

Hyper Parameter Tuning Method based on Sampling for Optimal LSTM Model

Hyemee Kim*, Ryeji Jeong*, Hyerim Bae*

Abstract

As the performance of computers increases, the use of deep learning, which has faced technical limitations in the past, is becoming more diverse. In many fields, deep learning has contributed to the creation of added value and used on the bases of more data as the application become more divers. The process for obtaining a better performance model will require a longer time than before, and therefore it will be necessary to find an optimal model that shows the best performance more quickly. In the artificial neural network modeling a tuning process that changes various elements of the neural network model is used to improve the model performance. Except Grid Search and Manual Search, which are widely used as tuning methods, most methodologies have been developed focusing on heuristic algorithms. The heuristic algorithm can get the results in a short time, but the results are likely to be the local optimal solution. Obtaining a global optimal solution eliminates the possibility of a local optimal solution. Although the Brute Force Method is commonly used to find the global optimal solution, it is not applicable because of an infinite number of hyper parameter combinations. In this paper, we use a statistical technique to reduce the number of possible cases, so that we can find the global optimal solution.

▶ Keyword: LSTM, Optimization, Hyper parameter tuning, Sampling

I. Introduction

최근 딥러닝은 수준이 높아짐과 더불어 다양한 활용방안이 제시되고 있다.[1] 동시에 인공 신경망 모델을 예측에 사용 시 높은 성능을 위하여 최적 모델을 찾는 튜닝 과정은 필수적이다. 가장 좋은 성능을 갖는 모델을 찾는 튜닝 방법은 모든 경우에 대한 성능을 확인하고 각각을 비교하여 최종적으로 의사 결정하는 완전 탐색법(Brute force method)이 최상의 결과를 얻을 수 있게 한다.[2] 모델을 결정하는 여러 하이퍼 파라미터 중 일부는 연속적인 값을 갖는다. 이에 기인하여 만들어 질 수 있는 모델의 경우의 수는 무한하게 되어 모든 경우의 수를 고려해야 하는 완전 탐색법을 사용하게 되면 결과를 얻는 데까지 무한한 시간을 필요로 한다.[3] 즉, 완전 탐색법을 통한 결과를 얻는 것은 불가능함을 의미한다. 일반적으로 해를 찾을 수 없는 경우

휴리스틱 알고리즘을 활용하게 된다. 인공 신경망 모델 튜닝과정에서도 역시 휴리스틱 알고리즘을 주로 사용하며 휴리스틱 알고리즘을 중심으로 발전되었다. 휴리스틱 알고리즘은 임의의 한 점에서 시작되어 확장되는 방식으로 전체가 고려되지 않을 수 있다. 휴리스틱 알고리즘을 사용한다는 것은 지역 최적값에서 최적값을 찾는 과정을 멈출 수도 있는 가능성을 안고 있다는 문제점을 수반한 상태에서 튜닝을 진행한다는 것이다.[4]

본 연구에서는 휴리스틱 알고리즘을 사용함으로써 발생하는 문제점을 해소할 수 있는 방법론을 찾는데 목적을 둔다. 이는 탐색하려는 범위 전체를 고려함으로써 결과에 대한 높은 신뢰도를 얻고자 한다. 하지만 전 범위 탐색 시 필요한 시간이 무한히 필요하기 때문에 표본 추출을 통해 그 과정에 필요한 시간을 현저히 줄이고자

• First Author: Hyemee Kim, Corresponding Author: Hyerim Bae

*Hyemee Kim (khm0219@pusan.ac.kr), Department of Industrial Engineering, Pusan National University

*Ryeji Jeong (ellipse620@pusan.ac.kr), Department of Industrial Engineering, Pusan National University

*Hyerim Bae (hrbae@pusan.ac.kr), Department of Industrial Engineering, Pusan National University

• Received: 2018. 12. 04, Revised: 2018. 12. 07, Accepted: 2018. 12. 07.

• This work was supported by the 2016 Financial Supporting Project of Long-term Overseas Dispatch of PNU's Tenure-track Faculty.

한다. 본 연구에서는 시계열 데이터를 활용할 수 있는 인공신경망 모델인 LSTM에 적용을 우선적으로 시행한다.

II. Preliminaries

1. RNN(Recurrent Neural Network)[5]

인공 신경망 모델은 크게 CNN(Convolution Neural Network)과 RNN으로 나뉜다. 그 중 RNN은 순서가 있는 데이터를 기반으로 하여 분석을 진행하는 신경망 모델이다. 여기에 적용될 수 있는 대표적인 데이터 형태는 주식 등 시간에 따라 값의 변동이 있는 데이터와 주어진 단어 뒤에 특정하게 정해진 단어가 배치되는 순서가 있는 우리가 사용하는 언어인 자연어 데이터가 있다. 일반적으로 시계열 데이터는 미래 시점에 발생할 수 있는 값의 예측에 사용되며, 자연어 데이터는 자연어 처리와 음성 인식 등 언어로서 배열순서가 중요한 분야에서 사용된다.

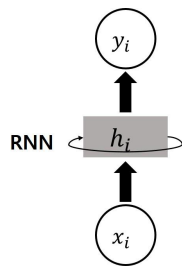


Fig. 1. Basic RNN construction

순서를 갖는 데이터를 다루는 RNN 모델은 방향을 갖는다는 것과 일맥상통한다. 즉, 어떤 데이터 뒤에 위치한 데이터가 해당 데이터 앞에 위치시켜 분석할 수 없다. RNN은 Fig. 1.과 같은 구조를 갖는다. i 시점에 입력되는 데이터 x_i 는 은닉된 상태 값인 h_{i-1} 와 함께 입력된다. 입력된 두 값을 활성화 함수에 대입시켜 새로운 값을 얻게 되는데 이 값은 은닉 상태 h_i 을 갱신시키고 출력값 y_i 를 얻게 된다. 다음 시점인 $i+1$ 에서는 x_{i+1} 와 h_i 를 입력한 후 은닉 상태 값을 갱신하고 출력값을 얻는다. 이와 같은 분석 과정을 바탕으로 적절한 은닉층의 함수를 얻기 위해 순전파(forward propagation)와 역전파(back propagation) 과정을 반복적으로 거치게 된다.

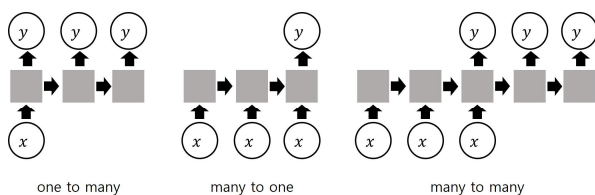


Fig. 2. Various structures of RNN

앞서 설명한 일대일(one-to-one) 형태인 Fig. 1.을 기본 형태로 다양한 모양의 RNN이 제시되었다. Fig. 2.의 첫 번째 제시 형태인 다대일(many-to-one)의 RNN은 주어진 글의 의미를 긍정 또는 부정으로 판단하고 어느 정도 수준의 감정을 갖는지 점수를 곱과 값으로 얻을 수 있는 형태분석인 감성분석 시 사용된다.[6] 다음 제시 형태는 다대다(many-to-many) 형태의 RNN이다. 예로 영어를 한국어로 번역한다고 하자. 영어와 한국어는 어순이 달라 단어 하나하나 번역하는 것이 아닌 문장 전체를 파악한 후 번역이 진행되어야 한다. 이러한 과정을 바탕으로 번역을 시행할 수 있는 형태가 다대다 RNN이다.[7] 이 외에도 일대다(one-to-many) 형태의 RNN이 사용되기도 한다.

2. LSTM(Long Short Term Memory)[8]

RNN의 활성화가 한계에 부딪힐 수밖에 없게 만들었던 문제점은 장기 의존성 문제(Gradient Vanishing Problem)이다. 장기 의존성 문제는 기준이 되는 시점의 데이터와 그 이전의 데이터 사이의 거리가 멀 경우 이전 데이터는 기준 시점 데이터에 영향을 미치지 못하는 문제를 의미한다. RNN의 장기 의존성 문제 해결을 위한 개선 모델로서 LSTM이 등장하였다.

LSTM과 RNN의 가장 큰 차이는 RNN의 은닉층이 단순한 함수형태에서 게이트(gate)를 포함하는 셀(cell)의 형태가 되었다는 것이다. 게이트는 셀에서 작용하는 여러 데이터의 활용여부를 결정하게 된다. 연산 되는 형태를 살펴보면 RNN은 모두 곱셈으로 이루어져 있어 값이 희석되지만, LSTM은 덧셈과 곱셈을 적절하게 사용하는 게이트를 추가적으로 만든 것이다. 이와 같은 과정을 추가함으로써 정보를 보다 오랜 시간 보존하여 활용할 수 있게 한다.

III. The Proposed Scheme

1. Proposed Method

본 연구에서 딥러닝의 성능을 좌우하는 하이퍼 파라미터의 최적 조합을 찾기 위해 표본 추출 결과를 토대로 최적 조합이 발생할 가능성이 적은 조합의 집단을 배제하여 빠르고 정확한 결과를 얻고자 한다. 이는 일상에서 가장 쉽게 얻을 수 있는 데이터 형태인 시계열 데이터를 활용할 수 있는 딥러닝 방법론인 LSTM에 적용한다. 구체적인 하이퍼 파라미터의 튜닝 절차는 Fig. 3.과 같다.

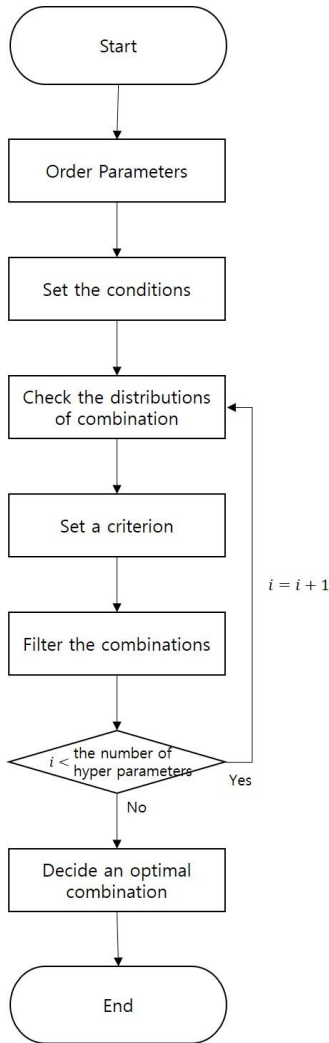


Fig. 3. Algorithm flow chart

하이퍼 파라미터 최적화를 위해 진행하는 첫 번째 단계로서 LSTM 모델의 성능에 영향을 주는 하이퍼 파라미터를 어떤 순서로 고려할 것인지 결정한다. 본 연구에서 제안하는 방법론에서는 첫 번째 반복에서 2개의 하이퍼 파라미터를 고려하며 반복이 진행될수록 고려되는 하이퍼 파라미터의 수가 하나씩 증가하여 모든 하이퍼 파라미터가 고려될 때까지 과정을 반복하게 된다. 이 과정을 진행하기에 앞서 어떤 하이퍼 파라미터를 먼저 고려할 것인지 결정되어야 한다.

고려되는 하이퍼 파라미터의 순서 배치가 완료된 후, 조건 결정 과정에서 탐색할 하이퍼 파라미터의 범위를 제한한다. LSTM에서는 활성화 함수와 같이 불연속적인 값을 갖는 하이퍼 파라미터도 있지만, 그렇지 않은 하이퍼 파라미터도 존재한다. 때문에 해당 과정 없이 발생 가능한 경우를 모두 파악하기란 불가능하다. 따라서 연속적인 값을 갖는 하이퍼 파라미터를 적은 수의 특정 값으로 결정하는 과정이 필요하다. 여기서 특정 값을 조건이라고 명시하였으며, 이에 따라 하이퍼 파라미터는 다수의 조건을 갖는다. 본 연구에서는 조건을 임의로 결정한 상태에서 연구를 진행하였다.

이전 과정에서 결정된 하이퍼 파라미터의 고려 순서와 각 하이퍼 파라미터의 조건에 의해 조합이 만들어진다. 첫 번째 반복의 경우 2개의 하이퍼 파라미터만이 고려되기 때문에 각각의 하이퍼 파라미터가 갖는 조건에 의해서만 조합이 생성된다. 생성된 조합마다 특정 횟수의 표본 추출 과정을 진행하여, 해당 과정에서 얻은 결과 값을 바탕으로 각 조합의 분포를 얻는다. 표본 추출을 기반으로 분포를 찾는 이 과정은 t-분포를 가정하여 분포를 결정하였으며, 이는 각 조합이 모델 성능을 평가한 값의 분포이다. 이때 각 조합에서 결정되지 않은 하이퍼 파라미터는 무작위로 결정되는 형태로 표본 추출이 이루어지게 된다.

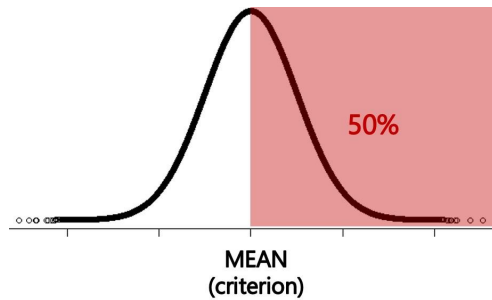


Fig. 4. Distribution of sample with criterion

기준 결정 과정에서는 각 조합의 표본 평균을 활용한다. 오차의 표본 평균을 각 조합의 성능을 평가 요소로 삼아 가장 낮은 값을 갖는 조합의 평균을 의사결정 기준으로 결정하게 된다. 기준을 포함하는 조합의 분포는 평균이 기준이므로 Fig. 4.과 같이 기준 이하에 50%가 존재하게 된다. 그 외의 다른 조합이 기준 이하에 차지하는 비율은 최대 50%까지 존재할 수 있다.

본 알고리즘은 과정을 최소화하는 것을 목표로 하고 있어, 정규 분포로 추정하기에는 부족한 수의 표본을 활용하여 분포를 추정하며 모 분산이 어떤 형태를 갖는지 알 수 없다. 이에 따라 모집단은 정규 분포를, 각 조합은 Student t 분포를 따르는 것으로 가정[9]하며 각 표본의 수와 결과 값을 활용하여 분포를 추정한다.

First loop			Second loop	
	P_{11}	P_{12}	$C_{1,2}$	$C_{2,2}$
P_{21}	$C_{1,1}$	$C_{2,1}$	P_{31}	
P_{22}	$C_{1,2}$	$C_{2,2}$	P_{32}	
P_{23}	$C_{1,3}$	$C_{2,3}$	P_{33}	

P_{ij} j th value of hyper parameter P_i

$C_{a,b}$ Combination of hyper parameter value P_{ia} and hyper parameter value P_{jb}

Fig. 5. Example of filtering combinations

결정된 기준을 바탕으로 진행 중인 과정에서 존재하는 모든 조합을 다음 과정에서의 사용 유무를 결정한다. Fig. 5.의 첫 번째 과정에서 하이퍼 파라미터 P_1 과 하이퍼 파라미터 P_2 를 고려한다고 가정하자. 이때, P_1 에서 고려되는 값은 P_{11} 와 P_{12} 두

값이며, P_2 에서 고려되는 값은 P_{21}, P_{22}, P_{23} 세 개의 값으로, 이는 ‘조건 결정’이 완료된 형태라 할 수 있다. 두 하이퍼 파라미터로부터 가능한 조합은 모두 6개이며, 이를 각각 $C_{1,1}, C_{1,2}, C_{1,3}, C_{2,1}, C_{2,2}, C_{2,3}$ 이라고 하자. 여섯 개의 조합을 하이퍼 파라미터로 갖는 모델의 조합의 성능을 알기 위해 표본 추출의 적당한 횟수만큼 학습 및 평가 과정을 각 조합별로 거친다. 이를 진행할 때, 고려되지 않은 하이퍼 파라미터들은 표본 추출 시마다 무작위로 결정되어 모델을 생성한다.

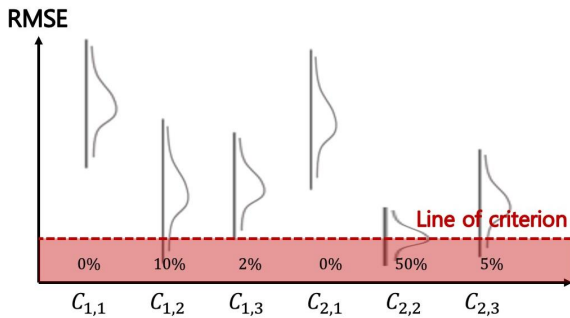


Fig. 6. Example of setting a criterion

각 조합의 표본 추출 과정이 완료되면 조합별 표본 평균을 알 수 있다. 평균값 중 가장 높은 성능을 표현하는 값을 기준으로 하여 나머지 조합을 평가한다. Fig. 6.에서는 $C_{2,2}$ 의 평균이 가장 적은 RMSE를 가지므로 이를 기준으로 삼는다. Fig. 6.에서는 $C_{2,2}$ 의 평균을 기준으로 하여 다른 조합의 분포가 기준 RMSE보다 낮은 값을 가질 확률이 10% 이상일 경우에 다음 과정에서의 해당 조합을 고려한다고 설정하였다. 이에 따라 다음 과정에서 고려되는 조합은 $C_{1,2}, C_{2,2}$ 이다. 선별된 두 조합을 하나의 하이퍼 파라미터로 보고 다음 과정에서 하이퍼 파라미터 P_3 를 추가적으로 고려하여 앞의 과정을 반복하게 된다. 이 과정을 고려하려고 하는 모든 하이퍼 파라미터에 대하여 이행하면 반복을 끝낸다.

Fig. 5.과 Fig. 6.을 통한 예시에서는 해당 분포가 기준이 되는 특정 수준보다 높은 성능을 보이는데 대한 여부를 판단하기 위해 설정한 확률은 10%였다. 이는 임의로 결정하여 과정을 진행한 것이다. 이는 사용자의 필요에 의해 결정된다. 높은 확률을 필요로 할 경우 만족하는 조합의 수가 줄어들기 때문에 보다 빠르게 결과를 얻을 수 있지만 낮은 성능을 보이며, 낮은 확률을 필요로 하는 경우 성능은 높지만 결과를 얻는데 다소 긴 시간을 필요로 한다.

반복이 끝난 후 마지막 과정에서 고려된 조합 중 가장 좋은 결과를 보이는 하이퍼 파라미터 조합을 최적 조합으로 결정한다. 이 모든 과정이 진행과 동시에 고려되는 조합의 구체화가 이루어진다. 다시 말해, 첫 번째 과정에서 하나의 조합이 나머지가 고려되지 않은 조합의 모든 경우의 수를 포함하는 반면, 마지막 과정에서는 하나의 조합은 실제로 하나의 조합만을 의미한다.

2. Experiment

금물가와 BDI 데이터를 활용하여 LSTM 모델에 적용되었으며, 이 모델의 최적 하이퍼 파라미터 조합을 찾기 위해 본 알고리즘을 활용하였다. ‘파라미터 순서 배치’, ‘조건 결정’에서 의사 결정이 필요한 부분은 임의로 결정되어 진행되었다. 고려한 하이퍼 파라미터는 총 5가지이며, 은닉층 노드의 수, 은닉층의 수, 시간 단계(time step), 셀(cell) 활성화 함수, 완전 연결(fully connected) 활성화 함수의 순서로 임의 결정하였다. 아래 Table. 1.의 내용은 임의로 결정되었으며, 이를 기준으로 각 하이퍼 파라미터 별로 고려될 조건이 결정되었다. 금물가를 활용한 첫 번째 실험의 경우 a는 4, b는 5이며, BDI 데이터를 활용한 두 번째 실험은 각각 6, 5이다. 조합 선별 과정에서 해당 조합의 존재 여부를 결정하는 기준 확률은 5%로 임의 지정하여 실험을 진행하였으며, 하나의 조합 당 20번의 표본 추출을 위해 실험이 20회 반복되었다.

Table 1. Condition of experiments

a = the number of input variables
b = any integer, $b > 0$

	condition 1	condition 2	condition 3
the number of hidden nodes	0.5a	a	1.5a
the number of hidden layers	1	2	3
time steps	b-5	b	b+5
cell activation function	sigmoid	tanh	ReLU
fully connected activation function	sigmoid	tanh	ReLU

금물가 데이터의 첫 번째 반복은 다음과 같이 이루어졌다. 첫 번째 반복에서 고려된 하이퍼 파라미터는 은닉층 노드의 수와 은닉층의 수이다. 은닉층 노드의 수의 조건은 2, 4, 6이며, 은닉층의 수는 1, 2, 3이다. 이에 따라 발생하는 조합과 조합별 평균은 Table. 2.와 같다.

Table 2. Mean RMSE from experiment’s first loop using gold price

		the number of hidden nodes		
		2	4	6
the number of hidden layers	1	0.435902	0.457579	0.426798
	2	0.463402	0.423339	0.384658
	3	0.496571	0.455801	0.428734

첫 번째 과정 진행 시 고려되지 않은 하이퍼 파라미터인 시간 단계, 셀 활성화 함수, 완전 연결 활성화 함수는 모두 무작위로 결정된다. 예를 들어, 은닉층 노드의 수가 2이고, 은닉층의 수가 1인 조합에서는 20번의 실험이 진행되는데, 이 때 확정되지 않은 나머지 세 가지 하이퍼 파라미터의 조합은 무작위로 결정되어 20번의 실험은 다양한 형태의 모델이 되어 진행된다.

Table. 2.에서 볼 수 있듯 첫 번째 과정에서 얻어진 조합별 평균값 중 가장 작은 RMSE는 0.384658이다. 즉, 첫 번째 반복

과정에서 결정된 기준은 은닉층 노드의 수가 6, 은닉층의 수가 3인 조합의 평균 0.384658이며, 각 조합의 분포에서 기준보다 작을 확률은 Table. 3.와 같다.

Table 3. Probability that the distribution is bigger than the criterion from experiment's first loop using gold price

		the number of hidden nodes		
		2	4	6
the number of hidden layers	1	0.023755	0.015174	0.067371
	2	0.012135	0.090607	0.5
	3	0.000161	0.006362	0.047751

실험 시 다음 과정에서의 고려 여부를 결정하는 확률은 5%로 설정하였다. 이에 따라 다음 과정에서 고려되는 하이퍼 파라미터의 조합은 은닉층 노드의 수와 은닉층의 수가 각각 (6, 1), (4, 2), (6, 2)이다. 이 과정을 통해 6개의 조합이 탈락하였다. 이는 $120(=6 \times 20)$ 번의 실험으로 원래 이루어져야하는 $3240(=6 \times 3 \times 3 \times 3 \times 20)$ 번의 실험을 대체하였다.

Table 4. Experiment result - performance(RMSE)

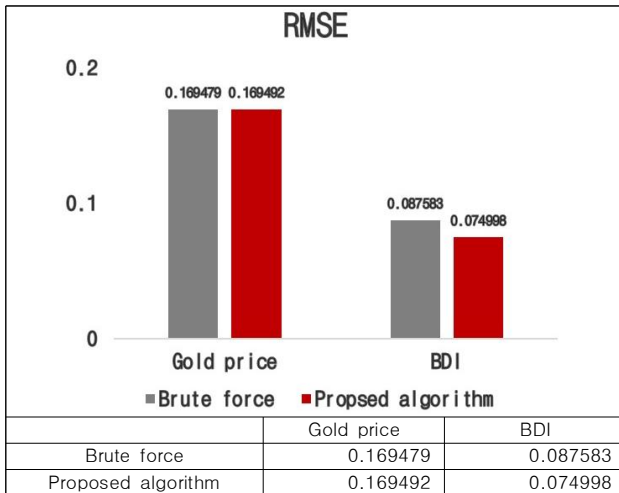
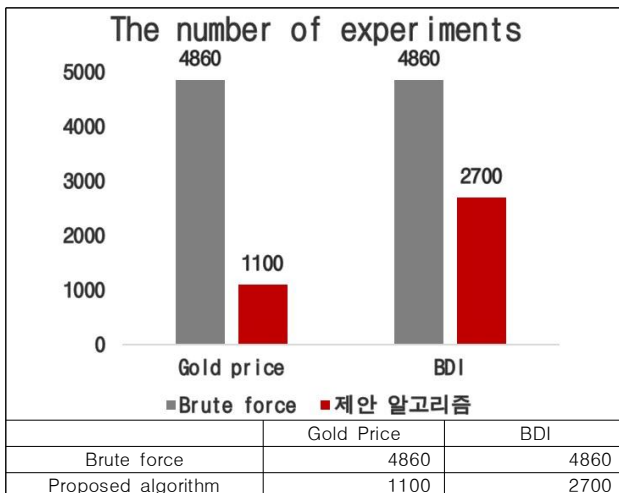


Table 5. Experiment result - the number of experiments



첫 번째로 활용한 데이터는 금물가로 1969년 12월부터 2018년 4월까지의 월 단위 데이터이다. 임의로 설정된 입력 변수는 미국, 유럽, 영국, 일본의 금물가이며 출력 변수는 미국의 금물가이다. 제한된 환경에서 완전 탐색법으로 찾아진 최적 모델의 RMSE는 0.169479이며, 제안 알고리즘의 RMSE는 0.169492로 성능 면에서 큰 차이를 보이지 않음을 알 수 있다. 반면, 완전 탐색법의 경우 4860회의 실험을 필요로 하였으나, 제안 알고리즘은 1100회의 실험만 진행되어 짧은 시간 안에 좋은 결과를 얻을 수 있었다.

BDI(Baltic Dry Index)와 관련된 데이터를 활용하여 다음 실험을 진행하였으며, 2009년 5월부터 2017년 11월까지 월 단위의 데이터를 활용하였다. BDI를 예측하기 위해 BDI, BCI(Baltic Capesize Index), BPI(Baltic Panamax Index), BSI(Baltic Supramax Index), 다우(DOW), 다우 거래량의 6가지 데이터를 임의로 사용하였다. BDI 예측 모델 역시 하이퍼 파라미터의 범위가 제한된 환경에서 튜닝 과정을 거치게 되었을 때 완전 탐색법과 제안 알고리즘을 비교하면 실험 횟수는 각각 4860회에서 2700회로, 최적 모델의 RMSE는 0.087583에서 0.074998로 성능은 큰 차이가 없지만 짧은 시간 내에 결과를 이끌었다는 것을 알 수 있다.

IV. Conclusions

인공 신경망 모델의 최적화된 형태를 찾기 위한 일반적인 튜닝 방법은 그리드 서치(grid search)와 매뉴얼 서치(manual search)이다.[10] 더불어 흔히 사용되는 튜닝 방법으로 기존에는 유전자 알고리즘(GA, Genetic Algorithm)[11], 베이저안 최적화(Bayesian optimization)[12,13], 입자 무리 최적화 알고리즘(particle swarm optimization)[14], BA(Bat Algorithm)[15], 퇴각검색(backtracking search algorithm)[16] 등의 휴리스틱 알고리즘을 사용해왔다. 많은 휴리스틱 알고리즘은 특정 점의 주변 구간만을 고려하게 되어 제한된 구간을 탐색에서 범위가 확장되는 형태를 띠게 되지만, 본 연구에서 제안하는 알고리즘은 전체 범위에서 점차적으로 범위를 좁히는 방식을 취하며 이는 이전에 인공 신경망 모델에 적용되지 않았던 실험계획법으로부터 착안되었다.

본 알고리즘은 모든 경우에서 발생할 수 있는 결과 값의 추정에서 시작되어 점차적으로 범위를 줄여 최적 값을 찾게 된다. 이는 최적 해에서 멈출 가능성이 있고 지역 최적해의 여부를 확인할 수 없는 휴리스틱 알고리즘의 단점을 보완하여 전역 최적 해에 근접한 하이퍼 파라미터 조합을 찾을 수 있다. 전역 최적 해를 찾는 기본적인 방법인 완전 탐색 방법과 성능 차이는 크지 않지만, 실험 횟수를 크게 감소시켜 보다 빠른 시간 내에 최적 해를 찾을 수 있다.

이와 같은 접근은 실험계획법의 바탕이 되는 아이디어인 고려되는 조합을 제거하는 것에서부터 시작되었다. 이를 통해 전

통적인 통계 기법인 실험계획법을 딥러닝에 적용하는 시도로서 역할 한다. 또한 산업에서 제어를 필요로 하는 다양한 하이퍼 파라미터의 집합을 최적화하고자 할 때 사용함으로써 빠르게 최적 조합을 얻을 수 있도록 돕는다. 본 연구의 실험에서 사용된 모델과 더불어 하이퍼 파라미터가 존재하는 이미지 인식 모델, 최적해 결정 모델 등에 역시 활용할 수 있다.

하이퍼 파라미터의 순서는 제안하는 알고리즘에 큰 영향을 미친다. 알고리즘 실행 초기의 조합들은 시행 시점에 고려되지 않고 있는 조건의 모든 값을 포괄하고 있기 때문에 하나의 조합은 여러 개의 조합을 내포하고 있다. 따라서 초기에 하나의 조합이 제거되면 후에 고려될 여러 조합들이 한 번에 제거되는 것이기 때문에 초기에 최대한 많은 조합을 제거하는 것이 유리하다. 이를 위해서는 조합의 결과가 넓게 분포하고 있어야 하며, 모델 성능이 민감하게 반응하는 조건이 우선적으로 고려될 수 있는 순서 배치 방법을 적용하면 더 높은 성능을 보일 것으로 예상된다.

조건의 범위를 축소하는 과정은 연속적인 값을 갖는 조건은 일정 범위 내에서 큰 차이를 보이지 않는 결과를 보인다는 가정 하에서 이루어진다. 조건의 변화에 따라 결과의 차이가 두드러지지 않는다면 모든 변화를 고려할 필요 없이 해당 범위를 대표할 수 있는 특정 값의 조건만을 평가하면 된다. 조합을 최소화하기 위해 우선적으로 조건의 범위를 축소하게 되는데 이 과정에서 분명하게 차이가 있는 대푯값을 지정하여 하이퍼 파라미터 조합이 전체를 포괄할 수 있도록 추가적인 연구가 필요하다.

REFERENCES

- [1] L. Deng, and D. Yu, "Deep learning: methods and applications," *Foundations and Trends in Signal Processing*, Vol. 7, No. 3-4, pp. 197-387, 2014.
- [2] C. Paar, and J. Pelzl, "Understanding cryptography: a textbook for students and practitioners," Springer Science & Business Media, pp. 7, 2009.
- [3] R. E. Korf, "Depth-first iterative-deepening: An optimal admissible tree search," *Artificial intelligence*, Vol. 27, No. 1, pp. 97-109, 1985.
- [4] T. Pukkala, and J. Kangas, "A heuristic optimization method for forest planning and decision making," *Scandinavian Journal of Forest Research*, Vol. 8, No. 1-4, pp. 560-570, 1993.
- [5] A. Graves, "Supervised sequence labelling with recurrent neural networks," Springer, pp. 35-42, 2012.
- [6] R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning, A. Ng, and C. Potts, "Recursive deep models for semantic compositionality over a sentiment treebank," *Empirical Methods in Natural Language Processing*, pp. 1631-1642, Seattle, Washington, USA, October 2013.
- [7] T. Mikolov, M. Karaflat, L. Burget, J. Cernocky, and S. Khudanpur, "Recurrent neural network based language model," the International Speech Communication Association, Makuhari, Chiba, Japan, September 2010.
- [8] S. Hochreiter, and J. Schmidhuber, "Long short-term memory," *Neural computation*, Vol 9, No. 8, pp. 1735-1780, November 1997.
- [9] R. V. Hogg, and E. A. Tanis, "Probability and statistical inference," Pearson Educational International, pp. 204-205, 2015.
- [10] J. Bergstra, and Y. Bengio, "Random search for hyper-parameter optimization," *Journal of Machine Learning Research*, Vol. 13, pp. 281-305, February 2012.
- [11] H. K. Lam, S. H. Ling, F. H. Leung, and P. K. S. Tam, "Tuning of the structure and parameters of neural network using an improved genetic algorithm," *Industrial Electronics Society 2001(IECON'01)*, pp. 25-30, Denver, Colorado, USA, November 2001.
- [12] J. Snoek, H. Larochelle, and R. P. Adams, "Practical bayesian optimization of machine learning algorithms," *Advances in neural information processing systems*, pp. 2951-2959, 2012.
- [13] Y. Xia, C. Liu, Y. Li, and N. Liu, "A boosted decision tree approach using Bayesian hyper-parameter optimization for credit scoring," *Expert Systems with Applications*, Vol. 78, pp. 225-241, 2017.
- [14] B. Qolomany, M. Maabreh, A. Al-Fuqaha, A. Gupta, and D. Benhaddou, "Parameters optimization of deep learning models using particle swarm optimization," In *Wireless Communications and Mobile Computing Conference(IWCMC)*, pp. 1285-1290, Valencia, Spain, June 2017.
- [15] A. Tharwat, A. E. Hassanien, and B. E. Elnaghi, "A ba-based algorithm for parameter optimization of support vector machine," *Pattern Recognition Letters*, Vol. 93, pp. 13-22, 2017.
- [16] C. Zhang, J. Zhou, C. Li, W. Fu, and T. Peng, "A compound structure of ELM based on feature selection and parameter optimization using hybrid backtracking search algorithm for wind speed forecasting," *Energy Conversion and Management*, Vol. 143, pp. 360-376, 2017.

Authors



Hyemee Kim is an undergraduate student of Pusan National University, majoring in Industrial Engineering since 2015, respectively. Kim joined the research member of business and service computing laboratory of the industrial engineering

at Pusan National University, Pusan, Korea, in 2017. She is currently an undergraduate researcher in the laboratory. Her interests lie in data mining and deep learning.



Ryeji Jeong is an undergraduate student of Pusan National University, majoring in Industrial Engineering and Global Studies since 2014 and 2016, respectively. Jeong joined the research member of the applied statistics laboratory of the industrial

engineering at Pusan National University, Pusan, Korea, in 2017. She is currently an undergraduate researcher in the laboratory. Her interests lie in applied statistics, quality control, and quality assurance.



Hyerim Bae received B.S., M.S. and Ph.D. degrees in Industrial Engineering Department from Seoul National University, Korea, in 1996, 1998, and 2002, respectively. Dr. Bae has been a professor at Department of Industrial Engineering

Pusan National University since 2004. His research interests include process mining, operational big data analytics, BPM, ICT logistics convergence.