一种基于全同态加密的神经网络模型

刘 傲（SY1806515）

摘 要：将隐私保护技术应用于深度学习一直是深度学习和隐私保护领域中一个重要的研究方向。本文旨在研究一种基于完全同态加密的神经网络，该神经网络可以对加密后的数据进行秘密的深度学习。本文提出了构建该新型神经网络的方法，包括同态加密算法和新型神经网络构造。通过在 MNIST 数据集中进行测试，本文中提出的基于全同态加密的神经网络，在保证用户隐私的情况下，其预测准确率仅比相同结构的传统神经网络低 0.33%。因此，本文中的神经网络模型可以在保护用户数据隐私的情况下确保深度学习结果的正确性。

关键词: 隐私保护；全同态加密；神经网络；深度学习

1. 引言

近年来，神经网络在计算机科学的各个领域内都很受人关注。神经网络是基于已有的训练数据，构建一个神经网络模型，然后使用该模型对未来的测试数据进行分析与预测。因此，它对于数据是有很强的依赖性的，构建一个神经网络模型往往需要非常庞大的数据集。但是在一些领域内，数据集是包含许多的个体隐私，如个体的生物信息或是个体的医疗信息等。这些数据对安全性和保密性有很高的要求，要直接基于这些原始数据集上构建神经网络模型是不被允许的。对于该问题，一种解决方案是首先对原始的数据进行加密，然后基于密文进行神经网络的构建。该方案所采用的加密算法需要满足：在密文上的运算等同于在明文上的运算。

同态加密是由 Rivest 等人在 1978 年提出[1]：允许用户对密文进行加法操作或乘法操作，它等同于对明文进行加法或乘法操作；若同时允许加法和乘法操作，则称之为全同态加密。在 2009 年，Gentry 提出了第一个全同态加密方案[2]。之后，Bos 等人在 2013 年提出一种基于环的全同态加密方案[3]，该方案一定程度上提高了加密效率。在 2016 年，Dowlin 等人基于此，提出一种加密神经网络[4]，该神经网络通过基于全同态加密后的数据进行训练，然后对该数据进行预测，最终获得了很好的准确率。虽然如此，但目前将全同态加密应用于神经网络中，仍有如下问题：由于全同态加密方案是基于整数运算，而神经网络是基于实数运算，因此需要进行实数的编码操作。这会带来两方面的问题：神经网络的准确性问题和神经网络很难训练的问题。

本文旨在设计一种基于全同态加密的神经网络，该神经网络可以对加密后的数据进行秘密的深度学习。 本文首先改进现有的一种完全同态加密算法，使其更加适合于该神经网络模型。然后给出基于该同态加密算法构建神经网络的方法。最后通过在 MNIST 数据集上进行实验，验证本文中所提出的基于全同态加密的神经网络的性能和预测准确率。

2. 基于全同态加密神经网络结构的设计

2.1 全同态加密方案

Angle[5]等人设计出了高效的整数向量的全同态加密方案，本文对其做了一些改进，使得改进后的同态加密方案更适合神经网络结构。首先定义一个等式，基于该等式实现加密和解密。对于明文向量 $X \in Z^n$，密文向量 $C \in Z^n$，秘钥 $S \in Z^{n\times n}$ 且 S 为一个可逆矩阵。

$$SC = mX + E \tag{1}$$

式中 m 为一个大整数，E 为一个随机噪声向量，并且满足向量 E 中的每一项 e 的值都远小于 m 。

基于上述等式可以定义加密过程(2)和解密过程(3)：

$$C = mS^{-1}X + S^{-1}E \tag{2}$$

$$X = \left[\frac{SC}{m}\right] \tag{3}$$

由于上式(1)是一个 Learning with Errors(LWE)问题，而求解 LWE 问题是困难的，因此该同态加密的方案是基于 LWE 安全的。

2.2 传统的神经网络

神经元是组成神经网络的单元，它是对输入数据进行线性运算，然后通过一个激活函数对之前结果进行非线性变换，最后输出结果。设 $x_1, x_2, \ldots, x_n$ 为神经元的输入，$w_1, w_2, \ldots w_n$ 和 b 为神经元的参数，则神经元的输出为：

$$y = f(\sum_{i=1}^{n} w_i x_i + b) \tag{4}$$

其中 $f : f(u) = \dfrac{1}{1+e^{-u}}$ 为神经元的激活函数。

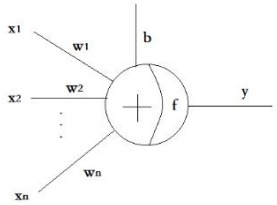

图 2.1 神经元的结构

神经网络是由大量的神经元相互联结而成，这些神经元分别对各自输入的数据进行计算，最终得到需要的结果。在原始结果已经确定的情况下，神经网络可以根据计算结果与真实结果的差异，调整各个神经元的参数，以此来获得一个队现实模拟的模型。这个过程是一种自适应的过程，它被称为学习过程或训练过程。

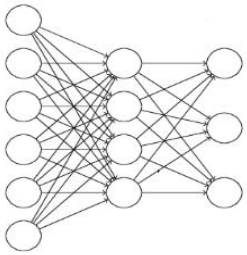

图 2.2神经网络结构

2.3 基于同态加密的神经网络

由于全同态加密算法只支持对密文的加法和乘法操作，因此需要对传统的神经网络中一些不满足加法和乘法的运算进行修改。

1)激活函数：

传统的激活函数为 sigmod 函数：$f: f(u)=\frac{1}{1+e^{-u}}$ 或 ReLU 函数：$f: f(u)=\max(u,0)$。这些函数不满足加法和乘法操作，由傅里叶变化可知我们可以采用多项式对任何函数进行线性逼近。因此，可以定义如下的激活函数：

$$g: g(u)=a+bu+cu^2+du^3 \qquad (5)$$

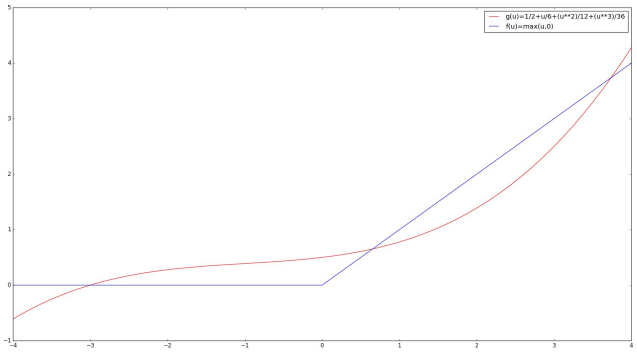

图 2.3 对 ReLU 的线性逼近

2) 损失函数：

在训练神经网络时，需要定义损失函数，通过极小化损失函数来求使得神经网络性能最好的参数。在基于全同态加密的神经网络中只能使用如下的损失函数或类似的多项式的损失函数。

设$t_1,t_2,\ldots,t_n$为真实值，而$y_1,y_2,\ldots,y_n$为神经网络的输出值，则损失函数为：

$$E=\frac{1}{2}\sum_{i=1}^{n}(t_i-y_i)^2 \qquad (6)$$

3)．神经网络的参数

以一个神经元的输入输出为例，在新定义下的激活函数下，神经元在密文下的输出为：

$$y_1=g(WC+b)$$

若在对应的明文下，将神经元的参数更改为$W'=WS^{-1}$，则可以在明文下得到类似的输出：

$$y_2=(W'X+b)$$

由于本文取的噪声E相对于mX非常小，因此两个神经元分别在明文下的输出y_2和在密文的下输出y_1近似相等。由此，通过上述密文下学习到的神经元转化为在明文下可以使用的神经元。通过这些神经

元从而构成明文下和密文下等价的神经网络。

3. 实验

整个实验步骤为：首先通过上述方式，构造一个基于全同态加密的神经网络模型和一个相同结构的传统神经网络。然后选取 MNIST 中的 42000 组数据作为训练集，使用同态加密算法先进行加密，接着在密文上训练本文提出的基于全同态加密的神经网络模型；而传统的神经网络使用原始数据进行训练。最后使用两种不同的神经网络对 28000 组测试数据进行预测。接下来，本文将从几个方面来说明实验中模型所表现出的结果。

1）加密算法的时间

在实验中选取的大整数 M 是大于 10^{10} 的一个随机数，秘钥 S 为 784×784 的随机可逆的方阵，噪声 E 为随机产生的向量，其中 E 中的每个元素小于 10^3 。通过该参数对 42000 组数据进行加密和解密的时间为表 1。

Table1：加密和解密过程的时间

	数据总量	时间（s）
加密过程	42000	13.01
解密过程	42000	149.37

由表可知，加密过程的运行时间很小，且远小于解密过程的运行时间，由于在该模型不需要解密过程，因此加密算法不会明显增加模型总的运行时间。

2）神经网络的训练时间和准确率

本文的期望是在短时间内可以通过训练获得高准确率的神经网络，因此训练时间和准确率是一个重要指标。在实验中分别使用相同结构的传统神经网络和基于全同态加密的神经网络进行学习，然后分别获取到不同的神经网络最终需要的训练时间和准确率。

Table2： 训练时间和准确率

	训练的数据量	训练时间(s)	测试集中的准确率(%)
基于全同态加密的神经网络	42000	570	87.64
传统的神经网络	42000	2860	87.97

从上述结果可知，在保证用户的隐私的情况下，本文中的神经网络模型训练时所需要的时间大约是传统神经网络模型所需要时间的 5 倍；在预测的准确率方面，本文的模型的准确率只比传统模型的准确率低大约 0.33%。

4. 总结与讨论

通过上述实验结果可以看出：本文所设计的基于全同态加密的新型神经网络在预测的准确率上与传统的神经网络的准确率是几乎相同的。并且本文所改进的全同态加密算法可以快速的实现对原始数据的加密。虽然解密算法很耗时，但在训练和预测阶段是无需使用解密算法，只需要通过本文中的方法，用加密算法的私钥对模型的参数做一次线性变换就可以得到一个在密文训练在明文预测的神经网络。

本文中的模型仍然需优化的方面是加密后所带来的训练时间的增加，即由于加密导致在等量数据集上，训练到同等准确率的情况下所增加的训练时间。该时间是由于全同态加密所导致的一些原始特征的弱化而引入的。一种提升训练速度的方式是使用 GPU 或特定的硬件进行加速，从而使得训练时间减少。另外本文所使用的是第一层是全联结的神经网络结构，若神经网络的第一层是卷积层或其他网络结构，该方法是否还可以有效工作，还需要之后进一步的实验和论证。

参考文献

[1] Rivest R L, Adleman L, Dertouzos M L. On data banks and privacy homomorphisms[J]. Foundations of Secure Computation, 1978:169-179.

[2] Gentry, Craig. Fully homomorphic encryption using ideal lattices[J]. Stoc, 2009, 9(4):169-178.

[3] Bos J W, Lauter K, Loftus J, et al. Improved Security for a Ring-Based Fully Homomorphic Encryption Scheme[C] IMA International Conference on Cryptography and Coding. Springer, Berlin, Heidelberg, 2013:45-64.

[4] Dowlin N,Gilad Bachrach R,Laine K,et al． Cryptonets：Applying neural networks to encrypted data with high throughput and accuracy[C]International Conference on Machine Learning(ICML)．2016：201-210．

[5] Brakerski Z, Vaikuntanathan V. Efficient Fully Homomorphic Encryption from (Standard) LWE[J]. foundations of computer science, 2011: 97-106.

[6] Angel Yu, Wai Lok Lai, James Payor. Efficient Integer Vector Homomorphic Encryption. 2015

[7] 周志华. 机器学习 : = Machine learning[M]. 清华大学出版社, 2016.