

Spiking Neural Networks 의 구간연산 학습알고리즘

이경희*

*평택대학교 정보통신학과

khlee@ptu.ac.kr

Interval Arithmetic Learning Algorithm for Spiking Neural Networks

Kyunghee Lee*

*Dept. of Info. & Comm., PyeongTaek University

요 약

본 논문에서는 스파이킹 뉴런(Spiking Neuron)들이 쿨롱에너지 포텐셜 (Coulomb Energy Potential) 을 가지는 스파이킹 신경회로망에서의 학습알고리즘을 일반화하여 구간연산(Interval Arithmetic)의 학습이 가능한 학습알고리즘을 제안한다. 제안하는 학습알고리즘은 입력 데이터로서 구간(Interval) 데이터와 포인트(Point) 데이터를 모두 학습 할 수 있는 일반화된 학습알고리즘으로서 간단한 컴퓨터 시뮬레이션을 통하여 범위(Lower bound & Upper bound)를 가지는 구간데이터와 포인트데이터의 통합적인 학습이 가능하고 전문가시스템(expert system)에서의 “don’t care attributes”의 학습 등에도 활용이 가능함을 보인다.

1. 서론

스파이킹 신경회로망 모델은 기존의 전통적인 신경회로망 모델과는 다르게 뉴런(Neuron)에서 발생하는 스파이크(Spike)에 대한 정확한 시간정보를 중심으로 반응하는 모델로서 음성인식, 동작인식 등 시간적 정보에 대한 실시간 처리를 필요로 하는 분야에서 좋은 결과를 내고 있다[1,2]. 대부분의 스파이킹 신경회로망에서는 학습데이터로 포인트(Point) 데이터를 사용하고 있다[3-4]. 본 논문에서는 구간연산이 가능한 스파이킹 모델로 확장하고 구간데이터와 포인트데이터를 모두 학습할 수 있는 구간연산 학습 알고리즘으로의 확장을 제안하고 간단한 컴퓨터 시뮬레이션을 통하여 구간데이터와 포인트데이터가 동일하게 학습될 수 있음을 보인다.

2. 스파이킹 신경회로망과 구간연산 학습

2.1 구간연산

신경회로망에서의 학습알고리즘을 개발하기 위한 기본적인 구간연산은 다음과 같다[5].

구간들에 대한 합:

$$A + B = [a^l, a^u] + [b^l, b^u] = [a^l + b^l, a^u + b^u] \quad (2-1)$$

구간에 대한 실수의 곱:

$$m \cdot A = m \cdot [a^l, a^u] = \begin{cases} [m \cdot a^l, m \cdot a^u] & \text{if } m \geq 0 \\ [m \cdot a^u, m \cdot a^l] & \text{if } m < 0 \end{cases} \quad (2-2)$$

구간에 대한 지수함수:

$$\exp A = \exp[a^l, a^u] = [\exp a^l, \exp a^u] \quad (2-3)$$

여기서 1 은 데이터의 최하치, u 는 데이터의 최대치를 각각 나타낸다.

2.2 구간연산 학습

스파이킹 신경회로망은 스파이크 발화 시점들의 타이밍 정보가 중요한 신경코드 역할을 하며 정교한 발화 타이밍 정보에 의한 동작이 이루어지는 모델이다[3,4]. 또한, 시간 t 에서 다중 스파이크에 의한 스파이킹 뉴런 j 의 post-synaptic, $x_j(t)$, 과 스파이킹 뉴런 j 의 활성화 함수는 다음과 같이 정의된다[5,6].

$$x_j(t) = \sum_{i \in I_j} \sum_{k=1}^m w_{ij}^k y_i^k(t) \quad (2-4)$$

$$y_j^k(t) = \varepsilon(t - t_j - d^k) \quad (2-5)$$

$$\varepsilon(t) = 1/\tau \exp(1 - t/\tau) \quad (2-6)$$

$$y_j(\text{Net}) = y_j([\text{net}^l, \text{net}^u]) = [y_j(\text{net}^l), y_j(\text{net}^u)]$$

$$\text{where } y_j^k(t) = \varepsilon(t - t_j - d^k) \quad (2-7)$$

M 개의 메모리를 갖는 스파이킹 신경회로망의 경우 쿨롱포텐셜(Coulomb Potential)에 의한 에너지는 다음과 같이 정의할 수 있다[7].

$$\psi = \frac{1}{(2L)} \sum_{a,b} \sum_{j \in J} \text{sign}(a, b) \|t_j^a - t_j^b\|^{-L} \quad (2-8)$$

구간연산 스파이킹 네트워크에서의 학습은 에너지 함수를 최소화하기 위한 가중치의 조정은 다음과 같다.

$$\Delta w_{ij}^k = -\eta \frac{\partial \psi}{\partial w_{ij}^k} \quad (2-9)$$

따라서 스파이킹 네트워크의 학습은 다음과 같다.

$$\Delta w_{ij}^k = \frac{1}{2} \eta \cdot \left\{ y_i^k(t_j^{a^l}) \delta_j^{a^l} - y_i^k(t_j^{b^l}) \delta_j^{b^l} + y_i^k(t_j^{a^u}) \delta_j^{a^u} - y_i^k(t_j^{b^u}) \delta_j^{b^u} \right\} \quad (2-10)$$

where $w_{ij} \geq 0$

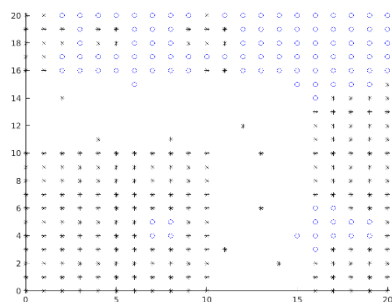
3. 컴퓨터 시뮬레이션

스파이킹 신경회로망에서 제안한 구간연산 학습알고리즘의 성능을 확인하기 위한 간단한 컴퓨터 시뮬레이션 결과를 보인다. 첫번째 실험은 2 개 영역에 위치한 데이터 학습실험으로 1 계층의 스파이킹신경망을 사용하였다. 표 1 에 실험에 사용한 구간 데이터와 포인트 데이터를 나타낸다. 첫번째 실험에 사용한 주요 파라미터로는 학습율=0.0000002, L=2, M=600, 가중치의 초기값=0.0~1.0, 학습횟수 (epoch)=8 의 값을 선택하여 진행하였다. 그림 1 에서는 학습의 진행과정을 보인다. 학습이 진행되면서 class1 과 class2 의 영역 데이터에 대한 학습이 잘 이루어지는 것을 볼 수 있다. 특히, 그림 1(d)은 학습 완료 시 2 개 영역의 구분이 점선과 같이 나누어짐을 표시한 결과이다.

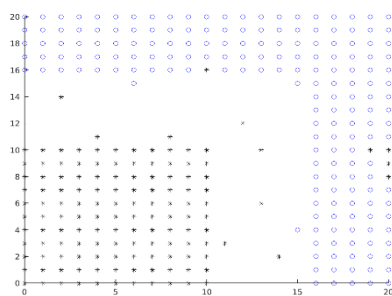
<표 1> 2 개 영역의 학습을 위한 구간 및 포인트 데이터

	학습데이터 (구간데이터 및 포인트 데이터)-2개영역
C.1 X	[0.00 10.00] [10.00 10.00] [0.00 0.00] [5.00 5.00] [1.00 1.00] [8.00 8.00] [9.00 9.00] [10.00 10.00]
C.1 Y	[0.00 10.00] [4.00 4.00] [1.00 1.00] [5.00 5.00] [9.00 9.00] [0.00 0.00] [9.00 9.00] [6.00 6.00]
C.2 X	[0.00 20.00] [16.00 20.00] [1.00 1.00] [6.00 6.00] [12.00 12.00] [17.00 17.00] [15.00 15.00] [16.00 16.00]
C.2 Y	[16.00 20.00] [0.00 20.00] [17.00 17.00] [15.00 15.00] [17.00 17.00] [1.00 1.00] [4.00 4.00] [6.00 6.00]

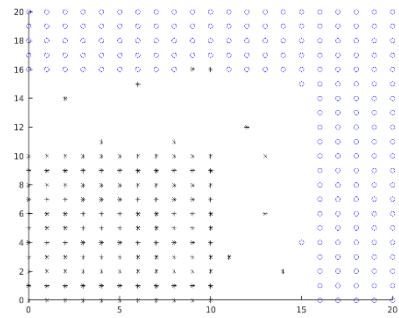
두번째 실험은 5 개의 영역에 위치한 데이터의 학습실험으로 표 2 에 구간데이터 및 포인트데이터를 나타낸다. 두번째 학습에 사용한 주요 파라미터로는 학습율=0.0000005, 학습횟수(epoch)=12, 기타의 파라미터는 첫번째 실험에서와 동일하다. 첫번째 실험에서와 같이 학습이 진행되면서 중심영역 데이터인 class2 와 4 개의 모퉁이 영역 데이터인 class1 에 대한 구간학습이 제대로 이루어진 것을 알 수 있다(그림 2).



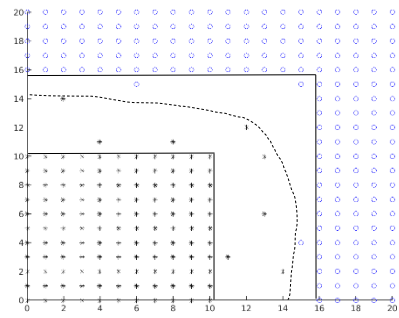
(a)



(b)



(c)

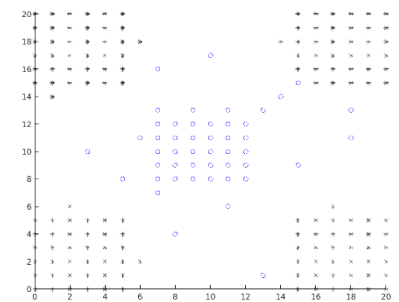


(d)

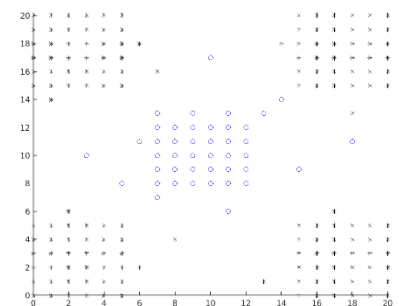
(그림 1) 2 개 영역 데이터의 학습 진행과정: ((a) epoch=2, (b) epoch=4, (c) epoch=6, (d) epoch=8))

<표 2> 5 개 영역 학습을 위한 구간 및 포인트 데이터

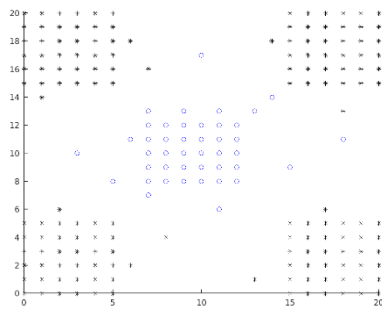
	학습데이터 (구간데이터 및 포인트 데이터)-5개영역
C.1 X	[0.0 5.0] [4.0 4.0] [5.0 5.0] [15.0 20.0] [16.0 16.0] [17.0 17.0] [15.0 20.0] [15.0 15.0] [16.0 16.0] [0.0 5.0] [3.0 3.0] [4.0 4.0]
C.1 Y	[0.0 5.0] [1.0 1.0] [4.0 4.0] [15.0 20.0] [16.0 16.0] [15.0 15.0] [0.0 5.0] [1.0 1.0] [4.0 4.0] [15.0 20.0] [14.0 14.0] [15.0 15.]
C.2 X	[8.0 12.0] [13.0 13.0] [8.0 8.0] [11.0 11.0] [12.0 12.0] [10.0 10.0] [9.0 9.0]
C.2 Y	[8.0 12.0] [8.0 8.0] [8.0 8.0] [8.0 8.0] [11.0 11.0] [10.0 10.0] [12.0 12.0]



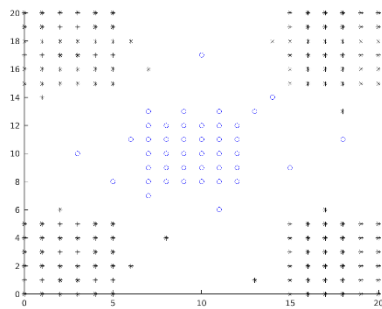
(a)



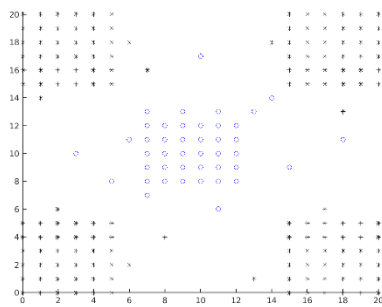
(b)



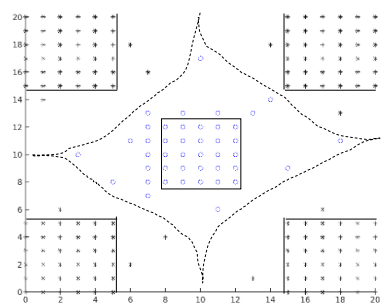
(c)



(d)



(e)



(f)

(그림 2) 5 개 영역 데이터의 학습 진행과정: ((a) epoch=2, (b) epoch=4, (c) epoch=6, (d) epoch=8, (e) epoch=10, (f) epoch=12)

그림 2(f)은 학습 완료 시 5 개 영역의 구분이 점선과 같이 나누어짐을 표시한 결과이다.

4. 결론

본 논문에서는 구간데이터와 포인트데이터를 모두 입력 데이터로 학습할 수 있는 쿨롱에너지 기반 스파

이킹 신경회로망의 학습알고리즘을 제안하였다.

따라서, 전통적인 인공지능시스템에서 특정한 범위(lower bound & upper bound)의 값을 가지는 지식 표현 데이터(구간데이터)가 포인트데이터와 동일한 입력데이터로서 스파이킹 신경회로망에서 자연스럽게 학습이 가능한 알고리즘이다. 또한 영역내의 많은 개수의 포인트 데이터를 작은 개수의 구간데이터로 변환 입력이 가능함으로써 학습의 시간을 줄일 수 있고 다양한 형태의 특정 영역학습에 쉽게 적용이 가능할 것이다. 그러나 적용되는 문제의 유형에 따라 적절하게 선택이 되어야 하는 시간 관련 파라미터 등은 학습 메커니즘에 중요한 연구분야이며 추가적인 연구가 필요하다.

참고문헌

- [1] H.Fang, Y.Wang, J.He, and S.Liu, "Temporal pattern recognition using spiking neural networks for cortical neuronal spike train decoding", Proceedings of the 17th World Congress The international Federation of Automatic Control, 5203-5208, 2008.
- [2] A.gupta, Lyte N.Long, "Character Recognition using Spiking Neural Networks", 2007 International Joint Conference on Neural Networks (IJCNN 2007), 53-58, 2007.
- [3] Booi, O., and Nguyen, H., "A gradient descent rule for spiking neurons emitting multiple spikes," Information Processing Letters, Vol. 95, No. 6, 552-558, 2005.
- [4] S.Shrestha and Q.Song, "Adaptive Delay Learning in SpikeProp Based on Delay Convergence Analysis," 2016 International Joint Conference on Neural Networks (IJCNN 2016), 277-284, 2016.
- [5] G.Alefeld and J.Herzberger, Introduction to Interval Computations. Academic Press. NewYork.
- [6] S.Bohte, "The Evidence for Neural Information Processing With Precise Spike-Time: A Survey," Natural Computing, vol. 3, 2, 195-206, 2004.
- [7] S.Shrestha and Q.Song, "Weight Convergence of SpikeProp and Adaptive Learning Rate," in Proceedings of 51st Annual Allerton Conference on Communication Control and Computing, 506-511, 2013.
- [8] C.Scofield, "Learning internal representations in the coulomb energy network", Proceedings of the IEEE International Conference on Neural Networks, Vol.1, 271-276, 1988.