

딥 러닝 프레임워크의 비교 및 분석

이요섭* · 문필주**

A Comparison and Analysis of Deep Learning Framework

Yo-Seob Lee* · Phil-Joo Moon**

요 약

딥 러닝은 사람이 가르치지 않아도 컴퓨터가 스스로 사람처럼 학습할 수 있는 인공지능 기술이다. 딥 러닝은 세상을 이해하고 감지하는 인공지능을 개발하는데 가장 촉망받는 기술이 되고 있으며, 구글, 바이두, 페이스북 등이 가장 앞서서 개발을 하고 있다. 본 논문에서는 딥 러닝을 구현하는 딥 러닝 프레임워크의 종류에 대해 논의하고, 딥 러닝 프레임워크의 영상과 음성 인식 분야의 효율성에 대해 비교, 분석하고자 한다.

ABSTRACT

Deep learning is artificial intelligence technology that can teach people like themselves who need machine learning. Deep learning has become of the most promising in the development of artificial intelligence to understand the world and detection technology, and Google, Baidu and Facebook is the most developed in advance. In this paper, we discuss the kind of deep learning frameworks, compare and analyze the efficiency of the image and speech recognition field of it.

키워드

Machine Learning, Deep Learning, AI
머신 러닝, 딥 러닝, 인공 지능

1. 서 론

머신 러닝(machine learning; 기계 학습)은 인공지능의 한 분야로서, 컴퓨터가 학습할 수 있도록 하는 알고리즘과 기술을 개발하는 분야로서, 알고리즘을 이용해 데이터를 분석하고, 분석을 통해 학습하며, 학습한 내용을 기반으로 판단이나 예측을 한다[1].

딥 러닝은 머신 러닝의 한 분야로서, 뇌의 뉴런과 유사한 정보 입력층 계층을 활용해 데이터를 학습한다. 굉장한 양의 연산을 필요로 하기 때문에 초기에는

어려웠지만 슈퍼컴퓨터를 기반으로 이러한 문제점을 해결했으며, 현재는 병렬연산에 최적화된 GPU를 사용하여 딥 러닝 기술이 발전하게 되었다.

딥 러닝의 가장 중요한 역할을 본다면 “학습에 의한 예측”이다. 사람의 경우, 학습을 해도 어느 시점부터는 잊어버리지만 기계에 의해 학습되어 저장된 것은 잊어버릴 수가 없다. 학습한 것을 모두 기억한다면 조만간 인간보다 더 똑똑한 인공지능이 나타날 수 있을 것이고 학습 시간도 상상할 수 없을 만큼 인간보다 빠를 것이다[2-5].

* 평택대학교 컴퓨터학과(yslee@ptu.ac.kr)

** 교신저자 : 평택대학교 정보통신학과

• 접수일 : 2016. 10. 26

• 수정완료일 : 2017. 02. 13

• 게재확정일 : 2017. 02. 24

• Received : Oct. 26, 2016, Revised : Feb. 13, 2017, Accepted : Feb. 24, 2017

• Corresponding Author : Phil-Joo Moon

Dept. of Information & Communication, Pyeongtaek University,

Email : pjmoon@ptu.ac.kr

딥 러닝은 충분히 많은 데이터를 가지고 반복적인 학습을 통해 효율적으로 올바른 결론을 도출해내기 위한 과정이다.

딥 러닝은 세상을 이해하고 감지하는 인공지능을 개발하는데 가장 촉망받는 기술이 되고 있으며, 구글, 바이두, 페이스북 등이 가장 앞서서 개발을 하고 있으며, GPU를 사용한 다양한 딥 러닝 연구들과 성과들이 적용되고 있다.

본 논문에서는 심층 신경망의 정의와 종류에 대해 논의하고, 이 심층 신경망에서 동작하는 딥 러닝 프레임워크들의 제공 기능들을 비교 및 분석을 통하여 원하는 학습 대상에 맞는 최적인 딥 러닝 프레임워크를 선택하여 적용하는데 유용성을 제공하고자 한다.

II. 딥 러닝

딥 러닝(deep learning)은 머신 러닝의 한 분야로서, 여러 비선형 변환기법들로 구성된 다중 처리 계층을 사용하여 높은 수준의 추상화, 즉, 다량의 데이터나 복잡한 자료들 속에서 핵심적인 내용 또는 기능을 요약하는 작업을 하는 알고리즘의 집합이다[2].

어떠한 데이터가 있을 때 이를 컴퓨터가 알아 들을 수 있는 형태(예를 들어 이미지의 경우는 픽셀정보를 열벡터로 표현하는 등)로 표현하고 이를 학습에 적용하기 위해 많은 연구(어떻게 하면 더 좋은 표현기법을 만들고 또 어떻게 이것들을 학습할 모델을 만들지에 대한)가 진행되고 있으며, 다양한 딥 러닝 아키텍처들은 이러한 노력의 결과로 다양한 딥 러닝 기법들이 컴퓨터 비전, 음성인식, 자연어처리, 음성/신호처리 등의 분야에 적용되어 최첨단의 결과들을 보여주고 있다.

딥 러닝 기술의 주요 응용분야들은 영상 인식, 음성 인식, 자연어 처리, 약물 성분 및 독성 판단, 고객중심 관리 등이다.

III. 심층 신경망

심층 신경망(Deep Neural Network, DNN)은 입력 계층(input layer)과 출력 계층(output layer) 사이에 복수개의 은닉 계층(hidden layer)들로 이뤄진 인공신

경망(Artificial Neural Network, ANN)이다.

심층 신경망은 표준 오류역적과 알고리즘으로 학습될 수 있다. 이때, 가중치(weight)들은 (1)식을 이용한 확률적 경사 하강법(stochastic gradient descent)을 통하여 갱신될 수 있다.

$$\Delta w_{ij}(t+1) = \Delta w_{ij}(t) + \eta \frac{\partial C}{\partial w_{ij}} \quad (1)$$

여기서, η 는 학습률(learning rate)을 의미하며, C 는 비용함수(cost function)를 의미한다. 비용함수의 선택은 학습의 형태(지도 학습, 자율 학습 (기계 학습), 강화 학습 등)와 활성화함수(activation function)같은 요인들에 의해서 결정된다[1].

심층 신경망은 합성곱 신경망, 순환 신경망, 제한 불츠만 머신, 심층 신뢰 신경망 등이 있다[2].

3.1 심층 신경망(DNN)

심층 신경망(Deep Neural Network, DNN)은 입력 계층(input layer)과 출력 계층(output layer) 사이에 복수개의 은닉 계층(hidden layer)들로 이뤄진 인공신경망이다[2]. 심층 신경망은 일반적인 인공신경망과 마찬가지로 복잡한 비선형 관계(non-linear relationship)들을 모델링할 수 있다. 예를 들어, 사물 식별 모델을 위한 심층 신경망 구조에서는 각 객체가 이미지 기본 요소들의 계층적 구성으로 표현될 수 있다. 이때, 추가 계층들은 점진적으로 모여진 하위 계층들의 특징들을 규합시킬 수 있다. 심층 신경망의 이러한 특징은, 비슷하게 수행된 인공신경망에 비해 더 적은 수의 유닛(unit, node)들 만으로도 복잡한 데이터를 모델링할 수 있게 해준다.

3.2 합성곱 신경망(CNN)

합성곱 신경망(Convolutional Neural Network, CNN)은 최소한의 전처리(preprocess)를 사용하도록 설계된 다계층 퍼셉트론의 한 종류이다.

convolution은 특정 시스템에 입력이 가해졌을 때 시스템의 반응이 어떻게 되는지 해석하기 위한 용도로 사용된다. 영상으로부터 특정 특징들을 추출할 때 사용된다.

입력 영상으로부터 convolution을 통해 feature

1) 딥 러닝, https://ko.wikipedia.org/wiki/딥_러닝.

map을 만든다. subsampling은 feature map의 크기를 줄여주면서 local feature를 생성한다. 이 local feature에 대해 다시 convolution과 subsampling을 수행하여 global feature를 얻을 수 있게 된다. 여러 번의 convolution과 subsampling을 거치게 되면 feature map의 크기가 작아지면서 전체를 대표하는 강한 특징들만 남게 되고, 이 특징들은 fully connected network의 입력으로 제공되며, 학습을 통해 최적의 인식결과를 얻게 된다.

3.3 순환 신경망(RNN)

순환 신경망(Recurrent Neural Network, RNN)은 인공신경망을 구성하는 유닛 사이의 연결이 Directed cycle을 구성하는 신경망을 말한다. RNN은 임의의 입력을 처리하기 위해 신경망 내부의 메모리를 활용할 수 있다. 이러한 특성에 의해 순환 신경망은 필기체 인식과 같은 분야에 활용되고 있고, 높은 인식률을 나타낸다. RNN은 글, 유전자, 손글씨, 음성 신호, 센서가 감지한 데이터, 주가 등 배열의 형태를 갖는 데이터에서 패턴을 인식한다.

3.4 제한 볼츠만 머신(RBM)

제한 볼츠만 머신(Restricted Boltzmann Machine, RBM)은 볼츠만 머신에서, 층간 연결을 없앤 형태의 모델이다. RBM은 차원 감소, 분류, 선형 회귀 분석, 협업 필터링(collaborative filtering), 특징값 학습(feature learning) 및 주제 모델링(topic modelling)에 사용할 수 있다. 층간 연결을 없애면, 머신은 가시 유닛(Visible Unit)과 은닉 유닛(Hidden Unit)으로 이루어진 무방향 이분 그래프 형태의 모양이 된다. RBM은 음성 인식 소프트웨어의 성능을 향상시킨다.

3.5 심층 신뢰 신경망(DBN)

심층 신뢰 신경망(Deep Belief Network, DBN)이란 기계학습에서 사용되는 그래프 생성 모형(generative graphical model)으로, 딥 러닝에서는 잠재변수(latent variable)의 다중계층으로 이루어진 심층 신경망을 의미한다. 계층 간에는 연결이 있지만 계층 내의 유닛 간에는 연결이 없다는 특징이 있다.

IV. 딥 러닝 프레임워크

딥 러닝 프레임워크들은 Caffe, CNTK, TensorFlow, Theano, Torch, MXnet, Chainer, Keras 등이 있으며, 분석하려는 대상에 따라 그 대상에 맞는 프레임워크를 선택하면 좋은 결과를 얻을 수 있다.

영상의 분류나 영상 속의 특정 객체를 검출하는 부분에서 CNN이 우수한 성능을 보인다. 영상 데이터를 대상으로 분석하거나 예측하려는 경우에는 CNN을 지원하는 프레임워크를 사용하면 좋은 결과를 얻을 수 있다는 것이다.

RNN은 학습이 진행되어도 과거 학습의 정보를 잃지 않고 연속적인 정보의 흐름을 학습에 반영 가능하기 때문에 음성이나 언어 등 연속된 입력 데이터 분석에 활용된다.

4.1 Caffe

Caffe는 Berkeley Vision and Learning Center에서 개발되었으며, C++로 ConvNets를 구현하였다. As shown on its official page, Caffe는 하루에 AlexNet에서 single NVIDIA K40 GPU로 6,000만개 이상의 이미지들을 처리할 수 있다. 이미지 분류를 위한 툴킷으로 사용할 수 있다[6].

Caffe는 코맨드 라인, 파이선, MATLAB 인터페이스로 cmdcaffe, pycaffe, matcaffe 등을 제공한다.

4.2 CNTK

CNTK(Computational Network Toolkit)는 마이크로소프트사의 통합 딥 러닝 툴킷이다.

CNTK는 CNN, RNN, LSTM 등의 최근 심층 신경망 아키텍처들의 구조적 구현을 허용한다. auto differentiation, stochastic gradient descent (SGD), backpropagation 등도 지원한다. 다중 머신과 다중 GPU를 이용한 병렬화도 지원한다[7].

CNTK는 Microsoft Azure의 가상 머신이나 Linux의 Docker container를 통해 사용할 수 있다.

CNTK는 GPU 클러스터들에서 학습된 분산 신경망을 지원한다.

4.3 TensorFlow

TensorFlow는 다양한 하드웨어 환경에서 인공지능망 모델을 쉽게 생성하고 학습할 수 있는 인터페이스를 제공한다. 머신러닝 연구자는 텐서플로우를 통해 인공지능망 모델을 학습하는데 필요한 자세한 하드웨어 조작에 신경을 필요 없이 인공지능망 모델 구조에만 집중할 수 있다. 따라서 연구자는 해결하고자 하는 문제에 최적화된 모델을 생성하는데만 집중할 수 있기 때문에 생산성 향상을 도모할 수 있다[8].

4.5 Torch

Torch는 Lua로 작성되었으며, NYU, Facebook AI lab, Google DeepMind에서 사용되었다.

Torch는 GPU를 먼저 사용한 머신 러닝 알고리즘을 지원하는 프레임워크로서, 쉽고 빠른 스크립팅 언어 LuaJIT와 C/CUDA 구현 덕분에 사용하기 쉽고 효율적이다²⁾.

표 1. 딥 러닝 프레임워크의 플랫폼, 인터페이스 비교
Table 1. The comparison of platform, interface of deep learning frameworks

Name	License	Open source	Platform	Written in	Supported Interface
Caffe	BSD 2-Clause License	Yes	Ubuntu, Mac OS, AWS, unofficial Android port, Windows	C++, Python	C, C++, Python, MATLAB, Command line interface
CNTK	MIT license	Yes	Windows, Linux	C++	C++, Command line interface
TensorFlow	Apache 2.0	Yes	Linux, Mac OS	C++, Python	C++, Python
Theano	BSD license	Yes	Cross-platform	Python	Python
Torch	BSD license	Yes	Linux, Android, Mac OS, iOS, Windows	C, Lua	C, C++, Lua
MXnet	Apache 2.0	Yes	Ubuntu, Mac OS, Windows, AWS, Android, iOS, JavaScript	C++, Python, Julia, Matlab, Go, R, Scala	Python, R, C++, Julia
Chainer	MIT license	Yes	Linux, Mac OS, Windows	Python	Python
Keras	MIT license	Yes	Linux, Mac OS, Windows	Python	Python

4.4 Theano

Theano는 다차원 배열(numpy.ndarray)의 수학적 표현을 효율적으로 정의, 최적화, 평가하는 즉, 수식 및 행렬 연산을 쉽게 만들어주는 파이썬 라이브러리로서, Multi-GPU, CuDNN v5를 지원한다[9].

Theano 기반 위에 올려 쉽게 사용할 수 있는 Keras, Pylearn2, Lasagne, Blocks 같은 라이브러리들이 있다.

4.6 MXnet

MXnet은 효율성과 생산성을 극대화하기 위해 심볼릭 프로그래밍과 명령형 프로그래밍의 혼합된 형태를 허용하도록 효율성과 유연성을 위해 설계된 딥 러닝 프레임워크이다³⁾.

2) Torch, <http://torch.ch/>.

3) MXnet, <https://github.com/dmlc/mxnet>.

표 2. 딥 러닝 프레임워크의 선행학습모델, cuDNN, 지원하는 심층신경망 비교
 Table 2. The comparison of pretrained model, cuDNN, supported DNN of deep learning frameworks

Name	Has pretrained models	cuDNN	Recurrent nets	Convolutional nets	RBM/DBNs
Caffe	Yes	v5	Yes	Yes	No
CNTK	No	v5.1	Yes	Yes	No
TensorFlow	Yes	v5.1	Yes	Yes	No
Theano	Through Lasagne's model zoo	v5	Yes	Yes	Yes
Torch	Yes	v5	Yes	Yes	Yes
MXnet	Yes	v3	Yes	Yes	Yes
Chainer	Through Caffe's model zoo	v4	Yes	Yes	Yes
Keras	Yes	depends on the version of TensorFlow and Theano installed with Keras	Yes	Yes	Yes

4.7 Chainer

Chainer는 define-by-run 원칙에 의해 설계된 딥 러닝 프레임워크로서, forward 함수만 정의해주면 네트워크 구조가 자동으로 정해진다. Chainer는 임의의 제어 흐름문을 사용할 수 있도록 실행시 네트워크를 수정할 수 있다[10].

4.8 Keras

Keras는 Python으로 작성된 작고, 고도의 모듈화된 신경망 라이브러리이다. Keras는 TensorFlow 또는 Theano위에서 실행할 수 있으며, 빠른 실험을 활성화하는데 중점을 두고 개발되었다⁴⁾.

V. 딥 러닝 프레임워크의 비교 및 분석

본 장에서는 현재 많이 사용 중인 딥 러닝 프레임워크들(Caffe, CNTK, TensorFlow, Theano, Torch, MXnet, Chainer, Keras)을 조사하여 비교 및 분석하였다⁵⁾⁶⁾. 각 프레임워크에서 사용중인 라이선스, 오픈소

스 여부, 지원 플랫폼, 구현 언어, 인터페이스, 선행학습 모델, CuDNN 사용여부, 지원하는 심층신경망 등을 비교 분석하였다. 표 1은 딥 러닝 프레임워크의 플랫폼과 인터페이스 등을 비교한 내용이다.

대부분의 프레임워크들이 파이선과 C++를 사용하는 인터페이스를 사용하는 것을 알 수 있다. Torch의 경우는 Lua를, MXnet의 경우에는 Julia를 사용하고 있다. Caffe는 코맨드 라인 인터페이스, 파이선 인터페이스, MATLAB 인터페이스로 cmdcaffe, pycaffe, matcaffe 등을 제공하고 있다.

Caffe, Teano, Torch 등은 BSD 라이선스를 사용하고 있으며, CNTK, Chainer, Keras는 MIT 라이선스를, TensorFlow, MXnet은 Apache 라이선스를 사용하고 있다.

MIT 라이선스는 GPL과 달리 카피 레프트는 아니며, 오픈 소스 여부에 관계없이 재사용을 인정하고 있

4) Keras, <https://keras.io/>.

5) "Deep Learning Frameworks", <https://developer.nvidia.com/deep-learning-frameworks>.

6) "DL4J vs. Torch vs. Theano vs. Caffe vs. TensorFlow", <https://deeplearning4j.org/compare-dl4j-torch7-pylearn>.

다. BSD 라이선스의 경우 해당 소프트웨어는 아무나 개작할 수 있고, 수정한 것을 제한 없이 배포할 수 있다. Apache 2.0 라이선스는 누구나 해당 소프트웨어에서 파생된 프로그램을 제작할 수 있으며 저작권을 양도, 전송할 수 있다. 아파치 라이선스, 버전 2.0을 포함시켜야 하며 아파치 소프트웨어 재단에 개발된 소프트웨어라는 것을 명확하게 밝혀야 한다[11].

대부분의 프레임워크들이 MacOS와 Linux, Windows 등을 지원하고 있다. Torch와 MXnet이 안드로이드와 iOS를 지원하고 있으며, Theano는 다른 환경의 운영체제에서 공통으로 사용되는 크로스 플랫폼을 지원하고 있다.

표 2는 딥 러닝 프레임워크의 선행학습모델과 cuDNN 그리고 지원하는 심층신경망을 비교 및 분석한 내용이다.

CNTK를 제외한 대부분의 프레임워크들은 미리 학습된 모델을 지원하고 있으며 Theano는 Lasagne의 zoo 모델을 통해서, Chainer는 Caffe의 zoo 모델을 통해서 지원하고 있다.

대부분의 프레임워크들은 cuDNN을 지원하고 있으며, Keras는 Keras가 설치된 TensorFlow와 Theano의 버전에 따라 종속적인 것을 알 수 있다.

NVIDIA의 CuDNN은 높은 수준의 머신 러닝 프레임워크로 통합되도록 설계된 심층 신경망 프리미티브들의 GPU 가속화 라이브러리로서, cuDNN의 최적화된 루틴은 딥 러닝 개발자들이 튜닝 단계에서 시간을 낭비하지 않고 뉴럴 네트워크 모델의 디자인과 훈련에 집중할 수 있도록 돕는다[13].

대부분의 프레임워크들이 CNN, RNN, RBM, DBN을 지원하고 있기 때문에 영상 분류나 영상속의 객체 검출, 음성, 언어 데이터의 분석이나 예측에 효율적으로 활용할 수 있다. Caffe, CNTK, TensorFlow 만이 RBM, DBN을 지원하지 않고 있다.

표 3은 CNN 아키텍처인 AlexNet을 기반으로 딥 러닝 프레임워크들을 테스트한 결과를 나타낸다. CNN의 경우에는 cuDNN을 이용한 TensorFlow가 가장 빠른 결과를 보이는 것을 알 수 있다[7].

표 4는 간단한 RNN 구현을 통해 딥 러닝 프레임워

크들을 테스트한 결과를 나타낸다. RNN의 경우에는 Theano가 가장 빠른 결과를 나타내는 것을 알 수 있다[8].

표 3. 딥 러닝 프레임워크들의 convnet 벤치마크
Table 3. The convnet benchmarks of deep learning frameworks

Library	Class	Time (ms)	forward (ms)	backward (ms)
TensorFlow	conv2d	81	26	55
Chainer	Convolution 2D	177	42	136
Caffe	Convolution -Layer	324	121	203
Torch-7	SpatialConvolutionMM	342	132	210

표 4. 딥 러닝 프레임워크들의 RNN 벤치마크
Table 4. The RNN benchmarks of deep learning frameworks

Library	Compile (s)	Train (ms)	Forward (ms)
Theano	4.31	104.6	30.9
Torch	0.05	259.93	103.06
TensorFlow	1.64	278.4	111.5

VI. 결 론

본 논문에서는 현재 많이 사용 중인 딥 러닝 프레임워크들을 조사하여 비교 및 분석하였다.

대부분의 프레임워크들이 오픈소스를 제공하고 있지만, BSD 라이선스, MIT 라이선스, Apache 라이선스를 사용하고 있어서 용도에 맞게 선택하여 사용하여야 할 것이다.

또한, MacOS, Linux, Windows, 안드로이드, iOS 등을 지원하고 있다. 개발 플랫폼에 따라 적절하게 사용할 수 있다.

7) Convnet Benchmarks, <https://github.com/soumith/convnet-benchmarks>.

8) RNN Benchmarks, <https://github.com/glampl/rnn-benchmarks>.

대부분의 프레임워크들은 또한 cuDNN을 지원하고 있다. 버전에 따라 차이가 있지만, CuDNN을 통해 딥 러닝 개발자들이 튜닝 단계에서 시간을 낭비하지 않고 뉴럴 네트워크 모델의 디자인과 훈련에 집중할 수 있을 것이다.

이미지나 비디오, 텍스트 등의 데이터를 분류하는데 최적인 CNN의 경우에는 TensorFlow가, 필기체 인식, 자연어 처리에 최적인 RNN의 경우에는 Theano가 가장 효율적인 것으로 나타났다.

분석 결과는 원하는 학습 대상에 맞는 최적인 딥 러닝 프레임워크를 선택하여 적용하는데 많은 도움을 줄 수 있을 것이라고 생각한다.

References

- [1] Y. LeCun, Y. Bengio, and G. Hinton. "Deep learning," *Nature*, vol. 521, no. 7553, 2015, pp. 436-444.
- [2] J. Schmidhuber, "Deep Learning in Neural Networks: An Overview," *Neural Networks*, vol. 61, Jan. 2015, pp. 85-117.
- [3] J. Choi, "Speech and Noise Recognition System by Neural Network," *J. of the Korea Institute of Electronic Communication Sciences*, vol. 5, no. 4, 2010, pp. 357-362.
- [4] J. Cho and H. Yang, "A Car License Plate Recognition Using Colors Information, Morphological Characteristic and Neural Network," *J. of the Korea Institute of Electronic Communication Sciences*, vol. 5, no. 3, 2010, pp. 304-308.
- [5] J. Lee and J. Kim, "Recognition of a New Car Plate using Color Information and Error Back-propagation Neural Network Algorithms," *J. of the Korea Institute of Electronic Communication Sciences*, vol. 5, no. 5, 2010, pp. 471-476.
- [6] Y. Jia, "Caffe: Convolutional Architecture for Fast Feature Embedding," *Proc. MM '14 Proc. of the 22nd ACM Int. Conf. on Multimedia*, Orlando, USA, November 03 - 07, 2014, pp. 675-678.
- [7] F. Seide and A. Agarwal, "CNTK: Microsoft's Open-Source Deep-Learning Toolkit," *KDD '16: Proc. of the 22nd ACM Special Interest Group (SIG) on Knowledge Discovery and Data Mining Int. Conf. on Knowledge Discovery and Data Mining*, San Francisco, USA, Aug. 2016, pp. 2135-2135.
- [8] M. Abadi, "TensorFlow: learning functions at scale," *ICFP 2016: Proc. of the 21st ACM Special Interest Group on programming languages Int. Conf. on Functional Programming*, Nara, Japan, September 18-24, 2016.
- [9] F. Bastien, P. Lamblin, R. Pascanu, J. Bergstra, I. Goodfellow, A. Bergeron, N. Bouchard, D. Warde-Farley, and Y. Bengio, "Theano: new features and speed improvements," *Workshop on Deep Learning and Unsupervised Feature Learning at Neural Information Processing Systems*, Lake Tahoe, USA, December 03-08, 2012.
- [10] S. Tokui, K. Oono, S. Hido and J. Clayton. "Chainer: a next-generation open source framework for deep learning," *In Workshop on Machine Learning Systems at Neural Information Processing Systems*, Montreal, Canada, December 07-12, 2015.
- [11] Y. Lee and P. Moon, "An Analysis of Deep Learning Toolkits Technology Trends," *2016 Conf. of the Korea Institute of Electronic Communication Sciences*, Youngju, Korea, vol. 10, no. 2, June 03-04, 2016, pp. 522-525.

저자 소개



이요섭(Yo-Seob Lee)

1990년 숭실대학교 전자계산학과
졸업 (공학사)

1992년 숭실대학교 대학원 컴퓨터
학과 졸업 (공학석사)

1999년 숭실대학교 대학원 컴퓨터학과 졸업 (공학박사)

현재 평택대학교 컴퓨터학과 교수

※ 관심분야 : 모바일 애플리케이션, 머신 러닝



문필주(Phil-Joo Moon)

1988년 숭실대학교 전자계산학과
졸업 (공학사)

1991년 숭실대학교 대학원 컴퓨터
학과 졸업 (공학석사)

1998년 숭실대학교 대학원 컴퓨터학과 졸업 (공학박사)

1988년 ~ 2001년 ETRI 책임연구원(팀장)

2001년 ~ 현재 평택대학교 정보통신학과 교수

※ 관심분야 : 엑세스망기술, 머신 러닝, IoT