5ZC-04

完全準同型暗号における BNN を用いた高速な秘匿推論手法の実装と評価

橋詰 陽太 [*1] 古川 修平 [*2] 松本 直樹 [*3] 伴野 良太郎 [*4] 松岡 航太郎 [*5] 佐藤 高史 [*6]
京都大学 [*1] 放送大学 [*2] 京都大学 [*3] 京都大学 [*4] 京都大学 [*5] 京都大学 [*6]

1 序論

個人情報や医療情報について機械学習を利用したデータの活用を進めるうえで、情報漏洩に対する対策が大きな課題となっている。課題に対する一つの解決策は、データを暗号化した状態で計算処理を行うことが可能な完全準同型暗号（FHE）の利用である。GateNet [3] では、ビット演算に特化した FHE の一種である TFHE [1] を用いて 2 値化ニューラルネットワーク（BNN）[2] を動作させることにより秘匿推論を実現している。TFHE では、NAND などの論理ゲートと等価な処理を、暗号文に対して行うことができるため、BNN の推論処理を論理回路として実装し、各論理ゲートを対応する TFHE の演算に置き換えることで、暗号文に対して推論処理を適用することができる。しかし [3] では実装は与えられず、その推定動作速度は極めて低速であるとしている。

本研究では、TFHE 上での BNN を用いた秘匿推論手法において、FPGA 向けの最適化手法が TFHE 上の推論処理においても有効であり、高速な秘匿推論を実現できることを実験により確認した。我々は、TFHE 上でのパラメータが埋め込まれた BNN の実装に加えて、FPGA 等の一般的な論理回路向けの最適化手法として知られている (1) 3 値化による精度向上と高速化 [4] (2) Binary Adder Tree（BAT）による線形層の高速化 [3] (3) Shift-based Batch Normalization（SBN）の前計算による精度向上と高速化 [6] の 3 つを組み込み、性能を評価した。MNIST データセットを用いた性能評価では、最適化手法を適用することにより、精度を改善しつつ 4〜5 倍の高速化が達成できることを確認し、FPGA 等の一般的な論理回路向けの最適化手法が TFHE 上の処理においても有効であることを実験的に確認した。

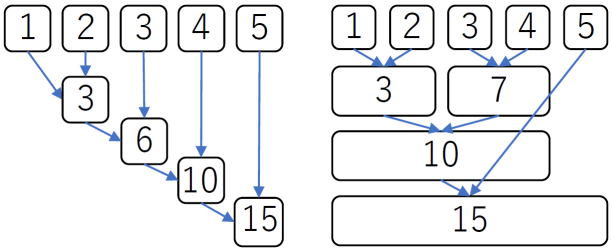

図 1 Binary Adder Tree（BAT）

2 適用する BNN 最適化手法

2.1 3 値化による精度向上と高速化

Ternary Neural Network（TNN）[4] は重みを 3 値とした BNN の変種である。BNN の全結合層（FC: Fully Connected layer）においては、非負の重みを 1、負の重みを -1 と 2 値化して計算を行う。一方 TNN の FC では、閾値を μ として、重みが μ 以上ならば 1、$-\mu$ 以下ならば -1、それ以外ならば 0 と 3 値化して計算を行う。BNN と比較して TNN では 3 値化により精度が改善し、また重みが 0 である場合は計算を行う必要がないため回路規模を削減できることが知られている [4]。

2.2 BAT による線形層の高速化

BNN（TNN）の FC における行列とベクトルの積の計算では、総和計算を必要とする。図 1 左に示すように、単純な総和計算では初項から順に足し上げるため、項数を n とすると加算器の段数は $O(n)$ となる。BAT では、図 1 右に示すように加算器を木構造で配置することで加算器の段数を $O(\log n)$ に抑え、より多くの加算器を並列に評価できるようにし高速化を図っている [3]。

2.3 SBN の前計算による精度向上

Batch Normalization（BN）は学習を高速化する等の効果があると知られている。BN では演算量が大きい乗算・除算が多く必要となるため、BNN（TNN）では乗算・除算をシフト演算を用いて近似した SBN を用いて高速化している [2]。しかし、SBN は掛かる係数を 2 のべき乗で近似しており、これが精度を下げる原因となっ

The Implementation and Evaluation of Privacy-Preserving Inference Using BNN over FHE
[*1] Yota HASHIZUME, Kyoto University
[*2] Shuhei FURUKAWA, The Open University of Japan
[*3] Naoki MATSUMOTO, Kyoto University
[*4] Ryotaro BANNO, Kyoto University
[*5] Kotaro MATSUOKA, Kyoto University
[*6] Takashi SATO, Kyoto University

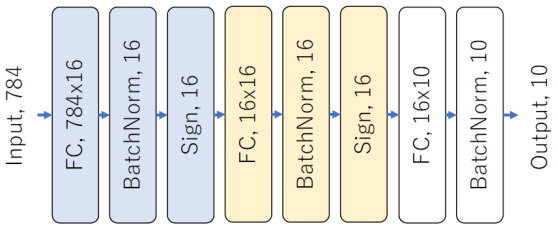

図 2　評価に用いたモデルの構成

表 1　MNIST データセットにおける各手法の精度

手法	精度（%）
BNN	84.71
（1） TNN	85.16
（2） BAT	BNN に同じ
（3） SBN の前計算	86.45
（1）＋（2）＋（3）	89.00

ている。最終層以外では SBN の出力が符号関数に入る以上、符号が変化する閾値のみわかれば十分である。そのため、閾値を予め計算しておくことで、計算精度を保つことができる [6]。

3　実験と評価

我々は各最適化手法を組み込んだ BNN の実装[*1]を行い、その精度やゲートの総数、TFHE 上で動作させたときの処理速度の評価実験を行った。TFHE 上での実装の評価には Iyokan [5] を用い、全論理ゲートの処理を GPU 上で行った。実験における計算機環境として、CPU には Intel Xeon Silver 4216（16C32T）を 2 基、GPU には NVIDIA A100 を 2 基使用した。RAM は 128GB、OS は Ubuntu 20.04.3 LTS であった。

表 1 に、MNIST データセットを用いて学習を行った後、テストデータで計測した精度の結果を示す。BNN を基準に、（1）TNN と（3）SBN の前計算、すべての手法を組み合わせた場合で精度が改善されることを確認した。なお BAT については総和回路の最適化であるため、計算処理自体は BNN と等価であり精度も BNN と同じとなる。

図 3 に各手法について合成した回路のゲート総数と Iyokan を用いて処理した際に要した処理時間の平均を示す。BNN の実装では総数で 80 万以上のゲートを評価する必要があったが、最適化手法をすべて組み込んだ

[*1] https://github.com/hzume/seccamp_nn

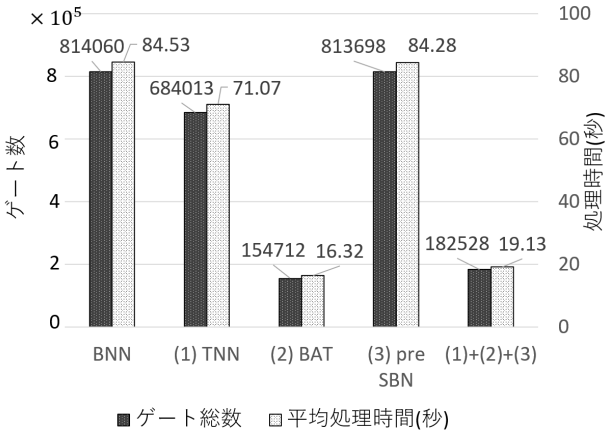

図 3　各手法のゲート総数と平均処理時間

状態では、約 18 万ゲートにとどまり、大幅に削減されていることが分かる。処理時間はゲートの削減に比例し、BNN の実装における平均処理時間は約 84 秒かかっていたが、最適化手法をすべて組み込んだ状態では約 19 秒と大幅に処理時間を短縮できていることが分かる。

これらの実験より、TFHE 上の BNN に対して FPGA 等の論理回路向けに用いられている最適化手法を適用することにより、精度を改善し、なおかつ処理時間を大幅に削減できることを確認した。

謝辞

本研究は、CREST, JPMJCR19K5 の支援を受けたものである。

本研究の一部は、情報処理推進機構とセキュリティ・キャンプ協議会によるセキュリティ・キャンプ全国大会 2021 オンライン L-III ゼミにおける成果に基づく。L トラックプロデューサーの竹迫良範氏をはじめとする関係者の方々に深謝を申し上げる。

参考文献

[1] Chillotti, I., Gama, N., Georgieva, M. and Izabachène, M.: TFHE: Fast Fully Homomorphic Encryption Over the Torus, *J. Cryptol.*, Vol. 33, No. 1, pp. 34–91 (2020).
[2] Courbariaux, M. and Bengio, Y.: BinaryNet: Training Deep Neural Networks with Weights and Activations Constrained to +1 or −1, *CoRR*, Vol. abs/1602.02830 (2016).
[3] Fu, C., Huang, H., Chen, X. and Zhao, J.: GateNet: Bridging the gap between Binarized Neural Network and FHE evaluation, *ICLR Workshop on Security and Safety in Machine Learning Systems* (2021).
[4] Li, F. and Liu, B.: Ternary Weight Networks, *CoRR*, Vol. abs/1605.04711 (2016).
[5] Matsuoka, K., Banno, R., Matsumoto, N., Sato, T. and Bian, S.: Virtual Secure Platform: A Five-Stage Pipeline Processor over TFHE, *USENIX Security Symposium* (Bailey, M. and Greenstadt, R., eds.), USENIX Association, pp. 4007–4024 (2021).
[6] Yonekawa, H. and Nakahara, H.: On-Chip Memory Based Binarized Convolutional Deep Neural Network Applying Batch Normalization Free Technique on an FPGA, *IEEE International Parallel and Distributed Processing Symposium Workshops, May 29 - June 2, 2017*, pp. 98–105 (2017).