

Backpropagation 신경망의 학습속도 개선방법에 대한 제언

조 성배^o, 윤 현수, 김 진형

한국과학기술원 전산학과 인공지능연구실

Some Remarks on Speedup Methods of Backpropagation Learning

Sung-Bae Cho, Hyunsoo Yoon and Jin H. Kim

Artificial Intelligence Lab., Dept. of Computer Science, KAIST

요 약

기존의 방법으로 풀기 어려운 여러가지 문제를 해결하기 위하여 최근에 인간의 두뇌구조를 모델링한 신경망이 새로운 계산방법으로 도입되고 있다. 하지만, 신경망이 현실적인 문제에 적용되기에는 그 학습 속도가 너무 느리다는 단점이 있다. 따라서, 신경망의 학습에 가장 널리 사용되고 있는 Backpropagation과 같은 학습 알고리즘의 속도를 개선하기 위하여 알고리즘 자체의 체계적인 연구가 필요하며, 이와 같은 연구를 통한 보다 개선된 학습방법이 절실한 실정이다.

본 논문에서는 Backpropagation 학습방법의 속도를 개선하고자 하는 이제까지의 연구를 정리하고 분석하며, 이를 토대로 좀 더 빠른 새로운 방법의 제시에 필요한 고려사항에 대하여 논의하고자 한다.

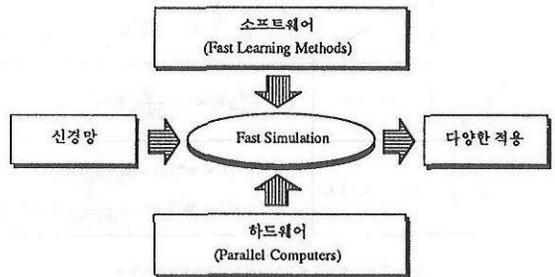
I. 서 론

최근에 인간의 두뇌 신경조직을 모델로 하여 단순한 기능을 하는 처리기들의 대규모 상호 연결을 통하여 문제를 해결하고자 하는 신경망이 새로운 계산기법으로 주목받고 있다. 신경망은 주어진 문제를 해결할 수 있는 알고리즘을 알아내기 어려운 경우에도 자체의 적응성에 의해 입력력 관계를 자동적으로 생성해 낼 수 있는 학습 능력(learning capability)과 자체의 일반화 능력(generalization capability) 때문에, 기존의 방법으로 해결하기 어려운 문제에 적용하려는 노력이 계속되고 있다. 그 결과로 해당 문제영역에 적합한 많은 신경망 학습 알고리즘이 제안되어 사용되고 있지만, 그 중에서도 Backpropagation은 그 자체의 단순함에도 불구하고 가장 널리 이용되고 있다.

학습 알고리즘의 성능을 평가하는 보편적인 기준으로는 학습 속도와 일반화 정도를 들 수 있는데, 전자에 비해 후자의 기준이 풀고자 하는 문제의 성격에 크게 의존하는 경향이 있기 때문에 이제까지의 많은 연구가 학습 알고리즘의 속도를 향상시키고자 하는데 집중되고 있다. 특히, 아직까지 일반적으로 사용될 수 있는 신경망 컴퓨터(Neuro-computer)가 제시되지 않고 있어 대부분의 경우에 시뮬레이션에 의해 신경망을 사용하고 있어서 이의 연구가 활발히 진행되고 있는데, 신경망 시뮬레이션의 한계극복을 위한 접근방법으로는 하드웨어적으로 병렬처리 구조를 이용하는 것과 소프트웨어적으로 빠른 학습 알고리즘을 고안하는 것이 있을 수 있다. 바람직한 것은 <그림 1.1>과 같이 두 접근방법의 상호 보완에 의해 보다 빠르게 학습시키는 것이 라 하겠다.

본 연구에서는 이와 같은 맥락에서 Backpropagation 신경망의 학습과정을 분석하고, 기존의 속도개선 방법을 정리하여 비교, 분석하고자 한다. 또, 이제까지 발표된 학습속도 개선방법의 시간 및 기억장소 복잡도의 분석을 통해 새로운 학습속도 개선방법의 비교기준을 설정할 수 있을 뿐만 아니라, 보다 빠른 학습을 위해 병렬처리에 적합한 알고리즘의 가능성도 모색해 볼 수 있을 것이다.

본 논문의 내용을 살펴보면, II장에서는 Backpropagation 신경망의 학습과정을 분석하여 기존의 개선방향을 알아보고, III장에서는 이러한 기존의 연구들을 비교, 분석한다. 이와 같은 분석을 통해 IV장에서는 새로운 속도개선 방법의 도출을 위한 토의사항을 논의하며, 마지막으로 V장에서 결론 및 향후 연구방향을 기술한다.



<그림 1.1> 학습속도 개선의 두 접근방법

II BP 신경망의 학습과 속도개선 방법

2.1 Backpropagation 알고리즘

신경망의 기본 아이디어는 데이터로 주어진 입력과 출력의 쌍에 존재하는 사상(mapping)을 자체의 적응성(adaptability)에 의하여 생성해 내는데 있다. 이때, 신경망이 제대로 동작하기 위하여 연결강도(connection weight)를 반복법에 의해 조정해 나가는 것을 학습이라고 하는데, 매리미터를 추정하는 전통적인 방법으로는 실제 출력과 원하는 출력 사이의 차이를 최소화 시키는 것으로 여러가지 유용한 최적화 기법이 사용될 수 있으며, 그 중에서 가장 널리 사용되는 것으로는 경사정보(gradient information)를 이용하는 Backpropagation을 들 수 있다

Backpropagation은 지도 학습(supervised learning) 알고리즘으로서 이상적인 출력과 실제 출력사이의 평균 제곱 오차에 해당하는 비용함수의 값을 최소화하기 위하여 경사추적 방법을 사용한다. 이것은 결국 (식 2.1)과 같은 신경망의 전체 오차 E를 최소화 하도록 하는 연결강도를 구하기 위해 (식 2.2)와 같은 반복식을 적용하는 것이다

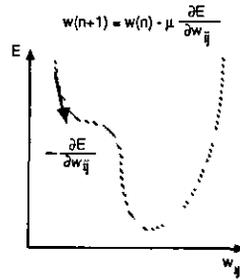
$$E = \sum E_p = \sum \sum (t_p - o_p)^2 \quad \text{----- (식 2.1)}$$

$$W(n+1) = W(n) + \Delta W(n) \quad \text{----- (식 2.2)}$$

(식 2.1)에서 t_p 는 주어진 입력에 대한 올바른 출력이고, o_p 는 $o_p = \sum_j w_{ij}$ 와 같은 계산에 의하여 구해진 현재 신경망의 실제 출력이다. Backpropagation의 연결강도 변경식은 앞에서 설명한 바와 같이 $W(n+1) = W(n) + \Delta W(n)$ 인데, 이것은 수치해석적인 고정점 반복법(fixed point iteration)에 의하여 전체 오차를 최소화 하도록 하는 연결강도의 집합 W를 구하는 것이다. 일반적인 비선형 최적화 문제와 Backpropagation의 비교가 <표 2.1>에 있으며, 두 방법의 일반적인 문제해결 방법은 다음과 같다

```
for (k=0, evaluate(w_k) != CONVERGED, k++) {
    d_k = determine_search_direction (),
    alpha_k = determine_step (),
    w_{k+1} = w_k + alpha_k d_k,
}
```

이러한 식의 동작원리를 알아보기 위하여 <그림 2.1>은



<그림 2.1> 일차원상의 오차곡선

일차원 상에서의 오차곡선을 나타내고 있다. 이 그림은 신경망의 오차를 연결강도의 식으로 표현하였을 경우의 오차곡선이며 이 경우에 학습은 곡선상의 임의의 초기 지점에서 오차를 최소로 만드는 방향, 즉 최대 경사(gradient descent) 방향으로 적당한 정도로 움직여 나가면서 전체 오차가 최소로 되는 지점의 연결강도를 구하는 것이라 볼 수 있다

2.2 학습속도의 개선방법

Backpropagation은 비교적 간단한 식을 사용함에도 불구하고 많은 응용 분야에서 좋은 성능을 나타내는데 고무되어 이의 특성을 분석하고자 하는 연구가 많이 진행 되고 있으며, 최근에 le Cun은 Backpropagation의 역사에 대해 간략히 정리하고 이의 신경망적 해석을 통한 정형화를 시도하였다 [10]. 하지만, Backpropagation은 비선형 최적화 입장에서 단순한 경사 추적 기법을 사용하기 때문에 문제의 크기나 복잡도가 커지는 현실적인 문제를 풀기 위하여 사용하고자 할 경우에는 학습시키기 위한 시간이 증가하고, 인식능력이 크게 떨어지는 단점이 있다. 이와 같은 문제를 해결하기 위해 Backpropagation의 수렴속도를 개선시키고자 하는 많은 연구가 있었는데, 이를 정리해 보면 다음과 같이 크게 세가지 방향으로 나누어 볼 수 있다

첫째는 경사 추적 기법을 신경망의 학습에 적용하면서 얻은 여러가지 경험적 지식(heuristic knowledge)을 이용하여 학습속도를 빠르게 하고자 하는 방법으로, 그 기본 아이디어는 오차곡선의 형태에 상관없이 같은 정도의 학습률로 다음 단계의 연결강도 값을 구하는 것을 Backpropagation의 학습속도를 느리게 하는 주요인으로 보고, 이를 개선하기 위해 학습률을 동적으로 변경시키거나 [8], 경사도 값을 재사용하는 [7]등의 방법을 사용하는 것이다

둘째는 신경망의 학습이 비선형 최적화 문제를 푸는 것 [1]이라는 아이디어에서 이를 수치해석적인 방법으로 개선하고자 하는 것으로, 대부분의 연구가 반복법에 의해서 다음 단계의 연결강도 값을 구할때 경사도 이외에 이차 미분값을 이용하는 이차 제수 방법 (2nd order method) [14]의 변형된 형태이다

셋째는 직접적인 학습 알고리즘의 개선보다는 학습에 사용되는 패턴의 형태나 순서를 조정하는 방법으로 학습중에 학습이 잘 안되는 패턴을 좀 더 학습에 참여 시킨다거나 [11], 훈련 패턴사이의 상관도 (degree of correlation)를 줄이는 전처리 후에 학습시키는 [13]등의 방법이 있다

다음 장에서는 이들 방법들에 대해 좀 더 자세히 알아보고 이들의 시간과 기억장소 사용정도를 분석해 본다

	Nonlinear optimization	Backpropagation approach
Objective function		$E(W) = \frac{1}{2} \sum_p \sum_i \left[\left(\sum_{j=1}^n W_{ij}^{(m)} \left(\sum_{k=1}^n W_{kj}^{(m)} X_k \right) \right) - Y_i \right]^2$
Function to be solved	$f(x) = 0$	$\frac{\partial}{\partial W^{(m)}} E(W) = \frac{\partial}{\partial W^{(m)}} E(W) = 0$
Iterative formula	$x_{k+1} = x_k - f(x_k)$	$W^{(m)}(n+1) = W^{(m)}(n) - \eta \frac{\partial}{\partial W^{(m)}(n)} E(W)$ $W^{(m)}(n+1) = W^{(m)}(n) - \eta \frac{\partial}{\partial W^{(m)}(n)} E(W)$

단 $W^{(m)}$ 은 입력층과 중간층 사이의 연결강도를 의미하며,
 $W^{(m)}$ 은 중간층과 출력층 사이의 연결강도를 의미한다

<표 2.1> 비선형 최적화 문제와 Backpropagation의 비교

III. 학습속도 개선방법의 분석

3.1 경험적 방법

Backpropagation의 학습속도 향상을 위하여 가장 많이 연구되고 있는 것으로는 각 연결강도의 값을 신경망의 전체 오차를 가장 크게 감소시키는 방향으로 반복해서 바꾸어 나가는 과정을 체계적으로 분석하여 이로부터 얻어진 경험적 지식을 사용하는 방법이 있다. 이 부류에 속하는 대부분의 방법들은 최소화하고자 하는 오차곡선의 특성이 Backpropagation의 경사 추적 기법의 속도를 느리게 한다는 점을 해결하고자 하였다.

3.2 수치해석적 방법

Backpropagation으로 가능한 빠르게 수렴시키기 위해서는 최소점을 overshooting하지 않는 한도 내에서 연결강도의 값을 크게 변화시켜야 한다. 그런데, Backpropagation에서 사용하는 한 점상에서 얻어지는 일련의 편미분 방정식의 값은 그 값을 얼마만큼 변경해야 하는가에 대해 너무 적은 정보를 제공한다. 따라서, 오차곡선의 곡률(curvature)과 같은 좀 더 고차원의 미분값에 대한 정보를 얻을 수 있다면 보다 빨리 학습시킬 수 있을 것이다. 이와 같은 아이디어에서 각 연결강도에 대해 전체 오차의 이차 미분값을 구하기 위하여 Newton 방법과 같이 수치해석적으로 잘 알려진 최적화 기법을 사용하는 방법이 많이 연구되고 있다. 이들 방법의 핵심은 순수한 이차 미분값을 구하는 데는 너무 많은 계산량이 필요하기 때문에 이의 계산량을 줄이기 위하여 이차 미분의 근사값을 최적으로 구하고자 하는 것이다.

3.3 학습 전략적 방법

이제까지 소개한 방법들은 학습 알고리즘 자체의 개선을 통해서 빠르게 학습시키고자 하는 것이었는데, 신경망의 학습속도에 영향을 미치는 요인은 알고리즘 외적인 것도 많이 있음을 볼 수 있다. 그 대표적인 예로 학습시켜야 할 패턴의 특성이 있다. 즉, 학습해야 할 패턴이 학습하기 매우 어려운 것과 비교적 쉬운 것이 섞여 있을 경우에 전체적으로 학습속도가 느리게 된다는 학습패턴들이 서로 유사하여 학습시키기 어려운 경우가 빈번히 발생한다. 본 절에서 소개하는 방법들은 이와 같은 문제를 해결하여 학습속도를 개선 하고자 하는 학습전략에 근거한 방법이다.

IV. 비교 및 토의

4.1 속도개선 방법의 비교

부류	장점	단점	대표적 방법
경험적 방법	· 알고리즘이 간단	· 이론적 뒷받침 부족	· Delta-bar-delta · Quickprop
수치해석적 방법	· 수학적 전개가능 · Quadratic 수렴속도	· 많은 계산량 필요	· 2nd order method
학습전략적 방법	· 아이디어가 간단	· 알고리즘화하기 어려움	· review method · preparation method

<표 4.1> 속도개선 방법의 장단점

본 절에서는 III장에서 분석한 기존의 속도개선 방법들의 특성과 복잡도를 비교하고자 한다. 이 연구에서는 완벽한 비교 보다는 그 필요성을 중심으로 소개한다. 먼저 <표 4.1>에 3가지 부류의 방법에 대한 간략한 장단점을 정리하였다. 이 표에 의하면 각 부류의 방법이 상호 보완적으로 사용되면 보다 좋은 효과를 얻을 수 있음을 알 수 있다. 즉, 이론적으로 잘 정립되어 있는 수치해석적 방법을 기반으로 몇가지 경험적 방법을 적용하고, 학습 전략적 방법으로 학습 스케줄링하는 방식을 고안할 수 있을 것이다.

<표 4.2>는 속도 개선방법의 계산량과 기억장소를 구하기 위한 시도의 일환으로 경험적 방법의 대표적인 방법인 Jacobs의 Delta-bar-delta [8]와 Fahlman의 Quickprop [5]을 구현하는데 필요한 추가의 계산량과 기억장소를 나타내고 있다. 이것에 의하면 알고리즘의 성능이 단순한 반복횟수에만 의존하는 것은 아님을 알 수 있다. 또, <표 4.3>은 이들을 포함한 몇가지 방법을 사용하여 XOR 문제를 풀었을 경우의 결과가 있다. 이 두가지 표를 종합하여 보면 Fahlman의 방법이 한번 반복하는데 소모되는 계산량이 적을 뿐만 아니라, 학습될 때까지 소요된 총 반복 횟수도 가장 적은 것을 볼 수 있다.

방법	추가계산량	추가기억장소
Delta-bar-delta (Jacobs)	$n(4M + a)$	$L + 2$
Quickprop (Fahlman)	$n(M + a)$	2

n : node 수
M : 곱셈
a : 덧셈
L : link 수

<표 4.2> 속도개선 방법의 계산량과 기억장소

횟수	방법	Backpropagation	0.1 added BP	Delta-bar-delta	Quickprop
평균		333.57	271.88	250.4	24.22
분산		148.45	109.34	59.5	16.3

<표 4.3> XOR 문제풀기 위한 반복횟수

4.2 토의 사항

신경망이 실제 문제에서 매우 유용할 수 있다는 가능성이 밝혀진 이후로 그 학습속도를 좀 더 현실적으로 빠르게 하고자 하는 연구가 계속되고 있는데, 이제까지 제안된 방법만도 수십 가지가 넘을 뿐만 아니라 아직까지도 표준화된 방법은 발표되지 않고 있는 실정에서 여러 연구가들에 의해 새로운 방법이 계속 제안될 전망이다. 그러나, 이러한 연구를 통해 서로 자신의 문제 영역에서 유용한 새로운 방법을 제안하고는 있지만, 그 방법이 실제로 유용한 것인지를 결정할 수 있는 기준(criteria)은 아직까지 제시되지 않고 있다. 따라서, 이 절에서는 앞 장에서 정리한 이제까지의 연구를 토대로 학습속도를 빠르게 하고자 하는 새로운 방법을 고안하고자 할 때 고려하여야 할 사항들을 제시하고 논의해 보고자 한다.

첫번째 논의는 제안된 방법의 복잡도에 관한 것이다. 속도 개선을 위한 알고리즘의 우수성을 가능하는 척도로는 신경망이 주어진 문제를 해결할 수 있도록 학습시키는데 소요된 반복횟수(epoch)를 사용하고 있다. 하지만, 이 이외에도 학습 알고리즘의 복잡도를 측정하기 위해서는 한번 반복하는데 소요되는 계산량을 측정해야 할 것이다. 따라서, 학습속도 개선 알고리즘의 성능을 가능하는데 반복 횟수 뿐만 아니라 사용된 계산량도 고려되어야 한다.

두번째는 성능측정을 위해 사용한 실험의 공정성에 대한 것이다 기존의 많은 방법들이 자신의 성능을 비교하기 위하여 나름의 문제를 선정하여 실험하고 있는데, 보다 공정한 결과물 얻기 위해서는 공정적으로 인정받을 수 있는 문제를 선정하여야 할 것이다 일반적으로 널리 사용되고 있는 실험문제로는 XOR 문제가 있는데, 이것은 제안된 방법의 분별력(discrimination power)을 측정하는데 사용될 수 있다 또, 일반화 정도(generalization capability)를 측정하는데는 encoder/decoder 문제 [5]가 적합할 것이다

세번째는 제안된 방법의 이론적 뒷받침에 관한 것인데, 이 제까지의 많은 연구들은 제안한 방법이 학습속도를 빠르게 할 수 있었던 구체적인 이론 보다는 단순히 제시한 후에 실험을 통해 검증하는 방법을 사용하여 왔다. 하지만, 좀 더 정형화된 학습 알고리즘의 연구를 위해서는 제안한 방법의 이론적 뒷받침이 필요하다

마지막으로 학습 알고리즘의 개선에 관한 연구에서 고려하여야 할 것은 제안된 방법이 단순히 속도만을 빠르게 하는 것인가에 대한 것이다 일반적으로 기존의 수치해석 연구에 의하면, 반복법을 사용하여 해를 찾아내는 경우에 속도를 빠르게 하기 위하여 값을 너무 크게 변화시키면 안정된 답을 얻을 수 없다는 것이 알려져 있다 즉, 학습속도의 개선에 아울러 일반화 정도도 고려하여 학습 알고리즘을 개발하여야 할 것이다

V 결 론

Backpropagation 알고리즘이 그 자체의 단순성에도 불구하고 여러 응용 분야에서 좋은 성능을 보이는데 고무되어 이의 학습속도를 현실화 하고자 하는 연구가 많이 진행되고 있다 본 연구에서는 기존의 Backpropagation 학습속도를 개선하고자 하는 방법들을 경험적 방법, 수치해석적 방법 및 학습 전략적 방법으로 나누어 정리하고, 그들을 시간 및 기억장소의 복잡도를 통해 비교해 보고자 하였다 또, 이러한 정리를 통해 얻을 수 있는 새로운 학습속도 개선방법을 제안하고자 할때 고려하여야 할 사항에 대해 논의하여 보았다

이러한 연구는 기존의 속도개선 방법들을 기반으로 새로운 알고리즘을 개발하는데 일종의 비교기준을 설정할 수 있도록 할 수 있을 것이며, 이를 좀 더 발전시켜 하드웨어적으로 속도를 향상시키고자 하는 방법인 병렬처리 기법에 적합한 알고리즘의 가능성을 모색할 수도 있을 것이다

참고 문헌

- [1] Angus, J E, "On the Connection between Neural Network Learning and Multivariate Nonlinear Least Squares Estimation," *Journal of Neural Networks*, Vol 1, pp 42-47, 1989
- [2] Becker, S, and le Cun, Y, "Improving the Convergence of Back-Propagation Learning with Second Order Methods," In *Proceedings of the 1988 Connectionist Models Summer School*, pp 29-37, 1989
- [3] Chen, J R, and Mars, P, "Stepsize Variation Methods for Accelerating the Back-Propagation Algorithm," In *Proceedings of the IEEE International Joint Conference on Neural Networks*, Vol 1, pp 601-604, 1990

- [4] Dewan, H M, and Sontag, E D, "Extrapolatory Methods for Speeding Up the BP Algorithm," In *Proceedings of the IEEE International Joint Conference on Neural Networks*, Vol 1, pp 613-616, 1990
- [5] Fahlman, S E, "An Empirical Study of Learning Speed in Back-Propagation Networks," *Technical Report CMU-CS-88-162*, Carnegie-Mellon University, 1988
- [6] Hagiwara, M, "Accelerated Back Propagation Using Unlearning Based on Hebb Rule," In *Proceedings of the IEEE International Joint Conference on Neural Networks*, Vol 1, pp 617-620, 1990
- [7] Hush, D R, and Salas, J M, "Improving the Learning Rate of Back-Propagation with the Gradient Reuse Algorithm," In *Proceedings of the IEEE International Conference on Neural Networks*, Vol 1, pp 441-447, 1988
- [8] Jacobs, R A, "Increased Rates of Convergence Through Learning Rate Adaptation," *Journal of Neural Networks* Vol 1, pp 295-307, 1988
- [9] Kollias, S, and Anastassiou, D, "Adaptive Training of Multilayer Neural Networks using a Least Squares Estimation Technique," In *Proceedings of the IEEE International Conference on Neural Networks*, Vol 1 pp 383-390, 1988
- [10] le Cun, Y, "A Theoretical Framework for Back-Propagation," In *Proceedings of the 1988 Connectionist Models Summer School*, pp 21-28 1988
- [11] Mon, Y, and Yokosawa, K, "Neural Networks that Learn to Discriminate Similar Kanji Characters," *Advances in Neural Information Processing Systems 1*, pp 332-347, 1989
- [12] Ohnishi, N, Okamoto, A, and Sugie, N, "Selective Presentation of Learning Samples for Efficient Learning in Multi-Layer Perceptron," In *Proceedings of the IEEE International Joint Conference on Neural Networks* Vol 1, pp 688-691, 1990
- [13] Orfanidis, S J, "Fast Neural Nets with Gram-Schmidt Orthogonalization," In *Proceedings of the IEEE International Joint Conference on Neural Networks*, Vol 1, pp 692-695, 1990
- [14] Parker, D B, "A Comparison of Algorithms for Neuron-like Cell," *Neural Networks for Computing*, pp 327-332, 1986
- [15] Shepanski, J F, "Fast Learning in Artificial Neural Systems Multilayer Perceptron Training using Optimal Estimation," In *Proceedings of the IEEE International Conference on Neural Networks*, Vol 1, pp 465-472, 1988
- [16] Sung-Bae Cho, and Jin H Kim, "An Accelerated Learning Method with Backpropagation," In *Proceedings of the IEEE International Joint Conference on Neural Networks*, Vol 1, pp 605-608, 1990