KU LEUVEN

FACULTY OF ENGINEERING TECHNOLOGY

DE NAYER (SINT-KATELIJNE-WAVER) CAMPUS

Object Detection on Encrypted Data

Using Secure Multiparty Computation

Felix LERNER

Promotor: Prof. dr. ir. Toon Goedemé
Co-promotor: Dr. Pradip Mainali (Onespan)

Masterproef ingediend tot het behalen van de graad van master of Science in de industriële wetenschappen: Industriele Wetenschappen Electronica-ICT

Academiejaar 2019 - 2020

Voorwoord

Het voorwoord vul je persoonlijk in met een appreciatie of dankbetuiging aan de mensen die je hebben bijgestaan tijdens het verwezenlijken van je masterproef en je hebben gesteund tijdens je studie.

Samenvatting

De (korte) samenvatting, toegankelijk voor een breed publiek, wordt in het Nederlands geschreven en bevat **maximum 3500 tekens**. Deze samenvatting moet ook verplicht opgeladen worden in KU Loket.

Abstract

We are affected with machine learning in many aspects of our daily lives, applications ranges from facial recognition to enhanced healthcare to self-driving cars. As companies outsource image classification tasks to cloud computing service providers, we see a rise in privacy concerns for both the users wishing to keep their data confidential, as for the company wishing to keep their classifier obfuscated.

Keywords: Computer vision, Cryptography, machine learning, secure multiparty computation, deep learning, object detetcion, privacy preserving, MLaaS, encryption

Contents

List of Figures

List of Tables

Lijst van symbolen

Maak een lijst van de gebruikte symbolen. Geef het symbool, naam en eenheid. Gebruik steeds SI-eenheden en gebruik de symbolen en namen zoals deze voorkomen in de hedendaagse literatuur en normen. De symbolen worden alfabetisch gerangschikt in opeenvolgende lijsten: kleine letters, hoofdletters, Griekse kleine letters, Griekse hoofdletters. Onderstaande tabel geeft het format dat kan ingevuld en uitgebreid worden. Wanneer het symbool een eerste maal in de tekst of in een formule wordt gebruikt, moet het symbool verklaard worden. Verwijder deze tekst wanneer je je thesis maakt.

b	Breedte	$[mm]$
A	Oppervlakte van de dwarsdoorsnede	$[mm^2]$
c	Lichtsnelheid	$[m/s]$

Lijst van afkortingen

MPC Secure Multiparty Computation
MLaaS Machine Learning as a Service
CNN Convolutional Neural Network
ReLU Rectified Linear Unit

1 Introduction

Deep learning based object detection on images is a hot topic for researchers and interest in machine learning is steadily growing among miscellaneous businesses.
Secure multiparty computation (MPC) is a subfield of cryptography, making it possible for a party to run an algortihm on confidential data, that is supposed to stay unknown even to the party running the algorithm.
Both concepts aren't new concepts, but with the rise of big data and processing power, there has been an increase in research into these fields.
In this thesis we present the applicability of MPC for deep learning based object detection.

1.1 Problem

The use of third party MLaaS (Machine Learning as a Service) providers or any cloud computing solution, as processing power for an image classification task, raises privacy concerns as sensitive images of users need to be sent to servers running an instance of the neural network. It's important to note that the transport of the image from the client to the server is secure, since the parties can make use of reliable HTTPS (Hypertext Transfer Protocol Secure) connections. The user's images, however, are stored in plaintext on the server, aswell as the computed output of the image. Furthermore the whole design of the neural network including all trained parameters needs to be stored on the servers of the third party, for the image classifier to function. Both of these remote storage solutions require a considerably amount of trust in the third party. Since the third party could potentially exploit the user data for commercial purposes or even steal the intellectual property of the image classifier. In this thesis we try to tackle the need to trust a third party MLaaS provider.

We want it to compute an encrypted image on an obfuscated neural network to ouput a correct encrypted result.

1.2 Hypothesis

How can we securely compute deep learning based object detection algorithms for privacy preserving image classification?
With the use of MPC protocols we can implement methods such that we can compute a whole neural network in the secure field. We predict a drastically decrease in perfomance and will try to find performance optimizations along the way of implementing a proof of concept.

2

Literature study

2.1 Convolutional neural network

Convolution neural networks (CNN) is a special type of neural network used for images. The spatial properties of the pixels in the image are used during the evaluation of the input, meaning the neighbouring pixels of a central pixel impact the output to the next layer of that central pixel while pixels further away do not. CNNs are made of multiple layers. Typically, as you move further from the input layer to the output layer the dimensionality reduces, we can say the input gets mapped on a desired output manifold. Inputs that we classify as similar are supposed to be in the same region in the output manifold.

2.1.1 Convolution layer

In this layer a discrete convolution of a kernel K shifting over an image I is performed, as shown in equation 2.1. The kernel has parameters also called weights, so that certain features get extracted from the input.

$$(I * K)[m,n] = \sum_{j}\sum_{k} I[m-j, n-k]K[j,k] \tag{2.1}$$

The output of this layer is a convolved feature map.

2.1.2 Activation function

Since these convolutions are simple lineair operations and most image classification tasks require non-lineair classifiers, non-lineairity needs to be added to the neural network. This is achieved through adding a non-linear activation function after a convolution or fully connected layer. The most popular activation function is the rectified linear unit (ReLU) as seen in equation 2.2.

$$f(x) = max(0,1) \tag{2.2}$$

2.1.3 Pooling layer

The dimensionality reduction we talked about in the beginning of this chapter happens in the pooling layer. A kernel is shifted over the image. In the case of max pooling the kernel selects the maximum value of the portion of the image it covers, to create the new dimensionality reduced image.

2.1.4 Fully connected layer

The fully connected layer is a multilayer perceptron that discriminates different object classes and identifies identical ones. All elements in vector h_{i-1}^{out} have their own bias B_i and weight W_i so that h_i^{in} can be calculated for each layer i according to equation 2.3.

$$h_i^{in} = h_{i-1}^{out} \cdot W_i + B_i \tag{2.3}$$

2.2 Secure multiparty computation

Secure multiparty coputation is a protocol that is used between n number of parties P. Each of these parties has private data also called a secret S. With MPC it is possible for these parties to compute a public[1] function f on the secrets. Such that a party $p_i \in P$ only knows his secret $s_i \in S$ and the public securely computed output $f(s_0, s_2, s_{n-1})$ after the protocol has succesfully finished. A classic application is Yao's Millionaires' problem Yao (1982) in wich two millionaires wish to know who is richer, there is catch however. Instead of making their balances publicly known. They wish to keep their balances a secret. In this case the number of parties n is 2 the secrets s_0 and s_1 are their balances. The public function $f(s_0, s_1) = 1$ if $s_0 < s_1$ and 0 otherwise.

We categorize 2 types of parties based on their willingness to deviate from the correct predefined protocol.

- Honest but curious parties (passive): Parties wish to know other parties secrets but will not deviate from the protocol at any time.
- Malicious parties (active): Parties wish to know other parties secrets and wish to change output of computation to favourable result. Parties will deviate from the protocol to cheat and change the outcome at any time.

If the two millionaires are honest but curious parties, they will not deviate from the protocol and they will computer the correct output as a result they will know who is the richer millionair but they won't

[1] public or global means known to all parties, while private or local means known only by the corresponding party

know how much money the other one has. In the other case one of the two millionaires is corrupt and acts maliciously, the honest millionair will follow protocol while the dishonest millionair will deviate from the protocol to change the result in his favour. In the event that the dishonest millionair is poorer he will change the outcome thus appearing richer. From now on, we assume the parties are honest but curious parties, unless specified otherwise. We also assume the communication between the different parties to be secure.

2.2.1 Secret sharing

In order to do secure computing, the parties need to split their secret into secret shares. A secret sharing method can be used by the secret holder to split a secret into a number of shares. Combining these shares will reveal the secret, while individual shares alone will leak nothing about the secret. In a (t,n) threshold secret sharing sheme parties must combine atleast t shares of the total n shares, to obtain the secret. We can now set a threshold t high enough, denying the secret to small curious parties and allowing to reveal the secret when a majority ($\geq t$) consensus is reached. Shamir's secret sharing scheme Shamir (1979) is based on polynomial interpolation and the essential idea is that it takes atleast t points in order to define a polynomial $p(x)$ of degree $t-1$. Given a set of t points in a 2-dimensional carthesian system $(x_1,y_1),(x_2,y_2),...,(x_t,y_t)$, there exists only one polynomial of degree $t-1$. This can be proven and the mathematical construction of a polynomial $p(x)$ of degree $t-1$ based on a set of t points can be calculated using Lagrange's interpolation formula 2.4.

$$p(x) = \sum_{i=1}^{t} y_i \delta_i(x) \quad \text{with} \quad \delta_i(x) = \prod_{1 \leq j < t; i \neq j} \frac{x - x_i}{x_j - x_i} \tag{2.4}$$

With this in mind, a secret dealer can now share his secret s to n parties by choosing a random $t-1$ degree polynomial $p(x) = a_0 + a_1 x + ... + a_{t-1} x^{t-1}$ in which a_0 is the secret or the number representation of the secret if the secret is not a number. The dealer now calculates n points on the polynomial starting from $x = 1$, because the secret is located at $x = 0$. Each party $p_i \in p_1, p_2, ..., p_n$ is given a different single point (x_i, y_i), at this stage the secret is shared. To recombine the secret, the parties simply broadcast or send their shares to a central entity, if more than t shares are known, it suffices to calculate the Lagrange polynomial $p(x)$ and $s = p(0)$. In the case of not having enough shares, the Lagrange polynomial becomes impossible to calculate since every polynomial is equally likely, thus revealing absolutely nothing about the secret.

2.2.2 Operations

2.2.2.1 Artithmetic operators

2.2.2.2 Relational operators

2.2.3 Share recombination

2.3 Conclusion

3 Implementation

3.1 Specifications

3.2 Design

3.3 Conclusion

4
Evaluation

4.1 Results

4.1.1 Reliability results

4.1.2 Timing results

4.2 Discussion

4.3 Conclusion

Er zijn twee manieren om formules in LaTeX in te voeren:

- Inline: $a^2 + b^2 = c^2$ (`$a^2+b^2 = c^2$`)
- In een equation omgeving (`\begin{equation} a^2+b^2 = c^2 \end{equation}`):

$$a^2 + b^2 = c^2 \tag{4.1}$$

Griekse letters geef je in d.m.b. het backslash commando. Bijvoorbeeld de letter sigma σ verkrijg je door `$\sigma$` inline in te geven. Dit is analoog voor griekse letters in de equation omgeving. Een beknopte lijst van symbolen vind je op de Wikibooks pagina voor LaTeX (link). Alle andere nuttige informatie omtrent het gebruik van LaTeX voor formules vind je hier ook terug.

5
Conclusion

Voor het verwijzen naar informatiebronnen wordt gebruik gemaakt van het numerisch systeem of van het auteur-jaar systeem. Dit kies je door volgend commando in het latex bronbestand aan te passen:

- numerisch (IEEE) : `\bibliographystyle{ieee}`
- alfabetisch (APA) : `\bibliographystyle{apalike}`

Plaats je bronnen in een *bibtex* bestand (evt. via software zoals bv. Jabref Endnote of Mendeley), waarnaar je verwijst vanuit je thesis text a.d.h.v. het commando `\cite`. Enkele links naar nuttige software in deze context:

- JabRef (Open Source)
- Mendeley (Freeware)
- EndNote (Paid license)

Indien je zelf een .bibtex bestand wil aanleggen dien je volgende syntax te volgen voor een tijdschriftartikel:

```
@article{hughes2005,
title={Isogeometric analysis: CAD, finite elements, NURBS, exact geometry
and mesh refinement},
author={Hughes, Thomas JR and Cottrell, John A and Bazilevs, Yuri},
journal={Computer methods in applied mechanics and engineering},
volume={194},
number={39},
pages={4135--4195},
year={2005},
publisher={Elsevier}
}
```

Enkele voorbeelden van het gebruik van bronnen in een tekst (in APA stijl):

Recent werd het Higgs boson experimenteel vastgesteld door Aad et al. Aad et al. (2012) (syntax: `\cite{aad2012}`).

Als alternatief voor het discretiseren van een CAD model vooraleer een eindige elementenanalyse te kunnen toepassen, stellen Hughes et al. voor om de nodige elementenformulering rechtstreeks uit de NURBS beschrijving van de CAD geometrie te halen Hughes et al. (2005) (syntax: `\cite{hughes2005}`). Daarnaast introduceren ze tevens een k-iteratieve procedure als een verfijning van de geldende p- en h-iteratieve procedures in eindige elementen methoden Cottrell et al. (2009) (syntax: `\cite{cottrell2009}`).

Bibliography

Aad, G., Abajyan, T., Abbott, B., Abdallah, J., Khalek, S. A., Abdelalim, A., Abdinov, O., Aben, R., Abi, B., Abolins, M., et al. (2012). Observation of a new particle in the search for the standard model higgs boson with the atlas detector at the lhc. *Physics Letters B*, 716(1):1–29.

Cottrell, J. A., Hughes, T. J., and Bazilevs, Y. (2009). *Isogeometric analysis: toward integration of CAD and FEA*. John Wiley & Sons.

Hughes, T. J., Cottrell, J. A., and Bazilevs, Y. (2005). Isogeometric analysis: Cad, finite elements, nurbs, exact geometry and mesh refinement. *Computer methods in applied mechanics and engineering*, 194(39):4135–4195.

Shamir, A. (1979). How to share a secret. *Communications of the ACM*, 22(11):612–613.

Yao, A. C. (1982). Protocols for secure computations. In *23rd annual symposium on foundations of computer science (sfcs 1982)*, pages 160–164. IEEE.

A

Uitleg over de appendices

Bijlagen worden bij voorkeur enkel elektronisch ter beschikking gesteld. Indien essentieel kunnen in overleg met de promotor bijlagen in de scriptie opgenomen worden of als apart boekdeel voorzien worden.

Er wordt wel steeds een lijst met vermelding van alle bijlagen opgenomen in de scriptie. Bijlagen worden genummerd het een drukletter A, B, C,...

Voorbeelden van bijlagen:

Bijlage A: Detailtekeningen van de proefopstelling

Bijlage B: Meetgegevens (op USB)

FACULTY OF ENGINEERING TECHNOLOGY
DE NAYER (SINT-KATELIJNE-WAVER) CAMPUS
Jan De Nayerlaan 5
2860 SINT-KATELIJNE-WAVER, België
tel. + 32 16 30 10 30
fet.denayer@kuleuven.be
www.fet.kuleuven.be