J. Internet Comput. Serv.
ISSN 1598-0170 (Print) / ISSN 2287-1136 (Online)
http://www.jics.or.kr


# 적은 양의 데이터에 적용 가능한 계층별 데이터 증강 알고리즘

# A layered-wise data augmenting algorithm for small sampling data

조 희 찬[1] 문 종 섭[1*]
Hee-chan Cho Jong-sub Moon

**요 약**

데이터 증강(Data Augmentation)은 적은 양의 데이터를 바탕으로 다양한 알고리즘을 통해 데이터의 양을 늘리는 기술이다. 현실 문제를 해결하기 위해 기계학습 및 딥러닝 기법을 사용하는 경우, 데이터 셋이 부족한 경우가 많다. 데이터의 부족은 모델 학습 시, 데이터 셋의 특징을 잘 반영하지 못하는 것 이외에도 과소적합 및 과적합에 빠질 위험이 크다. 따라서 본 논문에서는 오토인코더와 고유값 분해를 기반으로 하는 데이터 증강 기법을 통해 데이터를 증강 시키고 이를 심층 신경망의 각 층 마다 적용하여, 심층 신경망을 효과적으로 사전 학습하는 방법을 제시한다. 이후, WOBC 데이터와 WDBC 데이터에 대해 실험을 통하여 논문에서 제안하는 방법이 분류 정확도를 향상시키는지 측정하고 기존 연구들과 비교함으로써 제안한 방법이 실질적으로 의미가 있는 데이터를 생성하고 모델의 학습에 효과적임을 보인다.

☞ 주제어 : 키워드 : 딥러닝, 데이터 증강, 고유값 분해

**ABSTRACT**

Data augmentation is a method that increases the amount of data through various algorithms based on a small amount of sample data. When machine learning and deep learning techniques are used to solve real-world problems, there is often a lack of data sets. The lack of data is at greater risk of underfitting and overfitting, in addition to the poor reflection of the characteristics of the set of data when learning a model. Thus, in this paper, through the layer-wise data augmenting method at each layer of deep neural network, the proposed method produces augmented data that is substantially meaningful and shows that the method presented by the paper through experimentation is effective in the learning of the model by measuring whether the method presented by the paper improves classification accuracy.

☞ keyword : Deep learning, data augmentation, Eigen decomposition

## 1. 서 론

딥러닝은 자연어 처리, 사물 분류, 감정 분석 등 다양한 연구 분야에서 성공적인 결과를 거두고 있다[1]. 딥러닝에서 사용되는 심층 신경망(deep neural network)은 여러 개의 은닉 층(hidden layer)을 포함하며, 모델의 입력과 출력 사이의 관계를 학습하는 매우 표현적인 모델을 만든다. 학습에 사용될 수 있는 충분한 양의 데이터 셋은 심층 신경망 모델 학습의 성공적인 결과를 위한 필수 요소 중 하나이다.

하지만 일반적으로 문제 해결을 위해 딥러닝 기법을 사용하기에는 데이터 셋의 충분한 확보가 어려운 경우가 많다. 이렇게 제한된 훈련 데이터 셋으로 학습을 진행하는 경우, 학습이 제대로 이루어지지 않아 학습데이터에 대한 성능도 좋지 않고 학습되지 않은 데이터 셋에 대한 분류 성능이 떨어지는 과소적합(underfitting) 현상과, 학습은 성공적으로 되어 학습데이터에 대한 성능은 우수함에도 불구하고 학습된 모델의 학습되지 않은 데이터 셋에 대한 분류 성능이 떨어지는 과적합(overfitting) 현상이 발생한다[2].

위와 같은 문제들을 해결하기 위해 제한된 볼츠만 머신(Restricted Boltzmann Machine: RBM)을 통해 심층신경망 각 계층을 효과적으로 사전 학습(pre-training)하는 방법[3]과 신경망 전체를 다 학습시키는 것이 아닌 일부 노드를 무작위로 학습시키는 드롭아웃(Dropout)[4] 등 많은

1 Graduate School of Information Security, Korea University, Seoul, 02481, Korea
* Corresponding author (jsmoon@korea.ac.kr)
[Received 21 October 2019, Reviewed 27 October 2019, Accepted 20 November 2019]

방법들이 제안되었는데, 그중 한 가지는 데이터 증강(Data Augmentation)이다.

데이터 증강은 데이터 셋이 부족한 상황에서 특정 알고리즘에 따라 데이터의 특징을 반영하거나 원본 데이터의 확률을 반영한 데이터를 생성하여, 데이터의 양을 늘리는 기술이다.

데이터 증강 기법은 대표적으로 이미지 데이터의 경우에 가우시안 잡음(gaussian noise) 추가, 색 반전, 흐림 효과(blur), 대비 효과(contrast), 반전 효과(flipping), 세그먼테이션(segmentation), 이미지 자르기(cropping) 등이 있다[5]. 다른 종류의 데이터 역시도, 데이터의 특징은 유지하면서, 데이터를 변형하는 데이터 증강 기법을 적용함으로써 더 많은 학습 데이터를 생성한다.

본 논문에서는 학습 성능을 향상시키기 위해 심층 신경망 계층 별 데이터 증강 개념을 도입한 새로운 데이터 증강 기법을 제안한다. 이 기법은 심층 신경망의 계층마다 데이터 증강을 수행하는데, 이때 각 계층은 해당 계층의 입력 데이터에 대하여, 고유값(eigen value)을 이용한 샘플링 알고리즘을 적용하여 입력 데이터에 대한 데이터 증강을 수행하고 파라미터의 사전 학습을 위해 적층 오토인코더(Stacked Autoencoder)를 사용한다.

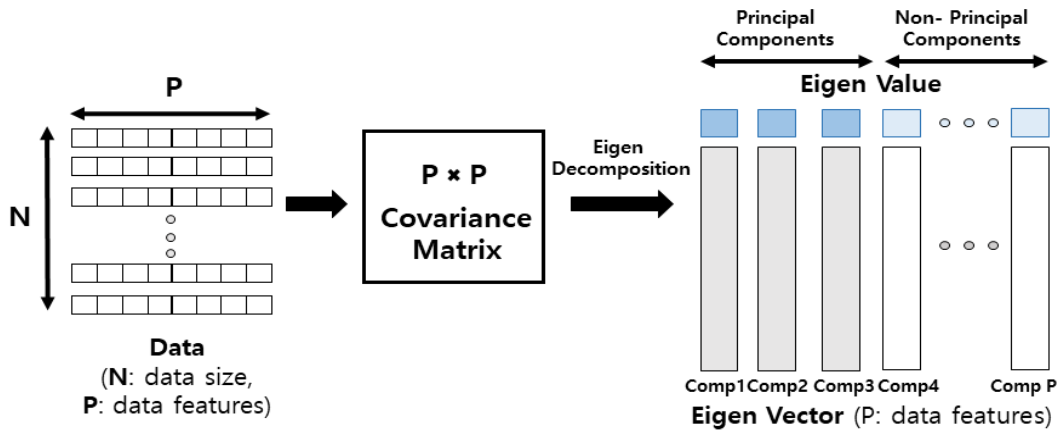
본 논문의 구성은 다음과 같다. 2장은 관련연구로서 모델에서 사용될 오토인코더와 고유값 분해, 데이터 증강 기법의 대표적인 방법들을 설명하고, 3장에서는 제안하는 딥러닝 모델의 전체적인 구조를 설명한다. 또한 데이터 증강 알고리즘 및 제안하는 계층별 데이터 증강 기법을 정형적으로 기술한다. 4장에서는 제안 기법의 성능을 평가하기 위한 실험을 보인다. 마지막으로 5장에서는 결론을 기술하고 향후 연구 방향을 제시한다.

## 2. 관련 연구

### 2.1 데이터 증강 기법

딥러닝 모델의 과부족 학습 데이터 문제를 해결하는 방법으로 주로 사용되는 것 중 하나는 훈련 데이터 셋의 양을 증가시키는 것이다. 하지만, 훈련 데이터 셋의 충분한 확보는 시간 및 비용적인 측면 등의 이유로 쉽지 않다. 따라서 충분하지 않은 데이터 셋으로도 모델을 학습하기 위하여 사용되는 방법 중의 하나가 바로 데이터 증강이다. 데이터 증강이란 원본 데이터 셋을 기반으로 새로운 데이터를 만드는 기법을 말한다.

학습에 필요한 새로운 데이터를 샘플링 하는 연구는 이전부터 다양한 방법으로 이루어지고 있는데, 무의미한 데이터를 단순 샘플링 하는 대신, 지능적으로 샘플링 하는 관련 연구로는 Chwala 등이 발표한 Synthetic Minority Oversampling Technique(SMOTE)를 예로 들 수 있다[6]. SMOTE 기법은 기존 데이터의 개수가 부족한 클래스의 샘플을 보간하여 새로운 소수 클래스의 데이터를 생성하며, 소수 클래스의 데이터에 대해 K-nearest neighbor[7] 기법을 사용하여 새로 합성된 데이터가 소수 클래스의 특성이 유실되지 않고 반영된다. Hu 등은 SMOTE를 발전 시켜 Modified Synthetic Minority Oversampling Technique(MS




(그림 1) 고유값 분해를 이용한 고유 벡터 및 고유값 계산

(Figure 1) Calculate eigen vector and eigen value using eigen decomposition

MOTE) 기법을 제시하였는데 MSMOTE 기법은 소수 클래스의 샘플을 3개의 그룹(Security, Noise, Border)으로 나누고 각 그룹은 각각 분류 모델의 성능을 향상을 야기하는 데이터 포인트, 분류기의 성능을 감소시키는 데이터 포인트, 둘 중에 하나로 분류하기 힘든 데이터 포인트의 특성을 가진다[8].

이외에도 Integrated Oversampling(INOS)[9]라는 하이브리드 샘플링 기법이 제안되었는데 기존의 방법으로는 분류하기 어려운 클래스 경계에 가까운 데이터를 제외하지 않으며, 원래의 소수 클래스의 데이터들의 주요 공분산 구조를 학습할 수 있다.

### 2.2 주성분 분석

주성분 분석(Principal Component Analysis)[10]은 변수들 사이의 분산과 공분산 관계를 이용하여 이 변수들의 선형 결합으로 표시되는 주성분을 찾고, 주성분을 통해 전체 변동을 설명하고자 하는 다변량 분석법이다.

주성분 분석은 데이터의 간소화나 선형 관계식을 통하여 데이터의 차원을 감소시켜 해석의 용이성을 증가시키는 목적으로 하며, 주성분들은 고유값 분해를 통하여 계산된다.

고유값 분해(eigen decomposition)는 행렬을 여러 개의 고유 벡터(eigen vector)와 고유값으로 분해하는 기법이다[11]. 데이터의 경우 공분산 행렬을 계산하여 고유 벡터와 고유값을 구하게 되면, 내림차순으로 정렬되게 되며, 이는 고유벡터 축(eigen vector axis)으로 변환된 데이터의 분산이 큰 순서대로 정렬된 것과 같다. 일반적으로 데이터는 구성요소들 간에 복잡한 의존성이 있기 때문에, 독립적으로 만들어주는 목적으로 고유값 분해를 수행한다. 그림 1과 같이 $p$차원 입력 데이터의 공분산 행렬 $\Sigma_x$을 구하여, 고유값 분해를 수행하면 고유벡터와 고유값으로 이루어진 구성요소들로 나뉘게 되며, 구성요소들은 데이터를 표현하는데 중요한 주성분(principal components)과 비주성분(non principal component)으로 나눠지게 된다.

### 2.3 오토인코더

오토인코더는 인공신경망의 일종으로 입력 계층, 출력 계층, 하나의 은닉 계층으로 구성되며, 입력 데이터와 복원된 출력 데이터 간 손실이 최소화되도록 학습된다[12]. 전통적인 오토인코더는 은닉 계층의 벡터를 입력 및 출력 계층의 벡터 수보다 감소시켜 뉴런의 병목현상을 이용하며, 이러한 제약조건은 차원 축소를 통하여 입력데이터의 주요 특징을 학습 가능하게 한다[13].

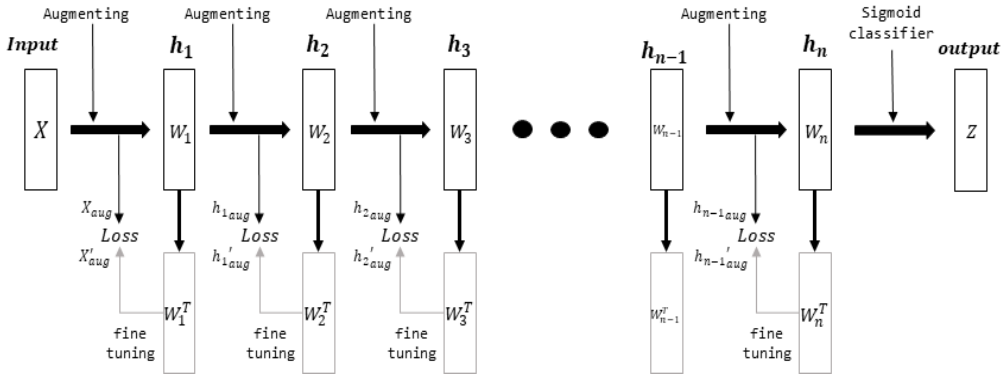
## 3. 제안하는 방법

### 3.1 개요

적층 오토인코더는 2개 이상의 은닉 계층을 가지는 오토인코더로써, 층이 깊어질수록 복잡한 부호화의 학습이 가능하다.

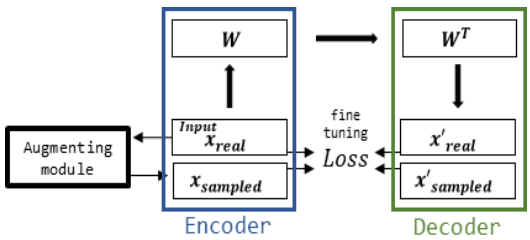
본 논문에서 제안하는 계층 별 데이터 증강 알고리즘을 적용한 적층오토인코더의 구조는 그림 2와 같다.

제안된 기법은 첫 번째 은닉 계층을 데이터 증강 알고리즘을 통해 증강시킨 데이터를 입력으로 그림 3과 같은




(그림 2) 제안된 데이터 증강 기법을 적용한 모델 구조
(Figure 2) The structure of model using proposed augmenting method

오토인코더 모듈을 통해 비지도 학습하고 이 과정을 전체 적층 오토인코더의 각 은닉 계층에 대해서 순차적으로 적용하여 모든 신경망의 파라미터를 사전 학습 하는 형태로 구성된다. 사전 학습을 마친 후에는 미세조정(fine-tuning)을 수행하며, 결과 예측을 위한 활성화 함수로는 시그모이드 함수를 사용하였다.




(그림 3) 오토인코더 모듈
(Figure 3) An autoencoder module

## 3.2 계층 별 데이터 증강 알고리즘

주어진 입력 데이터에 대하여 고유값 분해를 수행하면 결과적으로 데이터를 표현하는데 주요한 주성분과 큰 영향이 없는 비주성분을 구할 수 있다. 논문에서는 $p$차원 입력 데이터 $x$에 대하여 주성분 분석을 수행한 후, 이를 이용하여 샘플링 된 데이터 $x'$을 생성한다.

표 1은 고유값 분해를 이용하여 심층 신경망의 계층별 학습 시 입력 데이터를 증강하는 알고리즘이다. 먼저, $p$차원 입력 데이터 $x$, 샘플링 할 데이터의 양 $m$, 실험을 통해 경험적으로 결정된 주성분 개수 $d$를 입력으로 받는다. 이후, $x$의 공분산 행렬 $\Sigma_x \in R^{p\times p}$와 평균 $\mu_x \in R^p$을 구한다. $\Sigma_x$을 고유값 분해하여 고유 값 $\Lambda$와 고유 벡터 $U$를 계산하며 $\Lambda, U$의 값들은 내림 정렬된다. 입력으로 준 주성분 개수 $d$를 기준으로 $U$ 의 구성요소들을 첫 번째 값부터 $d$ 번째까지 나눈 $U_{1:d}$을 주성분 그리고, $d$번째부터 나머지 $U_{d:p}$를 비주성분 이라한다. 이후 $x$를 $U_{1:d}$에 사영한 $q$와 입력 데이터의 평균 $\mu_x$을 비주성분에 사영한 $s$를 계산하며 $q$는 주성분 분석 결과와 동일하다. 이는 $x$의 특성을 유지하면서 차원을 감소시켜, 증강 알고리즘의 계산량을 줄이는 효과가 있다. 또한, s는 $x$를 나타내는데 중요하지 않기 때문에 적당한 값을 계산하고 반복적으로 사용함으로써 증강 알고리즘의 계산량을 줄인다.

$q$의 평균$\mu_q$와 분산$\Lambda_{1:d}$을 이용하여 가우시안 분포 $N(\mu_q, \Lambda_{1:d})$로부터 $q$'을 샘플링 할 수 있고, $q$'과 $s$를 합친 값에 $U^T$를 곱하여 역사영하면 입력데이터 $x$의 특징을 반영한 샘플링 데이터 $x'$가 생성된다. 마지막으로 입력데이터 $x$와 샘플링 된 데이터 $x'$를 합쳐 증강된 입력데이터 $x^{aug}$를 생성 해낸다.

(표 1) 고유값 분해를 이용한 데이터 증강 알고리즘
(Table 1) Data augmentation algorithm using eigen decomposition

| **Augmenting algorithm** |
|---|
| **inputs**<br>- $p$ sized data $x$<br>- augmenting size $m$<br>- number of principal components $d$<br>**outputs**<br>- augmented data $x^{aug}$ |
| **1.** calculate $\Sigma_x \in R^{p\times p}$ of $x$ |
| **2.** calculate $\mu_x \in R^p$ of $x$ |
| **3.** $\Lambda, U \leftarrow$ do eigen decomposition for $\Sigma_x$ s. t $\Lambda = U^T \Sigma_x U$ |
| **4.** $\Lambda_{1:d}, U_{1:d} \leftarrow$ the largest $d$ eigen values and eigen vectors in $\Lambda, U$ |
| **5.** $\Lambda_{d:p}, U_{d:p} \leftarrow$ the smallest $p-d$ eigen values and eigen vectors in $\Lambda, U$ |
| **6.** $q \leftarrow x U_{1:d}$ |
| **7.** $s \leftarrow \mu_x U_{d:p}$ |
| **8.** calculate $\mu_q$ of $q$ |
| **9.** sample $q' \sim N(\mu_q, \Lambda_{1:d})$ s. t. $\lvert q'\rvert = m$ |
| **10.** $x' = \begin{bmatrix} q' \\ s' \end{bmatrix} U^T$ |
| **11.** $x^{aug} \leftarrow$ concatenate $x$ with $x'$ |

## 3.3 계층별 데이터 증강을 통한 사전 학습

일반적인 심층 신경망 학습의 경우에 기울기 소실[14] 문제와 오버피팅 문제로 인해 학습에 어려움이 있다. 이러한 문제를 해결할 수 있는 방법 중 하나는 계층별 탐욕 학습(Greedy layer-wise training)[3] 이다. 이 알고리즘은 하나의 인접 계층씩 학습 과정을 반복하는 사전 학습과 사전 학습이 완료된 후 전체 모델에 대해서 학습하는 미세조정으로 구성된다. 이러한, 계층별 탐욕 학습 방법은 적층 오토인코더 및 심층신경망의 학습에 효과가 있음이 증명되었다[15].

표 2는 본 논문에서 제안하는 증강 데이터를 이용한 계층별 사전 학습 알고리즘이다. 먼저 데이터를 학습시킬 $L$개의 계층을 가진 심층신경망을 구성하여 파라미터를 랜덤하게 초기화한다. 매 계층의 파라미터를 사전 학습하기 위해 학습시킬 계층의 입력으로 들어올 이전 계층의

데이터를 고유값 분해를 이용한 데이터 증강알고리즘을 통해 증강한다. 이후 그림 3과 같은 오토인코더 모듈을 구성하여 비지도 학습 방법으로 해당 계층의 파라미터 학습을 진행한다. 학습이 완료된 계층의 파라미터를 고정시킨 후, 동일한 방법으로 다음 계층의 학습을 진행한다. 이때, 다음 계층의 입력으로는 이전 계층의 결과를 입력으로 사용하게 되며 첫 번째 입력데이터의 경우에 한해서 주성분분석을 통한 입력데이터의 차원감소가 가능하다.

(표 2) 증강된 데이터 기반 계층별 사전학습 알고리즘
(Table 2) Layer-wise pre-training algorithm using augmented data

| **Data augmented layer-wise pre-training algorithm** |
|---|
| **inputs**<br>- randomly initialized deep neural net with $L$ layers<br>- training data set $X$<br>- mini-batch size $n$<br>- augmenting size $m$<br>- number of principal components $d$ |
| **outputs**<br>layer-wise trained deep neural net with $L$ layers |
| **1. for** $l \leftarrow 1$ to $L$ **do**<br>**2.** $x \leftarrow$ get $n$ sized mini-batch of $X$<br>**3.** $z_0 \leftarrow x$<br>**4.** **for** $i \leftarrow 1$ to $l-1$ **do**<br>**5.** $z_i \leftarrow h_i(z_{i-1})$<br>**6.** **end for**<br>**7.** $z_{l-1}^{aug} \leftarrow$ augment($z_{l-1}, m, d$)<br>**8.** train the $l$-th layer using $z'_{l-1}$<br>**9.** **end for** |

위 입력데이터 증강 및 학습 과정을 모델의 모든 은닉계층들을 대상으로 반복하여 사전 학습을 수행한다. 전체 신경망에 대하여 미세조정을 통해 학습을 완료한다. 이때 파라미터를 학습하기 위해 정답 값으로 라벨 값에 원핫 인코딩을 적용한 값을 사용하였고, 시그모이드 함수를 활성화 함수로 이용하여 데이터의 결과를 평가하였다.

### 3.4 시그모이드 함수

제안된 방법을 사용하여 파라미터의 사전 학습이 모두 완료된 심층신경망을 대상으로 미세조정을 실행하는데, 이때 파라미터를 학습하기 위한 정답 값은 원핫 인코딩이 적용된 라벨 값을 사용하였고, 결과 예측을 위한 활성화 함수로는 시그모이드 함수[16]를 사용하였다.

시그모이드 함수는 로지스틱 회귀분석 또는 인공신경망의 이진 분류에서 마지막 레이어의 활성화 함수로 사용된다. 임의의 데이터 $x$가 주어졌을 때 $x$를 예측하는 식 (1)은 다음과 같이 나타낼 수 있다.

$$p(x) = \frac{1}{1+e^{-x}} \quad (1)$$

모델의 미세조정 과정은 시그모이드 함수를 계산하여 나온 값과 정답 값을 비교하여 오차를 줄여나가는 방향으로 진행되며, 손실함수로는 크로스엔트로피 함수를 사용한다.

## 4. 실 험

### 4.1 실험 환경

본 논문에서 실험에 사용된 시스템 환경은 다음 표 3과 같다.

(표 3) 실험 환경
(Table 3) Experimental environment

| **System Spec** | |
|---|---|
| **OS** | Ubuntu 16.04 |
| **CPU** | AMD Ryzen 7 1800X |
| **GPU** | NVIDIA GeForce GTX 1080 Ti |
| **RAM** | 64GB |

본 논문에서 사용된 데이터 셋은 UCI Machine Learning Repository[17] Wisconsin Diagnostic Breast Cancer(WDBC) 데이터 셋과 Wisconsin Original Breast Cancer(WOBC) 데이터 셋으로서 패턴인 식 및 기계학습에 널리 사용되는 자료이다.

WDBC 데이터 셋은 569명을 대상으로 조사되었고 악성과 양성을 나타내는 클래스 변수 1개 및 30개의 독립변수로 구성된다. 30개의 독립 변수는 세포 특성을 나타내는 10개의 변수들에 대해 평균값 표준편차 그리고 최대값을 나타내는 3개의 변수로 구성되어있다.

WOBC 데이터 셋은 699명을 대상으로 조사되었고 세침흡인 세포검사의 세포 특성을 나타내는 9개의 변수와 이를 악성인지 정상인지 나타내는 클래스 변수로 구성되어 있다.

## 4.2 실험 결과

본 논문에서 제안하는 계층별 데이터 증강 기법의 측정하기 위해서, 실험에 사용된 데이터 셋을 활용한 기존 논문, 제안한 방법을 적용하지 않았을 때의 심층 신경망, 그리고 제안한 방법을 적용했을 때의 심층 신경망 모델의 정확도를 비교한다.

### 4.2.1 모델 파라미터

WDBC 데이터 셋과 WOBC 데이터 셋을 테스트한 모델의 파라미터는 은닉 계층의 개수(HL), 계층별 가중치 개수(W), 활성화 함수(AF), 학습율(LR), 주요구성요소비율(N_SIG), 배치사이즈(N_BATCH), 샘플사이즈(N_SAMPLE)가 있다. 표 4와 표 5는 각각 WDBC 데이터와 WOBC 데이터를 학습한 모델의 파라미터 값이다.

(표 4) WDBC 데이터 모델 파라미터
(Table 4) parameter for WDBC data set

| WDBC Model parameter | |
|---|---|
| **HL** | 9 Layers |
| **W** | (Input), 10, 20, 30, 30, 20, 20, 20, 20, 10, (Output) |
| **AF** | ReLu |
| **LR** | 0.0001 |
| **N_SIG** | 0.25 |
| **N_BATCH** | 25 |
| **N_SAMPLE** | N_BATCH * 10 |

(표 5) WOBC 데이터 모델 파라미터
(Table 5) parameter for WOBC data set

| WOBC Model parameter | |
|---|---|
| **HL** | 12 Layers |
| **W** | (Input), 10, 20, 30, 40, 40, 30, 20, 20, 20, 30, 20, 10, (Output) |
| **AF** | ReLu |
| **LR** | 0.0001 |
| **N_SIG** | 0.25 |
| **N_BATCH** | 50 |
| **N_SAMPLE** | N_BATCH * 10 |

위 파라미터는 분류 성능의 최대화를 위한 파라미터가 아닌, 제안된 기법을 적용한 신경망과 적용하지 않은 신경망의 차이에 중점을 두고 실험에 근거하여 결정된 파라미터 값이다.

### 4.2.2 결과 비교

WDBC 데이터 셋과 WOBC 데이터 셋을 대상으로 기존 논문들에서 각각의 분류방법으로 의사결정트리(TREE), 로지스틱회귀분석(LOGISTIC), 판별분석(LDA), 적용한 정확도와 본 논문에서 구현한 인공신경망 모델에 제안한 방법을 적용하지 않았을 때의(NN) 정확도와 적용했을 때의(Augmented NN) 정확도를 비교한다. 표 6과 표 7에서 볼 수 있듯이, 제안한 방법을 적용한 인공 신경망이 다른 모델보다 성능이 좋은 것을 확인 할 수 있다. 이는 제안한 방법을 통해 계층별로 원본 입력 데이터의 특징을 포함하고 있는 증강된 데이터를 입력으로 심층 신경망을 비지도 학습시켜 파라미터 초기값을 사전 학습시키고 이를 기반으로 미세조정을 진행하였기 때문에 높은 정확도를 나타낸다.

(표 6) WDBC 데이터 셋 성능 비교
(Table 6) Performance comparison for WDBC data

| model | Accuracy |
|---|---|
| TREE[18] | 0.9373 |
| LOGISTI18] | 0.9427 |
| LDA[18] | 0.9568 |
| NN | 0.94736844 |
| **Augmented NN** | **0.9737** |

(표 7) WOBC 데이터 셋 성능 비교
(Table 7) Performance comparison for WOBC data

| model | Accuracy |
|---|---|
| TREE[18] | 0.9390 |
| LOGISTI18] | 0.9556 |
| LDA[18] | 0.9523 |
| NN | 0.9474 |
| **Augmented NN** | **0.9649** |

## 5. 결 론

본 논문에서는 충분하지 않은 데이터 셋으로 인한 심층신경망의 학습의 어려움을 해결하기 위해 신경망의 모

든 계층에 대해 데이터 증강을 수행하는 계층별 데이터 증강 기법을 제안하였다.

계층별 데이터 증강은 기존의 입력 계층에 국한된 증강 기법이 아닌 계층 단위의 증강을 통해 학습 능력 향상을 위한 매 계층에 대한 입력 데이터를 생성한다. 또한, 고유값 분해를 기반으로 하는 증강 알고리즘을 사용하여 기존 데이터의 특징을 반영하는 샘플링 데이터를 생성함은 물론, 이미지나 통계데이터 등 데이터 종류에 상관없이 적용하여 증강이 가능하다.

WDBC 데이터 셋과 WOBC 데이터 셋을 대상으로 본 논문에서 제안한 계층별 데이터 증강 모델의 성능을 실험한 결과 검증 데이터 셋과 테스트 데이터 셋에 대하여 기존의 다른 모델보다 높은 분류 성능을 발휘함을 확인하였다.

계층별 데이터 증강 알고리즘은 데이터의 차원이 적당한 데이터에 대해서는 연산 시간이 합리적이고 좋은 성능을 보이지만, 이미지와 같이 데이터의 차원이 복잡한 데이터에는 증강 연산 시간이 많이 소요된다는 단점이 있다. 따라서 향후 연구로는 데이터의 계층별 증강 단계에서 연산량을 줄여나갈 수 최적의 알고리즘을 찾아내고자한다.

## 참고문헌(Reference)


[1] Y. LeCun, Y. Bengio, A. Courville, and G. Hinton, “Deep Learning,” Cambridge: MIT Press, 2016.

[2] C. Rich, S. Lawrence, and C. -L. Giles, ″Overfitting in neural nets: Backpropagation, conjugate gradient, and early stopping,″ Advances in neural information processing systems, 2001. https://papers.nips.cc/paper/1895-overfitting-in-neural-nets-backpropagation-conjugate-gradient-and-early-stopping.pdf

[3] G. E. Hinton, S. Osindero and Y. W. Teh, “A fast learning algorithm for deep belief nets,” Neural computation, 18(7), pp.1527-1554, 2006. https://doi.org/10.1162/neco.2006.18.7.1527

[4] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, andR. Salakhutdinov, “Dropout: A Simple Way to Prevent Neural Networks from Overfitting,” Journal of Machine Learning Research, Vol.15, No.1, pp.1929-1958, 2014. http://jmlr.org/papers/v15/srivastava14a.html

[5] C. Shorten and T. M. Khoshgoftaar, “A survey on Image Data Augmentation for Deep Learning,” Journal of Big Data, 6(1), 60, 2019. http://doi.org/10.1186/s40537-019-0197-0

[6] N. V. Chawla, L. O. Hall, K. W. Bowyer, and W. P. Kegelmeyer, ″Smote: Synthetic minority oversampling technique,″ Journal of Artificial Intelligence Research, Vol.16, pp.321-357, 2002.

[7] N. S. Altman, “An introduction to kernel and nearest-neighbor nonparametric regression,” The American Statistician, 46(3), pp.175-185, 1992. http://doi.org/10.1080/00031305.1992.10475879

[8] S. Hu, Y. Liang, L. Ma and Y. He, ″MSMOTE: improving classification performance when training data is imbalanced,″ 2009 second international workshop on computer science and engineering, Vol.2, pp.13-17, 2009.

[9] H. Cao, X.-L. Li, D.-K. Woon, and S.-K. Ng, “Integrated oversampling for imbalanced time series classification,” IEEE Trans. Knowl. Data Eng., vol. 25, no. 12, pp.2809–282, Dec. 2013. https://doi.org/10.1109/TKDE.2013.37

[10] S. Wold, K. Esbensen, and P. Geladi, “Principal component analysis,” Chemometrics and intelligent laboratory systems, 2(1-3), pp.37-52. 1987.

[11] H. Abdi, “The eigen-decomposition: Eigenvalues and eigenvectors,” Encyclopedia of measurement and statistics, pp.304-308, 2007. https://personal.utdallas.edu/~herve/Abdi-EVD2007-pretty.pdf

[12] P. Vincent, H. Larochelle, Y. Bengio and P. A. Manzagol, “Extracting and composing robust features with denoising autoencoders,” In Proceedings of the 25th international conference on Machine learning, pp.1096-1103, 2008.

[13] G. E. Hinton and R. R. Salakhutdinov, “Reducing the Dimensionality of Data with Neural Networks,” Science, Vol. 313, no. 5786, pp.504-507, 2006. https://doi.org/10.1126/science.1127647

[14] S. Hochreiter, The vanishing gradient problem during learning recurrent neural nets and problem solutions. International Journal of Uncertainty, Fuzziness and

Knowledge-Based Systems, 6(02), pp.107-116, 1998.
[15] Y. Bengio, P. Lamblin, D. Popovici, and H. Larochelle, “Greedy Layer-Wise Training of Deep Networks,” Adv. Neural Inf. Process. Syst., Vol. 19, no. 1, pp. 153-160, 2007.
[16] Finney, D. John, “Probit analysis: a statistical treatment of the sigmoid response curve,” Cambridge university press, Cambridge, 1952.
[17] UCI Machine Learning Repository. University of California, Center for Machine Learning and Intelligent Systems. Available at https://archive.ics.uci.edu/ml/datasets.php
[18] Lim, J. S., Oh, Y. S., & Lim, D. H, “Bagging support vector machine for improving breast cancer classification,” J Health Info Stat, 39(1), pp.15-24. 2014. https://e-jhis.org/journal/view.php?number=426

## ◐ 저 자 소 개 ◑

**조 희 찬(Hee-chan Cho)**
2018년 고려대학교 전자 및 정보공학과
2018년~현재 고려대학교 정보보호대학원 석사과정
관심분야 : 인공지능, 시스템보안
E-mail : h2echan2@korea.ac.kr

**문 종 섭(Jong-sub Moon)**
1981년 서울대학교 계산통계학과
1983년 서울대학교 대학원 계산통계학과
1991년 Illinois Institute of Technology 전산학 박사
1993년~현재 고려대학교 전자 및 정보공학부 교수
관심분야 : 생체인식, 운영체제, 침입탐지
E-mail : jsmoon@korea.ac.kr