

http://dx.doi.org/10.17703/JCCT.2018.4.2.197

JCCT 2018-5-26

인공지능을 이용한 과일 가격 예측 모델 연구

Fruit price prediction study using artificial intelligence

임진모*, 김월용*, 변우진**, 신승중***

Jin-mo Im*, Weol-Young Kim*, Woo-Jin Byoun, Seung-Jung Shin*****

요약 현재 우리가 사는 21세기에서 가장 핫한 이슈중 하나는 AI이다. 농경사회에서 산업혁명을 통해 육체노동의 자동화를 이루었듯이 정보사회에서 SW혁명을 통해 지능정보사회가 도래했다. Google '알파고'의 등장으로 인해 컴퓨터가 스스로 학습하고 예측하는 machine learning (머신러닝) 사례를 보면서 이제 바둑의 세계 까지 인간이 컴퓨터를 이길 수 없는, 다시 말하면 컴퓨터가 인간을 뛰어넘는 시대가 왔다. 기계학습ML(machine learning)은 인공 지능 분야로, 인공지능 컴퓨터가 인간을 뛰어넘는 시대가 도래했다.

기계학습ML(machine learning)은 인공지능의 분야로, 인공지능 컴퓨터가 혼자 학습 하도록 알고리즘 기술 개발을 하는 뜻을 의미하는데, 많은 기업들이 머신러닝을 바둑의 세계까지 인간이 컴퓨터를 이길 수 없는, 다시 말하면 컴퓨터가 인간을 뛰어넘는 시대가 왔다. 많은 기업들이 머신러닝을 용하는데 그 예로는 Facebook에서 이미지를 계속 학습 하여 나중에 그 이미지가 누구인지 알려주는 것도 머신러닝의 한 사례이다. 또한 구글의 데이터 센터 최적화를 위해서 효율적인 에너지 사용 모델 구축을 위해 neural network(신경망)을 활용하였다. 또 다른 사례로 마이크로소프트의 실시간 통역 모델은 번역 학습을 통해 언어관련 인풋 데이터가 증가할수록 더 정교한 번역을 해주는 모델이다. 이처럼 많은 분야에 머신러닝이 점차 쓰이면서 이제 우리 21세기 사회에서 앞으로 나아가려면 AI산업으로 뛰어들어야 한다.

주요어 : 인공지능, 기계학습, 딥러닝, 알고리즘

Abstract One of the hottest issues in our 21st century is AI. Just as the automation of manual labor has been achieved through the Industrial Revolution in the agricultural society, the intelligence information society has come through the SW Revolution in the information society. With the advent of Google 'Alpha Go', the computer has learned and predicted its own machine learning, and now the time has come for the computer to surpass the human, even to the world of Baduk, in other words, the computer. Machine learning ML (machine learning) is a field of artificial intelligence.

Machine learning ML (machine learning) is a field of artificial intelligence, which means that AI technology is developed to allow the computer to learn by itself. The time has come when computers are beyond human beings. Many companies use machine learning, for example, to keep learning images on Facebook, and then telling them who they are. We also used a neural network to build an efficient energy usage model for Google's data center optimization. As another example, Microsoft's real-time interpretation model is a more sophisticated translation model as the language-related input data increases through translation learning. As machine learning has been increasingly used in many fields, we have to jump into the AI industry to move forward in our 21st century society.

Key words : ML, AI, Deep Running, algorithm

*정회원, 한세대학교 IT융합학과

**정회원, 한세대학교 컴퓨터공학과

***중신회원, 한세대학교 ICT디바이스학과(교신저자)

접수일: 2018년 3월 2일, 수정완료일: 2018년 3월 23일

게재확정일: 2018년 4월 7일

Received: March 2, 2018 / Revised: March 23, 2018

Accepted: April 7, 2018

***Corresponding Author : gandalf@kitech.re.kr

Dept. of IT, Hansei University, Korea

I. 서 론

요즘 과일 가격은 농업경영체의 수익을 결정하는 중요한 요소중 하나로 가격의 불안정성은 농업경영체경영의 위험을 초래하는 한다. 과일의 가격은 시장 또는 마트에서의 수요와 공급량에 의해 결정되며, 수요와 공급에 따라 가격의 영향을 받는다.

과일들의 공급량은 가격, 대체과일이 가능한 타 과일의 가격, 생산요소의 가격 등에 의해 결정되며, 수요량은 과일의 가격, 대체과일이 가능한 다른 과일들의 가격, 소비자의 소득, 인구량, 소비자의 기호에 따라 결정된다. 하지만 기존 이러한 원칙적인 요소를 벗어나 우리의 틀을 깨는 요소 중 하나가 바로 기상이다. 과일의 공급량과 가격은 상품특성상 강우, 온도 같은 기상조건에 의해 많은 영향을 받으며, 이러한 요소들로 인해 수요량 역시 많은 영향을 받는다.

대형과일유통업체 주씨같은 경우 항시 저렴하고 일정한 가격으로 소비자들에게 신선한 과일 음료를 제공해야 함으로 기상에 따른 과일 상품의 수급조절을 유용하게 하여야하는데, 실제로 날씨에 따른 가격변동에 따른 과일 공급이 어려울 수 있다.

본 프로젝트 연구는 머신러닝을 사용하여 데이터 학습을 통해 날씨,기후,공급,수요량을 측정하여 미래의 데이터 결과를 예측하는 연구이며, 접근성이 용이하고 편리한 구글의 Tensorflow 라이브러리의 사용과 RNN(Recurrent Neural Network) 알고리즘을 이용하여 과일들의 가격을 예측하고 그에따른 수급조절을 하는 모델을 제안 연구하고자 한다.

II. 관련 연구

1. 머신러닝(Machine Learning)

1) 머신러닝 개요

머신 러닝의 기계 학습은 인공지능 컴퓨터의 한 분야로, 패턴 분석과 컴퓨터 딥 러닝 이론의 연구로부터 진화한 분야이다. 머신 러닝은 실질적 경험한 데이터를 기반으로 하여 학습을 자동적으로 하고 예측을수행하면 스스로의 딥 러닝 기능으로 성능을 향상시키는 시스템과 알고리즘 패턴을 분석하고 구축하는 기술 중 하나이다. 머신 러닝의 컴퓨터 알고리즘들은 규격화된 정적인 프로그램 명령들을 수행하는 것이 아니라 수집한 데

이터들로 미래를 예측하거나 결정을 할 수 있도록 모델을 구축하는 방식이다.

머신 러닝은 인공지능 컴퓨터를 포함한 거의 모든 분야에서 활용되며, 컴퓨터 시각(문자, 시각 등 인식), 자연어 처리(번역, 분석), 정보 검색 및 검색 엔진, 음성 인식 및 필기 인식, 컴퓨터 그래픽 및 게임, 생물 정보학, 로봇틱스등의 분야에서 응용되고 있다.

2) 머신러닝 용어

(1) 분류(Classification) : 분류는 데이터를 정해진 카테고리에 지정하는 모델을 구축하는 것과 관련이 깊다. 이 모델들은 알고리즘이 학습할 수 있도록 미리 라벨링된 학습 데이터셋을 입력하여 구축된다. 그리고나서 그 모델을 분류가 보류된 다양한 데이터셋을 입력하여 사용하며, 학습 데이터셋으로부터 학습한 것을 근거로 분류에 대한 예측을 할 수 있다.

(2) 회귀(Regression) : 분류가 이산 클래스를 예측하는 것과 연관이 있는 반면, 회귀는 예측할 “클래스”가 연속적인 숫자값으로 구성되는 것에 대해 적용된다.

(3) 딥러닝(Deep Learning) : 딥러닝은 문제를 해결하기 위해 심층인경망 - 즉, 뉴런을 갖는 다중의 은닉레이어를 포함하는 신경망 구조 - 기술을 적용하는 과정이다. 딥러닝은 데이터 마이닝과 같이 머신러닝 알고리즘의 한 종류인 심층 신경망 구조를 도입한 프로세스이다.

(4) 신경망(Neural Networks) : 신경망은 개념적으로 연결된 수많은 인고인경들로 구성되어 있으며, 이들은 서로 데이터를 주고받고 관련 가중치들(Weights)은 네트워크의 경험에 근거하여 튜닝된다. 뉴런들은, 만약 이들의 관련 가중치들의 조합과 이들에게 공급된 데이터 값이 조건에 충족된다면, 활성화 임계값이 다음 뉴런에 전달되도록 한다. 이렇게 활성화 된 뉴런의 조합은 “학습”으로 이루어진다.

2. Tensorflow 개요

구글에서 오픈소스로 공개한 기계학습 라이브러리로써 딥 러닝과 기계학습 분야를 비전문가도 사용하기 쉽고, 다양한 기능들을 제공한다. 2016 년 이후 알파고의 등장과 함께 국내에서도 관심이 높아진 추세이며 관련 컨퍼런스들이 많이 개최되고 있다.

고급 프로그래밍 언어 중 하나인 Python 을 활용하

면 연산처리를 쉽게 작성할 수 있다. 다른 프로그래밍 언어들도 사용이 가능 하지만, Python 관련 자료가 가장 많다. 때문에 개발 되고 나서 그리 오래되지 않았음에도 불구하고 다양한 분야에서 활용되고 있다.

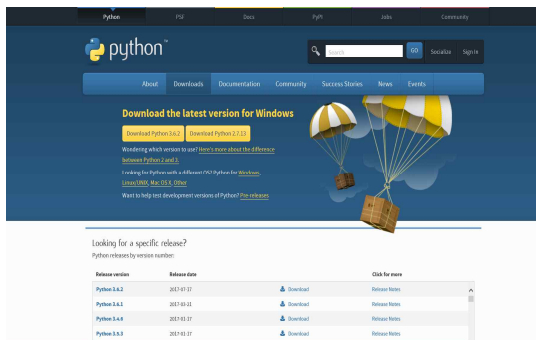
3. Python 개요

C언어를 기반으로 설계된 오픈소스 고급 프로그래밍 언어이다. 독립적인 플랫폼이기 때문에 여러 플랫폼에서 사용이 용이하고 기본으로 제공되는 많은 라이브러리가 제공된다. C, C++, Java 같은 프로그래밍 언어보다 훨씬 배우기 쉽고 문법도 간단하다. 또한 C, C++, Java보다 약 1/3 또는 1/5정도의 코드만으로도 같은 기능을 수행할 수 있어 개발 생산성이 매우 높다. 쉽게 말하여 포괄적인 명령을 내려도 컴퓨터가 알아서 일을 잘 처리한다. 이런 점들 때문에 프로그래밍하는 시간이 줄어들고 개발시간을 크게 줄일 수 있어 현재 미국의 구글, 나사, 인스타그램 등의 많은 회사 및 역동적인 어플리케이션의 회사들에서 사용 되고 있다.

Python은 통계 및 수학적 라이브러리가 다른 언어의 라이브러리보다 다양하고 쉽게 사용되기에 수학적으로 많이 사용되는 프로그램에서 사용되고 있다.

1) Python 설치 과정

윈도우 시스템에서는 TensorFlow가 Python3.5버전부터 실행 가능하기 때문에 버전이 3.5이상되는지 확인하여야 한다. <https://www.python.org/downloads/> 통해 사이트에 접속하여 컴퓨터 환경에 맞는 버전을 다운 받는다.



[그림 1] Python 공식홈페이지의 다운로드

그림 1. Python 공식홈페이지의 다운로드
 Figure 1. Download Python official homepage

4. Anaconda3 개요

Anaconda는 Python 또는 R이란 프로그램에 필요한 규격화된 정적인 프로그램 명령들을 수행하는 것이 아니라 수집한 데이터들로 미래를 예측하거나 결정을 할 수 있도록 모델을 구축하는 방식이다. 및 라이브러리 패키지 관리 등을 쉽게 해주는 도구이다. 상업용으로도 무료로 사용할 수 있어 많은 회사들에서도 사용하고 있다.

원하는 목적에 따라서 다양한 라이브러리로 짜여진 Anaconda package list가 있고, 이를 사용하여 개발환경에 필요한 여러가지 라이브러리들을 짧은 명령으로 손쉽게 설치 및 다운로드 할 수 있다는 장점이 있다.

1) Anaconda3 설치

<https://www.continuum.io/downloads#windows> 링크를 통해 사이트에 접속하여 각자 원하는 버전의 설치 파일 프로그램을 다운로드 받는다.

이번 프로젝트에서는 Windows환경에서 Python3.5 버전으로 구동하였으므로 Window용 Anaconda4.4를 다운로드 선택하여 준다.

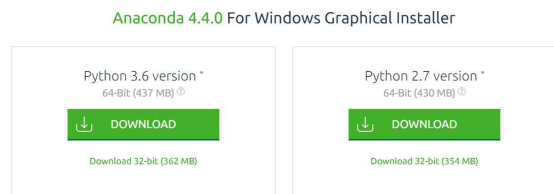


그림 2. Anaconda 다운로드 페이지
 Figure 2. Anaconda download page

5. PyCharm 개요

PyCharm은 JetBrains가 개발한 파이썬 통합개발 제품이다. 2010년에 출시되었으며, 만들어진지는 오래되지 않았다. 출시되고 나서 이는 강력한 개발도구가 되었으며 초보자도 손쉽게 Python에 입문하게 되었으며 전문가들도 기존 Python에 대한 편리함을 느낄 수 있다. 파일을 모아서 볼 수도 있으며 Python과 Python 배포판(Anaconda2, xypython, WinPython, Enthought Canopy)을 지정하여 각자 원하는 Python의 형식으로 구현 할 수 있다.

프로젝트에서는 Python은 한줄한줄 실행되는 스크립트 언어라서 오류를 색출하기가 쉽지 않을 것으로 판단되

어 디버깅 기능을 제공하는 IDE(Integrated Development Environment)를 사용하였다.

6. 공공 데이터

공공 데이터는 데이터베이스, 전자화된 내용의 파일 등 공공기관에 어떤 목적을 위해 생성 관리하고 있는 전자적인 방식으로 처리된 정보이다.

공공기관이 제공, 관리 하고 있는 자료, 정보를 말하며, 기관이 업무 수행을 하며 만들어낸 여러 형태(텍스트, 수치, 이미지, 오디오, 동영상 등)의 모든 자료와 정보가 대상이 된다. 우리나라에서는 공공데이터를 누구나 무료로 자유롭게 사용, 제공 받을 수 있도록 개방·제공을 의무화하는 것을 정부 핵심 정책으로 세웠다. 이 같은 정보는 공공데이터포털 사이트를 통해 이용가능하다.



그림 3. 기상자료 개방 홈페이지.
Figure 3. Open weather data homepage.



그림 4. 기상자료 개방 포털 자료조회
Figure 4. View weather data Open portal data

기상자료 개방포털에서 기상정보를 다운받는다.

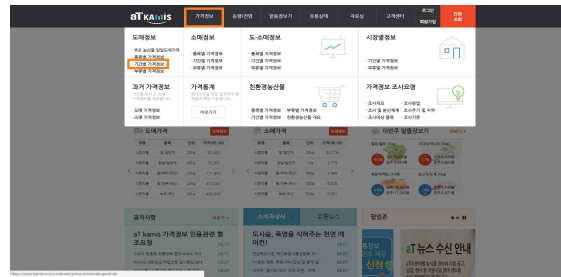


그림 5. 농산물유통정보 홈페이지
Figure 5. Agricultural Product Distribution Information Homepage

* (도매가격 - 기간별) 과일류 / 사과 / 후지, 등급: 상품, 단위: 10kg (단위: 원)

구분	08/21	08/22	08/23	08/24	08/25	08/28	08/29	08/30
평균	39,800	39,800	39,800	39,800	39,800	39,800	39,600	39,600
평년	50,711	50,711	50,711	50,711	50,711	53,356	53,356	59,933
서울	36,000	36,000	36,000	36,000	36,000	36,000	35,000	35,000
부산	38,000	38,000	38,000	38,000	38,000	38,000	38,000	38,000
대구	45,000	45,000	45,000	45,000	45,000	45,000	45,000	45,000
광주	40,000	40,000	40,000	40,000	40,000	40,000	40,000	40,000
대전	40,000	40,000	40,000	40,000	40,000	40,000	40,000	40,000

그림 6. 연간 사과 데이터 자료
Figure 6. Annual apple data

또 위 [그림 5], [그림 6] 같이 연간 사과 데이터들을 다운받아 기상청 데이터와 매칭준비를 한다.

III. Tensorflow 과일 가격 예측 구조

1. 프로젝트 구성

프로젝트 구성요소는 데이터 셋과 인공신경망 두 개로 구성되어있다. 데이터 셋에 필요한 데이터를 구축하고 인공신경망에 입력된 코드로 데이터들을 불러와 사용하면서 사용자들이 원하는 정보를 예측하는 구성이다. 데이터 셋을 구축하여 인공신경망이 예측에 필요한 데이터들을 알맞게 정제하는 일을 진행하여야 한다. 기본 데이터가 탄탄해야 예측이 잘 될 수 있으며 개발자가 원하는 결과 값이 도출되기 때문이다. 다음으로 데이터 셋 구조, 인공신경망 구조를 알아보자.

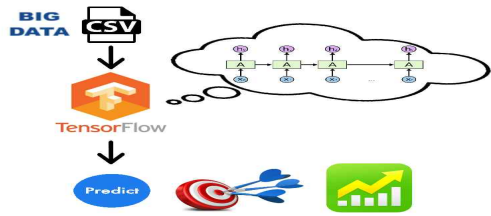


그림 7. 예측 구성도
 Figure 7. Forecast composition

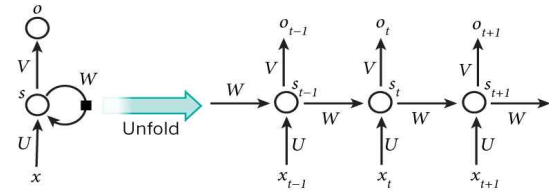


그림 9. 순환 신경망 구조순환 신경망 구조
 Figure 9. Circular neural network structure

2. 날씨와 농산물 가격 데이터 동기화

이제 저장된 날씨 데이터와 농산물 가격 데이터를 한 .csv 파일로 합쳐서 저장한다. 러닝모델이 저장된 csv파일을 불러올 때, 들어가는 input data 는 총 14가지로, [그림 8] 에서와 같이 평균가격이 합쳐진 데이터를 집어 넣는다.

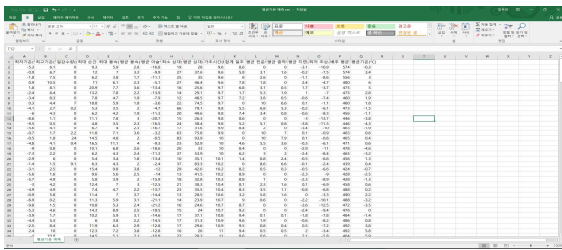


그림 8. 날씨와 과일가격 데이터동기화
 Figure 8. Synchronize weather and fruit price data

2. RNN(Recurrent Neural Network) 알고리즘

RNN에 대한 기본적인 아이디어는 순차적인 정보를 처리하는 데 있다. 기존의 신경망 구조에서는 모든 입력과 출력이 각각 독립적이라고 가정하지만, 많은 경우에 이는 옳지 않은 방법이다. 이에 대한 예로, 이 프로젝트와 같이 미래의 가격을 예측하고 싶다면 이전에 나온 가격들을 알아야만 한다. RNN이 recurrent하다고 불리는 이유는 동일한 태스크를 한 시퀀스의 모든 요소마다 적용하고, 출력 결과는 이전의 계산 결과에 영향을 받기 때문이다. 일반적인 RNN 구조는 다음과 같이 생겼다.

3. LSTM(Long Short Term Memory)

주로 LSTM 으로 불리는 Long Short Term Memory networks는 특이한 종류의 RNN이다. LSTM 은 위에서 언급한 RNN알고리즘의 장기 의존성 문제점을 해결할 수 있다. 이러한 LSTM은 Hochreiter와 Schmidhuber (1997)에 의해 소개되었다. LSTM은 장기 의존성 문제를 해결하기 위해 명시적으로 디자인되었다. 오랜 기간 데이터를 기억하는 일은 LSTM에 있어 다른 작업 없이도 기본적으로 행하는 기본 특성이다.

대부분의 RNN은 뉴럴 망의 반복되는 체인으로 구성되어있다. 표준 RNN에서 이러한 반복 모듈은 아주 단순한 구조를 가진다. 예를 들어, 단일 tanh 레이어가 있다.

4. LSTM 핵심 아이디어

LSTM의 핵심은 셀 스테이트(the cell state)이다. 다이어그램 상단에 있는 수평선이다. 셀 스테이트는 하나의 컨베이어 벨트와 같다. 이것은 아주 마이너 한 선행 연산을 거치고 전체 체인을 관통한다. 이러한 구조로 인해 데이터는 큰 변함없이 지속적으로 다음 단계에 전달된다.

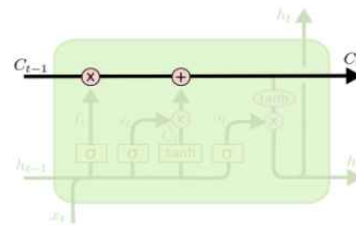


그림 10. 셀 구조
 Figure 10. The cell state

LSTM은 셀 스테이트에 정밀하게 정제된 구조를 가지고 있는 게이트라는 요소를 활용함으로 정보를 더 하거나 제거하는 기능을 가지고 있다. 게이트들은 선택 적을 정보들이 흘러들어갈 수 있도록 만드는 장치이다. 이들은 시그모이드 뉴럴넷(sigmoid neural net layer) 과 점단위의 곱하기 연산으로 이루어져 있다.

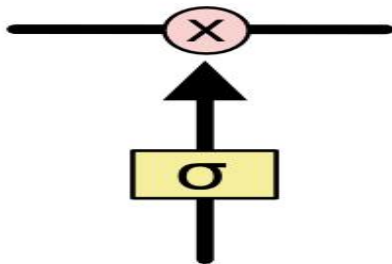


그림 11. 시그모이드 레이어
Figure 11. Sigmoid layer

시그모이드 레이어(The sigmoid layer)는 0 혹은 1 의 값을 출력한다. 그리고 각 구성요소가 얼마만큼의 영향을 주게 될지를 결정해주는 역할을 한다. 0이라는 값을 가지게 된다면, 해당 구성요소가 미래의 결과에 아무런 영향을 주지 않도록 만든다. 반면에 1이라는 값 은 해당 구성요소가 확실히 미래의 예측결과에 영향을 주도록 데이터가 흘러가게 만든다.

표 1. 가격 예측 그래프 코드
Table 1. Price forecast graph code

```
#신경망구축에 필요한 tensorflow
라이브러리와 numpy 그리고 시각화도구
matplotlib를 import 한다.
import tensorflow as tf
import numpy as np
import matplotlib.pyplot as plt

#Variable의 랜덤함수 시드를 설정한다.
tf.set_random_seed(777)

#MinMaxScaler 정의 -> data를 0부터
1사이의 값으로 변환(normalize)
def MinMaxScaler(data):
    numerator = data - np.min(data, 0)
    denominator = np.max(data, 0) -
```

```
np.min(data, 0)
    return numerator / (denominator +
1e-7)

#train Parameter 상수값 설정
seq_length = 7 #7일 단위로 학습시키고
8일째를 예측
input_dim = 14 #input 데이터의 개수
hidden_dim = 16 #은닉층의 개수
output_dim = 1 # target label의
개수(배추가격)
learning_rate = 0.01
iterations = 10001

값 설정이 끝나면 위에서 만든 데이터 셋을
불러온다. 그 후 데이터양에 따라 적절하게
Training Data와 Test Data로 분할한다.

#data load
#만들어둔 데이터셋을 Load한다.
xy = np.loadtxt('2008~2016
일간날씨+사과평균가격(대구).csv',
delimiter=',')
xy = MinMaxScaler(xy) #Normalize
x = xy #전체 데이터(input)
y = xy[:,[-1]] #target Label(사과가격)

#build dataset
#Normalize가 끝난 데이터를 7일단위로
슬라이스 해서 3차원형태로 만든다.
dataX = []
dataY = []
for i in range(0, len(y) - seq_length):
    _x = x[i:i+seq_length]
    _y = y[i+i+seq_length]
    dataX.append(_x)
    dataY.append(_y)

#train/test set 나누기
train_size = int(len(dataY) * 0.7) #train size
= 70%
```



```
test_size = len(dataY) - train_size #test size
= 30%
trainX, testX = np.array(dataX[0:train_size]),
np.array(dataX[train_size:len(dataX)])
trainY, testY = np.array(dataY[0:train_size]),
np.array(dataY[train_size:len(dataY)])
```

데이터 셋 분할이 끝났으면 해당 데이터를 받을 Placeholder와 인공신경망 구조인 LSTM을 구축한다.

```
#input placeholders
#input placeholder의 파라미터로 seq_length,
input_dim를 넘겨준다.
X = tf.placeholder(tf.float32, [None,
seq_length, input_dim]) #3차원 형태의 input
placeholder
Y = tf.placeholder(tf.float32, [None, 1])
#2차원 형태의 output placeholder
```

```
#build a LSTM network (build Rnn)
#tensorflow를 이용하면 LSTM 구조를 cell
형태로 빠르고 쉽게 생성해준다.
cell =
tf.contrib.rnn.BasicLSTMCell(num_units=hid
den_dim,
state_is_tuple=True,activation=tf.tanh)
```

```
#생성된 cell을 dynamic_rnn함수로 활성화
시켜준다. 파라미터로 생성된 cell과 input
placeholder를 넘겨주고 리턴값으로 output을
받는다.
outputs, _states = tf.nn.dynamic_rnn(cell, X,
dtype = tf.float32)
```

```
#그 후 출력된 output을 fully_connected에
통과 시켜 최종 예측값을 리턴 받는다.
Y_pred =
tf.contrib.layers.fully_connected(outputs[:,
-1], output_dim, activation_fn=tf.tanh)
```

다음 그래프들은 각각 9개월 치 데이터를 학습 후

약 3개월간 사과 가격을 예측,한 그래프를 시각화한 모습이며 실제 가격과 거의 유사하게 예측한 모습을 볼 수 있다.

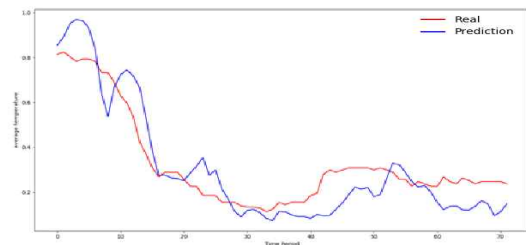


그림 12. 3개월간 사과가격 예측그래프
 Figure 12. Three-month apple price forecast graph

V. 결 론

과일 가격의 등락폭을 예측할 수 있는 점이 가져올 수 있는 기대효과로 가격의 등락폭을 이용해 해당 년도의 과일 공급량을 안정적으로 유지 할 수 있을 것이라는 점이다. 인공신경망을 통해 농산물 가격을 예측 한 결과, 농산물 가격이 크게 올라간다고 예측 그래프가 그려졌다면 해당 기간동안 어떠한 영향으로 인해 과일 수확이 원활히 진행 되지 않았다는 것을 생각 할 수 있다. 그러므로 과일 재배에 영향을 끼칠 수 있는 요인에 대한 대비를 미리 할 수 있기 때문에 과일 공급량을 안정적으로 유지할 수 있는데 기여할 수 있을 것이라고 생각된다. 그리고 날씨데이터를 정제하고 다양한 데이터변수를 추가하면 좀 더 높은 예측률을 기대할 수 있다.

References

- [1] J. O. Nam, S. K. No, E. Y. Park. "A Study on Forecasting Oyster Price with Time Series Analysis", Marine Policy Research, Vol.27 No.1 pp.65-94 June 2012
- [2] J. O. Nam, E. Y. Park, S. K. No, "Forecast for Laver Producer Price Using Time Series Models", Marine Policy Research, Vol. 29, No. 2, pp.271-303 December 2014
- [3] S. M. Ahn, Deep learning Architectures and Applications, Intelligence Information Research Center, Vol.22 No.2, pp.127-142, June 2016
- [4] J. W. Kim, H. A. Pyo, J. W. Ha, C. K. Lee, "Deep learning algorithms and applications", Korea Information Science Society, Vol. 33 No.8,

- pp.25-31, August 2015
- [5] Y. J. Song, J. W. Lee, J. W. Lee, "Performance Evaluation of Price-based Input Features in Stock Price Prediction using Tensorflow", Korea Information Science Society, Vol. 23 No. 11. pp.625-631, 2017
- [6] I. T. Joo, S. H. Choi, "Stock Prediction Model based on Bidirectional LSTMRecurrent Neural Network", Korea Information Electron Communication Technology, Vol. 11, No.2, pp.204-208, April 2018,
- [7] S. B. Lee, H. O. Kim, H. K. Seok, J. H. Nang, "Comparison of Fine-Tuned Convolutional Neural Networks for Clipart Style Classification", Vol. 9, No. 4, pp.1-7, 2017