# On-device Learning for Secure Internet of Things

Shuai Zhu
*RISE Research Institutes of Sweden*
Stockholm, Sweden
shuai.zhu@ri.se

Fatemeh Rahimian
*RISE Research Institutes of Sweden*
Stockholm, Sweden
fatemeh.rahimian@ri.se

Thiemo Voigt
*Uppsala University*
Uppsala, Sweden
*RISE Research Institutes of Sweden*
Stockholm, Sweden
thiemo.voigt@angstrom.uu.se

***Abstract*—The recent breakthroughs in machine learning (ML) and deep learning (DL) have catalyzed the design and development of various intelligent systems over wide application domains. While most existing machine learning models require large memory and computing power, ML/DL applications on edge devices have been extensively studied in recent years. Most early systems exploit the inference capabilities of ML and DL models that are already trained on data captured from different mobile and embedded sensing components for specific application goals, such as classification and segmentation. More recently, on-device learning (ODL) has gained attention, which refers to using resource-constrained devices, such as mobile phones and embedded systems, for ML/DL model training. The reason is that ODL makes ML-based systems smarter and more robust, for example, ODL can help to cope with data drift problems by tuning deployed models according to deployed environments on the fly. However, most state-of-the-art ODL systems are designed for less resource-constrained devices, such as smartphones. This research project aims to enable ODL on devices with extremely limited hardware resources, for example, low-power IoT devices. A potential use case is to apply ODL to IoT security, for instance, jamming attack detection and classification for low-power wireless networks.**

***Index Terms*—On-device Machine Learning, Internet of Things, Jamming attack detection**

## I. Introduction

Advances in Machine Learning (ML) - including Deep Learning (DL) - have resulted in tremendous improvements in solving various types of problems in computer vision, natural language processing, machine translation, etc., and across different applications domains such as biology and healthcare, automotive industries, smart cities and many more. Training ML models often requires large cloud servers with high-throughput accelerators like GPUs. However, such resources are not always available, especially for edge devices. Recently, some efforts have been made to deploy and train ML models on less resource-rich edge devices, such as smartphones, but similar work for IoT devices with much tighter constraints is still lagging behind. We aim to push this boundary by deploying and training ML models in resource-constrained devices or sensor networks with limited memory, battery, and connectivity. This will open up many possibilities in the near future and can even address some of the existing challenges. For example, privacy-preserving computation is one of the key obstacles that on-device learning can directly address. Our work will also enable learning in the absence of an Internet connection and will eliminate the need to upload data to the cloud and/or download an updated model back, and that in itself saves bandwidth and reduces latency and energy. The latter is important for low-power IoT devices, where communication is typically far more expensive than computing. When dealing with time-critical applications, it is also important to react quickly without communicating with a server or over a network. Furthermore, devices can become smarter as they adapt to the changes in the deployment environment or customize models for end-users. For example, a medical device can, over time, learn to provide personalized diagnoses or services that fit the specific conditions of an individual patient.

Thanks for the financial support from the Swedish Foundation for Strategic Research.

## II. Goals

The goal of our project is to make resource-constrained IoT devices smarter by developing system support for on-device learning, which includes the initial training based on data available offline, the compression and transfer of the initial model to a deployed IoT network and the continued on-device training for adaptation to the environment. To achieve this goal, we need to:

- Develop machine learning models and system support for learning on resource-constrained IoT devices.
- Devise methods for decentralized learning over a network of resource-constrained devices.
- Implement, evaluate, and demonstrate a complete end-to-end system in a real application scenario related to IoT security.

## III. Background and Related Work

Largely, on-device ML consists of on-device inference and training operations. On-device inference refers to deploying pre-trained ML models on devices and then running inference locally. Researchers have presented extensive works using such an "inference-embedded" system architecture [1]–[3]. Note that, due to computational power limitations, cloud-based (or server-based training) is the mainstream approach, where local data is shared with the server, and the training operations are executed at remote platforms. Later, updated models can be

re-distributed to local IoT devices to perform local inference operations. However, since being able to train an ML model locally can preserve the precious network bandwidth and limited battery budgets and, at the same time, contain the raw data locally to preserve privacy. Therefore, researchers have recently started to propose schemes to train ML models on-device despite their computational resource limitations [4]–[6]. Still, these works are in their early stages, and only a few of them target highly constrained IoT devices. On the other hand, Dhar et al. [7] and Zhu et al. [8] give surveys for ODL from the algorithmic perspective and the systematic view, respectively.

## IV. Methodology

Training models from scratch is too resource-intensive and energy-consuming for edge devices, even with state-of-the-art optimization methods [4]. However, it is unnecessary to train models from scratch as the ML community has presented numerous NNs, datasets, and pre-trained models. Depending on the problem at hand, system researchers can select a suitable NN and train a model in the cloud on a large and generic dataset, then deploy the model to devices and fine-tune the model on a new and small dataset. This workflow is more efficient and practical than deploying a randomly initialized model to the device and training from scratch. As shown in Figure 1, the workflow consists of two steps. Firstly, we can pre-train the model on the server/cloud and then compress it to fit edge devices with suitable techniques, such as quantization and pruning. Secondly, we deploy the model on target devices and perform both training and inference locally.

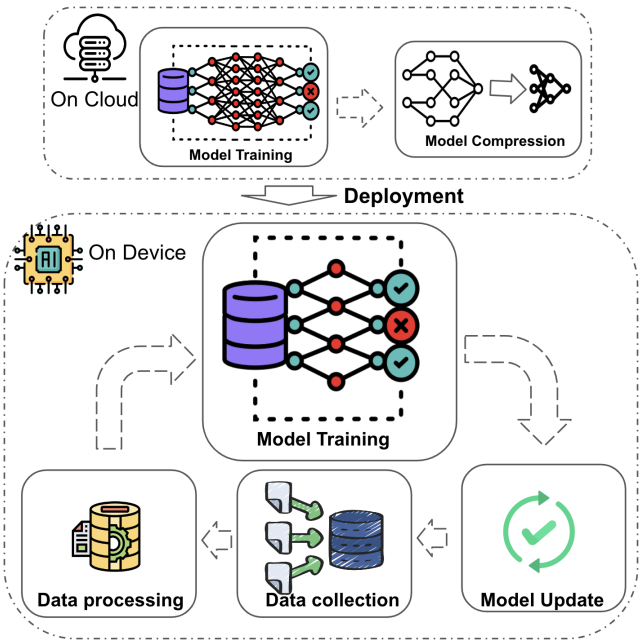


Fig. 1. A generic workflow of on-device training

## V. Use case

One potential use case is to extend JamSense [9], a system that is able to detect multiple sources of interference and classify jamming attacks for low-power wireless networks. The parameters of its K-means algorithm are predetermined and hard-coded based on data collected from experiments where nodes are in fixed positions. With ODL, the new system will be more robust to changes in the environment or application scenarios.

## VI. Conclusion

This project aims to make resource-constrained IoT devices smarter by enabling ML model training on devices. The smarter device would be able to keep learning locally to save bandwidth and reduce latency, as well as reduce energy consumption and preserve privacy. On the other hand, the model drift problem [10] is one of the main reasons that lead to the accuracy reduction of deployed pre-trained models ML models. ODL can cope with this problem by retraining these models to adapt to the environment changes.

## References


[1] Zhang, Q., Li, X., Che, X., Ma, X., Zhou, A., Xu, M., ... and Liu, X. (2022, April). A comprehensive benchmark of deep learning libraries on mobile devices. In Proceedings of the ACM Web Conference 2022 (pp. 3298-3307).

[2] Xu, M., Zhu, M., Liu, Y., Lin, F. X., and Liu, X. (2018, October). Deepcache: Principled cache for mobile deep vision. In Proceedings of the 24th annual international conference on mobile computing and networking (pp. 129-144).

[3] Fang, B., Zeng, X., and Zhang, M. (2018, October). Nestdnn: Resource-aware multi-tenant on-device deep learning for continuous mobile vision. In Proceedings of the 24th Annual International Conference on Mobile Computing and Networking (pp. 115-127).

[4] Xu, D., Xu, M., Wang, Q., Wang, S., Ma, Y., Huang, K., ... and Liu, X. (2022, October). Mandheling: Mixed-precision on-device dnn training with dsp offloading. In Proceedings of the 28th Annual International Conference on Mobile Computing And Networking (pp. 214-227).

[5] Profentzas, C., Almgren, M., and Landsiedel, O. (2022, September). MiniLearn: On-Device Learning for Low-Power IoT Devices. In International Conference on Embedded Wireless Systems and Networks.

[6] Wang, Q., Xu, M., Jin, C., Dong, X., Yuan, J., Jin, X., ... and Liu, X. (2022, June). Melon: Breaking the memory wall for resource-efficient on-device machine learning. In Proceedings of the 20th Annual International Conference on Mobile Systems, Applications and Services (pp. 450-463).

[7] Dhar, S., Guo, J., Liu, J., Tripathi, S., Kurup, U., and Shah, M. (2021). A survey of on-device machine learning: An algorithms and learning theory perspective. ACM Transactions on Internet of Things, 2(3), 1-49.

[8] Zhu, S., Voigt, T., Ko, J., and Rahimian, F. (2022). On-device Training: A First Overview on Existing Systems. arXiv preprint arXiv:2212.00824.

[9] Kanwar, J., Finne, N., Tsiftes, N., Eriksson, J., Voigt, T., He, Z., ... and Saguna, S. (2021, October). Jamsense: Interference and jamming classification for low-power wireless networks. In 2021 13th IFIP Wireless and Mobile Networking Conference (WMNC) (pp. 9-16). IEEE.

[10] Tsymbal, A. (2004). The problem of concept drift: definitions and related work. Computer Science Department, Trinity College Dublin, 106(2), 58.