Delving in the loss landscape to embed robust watermarks into neural networks

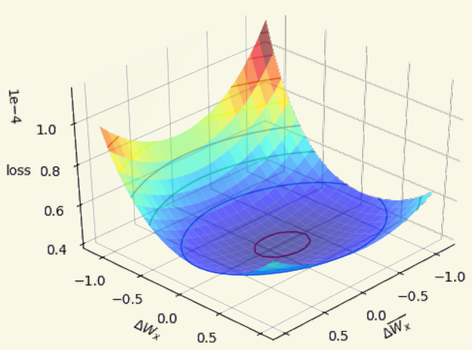
Enzo Tartaglione, Marco Grangetto, Davide Cavagnino, Marco Botta
{name.surname}@unito.it
Università degli Studi di Torino, Italy

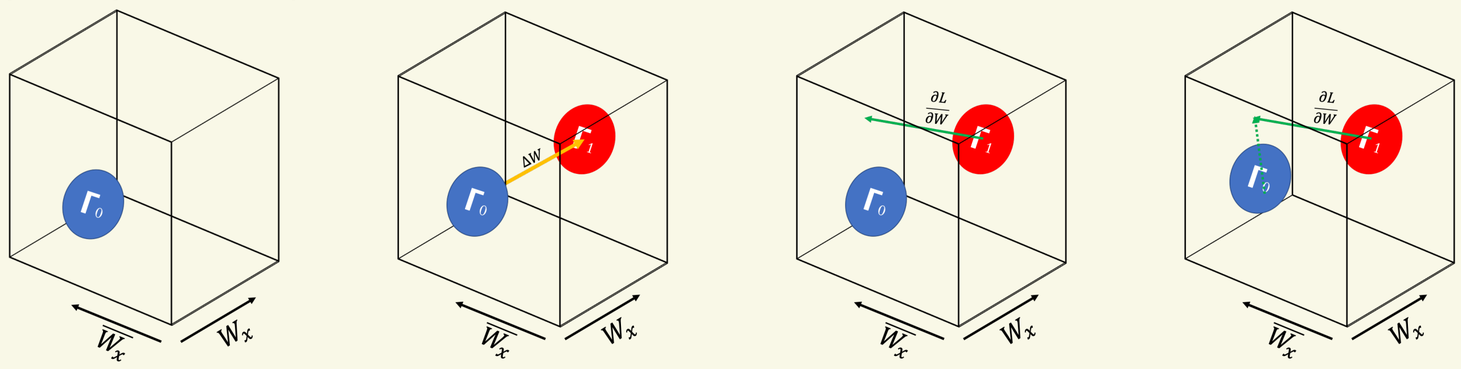
Abstract

In the last decade the use of artificial neural networks (ANNs) in many fields like image processing or speech recognition has become a common practice because of their effectiveness to solve complex tasks. However, in such a rush, very little attention has been paid to security aspects. In this work we explore the possibility to embed a watermark into the ANN parameters. We exploit model redundancy and adaptation capacity to lock a subset of its parameters to carry the watermark sequence. The watermark can be extracted in a simple way to claim copyright on models but can be very easily attacked with model fine-tuning. To tackle this culprit we devise a novel watermark aware training strategy. We aim at delving into the loss landscape to find an optimal configuration of the parameters such that we are robust to fine-tuning attacks towards the watermarked parameters. Our experimental results on classical ANN models trained on well-known MNIST and CIFAR-10 datasets show that the proposed approach makes the embedded watermark robust to fine-tuning and compression attacks.

How to delve in the loss landscape?

- We start from a *naive* and simple solution (Γ_0) that embeds the watermark in the ANN model weights;
- we embed the watermark in *all* the layers of the deep model, including the output layer;
- we minimize the loss on Γ_0 while maximizing it on R other models, having the same non-watermarked parameters $\overline{W}_x$ but adding a noise ΔW to the watermarked parameters W_x;
- this way, we guarantee the watermarked weights lay into a very steep valley, providing robust solutions to attacks without impacting the ANN performance.

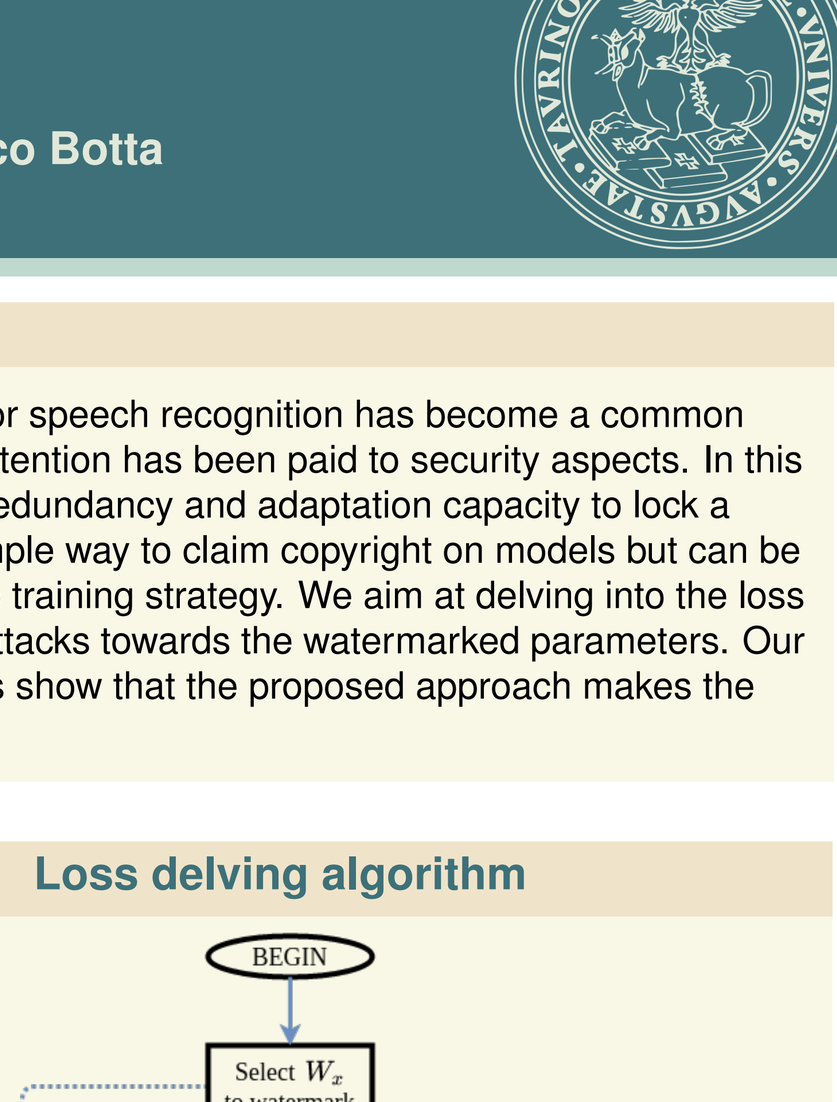




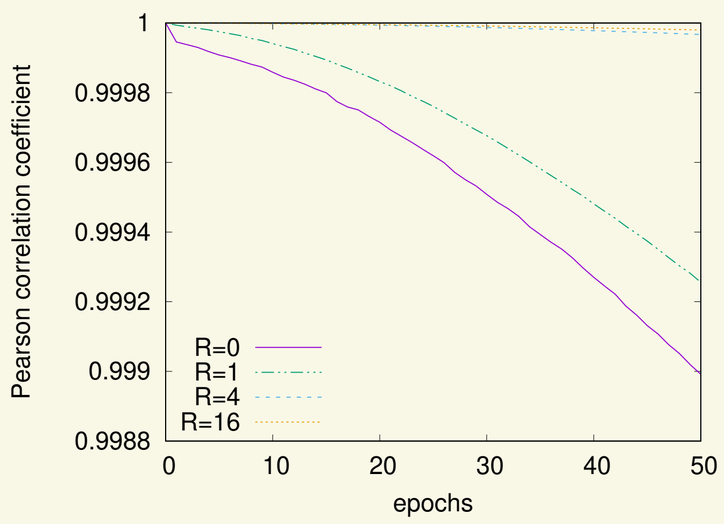
(a) (b) (c) (d)

From the model which is trained updating non-watermarked parameters (a) we add some noise on the watermarked parameters only in R replicas of our original network Γ_0 – in this case Γ_1 (b). Then, the gradient on Γ_1 is computed (c) and projected to the non-watermarked parameters space (d).

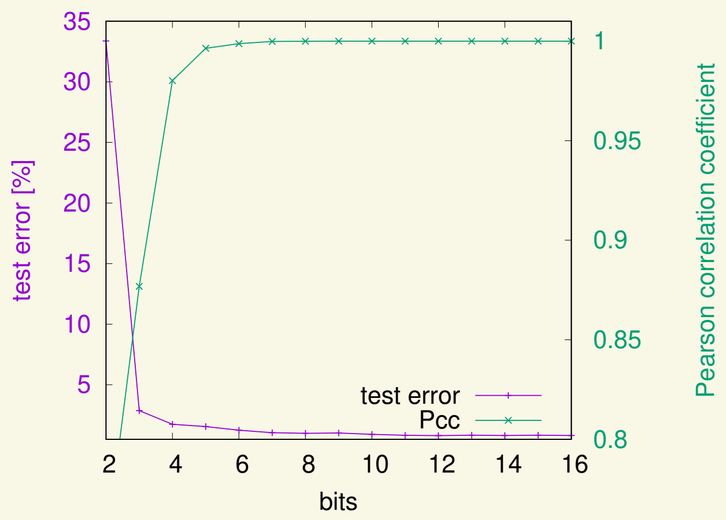
Loss delving algorithm



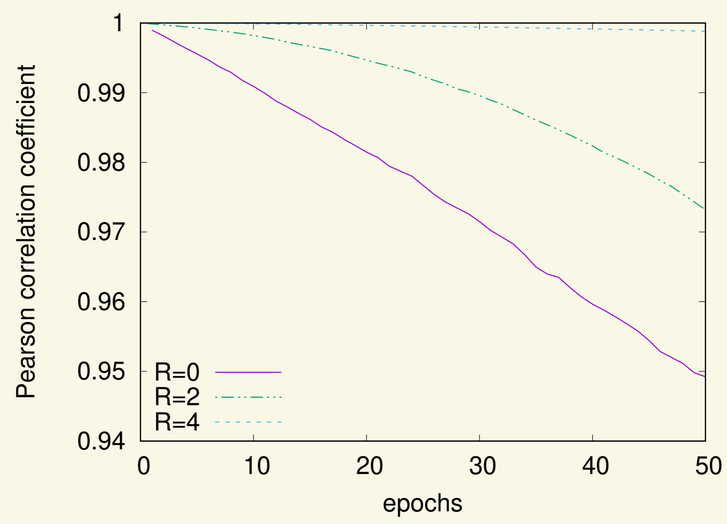
Results



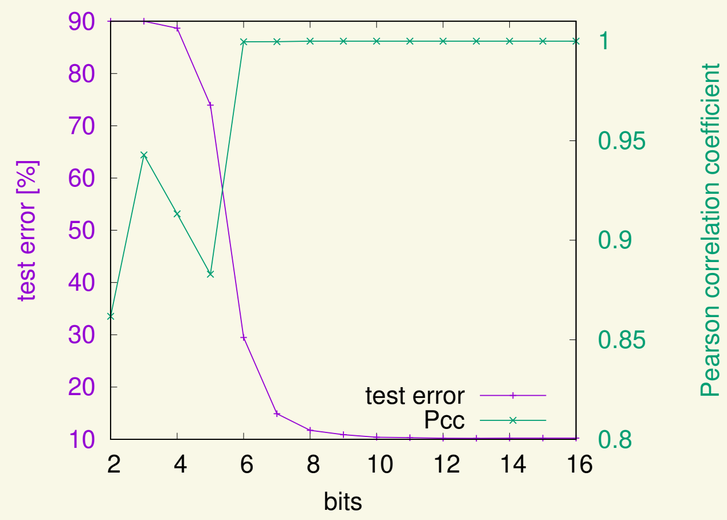
(e)



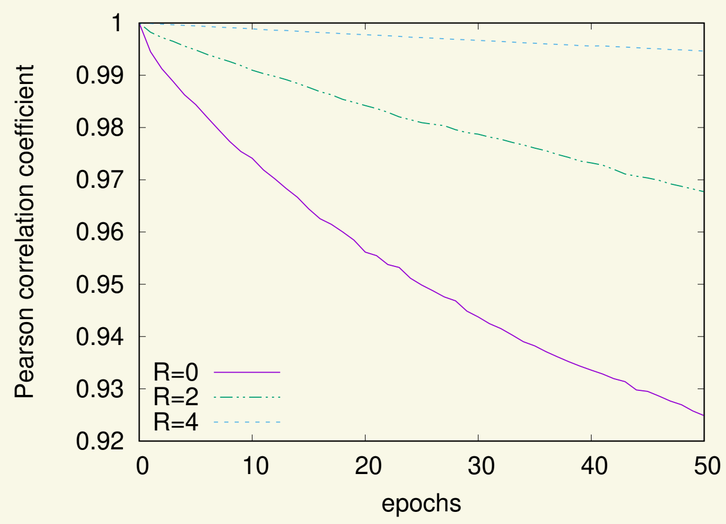
(f)



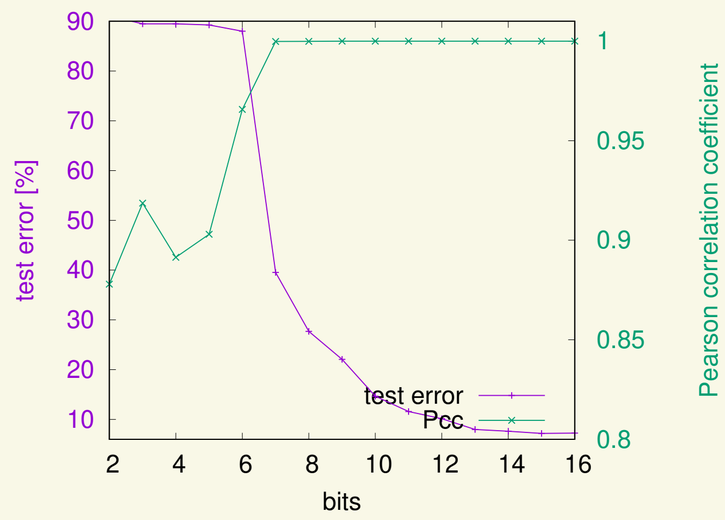
(g)



(h)



(i)



(j)

Fine-tuning attack on LeNet5-caffe (e) trained on MNIST, ALL-CNN-C (g) and ResNet-32 (i) trained on CIFAR-10.
Quantization attack on LeNet5-caffe (f), ALL-CNN-C (h) and ResNet-32 (j).