ISSN: 2508-7894 © 2017 KAIA. http://www.kjai.or.kr
Doi: http://dx.doi.org/10.24225/kjai.2017.5.2. 29

A Study on Reliability Analysis According to the Number of Training Data and the Number of Training

훈련 데이터 개수와 훈련 횟수에 따른 과도학습과 신뢰도 분석에 대한 연구

[1]Sung Hyeock Kim(김성혁), [2] Sang Jin Oh(오상진), [3]Geun Young Yoon(윤근영), [4] Wan-Kim(김완기)

[1] First Author Department of Medical IT Marketing, Eulji University, Korea. E-mail: kyf1992@naver.com

[2,3] Department of Medical IT Marketing, Eulji University, Korea. E-mail: getita2t@naver.com

[4] Corresponding Author Graduate School of MOT, Sogang University, Korea. Tel : +2-2-705-4780, E-mail: wkkim@sogang.ac.kr

Received: June 16, 2017. Revised: June 19, 2017. Accepted: June 20, 2017.

Abstract

The range of problems that can be handled by the activation of big data and the development of hardware has been rapidly expanded and machine learning such as deep learning has become a very versatile technology. In this paper, mnist data set is used as experimental data, and the Cross Entropy function is used as a loss model for evaluating the efficiency of machine learning, and the value of the loss function in the steepest descent method is We applied the Gradient Descent Optimize algorithm to minimize and updated weight and bias via backpropagation. In this way we analyze optimal reliability value corresponding to the number of exercises and optimal reliability value without overfitting. And comparing the overfitting time according to the number of data changes based on the number of training times, when the training frequency was 1110 times, we obtained the result of 92%, which is the optimal reliability value without overfitting.

Keywords: Overfitting, Deep-learning, Tensorflow, Mnist dataset, Artificial Intelligence.

1. 서론

최근 딥러닝의 개발, 빅데이터의 활성화, 하드웨어의 발전에 힘입어 처리 가능한 문제의 범위가 급속도로 확장되면서 다양한 분야에 접목되고 있다. 특히, 과거에는 기계학습이 많이 사용되지 않았던 자연 과학 분야 등에서도 기계학습을 도입하려는 움직임이 나타나면서 기계학습은 인공지능이나 패턴 인식 등에 쓰이는 것을 넘어서 매우 범용적인 기술이 되었다. 기계학습에서의 결정경계(decision boundary)가 완벽하게 구분하지 못하여 잘못 분류된 데이터가 생긴다. 그러면 결정경계를 직선으로 나타내지 않고 모든 데이터를 완벽하게 구분할 수 있도록 곡선으로 나타내면 더 좋은 시스템이 될 것이다. 하지만 이때 기계학습의 과도학습(overfitting) 문제가 발생한다. 일반적인 데이터에는 노이즈가 섞여 있으므로 주어진 학습데이터에 맞춰 시스템을 과도하게 학습시키는 것은 오히려 시스템의 성능을 저하한다. 따라서 기계학습 시스템의 복잡도는 풀고자 하는 문제의 복잡도와 일치해야 한다(Moon et al., 2016).

본 논문에서는 Mnist 데이터를 사용하고 교차 엔트로피를 이용한 분석을 하였으며 훈련데이터의 개수와 데이터 훈련 횟수에 따라 변화되는 신뢰도 값을 측정하여 Overfitting이 일어나는 값을 도출하였다. 그리고 이에 따른 향후 보완 방향을 제시한다.

2. 관련연구

2.1. 과도학습(Overfitting)

과도학습은 훈련하고자 하는 데이터를 너무 과도하게 학습한 경우를 뜻한다. 기존의 훈련한 데이터뿐만 아니라 새로운 데이터에 관하여 정확한 분류를 해야 하지만 분류의 정확성이 낮으면 문제가 발생한다. 여기서 과도학습이 발생하는 경우는 훈련 데이터의 신뢰도 값이 실험 데이터의 신뢰도 값보다 큰 경우이다.

2.2. 딥러닝(Deep learning)

딥러닝이란 인간 뇌의 학습처리 과정을 모방한 기계학습방법의 한 종류로, 사람의 사고방식을 컴퓨터에 학습시키는 것을 의미한다<Figure 1>. 1980년대 등장한 인공신경망(ANN:artificial neural network)에 기반을 두어 설계된 개념으로, 정보통신기술의 발전과 함께 단점으로 여겨지던 과적합 문제와 느린 계산속도 등의 한계를 극복하게 되며 관심이 증가하게 되었다. 기계학습과 딥러닝 모두 사람이 기계에 어떻게 학습할지를 세세하게 알려주는 것이며, 완성된 딥러닝 알고리즘의 경우 상대적으로 사람의 간섭 없이 컴퓨터 스스로 학습하는 비지도 학습(unsupervised learning)의 한 종류를 의미한다(Kim, 2016)

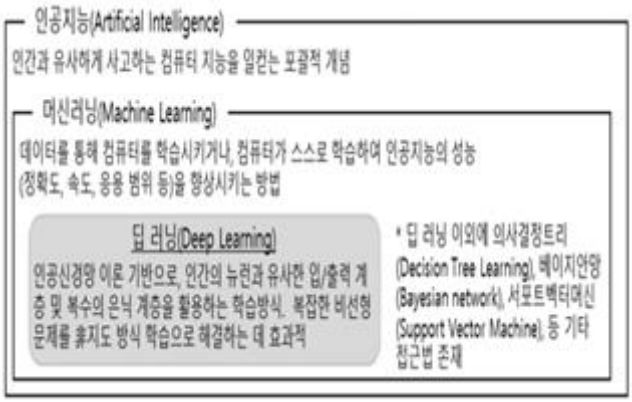


Figure 1. Concepts of artificial intelligence, machine learning, and deep learning

2.3. 텐서플로우(Tensorflow)

구글의 딥러닝 프로젝트 중 하나인 텐서플로우는 리눅스나 맥의 운영체제인 OS X 뿐 아니라 심지어 Bash가 설치된 윈도우 10 에서도 구동되는 2세대 기계학습 플랫폼이다. 텐서플로우는 이미지, 음성, 비디오 등의 대용량 데이터를 처리하며, 기계학습을 수행할 때 고속 병렬처리가 가능한 GPU도 활용한다. 텐서플로우의 핵심 알고리즘은 C++로 절차를 진행하기 위한 프론트엔드는 파이썬(Python)으로 작성되었다. 일반적인 지도학습 과정은 Figure 2와 같이 먼저 학습하고자 하는 데이터를 준비하고 이후 데이터를 가지고 학습을 수행한다. 이후 학습 중인 모델이 overfitting이나 under-fitting 인지를 검증한 후, 최적의 적합(generalized-fitting) 구간을 찾아 최종 학습모델을 정한다. 일련의 학습 과정을 모두 완료한 후 시험 데이터를 통해 최종 학습 모델의 성능을 평가한다(Kim et al., 2017).

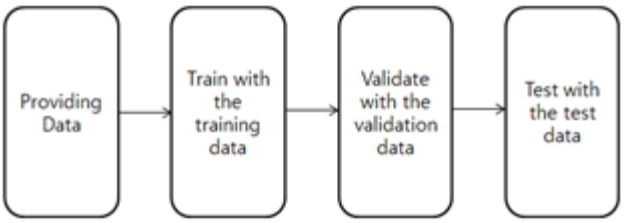


Figure 2. General guidance learning procedures

3. 실험 및 결과

3.1. 제안방법

손실함수는 신경망 성능의 '나쁨'을 나타내는 지표로, 현재의 신경망이 훈련데이터를 얼마나 잘 처리하지 '못'하느냐를 나타낸다. 손실 함수의 종류로는 평균 제곱 오차와 교차 엔트로피 오차, 미니배치 학습이 존재

한다. 본 논문에서는 Saito(2017) 에서의 방식과 동일하게 분석하고자 하는 데이터의 신뢰도를 교차 엔트로피 오차를 이용하여 분석하였다. 교차 엔트로피 오차의 수식은 다음과 같다.

$$E = -\sum_{k} t_k \log y_k \quad (1)$$

상기 Fomula 1은 Cross Entropy 수식으로 신경망의 출력 즉, 추정 값을 나타내며 정답 레이블을 나타낸다. 정답레이블은 정답일 경우 원소의 값이 1, 오답일 경우 0을 취하고 있는 원-핫코딩 표기법을 사용한다 (Saito, 2017). 본 논문에서는 이와 같은 교차 엔트로피 오차 방법을 Python으로 구현하여 신뢰도 분석 실험을 진행하고, Overfitting 발생을 검출하여 최적의 학습 데이터 환경을 제안하고자 한다.

3.2. 실험 및 결과

실험 환경으로 Intel(R) Core(TM) I5-4200u 1.60ghz(4CPUs)~2.3ghz의 CPU와 13z940FI의 메인보드, DDR3 4GB RAM과 128GB SSD로 구성된다. 운영체제는 64bit 윈도우 10 PRO 버전을 사용한다. 본 논문에서 실험데이터는 MNIST 데이터로 기계학습 분야에서 자주 쓰이는 데이터 셋이며, 간단한 실험부터 논문으로 발표되는 연구까지 다양한 곳에서 이용한다. 손 글씨(숫자) 이미지로 구성된 컴퓨터 비전 데이터 셋으로 훈련 이미지 60,000장, 시험 이미지 10,000장으로 구성된다<Table 1>. 일반적으로 이들 훈련 이미지를 사용하여 모델을 학습하고, 학습한 모델로 시험 이미지들을 얼마나 정확하게 분류하는지 평가한다. MNIST의 이미지 데이터는 28X28 크기의 회색조 이미지(1채널)이며, 각 픽셀은 0~255까지의 값을 취한다. 각 이미지에는 또한 ‘7’, ‘2’, ‘1’ 과같이 그 이미지가 실제 의미하는 숫자가 레이블로 데이터가 구성된다(Saito, 2017).

Table 1. Configuration of mnist data

Data set	Matrix	Data Source	Remark
mnist.train.images	55000 × 784	Learning image data	
mnist.train.labels	55000 × 10	Learning label data	
mnist.test.images	10000 × 784	Tested image data	
mnist.test.labels	10000 × 10	Tested label data	
mnist.validation.images	5000 × 784	Confirmed image data	
mnist.validation.labels	5000 × 10	Confirmed label data	

3.2.1. 훈련 횟수에 따른 신뢰도 분석 실험

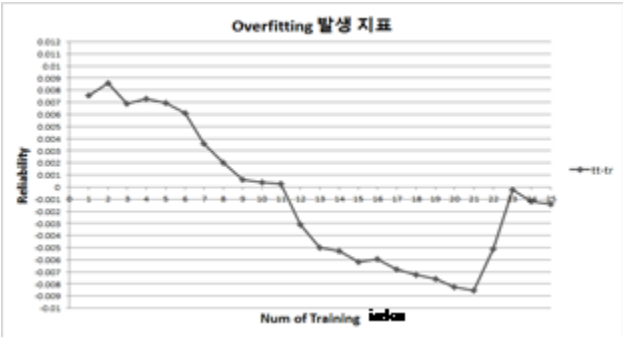
본 논문에서 제안한 훈련 횟수에 따른 신뢰도 실험을 하기 위해 텐서플로우(tensorflow)에서 제공하는 mnist.train 데이터를 통해 학습하고 mnist test 데이터를 통해 신뢰도를 확인하였다. 신뢰도 확인을 위하여 이미지를 분류하기 위한 분석모델로 텐서플로우에서 제공하는 tf.nn.softmax라는 함수를 통하여 소프트맥

스 회기분석(softmax regression) 기법을 사용하였다. 또한, 기계학습의 효율을 평가하기 위한 손실 모델로 Cross entropy 함수를 이용하였으며, 경사 하강법으로 손실함수 값을 최소화하기 위해 GradientDescentOptimize 알고리즘을 적용해가면서 backpropagation을 통해 weight와 bias를 업데이트하였다. 이에 따라 correctprediction에 기댓값과 정답이 동일한지를 확인하며 평균을 내 출력을 하였고 Accuracy에 테스트 데이터를 입력하여 확인한다.

Index	Num of Training
1	1
2	10
3	100
4	120
5	150
6	250
7	500
8	750
9	1000
10	1100
11	1110
12	1111
13	1112
14	1113
15	1114
16	1115
17	1116
18	1117
19	1118
20	1119
21	1120
22	1130
23	1200
24	1500
25	1750

Figure 3. Index conversion of number of training

총 55,000개의 학습데이터와, 5,000개의 확인용 데이터, 10,000개의 테스트용 데이터를 기준으로 위와 같은 과정을 통해 훈련횟수에 따른 신뢰도 값을 분석 하였으며, 실험에 앞서 위 Figure 3과 같이 훈련 횟수는 인덱스로 변환하여 실험을 진행하였다.



<Figure 4> Overfitting index according to training frequency

Overfitting이 발생한 시점을 분석하기 위해 훈련 데이터 신뢰도(tr, training data - reliability), 테스트 데이터 신뢰도 (tt, test data - reliability)를 나누어 신뢰도 변화를 측정하였다. 하지만 0~9의 10개로 구성된 mnist data의 특성상 훈련 데이터 신뢰도와 테스트 데이터 신뢰도의 차이를 명확히 나타내기 위하여 두 수치의 오차를 Overfitting 발생 지표로 표현하였고 이는 Figure 6과 같다. Overfitting은 훈련 데이터 신뢰도 수치가 테스트 데이터 신뢰도 수치보다 클 경우 발생하기 때문에 음수를 나타낼 경우 Overfitting이 발생한 것을 확인하였고, 양수를 나타낼 경우 Overfitting이 발생하지 않는 generalized-fitting 구간을 확인하였다. 이 때 x 축은 훈련 횟수의 인덱스를 나타내며 실제 훈련 횟수와 인덱스의 관계는 Figure 4와 같다. Figure 4와

Figure 5를 분석하면 인덱스값 21(1,121회) 기준에서 가장 높은 신뢰도가 도출되었다. 하지만 동일 구간에서 Overfitting이 발생하는 것을 확인하였으며 Overfitting이 전혀 발생하지 않는 선에서 최적의 신뢰도는 인덱스값 11(1,110회) 기준에서 0.921709 정도의 신뢰도가 도출되었다.

3.2.2. 훈련 데이터 개수에 따른 신뢰도 분석 실험

Index	Number_of_Data
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	10
11	11
12	12
13	13
14	14
15	15
16	50
17	100
18	1000
19	5000
20	10000

Figure 5. Index conversion of number of training data

훈련 데이터 개수 기준에 따른 신뢰도 분석 실험을 하기 위해 위와 같은 조건에서 나온 최적의 훈련횟수였던 1,110회는 시스템 성능의 문제와 과도한 학습시간으로 인해 실험에 비효율이 있다고 판단하여, 기준과 같은 조건에서 사전 연구보다 적은 훈련 횟수인 150회 기준으로 실험을 진행하였다. 훈련 횟수에 따른 신뢰도 분석 실험과 같이 먼저 데이터 개수를 인덱스로 변환하였고, 이는 위 Figure 5와 같다. 실험을 위하여 훈련 횟수에 따른 신뢰도 분석 실험과 같이 mnist data set을 이용하여 각각 훈련횟수 1100횟수기준, 150회 기준에서 데이터 개수에 따른 신뢰도 변화를 실험하였고 결과는 아래와 같다.

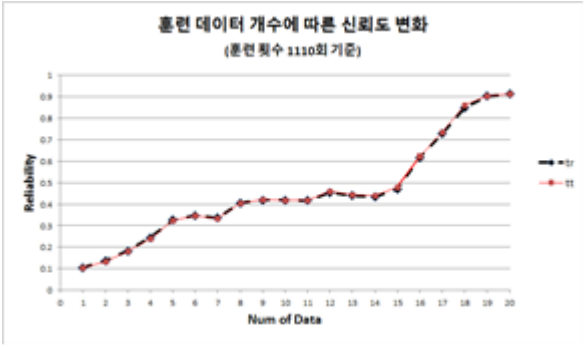


Figure 6. Change of reliability according to number of training data(1000 training standards)

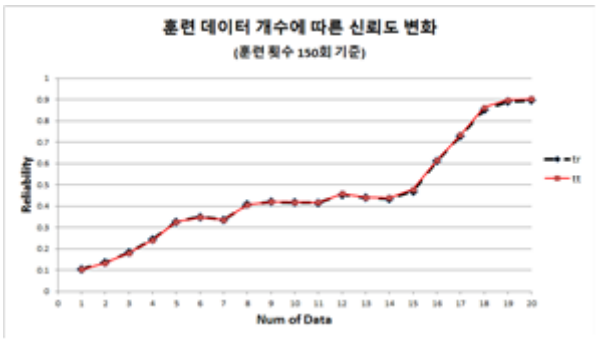


Figure 7. Change of reliability according to number of training data(150 training standards)

Figure 6와 Figure 7에서 확인 할 수 있듯이 실험 결과 훈련 데이터 신뢰도(tr)와 테스트 데이터 신뢰도(tt)의 변화에 있어서 훈련횟수 1110회 기준과 150회 기준의 신뢰도 변화의 그래프 모형이 비슷한 것을 알 수 있었으며, 비슷한 시점에서 Overfitting이 발생 하였다.

확실한 Overfitting 발생 구간을 검출하기 위하여 Figure 6의 훈련 횟수에 따른 신뢰도 분석 실험과 마찬가지로 Overfitting 시점이 보이지 않는 문제 해결을 위해 훈련데이터 신뢰도(tr)과 테스트 데이터 신뢰도(tt)의 차이를 그래프로 나타냈고, 이는 아래 Figure 8과 Figure 9과 같다.

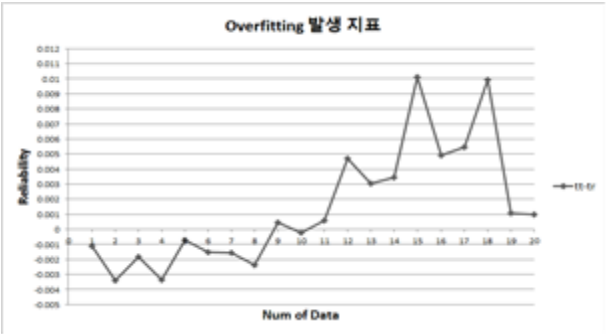


Figure 8. Overfitting index according to number of training data(1000 training standards)

Figure 9. Overfitting index according to number of training data(150 training standards)

Figure 9와 같이 훈련 횟수가 1110회 기준으로 실험한 결과 1 ~ 10개의 데이터로 훈련하였을 때,

Overfitting이 검출되었고, 11개의 데이터부터 Overfitting이 검출되지 않았다. 또한, 훈련 횟수가 150회 기준으로 실험한 결과 1 ~ 8개의 데이터로 훈련하였을 때, Overfitting이 검출되었고. 9개의 데이터부터 Overfitting이 검출되지 않았다. 그리고 50개, 100개, 1000개, 5000개, 10000개의 데이터를 분석한 경우에도 계속해서 Overfitting이 검출되었다.
훈련 횟수가 1110회 기준인 Figure 8과 150회 기준인 Figure 9를 비교 분석한 결과 1110회에서는 인덱스값 8(8개) 이전에 Overfitting이 검출되었고 150회에서는 인덱스값 9(9개) 이전에 Overfitting이 검출되었다. 상기와 같이 비슷한 시점에서 Overfitting이 검출된 것을 확인하였고, 본 논문에서는 훈련 데이터 개수 차이에 따라 신뢰도 값은 차이가 나지만 Overfitting이 일어나는 시점이 비슷하다는 것을 확인하였다.

4. Conclusion

딥러닝의 핵심은 새로운 문제에 대한 정확한 답을 찾아내는 것이다. 하지만 Overfitting은 기존의 학습에 너무 익숙해진 '과도학습' 문제로 실험 데이터에 대한 신뢰도 값이 훈련 데이터의 신뢰도보다 낮게 되는 문제가 발생한다. 어떠한 알고리즘이나 함수를 이용하던, 데이터 개수와 학습횟수는 어느 조건에서든 필요하며, 때문에 데이터 개수와 학습횟수를 중점으로 연구를 진행하게 되었다. 본 논문에서는 MNIST 데이터 셋을 이용하여 훈련 횟수에 따라 최적의 신뢰도 값을 찾아낼 수 있었으며, 이는 훈련횟수 1121회에서 신뢰도 92.5%로 가장 높았고, Overfitting이 일어나지 않는 최적의 신뢰도 값은 훈련횟수 1110회에서 92.1%였다. 훈련데이터 횟수에 변화가 있을 경우 훈련 횟수 1110회 기준에서 1~10개의 데이터로 훈련하였을 때 Overfitting이 검출되었고 11개의 데이터부터 Overfitting이 검출되지 않았으며, 훈련 횟수가 150회 기준으로 실험한 결과 1~8개의 데이터로부터 Overfitting이 검출되었고 그 이후엔 검출되지 않았다. 이를 통해, 1110회 기준 8개, 150회 기준 9개의 유사한 시점에서 Overfitting이 검출된 것을 확인하였고 훈련 데이터 개수에 따라 신뢰도 값은 차이가 나지만 유사한 시점에서 Overfitting이 검출되는 것을 확인하였다. 추후 연구에서는 1,110회 훈련 기준 데이터 개수 8개에서 Overfitting이 검출되었지만 데이터 개수 9개에서 Overfitting이 발생하였고 데이터 개수 10개부터 Overfitting이 발생하지 않은 점을 해결할 것이며, 컴파일 속도를 향상시켜 55,000개의 전체적인 데이터 개수에 따른 신뢰도 값을 찾아 세부적인 신뢰도 분석 실험을 진행하고, 훈련횟수와 개수가 아닌 신뢰도를 향상하기 위해 존재하는 다양한 방법 고려하는 실험을 병행할 것이다.

References

Kim, H. M. (2016). AlphaGo's Deep Learning applied financial case. *KB Knowledge Vitamin,* 16(31),1-2, Retrieved Jun 16, 2017, from https://www.kbfg.com/kbresearch/index.do?viewFunc=default_details&categoryId=3&articleId=1003274

Kim, J. K., Kim, Y.H., Lee, M.S., & Ahn, J. M. (2017). Performance Evaluation Methodology on BPSK with

Tensorflow, *The Proceeding of the Korean Institute of communications and Information Sciences 2017*, Winter, 330-340.

Moon, S. E., Chang, S. B., Lee, J. H. & Lee, J. S. (2016). Technological Trends for Machine Learning and Deep Learning, *The Korean Institute of communications and Information Sciences (Information & Communications Magazine), 33*(10), 49-56.

Saito, K. (2016). *Deep Running Starting from the Bottom.* Seoul, Korea : Hanbit Publishing Network, 113-115.