Implementation of Machine Learning Algorithms on Ultra-Low-Power Hardware for In-Sensor Inference

Edge computing has become increasingly popular for Internet of Things (IoT) applications powered by Machine Learning (ML) inference.

Moving the ML models directly to the data source can improve privacy, lower the latency, and yield higher energy efficiency, without requiring a constant internet connection as for a cloud-centric approach.

Nonetheless, ML models are known to be memory- and energy-hungry, requiring a significant amount of resources, not available on ultra-low-power edge devices, such as sensors, often based on Microcontrollers (MCUs).

As a consequence, an increasing research effort has been put into making ML more efficient, trading off limited accuracy for large savings in terms of energy or memory.

This research branch has taken the name of Edge AI and it is the focus of this thesis.

In particular, this dissertation focuses on optimizing two popular models for edge AI: tree ensembles and deep neural networks (DNNs).

Tree ensembles reach high accuracy with a limited memory and energy footprint, making them an ideal choice to deploy on resource-constrained hardware.

Nonetheless, accurate ensembles often feature many trees, rapidly growing in memory and inference latency.

In the first chapter of this work, I focus on how these models can be further optimized, reducing the memory footprint thanks to an efficient implementation and other approaches such as quantization.

Moreover, thanks to a dynamic inference approach, I show a way to reduce the inference latency with little to no accuracy drops.

All approaches detailed in this work concerning the optimization of tree ensembles have been collected and included in an open-source Python library.

The second chapter of this thesis focuses on deep learning (DL).

DL models often reach state-of-the-art accuracy, coming however at the cost of a high number of parameters to be stored and computations to be performed.

Therefore, I introduce a flow to obtain memory-inexpensive yet accurate DNNs that leverage sub-byte quantization and mixed precision.

Then, I introduce three dynamic inference approaches to lower the average energy and latency per inference of DNNs.

The first slices the network by its width, running only a subset of the channels and neurons depending on the input complexity.

The second leverages the different complexity of the classes in a dataset, running an easy and inexpensive model to recognize the simplest classes while leveraging larger DNNs only for the other classes.

The last one introduces an enhanced early-stopping mechanism tailored for datasets with class frequency imbalance, a common occurrence in edge ML, leading to higher accuracy and increased energy savings w.r.t. other approaches.

In conclusion, the contributions of this work are twofold.

A novel deployment flow for tree ensembles is introduced, focusing on optimizations both at compile and run time.

Then, multiple optimizations for efficient DNN deployments are proposed, both in terms of compile-time and run-time approaches, allowing the deployment of small yet accurate models even on the most constrained edge devices.