UNIVERSITY OF AMSTERDAM

PROJECT PROPOSAL

The Quest to Finding the Best Secure Neural Network Inference

November 21, 2022

Supervisors:
Zoltan Mann

Student:
Jorit Prins
12862789

Course:
Afstudeerproject bachelor Informatica

Course code:
5062ABI18Y

1 Relevance

The recent rise in Big Data increased the data exchange on the internet. With more and more computer resources available, researchers quickly started to utilise the possibilities of machine learning (ML) to analyse the data. Techniques like neural networks (NN) are promising ways to scientific breakthroughs. Machine learning has a wide variety of applications for classification such as traffic analysis, image recognition, intrusion detection, spam detection, medical or genomics predictions, financial predictions and face recognition (Dowlin et al. 2017; Gilad-Bachrach et al. 2016; He et al. 2015; Islam et al. 2011)

The use of ML typically consists of two phases: training and inference. In the first phase a NN is trained by feeding an extensive dataset to find the best parameters. NNs used for machine learning have to be maintained, evaluated and the training phase is often a tedious and time exhausting process. In the inference phase an input is applied to the trained NN. Because of the time consuming process of creating a NN, machine learning as a service (MLaaS) became popular. In MLaaS, a company offers a pre-trained NN to the clients. Now, clients only need to worry about the inference phase.

A typical MLaaS situation consist of two parties: the client holding an input x and a company holding neural network f. For this research we will focus on the inference part. The client wants to know the neural network applied to the input, *f(x)*, while keeping the sensitive contents of x and the result *f(x)* private from the company. The company wants to hold the intellectual property f private while still giving the opportunity to the client to use f.

However, MLaaS offers great threats to privacy. To train the model as accurately as possible a NN needs access to a large amount of precise data from clients, which may consist of sensitive information. Thus, clients may be reluctant to provide the NNs with their data. Other features, irrelevant to the prediction task, could also be derived from this data (Nasr, Shokri, and Houmansadr 2019). On the inference phase, input from the client to the NN can also be confidential. On the other hand, owners of a NN could be worried that an adversary could steal (parameters of) their (often costly) NN. Furthermore, the result of the NN could also be confidential resulting in the need to retain this information from unauthorized parties. The secure neural network inference (SNNI) problem entails calculating the applied input *f(x)* while still holding all the above security requirements.

No general implementation of an SNNI has been widely accepted to the authors knowledge. Rapid progress in this area has made it hard to get a good overview of technological advances.

Mann et al. (2022) has summarized several proposed approaches for SNNI. However, these approaches are often proof-of-concept and are not thoroughly tested. Moreover, the performance is often only tested on basic measures like efficiency or accuracy.

Other metrics like energy consumption, that could be of relevance, are not researched. This could be of importance because of limitations on the client side. For example when a device is battery powered or in the case of IoT devices that have limited power resources and where the overall energy consumption should be low. Companies, on the other hand, also want to keep energy consumption as low as possible because of budget limitations and thus should not encounter big energy overhead. Another reason to limit the energy consumption is the desire to reduce carbon emission on the fight against climate change.

2 Research question(s)

To contribute to the prior research in this area, I will discuss the energy implications of a suggested, open source implementation of an approach to SNNI. A few of these implementations are ABY2.0 (Patra et al. 2020), Chameleon (Riazi et al. 2018), Cheetah (Huang et al. 2022), CrypTFlow2 (Kumar et al. 2019) and Delphi (Mishra et al. 2020). The main research question is of this project is:

RQ: What are the energy implications of open source suggestions of SNNIs?

To help research the implications of the SNNI, I have defined a subset of research questions:

RQa: How do we best measure the energy consumption of a SNNI?

Once we have established a way to measure energy consumption of SNNIs we can start with the experiments of measuring the energy consumption. With these results we know want to see if there is a difference between the energy consumption on the client side of the implementation and the energy consumption on the server side. If there is a difference we shortly want to look at the implications of this difference. This implications could for example have impact on the aforementioned IoT devices, thus resulting in research question *RQb*:

RQb: If there is a difference between energy consumption on the client side and on the server side: what are the implications of the overhead on the client side and what are implications the overhead the server side?

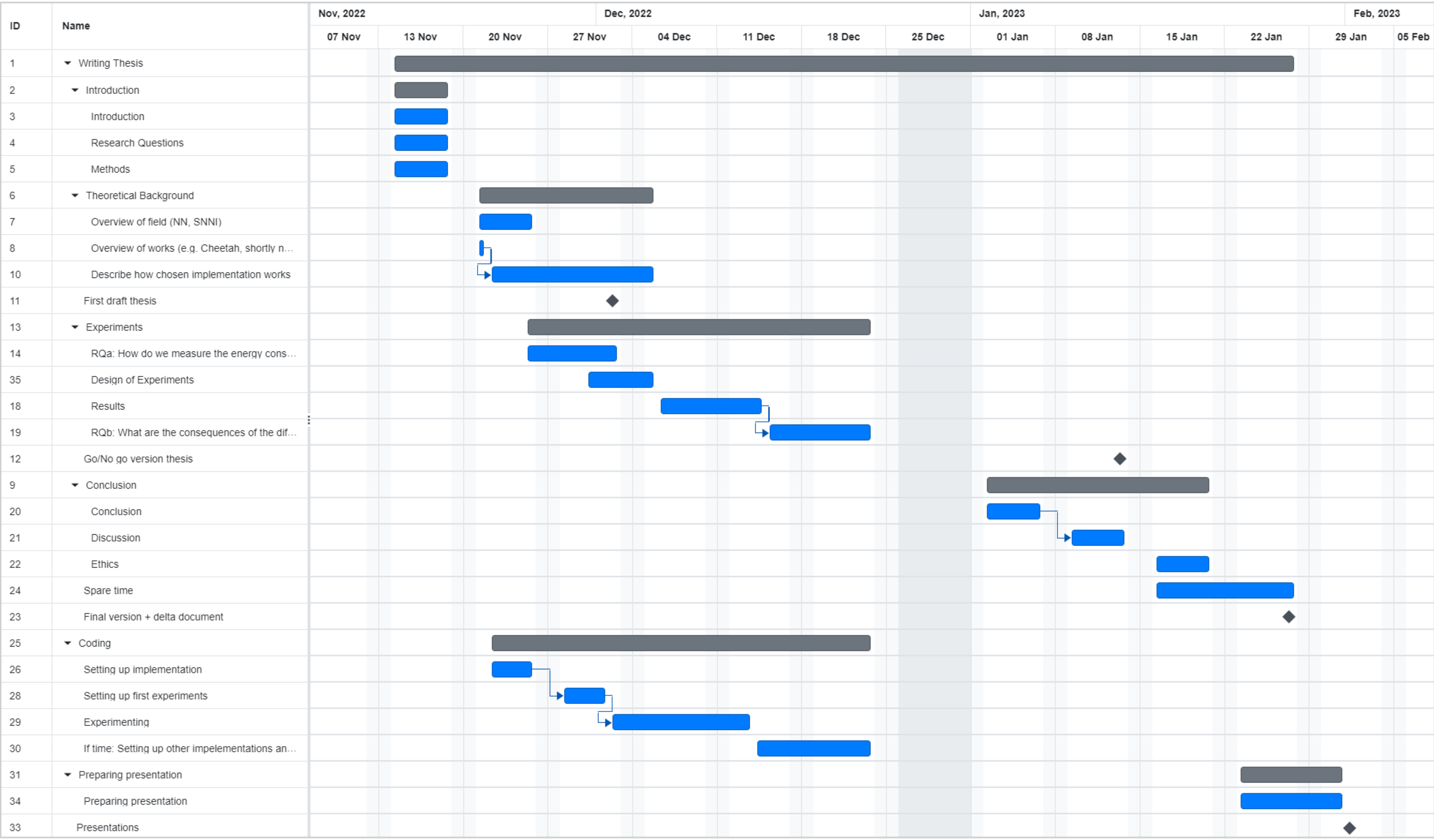
3 Method

To answer these aforementioned questions first I will have to select one implementation of a SNNI to start with. Once the implementation is chosen I will need to get it working on my own setup. Simultaneous to this we will answer question *RQa* with a literature search on how other authors measure energy consumption. There is a possibility that other authors have already researched the energy consumption of an approach to SNNI mentioned in the preceding section, or researched the energy implications of other programs.

Once the implementation is set up and we have answered *RQa* we can start testing the energy consumption of a NN with and without the SNNI. Now we can calculate the overhead of the SNNI. If there is time to set up another implementation I will compare the overhead between these implementations. With the results of this experiment we can answer the first part of *RQb*. The second part of *RQb* can be answered with a small literature search on the implications of energy consumption.

4 Schedule

(On next page)


ID
Name
Nov, 2022
Dec, 2022
Jan, 2023
Feb, 2023
07 Nov
13 Nov
20 Nov
27 Nov
04 Dec
11 Dec
18 Dec
25 Dec
01 Jan
08 Jan
15 Jan
22 Jan
29 Jan
05 Feb
1 Writing Thesis
2 Introduction
3 Introduction
4 Research Questions
5 Methods
6 Theoretical Background
7 Overview of field (NN, SNN)
8 Overview of works (e.g. Cheetah, shortly n…
10 Describe how chosen implementation works
11 First draft thesis
13 Experiments
14 RQa: How do we measure the energy cons…
35 Design of Experiments
18 Results
19 RQb: What are the consequences of the dif…
12 Go/No go version thesis
9 Conclusion
20 Conclusion
21 Discussion
22 Ethics
24 Spare time
23 Final version + delta document
25 Coding
26 Setting up implementation
28 Setting up first experiments
29 Experimenting
30 If time: Setting up other impelementations an…
31 Preparing presentation
34 Preparing presentation
33 Presentations

References

Dowlin, Nathan, Ran Gilad-Bachrach, Kim Laine, Kristin Lauter, Michael Naehrig, and John Wernsing (2017). "Manual for Using Homomorphic Encryption for Bioinformatics". In: *Proceedings of the IEEE*, pp. 1–16. DOI: 10.1109/jproc.2016.2622218. URL: https://doi.org/10.1109/jproc.2016.2622218.

Gilad-Bachrach, Ran, Nathan Dowlin, Kim Laine, Kristin Lauter, Michael Naehrig, and John Wernsing (2016). "CryptoNets: Applying Neural Networks to Encrypted Data with High Throughput and Accuracy". In: *Proceedings of The 33rd International Conference on Machine Learning*. Ed. by Maria Florina Balcan and Kilian Q. Weinberger. Vol. 48. Proceedings of Machine Learning Research. New York, New York, USA: PMLR, pp. 201–210.

He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun (2015). *Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification*. DOI: 10.48550/ARXIV.1502.01852. URL: https://arxiv.org/abs/1502.01852.

Huang, Zhicong, Wen-jie Lu, Cheng Hong, and Jiansheng Ding (Aug. 2022). "Cheetah: Lean and Fast Secure Two-Party Deep Neural Network Inference". In: *31st USENIX Security Symposium (USENIX Security 22)*. Boston, MA: USENIX Association, pp. 809–826. ISBN: 978-1-939133-31-1. URL: https://www.usenix.org/conference/usenixsecurity22/presentation/huang-zhicong.

Islam, Naveed, William Puech, Khizar Hayat, and Robert Brouzet (Sept. 2011). "Application of homomorphism to secure image sharing". In: *Optics Communications* 284.19, pp. 4412–4429. DOI: 10.1016/j.optcom.2011.05.079.

Kumar, Nishant, Mayank Rathee, Nishanth Chandran, Divya Gupta, Aseem Rastogi, and Rahul Sharma (2019). *CrypTFlow: Secure TensorFlow Inference*. DOI: 10.48550/ARXIV.1909.07814. URL: https://arxiv.org/abs/1909.07814.

Mann, Zoltán Ádám, Christian Weinert, Daphnee Chabal, and Joppe W. Bos (2022). *Towards Practical Secure Neural Network Inference: The Journey So Far and the Road Ahead*. Cryptology ePrint Archive, Paper 2022/1483. https://eprint.iacr.org/2022/1483. URL: https://eprint.iacr.org/2022/1483.

Mishra, Pratyush, Ryan Lehmkuhl, Akshayaram Srinivasan, Wenting Zheng, and Raluca Ada Popa (Aug. 2020). "Delphi: A Cryptographic Inference Service for Neural Networks". In: *29th USENIX Security Symposium (USENIX Security 20)*. USENIX Association, pp. 2505–2522. ISBN: 978-1-939133-17-5. URL: https://www.usenix.org/conference/usenixsecurity20/presentation/mishra.

Nasr, Milad, Reza Shokri, and Amir Houmansadr (May 2019). "Comprehensive Privacy Analysis of Deep Learning: Passive and Active White-box Inference Attacks against Centralized and Federated Learning". In: *2019 IEEE Symposium on Security and Privacy (SP)*. IEEE. DOI: 10.1109/sp.2019.00065.

Patra, Arpita, Thomas Schneider, Ajith Suresh, and Hossein Yalame (2020). *ABY2.0: Improved Mixed-Protocol Secure Two-Party Computation*. Cryptology ePrint Archive, Paper 2020/1225. https://eprint.iacr.org/2020/1225. URL: https://eprint.iacr.org/2020/1225.

Riazi, M. Sadegh, Christian Weinert, Oleksandr Tkachenko, Ebrahim M. Songhori, Thomas Schneider, and Farinaz Koushanfar (2018). *Chameleon: A Hybrid Secure Computation Framework for Machine Learning Applications*. DOI: 10.48550/ARXIV.1801.03239.