



A Study of Heart Disease Prediction Using Multilayer Perceptron based on Deep Learning

Chul-Jin Kim¹, Ji-Sung Kim²

¹*Department of Computer Systems and Engineering, Inha Technical College*

²*Department of Computer Science, HanYang University*

ABSTRACT

Deep Learning, an advanced technology for machine learning, is currently being applied to various industrial fields and is being actively applied and developed in the fields of finance, e-commerce, and Internet of Thing, which require prediction. In particular, this study will be applied to the prediction of heart disease in the medical field, and it will provide the advantage of early detection of heart disease or saving medical treatment cost such as angiography for non-heart disease patient. To apply the Multilayer Perceptron machine learning technique for Deep Learning, 2 Hidden Layers and 10 Perceptrons can be constructed to improve learning accuracy. By using Back Propagation during learning, it is reversed from the Output Layer to the Hidden Layer so as to reduce the error. The error is reduced by using ReLU or Sigmoid function, and the generated prediction model optimizes the model through the Adam optimization function. In a case study, heart disease data is learned using heart disease diagnosis and presence or absence of heart disease provided by the Machine Learning & Artificial Intelligence System Center at the University of California, Irvine. The Deep Learning module is developed using a Python-based TensorFlow and validated using randomly extracted data samples from heart disease data after learning.

© 2018 KKITS All rights reserved

KEYWORDS: Deep learning, Heart disease prediction, Multilayer perceptron, Back propagation, TensorFlow

ARTICLE INFO: Received 16 April 2018, Revised 11 May 2018, Accepted 8 June 2018.

*Corresponding author is with the Department of Computer Systems and Engineering, Inha Technical

College, 100 Inha-ro Nam-gu Incheon, 22212, KOREA.
E-mail address: cjkim@inhatec.ac.kr

1. 서론

의료 분야에 인공지능(Artificial Intelligence) 기술에 대한 적용이 활발하게 이루어지고 있다. 의사에 의한 진료는 아직까지 더 정확할 수 있으나 누락된 진료 기록이 존재한다면 학습을 통해 일정한 패턴을 갖는 질병을 예측할 수 있을 것이다. 심장병의 경우 가슴 통증, 혈압, 심박수, 혈관수 등의 진료 패턴이 존재하여 실제 심장병 환자의 진료 정보를 이용한다면 심장병에 대한 예측이 가능할 것이다. 또한 고비용의 혈관조영술도 비심장병의 심 환자들이 검사를 받을 필요가 없을 것이다. 본 연구에서는 이러한 일정한 패턴을 가진 심장병 환자의 진료 데이터를 기반으로 인공지능의 머신러닝(Machine Learning)[1] 기술인 딥러닝(Deep Learning)[2]을 통해 심장병 예측을 연구한다.

1장에서는 본 연구의 심장병 예측의 필요성을 설명한다. 2장에서는 관련 연구로서 딥러닝 기술을 파악하고, 심장병 관련 기존 연구들을 분석한다. 3장에서는 심장병 예측을 위한 심장병 환자의 데이터 정보와 딥러닝 기반의 다층 퍼셉트론을 제시하고 딥러닝의 개발 단계를 제시한다. 4장에서는 구글의 텐서플로를 이용하여 심장병 예측을 실험하며, 기존의 연구들과 정확도를 비교 분석한다. 5장에서 결론 및 향후 연구 과제를 제시한다.

2. 관련연구

2.1 딥러닝

머신러닝을 구현하는 기술로서 인공신경망(Artificial Neural Network)[3]을 활용한 딥러닝 기술이 최근 활발하게 연구되고 있으며, 자연어 처리, 음성 인식, 영상 인식 등 다양한 응용 분야에 적용되고 있다.

인공신경망은 <그림 1>에서와 같이 입력값과 가중치 그리고 활성화 함수(Activation Function)에 의해 학습되는 하나의 신경망으로 구성되며, 이러한 신경망이 서로 여러 층으로 연결되어 다층의 인공신경망을 구성한다. 이와 같이 딥러닝은 단일 신경망이 아닌 다층 신경망으로 학습의 정확도를 높일 수 있으며 입력층과 출력층에 다층의 은닉층이 존재하여 심도 있는 학습(Deep Learning)을 할 수 있다.

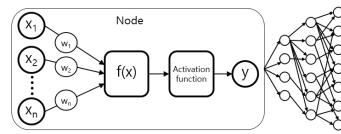


그림 1. 딥러닝의 뉴럴 네트워크

Figure 1. Neural Network of Deep Learning

대표적인 딥러닝 라이브러리로서 구글에서 제공하는 텐서플로(Tensorflow)[4]이다. 텐서플로는 오픈소스 소프트웨어로서 활용 사례들과 개발자들의 참여로 성능이 다른 라이브러리들에 비해 뛰어나며 풍부한 라이브러리들이 포함되어 있다.

본 연구에서는 심장병 예측의 머신러닝을 위해 텐서플로를 활용하며 파이썬(Python)[5]의 수학적 함수들을 함께 이용한다.

2.2 심장병환자 데이터셋의 특성을 고려한 사례기반추론 기법의 성능 예측

심장병 환자 데이터셋의 특성을 고려한 사례기반 추론 기법의 성능 예측 연구 [6]에서는 심장병 환자의 데이터를 기반으로 사례기반 추론 기법(kNN, k-Nearest Neighbor)[7]을 제시하였으며, 심장병 환자의 데이터에 대한 속성들을 이용하여 로지스틱(Logistic) 알고리즘, 베이지안망(Bayesian Network) 알고리즘과 예측성능을 비교하였다.

kNN은 분류할 데이터와 입력 데이터와의 거리

를 계산하여 가까운 거리의 데이터 k를 찾아 가장 빈도수가 높은 클래스로 분류하는 기법이다. 주로 군집화 과정에 사용되거나 분류 과정에 이용된다. 연구 [6]에서는 kNN 실험을 위해 Weka[8] 를 이용하였으며 로직스틱 알고리즘과 베이지안망 알고리즘에 비해 학습성능, 분류율, TP/FP, 정확도, 민감도/특이도가 우수한 것으로 분석되었다.

2.3 서포터 벡터 머신을 이용한 데이터의 결측값 처리에 기반한 심장병 진단 예측 향상

서포터 벡터 머신을 이용한 심장병 예측 향상 연구 [9]는 다층 퍼셉트론 모델을 기반으로 하는 데이터 마이닝 도구인 Weka[8]를 통해 심장병을 예측한다. 심장병 환자의 데이터는 연구 [6]과 동일한 데이터를 사용하며, 본 연구에서도 동일한 데이터를 이용하여 딥러닝을 수행한다.

연구 [9]에서는 다층 퍼셉트론의 모델 구조로 18 개의 입력층 노드와 2개의 출력층 노드, 그리고 10 개의 은닉층 노드로 구성된다. 학습률은 0.01이고 학습 횟수는 300회를 수행한다.

연구 [9]의 분류 정확도는 82.60%이며, 연구 [6]의 90.97%에 비해 낮게 분석되었다.

본 연구에서는 연구 [6], 연구 [9]와 비교하여 심장병 환자의 데이터를 다층 퍼셉트론 기반의 딥러닝 기법에 적용하고, 또한 학습의 횟수를 증가시켜 정확도를 향상시킬 수 있도록 연구한다.

3. 딥러닝 기반의 심장병 예측

본 연구에서는 딥러닝 기반의 심장병을 예측하기 위해 다층 퍼셉트론을 이용한다. 다층 퍼셉트론은 딥러닝 기법 중에 구글의 텐서플로[4]를 적용한다.

3.1 딥러닝 기반의 심장병 예측을 위한 학습 데이터

딥러닝 학습을 위한 심장병 관련 데이터는 미국 캘리포니아 대학교 어바인의 머신러닝 & 인공지능 시스템 센터(Center for Machine Learning and Intelligent Systems)에서 제공하는 데이터를 활용한다[10]. 이 센터에서는 심장병 데이터를 미국(클리브랜드, 롱비치), 헝가리, 스위스의 병원에서 기부 받은 데이터이다. 심장병을 나타내는 데이터의 속성은 <표 1>과 같다.

표 1. 심장병 데이터의 속성
Table 1. Attribute of Heart Disease Data

No.	Attribute
1	age
2	sex
3	chest pain type(4 values)
4	resting blood pressure
5	serum cholestorl in mg/dl
6	fasting blood sugar > 120 mg/dl
7	resting electrocardiographic results (values 0,1,2)
8	maximum heart rate achieved
9	exercise induced angina
10	oldpeak= ST depression induced by exercise relative to rest
11	the slope of the peak exercise ST segment
12	number of major vessels(0-3) colored by flourosopy
13	thal

- sex : 성별(1:남성, 2:여성)
- chest pain type(4 values) : 가슴 통증 타입(1: 전형적 협심증, 2 : 이례적인 협심증, 3: 비협심증 고통, 4: 증상 없음)
- resting blood pressure : 안정 시 혈압
- serum cholestorl in mg/dl : 혈청 콜레스테롤
- fasting blood sugar > 120 mg/dl : 공복 혈당(1: true, 2: false)

- resting electrocardiographic results (values 0,1,2) : 안정 시 심정도(0: 정상, 1:ST-T 보유, 2:(심장의)심실 문제)
- maximum heart rate achieved : 최대 심박수
- exercise induced angina : 운동으로 인한 협심증(1:있음, 2:없음)
- oldpeak= ST depression induced by exercise relative to rest : 휴식과 비례해서 운동으로 인한 ST 경사수치(1:위로 경사, 2:평탄, 3:아래로 경사)
- the slope of the peak exercise ST segment : 최대 ST 세그먼트 경사
- number of major vessels(0-3) colored by flourosopy : 주요 혈관의 수
- thal : 결합 정도(3: 정상, 6:결합 고정, 7:결합 없음)

심장병 데이터의 속성에 따른 심장병 유무를 나타내는 결과값은 <표 2>와 같다.

표 2. 심장병 데이터의 결과(클래스)
Table 2. Result(Class) of Heart Disease Data

No.	Class
1	heart disease

- heart disease : 심장병 진단(1:없음, 2:있음)

표 3. 심장병 데이터의 속성과 클래스 데이터
Table 3. Attribute and Class Data of Heart Disease

No	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1	66	0.0	4.0	178	228	1.0	0.0	165	1.0	1.0	2.0	2.0	7.0	2
2	38	1.0	1.0	120	231	0.0	0.0	182	1.0	3.8	2.0	0.0	7.0	2
3	62	0.0	4.0	150	244	0.0	0.0	154	1.0	1.4	2.0	0.0	3.0	2
4	55	1.0	2.0	130	262	0.0	0.0	155	0.0	0.0	1.0	0.0	3.0	1
...														
269	57	1.0	4.0	140	192	0.0	0.0	148	0.0	0.4	2.0	0.0	6.0	1
270	67	1.0	4.0	160	286	0.0	2.0	108	1.0	1.5	2.0	3.0	3.0	2

<표 1>과 <표 2>의 심장병 데이터(속성, 클래스)

에 대한 실사례는 <표 3>과 같다. 속성 1~13까지 <표 1>에서 제시한 속성 타입으로 포함하고 있으며, 14번째 데이터인 클래스는 1 또는 2에 의해 심장병 유무를 나타내고 있다. 이러한 속성과 클래스로 구성된 270명의 심장병 관련 정보를 이용하여 심장병 예측을 위한 딥러닝 학습을 수행한다.

3.2 심장병 예측을 위한 딥러닝

심장병 예측을 위해 본 연구에서는 다층 퍼셉트론(Multilayer Perceptron)[11]을 이용한다. 퍼셉트론(Perceptron)[12]은 입력 데이터와 활성화 함수(Activation Function)[13]를 사용하여 다음 노드로 전달하기 위한 가장 작은 신경망(Neural Network)으로서 다층 퍼셉트론은 입력층(Input Layer)과 출력층(Output Layer) 사이에 은닉층(Hidden Layer)을 추가한 신경망으로 본 연구에서는 <그림 2>에서와 같이 입력층, 은닉층, 출력층의 노드를 구성한다.

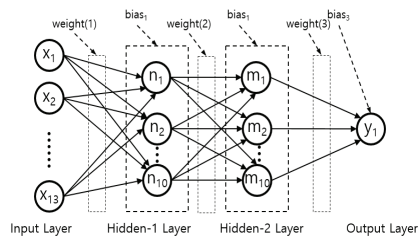


그림 2. 심장병 예측을 위한 다층 퍼셉트론
Figure 2. Multilayer Perceptron for Heart Disease Prediction

<그림 1>에서와 같이 입력층에 13개의 노드와 2개의 은닉층은 각각 10개의 퍼셉트론으로 구성한다. 각 층마다 가중치(Weight)와 바이어스(Bias)가 계산된다.

은닉층의 퍼셉트론 값에 대한 함수 <수식 1>과 같다. 입력값(x_i)과 가중치(w_{ij}), 그리고 바이어스(bias)를 이용하여 활성화 함수(σ)를 통해 값을 판단한다. <수식 1>에 대한 본 연구의 사례에서 입력

층의 노드 수 n은 13이며 은닉층의 퍼셉트론 수 m은 10으로 처리된다.

$$sum(PVH) = \sum_{i=1}^n \sigma \left(\sum_{j=1}^m (x_j w_{ij}) + bias_i \right)$$

PVH : Perceptron Value of Hidden Node
 σ : Activation Function
 n : number of Node in Hidden Layer
 m : number of Node in Input Layer

수식 1. 은닉층의 퍼셉트론 값의 합
 Exp. 1. Perceptron Values Sum of Hidden Layer

은닉층의 퍼셉트론 값을 합하여 실제값과 비교하기 위해 <수식 2>와 같이 출력층 값을 계산한다.

$$y_1 = \sigma(sum(PVH) + bias)$$

수식 2. 출력층의 퍼셉트론 값
 Exp. 2. Perceptron Values of Output Layer

출력층 값과 실제값을 비교하여 오차가 발생하면 전단계의 노드로 회귀하여 가중치를 수정 한 후 다시 계산해야 한다.

$$f(x) = \begin{cases} x, & x > 0 \\ 0, & x \leq 0 \end{cases}$$

수식 3. 렐루 활성화 함수
 Exp. 3. ReLU Activation Function

본 연구에서는 오차를 줄이기 위해 오차 역전파 (Back Propagation)의 최적화 과정[14]을 이용한다. 오차 역전파를 통해 가중치를 수정하기 위해 미분값의 기울기를 수정해야 한다. 기울기가 0에 가깝도록 오차를 줄이기 위해 가중치를 수정한다. 그런데 전단계의 층으로 넘어 갈수록 오차를 줄이지 못하고 0으로 수렴되어 가중치를 수정하지 못하게 되는 문제가 발생할 수 있다. 그래서 본 연구에서는 활성화 함수 중에 1보다 작은 값이 0으로 수렴되는 문제를 해결할 수 있는 렐루(ReLU) 활성화 함수[15]를 사용한다. <수식 3>와 같이 렐루는 입력 값 x가 0보다 작을 때 모든 값을 0으로 처리하고 0

보다 큰 값은 x를 그대로 사용한다. 이와 같이 렐루를 사용하여 가중치 수정이 맨 처음 층까지 갈 수 있도록 할 수 있다.

딥러닝의 성능 및 정확도를 향상시키기 위한 최적화 방법으로 경사하강법(Gradient Descent)[16]을 사용할 수 있다. 그러나, 경사하강법은 가중치를 수정할 때 전체 데이터를 미분해야 하므로 계산량이 많다. 이에 본 연구에서는 전체 데이터를 계산하지 않고 랜덤하게 데이터를 추출하는 고급 경사하강법을 사용한다. 고급 경사 하강법 중에 선택되는 데이터의 진폭을 줄이고 정확도를 높이기 위한 아담(Adam)[17]을 활용한다.

3.3 딥러닝 학습 프로세스

다층 퍼셉트론에 기반한 딥러닝 개발 프로세스는 <그림 3>과 같다.

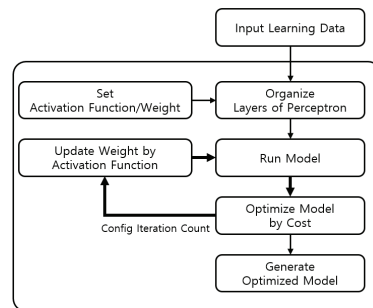


그림 3. 다층 퍼셉트론기반의 딥러닝 프로세스
 Figure 3. Deep Learning Process based on Multilayer Perceptron

- 학습 데이터 입력

딥러닝 학습의 정확도를 높이기 위해서는 신뢰성 있는 데이터를 구성하여 입력해야 한다. 본 연구에서는 <표 3>과 같이 병원에서 부터 기부 받은 신뢰성 있는 심장병 데이터를 활용하여 입력한다.

- 다층 퍼셉트론 구성

학습의 사례에 따라 입력층과 은닉층, 그리고 출

력층을 구성한다. 학습의 사례에 따라 각 층에 퍼셉트론(노드)의 개수를 정해야 한다. 다층 퍼셉트론 구성과 함께 활성화 함수(시그모이드, 렐루 등), 가중치, 학습률(Learning Rate)을 지정한다.

- 모델 실행

예측 모델을 생성하기 위해 학습을 실행한다. 학습은 입력층으로 부터 입력 데이터를 받아서 은닉층, 그리고 출력층으로 퍼셉트론의 개수에 따라 실행한다. 학습 데이터에 대해 지정된 반복 횟수만큼 반복 학습한다.

- 모델 최적화

오차를 줄이기 위해 경사하강법 등을 이용하며 찾아진 오차값에 의해 활성화 함수(시그모이드, 렐루, 등)의 가중치를 수정하여 학습을 재실행한다. 이러한 과정은 지정된 반복 횟수만큼 실행된다.

4. 실험 및 평가

지금까지 심장병 예측을 위한 딥러닝 기반의 다층 퍼셉트론을 적용하기 위한 심장병 학습 데이터와 딥러닝 학습 프로세스를 연구하였다.

본 실험에서는 딥러닝 오픈 소프트웨어인 텐서플로를 이용하여 심장병 학습 모듈을 개발하고 예측을 통해 정확도를 검증한다.

표 4. 학습 데이터 전처리
Table 4. Preprocessing of Learning Data

No	1	2	3	4	5	6	7	8	9	10	11	12	13	14	
1	66	0	4	178	228	1	0	165	1	1.0	2	2	7	1	
2	38	1	1	120	231	0	0	182	1	3.8	2	0	7	1	
3	62	0	4	150	244	0	0	154	1	1.4	2	0	3	1	
4	55	1	2	130	262	0	0	155	0	0.0	1	0	3	0	
						...									
269	57	1	4	140	192	0	0	148	0	0.4	2	0	6	0	
270	67	1	4	160	286	0	2	108	1	1.5	2	3	3	1	

<표 4>는 <표 3>의 심장병 데이터에 대해 학습 및 예측하기 적합한 데이터로 전처리한 데이터이

다. 속성 데이터(1~13)의 타입을 실수에서 정수 데이터로 변환하며 심장병을 나타내는 클래스(14) 값은 1과 2에서 0과 1로 변환한다. 10번 데이터는 실수값으로 된 데이터가 있으므로 실수를 유지한다.

학습 딥러닝 모듈 개발은 파이썬 기반의 텐서플로를 이용한다. <그림 4> 코드는 심장병 데이터의 속성 13개를 x_data에 저장하며, 클래스 데이터 1개를 y_data 에 저장한다.

```
Data_set = np.loadtxt("heartdisease.csv", delimiter=" ")
x_data = Data_set[:,0:13]
y_data = Data_set[:,13]
```

그림 4. 심장병 학습 데이터 입력 코드
Figure 4. Input Code of Heart Disease Learning Data

<그림 5> 코드와 같이 다층 퍼셉트론 적용은 1개의 입력층과 2개의 은닉층, 그리고 1개의 출력층으로 구성한다. 입력 데이터는 13개로 구성되며, 2개의 은닉층 각각은 10개의 퍼셉트론으로 구성한다. 오차를 줄이기 위한 오과 역전파 함수는 출력층에서 은닉층으로는 렐루를 사용하였으며, 은닉층 간에는 시그모이드를 사용한다.

```
W1 = tf.Variable(tf.random_uniform([13, 10], -1.0, 1.0))
b1 = tf.Variable(tf.random_uniform([10], -1.0, 1.0))
L1 = tf.add(tf.matmul(X, W1), b1)
L1 = tf.nn.sigmoid(L1)

W2 = tf.Variable(tf.random_uniform([10, 10], -1.0, 1.0))
b2 = tf.Variable(tf.random_uniform([10], -1.0, 1.0))
L2 = tf.add(tf.matmul(L1, W2), b2)
L2 = tf.nn.sigmoid(L2)

W3 = tf.Variable(tf.random_uniform([10, 1], -1.0, 1.0))
b3 = tf.Variable(tf.random_uniform([1], -1.0, 1.0))
L3 = tf.add(tf.matmul(L2, W3), b3)
L3 = tf.nn.relu(L3)

model = L3
```

그림 5. 퍼셉트론의 계층 구성 코드
Figure 5. Layers Organization Code of Perceptron

<그림 5>에서 학습된 모델은 <그림 6>에서와 같이 모델 최적화 과정을 수행한다. 텐서플로에서 제

공하는 아답을 활용하여 최적화한다.

```
cost = tf.reduce_mean(tf.square(model - Y))
optimizer =
    tf.train.AdamOptimizer(learningRate).minimize(cost)

sess = tf.Session()
sess.run(tf.global_variables_initializer())

for step in range(learningTime):
    sess.run(optimizer, feed_dict={X:x_data, Y:y_data})
    if(step+1) % reportStep == 0:
        print('learningStep : ', step+1, ' / loss : ',
              sess.run(cost, feed_dict={X:x_data, Y:y_data}))
```

그림 6. 모델 실행 및 최적화 코드
Figure 6. Run and Optimize Code of Model

본 실험에서는 학습 횟수(learningTime)은 20만 번 이상 수행하였으며 <그림 7>에서와 같이 10만 번 이상을 넘어도 오차에 대한 변동이 없는 것으로 분석되었다. 10만 번 이상의 학습에 의한 오차는 0.0502로 일관되게 나옴을 확인할 수 있다.

```
(base) C:\W\deep_class>python HeartDiseasePrediction.py
2018-03-16 11:20:06.593943: W
c:\W\tensorflow_1501918863922\work\tensorflow-
1.2.1\tensorflow\core\platform\cpu_feature_guard.cc:45] The TensorFlow lib
rary wasn't compiled to use FMA instructions, but these are available on your
machine and could speed up CPU computations.

learningStep : 211000 / loss : 0.0502413
learningStep : 212000 / loss : 0.050241
learningStep : 213000 / loss : 0.0502574
learningStep : 214000 / loss : 0.0502422
learningStep : 215000 / loss : 0.05024
learningStep : 216000 / loss : 0.0502397
learningStep : 217000 / loss : 0.0502394
learningStep : 218000 / loss : 0.0502411
learningStep : 219000 / loss : 0.0502388
learningStep : 220000 / loss : 0.0502396
```

그림 7. 딥러닝에 의한 오차 결과
Figure 7. Loss Results by Deep Learning

학습되어 도출된 모델을 통해 심장병을 예측하기 위해 <표 5>와 같은 심장병 환자의 샘플 데이터를 이용한다. 의료법의 ‘비밀누설 금지’에 의해 심장병 환자의 정보를 획득하는데 어려움이 있어 캘리포니아 대학교 어바인의 머신러닝 & 인공지능 시스템 센터(Center for Machine Learning and Intelligent Systems)에서 제공하는 데이터 [10]로부터 획득한 심장병 데이터 중에 4개를 임의로 추

출하여 적용하기로 한다. 심장병 진료 정보 1~13 데이터를 검증하며 오차값 0.0502에 따라 심장병 예측 14의 값을 예측한다.

표 5. 심장병 예측을 위한 평가 데이터
Table 5. Evaluation Data for Heart Disease Prediction

No	1	2	3	4	5	6	7	8	9	10	11	12	13	14
e1	41	1	4	110	172	0	2	158	0	0.0	1	0	7	1
e2	68	0	3	120	211	0	2	115	0	1.5	2	0	3	0
e3	54	0	2	132	288	1	2	159	1	0.0	1	1	3	0
e4	61	1	1	134	234	0	0	145	0	2.6	2	2	3	1

<그림 5>의 학습 모델에 따라 <표 5>의 심장병 데이터를 검증한 결과는 <그림 8>과 같다. 실제값과 예측값이 1, 0, 0, 1로 동일하게 예측한 것을 확인할 수 있다.

```
Prediction by [e1] Data , Probability : (1,0.75490999221801758)
Prediction by [e2] Data , Probability : (0,0.83375692367553711)
Prediction by [e3] Data , Probability : (0,0.92202425003051758)
Prediction by [e4] Data , Probability : (1,0.97085380554199219)
```

그림 8. 딥러닝에 의한 심장병 예측
Figure 8. Heart Disease Prediction by Deep Learning

지금까지 딥러닝 기반의 다층 퍼셉트론을 텐서플로에 이용하여 개발하였으며 심장병 데이터의 학습을 통해 정확도를 파악할 수 있었다. <표 6>에서와 같이 본 연구의 정확도가 94.98로 기존 연구들과 비교하여 향상되었음을 확인할 수 있다.

표 6. 정확도 비교
Table 6. Accuracy Comparison

Factors Studies	Number of Learning	Accuracy (%)
Research [6]	10	90.97
Research [9]	300	82.60
This Paper	100,000	94.98

연구 [6]은 사례기반 추론 기법(kNN)을 이용하였

으며, 연구 [9]는 다층 퍼셉트론을 기반으로 Weka 도구를 이용하여 심장병을 예측하였다. 이에 반해 본 연구에서는 다층 퍼셉트론을 기반으로 텐서플로의 딥러닝 도구를 이용하였으며, 학습 횟수 또한 증가시켜 정확도를 높일 수 있도록 하였다.

5. 결 론

본 연구에서는 딥러닝 기반의 다층 퍼셉트론을 이용하여 심장병을 예측하기 위한 연구를 수행하였다. 심장병 데이터는 미국 캘리포니아 대학교 어바인의 머신러닝 & 인공지능 시스템 센터에서 제공하는 데이터(심장병 증상, 심장병 유무)를 이용하였으며, 심장병 예측을 위해 텐서플로 기반의 딥러닝 모듈을 개발하였다. 딥러닝 학습을 통해 심장병을 예측하였으며 본 연구의 정확도가 기존 연구들에 비교하여 향상됨을 파악할 수 있었다.

향후 연구로는 본 연구에서 개발된 텐서플로 기반의 딥러닝 모듈을 활용하여 다른 질병 예측에 적용할 수 있도록 연구한다.

References

- [1] Machine Learning, https://en.wikipedia.org/wiki/Machine_learning, retrieved Jan. 2018.
- [2] Deep Learning, https://en.wikipedia.org/wiki/Deep_learning, retrieved Jan. 2018.
- [3] Artificial Neural Network, https://en.wikipedia.org/wiki/Artificial_neural_network, retrieved Jan. 2018.
- [4] TensorFlow, <https://www.tensorflow.org>, retrieved Jan. 2018.
- [5] Python, <https://www.python.org>, retrieved Jan. 2018.
- [6] M. H. Yoon, J. H. Kim, and H. Jin, *Prediction for performance of kNN in diagnosis considering features of coronary artery disease dataset*, Proceedings of the Institute of Electronics Engineers of Korea, Nov. 2013.
- [7] kNN, https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm, retrieved Feb. 2018.
- [8] Weka 3: Data Mining Software in Java, <https://www.cs.waikato.ac.nz/ml/weka>, retrieved Feb. 2018.
- [9] J. Y. Choi, and M. K. Kim, *An enhanced approach for predicting heart disease based on handling missing values using support vector machine*, Proceedings of Symposium of the Korean Institute of communications and Information Sciences, Jan. 2018.
- [10] Statlog (Heart) Data Set, [http://archive.ics.uci.edu/ml/datasets/statlog+\(heart\)](http://archive.ics.uci.edu/ml/datasets/statlog+(heart)), retrieved Feb. 2018.
- [11] Multilayer Perceptron, https://en.wikipedia.org/wiki/Multilayer_perceptron, retrieved Jan. 2018.
- [12] Perceptron, <https://en.wikipedia.org/wiki/Perceptron>, retrieved Jan. 2018.
- [13] Activation Function, https://en.wikipedia.org/wiki/Activation_function, retrieved Jan. 2018.
- [14] Back Propagation, <https://en.wikipedia.org/wiki/Backpropagation>, retrieved Jan. 2018.
- [15] Rectifier(neural networks), [https://en.wikipedia.org/wiki/Rectifier_\(neural_networks\)](https://en.wikipedia.org/wiki/Rectifier_(neural_networks)), retrieved Jan. 2018.
- [16] Gradient Descent, https://en.wikipedia.org/wiki/Gradient_descent, retrieved Feb. 2018.
- [17] AdamOptimizer, https://www.tensorflow.org/versions/r1.0/api_docs/python/tf/train/AdamOptimizer, retrieved Feb. 2018.

딥러닝 기반의 다층 퍼셉트론을 이용한 심장병 예측 연구

김철진¹, 김지성²

¹인하공전 컴퓨터시스템과 교수

²한양대학교 컴퓨터소프트웨어학과 석사과정

요 약

머신러닝의 고도화 기술인 딥러닝은 현재 다양한 산업 분야에 적용되고 있으며 예측이 요구되는 금융, 전자상거래, 사물 인터넷 분야에 활발하게 적용 개발되고 있다. 특히, 본 연구에서는 의료 분야 중에 심장병 예측에 적용하여 조기에 심장병을 발견할 수 있게 하거나, 비심장병 환자의 혈관조영술과 같은 고가의 진료 비용을 절감할 수 있는 장점을 제공할 것이다. 딥러닝 학습을 위해 다층 퍼셉트론 머신러닝 기술을 적용하고자 하며 2개의 은닉층과 10개의 퍼셉트론을 구성하여 학습의 정확도를 높일 수 있도록 한다. 학습 시 오차 역전파를 이용하여 출력층에서 은닉층으로 역전하여 오차를 줄일 있도록 한다. 렐루나 시그모이드 함수를 이용하여 오차를 줄이며 생성된 예측 모델은 아담 최적화 함수를 통해 모델을 최적화한다. 사례 연구에서 심장병 데이터는 캘리포니아 대학교 어바인의 머신러닝 & 인공지능 시스템 센터에서 제공하는 심장병 진료 및 심장병 유무 데이터를 활용하여 학습한다. 딥러닝 학습 모듈은 파이썬 기반의 텐서플로를 이용하여 개발하며 학습 후 심장병 데이터로 부터 무작위로 추출한 데이터 샘플을 이용하여 검증한다.

Engineering at Inha Technical College since 2009. His research focus is software engineering, object-oriented & component technique, software architecture, customization, and embedded software.

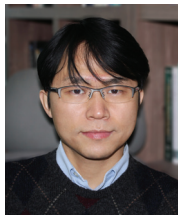
E-mail address: cjkim@inhac.ac.kr



Ji Sung Kim received a bachelor's degree from Inha Technical College in 2018 and is currently studying artificial intelligence with a master Candidate in Dept. of Computer Science from

Hanyang University.

E-mail address: intelli8786@gmail.com



Chul Jin Kim received his B.E. degree in Computer Science from the Kyonggi University in 1996. He received both M.S. and Ph. D. degree in Computer Science from the Soongsil

University in 1998 and 2004. He worked as a senior researcher in SAMSUNG Electronics from 2004 to 2009. He has been a professor in the Department of Computer Systems and