

## 농산물 생산성 향상을 위한 딥러닝 기반 농업 의사결정시스템

박진욱\*, 안희학\*\*, 이병관\*\*\*

### The Agriculture Decision - making System(ADS) based on Deep Learning for improving crop productivity

Jinuk Park\*, Heuihak Ahn\*\*, ByungKwan Lee\*\*\*

**요약** 본 논문에서 제안하는 “농산물 생산성 향상을 위한 딥러닝 기반 농업 의사결정 시스템”에서는 정밀농업을 지원하는 농장의 위치 정보를 기반으로 기상 정보를 수집하고, 수집한 기상 정보와 농작물의 실시간 데이터를 이용하여, 작물의 현재 상태를 예측하고 그 결과를 농장 관리인에게 알려준다. 제안하는 시스템은 첫째, 정밀농업을 지원하는 농장의 위치 정보를 기반으로 기상 정보를 수집하는 ICM(Information Collection System)을 설계하고, 둘째, 딥러닝 알고리즘을 기반으로 현재 날씨에 따라 농장 토지의 탄소, 수소, 산소, 질소, 수분 함유량이 재배하고 있는 작물에 적합 특정 작물을 재배하기 좋은 상태인지 판단하는 DRCM(Deep learning based Risk Calculation Module)을 설계하고, 셋째, DRCM의 결과를 기반으로 사용자에게 작물의 상태를 점검할 것을 알려주는 메시지를 전송하는 RNM(Risk Notification Module)을 설계한다. 제안하는 시스템은 기존의 시스템과 비교하였을 때, 데이터양의 증가로 인해 발생하는 정확도 감소 비율이 낮고, 분석 단계에 비지도학습을 적용하기 때문에 안정성을 향상 시킬 수 있다. 결과적으로 농장 데이터 분석 성공률이 약 5.15%가량 향상되었고, 환경 변화에 따른 작물 성장의 위험한 상태정보 다양하게 적용하였을 때, 위험한 상태정보에 대하여 상세하게 추론할 수 있었다. 이는 다양한 내·외부 환경으로부터 발생할 수 있는 작물의 질병을 미연에 예방할 수 있고, 작물이 성장하는데 최적화된 환경을 제공할 수 있는 효과를 나타낸다.

**Abstract** This paper proposes “The Agriculture Decision-making System(ADS) based on Deep Learning for improving crop productivity” that collects weather information based on location supporting precision agriculture, predicts current crop condition by using the collected information and real time crop data, and notifies a farmer of the result. The system works as follows. The ICM(Information Collection Module) collects weather information based on location supporting precision agriculture. The DRCM(Deep learning based Risk Calculation Module) predicts whether the C, H, N and moisture content of soil are appropriate to grow specific crops according to current weather. The RNM(Risk Notification Module) notifies a farmer of the prediction result based on the DRCM. The proposed system improves the stability because it reduces the accuracy reduction rate as the amount of data increases and is apply the unsupervised learning to the analysis stage compared to the existing system. As a result, the simulation result shows that the ADS improved the success rate of data analysis by about 6%. And the ADS predicts the current crop growth condition accurately, prevents in advance the crop diseases in various environments, and provides the optimized condition for growing crops.

**Key Words** : ADS(Agriculture Decision-making System), DRCM(Deep learning based Risk Calculation Module), Deep Learning, ICM(Information Collection Module), soil and weather information

\*Computer Engineering of Graduate School, Catholic Kwandong University

\*\*Corresponding Author : Department of Software, Catholic Kwandong University(hhahn@cku.ac.kr)

\*\*\*Department of Software, Catholic Kwandong University

Received August 21, 2018

Revised August 29, 2018

Accepted October 04, 2018

## 1. 서론

국내 농업은 농업 인구 고령화 및 인력과 농경지 감소 그리고 생산액 비중 감소 등의 문제점으로 인해 국내 농업 활성화를 위한 대책 마련이 시급한 상황이다. 이런 문제점들을 해결하기 위하여 ICT, BT, ET 등 첨단 기술과의 융합이 농업 문제의 해결 방안으로 빠르게 부상하고 있는 추세이며 특히, 고효율화, 고품질 지원이 가능한 IT 기반의 스마트 농업이 기상이변과 농지 감소 및 노동인구 등의 문제해결 방안으로 대두되고 있다. 이러한 추세에 발맞춰 국내 스마트 농업 생산 관련 시장은 2012년 2조 4,295억원에서 연평균 14.5%씩 성장하여 2018년에는 4조 7,474억원 규모가 될 전망이다므로 국내 스마트 팜을 지원할 수 있는 플랫폼 개발이 필요하다 [1].

현재 농업 인구의 감소 및 고령화로 농작업의 생산성과 정밀도를 향상시킬 수 있는 자동화 기술에 대한 요구가 급격히 증가하고 있다. 세계적으로 IoT기반의 농작업의 정보관리 시스템 개발을 통해 최적화된 농장 운용과 농작업 관리시스템에 대한 관심이 높아지고 있다. 그러나 국내의 경우 농업에 대한 하드웨어 플랫폼 제작기술과 제어 알고리즘, 농업 정보처리 및 유무선 통신기술 수준은 미약한 상황으로 선진국의 상용화된 고급형 제품과 경쟁하기 위해서는 관련 기술 개발이 시급하다.

본 논문에서 제안하는 “농산물 생산성 향상을 위한 딥러닝 기반 농업 의사결정 시스템”에서는 정밀 농업을 지원하는 농장의 위치 정보를 기반으로 기상 정보를 수집하고, 수집한 기상 정보와 농작물의 실시간 데이터를 이용하여, 작물의 상태에 대한 세 가지 위험도를 산출한다. 정밀농업 시스템은 최종 산출되는 딥러닝의 output에 따라 농장 관리인에게 알림을 통해 위험 정보를 전송한다. 따라서 본 논문에서 제안하는 기상 정보와 농장 데이터를 활용한 “농산물 생산성 향상을 위한 딥러닝 기반 농업 의사결정 시스템”은 다양한 내·외부 환경으로부터 발생할 수 있는 작물의 질병을 미연에 예방할 수 있고, 작물이 성장하는데 최적화된 환경을 제공할 수 있

는 효과를 나타낸다.

## 2. 관련 연구

### 2.1 스마트팜

클라우드 기반 스마트팜 서비스는 스마트팜 운영에 필요한 각종 기기를 클라우드 상에서 운영하는 방식이다. Stand-Alone방식 스마트팜에서는 온실별로 설치하는 센서, 제어기, 운영PC 등 각종 장치들을 가상화하여 서비스의 형태로 제공한다. 클라우드 기반 스마트팜 서비스는 사용자 관점에서 필요한 기능만을 서비스의 형태로 적용하기 때문에 도입비를 획기적으로 낮출 수 있고, 설치 및 관리가 용이하다[1].

스마트 팜 공장은 이전의 식물공장 개념을 확장하여 식물 생육을 위한 정보의 수집과 관리의 자동화와 시설의 운영이 가능하고, 수집된 정보를 이용한 최적의 작물 생육을 위한 정보 분석의 적용이 가능한 전사적 식물공장 관리 시스템이다[2].

저전력 장거리 무선통신기술(LPWA) 기반으로 원격감시 및 제어가 가능한 자가설치형 농업 자동화 시스템 "씽커넥트-Green" 을 설계 및 구현했다. 영농을 위해서는 물, 햇빛, 흙, 비료, 온도 조절 등이 필요하며 이러한 요소들을 자동화된 시스템을 이용하여 원격 모니터링 및 제어할 수 있다. 비닐하우스부터 화훼단지에 이르기까지 식물의 종류와 재배 환경에 따라 최적화하여 농업 자동화 시스템을 구축할 수 있다. 센서로부터 수집된 정보는 게이트웨이를 통해 서버에 저장되며, 빅데이터를 토대로 스마트폰을 이용하여 최적화된 재배 환경을 설정 및 운영할 수 있다[3].

계층분류 기법은 혼합 픽셀의 효과를 줄이고 분류 성능을 향상시키도록 입력 자료를 계층별로 정의하여 분류하는 방법으로 전북 김제시의 동계작물을 대상으로 Landsat-8 위성영상을 이용하여 스마트 팜 지도를 사용하여 농경지를 분류하였고, 무인기 영상의 훈련자료를 사용하여 밀과 보리, 청보리, IRG, 혼파 재배지로 분류하였으며, 제안된 분류방법이 작물 구분도 작성에 효과적이다[4].

IoT에 기반을 둔 축산사료의 측정 장치를 개발하여 자동 주문과 배송 시스템을 구현하고, 관련 영역의 효율성을 극대화하는 방안을 구현했다 [5].

THI 지수를 계산하여 번식 축사 환경에 적용하기 위해 무선 센서 네트워크와 농장 환경 자동 제어 시스템을 구현했다. 축산물 온도, 심박수 및 운동량과 같은 환경 정보를 수집한다. 또한 THI 지수를 계산하기 위해 온도, 습도, 조도 등 농장 환경 데이터를 수집한다. 그리고 수집된 데이터를 이용하여 자동으로 최상의 상태를 유지할 수 있도록 지능형 농장의 지붕 개방 및 전동 팬을 정확하게 제어한다[6].

## 2.2 딥러닝(Deep Learning)

기계학습 기술인 딥러닝을 적용한 질병개체 파악 방안은 정상상태와 질병상태의 가축들이 섞여있는 환경에서 심층신경망을 이용하여 가축의 상태를 분류하고, 딥러닝은 가축의 생체데이터 통계적 특성을 알지 못하는 상황에서 학습을 통하여 가축의 상태를 정확하게 구분할 수 있다[7].

DBN(Deep Belief Network)은 비지도 사전 학습을 통하여 초기 가중치를 설정한 후 기존의 인공신경망에 대한 문제점을 보완한다. 예측인자로는 기온, 강수일, 태양과 달 궤도 관련 자료를 선정했다. 전체 기간에서 일부를 학습 자료로 이용하여 예측모델을 생성하고 나머지는 생성된 모델의 검증 자료로 사용했다. 검증 결과로 획득한 예측 값 등은 확률 값을 갖고 임계값을 사용하여 강수유무를 판별했다. 강수 정확도의 척도로 양분예보기법 중 Bias(frequency bias) CSI(Critical Successive Index)를 계산했다[8].

CNN(Convolution Neural Network)은 학습을 통하여 분류기준에 따라 커널의 속성을 결정하고 최적의 특징영상을 추출하여 화소를 구분한다. CNN을 구성하여 기존의 영상처리 기법으로 해결이 어려웠던 분광특성이 유사한 물질간의 분류 및 GIS속성정보에 따른 분류를 수행하였으며, 항공초분광센서인 AISA(Airborne Imaging Spectrometer for Application)와 CASI(Compact Airborne Spectrographic Imaging)로 얻은 영상을 사용했다. 실험대상지역은 세 곳이고 사이트 1과 사이트 2

는 벼, 양파, 감자 등의 다양한 농작물을 포함하며 사이트 3은 단독주거시설과 공동주거시설 등으로 구성된 건물을 포함한다[9].

센서에서 발생하는 erratic fault와 spike fault, hard-over fault, drift fault 그리고 stuck fault 등의 고장유형에 대하여 기계학습 알고리즘인 SVM(Support Vector Machine)과 CNN을 적용하여 검출하고 분류했다. SVM의 학습 및 테스트를 위해 데이터 샘플들로부터 시간영역 통계 특징들을 추출하고 최적의 특징을 찾기 위해 유전 알고리즘(genetic algorithm)을 적용했다. 센서 고장을 분류하기 위하여 multi-layer SVM을 구성하여 multi-class를 이용했다. 데이터 샘플들을 사용하여 CNN을 학습시키고 앙상블 기법을 적용하여 성능을 높인다[10].

학습 및 생성 모델을 단순화함으로써, 계산을 감소시키고 중간 프레임의 변경된 영역만을 생성하는 견고한 모델을 구성할 필요가 있다. 변화된 영역에서 픽셀을 생성하는 단순화된 모델을 구현하기 위해 CNN과 RNN(Recurrent Neural Network)을 기반으로 동작 특징 추출 및 학습 모델을 구성했다. 또한 모션이 발생하는 영역에 대해 디컨 볼루션 계층을 통해 중간 프레임을 생성하기 위해 모션 확률 맵을 사용한 알고리즘을 사용한다[11].

기후 변화로 인한 심각한 피해를 처리하기 위해 고도의 정확도를 가진 심층 학습에 기반한 강수 예측 알고리즘을 이용한다. 한국의 지리적 특성과 계절적 특성이 명확히 구별되기 때문에 기상 요인은 시계열에서 반복적인 패턴을 갖는다. LSTM (Long Short-Term Memory)은 연속 데이터에 대한 강력한 알고리즘이기 때문에 이 연구에서 강수량 예측에 사용되었다. 수치 테스트를 위해 GNSS(Global Navigation Satellite System) 신호의 대류권 지연을 기반으로 PWV (Precipitable Water Vapour)를 계산한 다음 깊은 학습 기법을 강수 예측에 적용했다 [12].

### 3. 농산물 생산성 향상을 위한 딥러닝 기반 농업 의사결정 시스템

#### 3.1 전체 구성

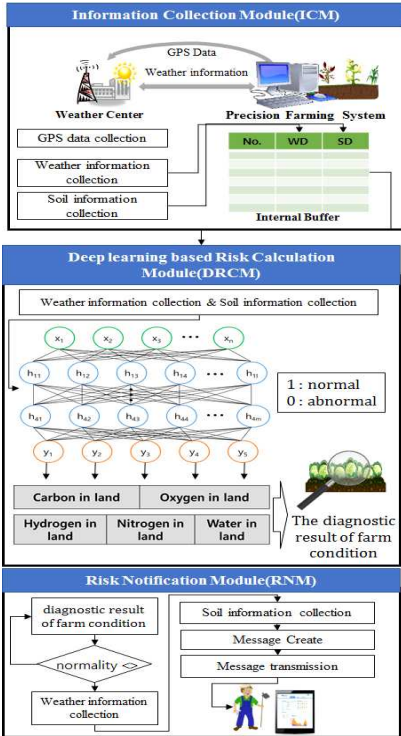


그림 1. 농산물 생산성 향상을 위한 딥러닝 기반 농업 의사결정 시스템  
 Fig. 1. The Agriculture Decision-making System(ADS) based on Deep Learning for improving crop productivity

본 논문에서 제안하는 “농산물 생산성 향상을 위한 딥러닝 기반 농업 의사 결정 시스템”에서는

첫째, 정밀농업을 지원하는 농장의 위치정보를 기반으로 기상 정보를 수집하는 ICM(Information Collection Module)을 설계하고,

둘째, 딥러닝 알고리즘을 기반으로 현재 날씨에 따라 농장 토지의 탄소, 수소, 산소, 질소, 수분 함유량이 재배하고 있는 작물에 적합 특정 작물을 재배하기 좋은 상태인지 판단하는 DRCM(Deep learning based Risk Calculation Module)을 설계하고,

셋째, DRCM의 결과를 기반으로 사용자에게 작물

의 상태를 점검할 것을 알려주는 메시지를 전송하는 RNM(Risk Notification Module)을 설계한다.

따라서 본 논문에서 제안하는 “농산물 생산성 향상을 위한 딥러닝 기반 농업 의사결정 시스템”은 다양한 내·외부 환경으로부터 발생할 수 있는 작물의 질병을 미연에 예방할 수 있고, 작물이 성장하는데 최적화된 환경을 제공할 수 있는 효과를 나타낸다.

#### 3.2 ICM(Information Collection Module) 설계

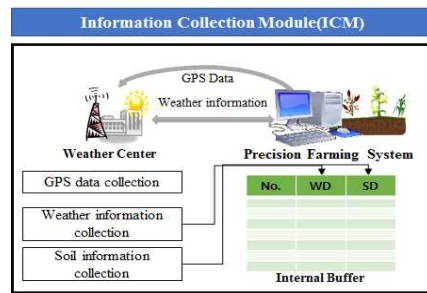


그림 2. 정보 수집 모듈  
 Fig. 2. ICM(Information Collection Module)

본 논문에서 제안하는 정보 수집 모듈에서는 정밀 농업 시스템의 위치 정보를 기반으로 기상청으로부터 날씨 정보를 수집하고, 농장의 토지에 설치된 센서들로부터 실시간 다량원소 데이터를 수집한다. 정보 수집 모듈의 동작은 다음과 같다.

첫째, 정밀농업 시스템은 농장 내부 중앙 서버에 설치되며, 정밀농업 시스템의 ICM은 GPS를 통해 자신의 위치 정보를 수집한다.

둘째, ICM은 자신의 위치정보를 기상청 서버에 전송하여 현재 위치에 대한 온도, 습도, 일조량, 풍량, 풍속 등의 기상 정보를 수집하여 내부 데이터 버퍼에 기록한다.

셋째, 작물이 심어진 토양의 센서들로부터 실시간 다량원소 데이터를 수집하여 내부 데이터 버퍼에 기록한다.

넷째, 내부 데이터 버퍼에 수집된 기상정보 및 토양 정보를 DRCM에 전달한다.

### 3.3 DRCM(Deep learning based Risk Calculation Module) 설계

본 논문에서 제안하는 DRCM은 DBN을 기반으로 하는 신경망 모델이며, 클라우드에서 동작한다. DRCM는 현재 날씨에 따라 농장의 상태가 특정 작물을 재배하기 좋은 상태인지 판단한다. 즉, DRCM는 작물별로 하나씩 생성된다.

#### 3.3.1 DRCM의 입력과 출력

DRCM의 입력으로 현재 기상상태와 토지의 상태가 사용된다. 기상상태는 현재 기온, 풍속, 일조량 강수량의 수치이다. 토지의 상태는 토지의 탄소, 수소, 산소, 질소와 수분 함유량이다. 기상 정보의 경우, 농장의 GPS를 기반으로 국가에서 제공하는 날씨 정보를 사용한다. 토지 상태의 경우 농장의 토지 내에 센서를 배치하여 그 센서들이 측정된 데이터를 클라우드로 전송하도록 한다.

DRCM의 출력은 현재 토지 상태가 재배 중인 작물에 적합한지를 판단할 수 있는 값이다. DRCM은 출력 레이어 노드의 개수를 5개로 설정한다. 출력 레이어의 각 노드는 현재 토지의 탄소, 수소, 산소, 질소, 수분 함유량이 재배하고 있는 작물에 적합한지를 알려준다. 입력과 출력에 대한 표가 그림3에 표현되어 있다. 그림 3의 output node 인 y1~y5는 각각 토지의 탄소, 산소, 수소, 질소, 수분 함유량이 적절한지를 나타낸다. 만약 노드의 출력이 1이라면 해당 항목은 정상이라는 뜻이며, 0이라면 해당 항목은 정상이 아니라는 뜻이다.

#### 3.3.2 DRCM의 신경망 구조

DRCM은 단층 신경망인 RBM(Restricted Boltzmann machine)을 여러 층으로 쌓은 딥러닝 모델인 DBN을 기반으로 동작한다. DBN은 입력 레이어 노드의 수와 출력 레이어 노드의 수가 같으면 비지도 학습이 가능하다. 그러나 DRCM은 입력 데이터와 출력 데이터의 개수와 형식이 다르기 때문에, 입력 레이어 부터 마지막 히든레이어 까지는 RBM기반 비지도 학습을 실행하고, 마지막 레이어와 출력 레이어 사이

에는 지도 학습 기반인 Back-propagation 방식을 사용하여 학습을 실행한다.

Case	Node output				
	y <sub>1</sub> Carbon in land	y <sub>2</sub> Oxygen in land	y <sub>3</sub> Hydrogen in land	y <sub>4</sub> Nitrogen in land	y <sub>5</sub> Water in land
1	1	1	1	1	1
2	1	1	1	1	0
3	1	1	1	0	1
4	1	1	0	1	1
5	1	0	1	1	1
6	0	1	1	1	1
7	1	1	1	0	0
8	1	1	0	0	1
...	...	...	...	...	...
32	0	0	0	0	0

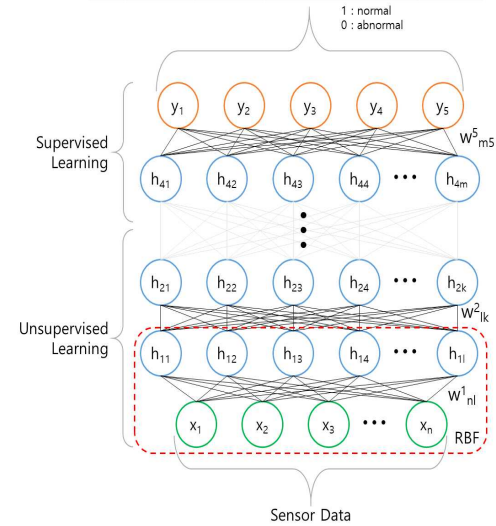


그림 3. DRCM의 구성도  
Fig. 3. The Construction of the DRCM

히든 레이어의 개수는 4개로 설정한다. 입력과 출력의 노드 수가 크지 않기 때문에 DRCM은 4개의 히든 레이어만으로 현재 기상상태와 토지 상태의 연관성을 계산할 수 있다. DRCM의 전체 구성은 그림 3과 같다.

각 뉴런이 하는 계산은 퍼셉트론과 유사하다. 즉, 입력 값의 가중치 합을 계산하여 이를 시그모이드 함수를 통하여 변환한다. 예를 들어 그림 3에서 h1노드들의 출력 값은 식(1)로 계산된다.

$$P(h_{1i}|x) = \sigma(\text{net}_{h_{1i}}) \text{-----}(1)$$

여기서,  $\sigma(\text{net}_{h_{1l}})$ 은 인공신경망에서 활성화 함수로 사용되는 시그모이드 함수로 식(2)와 같이 정의된다.

$$\sigma(\text{net}_{h_{1l}}) = \frac{1}{1 + \exp(-\text{net}_{h_{1l}})} \text{-----}(2)$$

여기서,  $\text{net}_{h_{1l}}$ 은 입력층  $x$ 의 가중치 합을 뜻한다.  $\text{net}_{h_{1l}}$ 은 식(3)을 통하여 계산된다.

$$\text{net}_{h_{1l}} = \sum_i^n w^{1i} x_i \text{-----}(3)$$

RBM기반 학습은 비지도 학습이기 때문에  $h_1$ 에서  $x$ 로 다시  $\text{net}$ 값을 계산한  $x'$  값을 활용하여 학습을 진행한다.

### 3.3.3 DRCM의 학습

DRCM은 입력 레이어부터 마지막 히든 레이어까지 아래 방법을 통하여 학습을 진행한다.

첫째, DRCM은 모든 첫 번째 히든 레이어의 값인  $P(h_{1l}|x)$ 를 계산한다.

둘째, DRCM은  $P(h_{1l}|x)$ 값과 입력값을  $\langle x_n, h_{1l} \rangle_0$ 로 만든다.

셋째, DRCM은 입력 레이어의 값인  $P(x_n|h_1)$ 를 계산하고,  $P(x_n|h_1)$ 를 바탕으로 다시  $P(h_{1l}|P(x_n|h_1))$ 를 계산한다.

넷째, DRCM은 기존 입력 값과  $P(x_n|h_1)$ 를 비교하고,  $P(h_{1l}|x)$ 과  $P(h_{1l}|P(x_n|h_1))$ 를 비교하여 일치하는 값들을  $\langle x_n, h_{1l} \rangle_1$ 로 만든다.

다섯째, 모든  $n$ 과  $l$ 쌍에 대하여 식 (4)와 같이 입력층과 첫 번째 히든레이어 사이의 연결 강도 변화량이 계산된다.

$$\Delta w^{1ni} = \alpha (\langle x_n, h_{1l} \rangle_0 - \langle x_n, h_{1l} \rangle_1) \text{---}(4)$$

여기서,  $\alpha$ 는 개발자가 미리 지정한 학습율이며, 연결 강도는 식(5)를 이용하여 수정된다.

$$\text{new}_w^{1ni} = \text{old}_w^{1ni} + \Delta w^{1ni} \text{-----}(5)$$

첫 번째 히든 레이어부터 마지막 히든레이어 까지 학습은 첫째~다섯째 단계를 반복하여 진행된다.

다음으로 DRCM은 마지막 히든 레이어와 출력 레이어 사이의 학습을 아래와 같이 진행한다.

첫째, 출력 레이어의 결과  $P(y_j|h_4)$ 을 계산하고,  $P(y_j|h_4)$ 을  $\hat{y}_j$ 라고 가정한다.

둘째,  $\hat{y}_j$ 와 목표치  $d$ 의 차이를 통하여 오차  $E$ 을 계산하면  $E$ 는 식 (6)과 같다.

$$E = \frac{1}{2} \sum_{i=1}^j (d - \hat{y}_i)^2 \text{-----}(6)$$

셋째, 식 (7)을 통하여 오차 신호를 구한다.

$$\delta_y = (d - y)y(1 - y) \text{-----}(7)$$

넷째, 오차 신호  $\delta_y$ 를 이용하여 마지막 히든 레이어와 출력 레이어 사이의 연결 강도를 수정한다.

$$\text{new}_w^{5mj} = \text{old}_w^{5mj} + \alpha \delta_y h_{4m} \text{-----}(8)$$

다섯째, 오차  $E$ 가  $E_{\text{max}}$ 보다 작아질 때까지 둘째부터 넷째 단계를 반복한다. 만약  $E$ 가  $E_{\text{max}}$ 보다 작아지면 학습을 종료한다.

Algorithm 1은 DRCM의 학습 절차에 대한 알고리즘이다.

Algorithm 1. DRCM learning method
1 Input = Training samples {x1, y1}, ..., {xN, yN};
2 (xi is sensor data, yi is reslut of xi)
3 Run:
4 Initialize sample weight w;
5 n is the number of input layer nodes
6 m is the number of hidden layer nodes
7
8 for (int k=1; k<=m; k++)
9 calculation p(h1k x);
10 end for
11 for (int k=1; k<=n; k++)
12 calculation p(xk h1);
13 end for

```

14 calculation <xnh1m>0;
15 for (int k=1; k<=m; k++)
16 calculation p(h1k|p(xn|h1));
17 end for
18 calculation <xnh1m>1;
19  $\Delta w^{1mn} = \alpha(\langle xnh1m \rangle_0 - \langle xnh1m \rangle_1)$ ;
20  $new\_w^{1mn} = old\_w^{1mn} + \Delta w^{1mn}$ ;
21
22 for (int j=2; j<=4; j++)
23 for (int k=1; k<=m; k++)
24 calculation p(h(j+1)k|hj);
25 end for
26 for (int k=1; k<=n; k++)
27 calculation p(hjk|h(j+1));
28 end for
29 calculation <hjmh(j+1)m>0;
30 for (int k=1; k<=m; k++)
31 calculation p(h(j+1)k|p(hjk|h(j+1)));
32 end for
33 calculation <hjmh(j+1)m>1;
34  $\Delta w^{jmn} = \alpha(\langle hjmh(j+1)m \rangle_0 - \langle hjmh(j+1)m \rangle_1)$ ;
35  $new\_w^{jmn} = old\_w^{jmn} + \Delta w^{jmn}$ ;
36 end for
37
38 while (E>Emax)
39 for(k=1; k<=5; k++)
40  $\hat{y}_k = P(y_k|h_4)$ ;
41 end for
42  $E = \frac{1}{2} \sum_{k=1}^5 (d - \hat{y}_k)^2$ ;
43 if (E>Emax)
44 for (k=1; k<=5; k++)
45  $\delta_{y_k} = (d - \hat{y}_k) \hat{y}_k (1 - \hat{y}_k)$ ;
46  $new\_w^{5mk} = w^{5mk} + \alpha \delta_{y_k} h_{km}$ ;
47 end for
48 else
49 break;
50 end if
51 end while
    
```

알고리즘은 크게 네 부분으로 나누어진다. 네 부분은 아래와 같은 절차에 따라 진행된다.

1) 데이터 입력과 변수 초기화 : DRCM은 학습을 위하여 결과를 알고 있는 훈련 샘플들을 입력받고 노드들의 수와 노드들 사이의 가중치를 초기화한다.

2) 입력 레이어와 첫 번째 히든레이어 사이의 학습: DRCM은 첫 번째 히든 레이어의 결과를 계산하고 입력 레이어와 첫 번째 히든레이어 사이의 연결 강도를 수정한다. 이는 알고리즘의 8번 줄부터 20번 줄에 해당한다.

3) 두 번째부터 네 번째 히든레이어 사이의 학습: DRCM은 두 번째 히든 레이어 부터 네 번째 히든레이어 사이의 연결 강도를 수정한다. 방식은 입력 레이어와 첫 번째 히든 레이어 사이의 연결 강도를 수정할 때와 같다. 이는 알고리즘의 22번 줄부터 36번 줄에 해당한다. 입력 레이어부터 네 번째 레이어까지는 RBM기반의 학습법을 사용하기 때문에, DRCM은 비지도 학습을 실시한다.

4) 네 번째 히든레이어와 출력 레이어 사이의 연결 강도 수정: DRCM은 네 번째 히든 레이어와 출력 레이어 사이의 연결 강도를 수정한다. 그러나 출력 레이어와 마지막 히든 레이어는 RBM을 사용한 비지도 학습을 실행할 수 없기 때문에 Back-propagation 학습법과 훈련 샘플들을 사용하여 연결 강도를 수정한다.

해당 알고리즘을 사용하여 신경망 모델의 학습이 완료되면 농장이나 인터넷에서 수신한 센서와 날씨 데이터를 이용하여 현재 농장 토양의 상태를 점검할 수 있다.

### 3.4 RNM(Risk Notification Module) 설계

본 논문에서 제안하는 RNM에서는 DRCM의 결과를 기반으로 사용자에게 작물의 상태를 점검할 것을 알려주는 메시지를 전송한다.

작물의 실시간 상태를 수집하여 DRCM으로 분석한 결과가 모두 정상인 상태와 다를 경우 메시지 생성을 시작한다. 이때 생성되는 메시지의 내용은 현재 실시간 기상정보 및 토양의 다량원소 수치 데이터와 위험도 추론 결과 데이터로 구성된다.

메시지는 작물의 토지 및 온도에 대한 점검 메시지이며, Wi-fi와 3G/4G 기반으로 사용자의 스마트폰에 메시지를 전송한다. 사용자는 메시지를 통해 현재 기상정보와 토지의 상태정보 그리고 추론 결과를 기반으로 작물이 질병이 들거나 성장에 방해 받기 전에 적합한 조치를 취할 수 있는 효과를 갖는다.

### 4. 성능 분석

본 논문에서 제안하는 DRCM의 성능분석을 위하여 농장의 상태를 판단하는 모델을 DRCM 학습 방식, LSTM 학습 방식 및 CNN 학습 방식으로 구성된 각 신경망 모델의 학습속도와 신뢰성을 비교하였다. 실험은 GeForce GTX 1080 CPU를 사용하는 서버 PC와 센서 데이터를 보내는 클라이언트 PC를 사용하여 진행하였다. 각각 알고리즘의 초기 연결 강도, 학습율, 학습율의 감소율 등 학습 초기에 개발자가 설정해야 하는 값은 통제 변수로 설정한다. 표 1은 시뮬레이션 통제 변수와 값을 나타낸다.

표 1. 시뮬레이션을 위한 통제변수와 값  
Table 1. Control variables for simulation

control variable	value
Number of Hidden Layer	4
Numder of each hidden layer's Node	15
Learning Rate	0.1
weight(all)	0.2

첫 번째 실험은 Training Sample에 따른 신경망 모델의 학습시간을 비교하였고, 그 결과는 그림 5와 같다.

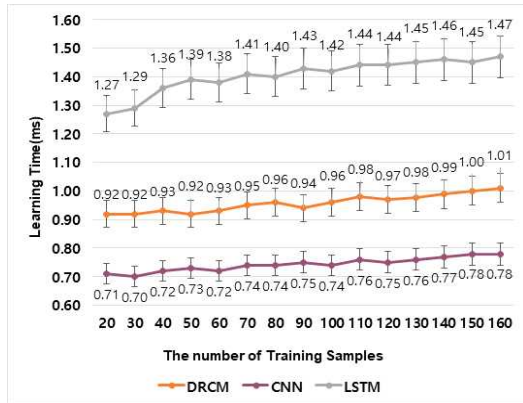


그림 5. Training Sample에 대한 학습시간에 대한 그래프  
Fig. 5. The Graph for Learning Time according to Training Samples

그림 5에서, DRCM은 Training Sample이 20개일 경우 0.92, 80개일 경우 0.96, 120개일 경우는 0.97,

150개일 경우 1.00으로 나타났다. DRCM의 평균 학습시간은 0.96이다.

CNN은 Training Sample이 20개일 경우 0.71, 80개일 경우 0.74, 120개일 경우는 0.75, 150개일 경우 0.78로 나타났다. CNN의 평균 학습시간은 0.74다.

LSTM은 Training Sample이 20개일 경우 1.27, 80개일 경우 1.4, 120개일 경우는 1.44, 150개일 경우 1.45로 나타났다. LSTM의 평균 학습시간은 1.4이다.

결과적으로 DRCM의 학습시간은 LSTM보다 약 0.44초 빠르고, CNN보다 약 0.23초 느리다. 그러나 두 번째 실험을 통하여 CNN보다 DRCM이 농장 진단에 더 적합하다는 것이 확인되었다.

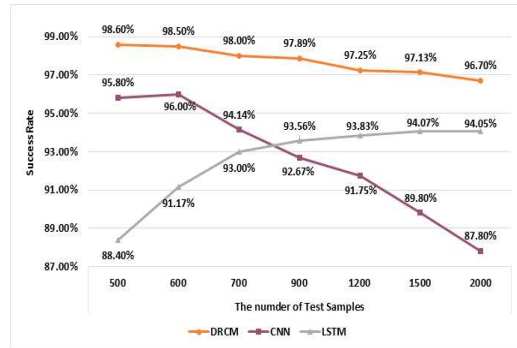


그림 6. Test Sample에 대한 진단 성공률 그래프  
Fig. 6. The Graph for Diagnosis Success Rate according to Test samples

두 번째 실험은 Test Sample에 대한 진단 성공률을 비교하였고, 그 결과를 나타내면 그림 6과 같다.

그림 6을 살펴보면, DRCM기반 신경망 모델은 500개 sample을 테스트할 경우 98.6%, 700개 sample을 테스트할 경우 98%, 1500개 sample을 테스트할 경우 97.13%, 2000개 sample을 테스트할 경우 96.7%의 성공률을 가졌다. CNN기반 신경망 모델은 500개 sample을 테스트할 경우 95.8%, 700개 sample을 테스트할 경우 94.14%, 1500개 sample을 테스트할 경우 89.8%, 2000개 sample을 테스트할 경우 87.8%의 성공률을 가졌다. LSTM 기반 신경망 모델은 500개 sample을 테스트할 경우 88.40%, 700개 sample을 테스트할 경우 93%, 1500개 sample을 테



스트할 경우 94.07%, 2000개 sample을 테스트할 경우 94.05%의 성공률을 가졌다.

DRCM은 평균적으로 CNN보다 5.16%, LSTM보다 5.14% 높은 성공률을 기록하였다. LSTM의 경우 sample의 수가 증가할수록 성공률도 증가했으나, DRCM과 비교하여 성공률이 너무 낮다.

결과적으로 실험을 통하여 제안하는 DRCM은 기존 LSTM보다 학습 속도가 빠르고, CNN보다 높은 성공률을 갖는 것을 알 수 있다.

## 5. 결론

본 논문에서는 다양한 내·외부 환경으로부터 발생할 수 있는 작물의 질병을 미연에 예방할 수 있고, 작물이 성장하는데 최적화된 환경을 제공할 수 있는 "농산물 생산성 향상을 위한 딥러닝 기반 농업 의사결정 시스템"을 제안하였다. 제안하는 정밀농업 시스템의 기대효과는 다음과 같다.

첫째, 작물의 기본 성장 요소를 고려하는 위험도 추론 모듈을 통해 다양한 작물들의 질병을 미연에 예방할 수 있다.

둘째, 실내,외 환경을 고려하여 작물의 최적화된 성장 환경을 제공해줄 수 있다.

셋째, 토지의 다량 원소뿐만 아니라 미량원소까지 고려하여 작물의 성장에 필수 영양소 성분 함량을 추론하는데 활용될 수 있다.

넷째, ADS의 DRCM은 기존 LSTM과 비교하였을 때, 데이터양의 증가로 인해 발생하는 정확도 감소 비율이 낮고, 분석 단계에 비지도학습을 적용하기 때문에 안정성을 향상시킬 수 있다.

다섯째, DRCM은 LSTM보다 약 0.44초 빠른 학습 시간을 갖는다.

여섯째, DRCM은 평균적으로 CNN보다 5.16%, LSTM보다 5.14% 높은 성공률을 갖는다.

## REFERENCES

[1] Small and Medium Business Administration, "Technology Roadmap for SME", 27, Small and Medium Business Administration, 2017

- [2] Seyong Lee, "Cloud-based smart farm technology", Information & communications magazine, Vol.34, No.1, pp.51-57, 2016.
- [3] Sung-Il Hwang, Jong-Moon Joo, Seong-Yong Joo, "ICT-Based Smart Farm Factory Systems through the Case of Hydroponic Ginseng Plant Factory", The Journal of Korean Institute of Communications and Information Sciences, Vol.40, No.4, pp.780-790, 2015.
- [4] JaeGu Back, Hying-Woo Lee, "Design and Implementation of Self-installing Agricultural Automation System for Remote Monitoring and Control Based on LPWA Technology", Journal of The Korea Internet of Things Society, Vol.3, No.1, pp.13-19, 2017.
- [5] Sang-il Na, Chan-won Park, Kyu-ho So, Jae-moon Park, Kyung-do Lee, "Satellite Imagery based Winter Crop Classification Mapping using Hierarchical Classification", Korean journal of remote sensing, Vol.33, No.5, pp.677-687, 2017.
- [6] Wonyoung An, YunHi Chang, "A Study on the Livestock Feed Measuring Sensor and Supply Management System Implementation based on the IoT", Journal of Korea institute of information, electronics, and communication technology, Vol.10, No.5, pp.442-454, 2017.
- [7] Yongju Park, Jun Moon, "Smart Dairy Management System Development Using Biometric/Environmental Sensors and Farm Control Gateway", IEMEK Journal of embedded systems and applications, Vol.11, No.1, pp.15-20, 2016.
- [8] Woongsup Lee, Seong Hwan Kim, Jongyeol Ryu, Tae-Won Ban, "Fast Detection of Disease in Livestock based on Deep Learning", Journal of the Korea Institute of Information and Communication Engineering, Vol.21, No.5, pp.1009-1015, 2017.
- [9] Ji-Hun Ha, Yong Hee Lee, Yong-Hyuk Kim, "Forecasting the Precipitation of the Next Day Using Deep Learning", Journal of Korean institute of intelligent systems, Vol.26, No.2, pp.93-98, 2016.
- [10] Ahram Song, Yongil Kim, "Deep Learning-based Hyperspectral Image Classification with Application to Environmental Geographic Information Systems", Korean Journal of Remote Sensing, Vol.33, No.6, pp.1061-1073, 2017.
- [11] Jae-Wan Yang, Young-Doo Lee, In-Soo Koo, "Sensor Fault Detection Scheme based on Deep Learning and Support Vector Machine", The journal of the Institute of Internet Broadcasting and Communication, Vol.18, No.2, pp.185-195, 2018.
- [12] Soo-Hwan Lee, "Image reconstruction techn

que using deep learning architecture”, Journal of the Korean Society of Marine Engineering, 42권, 2호, pp.121-126, 2018.

[13] Kim Hee-Un, Bae Tae-Suk, “Preliminary Study of Deep Learning-based Precipitation”, Journal of the Korean Society of Surveying, Geodesy, Photogrammetry and Cartography, Vol.35, No.5, pp.423-430, 2017

---

## 저자약력

---

### 박진욱(Jimuk Park)

[정회원]



- 2013년 2월 : 가톨릭관동대학교 체육교육과(체육학사)
  - 2015년 8월 : 가톨릭관동대학교 의료경영학과(의료경영학석사)
  - 2018년 10월 현재 : 가톨릭관동대학교 컴퓨터공학과 박사과정
- 헬스케어, 빅데이터, 네트워크보안, 사물인터넷

<관심분야>

### 안희학(Heuihak Ahn)

[정회원]



- 1981년 2월 : 숭실대학교 전자계산학과(공학사)
  - 1983년 2월 : 숭실대학교 전자계산학과(공학석사)
  - 1994년 8월 : 숭실대학교 전자계산학과(공학박사)
  - 1984년 4월 ~ 현재 : 가톨릭관동대학교 소프트웨어학과 교수
- 컴파일러, 프로그래밍 언어, 멀티미디어, 컴퓨터보안

<관심분야>

### 이병관(ByungKwan Lee)

[정회원]



- 1979년 2월 : 부산대학교 기계설계학과(공학사)
  - 1986년 2월 : 중앙대학교 전자계산학과(이학석사)
  - 1990년 2월 : 중앙대학교 전자계산학과(공학박사)
  - 1988년 3월 ~ 현재 : 가톨릭관동대학교 소프트웨어학과 교수
- 네트워크보안, 빅데이터, 데이터마이닝, 사물인터넷

<관심분야>