

リッチクライアント-エッジサーバ間での分散機械学習に関する一検討

高野紗輝 (指導教員：小口正人)

1 はじめに

近年 IoT デバイスが普及し、低遅延でネットワークの負荷分散が可能であるエッジコンピューティングに注目が集まっている。従来のエッジコンピューティングではエッジデバイス-エッジサーバ間において生データやそれに少し加工を施した容量の大きなデータの受け渡しが行われている。そのため、プライバシーの保護や通信コストの削減といった観点で問題が生じ、IoT の普及に対応するためにはさらなる工夫が必要である。

そこで本研究では、リッチクライアントの登場により、機械学習等の複雑な処理もエッジデバイス上で行うことが可能になったことと合わせ、リッチクライアントに適した分散機械学習モデルを検討する。

2 関連研究

2.1 エッジ/フォグコンピューティング

現在主流となっているクラウドコンピューティングは遅延が大きく、帯域も多く必要とするため、リアルタイムアプリケーションや大量のデータを送受信するアプリケーションの実装には向いていない。

一方でエッジコンピューティングは、物理的にエッジデバイスに近い位置で処理を行うことにより、低遅延であり、エッジ-クラウド間におけるネットワークの負荷を削減することも可能である [1]。

2.2 Federated learning

近年、高性能な CPU や GPU を搭載したリッチクライアントが登場し、IoT デバイス上でサーバが行っていた機械学習の処理を実行できるようになった。エッジデバイス上で機械学習を行うモデルとして Federated learning という分散型機械学習が提案された [2]。Federated learning はエッジコンピューティングとは異なり、プライバシーに配慮しながらエッジデバイスの情報をクラウドに集約し、クラウドが一括管理するモデルとして実装が進められている [3]。

3 提案モデル

従来のエッジコンピューティングモデルにおいて、リッチクライアントも学習に参加するモデルを図 1 に示すように提案する。

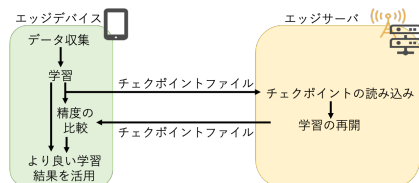


図 1: 提案モデル

従来のエッジコンピューティングではエッジサーバ上で行っていた機械学習等の高度なデータ処理を、エッジデバイスで実行する。その際、必要に応じてエッジサーバへの一部オフロードも可能であり、エッジデバイスで得られた学習結果をエッジサーバと共有することでさらに学習を進め、より良い学習結果をエッジデバイスで利用することができる。このような構成にす

ることで、エッジデバイスから外に出せないプライバシーに関わるデータも利用可能になり、エッジサーバとの通信コストも大幅に削減可能となる。

4 実験

4.1 実験環境

エッジサーバには、Intel Core i7-8700 の CPU、GeForce RTX 2080Ti の GPU、32Gbyte のメモリを搭載したサーバを用いる。また、エッジデバイスとして Quad-core ARM A57 @ 1.43 GHz の CPU、128-core Maxwell の GPU、4 GB 64-bit LPDDR4 25.6 GB/s のメモリを搭載した Jetson Nano を用いる。

4.2 予備実験

エッジサーバとエッジデバイス上で同一の機械学習処理を行った際の実行時間を比較した結果を図 2 に示す。

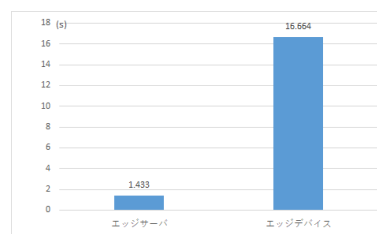


図 2: エッジサーバ、エッジデバイスによる実行時間

エッジサーバと比較してエッジデバイスはおよそ 11.6 倍の時間を要する。一方で、エッジサーバ、エッジデバイス共におよそ 96% と同等精度で学習を行うことが可能であった。ある程度低速ではあるもののエッジデバイス内のみでも十分学習できることが分かり、プライバシーが非常に重要なデータもそのような形で学習に用いる事が可能になる。

4.3 実験 1 (均質なデータによる学習)

4.3.1 実験概要

提案モデルにおいて、MNIST を使い、1 から 100 番目のデータをエッジデバイスに与え、それ以降の 101 から 60000 番目のデータをエッジサーバに与えた。エッジデバイス上で epoch 数を 2、各 epoch の step 数を 10 で学習を行い、エッジサーバでは、epoch 数を 2、各 epoch の step 数を 1000 で学習を再開させた。

4.3.2 実験結果

得られた学習精度を図 3 に示す。

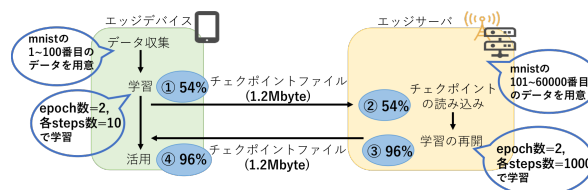


図 3: 学習精度

エッジデバイス上で 54% (①) まで学習でき、そこで得られたチェックポイントをエッジサーバへと渡し、エッジサーバ側で計測すると精度は同じく 54% (②) となっ

た。よって、エッジデバイスからエッジサーバへの学習処理のオフロードにより学習精度は落ちないことが読み取れる。この時点で得られた結果は十分な精度であるとは言えないが、エッジサーバで学習を再開させることでエッジサーバ上、エッジデバイス上共に96%(③、④)と高い精度の結果を得ることができた。

また、チェックポイントファイルの容量は共に1.2Mbyteと学習に使用したデータ数に寄らないため、多くのデータや容量の大きなデータを学習に使用する場合には通信量が削減できるという結果が得られた。

4.4 実験2 (偏りのあるデータによる学習)

4.4.1 実験概要

実験1の実験環境において、エッジデバイスとエッジサーバに与えるデータに偏りを持たせる。

4.4.2 実験結果

1. MNISTの"1"のデータのみをエッジデバイスに与え、エッジサーバには全ての数字のデータを与える。

図4に示す通り、エッジデバイスで得られる結果はエッジサーバに学習をオフロードしてから戻すことで多少精度が落ちるものの、依然として高い精度を得ることができた。

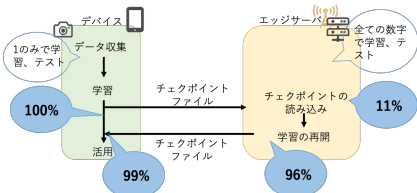


図4: エッジデバイスに"1"のみ、エッジサーバに全ての数字を与えた際の学習精度

2. MNISTの全ての数字のデータをエッジデバイスに与え、エッジサーバには"1"以外のデータを与える。

図5に示す通り、エッジサーバで学習した後の精度はエッジデバイスのみで学習する場合と比較すると向上が見られるものの、高い精度の結果を得ることはできなかった。

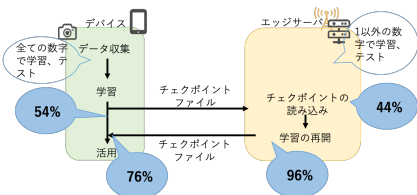


図5: エッジデバイスに全ての数字、エッジサーバに"1"以外の数字を与えた際の学習精度

3. MNISTの"1"のデータのみをエッジデバイスに与え、エッジサーバには"1"以外のデータを与える。

図6に示す通り、エッジデバイスとエッジサーバの持つデータが大きく異なる際には、エッジサーバに学習を引き継ぐことによりはじめにエッジデバイスで学習した結果が失われてしまい、学習精度が下がる結果となった。

以上の結果から、エッジデバイスにユーザー特有のデータが含まれている場合に学習をエッジサーバへと引き継ぐことで、逆に学習精度が下がってしまうという問題が生じる。提案モデルで示すように、エッジサーバで

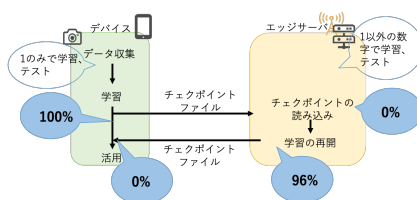


図6: エッジデバイスに1のみ、エッジサーバに"1"以外の数字を与えた際の学習精度

学習した結果とエッジデバイスのみで学習した結果を比較し、より良い学習結果を使用することでこの問題に対応する。例えば図3、図4のようにエッジサーバへ学習をオフロードして精度が上がった場合はその学習結果を用い、6のようにオフロードして精度が下がった場合は、その結果ではなくエッジデバイスで学習し続けた結果を用いるようにする。

5 提案モデルの検討

エッジデバイス上のみでの学習では精度が不十分な場合にエッジサーバへと学習を引き継ぐことでより精度の高い学習結果が得られ、本提案モデルには以下の利点があると言える。

- 元データの復元が難しい学習の重みのみをエッジデバイスの外へと持ち出すため、プライバシーの点で優れている。
- 必要に応じてエッジサーバへ情報を送るか送らないか判断して選別することができる形を作ることによって、機密性の高いデータを扱う場合や取得データを即時的に活用したい場合、エッジデバイスが一時的にネットワークに繋がっていない場合にも対応することが可能となる。
- エッジデバイス-エッジサーバ間はチェックポイントファイルのみを送受信するため、通信コストの削減が可能である。
- 従来エッジサーバが行っていた学習の一部をエッジデバイスにオフロードする形であると言え、エッジサーバの負荷分散が可能である。

6 まとめと今後の課題

従来のエッジコンピューティングで課題となっているプライバシーの保護や通信コストの削減を目的として、リッチクライアントに適した分散機械学習モデルの検討を行った。今後はスマートフォン上にあるような画像やSNSのテキスト情報を用いた実験を検討している。

参考文献

- [1] S. Yang. Iot stream processing and analytics in the fog. *IEEE Communications Magazine*, Vol. 55, No. 8, pp. 21–27, 2017.
- [2] Qiang Yang, Yang Liu, Tianjian Chen, and Yongxin Tong. Federated machine learning: Concept and applications. *ACM Trans. Intell. Syst. Technol.*, Vol. 10, No. 2, January 2019.
- [3] T. Yang, G. Andrew, Hubert Eichner, Haicheng Sun, W. Li, Nicholas Kong, D. Ramage, and F. Beaufays. Applied federated learning: Improving google keyboard query suggestions. *ArXiv*, Vol. abs/1812.02903, , 2018.