특집논문 (Special Paper)
방송공학회논문지 제25권 제2호, 2020년 3월 (JBE Vol. 25, No. 2, March 2020)
https://doi.org/10.5909/JBE.2020.25.2.157
ISSN 2287-9137 (Online) ISSN 1226-7953 (Print)

임베디드 시스템에서의 양자화 기계학습을 위한 효율적인 양자화 오차보상에 관한 연구

석 진 욱[a)‡]

Study on the Effective Compensation of Quantization Error for Machine Learning in an Embedded System

Jinwuk Seok[a)‡]

요 약

본 논문에서는 임베디드 시스템에서의 양자화 기계학습을 수행할 경우 발생하는 양자화 오차를 효과적으로 보상하기 위한 방법론을 제안한다. 경사 도함수(Gradient)를 사용하는 기계학습이나 비선형 신호처리 알고리즘에서 양자화 오차는 경사 도함수의 조기 소산(Early Vanishing Gradient)을 야기하여 전체적인 알고리즘의 성능 하락을 가져온다. 이를 보상하기 위하여 경사 도함수의 최대 성분에 대하여 직교하는 방향의 보상 탐색 벡터를 유도하여 양자화 오차로 인한 성능 하락을 보상하도록 한다. 또한, 기존의 고정 학습률 대신, 내부 순환(Inner Loop) 없는 비선형 최적화 알고리즘에 기반한 적응형 학습률 결정 알고리즘을 제안한다. 실험 결과 제안한 방식의 알고리즘을 로젠블록 함수를 통한 비선형 최적화 문제에 적용할 시 양자화 오차로 인한 성능 하락을 최소화시킬 수 있음을 확인하였다.

Abstract

In this paper. we propose an effective compensation scheme to the quantization error arisen from quantized learning in a machine learning on an embedded system. In the machine learning based on a gradient descent or nonlinear signal processing, the quantization error generates early vanishing of a gradient and occurs the degradation of learning performance. To compensate such quantization error, we derive an orthogonal compensation vector with respect to a maximum component of the gradient vector. Moreover, instead of the conventional constant learning rate, we propose the adaptive learning rate algorithm without any inner loop to select the step size, based on a nonlinear optimization technique. The simulation results show that the optimization solver based on the proposed quantized method represents sufficient learning performance.

Keyword : Machine Learning, Quantized Learning, Quantization, Learning Equation, Embedded System

Ⅰ. 서 론

기존의 기계학습 혹은 비선형 신호 처리의 경우, 부동 소수점 연산을 기반으로 연산을 수행한다. 그러나, 기계학습과 같이 다수의 연산 모듈을 사용하여 실시간성을 추구하는 대상의 경우 부동 소수점 계산을 위한 연산 모듈의 크기와 복잡도가 정수 연산보다 상대적으로 크기 때문에 소형, 경량의 하드웨어를 필요로 하는 분야에서는 적합하지 않다[1][2].

따라서, 비트 감소, 계산 속도 개선, 가용성 확장과 같은 엔지니어링 측면의 장점을 제공해 줄 수 있는 처리 데이터의 양자화는 다양한 엔지니어링 분야에서 연구되었다. 통상적인 경우, 양자화 된 도메인의 학습 방정식은 파라미터의 최하위 비트의 업데이트로 구현되므로, 업데이트 파라미터에 일정한 학습 속도를 적용하는 것과 동일하다. 그러나 학습률이 일정한 일반적인 확률론적 최급 강하 알고리즘은 1차 모멘트 수렴 혹은 분포 수렴과 같이 매우 약한 토폴로지에서 최적의 점으로 수렴하기에 성능 하락을 피할 수 없다[3][4][5].

본 논문에서는 이와 같은 양자화의 이점을 기계학습에 이식하기 위해, 양자화로 인한 성능하락을 극복하기 위한 적응적 학습률의 기계학습 적용 알고리즘을 제안하고, 이를 기반으로 양자화 오차 보상을 위한 보상 알고리즘과 이를 강화학습 등의 기계학습에 도입 가능한 비선형 최적화 문제에 적용하여 제안한 양자화 오차 보상 방법론이 타당함을 보인다.

a) 한국전자통신연구원(Electronics and Telecommunications Research Institute)
‡ Corresponding Author : 석진욱(Jinwuk Seok)
E-mail: jnwseok@etri.re.kr
Tel: +82-42-860-6365
ORCID: https://orcid.org/0000-0001-5318-1237
※ 이 논문의 연구 결과중 일부는 한국방송미디어공학회 “2019년 추계학술대회”에서 발표한 바 있음.
※ 본 논문은 2020년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임. (No.2017-0-00142, 스마트기기를 위한 온디바이스 지능형 정보처리 가속화 SW플랫폼 기술 개발)
※ This work was supported by Institute for Information and communications Technology Promotion(IITP) grant funded by the Korea government (MSIP). (2017-0-00072, Development of Audio/Video Coding and Light Field Media Fundamental Technologies for Ultra Realistic Tera-media)
· Manuscript received December 23, 2019; Revised March 3, 2020; Accepted March 6, 2020.

본 논문의 구성은 다음과 같다. 2 절에서는 양자화 및 주요 양자화 연산에 대한 정의를 두고, 3 절에서는 양자화 기반 최급 강하법의 정의와 적응적 학습률의 기계학습 정의를 위한 알고리즘을 제안하고 4절에서는 양자화 기계학습 시 발생하는 양자화 오차 보상 알고리즘을 설명한다. 5절에서는 제안한 기법의 성능을 실험을 통해서 확인하며. 마지막으로 6절에서는 본 논문에 대한 결론을 맺는다.

Ⅱ. 양자화 및 주요 양자화 연산 정의

1. 양자화 연산의 정의와 특성

먼저 변수 $x \in \mathrm{R}$ 의 양자화의 정의를 위해 다음과 같이 정수화를 위한 아래 자리 수 버림을 (Round-off) 다음과 같이 정의한다[3].

$$x \equiv \lfloor x \rfloor + \epsilon \quad (\epsilon \in \mathbf{R}[0,1)) \tag{1}$$

식 (1)에서 기호 $\lfloor x \rfloor \in \mathbf{Z}$ 는 자리 버림 연산을 나타내는 것으로서 x 보다 작은 정수 값 중 가장 가까운 값을 가지는 것으로 정의한다. 이를 사용하여 다음과 같이 가우스 기호 (Gauss symbol)를 정의한다[3].

$$[x] \equiv \lfloor x + 0.5 \rfloor = x + 0.5 - \epsilon \triangleq x + \varepsilon \tag{2}$$

이를 사용하여 x의 반올림 연산을 $[x]$으로 정의하면, 자리 오차 (round-off error) ϵ는 $\epsilon \in \mathbf{R}(-0.5, 0.5)$로 정의된다. 또한, 임의의 실수 열 x_k, $\forall k \in \mathbf{N}$, $x_k \in \mathbf{R}$에 대하여 각 x_k에 대한 자리 버림을 ϵ_k로 나타낼 수 있다고 하면

$$\left\lfloor \sum_{k=1}^{n} x_k \right\rfloor = \left\lfloor \sum_{k=1}^{n} (\lfloor x_k \rfloor + \epsilon_k) \right\rfloor = \sum_{k=1}^{n} \lfloor x_k \rfloor + \left\lfloor \sum_{k=1}^{n} \epsilon_k \right\rfloor \tag{3}$$

이에, 식 (1),(2)의 정의를 사용, 다음과 같이 양자화 연산을 정의한다.

$$x^{Q} \triangleq \frac{1}{Q_p} \left\lfloor Q_p \cdot \left(x + 0.5 \cdot Q_p^{-1} \right) \right\rfloor \tag{4}$$

식 (4)에서 Q_p는 양자화 계수로서 양자화의 수준을 결정한다. 예를 들어, 10^{-3} 수준의 고정 소수점으로 양자화를 수행하려고 한다면 $Q_p = 10^3$로 결정된다. 편의상 양자화 계수는 양의 정수로 놓는다 ($Q_p \in \mathbf{Z},\ Q_p > 0$). 특정 정수의 가우스 기호 없이 양자화를 표현하기 위하여 자리 오차를 도입하여 식 (4)를 다시 쓰면 다음과 같다.

$$x^Q = \frac{1}{Q_p}\lfloor Q_p \cdot x \rfloor = \frac{1}{Qp}\lfloor Qp \cdot (x + 0.5Q_p^{-1}) \rfloor = \frac{1}{Qp}(Qp \cdot x + \varepsilon) \\ = x + \varepsilon Q_p^{-1} \tag{5}$$

식 (5)에서 $x^Q \in \mathbf{R}$이지만, $Q_p x^Q = \lfloor Q_p \cdot\ x \rceil \in \mathbf{Z}$이다. 따라서 x^Q는 단순히 양자화가 정수화를 의미하는 것이 아닌 고정 소수점의 형태까지 의미하며 자리오차 ϵ의 범위는 $\epsilon \in \mathbf{R}(-0.5Q_p^{-1}, 0.5Q_p^{-1}] = \mathbf{R}(-5\cdot\ (10Q_p)^{-1}, 5\cdot\ (10Q_p)^{-1}]$ 이 된다.

양자화 연산의 4칙 연산을 고찰해 보면, 먼저, 식 (3), (4)와 같은 특성을 가지지만, 나눗셈의 경우는 양자화된 값으로 나타나지 않을 수 있으므로 이를 다음의 형태로 생각한다. 정수 $x, a \in \mathbf{Z}$에 대하여 몫과 나머지를 생각하면 x, a의 나눗셈은 다음과 같이 표현할 수 있다.

$$\frac{x}{a} = \left\lfloor \frac{x}{a} \right\rfloor + \frac{1}{a}\left(x - a\left\lfloor \frac{x}{a} \right\rfloor\right) \tag{6}$$

식 (6)에서 몫은 $\left\lfloor \frac{x}{a} \right\rfloor$, 나머지는 $x - a\left\lfloor \frac{x}{a} \right\rfloor$가 된다. 식 (6)에서 가우시안 연산을 적용, 몫과 나머지를 살펴보면

$$\left\lfloor \frac{x}{a} \right\rfloor = \left\lfloor \left\lfloor \frac{x}{a} \right\rfloor + \frac{1}{a}\left(x - a\left\lfloor \frac{x}{a} \right\rfloor\right) \right\rfloor = \left\lfloor \frac{x}{a} \right\rfloor + \left\lfloor \frac{1}{a}\left(x - a\left\lfloor \frac{x}{a} \right\rfloor\right) \right\rfloor \tag{7}$$

위에서 $\left(x - a\left\lfloor \frac{x}{a} \right\rfloor\right) < a$ 임을 알 수 있다. 반면, 식 (6)에 대하여 반올림 연산을 적용하면

$$\left\lceil \frac{x}{a} \right\rfloor = \left\lfloor \frac{x}{a} + 0.5 \right\rfloor = \left\lfloor \left\lfloor \frac{x}{a} \right\rfloor + \frac{1}{a}\left(x - a\left\lfloor \frac{x}{a} \right\rfloor\right) + \frac{a}{2a} \right\rfloor \\ = \left\lfloor \frac{x}{a} \right\rfloor + \left\lfloor \frac{1}{a}\left(x - a\left(\left\lfloor \frac{x}{a} \right\rfloor - \frac{1}{2}\right)\right) \right\rfloor \tag{8}$$

그러므로

$$\left\lceil \frac{x}{a} \right\rfloor = \begin{cases} \left\lfloor \frac{x}{a} \right\rfloor & x < a\left(\left\lfloor \frac{x}{a} \right\rfloor + \frac{1}{2}\right) \Rightarrow a \cdot \varepsilon > 0 \\ \left\lfloor \frac{x}{a} \right\rfloor + 1 & x \geq a\left(\left\lfloor \frac{x}{a} \right\rfloor + \frac{1}{2}\right) \Rightarrow a \cdot \varepsilon \leq 0 \end{cases} \tag{9}$$

식 (6)에서 나눗셈 연산에서 추가로 1 혹은 0이 더해져야 할 조건은 다음과 같다. 가우스 기호의 정의에 따라 1이 되기 위해서는 다음의 조건을 만족해야 한다.

$$x - a\left(\left\lfloor \frac{x}{a} \right\rfloor - 0.5\right) \geq a \Longrightarrow x \geq a \cdot \left\lfloor \frac{x}{a} \right\rfloor + 0.5a \\ \Longrightarrow x \geq a \cdot \left(\frac{x}{a} - \epsilon\right) + 0.5a \quad \because \text{by (1)}\ \ x = \lfloor x \rfloor + \epsilon \\ \Longrightarrow x \geq x - 0.5a + \varepsilon + 0.5a \because \text{by (2)} \varepsilon = 0.5 - \epsilon \\ \Longrightarrow a\varepsilon \leq 0 \tag{10}$$

특별히 $\left(x - a\left\lfloor \frac{x}{a} \right\rfloor\right) < a$ 이어서 $\left\lfloor \frac{x}{a} \right\rfloor < 1$ 의 경우에 이를 $Q_p \left\lfloor \frac{x}{a} \right\rfloor < Q_p$ 로 확장하는 경우의 연산은 다음으로 나타난다.

$$Q_p \cdot \frac{x}{a} = \left\lfloor \frac{Q_p x}{a} \right\rfloor + \frac{1}{a}\left(Q_p x - a\left\lfloor \frac{Q_p x}{a} \right\rfloor\right) \tag{11}$$

식 (9), (11)에 의하여 나눗셈의 양자화는 다음과 같이 유도된다. 먼저, 다음 값을 갖는 함수 $g(x,a) \in 0,1$ 을 정의하자.

$$g(x,a) = \left\lfloor \frac{1}{a}\left(Q_p x - a\left(\left\lfloor \frac{Q_p x}{a} \right\rfloor - \frac{1}{2}\right)\right) \right\rfloor = \begin{cases} 1 & a \cdot \varepsilon \leq 0 \\ 0 & a \cdot \varepsilon > 0 \end{cases} \tag{12}$$

식 (37)를 사용하여 나눗셈에 대한 양자화를 해석하게 되면 다음과 같다.

$$\frac{1}{Q_p}\left\lfloor Q_p \cdot \frac{x}{a} \right\rceil = \frac{1}{Q_p}\left\lfloor \left\lfloor \frac{Q_p x}{a} \right\rfloor + \frac{1}{a}\left(Q_p x - a\left(\left\lfloor \frac{Q_p x}{a} \right\rfloor - \frac{1}{2}\right)\right) \right\rfloor \\ = \frac{1}{Q_p}\left\lfloor \frac{Q_p x}{a} \right\rfloor + \frac{1}{Q_p}\left\lfloor \frac{1}{a}\left(Q_p x - a\left(\left\lfloor \frac{Q_p x}{a} \right\rfloor - \frac{1}{2}\right)\right) \right\rfloor \\ = \frac{1}{Q_p}\left(\left\lfloor \frac{Q_p x}{a} \right\rfloor + g(x,a)\right) \tag{13}$$

그러므로, 나머지 값의 분포에 따라 양자화 시, 가장 작은

값이 1 혹은 0을 가지게 된다.

2. 양자화된 학습방정식과 양자화 특성

기계학습의 학습 방정식에 양자화를 적용하게 되는 경우를 분석하자, 먼저, $x_t \in \mathbf{R^n}$에 대하여 다음과 같은 형태의 학습 방정식이 주어졌다고 가정한다.

$$x_{t+1} = x_t - \lambda_t h_t \tag{14}$$

식 (14)에서 $h_t \in \mathbf{R^n}$은 탐색방향 벡터이고 $\lambda \in \mathbf{R}(0,1]$은 학습률이며 $t \in \mathbf{R}$는 시간에 대한 파라미터이다. 식 (14)에서 x_t의 양자화를 $x_t^Q \equiv [x_t]$로 정의한다. 이때 우리의 목표는 학습방정식에서 도출된 수열 $\{x_t^Q\}_{t=0}^{\infty}$이 모두 양자화 되는 것이므로 각 t에 대해 양자화된 x_t^Q에서 x_{t+1}^Q를 얻도록 하면, 다음의 양자화된 학습 방정식을 얻게 된다.

$$x_{t+1}^Q = (x_t - (\lambda_t h_t))^Q \triangleq (x_t^Q - (\lambda_t h_t)^Q)^Q = x_t^Q - (\lambda_t h_t)^Q \tag{15}$$

학습률 $\lambda_t \in \mathbf{R}$와 탐색 방향 벡터 $h_t \in \mathbf{R}^n$가 각각 스칼라 및 벡터이므로, $\lambda_t h_t = (\lambda_t h_t)^Q$로 만드는 양자화는 구현이 어렵다. 그러므로, 최적의 학습률을 선 탐색(Line Search) 알고리즘을 통해 구하더라도, 이를 양자화가 가능하도록 다시 계산해 주어야 한다. 만일 단위 양자화 탐색 방향 벡터가 유리수로 잘 정의되었다면(i.e. $\frac{1}{Q_p} h_t \in \mathbf{Q}^n$), t번째 전체 데이터집합 반복 수행 (Epoch)에서 내부 반복 지표 (Internal Epoch index) k_t에 대하여 출력하는 함수 $z(k_t) \in \mathbf{Z}$, $z(k;t) > 0$를 놓고 이를 사용하여 학습률 λ_t를 $\lambda_t = z(k_t) Q_p^{-1}$로 놓으면 양자화된 갱신항은 다음과 같다.

$$(\lambda_t h_t)^Q = z(k_t) h_t^Q, \qquad h_t^Q \triangleq \left(\frac{1}{Q_p} h_t\right)^Q, \ \ h_t^Q \in \mathbf{Q}^n \tag{16}$$

이때, t 번째 반복(Iteration)에서 내부 반복 지표(Internal iteration index) k_t에 대하여 출력하는 함수 $z(k_t) \in \mathbf{Z}$, $z(k;t) > 0$를 놓고 이를 사용하여 학습률 λ_t를 $\lambda_t = z(k_t) Q_p^{-1}$로 놓으면 갱신항은 다음과 같다

$$(\lambda_t h_t)^Q = \frac{z(k_t)}{Q_p} h_t + \varepsilon = \left(\frac{z(k_t)}{Q_p} h_t\right)^Q \tag{17}$$

식 (37)을 잘 정의하여 $\frac{1}{Q_p} h_t \in \mathbf{Q}^n$가 되도록 양자화를 잘 정의하였다면 (혹은, 연산 기 내부 특성을 통해 이렇게 정의될 수 있다고 하면) $z(k_t) \in \mathbf{Z}$이므로

$$\left(\frac{z(k_t)}{Q_p} h_t\right)^Q = z(k_t) \left(\frac{1}{Q_p} h_t\right)^Q \tag{18}$$

식(37) 에서, $z(k_t) \in \mathbf{Z}(0, Q_p]$가 되며, 잘 정의된 $\frac{1}{Q_p} h_t \in \mathbf{Z}$을 기본 양자화 탐색 방향 벡터라고 하고 다음과 같이 정의한다.

$$h_t^Q \triangleq \left(\frac{1}{Q_p} h_t\right)^Q, \ \ h_t^Q \in \mathbf{Z} \tag{19}$$

식 (19)를 사용하여 식 (17)을 다시 정의하면

$$(\lambda_t h_t)^Q = z(k_t) h_t^Q \tag{20}$$

식 (20)에서 $z(k_t) \in \mathbf{Z}(0, Q_p]$이고, $h_t^Q \in Q^n$이므로 유리수 체의 정의에 의하여 갱신항은 양자화 계수에 의해 해상도가 정의되는 유리수가 된다. 따라서, $x_t \in \mathbb{Q}^n \subset \mathbb{R}^n$인 경우, 학습 방정식에서 생성된 수열 은 유리수 집합의 부분 집합이므로, 모든 학습반복 (Iteration)에 대하여 양자화된 값을 가지는 학습 방정식의 유도가 가능하다.

Ⅲ. 양자화 기반 최급 강하법과 적응적 학습률의 기계학습 적용 방법

식 (16)-(20)을 통해 양자화된 갱신항을 정의할 수 있게 되면, 식 (15)에 의하여 양자화된 변수는 모든 $t \in \mathbb{Z}^+$에 대하여, $x_t \in \mathbb{Q}^n$을 유지하게 되어 양자화 기반 최급 강하법을 기계학습에 적용할 수 있다. 그러나, 학습률 λ_t^Q는 다음의 정의에

따라, 매 전체 데이터 집합 반복수행에 대하여 경사 도함수 방향으로 목적함수 $f(x_t^Q+\lambda_t^Q h_t^Q)$를 최소화시켜야 한다.

$$\lambda_t^Q = \arg\min_{\lambda^Q} f\left(x_t^Q + \lambda^Q h_t^Q\right) \tag{21}$$

그러므로 식 (21)의 과정은 기계학습시, 최적의 학습률을 찾기 위한 내부 루프 (Inner-Loop)를 필요로 하게 되어 전체 데이터 집합때 마다, 내부적으로 반복 수행을 해야 한다. 특히, 양자화를 사용하는 기계학습의 경우, 양자화 오차에서 파생되는 학습 오차가 데이터당 반복 (Iteration)혹은 전체 데이터 집합 당 반복(Epoch)에서 더욱될 수 있으므로, 학습 오차를 최소화시킬 수 있는 최적 학습률을 내부 루프 없이 계산할 수 있는 알고리즘이 필요하다. 기존 기계학습에서는 적응적 학습률을 구현하기 위하여 일반적으로 Newton-Rapson 최적화법에 기반한 알고리즘 (AdaDelta, AdaGrad, ADAM등)이 존재했으나 이들 역시, 근사 Hessian을 사용하여 경사 도함수의 적용 비율에 대한 변화만을 주었을 뿐, 근본적으로 일정한 학습률을 사용하였다[6][7][8]. 때문에, 현재 제안된 거의 대부분의 확률적 최급 강하법에 의한 학습 방정식으로는 전체 데이터집합 반복 수행에서 국소적으로도 점근적인 수렴성(Asymptotically Convergence)을 Hessian의 최대값 제한과 같은 완비성 조건(Compact Condition)이 없이는 완전히 보장할 수 없다.

그러므로, 최소한 국소적 점근적 수렴성을 보장하기 위해서는 갱신항을 포함하는 학습 변수에 대한 목적함수 $f(x_t^Q+\lambda_t^Q h_t^Q)$와 갱신항을 포함하지 않는 학습변수에 대한 목적함수(Objective Function)의 차이가 항상 음의 값을 갖도록 다음의 관계가 성립하여야 한다.

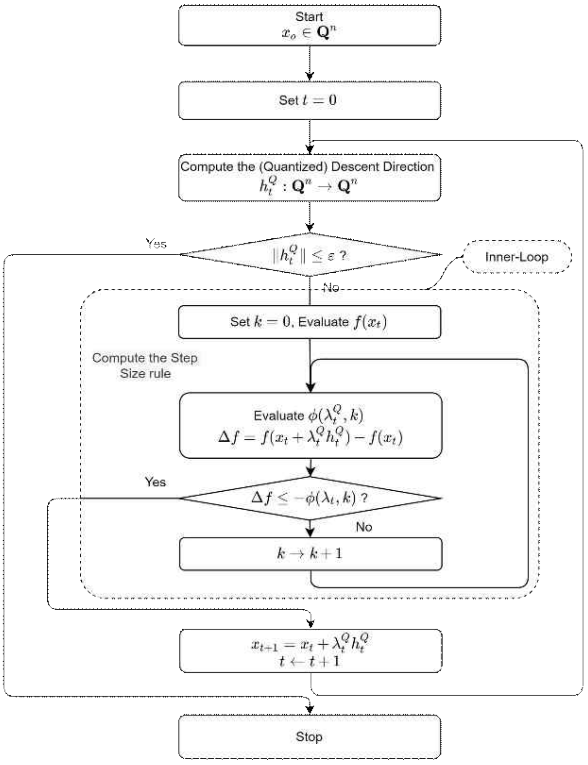

그림 1. 기존 비선형 최적화 기법에서 사용되는, 학습률 선택을 위한 내부 루프를 통해 결정되는 적응적 학습률 기반 학습 알고리즘
Fig. 1. The learning algorithm including the adaptive learning rate selected through an inner loop based on a conventional nonlinear optimization technique

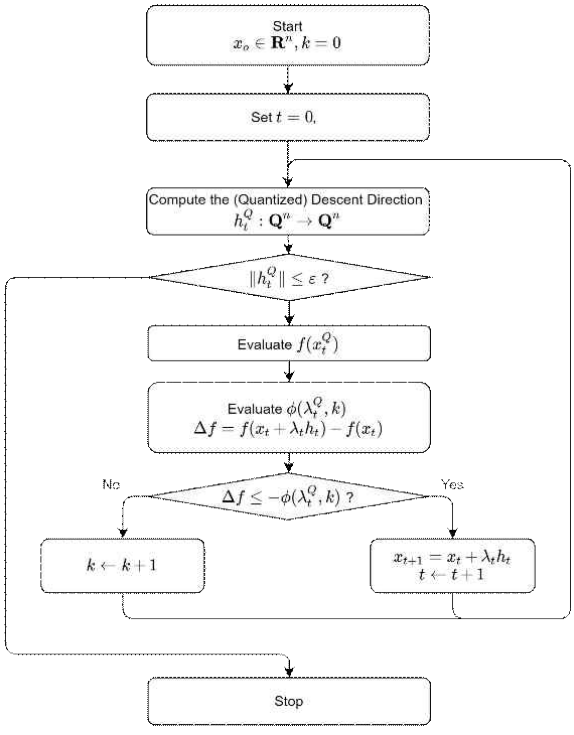

그림 2. 제안한 내부루프 없는 비선형 최적화 기법에 기반한 적응적 학습률을 포함한 학습 알고리즘
Fig. 2. The proposed learning algorithm without the inner loop for selection of the adaptive learning rate based on a nonlinear optimization technique

$$f(x_t^Q + \lambda_t^Q h_t^Q) - f(x_t^Q) \le -\phi(\lambda_t^Q, k) \tag{22}$$

식 (22)에서 $\phi(\lambda_t^q, k_t) \in \mathbf{R}^+$은 내부 루프 반복지표 k_t에 대한 단조감소 함수이다. 따라서, 내부 루프 반복지시자를 전체 데이터 집합 반복 수행 지시자에 통합하여 식 (22)을 만족하면 학습 변수가 갱신되도록 하며, 만족하지 못하면 변수가 갱신되지 않도록 하고, 내부루프 지시자를 증가시켜 함수 $\phi(\lambda_t^q, k_t)$가 감소하도록 한다. 이를 도시하면 그림 2와 같다.

제안한 알고리즘에서는, 식 (22)을 만족하면 알고리즘은 보통의 기계학습과 같다. 그러나, 식 (22)을 만족하지 못하면, 학습이 이루어지지 않고 더 작은 학습률로 다시 학습하는 것이기 때문에 본 과정은 마치 "Idle and Go"처럼 보인다. 또한 제안한 알고리즘은 만일, 유한한 내부 루프로 학습률을 찾을 수 있다면, 상대적으로 적은 전체 데이터 집합 반복수행으로 학습률을 찾을 수 있으므로 전체 학습 시간에서 차지하는 비중이 크지 않아, 효율적으로 적응적 학습률을 구할 수 있다.

Ⅳ. 양자화 오차 보상 방법

일반적으로 탐색 방향 벡터 $h_t \in \mathbf{R}^n$에 대한 양자화 벡터 $(h_t)^Q = Q_p h_t^Q \in \mathbf{R}^n$가 다음과 같이 구성되었다고 하자.

$$(h_t)^Q = \sum v_i e_i, \quad \forall i \in \mathbf{Z}[0, n-1],\ v_i \in \boldsymbol{Q} \tag{23}$$

식 (23)에서 $\mathbf{R}^n$는 유클리드 공간의 단위 직교 벡터로 $\forall i, j \in \mathbf{Z}[0, n),\ \| e_i \| = 1,\ e_i^T e_j = 0\ \ i \ne j$ 이다. 양자화 벡터 $(h_t)^Q$의 성분 중 가장 크기가 큰 성분을 $v_m = \max_i \| v_i \|$ 이라 하고 이것의 인덱스를 $m = \text{argmax}_i \| v_i \|$ 라 하자. 이때, 양자화 벡터 $(h_t)^Q$에서 $v_m = 0$로 놓은 것을 $\overline{v}$, v_m을 제외한 모든 성분이 0인 벡터를 $\hat{v}$라 하면 다음과 같다.

$$\overline{v} = \sum_{i=0}^{n-1} (\ \ 1 - \delta(i-m)) v_i e_i, \quad \hat{v} = \sum_{i=0}^{n-1} \delta(i-m) v_i e_i \tag{24}$$

이를 사용하여 양자화 벡터 $(h_t)^Q$를 다시 쓰면

$$(h_t)^Q = \overline{v} + \hat{v} = (\sum_{i=0}^{n-1} (1 - \delta(i-m)) + \delta(i-m)) v_i e_i \tag{25}$$

기존의 탐색방향 벡터 $(h_t)^Q$에서 가장 큰 성분에 대한 직교벡터를 구하기 위해 다음과 같이 벡터 z를 놓자

$$z_t = \overline{v} + r \cdot \hat{v}, \quad r \in \mathbf{R} \tag{26}$$

식 (26)에서 $r \in \mathbf{R}$는 $\hat{v}$에 대한 비례상수로서 이 값을 통해 직교 벡터 z를 구할 수 있다. 벡터 z와 벡터 $(h_t)^Q$와 직교성을 사용하여 r은 다음과 같이 구할 수 있다.

$$\begin{aligned} 0 &= \langle (h_t)^Q, z \rangle = \langle \overline{v} + \hat{v}, \overline{v} + r\hat{v} \rangle \\ &= \overline{v}^2 + (r+1)\langle \hat{v}, \overline{v} \rangle + r\hat{v}^2 \quad \because \langle \hat{v}, \overline{v} \rangle = 0 \\ &= \overline{v}^2 + r\hat{v}^2 \end{aligned} \tag{27}$$

$$\therefore r = -\frac{\overline{v}^2}{\hat{v}^2} = -\frac{\| (h_t)^Q \|^2 - v_m^2}{v_m^2} = 1 - \frac{\| (h_t)^Q \|^2}{v_m^2} \tag{28}$$

그런데, $|r| < 1$이므로 이를 그대로 학습 방정식에 적용하게 되면 정수 값으로 연산이 이루어지지 않으므로 보상 탐색 벡터는 양자화 된 값이 아닌, 일반적인 실수 벡터가 된다. 따라서 비례상수의 양자화를 고려하여 보상 탐색 벡터를 구하여야 한다. 식 (13)에서 $(h_t)^Q = Q_p h_t^Q$이므로 $v_m = Q_p v_m^Q$를 사용하여 정리하면

$$\hat{v} = \sum_{i=0}^{n-1} \delta(i-m) v_i e_i = Q_p \cdot v_m^Q e_m \tag{29}$$

식 (29)에서 $v_m^Q e_m \triangleq \hat{v}^Q$ 로 놓으면

$$\begin{aligned} z_t &= \overline{v} + r \cdot \hat{v} = \overline{v} + r \cdot Q_p \hat{v}^Q = \overline{v} + Q_p r \cdot \hat{v}^Q \\ &= \overline{v} + Q_p \hat{v}^Q - Q_p \hat{v}^Q + Q_p r \cdot \hat{v}^Q \\ &= (\overline{v} + \hat{v}) + Q_p (r-1) \hat{v}^Q \quad \because \hat{v} = Q_p \hat{v}^Q \end{aligned} \tag{30}$$

식 (24)(28) 그리고 $(h_t)^Q = Q_p h_t^Q$이므로,

$$z_t = Q_p h_t^Q - Q_p \frac{\| (h_t)^Q \|^2}{v_m^2} \hat{v}^Q \quad (31)$$

그러므로 식 (20)를 사용하여 $\hat{v}^Q$의 계수를 정리하면

$$Q_p \cdot \frac{\| (h_t)^Q \|^2}{v_m^2} = Q_p \cdot \frac{\sum_{i=0}^{n-1} v_i^2}{v_m^2} = \sum_{i=0}^{n-1} \frac{Q_p v_i^2}{v_m^2} \quad (32)$$

따라서, 식 (31), (32)을 사용하여 양자화를 하면

$$(z_t)^Q = Q_p h_t^Q - \left(\sum_{i=0}^{n-1} \left\lfloor \frac{Q_p v_i^2}{v_m^2} \right\rfloor + \left\lfloor \sum_{i=0}^{n-1} \frac{1}{v_m^2} \left(Q_p v_i^2 - v_m^2 \left\lfloor \frac{Q_p v_i^2}{v_m^2} \right\rfloor \right) + 0.5 \right\rfloor \right) \hat{v}^Q \quad (33)$$

식 (33)에서 $\left\lfloor \frac{Q_p v_i^2}{v_m^2} \right\rfloor$는 $\frac{Q_p v_i^2}{v_m^2}$의 몫이고 $Q_p v_i^2 - v_m^2 \left\lfloor \frac{Q_p v_i^2}{v_m^2} \right\rfloor$는 나머지이다. 나머지 부분을 다음과 같이 간략화 하면

$$Rem\left(\frac{Q_p v_i^2}{v_m^2} \right) = \frac{1}{v_m^2} \left(Q_p v_i^2 - v_m^2 \left\lfloor \frac{Q_p v_i^2}{v_m^2} \right\rfloor \right) \quad (34)$$

양자화된 직교 보상 탐색벡터는 다음과 같이 쓸 수 있다.

$$(z_t)^Q = Q_p h_t^Q - \left(\sum_{i=0}^{n-1} \left\lfloor \frac{Q_p v_i^2}{v_m^2} \right\rfloor + \left\lfloor Rem\left(\frac{Q_p v_i^2}{v_m^2} \right) \right\rfloor \right) \hat{v}^Q \quad (35)$$

식 (35)로 구해진 양자화된 직교 보상 탐색벡터는 양자화 기반 최급 강하법에서 $\phi(\lambda_t^Q, k)$ 함수가 충분히 작은 값임에도 학습 변수의 갱신이 이루어지지 않는 경우 탐색 방향벡터 h_t 대신 사용하게 된다. 이때, 식 (35)는 양자화 계수 Q_p의 스케일을 가지고 있으므로 이를 기본 양자화 보상 탐색벡터 $z_t^Q \equiv \frac{1}{Q_p}(z_y)^Q$로 놓고 h_t^Q 대신 z_t^Q를 사용하여 탐색한다. 이 때, 탐색결과 새로운 학습 변수로 갱신되면 다시 원래의 최적화 알고리즘을 사용하여 최적화 과정을 지속한다.

Ⅴ. 실험결과

본 알고리즘의 타당성을 실험하기 위하여 대표적인 비선형 최적화 문제인 로젠브록 함수의 최적화 문제에 제안한 양자화 방식을 적용하였다. 로젠브록 함수는 얇은 골짜기가 띠 형태로 존재하고 전역 최소점 부근에서 다수의 국소 최소점을 가지는 형상으로 때문에 각종 최적화 알고리즘의 성능시험에서 많이 사용된다. 특히 골짜기 영역에서는 학습률이 적절하지 않거나 알고리즘의 완비성이 떨어질 경우, 학습이 중단되거나 발산하게 된다. 또한 특정 시작점에서는 많은 알고리즘들이 발산하거나 학습, 혹은 최소화가 중단되는 특징이 있다[10][11]. 실험에 사용한 로젠브록 함수는 다음과 같다[6].

$$\forall x, y \in \mathbf{R}, f(x,y) = (a-x)^2 + b(y-x^2)|_{a=1,b=100} \quad (36)$$

식 (35)에서 최소점은 $(1,1) \in \mathbf{R}^2$이며 이때 함수 값은 $f(x,y)|_{x=1,y=1} = 0$이다. 알고리즘의 시작점은 알고리즘 실패가 잘 일어나는 (-1.232, 1.212)와, 최근 로젠블록 함수의 시작점으로 많이 선택되는 (-3.0, -4.0)을 사용하였다. 식 (37)의 경우 유리수체로 정의 가능하므로, $\frac{1}{Q_p} h_t \in \mathbf{Q}^n$를 잘 정의(Well-defined)할 수 있다. 최적화 알고리즘은 Quasi-Newton 방법인 BFGS(Broyden-Fletcher-Goldfarb-Shannon algorithm)[4]를 사용하였다. BFGS는 Newton-Rapson 기반의 최적화 알고리즘에서 나타나는 Hessian을 2개의 벡터 간 텐서곱으로 근사화 시킨 후 이를 업데이트 하면서 근사 Hessian을 구하여 최적화를 수행하는 방법이다. 학습률 선택을 위한 방법은 Armijo Rule을 사용하였다. Armijo Rule에 의한 학습률 선택 방법은 다음과 같다.

$$\lambda_t = \arg\max_{\beta^k} \left\{ \beta^k \middle| f(x_k + \beta^k h_t) - f(x_t) \le -\beta^k \alpha |\nabla f(x_t)|^2 \right\} \quad (37)$$

식 (37)에서 α는 0.0095, β는 0.9를 사용하였다.

실험 환경은 다음과 같다. 먼저 하드웨어는 삼성 Exynose-5 8-코어 CPU를 탑재한 83x32(mm) 크기의 임베디드 보드를 사용하였다. 동 시스템의 CPU는 2GHz에서 동작하는 ARM

A15 4-코어와 1.3GHz에서 동작하는 A7 4코어로 이루어져 있으며 933MHz로 동작하는 2GByte의 LPDDR3 메모리를 탑재하고, eMMC 5.0 HS4000 Flash Memory를 저장 장치로 사용한다. 소프트웨어 환경은 먼저, OS는 우분투 18.04 LTS이며 Python 3.7에서 numpy 및 Scipy 라이브러리를 사용하여 본 실험을 위한 코드를 작성하였다.

표 1은 본 논문에서 제안한 알고리즘에 대한 실험 결과이다. 표 1에서 양자화 계수 는 다음과 같이 서술된다. 먼저 Log Qp는 Q_p에 필요한 비트 수이고 Qp는 실제 값이다. 표 1에서 Best Epoch은 실제 최소값에 도달하기까지의 반복(Iteration)수이며, Cost값은 알고리즘이 찾은 최소 로젠브록 함수 값이다. “epsilon”은 현재 시간 인덱스 t와 직전 시간 t-1에서의 로젠브록 함수 값의 차이를 의s미한다. 이 값이 알고리즘 정지 조건인 10^{-3} 보다 작다면, 알고리즘은 정지한다. 실험결과, 양자화 계수가 128 즉, Log Qp로서 계산된 필요 비트가 7비트일 경우, 알고리즘이 찾은 최적 로젠블록 함수 값은 0으로 본 문제에서의 전역 최소값을 찾았으며 이때, Epsilon 값은 이다. 표 1에서 7비트 이상의 양자화 실험결과를 살펴보면 양자화시 반복회수가 부동 소수점을 사용한 결과대비 40.4%~44.1% 정도이다. 이 영향으로 속도는 각각 2.27~2.47배 빨라진다. 이는 부동소수점을 사용한 경사 도함수 기반 학습 방정식에서 약 50% 이상의 반복회수가 사실상 무의미하게 이루어짐을 의미하며, 양자화 연산을 통해 불필요한 반복을 줄이고, 적절한 양자화 오차 보상 방법이 구현되면, 부동 소수점 연산기 없이 대규모 신경망 하드웨어를 임베디드 시스템에 구현 시킬 수 있음을 보여준다.

표 1. BFGS 알고리즘과 제안한 알고리즘과의 최적화 성능: 초기값 (-1.232, 1.212)

Table 1. Performance Comparison between the BFGS and the proposed algorithm: Initial point at (-1.232, 1.212)

Log Qp	Qp	Best Epoch	Cost	epsilon	Processing Time (sec.)
L-BFGS		43	0	0	0.68813205
4	16	1	20.91852	-1.51367	0.00797892
5	32	2	19.42276	0	0.0090034
6	64	2	19.27893	0	0.01196694
7	128	46	0	6.14E-05	0.30318999
8	256	43	0	1.53E-05	0.28124738
9	512	42	0	3.45E-05	0.28921962
10	1024	41	0	3.45E-05	0.27825594

표 2. BFGS 알고리즘과 제안한 알고리즘과의 최적화 성능: 초기값 (-3.0, -4.0)

Table 2. Performance Comparison between the BFGS and the proposed algorithm : Initial point at (-3.0, -4.0)

Log Qp	Qp	Best Epoch	cost	epsilon	Processing Time (sec.)
L-BFGS		1730	0	1.0E-09	0.80185771
4	16	4	78.12500	-1.68307	0.01605585
5	32	9	70.75097	-0.30818	0.02991986
6	64	3	79.64070	-0.06783	0.02094483
7	128	7	68.77521	0	0.01097049
8	256	72	0	1.53E-05	0.04787230
9	512	68	0	8.59E-06	0.04221752
10	1024	66	0	8.59E-06	0.03490543

Ⅵ. 결 론

본 논문에서는 임베디드 시스템에서 비선형 최적화 방법론을 기반으로 양자화 학습을 최적하게 수행하기 위해 내부 루프 없는 적응적 학습률을 사용하는 학습 알고리즘과 양자화 오차를 최소화시킬 수 있는 보상 탐색 방법론을 제안하였다. 실험 결과 제안한 방법은 양자화 오차를 효과적으로 줄일 수 있음을 로젠블록 함수를 통한 비선형 최적화 문제를 통해 확인할 수 있었다. 향후 과제로는 제안한 알고리즘의 수학적 분석을 통한 수렴성 및 특성 분석과 상대적으로 대규모 데이터 및 대규모 기계학습 구조에서, 효과적으로 적용됨을 보이는 것이다.

참 고 문 헌 (References)

[1] R. M. Gray, and D. L. Neuhoff, "Quantization.", IEEE Transactions on Inform. Theory, Vol. 44, No. 6, pp. 2325 - 2383, June, 2006.

[2] D. Alistarh, D. Grubic, L. J. Ryota, and V. Milan, "QSGD: Communication-efficient sgd via gradient quantization and encoding.", Advances in Neural Information Processing Systems, Vol. 30, pp. 1709 - 1720, January, 2017

[3] G. Tenenbaum, ‘Introduction to Analytic and Probabilistic Number Theory’, Academic mathematical Society, 2014.

[4] D. G. Luenberger, Y. Ye, ‘Linear and Nonlinear Programming’, Springer, 2015.

[5] S. Sra, S. Nowozin, S.J.Wright, ‘Optimization for Machine Learning’,

MIT press, 2012.
[6] J. Duchi, E. Hazan, and Y. Singer. Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. The Journal of Machine Learning Research, 2011.
[7] M. Zeiler, "ADADELTA: an adaptive learning rate", arXiv preprint, https://arxiv.org/abs/1212.5701, arXiv:1212.5701, 2012.
[8] D. Kingma, J. Ba. Adam: A Method for Stochastic Optimization. International Conference for Learning Representations, 2015.
[9] S. M. Goldfeld, R. E. Quandt, and H. F. Trotter, "Maximization by Quadratic Hill-Climbing", Econometrica, pp. 541-551, July, 1966.
[10] S. Boyd, L. Vandenberghe, Convex Optimization, Cambridge University Press, Cambridge, 2004
[11] M.S. Bazaraa, H.D. Sherali, C.M. Shetty, Nonlinear Programming: Theory and Algorithms. Wiley-Interscience, New Jersey, 2006

저 자 소 개

석 진 욱

- 1995년 2월 : 홍익대학교 전자공학과 공학석사
- 1998년 8월 : 홍익대학교 전자공학과 공학박사
- 2000년 ~ 현재 : 한국전자통신연구원 책임연구원
- 2006년 ~ 2019년 : 과학기술연합대학원대학교 겸임교수
- 2020년 ~ 현재 : 과학기술연합대학원대학교 전임교수
- ORCID : https//orcid.org/0000-0001-5318-1237
- 주관심분야 : 영상처리/압축, 기계 학습, 비선형 확률제어