



숫자 인식 서비스 기반 다층 퍼셉트론에 대한 입력 데이터 축소 알고리즘

김창민* 박상주**

A Number Recognition Service Based on Multi-Layer Perceptron for an Input Conversion Algorithm

Chang-Min Kim*, Sang-Ju Park**

요 약

IT 융·복합 기술은 많은 양의 데이터를 생성하며, 이를 스스로 판별할 수 있는 딥러닝 기술은 현재 산업 분야에서 큰 관심을 받고 있다. 생물학적 뇌를 인공적으로 모델링 한 딥러닝은 고연산량이 요구된다. 본 논문은 딥러닝의 근본적 문제인 고연산량을 줄이기 위한 첫걸음으로 숫자 데이터 축소 알고리즘을 제안한다. 이미지 내 존재하는 특징을 파악 후 각 특징들의 개수를 측정하여 고정적인 배열 16 크기로 정리한다. 이를 통해 신경망의 연산 및 학습 시간을 감소시킨다. 제안한 숫자 데이터 축소 알고리즘의 성능을 평가하기 위해 가공된 데이터를 다층 퍼셉트론을 통해 학습 후 3계층-ANN, CNN-LeNet5와 비교한다. 실험결과 정확도 99.4%, 오차율 0.0264%로 다소 저조한 결과를 보였으나 시간측정에서는 기존 방식보다 3배 이상 빠른 연산을 보였다.

Abstract

IT fusion and composite technologies generate a large amount of data, and deep learning technologies that can identify them on their own are currently receiving great attention in the industrial field. Deep learning, which artificially models biological brains, requires a high amount of computation. In this paper, we propose a numerical data reduction algorithm as a first step to reduce high computations, a fundamental problem of deep learning. After identifying the all features in the image, measure the number of features and arrange them into a fixed array of 16 sizes. This reduces the computation and learning time of the neural network. In order to evaluate the performance of the proposed numerical data Conversion algorithm, the processed data is compared with 3-layer-ANN and CNN-LeNet5 after learning through multilayer perceptron. Experimental results showed somewhat poor results with accuracy of 99.4% and error rate of 0.0264%. However, the time measurement showed more than three times faster operation than the conventional method.

Keywords

artificial neural networks, multi-layer perceptron, mobile pattern recognition, recognition

* 상지대학교 컴퓨터공학전공 박사과정

- ORCID: <https://orcid.org/0000-0002-9735-6921>

** 안산대학교 인터넷정보과 교수(교신저자)

- ORCID: <https://orcid.org/0000-0002-3916-5721>

· Received: Jul. 08, 2019, Revised: Sep. 23, 2019, Accepted: Sep. 26, 2019

· Corresponding Author: Sang-Ju Park

155, Ansandaehak-ro, Sangnok-gu, Ansan-si, Gyeonggi-do, Republic of Korea

Tel.: +82-31-400-6974, Email: sangjoo77@hanmail.net

I. 서 론

IT분야의 발전으로 인하여 융·복합 기술이 점차 증가함에 따라 엄청난 양의 데이터가 생성되고 있다[1]. 이러한 데이터의 특징 추출 및 패턴 분석을 이용한 분류, 예측, 판단을 수행하는 지능형 시스템(음성, 얼굴, 숫자인식 등)이 산업분야에서 큰 주목을 받고 있다[1]-[3]. 그 중심에는 머신러닝과 딥러닝 기술이 있으며, 이 기술들은 뇌의 신경구조를 인공적으로 모델링(신경모델)하여 데이터를 학습한다. 신경모델은 여러 계층으로 구성되며, 각 계층은 많은 노드로 이루어진다. 각 노드는 인접된 계층의 노드와 완전연결 구조를 형성한다. 노드들은 입력 개수에 상응하는 가중치를 가지며, 학습을 통해 입력 데이터에 반응할 수 있는 값을 찾는다. 이 값을 찾는 과정에서 많은 연산이 요구된다. 모델의 구조 및 입력 데이터의 크기, 반복 횟수 등에 따라 연산량이 기하급수적으로 증가된다. 또한 신경모델의 발전으로 점차 계층이 늘어나고 있다. 즉, 처리해야하는 연산이 점차 증가되고 있다는 의미이다. 이는 차후 신경모델을 이용한 인식서비스가 보편화될 시 고 사양의 서버 및 PC가 요구된다. 물론 서버 및 PC를 증설 또는 업그레이드(하드웨어 측면)를 수행한다면 이 문제를 쉽게 해결할 수 있다. 하지만 이 방법을 수행할 수 없는 환경일 경우 인식서비스가 제한되며, 복합적인 서비스 제공이 어렵다. 또한 앙상블 학습 알고리즘(Ensemble learning algorithm)과 같이 다수의 인식모델 사용이 제한된다.

그래서 본 저자는 이러한 문제를 다소 해결하기 위한 데이터 축소 알고리즘을 제안한다. 제안한 데이터 축소 알고리즘은 학습모델의 입력 데이터(숫자)에 대한 사전처리를 통하여 데이터의 크기를 줄이는 방법으로, 축소된 데이터의 크기는 배열 16크기를 갖는다. 축소된 데이터를 학습모델이 학습함으로써 원본 데이터를 학습한 결과와 유사한 성능을 보인다. 이를 통해 고연산량을 줄이며, 학습속도를 높인다. 본 논문은 2장 관련연구에서 신경망 기법들을 설명하며, 3장에서 제안 알고리즘을 정의한다. 4장은 제안 알고리즘을 통해 가공된 데이터를 학습하여 기존 학습방법과 비교한 결과를 기술한다. 마지막으로 결론 및 향후 연구로 마무리한다.

II. 관련 연구

2.1 ANNs

ANNs(Artificial Neural Networks)의 기본적인 구조는 생물학적 뉴럴 네트워크(BNN, Biological Neural Networks)를 기반으로 한다[3][4]. BNN은 뉴런이라는 하나의 세포가 약 103~104의 다른 뉴런들과 상호 연결된 네트워크로 구성된 형태(인간의 뇌는 약 1,014~1,015개의 상호 연결됨)로, 밀리초 단위의 시간에 신호를 통신한다. 이러한 생물학적 뉴런을 McCulloch와 Pitts는 이진 원소로 판단하여 “A Logical Calculus of Ideas Immanent in Nervous Activity”을 통해 McCulloch-Pitts 모델을 제안하였다.

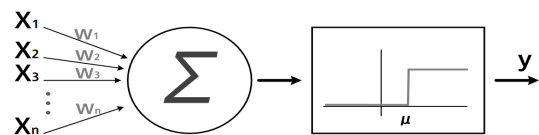


그림 1. 인공 뉴런
Fig. 1. Artificial neuron

그림 1은 인공 뉴런을 표현한 것으로, 합계 연산 부분과 임계값 비교를 위한 부분인 활성화 함수(Activation function)로 구성된다. 합계 연산은 입력과 각기 다른 가중치를 곱한 후 합산을 구하는 부분이며, 활성화 함수는 합계와 임계치를 비교하여 활성화 상태(0, 1)를 판단한다. 활성화 함수는 계단함수(Limiting), 임계 논리(Threshold logical) 등 다양한 함수가 존재한다. 하지만 이 모델은 불완전하며 연산의 모호함으로 인해 뇌 활동에 대한 정확한 모델링이 아닌 것으로 판명되었지만 오늘날의 인공지능망에 표본이 되었다.

2.2 다층 퍼셉트론 및 역전파 학습알고리즘

McCulloch-Pitts 모델 이후 프랭크 로젠블랫(Frank rosenblatt)에 의해 등장한 퍼셉트론(Perceptron)은 비교적 정확히 기술된 계산에 의해 최초의 신경망 모델이 되었다. 그는 인공 뉴런의 개념을 이용하여 단층 퍼셉트론을 제안하였지만 선형 분리의 문제를

해결하지 못했다. 이후 입·출력층 이외의 은닉층 (Hidden layer)을 추가한 다층 퍼셉트론(MLP, Multi-Layer Perceptron) 모델이 탄생되었으며, 오류 역전파 학습 알고리즘(Back-propagation)의 등장으로 단층 퍼셉트론의 문제 및 다양한 이슈를 해결하였다 [4]-[7].

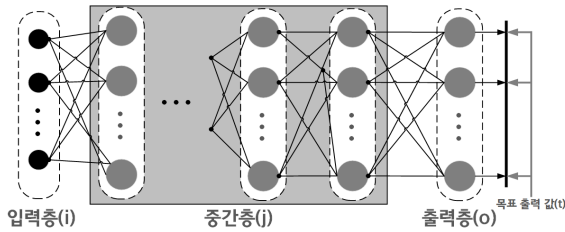


그림 2. MLP의 구조
Fig. 2. Multi-layer perceptron

그림 2는 MLP의 구조로, 각 계층 내 다수의 노드가 존재하며 각 노드는 인접된 계층의 모든 노드와 완전연결 구조를 형성한다. 입력층을 제외한 모든 계층은 이전 계층에서 도출된 결과를 입력으로 받으며, 각 노드는 일련의 연산과정(합계와 활성화 함수)을 수행 후 다음 계층으로 상태를 전달한다. 일반적으로 다층 퍼셉트론은 시그모이드 함수를 사용하며, 3계층으로 설계된다. 식 (1)은 시그모이드 함수이다.

$$\text{sigmoid}(x) = 1/(1 + \exp(-x)) \quad (1)$$

식 (1)의 x 는 입력 값이며, 지수함수를 통해 x 에 대한 0~1사이의 미분 가능한 수로 변환되어 시그모이드 그래프를 표현한다. 이 그래프는 S자와 유사한 시그모이드 커브 형태를 보인다. 이를 통해 각 노드는 활성화 상태를 1 또는 0으로 표현한다.

MLP은 일련의 신호를 입력층부터 각 계층을 순차적으로 처리하는 순방향 연산(Forward-propagate)을 수행한다. 이 과정에서 가중치(Weight) 및 바이어스(Bias)를 변화시킨다. 이는 주어진 노드들을 이용하여 목적 패턴을 분류할 수 있는 가중치 및 바이어스를 찾아내는 작업으로, 일반적인 신경망은 각 노드의 가중치들을 조정하기 위해 식 (2)와 같은 경사하강법(Gradient descent)을 이용한다.

$$\theta = \theta - \alpha \frac{\partial}{\partial \theta_j} J(\theta) \quad (2)$$

식 (2)의 θ 는 파라미터(가중치), α 는 학습률 (Learning rate), $(\partial/\partial \theta_j) \times J(\theta)$ 은 미분 값을 의미한다. 즉, 학습 신경망은 가중치 조정으로 미분 값을 사용하며, 이 미분 값 계산을 위해 오류 역전파 알고리즘을 사용한다.

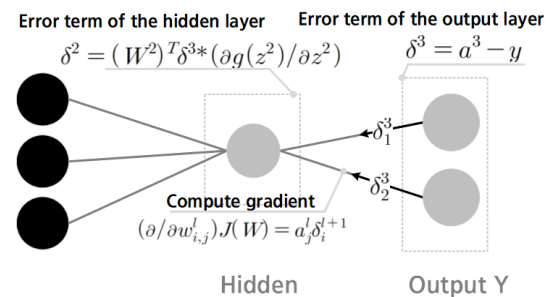


그림 3. Back-propagation 알고리즘의 수행과정
Fig. 3. Process of back-propagation algorithm

그림 3은 오류 역전파 알고리즘에 대한 수행과정을 도식화한 내용이다. 순방향 연산을 통해 도출된 a^3 을 실제 값 y 와 비교하여 오차 δ^3 을 산출한다. 오류 역전파 알고리즘은 산출된 δ^3 을 이전 계층의 각 노드에게 역으로 전파한다. 이전 계층은 식 (2)의 미분 값을 해당 층에서 도출된 a^l 과 역전파 δ^{l+1} 을 통해 해당 층의 오차를 계산한다. 즉, 식 (2)의 미분 값은 해당 층의 산출 결과와 역전파 받는 오차로 대체되어 각 노드의 가중치를 조정한다. 오차는 평균 제곱 오차(MSE, Mean Squared Error) 손실 함수를 사용하여 출력 값과 실제 값의 오차를 제공 후 더하여 계산된다. 이를 통해 산출된 값이 최소인 모델을 제작한다. 하지만 이 함수의 값이 0으로 측정되는 것은 어려운 일로, 반복적인 작업을 요한다. 그로 인해 반복 횟수가 기하급수적으로 증가한다. 그래서 MSE 함수의 값을 측정할 때 임계값 또는 최대 반복 횟수를 지정하여 학습을 수행한다.

2.3 데이터 축소를 위한 특징선택 및 특징추출

데이터 축소 기법(차원축소)을 통해 원본 데이터의 크기 감소 및 분류 성능 향상을 목표로 한다. 이

를 수행하는 기법으로 특징선택(FS, Feature Selection)과 특징추출(FE, Feature Extraction)이 대표적이다 [8][9]. FS은 모든 특징의 부분 집합을 선택하여 간결한 특징 집합을 만드는 것을 목적으로, 분류 정확도를 향상시키기 위해 원본 데이터 내 성능을 높일 수 있는 데이터의 부분 집합(Subset)을 찾아내는 방법이다. 즉, 분류기의 분류 목적에 가장 밀접하게 연관된 특징들만을 추출하여 새로운 데이터를 만든다. FS의 수행과정은 특징 부분 집합 생성, 평가, 정지조건 만족여부로 구분된다. 원본 데이터 내의 부분집합을 추출 후 평가하여 우수성을 측정한다. 측정 결과가 정지조건을 만족할 때까지 이 과정을 반복 수행한다. 조건을 만족하는 부분집합들을 이용하여 분류기의 학습 데이터로 사용한다. FS은 부분 집합을 생성할 때마다 평가 작업으로 인하여 수행 시간이 길다. FE은 원본 데이터 내의 주축을 찾고, 모든 데이터를 해당 축에 투영시킨다[9][10]. 일부 중요 특징을 부분집합으로 빼내는 FS와는 대비된 기법이다. FE을 수행하기 위해 주성분분석(PCA, Principal Component Analysis)를 주로 이용한다. PCA의 핵심 아이디어는 데이터의 분산(Variance)이 가장 큰 방향으로 정사영 시킨다. 즉, PCA는 가장 큰 특징(핵심 특징)들을 추출하는 방식으로 데이터를 축소시키는 방법이다. 두 기법들은 공통적으로 데이터를 축소시키기 위해 모든 특징 중 부분선별 또는 핵심 특징을 추출하고 나머지를 제거시킨다. 하지만 본 논문이 제안하는 데이터(차원) 축소 방식은 윤곽선 작업을 통해 찾을 수 있는 모든 특징을 일련의 패턴으로 함축시키는 방식을 보인다. 즉, 특징을 정리하여 분류하는 방식이다.

III. 제안 알고리즘

본 논문은 데이터의 축소를 위한 방법으로, 전체적인 특징에서 핵심 특징을 선택하는 특징선택과 데이터 분산을 이용한 투영된 데이터 생성기법인 특징추출과 같은 차원 축소 알고리즘이나, 특정 특징만을 활용하여 새로운 데이터를 생성시키는 것이 아닌 전체 특징을 분류 및 정리하여 고정적인 배열 16크기로 정렬시키는 방식을 보인다. 이러한 과정을

통해 인공지능망의 근본적인 문제인 대용량 연산 및 학습시간 등의 문제를 다소 해소하기 위함이다. 또한 본 저자는 데이터 축소 및 학습신경모델에 대한 정확한 경계를 표현하여, 사전처리와 학습 작업을 분산하여 시스템의 부하를 줄일 수 있을 것이라 기대한다. 본 논문은 그 연구의 첫걸음으로 MNIST의 숫자 이미지를 축소 후 ANN 모델을 설계하여 학습시킨다.

3.1 학습 데이터의 축소

ANN 모델의 학습 데이터를 고정적 크기로 변경하는 일련의 과정을 제안한다. 사용한 데이터는 0~9까지의 이미지이며 크기는 28×28이다. 이 이미지들을 1×16 크기로 축소시킨다. 이 데이터들은 1채널로 표현되며, 3 단계로 축소 작업이 수행된다.

3.1.1 윤곽선 검출

로드된 이미지들은 윤곽선 검출 알고리즘을 통해 윤곽선 데이터로 변환한다. 윤곽선 검출 알고리즘은 식 (3)과 같다. 식 (3)의 x 은 원본 이미지를 의미하며, x^1 , x^2 은 원본 이미지를 이용하여 파생된 중간 이미지맵이다. I 와 J 는 원본 이미지의 크기, m 을 median() 함수, n^1 , n^2 은 중간 이미지를 생성 시 사용할 필터의 크기이다.

$$\begin{aligned}
 (A) \quad x_{i,j}^1 &= \sum_{i=0}^{I-n^1+1} \sum_{j=0}^{J-n^2} |m(x[i:i+n^1, j:j+n^2]) - \\
 &\quad m(x[i:i+n^1, j+1:j+1+n^2])| \\
 x_{i,j}^2 &= \sum_{i=0}^{I-n^1} \sum_{j=0}^{J-n^2+1} |m(x[i:i+n^1, j:j+n^2]) - \\
 &\quad m(x[i+1:j+1+n^1, j:j+n^2])| \\
 (B) \quad \delta &= \text{mean}(x^1 + x^2) \\
 (C) \quad x_{edge} &= \text{mean}(x^1[0:I-n^1, 0:J-n^2] + \\
 &\quad x^2[0:I-n^1, 0:J-n^2]) \\
 x_{edge} &= \begin{cases} x_{edge} > \delta, 1.0 \\ x_{edge} < \delta, 0.0 \end{cases}
 \end{aligned} \tag{3}$$

식 3(A)는 원본 이미지에 필터 영역을 적용하여 포함된 영역의 값을 m 을 통해 중간 값을 산출한다. 이 중간 값은 다음 필터 영역에서 계산된 값과 빼기 연산을 수행 후 절댓값으로 표현된다. 필터 영역이 가로 또는 세로 방향으로 이동함에 따라 x^1 , x^2 가 각각 정의된다. 식 3(B)는 x^1 와 x^2 을 합칠 때 사용될 임계값 δ 을 구하는 과정이다. 식 3(C)의 x_{edge} 는 x^1 와 x^2 의 평균계산 후 δ 을 기준으로 1 또는 0으로 결과를 나타내며 그림 4와 같다.

Labels	Original	Edge
0, 1		
4, 8		

그림 4. 윤곽선 검출 결과 이미지
Fig. 4. Outline detection result image

3.1.2 이미지 분할

입력 이미지에 대한 윤곽선 검출은 그림 4와 같은 형식이며, 이러한 이미지를 8×8 크기로 분할한다. 이때 분할된 이미지들은 일정 영역이 겹쳐진 상태로 이미지들이 분할된다. 그림 5는 윤곽선 이미지를 8×8 크기로 분할하는 과정으로 그림 5(a)와 같이 윤곽선 이미지를 8×8 크기로 분할하면서 한 칸씩 이동($stride=1$)하여 전체를 스캔한다. 그림 5(b)는 각 라인별 분할된 데이터이며, 5(c)는 분할된 이미지를 합친 행렬이다.

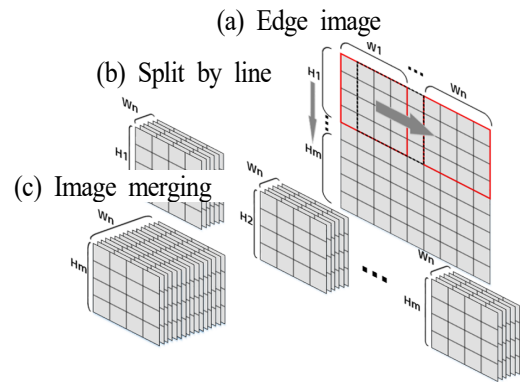


그림 5. 이미지 분할 및 병합과정
Fig. 5. Image split and merge process

3.1.3 이미지 축소 작업 및 형식 변환

위절에서 설명한 작업은 이미지를 간소화하기 위한 준비과정으로, 본 절에서는 간소화 작업 및 형식 변환을 수행한다. 그림 6은 이미지 간소화 작업에 대한 처리 과정으로 6(a)는 8×8 데이터를 4×4 필터로 스캔한다. 이때 필터의 이동 간격은 2($stride=2$)로, 8×8 데이터를 4번 이동하여 특징맵을 생성한다.

4등분된 특징맵은 각 영역에 배치 값을 선정하여 배치 순서에 맞게 특징맵을 정렬 후 그림 6(b)와 같이 각 특정 맵의 최댓값을 검출하여 행렬 x 를 생성한다. x 는 윤곽선 검출 작업으로 인하여 1 또는 0의 값을 지니고 있어 4비트의 2진수와 유사하다. 이러한 x 에 $T[1, 2, 4, 8]$ 을 곱하여 10진수 형태의 데이터로 변환 후 합산하여 Y 를 산출한다. 산출된 Y 는 표 1을 통해 분류한다.

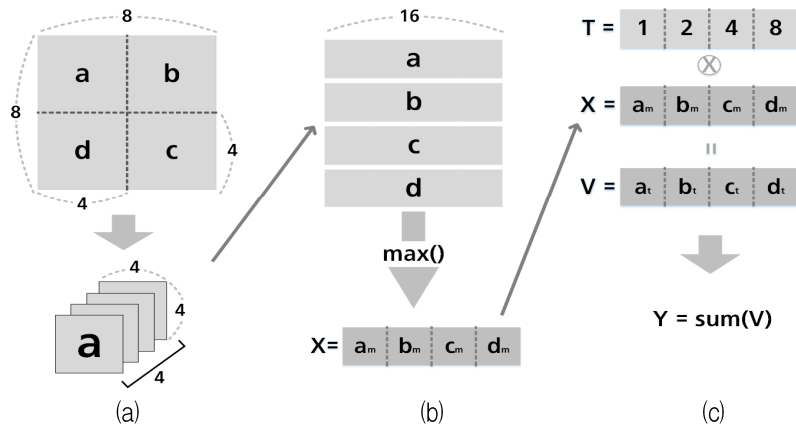


그림 6. 이미지 간소화 작업 및 변환, (a) 이미지 분할, (b) 최댓값 추출, (c) 매핑 및 임계값 지정
Fig. 6. Image streamline operations and transformation, (a) Image division, (b) Calculate the max value of each area, (c) Specify mappings and thresholds

표 1. 활성 영역에 대한 선의 종류
Table 1. Type of line for active area

Binary number		Decimal number		Type of line	
0000	1000	0	8	Blank	Point
0001	1001	1	9	Point	Verticality
0010	1010	2	10	Point	Diagonally
0011	1011	3	11	Horizontality	Curve
0100	1100	4	12	Point	Horizontality
0101	1101	5	13	Diagonally	Curve
0110	1110	6	14	Verticality	Curve
0111	1111	7	15	Curve	Active region

그림 6의 x는 2진수와 같은 형식으로 표현되며, 2진수를 표 1과 같이 선의 종류로 구분한다. 구분된 선은 10진수의 값을 인덱스(index) 값으로 표시하여 16크기의 배열을 구성한다. 즉, 이미지의 크기와 상관없이 0~15 범위 사이의 값으로 표현시킬 수 있다. 이러한 분류 과정을 통해 산출된 데이터는 그림 7과 같다.

그림 7은 y 축의 N*M은 이미지의 너비와 높이의 곱을 나타내며, x축은 인덱스를 의미한다. 그림 7(a)는 그림 6의 Y 값을 분류하여 산출된 그래프이며, 그림 7(b)는 그림 6(b)의 X 평균을 구하여 0.5 미만인 경우 -1을 곱하여 음의 영역을 표현한 것으로, 데이터의 특징을 극대화한다.

위와 같은 처리 과정을 통해 학습 데이터의 크기를 1차원 형식으로 변경시킨 후 ANN 모델을 설계하여 학습한다.

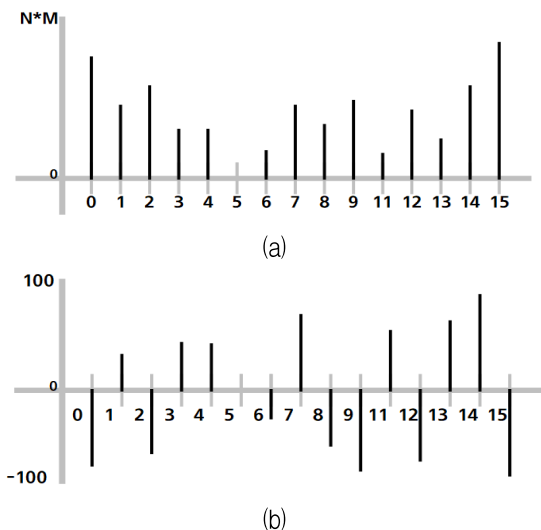


그림 7. 선분 분류에 대한 그래프
Fig. 7. Graph for the sorted segment

3.2 축소 데이터 학습을 위한 ANN모델 설계

본 절은 위에서 설명한 과정으로 축소된 데이터를 학습하기 위해 설계된 학습 신경모델에 대한 내용이다. 그림 8은 설계한 학습 신경모델이다.

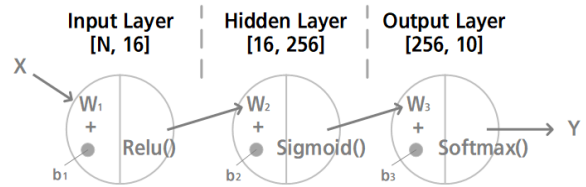


그림 8. 제안한 학습 모델
Fig. 8. Proposed learning model

그림 8은 본 논문이 제안한 축소된 데이터를 학습하기 위한 신경모델이다. 입력층과 하나의 은닉층, 출력층으로 기존의 3층 구조를 따른다. 활성화 함수는 relu() 함수와 sigmoid() 함수를 사용한다. 이는 relu() 함수를 통해 데이터들의 차이를 명확하게 구분하고 sigmoid() 함수로 부각된 특징들을 세밀하게 분류한다. 이를 통해 산출된 값은 softmax() 함수를 통해 결과 Y가 생성된다. 본 모델은 오류 역전과 알고리즘을 적용하여 결과 Y에 대한 오차율이 0.02% 미만으로 측정될 때까지 반복한다.

IV. 실험 및 고찰

본 실험은 제안한 축소 알고리즘을 통해 학습(숫자) 데이터를 배열 16 크기로 변환 후 3.2절에서 설계한 모델을 이용하여 학습한다. 이때 산출된 정확도 및 오차율은 기존 방식의 결과와 미미한 차이를 보인다면 우수하다고 판단한다. 이는 제안된 알고리즘이 학습 신경망에 근본적 문제인 고연산량을 줄이고자하는 것으로, 기존 방식과 유사한 성능(정확도, 오차율)을 보이며 학습시간 단축을 목표로 한다.

표 2. 실험환경
Table 2. Experiment environment

	Specification
CPU	Intel(R) Core(TM) i7-6700 CPU @ 3.40GHz
Memory	16GB
OS	64bit windows
VGA	NVIDIA GeForece GTX 1060 6GB

표 2와 같은 실험환경에서 두 가지의 실험을 진행한다. 첫 번째 실험은 학습 시간 및 오차율 측정이며, 두 번째 실험은 정확도 측정 실험이다.

본 실험들은 ANN의 3계층 모델과 CNN의 LeNet-5 모델을 비교하여 평가하였다. 본 논문이 제안하는 알고리즘은 입력 데이터의 축소로 기존 방식보다 높은 정확도 및 오차율을 측정하는 것이 아닌 고연산량을 줄이기 위해 제안된 것으로 최대 반복 횟수는 1,000회로 지정하며, 기준 오차율은 0.02%로 설정하였다. 본 실험에서 사용한 데이터는 MNIST와 EMNIST, 추가적인 MNIST 형식의 숫자 데이터들이다. 표 3은 실험 데이터셋의 구성이다.

표 3. 실험 데이터의 구성
Table 3. Experimental data

	Train data	Test data
Numbers	109,302	109,999

4.1 학습시간 및 오차율 측정 실험

학습시간 및 오차율을 측정하기 위해 20회 당 1세트로 구성하여, 총 5세트를 수행 후 본 실험의 데이터를 측정하였다.

표 4. 오차율 측정 테이블
Table 4. Error rate measurement table

	1	2	3	4	5
PreAlgo	0.027	0.026	0.026	0.027	0.026
ANN	0.019	0.019	0.019	0.019	0.019
CNN	0.018	0.018	0.018	0.018	0.018

표 4는 오차율 측정 테이블로 제안한 알고리즘(PreAlgo)은 약 0.026이며, 3계층-ANN은 약 0.019, CNN-LeNet5은 약 0.018를 보인다. 이 실험 결과를 고찰할 시 PreAlgo은 약 1.5배 정도의 다소 저조한 차이를 보인다. 하지만 본 논문의 실험 목표 및 축소된 데이터 활용 사항을 고려할 경우 이 정도의 차이는 감안 가능한 사항으로 판단된다. PreAlgo은 배열 16 크기를 이용하여 한 이미지 내에 존재하는 모든 특징을 특징별로 정리되어 이루어진 집합데이터이다. 이를 다시 말하면 특징들의 분류 작업 시 밀집된 특징(특징-몰립 현상)들 사이에 경계선을 그

려야하는 의미가 된다. 그러므로 기존방식과 다르게 적은 특징들의 조합으로 학습을 수행할 때 발생하는 차이와 축소과정에서 특징의 일부 손상 또는 제거로 발생하는 현상으로 오차율이 증가되는 것으로 판단된다.

표 5. 학습시간 측정 테이블
Table 5. Learning time measurement table

	1	2	3	4	5
ProAlgo	0.046	0.046	0.046	0.046	0.046
ANN	0.131	0.132	0.132	0.133	0.132
CNN	4.544	4.520	4.550	4.537	4.527

표 5는 학습시간을 측정한 데이터를 보인다. 본 데이터들을 고찰할 경우 PreAlgo은 0.04초, ANN은 0.13초, CNN은 4.53초로, PreAlgo가 약 3배 이상 빠른 연산을 보인다. 이는 학습 데이터의 크기를 16배 열로 변환하여 기존 모델들 보다 연산속도 및 연산량이 적어 학습이 빠르게 진행된다. 결론적으로 본 알고리즘은 다른 모델들 보다 다소 저조한 오차율을 보이는데 반해 학습시간 및 연산에 있어 다른 모델들보다 높은 성능을 보인다.

4.2 정확도 측정 실험

본 실험은 각 모델을 학습 후 테스트 데이터를 이용하여 정확도를 측정한다. 표 6은 정확도를 측정한 데이터이다.

표 6. 정확도 측정 테이블
Table 6. Accuracy measurement table

Models	PreAlgo	ANN	CNN
Accuracy	0.994	0.997	0.995

표 6을 고찰할 경우 3계층-ANN(0.997)이 가장 높으며 CNN-LeNet5(0.995), PreAlgo(0.994) 순을 확인할 수 있다. 모든 모델이 약 99%의 정확도를 보이나, 본 논문이 제안하는 알고리즘이 가장 낮은 결과를 나타내고 있다. 본 저자는 이를 오차율 측정 실험 때와 같이 “특징-몰립 현상”으로 인하여 “특징-결합 현상”이 발생된 것으로 판단하고 있다. 본 논문이 제안한 데이터 축소 알고리즘은 표 1(3.1.3절)을 통해 이미지 내의 존재하는 특징들을 파악 후

정리하여 단일 배열(16크기)로 변환한다. 이를 다시 표현한다면 일반적인 이미지가 표현할 수 있는 특징의 범위를 감소(다양한 특징이 16가지의 종류로 몰림)시킨 것이다. 즉, “특징-몰림 현상”으로 인하여 일부 서로 다른 특징들이 결합(인접)되는 현상이 발생된 것으로 판단하고 있다. 이 현상으로 인하여 정확도가 낮아지는 것으로 분석하고 있다.

4.3 실험결과 및 기대효과

본 논문은 학습 신경망의 입력 데이터를 줄일 수 있도록 데이터 축소 알고리즘을 제안한다. 이는 딥러닝의 고질적 문제인 고연산량을 다소 줄이기 위해 고안되었다. 축소 알고리즘을 통해 가공된 배열 16크기의 데이터를 학습 신경망의 입력으로 활용하여 학습된 결과와 기존방식으로 학습이 이루어져 도출된 결과를 비교할 시 PreAlgo는 정확도 및 오차율이 기존방식으로 처리된 결과보다 다소 저조한 결과를 보인다. 하지만 이러한 결과는 학습 데이터의 특징 감소로 발생하는 “특징-몰림” 및 “특징-결합” 현상으로 파악된다. 특징-몰림은 복잡한 특징들이 단순한 특징 패턴으로 변화되는 과정으로 인하여 특징 표현범위가 밀집되는 형태를 의미하며, 특징-결합은 특징 표현범위가 밀집됨에 따라 유사한 특징들이 서로 달라붙는 현상을 의미한다. 하지만 특징-몰림 및 특징-결합은 본 저자의 가설로 이 현상들이 아닌 다른 원인으로 오차율 및 정확도가 떨어질 수 있다. 이 현상들을 증명하기 위한 추가적인 연구 및 실험을 진행하고 있다.

본 논문이 제안한 숫자 데이터 축소 알고리즘이 더욱 발전되어 숫자만이 아닌 문자, 필기체 등 다양한 데이터에 따른 성능이 안전적으로 평가된다면 우리는 딥러닝의 고연산량을 다소 감소시킬 수 있다. 또한 본 알고리즘을 모바일 및 소형 단말기 기반 패턴 인식서비스 시스템에 접목한다면 전송되는 데이터 형식을 이미지 압축 기법이 아닌 단순 문자열(배열 16크기의 특징-집합데이터)로 대체할 수 있으며, 전송 데이터양을 줄여 보다 많은 양을 한 번에 묶어 전달할 수 있게 된다. 이러한 데이터를 전송받은 서버 및 PC는 인식서비스로 인한 시스템 부하를 줄일 수 있으며, 하드웨어 자원이 다소 낮을

경우라도 원활한 서비스 제공이 가능해 진다. 또한 점차 깊어지는 모델들을 보다 적은 레이어로 대체 가능할 수 있다고 판단한다.

V. 결론 및 향후 연구

본 논문은 지능형 시스템(음성, 얼굴, 숫자인식 등)에서 일반적으로 사용되는 딥러닝 학습 신경망(ANNs)에 대한 고연산량을 다소 감소시키기 위한 데이터 축소 알고리즘을 제안한다. 본 알고리즘은 기존 신경망 구조에서 사용되는 입력 데이터를 사전적으로 축소시키는 방법을 제안한다. 입력 데이터(숫자 이미지)를 윤곽선 검출 후 결과에 대한 선분을 분석하여 각 선에 대한 개수를 배열 16크기로 정리한다. 다시 말해 배열 16크기의 데이터는 선 종류에 대한 집계 값을 갖는다. 축소된 데이터는 3층-ANN 모델에 대한 학습 데이터로 활용된다. 제안한 축소 알고리즘을 통해 산출된 실험결과와 원본 데이터를 사용한 3층-ANN, CNN-LeNet5의 각 결과와 비교한다. 실험결과 기존 학습방식보다 정확도(99.4%) 및 오차율(0.0264%)이 다소 저조한 결과를 보였지만 그 차이가 미미하다. 하지만 학습시간측정 실험에서는 기존 학습방식보다 3배 이상의 성능을 보였다. 본 논문이 제안한 알고리즘은 학습 데이터의 축소 과정에서 특징의 일부가 제거 및 왜곡으로 인하여 숫자보다 복잡하고 어려운 패턴을 분류할 시 매우 저조한 결과를 보인다. 본 저자는 이러한 문제를 해소하기 위해 지속적으로 연구를 진행할 예정이다.

References

- [1] Kwang-Eun Ko and Kwee-Bo Sim, "Object recognition and detection technology trends using deep learning", Institute of Control, Robotics and Systems, Vol. 23, No. 3, pp. 17-24, Sep. 2017.
- [2] Dae-Ki Kang, "Deep Learning Based Machine Learning Technology Trends", IITP, pp. 12-24, 2016.
- [3] Gasteiger, Johann, and Jure Zupan, "Neural networks in chemistry", *Angewandte Chemie*

International Edition in English, Vol. 32, No. 4, pp. 503-527, Apr. 1993.

- [4] Seung Min Park, Kwee-Bo Sim, and Hong Gi Yeom, "CNN-based Classification of Brain Connectivity using Motor Imagery", Journal of Korean Institute of Intelligent Systems, Vol. 29, No. 2, pp. 124-129, Apr. 2019.
- [5] Lek, Sovan, and Jean-François Guégan, "Artificial neural networks as a tool in ecological modelling, an introduction", Ecological modelling, Vol. 120, No. 2-3, pp. 65-73, Aug. 1999.
- [6] Young Tae Kwak, Young Gik Lee, and Oh Seok Kwon, "Modified Error Back Propagation Algorithm using the Approximating of the Hidden Nodes in Multi-Layer Perceptron", Journal of KISS : Software and Applications, Vol. 28, No. 9, pp. 603-611, Sep. 2001.
- [7] Oh-Jun Kwon and Sung-Yang Bang, "Design of Fault Tolerant Multilayer Perceptron through Selective Augmentation", Journal of KISS(B): Software and Applications, Vol. 24, No. 7, pp. 694-704, Jul. 1997.
- [8] Bon-Hwa Ku, Gwan-Tae Kim, and Jeong-Ki Min, Hanseok Ko, "Deep Convolutional Neural Network with Bottleneck Structure using Raw Seismic Waveform for Earthquake Classification", Journal of the Korea Society of Computer and Information, Vol. 24, No. 1, pp. 33-39, Jan. 2019.
- [9] Li, Jundong, Cheng, Kewei, and Wang, Suhang and Morstatter, Fred and Trevino, Robert and Tang, Jiliang, and Liu, Huan, "Feature Selection: A Data Perspective", ACM Computing Surveys, Vol. 50, No. 6, Article No. 94, Dec. 2017. <https://doi.org/10.1145/3136625>.
- [10] Jeonghwan Kim and Jechang Jeong, "An Efficient Method to Compute a Covariance Matrix of the Non-local Means Algorithm for Image Denoising with the Principal Component Analysis", Journal of Broadcast Engineering, Vol. 21, No. 1, pp. 60-65. Jan. 2016.

저자소개

김 창 민 (Chang-Min Kim)



2014년 2월 : 상지대학교
컴퓨터정보공학부 학사
2016년 2월 : 상지대학교 대학원
컴퓨터공학과 석사
2016년 3월 ~ 현재 : 상지대학교
대학원 박사과정
관심분야 : 영상처리, 프로그래밍

언어, 데이터베이스

박 상 주 (Sang-Ju Park)



2007년 2월 : 상지대학교
컴퓨터정보공학부(공학사)
2009년 2월 : 상지대학교
컴퓨터정보공학부(공학석사)
2012년 2월 : 상지대학교
컴퓨터정보공학부 박사
2000년 3월 ~ 2012년 2월 :

안산대학교 겸임교수

2012년 3월 ~ 현재 : 안산대학교 컴퓨터 정보과 교수

2018년 3월 ~ 현재 : 수원대학교 경영대학원경영과
박사과정

관심분야 : 알고리즘, 영상처리, 프로그래밍 언어, RFID,
RFID안테나, USN 등