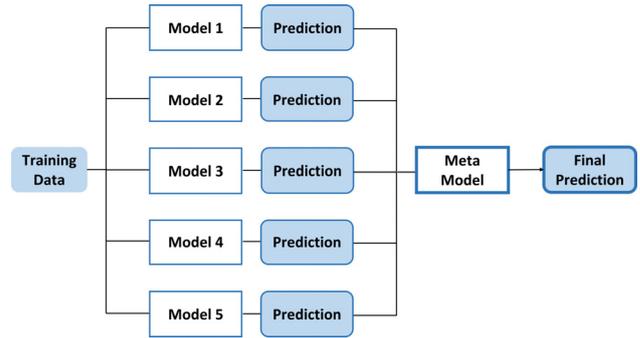


Fig. 4 Stacking architecture

3. 제안 방법론

앙상블 기계학습법은 위에서 설명한 바와 같이 여러 개의 단일 모델들을 결합하여 높은 예측 성능을 얻고자 하는 방법이다. 이러한 기법은 여러 모델의 예측값을 반영하여 최종 예측값에 대한 정확도를 높여주고 분산을 감소시킨다. 또한, 모델의 예측값을 결합하는 과정에서 단일 모델의 과적합 가능성을 줄여주게 된다. 따라서, 본 연구에서는 앙상블 기계학습법 중 배깅과 스택킹을 기반으로 모델의 HPC 압축강도 예측 성능을 높이는 방법을 제안하고자 한다.

이 연구에서 제안하는 방법은 기존의 스택킹 알고리즘을 기반으로 배깅 방법을 결합하여 최종 메타 모델(Meta Model)을 생성해 모델의 예측 성능을 높이는 것을 목표로 하였다. 즉, 제안된 방법은 기본적으로 단일 모델들을 결합하는 스택킹 기법에 추가적으로 배깅 기법을 포함하여 예측 성능을 개선하고자 하였다. 본 연구의 스택킹 기법에는 NR, GPR, ANN, SVM을 이용하여 단일 모델을 만들고, 추가적으로 배깅을 이용한 모델을 포함시켰다. 이때 활용한 배깅 기법의 경우 학습 그룹에 속한 데이터들을 무작위 복원 추출하여 ANN으로 학습시켜 나온 여러 예측값을 평균하여 사용하였다. 본 연구에서 배깅 기법의 기본 모델로 ANN을 사용한 이유는 ANN이 세미모수적(semi-parametric) 모델로서 은닉층의 뉴런 개수 등의 비모수(hyper-parameter)와 뉴런 사이 가중치인 모수(parameter)를 모두 포함하고 있기 때문이다.

Fig. 5는 본 연구에서 제안하는 스택킹과 배깅을 결합한 앙상블 방법을 개념적으로 보여주고 있다. 제안 방법 적용 절차를 구체적으로 설명하면 다음과 같다. 먼저, 데이터를 무작위로 섞어 학습 데이터(Training Data: TR)와 검증 데이터(Testing Data: TS)로 분류한다. 이 연구에서는 전체 데이터 중 80%는 학습 데이터로 20%는 검증 데이터로 활용한다. 학습 데이터를 NR, ANN, GPR, SVM에 적용하여 단일 모델을 생성하고, 추가적으로 배깅 기법을 적용하여 모델을 도출한다. 이때 배깅은 학습 데이터에 대하여 5번 무작위 복원추출 후 생성된

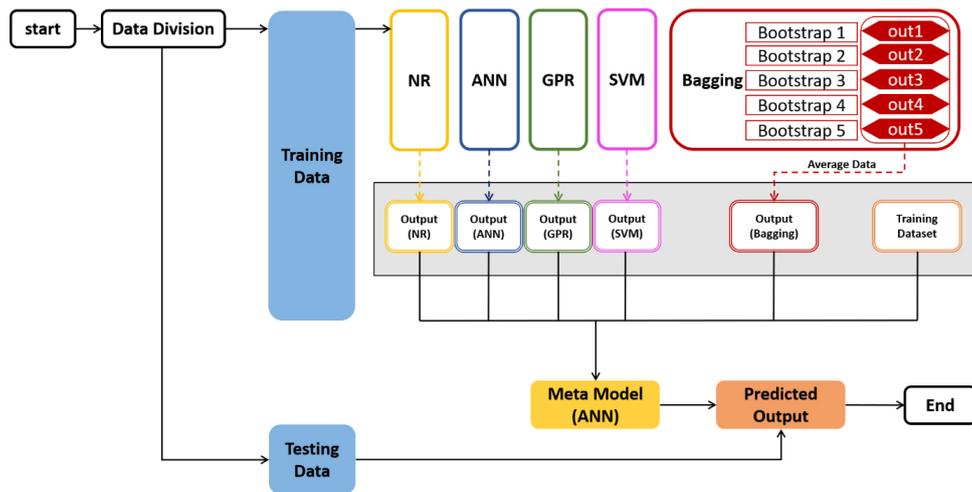


Fig. 5 Stacking model combined with bagging architecture

Bootstrap 그룹들에 대하여 ANN에 적용하고, 각 그룹의 예측값들에 대하여 평균을 내어 배깅 모델의 최종 예측값으로 만든다. 이러한 과정에서 추출된 단일 모델의 예측값들과 배깅 모델의 예측값을 기반으로 메타 모델을 위한 학습 데이터를 형성한다. 여기서, 메타 모델의 학습 데이터는 단일 모델 및 배깅 모델의 예측값을 입력 변수값으로 하여 병렬적으로 구성하고, 이에 해당하는 원래 값들을 목표 출력값으로 설정한다. 이러한 절차를 통하여 생성된 학습 데이터를 기반으로 ANN에 적용하여 최종 메타 모델을 생성하게 된다. 마지막으로 생성된 ANN 기반 메타모델은 한 번도 사용되지 않은 20%의 검증 데이터에 대하여 예측을 수행하고 모델의 정확도를 검증한다.

4. 결과 및 토의

본 연구에서 제안한 방법을 HPC 압축강도 실험결과에 적용하였다. 제안 방법의 유효성을 검증하기 위하여 동일한 학습 데이터를 기반으로 제안한 방법을 통하여 생성된 모델과 여러 가지 다른 모델들과의 성능을 비교하였다. 성능 비교는 모델의 정확도를 측정할 수 있는 R^2 , RMSE(Root Mean Squared Error), MAPE(Mean Absolute Percentage Error)와 VAF(Variance Account For)의 4가지 성능 지표를 활용하였다(Asteris and Mokos, 2020; Yilmaz *et al.*, 2010). R^2 는 값이 1에 가까울수록, RMSE와 MAPE는 값이 작을수록 VAF는 값이 클수록 모델의 정확도가 높음을 의미한다. Table 2는 4가지 성능 지표를 이용하여 모델의 정확도를 비교 정리하여 보여주고 있다. Fig. 6은 여러 모델 사이의 학습 데이터 및 검증 데이터에 대한 각각의 R^2 값을 방사형 그래프로 보여준다. Fig. 7은 대표적인 모델의 학습 데이터(TR) 및 검증 데이터(TS)에 대한 실제값과 예측값 사이의 관계를 데이터별로 보여주고 있다. 단일 모델의 경우, 제안된 방

Table 2 Details of performance indices of each model in training and testing phases

Model	Phase	R^2	RMSE	MAPE	VAF
MLR	Train	0.6217	10.3108	31.8897	62.1667
	Test	0.5871	10.5387	27.6473	59.1458
NR	Train	0.8177	7.1577	20.0210	81.7680
	Test	0.7708	7.8522	18.6949	77.3002
ANN	Train	0.9020	5.2475	13.5474	90.2033
	Test	0.8480	6.3941	15.1898	85.0939
SVM	Train	0.9377	4.1857	11.6657	93.7699
	Test	0.7718	7.8344	17.3709	77.4984
GPR	Train	0.9634	3.2088	8.0233	96.3358
	Test	0.8888	5.4686	12.4416	88.9443
Bagging (ANN)	Train	0.9412	4.0639	10.2739	94.1259
	Test	0.8820	5.6327	11.8529	88.2973
Stacking (ANN)	Train	0.9732	2.7422	6.8283	97.3386
	Test	0.8813	7.8522	12.6149	88.2673
Proposed Model	Train	0.9770	2.5450	6.3236	97.6952
	Test	0.8936	5.3491	11.8318	89.5642

법에 적용되는 NR, ANN, SVM, GPR뿐만 아니라 다중선형 회귀(Multiple Linear Regression: MLR) 분석을 추가로 수행하여 비교·분석하였다. 앙상블 기계학습법은 기존 알고리즘을 이용하여 개별적으로 배깅과 스택킹에 ANN을 적용하여 수행했으며, 이러한 결과를 제안된 방법의 정확도와 비교하였다.

MLR은 두 개 이상의 입력 변수와 출력 변수 사이의 관계를 선형 상관관계로 가정하여 모델링하는 방법을 의미한다. 본 연구에서 HPC 압축강도 데이터의 입력 변수 8개와 출력 변수 1개 사이의 관계를 선형적인 상관관계로 추계하였을 때, MLR의 파라미터는 총 9개로 α_0 부터 α_8 까지 각각 -23.6481, 0.1196, 0.1043, 0.0885, -0.1501, 0.3068, 0.0188, 0.0193, 0.1146의 값이

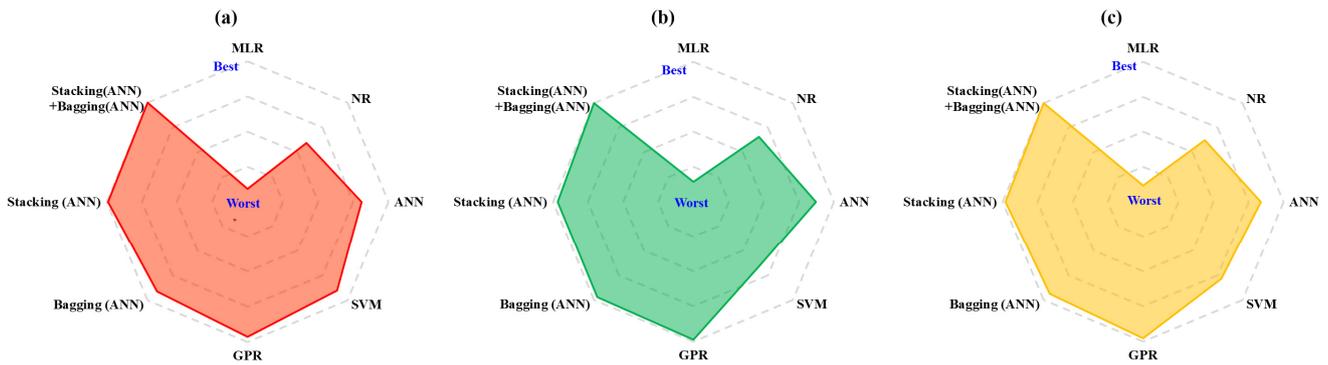


Fig. 6 R² performance comparison graph of the existing models and the proposed model on the data; (a) for training phase; (b) for testing phase; (c) for total R² performance

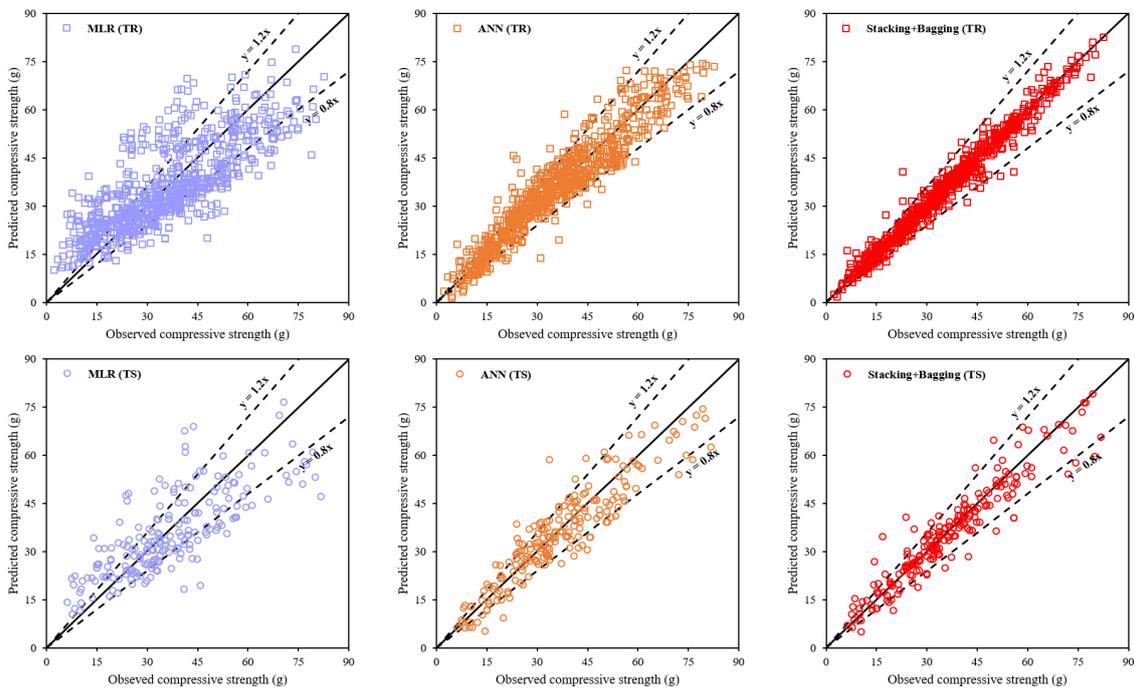


Fig. 7 Illustration of regression plot between actual and predicted value of CS

사용되었다. 이러한 MLR 모델에서 예측된 값에 대한 성능은 $R^2=0.5871$ 로 매우 낮은 정확도를 보였다. 사용된 데이터의 입력 변수와 출력 변수 사이가 강한 비선형적인 관계를 가지고 있는 것으로 분석되고, 이를 선형 관계로 나타내기에는 부적합한 것으로 판단된다.

위에서 기술한 바와 같이 선형 예측으로는 좋지 않은 성능을 보이기 때문에 n 차원의 일반 다항식 회귀 모델(식 (1) 참조)의 비선형 회귀분석(NR)을 수행하였다.

본 연구에서 사용된 NR의 다항식 차수는 $n=2$ 차이다. 이러한 NR 모델은 입력변수 간 상호 연계를 고려한 식으로서 다항식의 파라미터가 총 45개로 구성된다. NR 단일 모델은 학습 데이터와 검증 데이터에 대하여 정확도 값은 각각 $R^2=0.8177$, 및 $R^2=0.7708$ 을 나타냈고, 이를 MLR의 R^2 와 비교했을 때 각각

0.1960과 0.1837가량 성능이 향상되었다. 그러나 이 결과를 통해서도 단일 NR 모델이 비선형적인 관계를 정확하게 묘사하지 못한 것으로 보인다. 따라서, 단일 기계학습법, 앙상블 기계학습법 및 본 연구에서 제안하는 방법을 차례로 적용하였다.

먼저, SVM은 Mapping 기법을 사용하여 학습 데이터를 고차원으로 만드는 과정에서 Gaussian 커널 함수를 이용하였다. 커널 스케일은 heuristic 과정을 이용하여 자동으로 설정되게 하였다. 최적의 예측 모델을 구성하기 위한 SVR 솔버는 sequential minimal optimization을 이용하였다. SVM 모델의 학습 데이터에 대한 정확도는 $R^2=0.9377$ 이었고, 검증 데이터에 대한 정확도는 $R^2=0.7718$ 로 평가되어 SVM 모델은 MLR 및 NR 모델의 정확도를 크게 향상시키지 못하는 것으로 나타났다.

다음으로 동일한 학습 데이터에 ANN을 적용하여 모델을

생성하고 정확도를 분석하였다. 본 연구에서 사용한 ANN의 은닉층은 하나이며, 은닉층의 뉴런 개수는 10개로 설정하였다. 입력 변수가 8개이므로 입력층의 뉴런 개수도 8개이며, 출력층 뉴런의 최종값은 HPC 압축강도이다. 단일 모델로서의 ANN 수행에서 활성화 함수는 Sigmoid 함수를, 출력층에서는 선형 출력 함수를 사용하였다. 그리고 학습을 위한 알고리즘으로 Levenberg-Marquardt의 역전파 알고리즘을 활용하였고, ANN의 성능을 표현하는 오차 함수로는 평균 제곱 오차(Mean Squared Error)를 사용하였다. 총 반복 횟수(Epoch)는 21~29회로 나타났다. 이러한 단일 모델에서의 ANN은 학습 데이터에 대한 정확도가 $R^2=0.9020$ 이었고, 검증 데이터에 대한 정확도는 $R^2=0.8480$ 으로 평가되었다. 이를 MLR, NR 및 SVM 모델의 정확도와 비교해 보았을 때, 모델 성능이 좀 더 향상된 것으로 판단된다.

마찬가지로 GPR을 동일한 학습 데이터에 적용하였다. GPR 모델은 가중 계수로 상수값을 사용하였고, 커널 함수로 automatic relevance determination squared exponential 커널을 활용하였다. 로그가능도(Log-likelihood) 및 기울기 계산 방법은 QR 분해 기반 접근법을 사용하였고, 모델 도출을 위한 최적화 솔버는 quasi-newton 방법을 이용하였다. 스케일을 10으로 초기화하고, 신호와 잡음 표준편차를 응답 변수의 표준편차로 초기화하였다. 초기 커널 모수 값을 사용하여 GPR 모델을 피팅하고, 학습 데이터에서 예측 변수를 표준화하였다. 학습 데이터와 검증 데이터에 대한 GPR 모델의 정확도는 각각 $R^2=0.9634$, $R^2=0.8888$ 로서, 단일 기계학습모델 중에서 가장 높은 정확도를 보이는 것으로 판단된다. 수치적으로는 MLR 대비 정확도가 학습 및 검증 데이터에 대하여 각각 0.3417 및 0.3017가량 상승하였다.

GPR 모델이 다른 단일 모델과 비교하여 높은 정확도를 보이는 것은 다음과 같은 이유로 보인다. 1) GPR 모델은 비모수 모델로서 MLR 모델 및 NR 모델 등의 모수 모델에 비하여 비선형적인 데이터를 모사하는 유연도가 더 높기 때문인 것으로 판단된다. 2) GPR 모델은 SVM 모델과 비교하여 데이터의 불확실성에 대하여 확률적 예측과 예측의 불확실성 추정치를 제공함으로써 보다 정확한 예측을 가능하게 한 것으로 보인다. 3) GPR 모델은 유한한 수의 은닉층 및 뉴런 수를 가지는 ANN 모델에 비하여 비선형성이 큰 데이터를 매끄럽게 모사하기 때문에 더 높은 정확도를 보인 것으로 분석된다(Rasmussen, 2003). 이러한 단일 모델 예측 성능의 결과는 기존의 HPC 압축강도 예측 단일 모델 관련 연구 결과와도 일맥상통한다(Asteris *et al.*, 2021).

결과적으로 단일 기계학습법 모델 중에서는 GPR 모델이 가장 높은 정확도를 보였다. 따라서, 이러한 단일 학습 모델을 기

반으로 앙상블 기법을 활용하여 이러한 단일 기계학습법 모델보다 더 개선시키는 방법을 고안하고자 하였다. 먼저, 기존에 있는 앙상블 기법인 배깅 기법과 스택킹 기법을 각각 개별적으로 이용하여 정확도를 확인하였고, 본 연구에서 제안한 방법과 비교하여 제안하는 방법의 유효성을 분석하였다.

배깅 기법의 경우 주어진 학습 데이터에 대하여 무작위 복원 추출을 5번 수행하여, 5개의 개별적인 학습 데이터를 구성하였다. 무작위 복원 추출 시 선택되지 않은 데이터들은 검증용 데이터로 정리하여 활용하였다. 구성된 5개의 개별적인 학습 데이터를 ANN으로 모델링하여 검증용 데이터로부터 5개의 예측값을 얻고, 최종 예측값을 구하기 위하여 평균을 내어 사용하였다. ANN 기반으로 배깅 기법에 적용하였을 때, 검증 데이터에 대한 예측값은 $R^2=0.8820$ 으로 정확도가 가장 높았던 단일 모델의 GPR과 비슷한 정확도를 보였다.

스택킹 기법의 경우 4개의 단일 모델(NR, SVM, GPR, ANN)로부터 예측을 수행해 나온 예측값을 다시 학습 데이터로 만들어 ANN으로 학습시킨 후 메타 모델을 생성하여 최종 예측값을 도출하였다. 구체적으로 각각의 단일 모델들은 위에서 서술한 바와 같은 방법으로 수행되었으며, 단일 모델을 통해 나온 예측값을 학습 데이터로 이용하여 한 번 더 예측을 수행하였다. 스택킹 기법은 검증 데이터에 대해 $R^2=0.8813$ 의 정확도를 나타내어 배깅 모델 및 GPR 모델과 비슷한 정확도를 보였다. 결과적으로 대표적인 앙상블 기법인 배깅 기법과 스택킹 기법을 개별적으로 수행하였을 때, 대체로 높은 정확도를 보였으나 GPR 단일 모델의 정확도보다 높은 결과를 보이지 못했다.

다음으로 본 연구에서 제안하는 스택킹 기반에 배깅을 접목하는 앙상블 기법(즉, 배깅이 접목된 스택킹)을 동일한 학습 데이터에 적용하였다. 결과적으로 제안하는 방법으로 생성된 모델의 최종 예측값의 정확도는 검증 데이터에 대하여 $R^2=0.8936$ 으로 가장 높은 정확도를 보이는 것을 확인하였다. 이는 SVM 단일 모델의 검증 데이터에 대한 정확도보다 약 15.78% 향상된 수치이다(Table 3 참조). 제안된 방법은 기존 스택킹 기

Table 3 Percentage figures compared based on SVM model

Model	Phase	R ²	Percent Difference
SVM	Train	0.9377	-
	Test	0.7718	-
Bagging(ANN)	Train	0.9412	0.37%
	Test	0.8820	14.28%
Stacking(ANN)	Train	0.9732	3.79%
	Test	0.8813	14.19%
Proposed Model	Train	0.9770	4.19%
	Test	0.8936	15.78%

Table 4 The R^2 values and their means and standard deviations of each model for validation data in five different datasets(5-fold format)

	Proposed Model	MLR	NR	SVM	ANN	GPR	Bagging(ANN)	Stacking(ANN)
Dataset #1	0.8936	0.5871	0.7708	0.7718	0.8480	0.8888	0.8820	0.8813
Dataset #2	0.9373	0.6709	0.8527	0.7795	0.9097	0.9310	0.9131	0.9026
Dataset #3	0.9095	0.6103	0.8700	0.7297	0.8898	0.9088	0.8851	0.8913
Dataset #4	0.9164	0.5961	0.7634	0.8163	0.8811	0.9072	0.8814	0.9061
Dataset #5	0.9284	0.5282	0.7629	0.8499	0.8907	0.9243	0.8946	0.9122
Average	0.9200	0.6000	0.8040	0.7900	0.8840	0.9100	0.8910	0.9000
Standard deviation	0.0169	0.0457	0.0473	0.0409	0.0202	0.0147	0.0133	0.0110

법의 알고리즘에서 단일 모델의 수행 과정에 배깅 기법을 추가한 방법이다. 구체적으로 NR, ANN, SVM, GPR을 수행하여 나온 예측값과 무작위 복원 추출을 통해 5개의 학습 데이터를 구성하여 ANN을 적용해 나온 배깅 기법의 예측값까지 활용하는 것이다.

Table 4는 본 연구에서 사용한 1030개의 데이터를 20%씩 5-Fold로 만들어 4개의 Fold를 학습 데이터로 사용하고 나머지 하나의 Fold를 검증 데이터로 지정하여 총 5개의 서로 다른 구성의 데이터세트의 검증 데이터에 대한 모델의 예측성능을 분석한 표이다. 이러한 분석의 목적은 다른 구성의 데이터세트에 대한 제안된 방법의 유효성을 파악하고자 함이며, 이를 위하여 각 모델의 예측성능 결과를 R^2 값을 통해 나타내었다. 결과적으로 5개의 다른 구성의 데이터세트에 대해서도 제안된 방법의 성능이 일관적으로 가장 우수함을 확인할 수 있었다. 다음으로 R^2 값의 평균과 표준편차를 비교해 보면 제안된 방법이 평균적으로 가장 좋은 성능을 보이면서도 편차는 비교적 크지 않음을 확인할 수 있었다. 또한, 본 연구에서 사용한 주 데이터세트(Dataset #1)에 대해 제안된 방법을 10번 수행하였을 때, 도출된 모델의 R^2 값은 0.8993, 0.8940, 0.8912, 0.8955, 0.8897, 0.8899, 0.8900, 0.8928, 0.8924, 0.8939로 계산되었고, 이러한 값들의 표준편차는 0.0030으로서 모델의 편차가 크게 나타나지 않음을 확인할 수 있었다.

마지막으로 본 연구에서 사용된 컴퓨테이션 사양은 11th Gen Intel(R) Core(TM) i7-11700 @ 2.50GHz이며, RAM은 16.0GB이다. 이러한 환경에서 단일 모델의 경우 MLR, NR 및 SVM은 훈련에 소요된 시간이 1초 이내였으며, ANN과 GPR은 각각 3초 내외, 및 4초 내외로 소요되었고, 앙상블 기법 및 제안 방법에 대해서는 10초 정도 소요되었다. 즉, 각각의 모델을 형성하는데 소요된 계산시간은 모두 약 11초 이내로서 서로 간의 큰 차이는 발생하지 않음을 확인할 수 있었다(Table 5 참조).

결과적으로 본 연구에서 제안된 방법이 보다 더 개선된 모델을 생성할 수 있게 된 이유는 다음과 같은 것으로 추정된다. 메타 모델 생성 과정에서 학습 데이터 5개의 예측값의 입력 변

Table 5 The Computational time taken for each model to complete training and validation

Model	Computational Time(sec)
MLR	0.1105
NR	0.2215
SVM	0.0186
ANN	3.1539
GPR	4.7743
Bagging(ANN)	10.4565
Stacking(ANN)	10.4129
Proposed Model	10.8845

수를 수직적으로(직렬로) 나열하여 배치하는 것이 아니라(1개의 입력 변수 : 1개의 출력 변수), 이를 병렬적으로 나열하여(5개의 입력 변수 : 1개의 출력 변수) 배치하였다는 점에서 과적합이 방지된 것으로 보인다. 또한, 스택킹 앙상블 기법은 사용되는 모델의 개수와 성능에 따라 메타 모델의 성능이 개선될 수 있는데, 제안된 방법에 과적합을 방지할 수 있는 앙상블 배깅 기법을 스택킹 기법에 추가 삽입함으로써 보다 향상된 정확도 결과를 가져온 것으로 판단된다.

5. 결론

본 연구는 배깅 및 스택킹 기법의 앙상블 기계학습법을 이용하여 HPC 압축강도 예측모델의 정확도를 개선시키고자 하였다. 이를 위하여 기존의 앙상블 방법인 배깅 기법과 스택킹 기법을 결합하여 새로운 앙상블 기법을 제시하였고, 제안된 방법론을 통해 단일 기계학습모델의 문제점을 해결하여 모델의 예측 성능을 높이고자 하였다. 구체적으로 1030개의 HPC 압축강도 실험 결과를 활용하였고, 학습 데이터와 검증 데이터로 분류하였다. 단일 기계학습법으로 NR, SVM, ANN, 및 GPR을 활용하였으며, 앙상블 기계학습법으로 배깅 기법과 스택킹 기법을 이용하였다. 구체적으로 본 연구에서 제안하는 방법의 유효성을 확인하기 위하여 동일한 학습 데이터를 기반으로 단일 기계학습법 모델 및 개별적인 앙상블 기계학습법 모

델을 생성하였다. 또한, 제안된 방법의 모델의 정확도를 여러 가지 모델의 정확도와 비교하여 본 연구 방법의 유효성을 검증하였다. 모델의 정확도는 대표적인 4가지 성능 지표 비교를 통하여 확인하였다.

결과적으로 단일 기계학습모델은 GPR 모델의 정확도가 가장 우수하였다. 개별적인 앙상블 기법인 배깅 기법 모델과 스택킹 기법 모델은 단일 기계학습모델에서 가장 우수한 GPR 모델과 유사한 정확도를 보여주었다. 본 연구에서 제안한 방법은 개별적인 앙상블 기법 모델과 가장 우수한 단일 모델인 GPR 모델보다 높은 정확도를 가지는 모델이다. 이러한 결과는 정량적인 성능 지표 비교 및 예측값과 실제값의 비교를 통하여 확인할 수 있었다. 향후에는 결과에 대한 보다 심층적인 분석 및 검토를 통하여 본 연구보다 향상된 기법이 제시될 수 있을 것으로 판단된다.

감사의 글

이 논문은 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원(No. RS-2022-00144328)을 받아 수행되었습니다.

References

- Ahmad, S., Alghamdi, S.A. (2014) A Statistical approach to Optimizing Concrete Mixture Design, *The Scientific World J.*, 2014.
- Alpaydin, E. (2020) *Introduction to Machine Learning*, MIT Press, p.683.
- Apostolopoulou, M., Armaghani, D.J., Bakolas, A., Douvika, M. G., Moropoulou, A., Asteris, P.G. (2019) Compressive Strength of Natural Hydraulic Lime Mortars using Soft Computing Techniques, *Procedia Struct. Integr.*, 17, pp.914~923.
- Asteris, P.G., Kolovos, K.G., Douvika, M.G., Roinos, K. (2016) Prediction of Self-Compacting Concrete Strength using Artificial Neural Networks, *Eur. J. Environ. & Civil Eng.*, 20(sup1), pp.s102~s122.
- Asteris, P.G., Mokos, V.G. (2020) Concrete Compressive Strength using Artificial Neural Networks, *Neural Comput. & Appl.*, 32(15), pp.11807~11826.
- Asteris, P.G., Skentou, A.D., Bardhan, A., Samui, P., Pilakoutas, K. (2021) Predicting Concrete Compressive Strength using Hybrid Ensembling of Surrogate Machine Learning Models, *Cement & Concr. Res.*, 145, 106449.
- Atici, U. (2011) Prediction of the Strength of Mineral Admixture Concrete using Multivariable Regression Analysis and an Artificial Neural Network, *Expert Syst. with Appl.*, 38(8), pp.9609~9618.
- Breiman, L. (1996) Bagging Predictors, *Mach. Learn.*, 24(2), pp.123~140.
- Burges, C.J. (1998) A Tutorial on Support Vector Machines for Pattern Recognition, *Data Min. & Knowl. Discov.*, 2(2), pp. 121~167.
- Cheng, M.Y., Chou, J.S., Roy, A.F., Wu, Y.W. (2012) High-Performance Concrete Compressive Strength Prediction using Time-Weighted Evolutionary Fuzzy Support Vector Machines Inference Model, *Automat. Constr.*, 28, pp.106~115.
- Chou, J.S., Chiu, C.K., Farfoura, M., Al-Taharwa, I. (2011) Optimizing the Prediction Accuracy of Concrete Compressive Strength based on a Comparison of Data-Mining Techniques, *J. Comput. Civil Eng.*, 25(3), pp.242~253.
- Gartner, E. (2004) Industrially Interesting Approaches to “low-CO₂” Cements, *Cem. & Concr. Res.*, 34(9), pp.1489~1498.
- Jerath, S. (1983) Computer-aided Concrete Mix Proportioning, *J. Proc.*, 80(4), pp.312~317.
- Juenger, M.C.G., Winnefeld, F., Provis, J.L., Ideker, J.H. (2011) Advances in Alternative Cementitious Binders, *Cem. & Concr. Res.*, 41(12), pp.1232~1243.
- Kasperkiewicz, J., Racz, J., Dubrawski, A. (1995) HPC Strength Prediction using Artificial Neural Network, *J. Comput. Civil Eng.*, 9(4), pp.279~284.
- Kwag, S., Gupta, A., Dinh, N. (2018) Probabilistic Risk Assessment based Model Validation Method using Bayesian Network, *Reliab. Eng. & Syst. Safety*, 169, pp.380~393.
- Lam, L., Wong, Y.L., Poon, C.S. (1998) Effect of Fly Ash and Silica Fume on Compressive and Fracture behaviors of Concrete, *Cem. & Concr. Res.*, 28(2), pp.271~283.
- McClelland, J.L., Rumelhart, D.E., Hinton, G.E. (1986) *The Appeal of Parallel Distributed Processing*, MIT Press, Cambridge MA, 3, 44.
- Neshat, M., Adeli, A., Sepidnam, G., Sargolzaei, M. (2012) Prediction of Concrete Mix Design using Adaptive Neural Fuzzy Inference Systems and Fuzzy Inference Systems, *Int. J. Adv. Manuf. Technol.*, 63(1), pp.373~390.
- Ozbay, E., Gesoglu, M., Guneyisi, E. (2011) Transport Properties Based Multi-Objective Mix Proportioning Optimization of High Performance Concretes, *Mater. & Struct.*, 44(1), pp. 139~154.
- Öztaş, A., Pala, M., Özbay, E., Kanca, E., Çağlar, N., Bhatti, M. A. (2006) Predicting the Compressive Strength and Slump of High Strength Concrete using Neural Network, *Constr. & Build. Mater.*, 20(9), pp.769~775.

- Rasmussen, C.E.** (2003) Summer School on Machine Learning, *Gaussian Processes in Machine Learning*, Springer, Berlin, Heidelberg, pp.63~71.
- Rutkowska, G., Wichowski, P., Franus, M., Mendryk, M., Fronczyk, J.** (2020) Modification of Ordinary Concrete using Fly Ash from Combustion of Municipal Sewage Sludge, *Mater.*, 13(2), 487.
- Sun, L., Koopialipoor, M., Jahed Armaghani, D., Tarinejad, R., Tahir, M.M.** (2021) Applying a Meta-Heuristic Algorithm to Predict and Optimize Compressive Strength of Concrete Samples, *Eng. Comput.*, 37(2), pp.1133~1145.
- Syarif, I., Zaluska, E., Prugel-Bennett, A., Wills, G.** (2012) Application of Bagging, Boosting and Stacking to Intrusion Detection, *In International Workshop on Machine Learning and Data Mining in Pattern Recognition*, Springer, Berlin, Heidelberg, pp.593~602.
- Yeh, I.C.** (1998) Modeling of Strength of High-Performance Concrete using Artificial Neural Networks, *Cem. & Concr. Res.*, 28(12), pp.1797~1808.
- Yilmaz, I., Erik, N.Y., Kaynar, O.** (2010) Different Types of Learning Algorithms of Artificial Neural Network (ANN) Models for Prediction of Gross Calorific Value (GCV) of Coals, *Sci. Res. & Essays*, 5(16), pp.2242~2249.
- Zain, F.M., Abd, M.S.** (2009) Multiple Regression Model for Compressive Strength Prediction of High Performance Concrete, *J. Appl. Sci.*, 9(1), pp.155~160.
- Zhang, J., Huang, Y., Wang, Y., Ma, G.** (2020) Multi-Objective Optimization of Concrete Mixture Proportions using Machine Learning and Metaheuristic Algorithms, *Constr. & Build. Mater.*, 253, 119208.

요 지

고성능 콘크리트(HPC) 압축강도는 추가적인 시멘트질 재료의 사용으로 인해 예측하기 어렵고, 개선된 예측 모델의 개발이 필수적이다. 따라서, 본 연구의 목적은 배깅과 스택킹을 결합한 앙상블 기법을 사용하여 HPC 압축강도 예측 모델을 개발하는 것이다. 이 논문의 핵심적 기여는 기존 앙상블 기법인 배깅과 스택킹을 통합하여 새로운 앙상블 기법을 제시하고, 단일 기계학습 모델의 문제점을 해결하여 모델 예측 성능을 높이고자 한다. 단일 기계학습법으로 비선형 회귀분석, 서포트 벡터 머신, 인공신경망, 가우시안 프로세스 회귀를 사용하고, 앙상블 기법으로 배깅, 스택킹을 이용하였다. 결과적으로 본 연구에서 제안된 모델이 단일 기계학습 모델, 배깅 및 스택킹 모델보다 높은 정확도를 보였다. 이는 대표적인 4가지 성능 지표 비교를 통해 확인하였고, 제안된 방법의 유효성을 검증하였다.

핵심용어 : 고성능 콘크리트, 압축강도, 기계학습법, 앙상블 기법, 배깅 기법, 스택킹 기법