Efficient Privacy-preserving Neural Network Inference for Heart Arrythmia Detection

P. Chervet, A.-M. Olteanu, J. R. Troncoso-Pastoriza [•], D. Froelicher [••], J. Van Zaen, R. Delgado-Gonzalo, J.-P. Hubaux [••]

The raise of AI and machine learning as a service (MLaaS) poses a risk to the privacy of those using it. In today's data-driven application landscape, it is common that a party needs to process sensitive (personal) data using third party resources (such as computation, storage, or communication infrastructure), which constitutes a risk with respect to the privacy of such data. CSEM is working on performing neural network (NN) inference without revealing user input data to other parties involved and while hiding the model parameters from the user. Relying on homomorphic encryption and secure two-party computation, we present here a service as a client-server application for privacy preserving NN inference.

With the increasingly extensive use of machine learning in domains like facial recognition, credit card risk assessment, or medical diagnosis, the privacy threat for the people being analyzed or monitored is growing. Our work aims at preventing such threats by protecting the users' privacy, while still allowing useful analyses to be done.

We consider the case of a neural network (NN) used to detect heart arrhythmia from electrocardiogram (ECG) data. Considering the sensitivity of such data, our goal is to perform this task without revealing user input data to other parties involved and while hiding the model parameters from the user. Thus, we aim to protect the privacy of the users of the service and, at the same time, partially the confidentiality of the NN model (parameters are only visible to whoever runs the model). We use homomorphic encryption (HE) and secure two-party computation techniques, and implement the service as a client-server application for privacy-preserving NN inference.

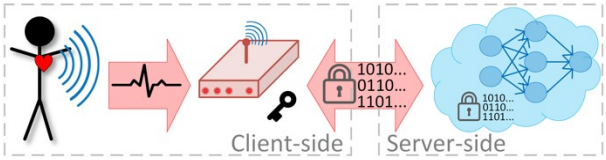

Figure 1: Health monitoring ecosystem composed of a client side, which is under the control of the user, and a server side which offers cloud computing capabilities (e.g., Amazon Web Services, Google Cloud, Microsoft Azure).

In our use case, the system is composed by the triplet Sensor-Gateway-Cloud, where the sensor collects the ECG data and sends it to the gateway. On the gateway, the client side of the application encrypts the data and, together with the server-side application on a cloud, runs the privacy-preserving NN.

As the basis of our work, we used a convolutional NN named DeepCardio [1] developed at CSEM. It takes single-lead ECG segments of 30 seconds duration, cut into 25 equally sized windows, and classifies them to detect and classify arrhythmias. The NN mainly consists of 6 convolutional layers with non-linear activation functions (ReLU and max pooling). At each layer, the number of channels is doubled, and the signal length is halved.

To implement a privacy-preserving version, we extended the approach described in GAZELLE [2] for the linear parts (convolution) we use HE and for the non-linear parts (activation functions) secure two-party computation, specifically garbled circuits (GC).

To evaluate our approach, we run the client side on a laptop and the server side on a powerful workstation. Figure 2 shows the average execution time per layer for one window. The time spent for the HE operations is dominated by actual computation time. The exponential growth is due to the doubling of the channels. The time spent for GC operations is dominated by data transmission time, similar for all layers because the number of GC evaluations stays constant.

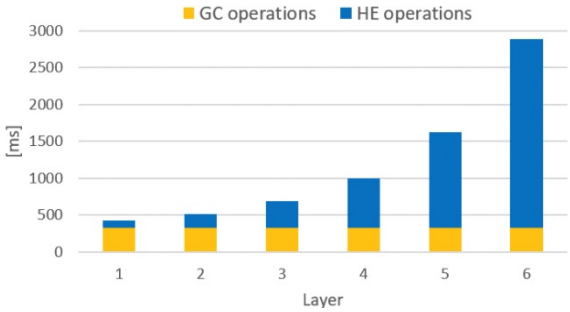

Figure 2: Execution time and distribution (between GC and HE operations) per layer for one window, averaged over 25 windows (one classification).

The results show an achieved latency of ~7 seconds for one window, resulting in ~175 seconds for one classification. However, we were able to identify the important bottlenecks and we propose approaches to improve it. The heavy computations performed using HE are mainly done on the server side; further parallelization of the server code is thus a good starting point. By optimizing the GC implementation and its parameters, an important gain in transmission time should be possible as well.

Our work shows that performing NN inference in a privacy-preserving way is possible and that there is promising potential to improve its current performance, thus, limiting the lekage of private data to computing centers and cloud service providers.

• Laboratory for Communications and Applications 1 (LCA1), Ecole polytechnique fédérale de Lausanne (EPFL), Switzerland

•• Laboratory for Data Security (LDS), Ecole polytechnique fédérale de Lausanne (EPFL), Switzerland

[1] J. Van Zaen, *et al.*, "Classification of cardiac arrhythmias from single lead ECG with a convolutional recurrent neural network", BIOSTEC 2019.

[2] C. Juvekar, *et al.*, "Gazelle: A low latency framework for secure neural network inference," USENIX Security 2018.