한국인터넷방송통신학회 논문지
제22권 제5호

ISSN : 2289-0238(Print) 2289-0246(Online)

오토인코더 기반 수치형 학습데이터의 자동 증강 기법

정주은, 김한준, 전종훈

To cite this article : 정주은, 김한준, 전종훈 (2022) 오토인코더 기반 수치형 학습데이터의 자동 증강 기법, 한국인터넷방송통신학회 논문지, 22:5, 75-86

The Journal of The Institute of Internet, Broadcasting and Communication (IIBC)
Vol. 22, No. 5, pp.75-86, Oct. 31, 2022. pISSN 2289-0238, eISSN 2289-0246

https://doi.org/10.7236/JIIBC.2022.22.5.75
JIIBC 2022-5-12

오토인코더 기반 수치형 학습데이터의 자동 증강 기법

Automatic Augmentation Technique of an Autoencoder-based Numerical Training Data

정주은[*], 김한준[**], 전종훈[***]

Ju-Eun Jeong[*], Han-Joon Kim[**], Jong-Hoon Chun[***]

요 약 본 연구는 딥러닝 기반 변분 오토인코더(Variational Autoencoder)를 활용하여 수치형 학습데이터 내 클래스 불균형 문제를 해결하고, 학습데이터를 증강하여 학습모델의 성능을 향상시키고자 한다. 우리는 주어진 테이블 데이터에 대하여 인위적으로 레코드 개수를 늘리기 위해 'D-VAE'을 제안한다. 제안 기법은 최적의 데이터 증강을 지원하기 위해 우선 이산화와 특징선택을 수반한 전처리 과정을 수행한다. 이산화 과정에서 k-means 클러스터링을 적용하여 그룹화한 후, 주어진 데이터가 원-핫 인코딩(one-hot encoding) 기법으로 원-핫 벡터(one-hot vector)로 변환한다. 이후, 특징선택 기법 중 RFECV 기법을 활용하여 예측에 도움이 되는 변수를 가려내고, 이에 대해서만 변분 오토인코더를 활용하여 새로운 학습데이터를 생성한다. 제안 기법의 성능을 검증하기 위해 4가지 유형의 실험 데이터를 활용하여 데이터 증강 비율별로 그 유효성을 입증한다.

Abstract This study aims to solve the problem of class imbalance in numerical data by using a deep learning-based Variational AutoEncoder and to improve the performance of the learning model by augmenting the learning data. We propose 'D-VAE' to artificially increase the number of records for a given table data. The main features of the proposed technique go through discretization and feature selection in the preprocessing process to optimize the data. In the discretization process, K-means are applied and grouped, and then converted into one-hot vectors by one-hot encoding technique. Subsequently, for memory efficiency, sample data are generated with Variational AutoEncoder using only features that help predict with RFECV among feature selection techniques. To verify the performance of the proposed model, we demonstrate its validity by conducting experiments by data augmentation ratio.

Key Words : Autoencoder, CTGAN, Data Augmentation, Deep learning, Table Data, Training Data, VAE

[*]준회원, 서울시립대학교 전자전기컴퓨터공학과
[**]정회원, 서울시립대학교 전자전기컴퓨터공학부 교수(교신저자)
[***]정회원, 명지대학교 융합소프트웨어학부 교수
접수일자 2022년 9월 4일, 수정완료 2022년 9월 30일
게재확정일자 2022년 10월 7일

Received: 4 September, 2022 / Revised: 30 September, 2022 / Accepted: 7 October, 2022
[*]Corresponding Author: khj@uos.ac.kr
Dept. of Electrical and Computer Engineering, University of Seoul, Korea

Ⅰ. 서 론

빅데이터 시대를 맞이하여, 우리는 방대한 데이터를 AI 학습데이터로 활용하려 노력하지만, 적정 수준의 학습모델을 생성하지 못하는 저품질 데이터를 자주 경험하게 된다. 특히, 대부분의 저품질 데이터는 클래스(레이블) 간의 불균형 문제를 가지는 사례가 많다. 불균형 데이터로 학습을 수행하는 경우, 소수 클래스는 무시되고 다수 클래스에 편향되는데, 이는 머신러닝 기반 학습모델의 성능을 저하시키는 주요 요인이 된다.

본 논문은 그림 1과 같이 AI 학습데이터로 활용될 수 있는 저품질 데이터의 클래스 불균형 문제점을 해결하기 위해서, 수치형 테이블 데이터에 대한 자동 증강(Data Augmentation) 기법을 제안한다. 충분하지 않은 데이터로 학습을 진행하면, 과소 적합(Underfitting)이나 과적합(Overfitting)이 발생할 위험이 높은데.[1], 편향성이 낮은 다량의 학습데이터를 확보하려면 시간과 비용이 들지만, 자동 증강을 통해 이 문제를 극복할 수 있다. 증강 기법은 주로 비정형 데이터인 이미지에서 연구가 활발하다. 적용 기법으로는 회전(Rotation), 크기(Scale), 반전(Flip), 외곽선 감지(contour detection) 로, 이를 활용하여 새로운 학습데이터를 만들어낸다[2].

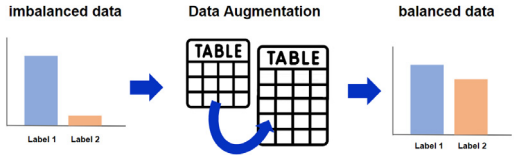


그림 1. 데이터 증강 프로세스
Fig. 1. Data Augmentation process

본 논문의 구성은 다음과 같다. 2절은 관련 연구로 제안 모델과 관련된 데이터 이산화, 클러스터링, 특징선택, 샘플링 기법, 생성 모델을 설명한다. 3절은 수치형 변수를 최적화하여, 변분 오토인코더로 샘플 데이터를 생성하는 'D-VAE'를 소개한다. 이를 통해 클래스 불균형 문제를 해결하여 고품질 학습데이터를 생성하는 방안을 서술한다. 4절은 수치형 데이터셋에 대한 실험을 통해 제안 기법의 효능을 보인다. 5절은 본 연구 내용을 요약하며 향후 연구 목표를 제시한다.

Ⅱ. 관련 연구

관련 연구에서는 제안 기법과 관련된 기초 이론을 정리한다. 수치형 정형 데이터를 최적화하는데 활용되는, 1. 데이터 이산화, 2. 클러스터링, 3. 특징선택 기법을 설명한다. 이를 이용한 전처리 과정을 거친 후, 학습데이터를 증강시키기 위해 사용 가능한 다양한 방법을 사용한다. 이는 데이터를 샘플링하는 4. 샘플링 기법과 학습데이터를 생성하는 5. 생성 모델로 분류할 수 있다. 샘플링 기법은 클래스 불균형한 데이터에 비율을 균형적으로 맞춘다. 자주 거론되는 기법은 오버샘플링과 언더샘플링이다. 생성 모델은 학습데이터의 분포를 학습하여 유사한 샘플 데이터를 생성한다. 데이터 증강하기 위하여 사용하며, 분포를 근사하거나 추정하는 방법으로 분류할 수 있다. 대표적인 생성 모델은 GAN, CTGAN, 오토인코더 등을 포함한다.

1. 데이터 이산화(Data Discretization)

데이터 이산화(Data Discretization)[3]는 데이터 전처리 기법으로, 연속형 변수를 이산형(범주형) 변수로 변환함을 의미한다. 즉, 이는 연속형 변수의 데이터 간격을 그룹화하거나 독립된 범주명으로 변환한다. 이를 통해, 주어진 데이터를 단순화하면 학습 프로세스가 빨라지며, 모델 변수와 클래스(레이블)와의 연관성이 커짐에 따라 결과 모델의 성능이 개선된다[4][5]. 의사결정트리(decision tree)를 포함한 대부분의 회귀 및 분류 모델은 수치값을 이산값으로 변환한 경우 해당 모델이 더 나은 성능을 가진다.

이산화 방법은 크게 3개 기준으로 나눌 수 있다. 첫째 기준은 이산화 방법을 지도학습과 비지도 학습으로 나눈다[6]. 지도학습 기반 이산화 방법은 엔트로피(entropy), 카이제곱(chi-square) 등의 통계량으로 독립변수와 클래스(레이블)와의 상관관계를 활용하여 이산화를 수행한다. 이는 기본적으로 클래스 정보로 학습데이터를 구분해야 하므로 클래스가 없는 경우에는 활용할 수 없다. 이에 반해 비지도 학습 기반 이산화 방법은 클래스 정보 없이, 사용자가 직접 간격을 지정하게 된다[7][8]. 둘째, 단변수적 또는 다변수적 이산화 방법으로 나눈다. 단변수적 이산화 방법은 모든 변수를 독립적으로 이산화한다. 대표적으로 동일 간격(equal-width), 동일 빈도(equal-frequency) 이산화 방법이 있다. 하지만 이는 이산화 과정에서 변수 간 상호작용 패턴이 손실될 가능성이 있다. 다변수적 이산화 방법은 단변수적 이산화 방법의 단점을 보완한 것이며, 이는 변수 간 상관관계를 고

려하여 이산화를 수행한다. 셋째, 이산화 진행 방향에 따라 분할과 병합으로 나눈다[9]. 분할 이산화 방법은 다중 지점에서 분할점을 찾아 세분화하는 것이며, 병합 이산화 방법은 다중 지점의 분할점을 찾은 이후 특정 조건에 따라 해당 구간을 합쳐 나간다[10].

2. 클러스터링(Clustering)

클러스터링(clustering)은 유사한 데이터를 같은 그룹으로 묶고, 유사하지 않은 데이터는 다른 그룹으로 분리한다. 클러스터링 유형은 계층형 클러스터링(hierarchical clustering)과 비계층형 클러스터링(non-hierarchical clustering)으로 분류된다. 계층형 클러스터링은 단계적으로 유사도가 높은 중간 클러스터링 결과를 병합하는 과정을 거치며, 이는 클러스터의 개수를 미리 요구하지 않는다. 비계층형 클러스터링은 반복적으로 유사도 계산을 통해 군집 내 소속 데이터에 대한 최적화 과정을 수행한다. 비계층형 클러스터링의 대표적인 알고리즘은 k-modes, k-means[11], k-prototypes[12]이 있다. 클러스터링은 대표적인 비지도 학습에 해당하며, 그 특성상 이산화 과정에 유용하게 적용된다.

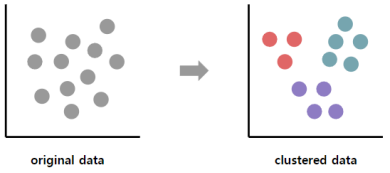


그림 2. 클러스터링 예시
Fig. 2. Clustering example

3. 특징선택(Feature Selection)

특징선택(feature selection)은 데이터 증강에 크게 기여할 수 있는 독립변수들에 대한 최적의 조합을 찾아내는 것이며, 이를 통해 독립변수 중에서 중복되거나 종속변수와 관련이 없는 변수들이 제거된다[13]. 특징선택은 결과적으로 학습데이터를 증강하고 품질을 개선하여 모델 성능을 높이는데 크게 기여하게 된다. 특징선택은 크게 랩퍼(wrapper), 필터(filter), 임베디드(embedded) 방식으로 구분하며, 본 연구는 필터 방식을 채택한다.

4. 샘플링 기법

오버샘플링(oversampling)은 높은 비율을 차지하는 샘플 개수에 맞춰 낮은 비율 클래스를 위한 샘플을 생성하고 보강하는 기법이다[14]. 이는 동일한 데이터를 반복학습하기 때문에 과적합을 유발할 가능성이 크다. 일반적으로 언더샘플링에 비해 오버샘플링이 예측 성능에 유리하며, 대표적인 기법은 SMOTE[15]다. 이는 가까운 이웃 사이의 차이를 계산하는 최근접 이웃 알고리즘을 활용하여 낮은 비율의 클래스 데이터를 생성한다.

언더샘플링(undersampling)은 적은 비율 클래스의 샘플 개수를 고려하여, 높은 비율을 차지하는 샘플들을 제거하는 방법이다[16]. 이는 주어진 데이터의 일부를 제거하기 때문에, 이로 인한 정보의 손실로 결과 모델의 성능을 낮출 위험이 있다.

5. 생성모델(Generative Model)

생성모델인 GAN (Generative Adversarial Networks)[17]은 주어진 데이터의 분포를 고려하여 유사한 샘플을 생성하는데 효과적인 기법으로 알려져 있다. 이는 주로 이미지와 같은 비정형 데이터에 활용되는데, 데이터 증강을 위한 샘플링 기법의 일반화에 기여할 수 있어 관련 연구가 활발하다.

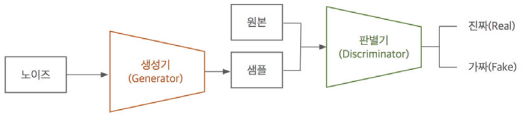


그림 3. GAN 아키텍처
Fig. 3. GAN Architecture

GAN은 데이터를 생성하는 생성기(generator)와 원본 데이터와 샘플 데이터를 구분하는 판별기(discriminator)가 경쟁하며 성능을 보완한다. 그림 3은 GAN 아키텍처이며, 진짜(Real) 같은 가짜(Fake) 데이터를 생성하는 것이 목적이다. GAN의 동작 과정은, 원본 데이터에 랜덤하게 잡음(noise)을 추가한 데이터를 생성기에 입력값으로 받는다. 생성기에서 생성된 데이터는 샘플 데이터이며, 원본 데이터와 샘플 데이터를 합쳐 판별기에 입력하여 학습한다. 판별기에 입력된 데이터가 진짜 데이터인지 가짜 데이터인지 판단하여 확률분포를 추정한다.

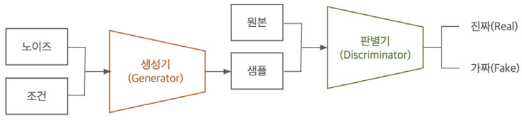


그림 4. CTGAN 아키텍처
Fig. 4. CTGAN Architecture

그림 4는 CTGAN(Conditional Tabular Generative Adversarial Network)[18] 아키텍처를 나타낸다. 이는 GAN에서 파생된 모델로, 이산형 변수와 수치형 변수를 포함하는 정형 테이블에 적용된다. GAN과 달리 CTGAN은 생성기 입력값에 조건(condition)을 추가한다. 조건은 원본 데이터의 빈도를 학습하여, 소수 클래스를 잊지 않고 생성하도록 하는 역할이다. 이는 클래스 불균형 문제를 가진 데이터에 대하여 과거에 제안된 Vanila GAN, WGAN, TGAN에 비해 재현율(Recall)과 균형회복(Balance) 성능이 우수하다[19]. 다만 CTGAN은 하이퍼파라미터인 레이어 개수와 배치 크기에 따라 성능이 민감하게 반응하는 단점을 가진다. 또한 CTGAN은 데이터 증강 시 여전히 소수 클래스보다 다수 클래스로 편향되는 문제를 안고 있다.

오토인코더(Autoencoder)[20]는 차원 축소를 지원하는 비지도 학습 기법이다. 입력 X와 출력 X'의 오차가 최소화되도록 학습하여 최대한 동일 값을 가지도록 하는 것이 목적이다.

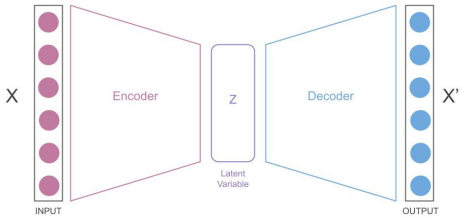


그림 5. 오토인코더 모델
Fig. 5. Autoencoder Model

그림 5의 오토인코더에서 인코더(encoder)는 입력값 X를 받아 저차원의 특징값으로 변환하며, 디코더(decoder)는 이 특징값을 받아 입력값과 유사한 출력값으로 변환하는 기능을 수행한다. 인코더와 디코더 사이에 있는 변수는 잠재변수(Latent Variable)라 일컬으며, 주어진 데이터에 대한 임베딩 벡터를 산출하는데 활용된다.

Ⅲ. 제안 기법

1. 자동 증강(Data Augmentation) 기법

자동 증강(Data Augmentation)의 목적은 학습데이터를 양을 늘려 클래스 불균형을 해결하고 학습 성능을 향상시키는 것이다. 또한, 이는 인간의 개입 없이 자동으로 데이터를 확장하여, 초기 주어진 충분하지 않은 데이터로부터 얻지 못한 패턴을 확보하는 데 도움을 준다. 데이터 자동 증강 기법은 충분한 학습데이터를 확보하는 시간과 노력을 줄여, 비즈니스 측면에서 경제적이다. 그러므로 고부가가치를 생산하는데 크게 기여할 수 있다. 자동 증강 기법에서는 원본 데이터의 맥락을 파악하여, 분석에 사용 가능한 고품질 데이터를 생성하는 것이다. 제안 기법에서는 수치형 정형 데이터의 값의 분포를 학습한 후 증강하여 성능을 향상시킨다. 증강한 데이터는 샘플 데이터라고 하며, 원본 데이터를 기반으로 양질의 샘플 데이터를 얻는다.

2. 변분 오토인코더(Variational Autoencoder)

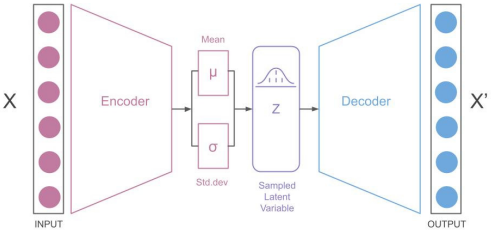


그림 6. 변분 오토인코더 모델
Fig. 6. Variational Autoencoder Model

변분 오토인코더(VAE, Variational Autoencoder)[21]는 오토인코더 기반의 생성 모델이다. 새로운 샘플 데이터를 학습데이터 분포와 유사하게 생성하는 것이 목적이다. 오토인코더와 외형은 동일하나, 잠재 공간(Latent Space)이 다른 특수한 분포를 띄게 한다. 잠재 공간은 데이터 특징으로, 일정한 분포를 나타낸다. 인코더는 정규분포를 따르는 Z의 분포를 표현하기 위해, Z의 평균과 분산을 추정한다. Z를 디코더에 투입하여 입력값을 복원하여 샘플 데이터를 생성한다.

자동 증강을 하기 위한 대표적인 생성 모델은 변분 오토인코더와 GAN이다. 두 모델은 차이점이 명확하다. 변분 오토인코더는 데이터 분포를 학습하기 위해 변분 추론(Variational Inference)한다. 변분추론은 사후확률분포를 다루기 쉬운 확률분포로 근사(approximation)하는 것이다. 그리고 분포가 잘 학습되면, 자동적으로 샘플링이 된다. GAN은 원본 데이터와 같은 샘플 데이터를 생성한다는 점에서 차이가 있다. 변분 오토인코더는 GAN에 비해 손실함수 평가 기준이 명확하기 때문에 학습이 안정적인 장점이 있다.

3. D-VAE 개요

'D-VAE'는 클래스 불균형을 해결하고 수치형 데이터를 증강시켜 고품질의 학습데이터를 얻는 것이 목적이다. 그림 7은 제안 기법의 수행과정을 보여준다.

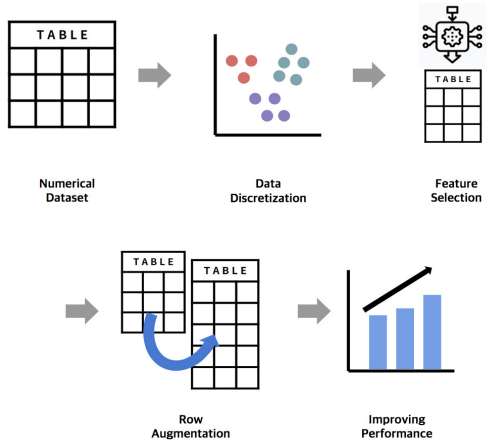


그림 7. D-VAE 프로세스
Fig. 7. D-VAE Process

제안 기법은 최적화 과정을 거친 후 데이터 증강한다. 데이터 전처리 시 최적화하는데, 클러스터링한 결과를 바탕으로 이산화와 특징선택을 거친다. 이산화 과정에서 최솟값과 최댓값을 구하여 균일한 너비로 지정한 구간만큼 그룹화를 진행한다. 이때 클러스터링의 대표적인 기법 중 하나인 단일변수 비지도 학습 k-means를 적용한다. 이는 중심점(centroid)를 기반으로, 평균 군집의 가장 가까운 중심을 찾아 그룹화한다. 이후 연속형 변수에서 이산화한 값을 변화하기 위해 원-핫 인코딩(one-hot encoding)으로 원-핫 벡터(one-hot vector)로 반환한다. 원-핫 벡터로 변환 시, 변수 간의 독립성을 부여해 해당 변수에만 1을 부여한다. 레이블 인코딩과 달리 원-핫 인코딩은 모델 학습 시 숫자의 대소 관계와 순서에 영향을 받지 않는 장점이 있어, 제안 기법에 원-핫 인코딩 기법을 적용한다. 이는 변수의 개수가 늘어날수록, 벡터를 저장하는데 필요한 공간이 계속 증가한다. 이러한 단점을 보완하기 위해 특징선택의 REFCV(Recursive Feature Elimination with Cross Validation)[22] 알고리즘을 사용한다. 이 알고리즘은 변수 중요도가 낮은 변수들을 제거하고, 각 변수 개수마다 교차 검증(Cross Validation)을 활용해 성능을 계산한다. 그리고 변수별 성능을 평균 내어 가장 높은 성능을 가지는 변수들을 최종 결과로 사용한다. 이는 RFC의 사용자가 변수를 직접 정의해야 하는 어려운 점을 보완한다. 두 단계의 최적화 과정을 거친 후, 변분 오토인코더로 학습데이터의 분포와 유사한 분포를 갖는 샘플 데이터를 생성한다. 사전 학습이 완료되면, 원-핫 벡터를 활용하여 심층신경망을 미세조정 한다. 이때 결과 예측을 위해 ReLU 함수를 사용한다, 손실함수로는 MSE(Mean Squared Error)를 사용하며, ReLU 함수로 계산한 값과 정답을 비교해 오차를 줄인다.

4. D-VAE 학습 과정

제안 기법은 다음의 4단계를 순차적으로 수행한다.

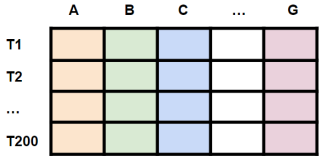


그림 8. D-VAE 프로세스 - 1단계
Fig. 8. D-VAE Process - Step 1

[1단계] 그림 8는 초기 입력되는 학습데이터이며, 학습 과정을 설명하기 위한 예시 데이터이다. 이는 A부터 G까지 총 7개의 변수와 200개의 레코드로 이루어진 수치형 정형 데이터셋이다.

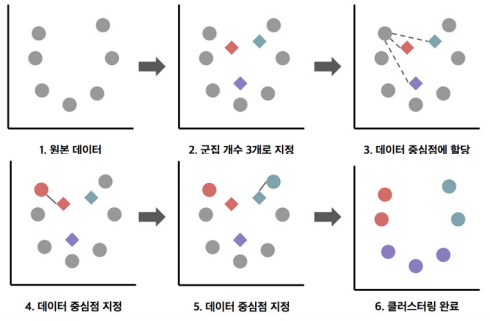


그림 9. k-means로 그룹화하는 과정
Fig. 9. The grouping process by K-means

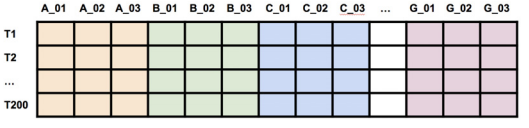


그림 10. D-VAE 프로세스 - 2단계
Fig. 10. D-VAE Process - Step 2

[2단계] 원본 데이터를 전처리 과정을 거쳐 최적화한다. 우선, 그림 9와 같이 클러스터링 과정을 거쳐 그룹화하고, 이산화하여 원-핫 인코딩으로 원-핫 벡터를 생성한다. 구간을 3개로 지정했다고 가정할 때, 그림 10와 같이 한 변수당 원-핫 벡터 변수가 3개씩 늘어난다.

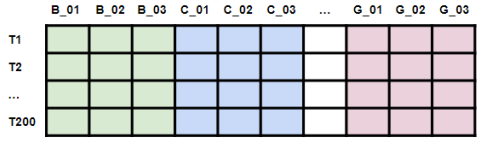


그림 11. D-VAE 프로세스 - 3단계
Fig. 11. D-VAE Process - Step 3

[3단계] 이산화한 데이터로 특징선택 과정을 거친다. REFCV 기법으로 레이블 예측 중요도가 낮은 변수를 제거한다. 레이블 예측 시 변수 'A'가 중요도가 낮다고 가정하면, 그림 11과 같이 변수 'A'가 제거된다.

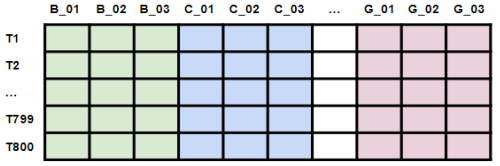


그림 12. D-VAE 프로세스 - 4단계
Fig. 12. D-VAE Process - Step 4

[4단계] 그림 12는 전처리 과정을 마친 데이터로, 변분 오코인코더를 활용하여 200개의 레코드에서 800개로 증강했다. 이처럼 D-VAE는 학습데이터의 레코드를 증강하여 양질의 데이터를 얻는다.

5. D-VAE 이산화 학습 예시

	ph	Hardness	Solids	Chloramines	Sulfate	Conductivity	Organic_carbon	Trihalomethanes	Turbidity	Potability
0	NaN	204.890455	20791.318981	7.300212	368.516441	564.308654	10.379783	86.990970	2.963135	0
1	3.716080	129.422921	18630.057858	6.635246	NaN	592.885359	15.180013	56.329076	4.500656	0
2	8.099124	224.236259	19909.541732	9.275884	NaN	418.606213	16.868637	66.420093	3.055934	0
3	8.316766	214.373394	22018.417441	8.059332	356.886136	363.266516	18.436524	100.341674	4.628771	0
4	9.092223	181.101509	17978.986339	6.546600	310.135738	398.410813	11.558279	31.997993	4.075075	0
...	...	...	...	...	...	...	...	...	...	...
3271	4.668102	193.681735	47580.991603	7.166639	359.948574	526.424171	13.894419	66.687695	4.435821	1
3272	7.808856	193.553212	17329.802160	8.061362	NaN	392.449580	19.903225	NaN	2.798243	1
3273	9.419510	175.762646	33155.578218	7.350233	NaN	432.044783	11.039070	69.845400	3.298875	1
3274	5.126763	230.603758	11983.869376	6.303357	NaN	402.883113	11.168946	77.488213	4.708658	1
3275	7.874671	195.102299	17404.177061	7.509306	NaN	327.459760	16.140368	78.698446	2.309149	1

3276 rows × 10 columns

그림 13. 이산화 전 테이블(변수 : 10개)
Fig. 13. Table before discretization(Features : 10)

제안 기법은 이산화 기법을 활용한다. 변수의 최솟값과 최댓값 사이에 있는 범위를 일정하게 지정한 구간만큼 나눈다. 그림 13은 실험에 사용된 수질 측정 데이터로서, 이는 독립변수 9개, 레이블 1개로 구성된다. 다음은 제안 기법의 이산화와 원-핫 인코딩 과정이 수행된 과정을 소개한다.

	ph_0	ph_1	ph_2	ph_3	ph_4	Hardness_0	Hardness_1	Hardness_2	Hardness_3	Hardness_4	...	Trihalomethanes_0	Trihalomethanes_1	Trihalomethanes_2	Trihalomethanes_3	Trihalomethanes_4
0	1	0	0	0	0	0	0	1	0	0	...	0	0	0	0	1
1	0	1	0	0	0	1	0	0	0	0	...	0	0	1	0	0
2	0	0	0	1	0	0	0	0	1	0	...	0	0	1	0	0
3	0	0	0	1	0	0	0	0	1	0	...	0	0	0	0	1
4	0	0	0	0	1	0	0	1	0	0	...	0	1	0	0	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
3271	0	1	0	0	0	0	0	1	0	0	...	0	0	1	0	0
3272	0	0	0	1	0	0	0	1	0	0	...	1	0	0	0	0
3273	0	0	0	0	1	0	1	0	0	0	...	0	0	0	1	0
3274	0	1	0	0	0	0	0	0	1	0	...	0	0	0	1	0
3275	0	0	0	1	0	0	0	1	0	0	...	0	0	0	1	0

3276 rows × 45 columns

그림 14. 이산화 후 테이블(변수 : 46개)
Fig. 14. Table after discretization(Features : 46)

그림 14는 원본 데이터에서 이산화를 수행한 결과다. 실험에서는 구간을 5개로 설정했기 때문에 독립변수가 9개에서 45개로 늘어나, 총 46개의 변수로 증가한다.

하단에는 사용자가 지정한 구간의 범위를 지정하는 예시이다. 표 1은 수질 측정 데이터의 'Hardness' 값 범위를 나타낸다. 변수의 최솟값은 40, 최댓값은 320이다. 원-핫 인코딩으로 변환하려면 반드시 이산화 과정을 거쳐야 한다. 이산화 과정으로는 입력값의 최솟값, 최댓값 기준으로 균일한 너비로 구간(bin)을 분할한다. 구간 분할로 형성된 경곗값을 기준으로 데이터 포인트를 나눈다. 원-핫 인코딩된 데이터를 표현하기 위해 트리 모델을 사용하였는데, 데이터셋에서 예측을 위한 가장 적절한 구간을 학습하는 장점이 있기 때문이다.

표 1. 'Hardness' 변수의 값 범위
Table 1. 'Hardness' feature value range

구분	값
최솟값	40
최댓값	320

구간을 5개로 설정하면, 표 2와 같이 그룹이 5개로 나뉜다. 그룹 1은 40-96, 그룹 2는 96-152, 그룹 3은 152-208, 그룹 4는 208-264, 그룹 5는 264-320이다.

표 2. 'Hardness' 변수의 구간별 범위(구간 5개)
Table 2. Range by group of 'Hardness' feature(5 groups)

그룹	범위
1	40-96
2	96-152
3	152-208
4	208-264
5	264-320

표 3은 원-핫 인코딩 전 원본 데이터의 ID와 값이다. 총 3,276의 레코드에는 최솟값과 최댓값 사이의 값이 분포되어 있다. 실수일 경우, 정수로 변환하여 원-핫 벡터로 변환한다.

표 3. 'Hardness' 변수의 값(원-핫 인코딩 전)
Table 3. Value of 'Hardness' feature(before one-hot encoding)

그룹	범위
1	253
2	117
3	50
4	320
5	40
...	...
3276	66

표 4는 표 3에서 구간이 5개라고 설정할 때 원-핫 인코딩이 수행된 결과다. 변수값을 기준으로 해당 범위의 그룹에는 1, 나머지 그룹에는 0이 부여된 것을 확인할 수 있다.

표 4. 'Hardness' 변수의 값(원-핫 인코딩 후)
Table 4. Value of 'Hardness' feature(after one-hot encoding)

ID	H_1	H_2	H_3	H_4	H_5
1	0	0	0	1	0
2	0	1	0	0	0
3	1	0	0	0	0
4	0	0	0	0	1
5	1	0	0	0	0
...					
3276	1	0	0	0	0

그림 15는 Hardness 변수가 실제로 그룹화된 실험 결과를 보여준다.

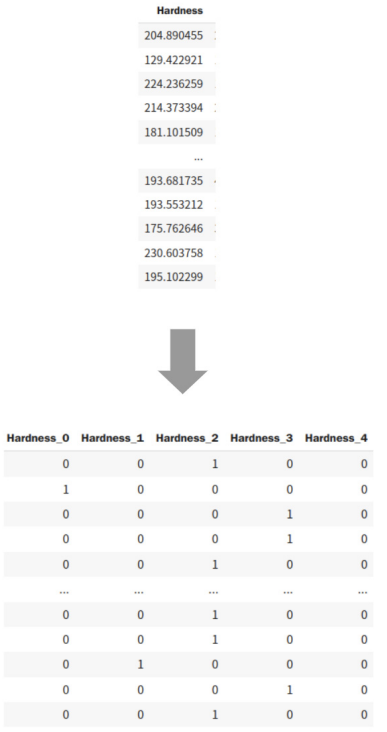


그림 15. 이산화 후 'Hardness' 변수의 변화
Fig. 15. Changes in the 'Hardness' feature after discretization

Ⅳ. 실험 및 결과

1. 실험 개요 및 데이터

우리는 기존 기법과 비교하여 제안 기법인 'D-VAE'의 우수성을 검증하기 위해. 실험 데이터로서 아래의 4개 수치형 데이터셋을 선정하였다.

- 유방암(WDBC) 데이터: 미국 Wisconsin 대학병원에서 1995년에 공개한 데이터로서, 총 569개의 레코드로 구성된다. 가는 주사바늘을 사용하여 낭종의 세포나 조직을 떼어내서 조직을 검사한 이미지를 디지털로 변환하여 측정한 30개의 수치형 변수와 클래스 변수인 'diagnosis'는 결과가 악성이면 1, 양성이면 0으로 표시된다.
- 당뇨병(Diabetes) 데이터: 피마인디언(Pima Indian) 당뇨병 데이터로서, 총 768개의 레코드로 구성된다. 나이, 임신 횟수, 포도당 부하 검사

수치, 혈압 등 신체 특성을 수치형 변수로 나타낸다. 전체 데이터 중 당뇨가 아닌 비율이 65.1%다. 클래스 변수인 'Outcome'은 당뇨일 경우 1, 그렇지 않은 경우 0으로 표시된다.

- 수질 측정(Water Quality) 데이터: 안전한 식수인지 판별하기 위한 수질 측정 데이터로서, 총 3,276개의 레코드로 구성된다. pH값, 클로라민, 전도도, 탁도 등 물의 오염도를 수치형 변수로 나타낸다. 클래스 변수인 'Potability'는 사람이 음용하기에 안전한지 나타낸다. 음용할 수 있으면 1, 그렇지 않으면 0으로 표시된다.
- 신용카드(CreditCard) 데이터: 2013년 9월에 유럽 카드 소지자의 신용카드 거래한 데이터로서, 총 284,807개의 레코드로 구성된다. 전체 데이터 중 사기(fraud)가 발생한 비율은 0.172%이며 클래스 분포가 매우 불균형하다. 본 실험에 사용한 데이터는 PCA 변환 결과인 수치형 변수로 이루어져 있으며, 클래스 변수인 'Class'는 사기인 경우 1, 그렇지 않으면 0으로 표시된다.

표 5. 실험에 사용한 데이터셋
Table 5. Dataset used in the experiment

데이터셋	변수 개수	레코드 개수
유방암	32	569
당뇨병	9	768
수질 측정	10	3,276
신용카드	31	284,807

실험 진행 시 부여한 조건으로는, 주어진 데이터셋에 존재하는 결측치는 평균대치법을 적용하여 보완하였고, 레코드의 고유번호는 학습모델에 도움이 되지 않아 삭제한다. 그리고 우리는 실제 이산화 과정을 구현하기 위해서 파이썬 패키지 중 구간 분할을 지원하는 KBinsDiscretizer 클래스를 사용하였다.

2. 실험 결과

제안 기법은 데이터 증강 전 최적화 과정을 거친다. 이후, REFCV 기법으로 레이블 예측에 도움이 되는 변수를 선정한다. 최적화 과정을 거친 후 변분 오토인코더로 데이터 증강한다. 데이터 증강 비율은 표 6에서 보는 바와 같이 5개 비율 (20%, 40%, 60%, 80%, 100%)로 구분하여 성능분석을 진행하였다.

표 6. 데이터 증강 비율
Table 6. Data Augmentation Ratio

데이터셋	증강 비율	레코드 개수
유방암	원본	569
	+20%	683
	+40%	797
	+60%	910
	+80%	1,024
	+100%	1,138

데이터셋	증강 비율	레코드 개수
당뇨병	원본	768
	+20%	922
	+40%	1,075
	+60%	1,229
	+80%	1,382
	+100%	1,536

데이터셋	증강 비율	레코드 개수
수질 측정	원본	3,276
	+20%	3,931
	+40%	4,586
	+60%	5,242
	+80%	5,897
	+100%	6,552

데이터셋	증강 비율	레코드 개수
신용카드	원본	284,807
	+20%	341,768
	+40%	398,730
	+60%	455,691
	+80%	512,653
	+100%	569,614

우리가 사용한 성능 평가 척도는 정확도와 F1-score이며, 예측모델 구성을 위해 Logistic Regression, XGBoost, Random Forest 알고리즘을 사용하였다. 표 7은 4개의 수치형 데이터셋에 원본 데이터와 100% 증강한 데이터의 성능을 비교한다. 클래스 불균형 해결 시 자주 사용되는 샘플링 기법인 CTGAN과 변분 오토인코더, 제안 기법 별 성능을 상호 비교한다. 그래프의 초록색 막대는 정확도, 파란색 막대는 F1-Score이다. 막대그래프의 X축은 예측모델이며, Y축은 성능을 나타낸다.

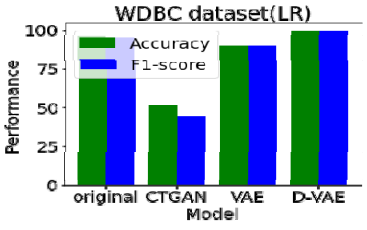
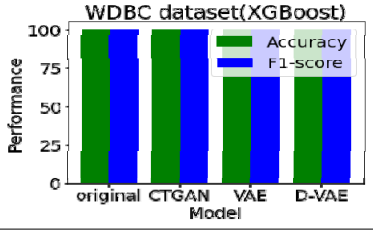
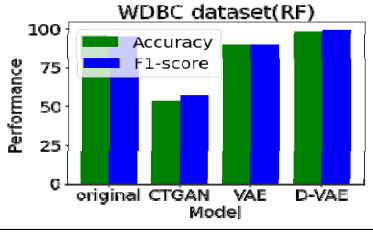
실험 결과로는, CTGAN에 비해 오토인코더 기반 모델의 성능이 평균 30~40% 정도 뛰어나다. 변분 오코인코더와 제안 모델의 성능은 비슷하나, 제안 모델이 기존 변분 오코인코더에 비해 레코드 수가 작은 소규모 데이터셋에서 평균 1~2% 정도 성능이 우수하다. 실험을 진행한 신용카드, 당뇨병, 유방암 데이터셋은 제안 모델에서 성능이 가장 우세하다. 그러나 수질 측정 데이터셋은 원본 데이터에서 +80% 증강했을 때는, 변분 오토인코더와

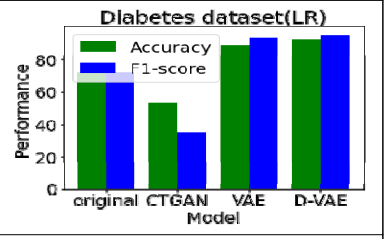
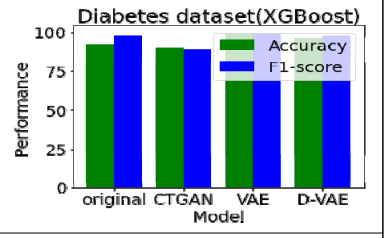
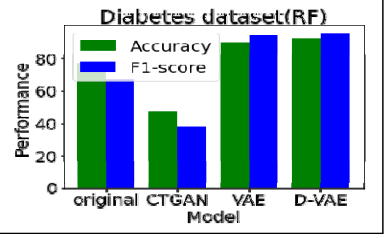
성능이 동일하다. 그러나 +100% 증강했을 때, Logistic Regression 2%, XGBoost 6%, Random Forest 9% 정도 낮은 결과를 보인다.

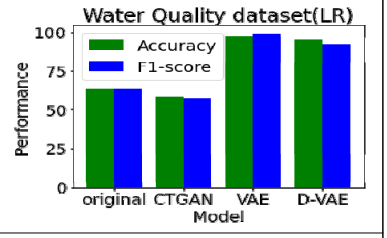
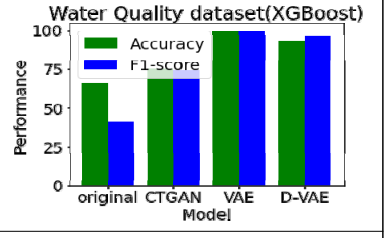
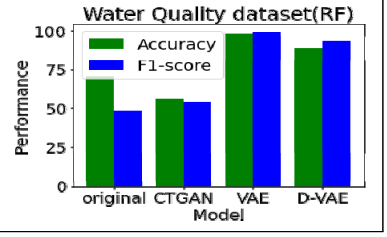
제안 기법의 목적 중 하나는 소수 클래스를 증폭시켜 불균형 데이터를 해결하는 것이다. 표 8은 원본 데이터와 +100% 증강한 샘플 데이터의 막대그래프와 차원축소 중 PCA 기법으로 클래스 분포를 비교하였다. 실험한 4가지 데이터셋 모두 레이블 변수의 클래스는 2가지로, 각각 빨간색과 파란색으로 나타냈다. 데이터를 차원 축소하여 원본 데이터와 100% 증강한 샘플 데이터의 클래스 분포를 시각화하였다. 원본 데이터에서는 하나의 클래스에 집중되어 클래스가 불균형하다. 하지만 데이터 증강 후 클래스가 고르게 분포되어 불균형 현상을 해결했다.

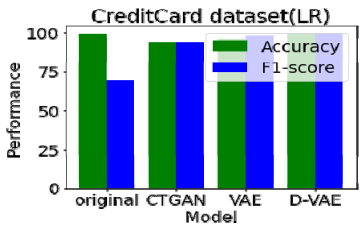
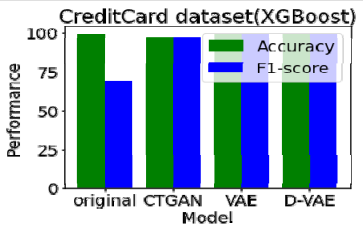
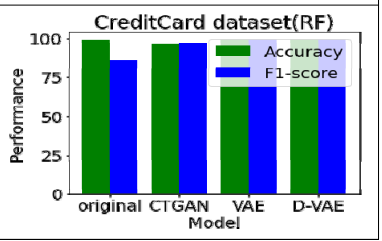
표 7. 데이터셋의 원본 데이터와 샘플 데이터(+100%)의 모델별 성능 비교

Table 7. Comparison of model-specific performance of the original data and sample data(+100%) of the dataset

유방암 데이터셋	
Number of Records : 569 → 1,138	
Logistic Regression	
XGBoost	
Random Forest	

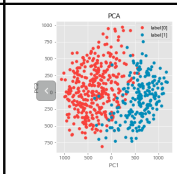
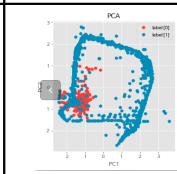
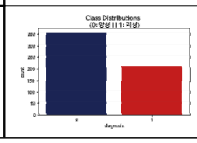
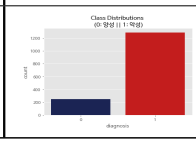
당뇨병 데이터셋	
Number of Records : 768 → 1,536	
Logistic Regression	
XGBoost	
Random Forest	

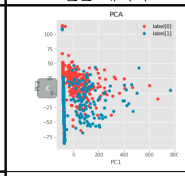
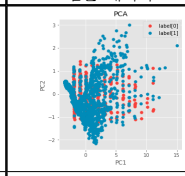
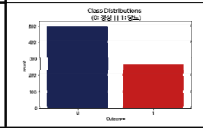
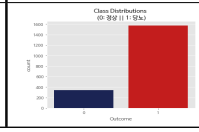
수질 측정 데이터셋	
Number of Records : 3,276 → 6,552	
Logistic Regression	
XGBoost	
Random Forest	

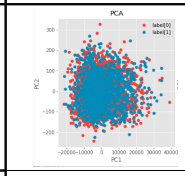
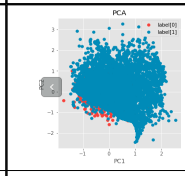
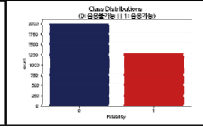
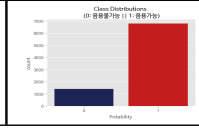
신용카드 데이터셋	
Number of Records : 284,807 → 569,614	
Logistic Regression	
XGBoost	
Random Forest	

제안 기법은 메모리 효율성을 위해 특징선택을 거쳐, 레이블 예측에 도움이 되는 변수만 학습에 활용한다. 성능을 측정하기 위해 특징선택을 수행 전과 수행 후 소요된 시간을 비교한다. 실험 시 데이터에 부여한 조건은, 이산화 시 원-핫 벡터로 변환하기 위해 구간을 5개로 설정하여 변수당 5개씩 변수가 증가한다. 단, 변수의 중복된 값이 30개 이상인 경우, 원-핫 벡터로 변환했다.

표 8. 데이터셋의 클래스 변수 분포
Table 8. Dataset class feature distribution

유방암 데이터셋		
종류	원본 데이터	샘플 데이터
PCA		
막대그래프		

당뇨병 데이터셋		
종류	원본 데이터	샘플 데이터
PCA		
막대그래프		

수질 측정 데이터셋		
종류	원본 데이터	샘플 데이터
PCA		
막대그래프		

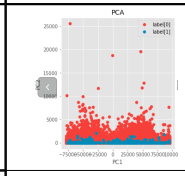
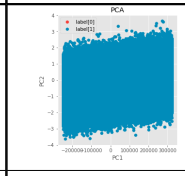
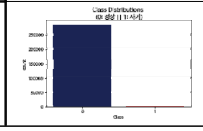
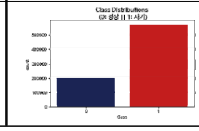
신용카드 데이터셋		
종류	원본 데이터	샘플 데이터
PCA		
막대그래프		

표 9는 특징선택 전/후 소요된 시간과 변수 개수를 비교한 실험 결과이다. 특징선택 중 REFCV 기법을 활용하여 레이블 예측에 기여하지 않는 변수는 제거하였다.

표 9. 특징선택 전/후 소요시간 비교(단위 : 초)
Table 9. Comparision of timed required before and after feature selection (unit : seconds)

데이터셋	전(변수 개수)	후(변수 개수)
유방암	13.2903 (151)	7.4577 (64)
당뇨병	12.6607 (37)	12.1015 (32)
수질 측정	23.3419 (46)	16.3832 (44)
신용카드	18.4951 (151)	11.0394 (102)

실험 결과로, 유방암 데이터는 5.8326초, 당뇨병 데이터는 0.5592초, 수질 측정 데이터는 6.9587초, 신용카드 데이터는 7.4557초 감소했다. 특징선택으로 변수 최적화 과정을 거치니, 모델 학습 시간이 단축되었다.

V. 결 론

본 논문은 학습데이터로 활용되는 수치형 정형 테이블을 증강하는 기법을 제안한다. 수치형 변수를 최적화하여 변분 오토인코더로 샘플링하는 'D-VAE'를 비교 분석하였다. 제안 기법을 통해 주어진 학습데이터 내 클래스 불균형 문제를 해결하였으며, 수치형 데이터를 활용한 실험을 통해 제안 기법의 효능을 확인하였다. 제안 기법의 성능이 CTGAN과 기존 변분 오토인코더보다 뛰어남을 실험을 통해 증명했다.

데이터 증강 비율을 고려했을 때, 제안 기법이 변분 오토인코더에 비해 적은 양의 레코드 수를 증강해도 성능이 향상했다. 더불어, 원-핫 인코딩의 단점을 보완하기 위해 특징선택 과정을 거쳐 모델 학습 시간을 줄이고, 성능을 보존한다. 향후 연구로는 수치형 데이터와 범주형 데이터가 혼합된 '혼합형' 테이블 데이터의 증강을 위해, 다변수적 방법을 활용하여 연구를 진행할 예정이다.

References

[1] Douglas M. Hawkins, "The problem of overfitting", Journal of chemical information and computer sciences, Vol. 44, No. 1, pp. 1-12, Dec 2004.
DOI: https://doi.org/10.1021/ci0342472

[2] Yongsoo Kwon, Seungyeon Hwang, Dongjin Shin, Jeongjoon Kim, "A Study on Application Method of Contour Image Learning to improve the Accuracy of CNN by Data", IIBC, Vol. 22, No. 4, pp.171-176, Aug 2022.
DOI: https://doi.org/10.7236/JIIBC.2022.22.4.171

[3] Ian H. Witten, Eibe Frank, Mark A. Hall, "Data mining: Practical machine learning tools and techniques[3rd edition]", Morgran Kaufmann, Jan 2011.
DOI: 10.1145/2020976.2021004

[4] James Dougherty, Bon Kohavi, Mehran Sahami, "Supervised and unsupervised discretization of continuous features", Proceedings of the 21th International Conference on Machine Learning 1995, pp. 194-202, Jun 1995.
DOI: https://doi.org/10.1016/B978-1-55860-377-6.50032-3

[5] Kim SeokKyoun, Lee Kwanwoo, "Application Method of Regular Expressions and Suffixes to improve the Accuracy of Automatic Domain Identification of Public Data", IIBC, Vol. 22, No. 4, pp. 81-86, Aug 2022.
DOI: https://doi.org/10.7236/JIIBC.2022.22.4.81

[6] Huan Liu, Farhad Hussain, Chew L. Tan, Manoranjan Dash, "Discretization: An enabling technique", Data Mining and knowledge Discovery Vol. 6, No. 4, pp. 393-423, Jul 2002.
DOI: https://doi.org/10.1023/A:1016304305535.

[7] Sotiris Kotsiantis, Dimitris Kanellopoulos, "Discretization techniques: A recent survey" GESTS International Transactions on Computer Science and Engineering, Vol. 32, No. 1, pp. 47-58, 2006.

[8] Kavita Das, Om P. Vyas, "A suitability study of discretization methods for associative classifiers", International Journal of Computer Applications, Vol. 5, No. 10, pp. 46-51, Aug 2010.
DOI:10.5120/944-1322

[9] Benjamin Johnston, Aaron Jones, "Applied Unsupervised Learning with Python", Packt Publishing, 2019.

[10] Istvan Jonyer, Lawrence B. Holder, Diane J. Cook, "Graph-Based Hierarchical Conceptual Clustering in Structural Databases", Proceedings of AAAI, 2000.

[11] Jackie A. Hartigan, Ma L. Wong, "Algorithm As 136:A K-Means Clustering Algorithm", Journal of the Royal Statistical Society. Series C, Vol. 28, No. 1, pp. 100-108, Jan 1979.
DOI: https://doi.org/10.2307/2346830

[12] Zhexue Huang, "Extensions to the k-means Algorithm for Clustering Large Data Sets with Categorical Values", Data Mining and Knowledge Discovery, pp. 283-304, Sep 1998.
DOI: https://doi.org/10.1023/A:1009769707641

[13] Manoranjan Dash, Hang Liu, "Feature Selection for classification", Intelligent data analysis, Vol. 1, No. 3, pp. 131-156, Jan 1997.
DOI: https://doi.org/10.3233/IDA -1997-1302

[14] Alexander Liu, Joydeep Ghosh Member, "Generative Oversampling for Mining Imbalanced Datasets", Proceedings of International Conference on Data Mining 2007, pp. 25-28, 2007.

[15] Hui Han, Wen-Yuan W, Bing-Huan M, "Borderline-SMOTE: a new over-sampling method in imbalanced data sets learning", ICIC, pp. 878-887, 2005.
DOI: https://doi.org/10.1007/11538059_91

[16] Xu-Ying L, Jianxin Wu, Zhi-Hua Z, Senior Member, "Exploratory Undersampling for Class-Imbalance Learning", IEEE Transactions on Systems, Vol. 39, No. 2, pp. 539-550, Dec 2009.
DOI: https://doi.org/10.1109/TSMCB.2008.2007853

[17] Ian J. Goodfellow, J Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, Yoshua Bengio, "Generative Adversarial

Networks", arXiv, Jun 2014.
DOI: https://doi.org/10.1145/3422622

[18] Lei Xu, Maria Skoularidou, Alfredo Cuesta-Infante, Kalyan Veeramachaneni, "Modeling Tabular using Conditional GAN" arXiv, Jul 2019.
DOI: https://doi.org/10.5555/3454287.3454946

[19] Jiwon Choi, Jaewook Lee, Duksan Ryu, Suntae Kim, "Identification of Generative Adversarial Network Models Suitable for Software Defect Prediction", Journal of KIISE, Vol. 49, No. 1, pp. 52-59, Jan 2022.
DOI: https://doi.org/10.5626/jok.2022.49.1.52.

[20] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, Pierre-Antoine M, "Extracting and composing robust features with denoising Autoencoders", Proceedings of the 25th international conference on Machine learning, Vol. 25, pp. 1096-1103, Jul 2008.
DOI: https://doi.org/10.1145/1390156.1390294.

[21] Diederik P. Kingma, Max Welling, "Auto-Encoding Variational Bayes", arXiv, May 2014.
DOI: https://doi.org/10.48550/arXiv.1312.6114.

[22] Puneet Misra, Arun S. Yadav, "Improving the classification accuracy using recursive feature elimination with cross-validation", International Journal on Emerging Technologies, Vol. 11, No. 3, pp.659-665, May 2020.

저 자 소 개

정 주 은(준회원)

- 2021년 ~ 현재 : 서울시립대학교 전자전기컴퓨터공학과 석사과정
- 관심분야 : 빅데이터, 데이터베이스, 딥러닝, 추천시스템

김 한 준(정회원)

- 1994년 : 서울대학교 계산통계학과 (이학사)
- 1996년 : 서울대학교 전산과학과(이학석사)
- 2002년 : 서울대학교 컴퓨터공학부 (공학박사)
- 2002년 ~ 현재 : 서울시립대학교 전자전기컴퓨터공학부 정교수
- 관심분야 : 데이터사이언스, 머신러닝, 텍스트마이닝, 데이터베이스, 정보검색

전 종 훈(정회원)

- 1986년 : Computer Science, University of Denver(학사)
- 1988년 : Computer Science, Northwestern University(공학석사)
- 1992년 : Computer Science, Northwestern University(공학박사)
- 1992년 ~ 1995년 : University of Central Oklahoma Department of Computing Science 조교수
- 1995년 ~ 현재 : 명지대학교 융합소프트웨어학부 정교수
- 관심분야 : 빅데이터, 데이터베이스, 정보검색, 지능형 소프트웨어, 의료정보시스템

※ 본 연구는 과학기술정보통신부 및 정보통신기술진흥센터의 대학 ICT 연구센터지원사업(IITP-2022-2018-0-01417)의 연구결과로 수행되었으며, 또한 2020년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원(No. 2020-0-00121, 데이터 품질 평가기반 데이터 고도화 및 데이터셋 보정 기술 개발)을 받아 수행되었음.