アニーリングマシンを活用したエッジAIにおける生成モデルの学習効率化のためのアーキテクチャ

鶴田 博文[1,a)] 松本 亮介[1,b)]

概要：IoT デバイスが生成するデータを活用した課題解決の手法として，人工知能（AI）を用いてデバイス上で知的なタスクを実行するエッジ AI が注目されている．エッジ AI に実装する AI 技術として，画像・文章生成や異常検知等の幅広い応用が可能な生成モデルが期待されている．生成モデルの学習には，一般に計算コストが高い確率分布のサンプリングが必要であり，学習速度や精度に強く影響を与える．新たな学習手法として，量子アニーリングを実装した D-Wave マシンを用いて生成モデルの一つであるボルツマンマシンの学習効率化に成功している．一方，この学習手法をエッジ AI に適用する場合，D-Wave マシンは費用や動作環境等の理由からエッジ領域への配置は困難であり，クラウドサービスとしての利用となるため，デバイスとクラウド間の距離に起因した通信遅延が発生し，学習・推論のボトルネックとなる．また，一般にエンドデバイスは処理能力が低いため，サンプリングをデバイス上で実行することは難しい．本研究では，生成モデルの学習アクセラレータとして，D-Wave マシンに比べ小型かつ低コストのアニーリングマシンをデバイスの近傍に配置することで，エッジ AI において生成モデルの学習をクラウドを介さずに効率化するアーキテクチャを提案する．提案手法により，学習・推論への通信遅延の影響を抑えつつ，汎用的かつ低スペックなデバイス上で高い応用性をもつ生成モデルの活用を可能にする．評価の結果から，アニーリングマシンを用いてボルツマンマシンの学習を効率化できること，およびアニーリングマシンをデバイスの近傍に配置することで，クラウド経由で D-Wave マシンを用いる場合と比べて，学習・推論時間の高速化が期待できることを示した．

System Architecture for Efficient Training of Generative Models in Edge AI Using Annealing Machines

Hirofumi Tsuruta[1,a)] Ryosuke Matsumoto[1,b)]

1. はじめに

IoT デバイスの普及により，多様性に富んだ大量のデータが日々生成され，データの蓄積や分析等をクラウドと連携しながら実現するサービスが増えている [1]．これらのデータを活用した新たな課題解決・価値創造を実現するために，データを深層学習に代表される人工知能（AI）技術を用いて学習し，エンドデバイス上で認識や予測等の知的なタスクを実行するエッジ AI[2] が注目されている．エッジ AI の実現には計算コストが高い AI の学習や推論の処理が必要であり，一般にエンドデバイスは処理能力が低いため，より潤沢な処理能力を有するクラウド上で実行する形態が取られることがある．一方で，クラウドを介する場合，デバイスとクラウド間の距離に起因した通信遅延が発生し，その遅延は概ね数百 ms オーダー [3] である．そのため，デバイス上で数十 ms またはそれ以下の応答性が求められる場合において，AI の学習や推論をクラウドを介さずにエンドデバイス上，あるいはデバイスに物理的に近いネットワーク上の計算資源で行うことが求められる．そこで，処理能力が高い小型デバイスの開発 [4] や，処理能力が低いデバイス上で実現可能な学習モデルの圧縮技術 [5] が研究されている．

エッジ AI に実装する AI 技術の一つとして，大量のデー

[1] さくらインターネット株式会社，さくらインターネット研究所
SAKURA internet Research Center, SAKURA internet Inc.
[a)] hi-tsuruta@sakura.ad.jp
[b)] r-matsumoto@sakura.ad.jp

タの構造を解析して，データ生成の仕組みそのものを学習する生成モデル [6] が注目されている．生成モデルは，学習後のモデルから新たなデータを生成することができるため，エンドデバイス上で画像・会話文生成等の創造的なタスクや異常検知，欠損値補完・ノイズ除去など幅広い応用ができる．多くの生成モデルの学習・推論には，確率分布からのサンプリングが必要であり，この処理が学習速度や精度に大きく影響を与えるため，効率的なサンプリングが可能なモデルや学習アルゴリズムの研究が進んでいる [7], [8]．

生成モデルの学習に対する新たなアプローチとして，量子アニーリング [9] を活用した手法が提案されている．D-Wave Systems 社が開発する量子アニーリングマシン（以下，D-Wave マシン）は，確率分布からのサンプリングを高速に実行可能であり，生成モデルの枠組みの一つであるボルツマンマシンの学習の高速化・高精度化に成功している [10], [11]．一方，量子アニーリングを活用した生成モデルの学習の仕組みをエッジ AI に適用する場合，D-Wave マシンは費用や極低温の動作環境を要するなどの理由からエッジ領域への配置は困難であり，ネットワークを経由したクラウドサービスとしての利用となる．すなわち，クラウドを介さずエッジ側で学習を行うことができないため，デバイスとクラウド間の物理的な距離に起因した通信遅延が発生する．D-Wave マシンのサンプリングにかかる時間が概ね μs オーダー [11] であること，およびクラウドとの通信遅延が概ね数百 ms オーダー [3] であることを考えると，生成モデルの学習や推論において通信遅延がボトルネックになる．また，例えば，デバイス上で実行するアプリケーションが会話文生成などの人間とのインターフェースを担うものであれば，人間が遅延を感じない数十 ms オーダーの応答性が要求されるため [12]，クラウドとの通信遅延が大きな問題となる．一方，計算コストが高いサンプリングは，処理能力が低いデバイス上では膨大な計算時間を要するため，サンプリングの処理を必要とする生成モデルをクラウドを介さずにエンドデバイスで活用することは困難である．

本研究では，生成モデルの学習・推論に必要なサンプリング処理のアクセラレータとして，D-Wave マシンに比べ小型かつ低コストのアニーリングマシンをデバイスの近傍に配置することで，エッジ AI において生成モデルの学習をクラウドを介さずに効率化し，かつ推論時のデバイスの応答性を向上させるアーキテクチャを提案する．ここで，アニーリングマシンとは，量子現象に着想を得たデジタル回路上で動作するアニーリング専用機のことであり，例として CMOS アニーリングマシン [13] やデジタルアニーラ [14]，Graphics Processing Unit（GPU）マシン上で動作するソフトウェア実装 [15], [16] などを指している．提案するアーキテクチャにより，デバイスは汎用的かつ低スペックでありながら，計算コストが高いサンプリングの処理をアニーリングマシンが担うことで，画像・会話文生成等の創造的なタスクや収集データのノイズ除去等の高い応用性を持つ生成モデルをデバイス上で活用することを可能にする．さらに，提案手法ではデバイスとアニーリングマシン間の通信遅延を数 ms 程度に抑えつつ，アニーリングマシンがサンプリング処理を高速化することで，クラウド経由で D-Wave マシンを用いる場合と比べて，生成モデルの学習・推論時間の高速化が期待できる．

本稿の構成を述べる．2 章では，エッジ AI において生成モデルを活用するための関連技術を整理し，課題について述べる．3 章では，本論文での提案を述べる．4 章では，提案手法の評価の結果を述べる．最後に，5 章でまとめとする．

2. エッジ AI における生成モデルの活用と課題

画像・会話文生成等の創造的なタスクや異常検知などの幅広い応用性を持つ生成モデルの多くは，一般に計算コストが高い確率分布からのサンプリングが必要であるため，クラウドを介さずに処理能力が低いデバイス上で効率的に学習・推論を行うことは困難である．一方で，生成モデルの学習・推論をクラウドを介して行う場合，デバイスとクラウド間の距離に起因した通信遅延が学習や推論のボトルネックになる．本章では，エッジ AI において生成モデルを活用するための関連技術を整理し，課題について述べる．

2.1 エッジ AI

IoT デバイスが生成する多種多様なデータを深層学習に代表される AI 技術を用いて学習し，エンドデバイス上で認識や予測等の知的なタスクを実行するエッジ AI[2] が注目されている．エッジ AI はデータが発生する現場において，データを活用した新たな課題解決・価値創造のアプローチとして，自動運転車や産業ロボット，監視カメラ，ヘルスケアデバイスなど幅広い分野への適用が進められている [17]．エッジ AI の実現方法は，AI の学習と推論の二つのプロセスをどの計算資源上で実行するかにより多岐に渡っているため [2]，目的や要求に応じて適切に選択する必要がある．

一般にエンドデバイスは処理能力が低いため，計算コストが高い学習や推論の一部の処理を，より潤沢な処理能力を有するクラウド上で実行する形態が取られることが多い．この場合，デバイスに要求される処理能力が低いため，汎用的かつ低スペックなデバイスを活用できる利点がある．一方で，クラウドを介する場合，デバイスとクラウド間の距離に起因した通信遅延が発生し，その遅延は概ね数百 ms オーダー [3] である．例えば，デバイス上で実行するアプリケーションが会話文生成などの人間とのインターフェースを担うものを想定した場合，人間が遅延を感じない数十 ms オーダーの応答性が要求されるため [12]，デバイスとクラウド間の通信遅延のみで要求される応答時間を超える．

このような場合，AI の学習や推論をクラウドを介さずにエンドデバイス上，あるいはデバイスに物理的に近いネットワーク上の計算資源で行うことが求められる．

2.2 エッジ AI における生成モデル

エッジ AI に実装する AI 技術の一つとして，データ生成の仕組みそのものを学習する生成モデル [6] が注目されている．生成モデルは，観測データの背後にあるデータの生成過程をモデル化するため，学習後のモデルから新たなデータが生成できる．この特徴を活かして，生成モデルはデバイス上で画像・会話文生成等の創造的なタスクや異常検知，収集データの欠損値補完・ノイズ除去など幅広い応用が可能である．

生成モデルの多くは学習・推論のプロセスにおいて，高次元確率分布からのサンプリングが必要であり，この処理が学習速度や精度に大きく影響を与える．一般に，このサンプリングの処理は計算コストが高いため，クラウドを介さずに処理能力が低いデバイス上で効率的に学習や推論を行うことは難しい．一方で，クラウドを介する場合，2.1 節で述べた通り，デバイスとクラウド間の距離に起因した通信遅延がデバイスの応答性を低下させ，問題となるケースが存在する．

2.3 ボルツマンマシンと量子アニーリング

生成モデルの学習に対する新たなアプローチとして，量子アニーリングを実装した D-Wave マシンを活用した手法が提案され，ボルツマンマシンの学習において，従来の学習手法と比較してより少ないパラメータ更新回数で高い精度が得られることが示されている [10], [11]．D-Wave マシンで学習に成功したボルツマンマシン [18] は，Hinton らに提案された生成モデルの有力な枠組みの一つである．ボルツマンマシンは，高次元の複雑な確率分布を学習できるという特徴を活かして，これまでに会話や人体モーションのモデリング [19], [20]，異常検知 [21] など，幅広い分野に応用されている．

量子アニーリングを活用したボルツマンマシンの学習の仕組みをエッジ AI に適用する場合，D-Wave マシンは費用や極低温の動作環境を要するなどの理由からエッジ領域への配置は困難である．D-Wave マシンを利用する際は，D-Wave Systems 社が提供するクラウドサービス [22] を Web API 経由で利用する．すなわち，クラウドを介さずエッジ側で学習を行うことができないため，デバイスとクラウド間の物理的な距離に起因した通信遅延が発生する．D-Wave マシンのサンプリングにかかる時間が概ね μs オーダー [11] であること，およびクラウドとの通信遅延が概ね数百 ms オーダー [3] であることを考えると，ボルツマンマシンの学習や推論において通信遅延がボトルネックになる．

2.4 D-Wave マシンとアニーリングマシン

D-Wave マシンは，量子アニーリングを動作原理として実装したハードウェアである．D-Wave マシンは，量子力学的なゆらぎを利用してイジングモデルの基底状態を探索することにより組合せ最適化問題を解く．これまでに，D-Wave マシンは組合せ最適化問題の一般的な解法として様々な分野の問題への適用が進んでいる [23]．一方，D-Wave マシンは動作温度が極低温であるものの絶対零度でないことや，製造技術上の問題などから厳密な最適解ではない状態を解候補として出力することが知られている [24]．この性質を利用して，D-Wave マシンを確率分布からのサンプリングに活用する試みが進められており，その一つが前節で述べたボルツマンマシンへの応用である．

D-Wave マシンのような量子ビットの振る舞いをデジタル回路上でシミュレートするものとして，アニーリングマシンがある．ここでのアニーリングマシンとは，例として CMOS アニーリングマシン [13] やデジタルアニーラ [14], GPU マシン上で動作するソフトウェア実装 [15], [16] などを指している．アニーリングマシンは，D-Wave マシンとは異なり量子力学の原理に基づくものではないが，イジングモデルの基底状態探索を高速に実行することができる．上記で挙げたアニーリングマシンは，大規模かつ NP 困難な組合せ最適化問題の解法として，その有効性が示されている．例えば，CMOS アニーリングマシンは，10 万ビットの組合せ最適化問題を従来手法であるシミュレーテッドアニーリングを CPU 上で実行するよりも約 150 倍速い 10 ms 程度で解くことができる [13]．さらに，アニーリングマシンは，D-Wave マシンと比べて小型かつ低コストであり，常温で動作するなど特殊な動作環境を要しないという特徴を有している．

3. 提案するエッジ AI における生成モデルの活用のためのアーキテクチャ

2 章で述べた通り，生成モデルの一つであるボルツマンマシンの学習における新たなアプローチとして，D-Wave マシンを活用し，ボルツマンマシンの学習・推論のプロセスに必要な確率分布からのサンプリングを効率化する手法が提案され，既存の学習手法よりも少ないパラメータ更新回数で高い精度が得られることが示されている．この学習の仕組みをエッジ AI に適用する場合，D-Wave マシンはネットワークを経由したクラウドサービスとしての利用となるため，デバイスとクラウド間の物理的な距離に起因した通信遅延が学習・推論のボトルネックとなる．一方，一般に確率分布からのサンプリングは計算コストが高いため，処理能力が低いデバイス上で効率的に学習・推論を行うことは困難である．これらの課題を解決するために，生成モデルの学習・推論に必要なサンプリング処理のアクセラレータとして，D-Wave マシンに比べ小型かつ低コスト

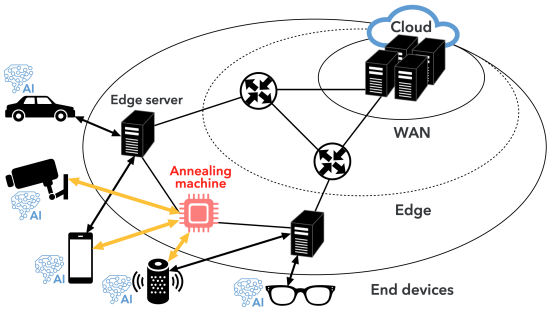

図 1　提案するアーキテクチャの概要図

のアニーリングマシンをデバイスの近傍に配置することで，エッジ AI において生成モデルの学習をクラウドを介さずに効率化し，かつ推論時のデバイスの応答性を向上させるアーキテクチャを提案する．

3.1　アーキテクチャ概要

図 1 は提案するアーキテクチャの概要図である．エンドデバイスはデータを生成し，デバイス上で生成モデルを活用して知的なタスクを実行する．エッジ領域に配置したアニーリングマシンは，サンプリング処理のアクセラレータとして複数台のデバイスに共有される．デバイスにおいて，生成モデルの学習や推論の過程でサンプリング処理が必要な際に，デバイスはアニーリングマシンに計算を要求する．アニーリングマシンはデバイスからの要求を受け，サンプリングを実行し，計算結果をデバイスに返す．

3.2　エッジ AI における生成モデルの学習および推論

2.2 節で述べた通り，多くの生成モデルは学習や推論のプロセスにおいて，計算コストが高い確率分布からのサンプリングが必要であるため，処理能力が低いデバイス上で効率的に学習・推論を行うことは困難である．提案手法では，生成モデルの学習や推論に必要なサンプリング処理のアクセラレータとしてアニーリングマシンを活用する．2.3 節で述べた通り，D-Wave マシンは生成モデルの一つであるボルツマンマシンの学習・推論に必要な確率分布からのサンプリングを高速に実行できる．アニーリングマシンは，量子力学的なゆらぎを利用しないなど動作原理は D-Wave マシンと異なるが，イジングモデルの基底状態を探索するという目的や対象とする問題は D-Wave マシンと同様である．そのため，アニーリングマシンを D-Wave マシンと同様に生成モデルの学習や推論に必要な確率分布からのサンプリングに活用できると考えられる．

アニーリングマシンを活用することで，生成モデルの学習・推論に必要なサンプリング処理の高速化が期待できる．2.4 節で述べた通り，例えば CMOS アニーリングマシンでは，大規模な組合せ最適化問題を CPU マシンよりも 100 倍以上高速かつ数十 ms オーダーで解く．サンプリングでは，最適化問題を解く操作を多数回繰り返し，得られた分布をもつ解候補の平均値を計算する．したがって，これまで示されているアニーリングマシンの組合せ最適化問題に対する高速性は，生成モデルの学習・推論に必要なサンプリング処理においても同様に得られると考えられる．これにより，デバイスは汎用的かつ低スペックでありながら，計算コストが高いサンプリングの処理をアニーリングマシンが担うことで，幅広い応用性を持つ生成モデルをデバイス上で活用することを可能にする．また，D-Wave マシンを用いた学習手法では，高速なサンプリング処理によりモデルの期待値を直接推定するというアプローチをとり，生成モデルの一つであるボルツマンマシンの学習効率化に成功している．提案手法においても，アニーリングマシンを用いて同様のアプローチにより学習を行うため，既存の学習手法よりも学習を効率化することが期待できる．一方で，D-Wave マシンによるボルツマンマシンの学習効率化には量子効果が寄与していることが示唆されており [10]，アニーリングマシンでは量子効果を利用していない．そのため，アニーリングマシンにおいても同様にボルツマンマシンの学習を効率化できるかは検証が必要である．この点については，4 章で評価し，議論する．

3.3　デバイスの応答性と通信遅延

デバイス上で数十 ms オーダーあるいはそれ以下の応答性が求められる場合，デバイスとクラウド間の通信遅延が問題になる．例えば，デバイス上で実行するアプリケーションが会話文生成などの人間とのインターフェースを担うものであれば，人間が遅延を感じない数十 ms オーダーの応答性が要求されるため [12]，クラウドとの通信遅延のみで要求される応答時間を超える．提案手法では，デバイスの近傍にアニーリングマシンを配置することで，推論時のデバイスの応答性に対する通信遅延の影響を抑えるアーキテクチャをとる．アニーリングマシンは，D-Wave マシンと比べて小型かつ低コストであり，常温で動作するなど特殊な動作環境を要しないため，デバイスに近いエッジ領域に配置することに適している．

生成モデルの学習・推論に必要なサンプリング処理に D-Wave マシンを用いる場合，D-Wave マシンはサンプリング時間が概ね μs オーダー [11] の非常に高速であるが，ネットワークを経由したクラウドサービスとしての利用となるため，数百 ms オーダー [3] の通信遅延が発生する．この場合，サンプリング時間が高速であっても，デバイスの応答時間は数百 ms オーダー以上となる．一方，提案手法では，アニーリングマシンをエッジ領域に配置することで通信遅延は数 ms オーダーに抑えられる．そのため，デバイスの応答時間に対する通信遅延の影響が小さく，アニーリングマシンのサンプリング時間を高速化できれば，それに伴いデバイスの応答時間も高速化できる．サンプリング時間や通信遅延がデバイスの応答性に与える影響や，提案

表 1　実験環境

	項目	仕様
Device	CPU	Intel Xeon CPU E5-2650 v3 2.30GHz 1core
	Memory	1 GBytes
	Software	bmpy v0.1.0
Annealing machine	CPU	Intel Xeon CPU E5-2650 v3 2.30GHz 8core
	Memory	32 GBytes
	Software	Sqaod v1.0.2

手法により期待できるデバイスの応答時間は，4 章で評価し，議論する．

4. 評価

提案するアーキテクチャは，デバイス近傍に配置したアニーリングマシンを活用することで，ボルツマンマシンの学習を効率化し，かつ推論における通信遅延の影響を抑えてデバイスの応答性を向上させる．本章では，まずアニーリングマシンを用いてボルツマンマシンを効率的に学習できるかについて評価する．評価においては，ボルツマンマシンの従来の学習手法と学習速度や精度について比較を行う．次に，アニーリングマシンを通信遅延が小さいデバイス近傍に配置することで推論時のデバイスの応答性を高速化できるかについて評価する．特に，クラウドを利用する場合，すなわち数百 ms オーダーの通信遅延が発生するものの潤沢な計算資源を利用して処理速度を高められる場合と比較して，提案するアーキテクチャがデバイスの応答性の観点で有効性を示すかを議論する．

4.1 実験条件

表 1 は，各ロールの性能と用いたソフトウェアのバージョンを示している．各ロールの OS は全て Ubuntu 18.04.4 Kernel 4.15.0 である．ここで，表 1 で挙げた二つのロールの役割と用いるソフトウェアについて述べる．デバイスのロールは，学習データを保持し，ボルツマンマシンの学習・推論を行う役割を担い，学習・推論の過程でアニーリングマシンにサンプリングを要求するクライアントとして動作する．ボルツマンマシンの学習・推論には，その過程で必要なサンプリングの処理を独自に組み込めるように実装した bmpy[25] を用いる．

アニーリングマシンのロールは，デバイスからのサンプリング要求を受け，計算結果を返す．サンプリングの処理には，経路積分量子モンテカルロ法を用いて量子アニーリングをシミュレートする Simulated Quantum Annealing（SQA）[26] を用いる．評価においては，SQA をソフトウェアとして実装したものを CPU マシン上で動作させる．SQA の計算処理をなるべく高速に実行するために，マルチコア上で SQA の並列処理が可能な Sqaod[27] を採用した．アニーリングマシンでは，デバイスからのサンプリング要求を受けるため，Sqaod を用いて Web API を構築した．

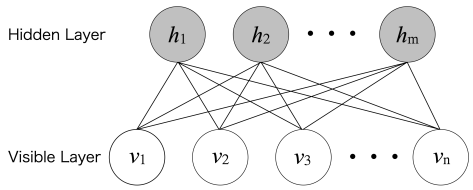

図 2　RBM のグラフ構造

学習データには，手書き数字画像のデータセットである MNIST[28] を用いた．MNIST には，60000 枚の訓練用の画像データと，10000 枚の検証用の画像データが含まれている．1 つの画像は，0 から 9 までの数字を表現するグレースケールの 784 ピクセルで構成される．学習には，グレースケールの画像をバイナリ化したものを用いる．

4.2 制限ボルツマンマシン

評価に用いる生成モデルとして，ボルツマンマシンの一種である制限ボルツマンマシン（Restricted Boltzmann Machine; RBM）[29] を用いた．図 2 は，RBM のグラフ構造を示している．RBM は，図 2 のように可視層と隠れ層から構成される 2 部グラフであり，ユニット間の結合が異層間のみに制限された構造をもつ．データに対応する n 次元の二値の可視変数 $\boldsymbol{v} \in \{0,1\}^n$ と m 次元の隠れ変数 $\boldsymbol{h} \in \{0,1\}^m$ において，同時確率分布関数はエネルギー関数 $E(\boldsymbol{v},\boldsymbol{h})$ を用いて以下のように定義する．

$$p(\boldsymbol{v},\boldsymbol{h}) = \frac{1}{Z}\exp(-E(\boldsymbol{v},\boldsymbol{h})) \tag{1}$$

$$E(\boldsymbol{v},\boldsymbol{h}) = -\sum_i b_i v_i - \sum_j c_j h_j - \sum_{i,j} v_i h_j w_{ij} \tag{2}$$

ここで，Z は分配関数，b_i および c_j はそれぞれ可視ユニットおよび隠れユニットに対応したバイアス，w_{ij} はユニット間の相互作用を表現する重みである．

RBM の学習では，観測データ群に対する対数尤度関数の最大化により，観測データ群を最も高い確率で生成するようにモデルのパラメータを調整する．対数尤度関数の最大化は，以下の式のように対数尤度 $\log P$ のパラメータに対する勾配を利用した勾配上昇法で反復的に解かれる．

$$\frac{\partial \log P}{\partial w_{ij}} = \langle v_i h_j \rangle_{data} - \langle v_i h_j \rangle_{model} \tag{3}$$

ここで，右辺第 1 項 $\langle v_i h_j \rangle_{data}$ は観測データの期待値であり，データと RBM の条件付き独立性から計算できる．一方，第 2 項 $\langle v_i h_j \rangle_{model}$ は式 (1) で定義される同時確率分布の期待値であり，可視および隠れユニットの全ての可能な実現値の組み合わせの総和であるため，ユニット数の増加とともに厳密に計算することが困難になる．このような問題は組み合わせ爆発と呼ばれ，ボルツマンマシンの学習を困難にする根本的な要因である．RBM の学習は，式 (3) の右辺第 2 項の期待値の推定をいかにして行うかが，学習速度や学習後の精度に大きく影響を与える．

4.3 アニーリングマシンを用いた学習の評価

RBM の学習は，厳密には計算困難な期待値計算が必要であるため，様々な近似学習アルゴリズムが提案されている．現在，RBM の学習手法として，コントラスティブ・ダイバージェンス（Contrastive Divergence; CD）法 [30] が最も広く利用されている．CD 法は，サンプリングの初期値にデータを用いるという発想と RBM の条件付き独立性により，サンプリングの計算コストを大幅に軽減している．CD 法は，理論的には式 (3) に基づく厳密な勾配に従わない近似した学習手法であるが，CD 法を用いて RBM の学習に成功することが経験的に知られている．一方，アニーリングマシンを用いた学習手法では，アニーリングマシンのイジングモデルの基底状態探索に対する高速性を活用することで，近似を介さず確率分布のサンプリングを通して式 (3) を直接推定するアプローチである．

アニーリングマシンを用いて RBM を効率的に学習可能であるか評価するために，RBM を CD 法および SQA を用いて学習し，比較を行った．RBM の学習の処理は主に，サンプリングと行列演算から構成される．前者は，式 (3) 右辺第 2 項の期待値を推定するための処理である．後者は，推定した期待値を用いたパラメータ更新等の処理である．CD 法を用いた学習では，全ての処理をデバイス上で実行する．一方，SQA を用いた学習では，行列演算はデバイス上で実行し，サンプリングの処理のみを Web API 経由してアニーリングマシンで実行する．いずれの学習手法においても，60000 枚の訓練用画像を 100 枚ずつのミニバッチに分割し，ミニバッチ単位でパラメータ更新を行う．ミニバッチ単位での学習の繰り返し回数をイテレーションと呼び，全学習データを用いた学習の繰り返し回数をエポックと呼ぶ．今回の条件では，1 エポックあたり 600 イテレーションが含まれおり，1 イテレーションの中でサンプリングの処理が 1 回実行される．RBM はエポックを繰り返すことで学習を進めていく．

それぞれの学習手法の評価の指標として，再構成誤差 [29] を用いた．MNIST の 10000 枚の検証用の画像それぞれについて，RBM の条件付き独立性に基づき画像を再構成する．再構成誤差は，得られた再構成画像が元の画像をどれだけ再現できたかを以下の式から評価する．

$$error = \frac{\sum_i (image(i) \neq reconstructedImage(i))}{total\ pixels} \quad (4)$$

再構成誤差は，学習後のモデルが学習データの生成過程をどれだけうまく学習できているかの指標となり，低い値ほど良いモデルと言える．

表 2 は，SQA による RBM の学習に用いたパラメータを示している．いずれの学習手法においても，RBM の隠れ層のユニット数は 100 を，学習率には 0.01 を用いた．図 3 は，CD 法および SQA を用いて RBM を学習した際のエポック数に対する再構成誤差を示している．どちらの学習手法においても，エポック数が大きくなる，すなわち学習が進むにつれて，再構成誤差が小さくなっている．また，SQA は CD 法に比べて，少ないエポック数で低い再構成誤差に到達しており，収束後の再構成誤差も低い値を示している．したがって，今回の条件において，SQA は CD 法よりも RBM を効率的に学習できていると言える．CD 法は多くの性能改善をした派生アルゴリズム [31], [32] が提案されているため，今回のような SQA による学習効率化が広範な問題や条件においても期待できるかを判断するには，更なる検証が必要である．

表 2　SQA による学習に用いたパラメータ

パラメータ	設定値
トロッター数	200
アニーリングステップ数	100
温度	0.005
初期の横磁場の強さ	5
最終の横磁場の強さ	0.001

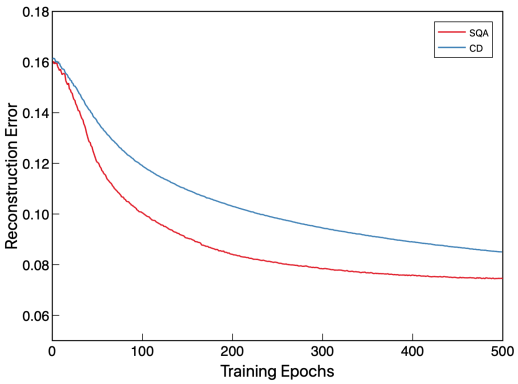

図 3　エポック数に対する再構成誤差

図 3 では，学習の進行をエポックで表現した．このエポックあたりに要する時間は学習手法に強く依存する．CD 法および SQA のエポックあたりの平均学習時間は，CD 法が 5.21 s，SQA が 282 s であった．CD 法は SQA に比べてエポックあたりの学習時間が短い．これは，CD 法は近似によりサンプリングの計算コストを大幅に軽減しているのに対し，SQA では近似を介さず計算コストが高い同時確率分布のサンプリングを通して式 (3) を直接推定することに起因する．SQA を用いた学習では，サンプリングにかかる時間がエポックあたりの学習時間に大きく影響を与えるため，より処理能力が高いアニーリングマシンを用いることで学習時間の短縮が期待できる．

4.4 通信遅延の影響の評価

デバイスとアニーリングマシン間の通信遅延が RBM の推論時間に与える影響を評価するために，提案手法であるアニーリングマシンをエッジ領域に配置した環境およびアニーリングマシンをクラウドに配置した環境のそれぞれについて，推論時間を評価した．評価においては，アニーリ

表 3　RBM の推論時間の計測結果

計測項目	提案手法	クラウド環境
推論時間/ms	351	701
サンプリング時間/ms	291	286
サンプリング時間が占める割合/%	82.9	40.8

ングマシンの位置を北海道石狩市のデータセンターに固定し，通信遅延が小さいエッジ環境としてデバイスを同データセンター内に配置した場合，通信遅延が大きいクラウド環境としてデバイスをアメリカ合衆国バージニア州のデータセンター内に配置した場合で比較を行う．提案手法およびクラウド環境のそれぞれにて，ping コマンドを用いてラウンドトリップタイムを測定したところ，0.612 ms および 176 ms であった．

ここでの RBM の推論は，デバイス上で学習後のモデルから画像を 1 枚生成することに相当する．推論の処理は，デバイスがアニーリングマシンに対して RBM からのサンプリングを要求し，アニーリングマシン上でサンプリングを実行され，その結果をデバイスが受け取るという流れで実行される．すなわち，推論時間には，アニーリングマシンとの通信遅延および，アニーリングマシン上でのサンプリング時間が含まれている．表 3 は，RBM の推論時間およびサンプリング時間の計測結果を示している．アニーリングマシン上でのサンプリング時間が推論時間に対して占める割合も併せて示している．提案手法およびクラウド環境の推論時間はそれぞれ，351 ms および 701 ms であり，クラウド環境が提案手法に比べて 350 ms 遅い結果となった．両環境において，サンプリング時間は同等であることや同性能のデバイスを用いていることを考慮すると，この差分は，クラウド環境が提案手法に比べてデバイスとアニーリングマシン間の物理的な距離が遠くなることで生じる通信遅延に起因していると考えられる．

デバイスがサンプリングの処理をアニーリングマシンに要求する際には，表 2 に示した SQA のパラメータに加えて，RBM のバイアスおよび重みパラメータの情報を渡す必要がある．実験では，これらの情報を HTTP リクエストのリクエストボディに JSON 形式で記述しており，リクエストボディのサイズは 0.56 MB であった．より複雑なデータを取り扱う場合，RBM のユニット数がさらに大きくなることでリクエストボディのサイズが大きくなり，通信遅延の影響はより顕著になると考えられる．また，RBM の学習においては，4.3 節で述べた通り，1 エポックあたり 600 回のサンプリングが実行されるため，学習時間に対する通信遅延の影響は推論時間よりも顕著に大きくなる．

ここで，サンプリングの処理に D-Wave マシンを用いた場合との比較について議論する．前提として，論文執筆時に現行の D-Wave 2000Q[33] では，ビット数およびビット同士の結合の制約から実用的なユニット数を持つ RBM を学習できない．今回用いた RBM も同様に D-Wave マシンでは学習できないため，D-Wave マシンを用いた場合との比較を直接行うことはできない．一方，D-Wave マシンをネットワークを経由したクラウドサービスとして利用した場合の通信遅延の影響は，今回の評価のクラウド環境の結果と同程度と考えられる．もし，D-Wave マシンで今回用いた RBM の学習が可能な場合，サンプリング時間はアニーリングマシンより高速であり，概ね μs オーダーであることが予想される．しかしながら，表 3 の結果から，例えサンプリング時間が μs オーダーであっても，通信遅延がボトルネックとなり，推論時間は提案手法よりも遅い 400 ms 程度であることが予想される．一方，アニーリングマシンをエッジ領域に配置した提案手法では，推論時間に対してサンプリング時間が支配的であるため，よりサンプリング時間が短い高性能なアニーリングマシンを用いることで推論時間を大幅に短縮できる．例えば，アニーリングマシンのサンプリング時間が 30 ms 程度まで高速化できれば，推論時間が 100 ms を下回る．以上の結果から，アニーリングマシンをデバイス近傍に配置した提案アーキテクチャにより，クラウド経由で D-Wave マシンを用いる場合と比べて通信遅延の影響を抑えることで，推論時のデバイスの応答性を向上させることが期待できる．

5. まとめ

本研究では，生成モデルの学習・推論に必要なサンプリング処理のアクセラレータとして，D-Wave マシンに比べ小型かつ低コストのアニーリングマシンをデバイスの近傍に配置することで，エッジ AI において生成モデルの学習をクラウドを介さずに効率化し，かつ推論時のデバイスの応答性を向上させるアーキテクチャを提案する．このアーキテクチャにより，デバイスは汎用的かつ低スペックでありながら，計算コストが高いサンプリングの処理をアニーリングマシンが担うことで，高い応用性を持つ生成モデルをデバイス上で活用することを可能にする．さらに，評価の結果から，アニーリングマシンを用いて生成モデルの一つである RBM の学習を効率化できること，およびアニーリングマシンをデバイスの近傍に配置することで，クラウド経由で D-Wave マシンを用いる場合と比べて，学習・推論時間の高速化が期待できることを示した．特に，アニーリングマシンのサンプリング時間が 30 ms 程度まで高速化できれば，推論時のデバイスの応答性を 100 ms 以下まで向上できることを示した．

今後の展望として，提案するアーキテクチャを採用した実用的なシステムの構築を通して，その有用性を評価していきたい．特に，デバイス上で数十 ms オーダーの応答性が要求されるような環境に対して，提案するアーキテクチャが実用に耐えうるかを評価したい．そのために，より処理能力が高いアニーリングマシンの採用や，アニーリン

グマシンとデバイス間の通信プロトコルの変更など，高い応答性を実現するためのシステムの検討を進めていく．

さらに，提案アーキテクチャのようなクラウドを介さないエッジ AI の仕組みは，本論文で示した通信遅延が小さいことのみでなく，データのプライバシー保護やネットワーク帯域の節約，運用コストの削減などの利点が考えられる．今後は，これらの利点においても，提案するアーキテクチャが有効性を示すことを評価していきたい．

参考文献

[1] A. R. Biswas and R. Giaffreda, “IoT and cloud convergence: Opportunities and challenges”, in *IEEE World Forum on Internet of Things (WF-IoT)*, Mar. 2014, pp. 375-376.

[2] Z. Zhou, X. Chen, E. Li, L. Zeng, K. Luo, and J. Zhang, “Edge Intelligence: Paving the Last Mile of Artificial Intelligence With Edge Computing”, *Proceedings of the IEEE*, vol. 107, no. 8, pp. 1738-1762, Jun. 2019.

[3] Z. Chen et al., “An empirical study of latency in an emerging class of edge computing applications for wearable cognitive assistance”, in *Proceedings of the Second ACM/IEEE Symposium on Edge Computing*, Oct. 2017, pp. 1-14.

[4] Google, “Edge TPU - Run Inference at the Edge”, https://cloud.google.com/edge-tpu.

[5] Y.Cheng, D. Wang, P. Zhou, and T. Zhang, “Model Compression and Acceleration for Deep Neural Networks: The Principles, Progress, and Challenges”, *IEEE Signal Processing Magazine*, vol. 35, no. 1, pp. 126-136, Jan. 2018.

[6] R. Salakhutdinov, “Learning Deep Generative Models”, *Annual Review of Statistics and Its Application*, vol. 2, no. 1, pp. 361-385, Apr. 2015.

[7] D. P. Kingma and M. Welling, “Auto-Encoding Variational Bayes”, arXiv:1312.6114 2013.

[8] J. Gui, Z. Sun, Y. Wen, D. Tao, and J. Ye, “A Review on Generative Adversarial Networks: Algorithms, Theory, and Applications”, arXiv:2001.06937 2020.

[9] T. Kadowaki and H. Nishimori, “Quantum annealing in the transverse Ising model”, *Phys. Rev. E*, vol. 58, no. 5, pp. 5355-5363, Nov. 1998.

[10] S. H. Adachi and M. P. Henderson, “Application of Quantum Annealing to Training of Deep Neural Networks”, arXiv:1510.06356 2015.

[11] D. Korenkevych, Y. Xue, Z. Bian, F. Chudak, W. G. Macready, J. Rolfe, and E. Andriyash, “Benchmarking Quantum Hardware for Training of Fully Visible Boltzmann Machines”, arXiv:1611.04528 2016.

[12] I. Grigorik, “Primer on Web Performance”, in *High Performance Browser Networking*, O’ Reilly Media, Inc., 2013, ch. 10.

[13] M. Yamaoka, T. Okuyama, M. Hayashi, C. Yoshimura, and T. Takemoto, “CMOS Annealing Machine: an In-memory Computing Accelerator to Process Combinatorial Optimization Problems”, in *IEEE Custom Integrated Circuits Conference (CICC)*, Apr. 2019, pp. 1-8.

[14] M. Aramon, G. Rosenberg, E. Valiante, T. Miyazawa, H. Tamura, and H. G. Katzgraber, “Physics-Inspired Optimization for Quadratic Unconstrained Problems Using a Digital Annealer”, *Front. Phys.*, vol. 7, no. 48, Apr. 2019.

[15] T. Okuyama, T. Sonobe, K. Kawarabayashi, and M. Yamaoka, “Binary optimization by momentum annealing”, *Phys. Rev. E*, vol. 100, no. 1, 012111, Jul. 2019.

[16] H. Goto, K. Tatsumura, and A. R. Dixon, “Combinatorial optimization by simulating adiabatic bifurcations in nonlinear Hamiltonian systems”, *Science Advances*, vol. 5, no. 4, eaav2372, Apr. 2019.

[17] G. Plastiras, M. Terzi, C. Kyrkou and T. Theocharidcs, “Edge Intelligence: Challenges and Opportunities of Near-Sensor Machine Learning Applications”, in *IEEE 29th International Conference on Application-specific Systems, Architectures and Processors (ASAP)*, Jul. 2018, pp. 1-7.

[18] D. H. Ackley, G. E. Hinton, and T. J. Sejnowski, “A learning algorithm for boltzmann machines”, *Cognitive Science*, vol. 9, no. 1, pp. 147-169, 1985.

[19] A. R. Mohamed, G. E. Dahl, and G. E. Hinton, “Acoustic Modeling Using Deep Belief Networks”, *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 1, pp. 14-12, Jan. 2011.

[20] G. W. Taylor, G. E. Hinton, and S. T. Roweis, “Modeling Human Motion Using Binary Latent Variables”, in *Advances in Neural Information Processing Systems 19*, 2006, pp. 1345-1352.

[21] U. Fiore, F. Palmieri, A. Castiglione, and A. D. Santis, “Network anomaly detection with the restricted Boltzmann machine”, *Neurocomputing*, vol. 122, pp. 13-23, Dec. 2013.

[22] D-Wave Systems, “Take the Leap”, https://www.dwavesys.com/take-leap.

[23] C. C. McGeoch and C. Wang, “Experimental evaluation of an adiabiatic quantum system for combinatorial optimization”, in *Proceedings of the ACM International Conference on Computing Frontiers*, May 2013, pp. 1-11.

[24] M. Benedetti, J. Realpe-Gómez, R. Biswas, and A. Perdomo-Ortiz, “Estimation of effective temperatures in quantum annealers for sampling applications: A case study with possible applications in deep learning”, *Phys. Rev. A*, vol. 94, no. 2, 022308, Aug. 2016.

[25] H. Tsuruta, “bmpy”, https://github.com/tsurubee/bmpy.

[26] E. Crosson and A. W. Harrow, “Simulated Quantum Annealing Can Be Exponentially Faster Than Classical Simulated Annealing”, in *IEEE 57th Annual Symposium on Foundations of Computer Science (FOCS)*, Oct. 2016, pp. 714-723.

[27] S. Morino, “Sqaod”, https://github.com/shinmorino/sqaod.

[28] Y. LeCun, C. Cortes, and C. J. C. Burges, “THE MNIST DATABASE of handwritten digits”, http://yann.lecun.com/exdb/mnist/.

[29] G. E. Hinton, “A Practical Guide to Training Restricted Boltzmann Machines”, in *Neural Networks: Tricks of the Trade*, vol. 7700, Springer, 2012, pp. 599-619.

[30] G. E. Hinton, “Training Products of Experts by Minimizing Contrastive Divergence”, *Neural Computation*, vol. 14, no. 8, pp. 1771-1800, Aug. 2002.

[31] T. Tieleman and G. Hinton, “Using Fast Weights to Improve Persistent Contrastive Divergence”, in *The Twenty-sixth International Conference on Machine Learning*, June 2009, pp. 1033-1040.

[32] E. Romero, F. Mazzanti, J. Delgado, and D. Buchaca, “Weighted contrastive divergence”, *Neural Networks*, vol. 114, pp. 147-156, June 2019.

[33] D-Wave Systems, “The D-Wave 2000Q System”, https://www.dwavesys.com/d-wave-two-system.