Resource Efficient Federated Learning for Deep Anomaly Detection in Industrial IoT applications

Alexandros Gkillas
Industrial Systems Institute,
Athena Research Center, Patras Science Park, Greece
Dept. of Computer Engineering & Informatics,
University of Patras

Aris Lalos
Industrial Systems Institute,
Athena Research Center, Patras Science Park, Greece

***Abstract*—Anomaly data constitute a thorny problem in numerous industrial applications. In recent years, deep learning enabled anomaly detection has emerged as a critical direction, however the improved detection accuracy and reconstruction is achieved with the utilization of large neural networks with more layers and nodes, increasing their storage and computational cost. Moreover, the data collected in edge devices contain user privacy, introducing challenges that can be successfully addressed by the very recent on-device privacy-preserving distributed machine learning paradigm, known as federated learning (FL). This paradigm allows edge devices to locally train and exchange models increasing also the communication cost. Thus, in order to deal with the increased communication, processing and storage challenges introduced by FL based deep anomaly detection NN pruning is expected to have significant benefits towards both reducing the processing, storage and communication complexity but also towards avoiding the over fitting problem. With this focus, a novel compression-based optimization problem is proposed at the server-side of a FL paradigm that simultaneously fusses the local models broadcast by the edge devices and performs pruning generating a much more compressed model to be deployed at the edge devices with no accuracy loss even if the training is performed using a small subset of data instances, achieving compression rates greater than 99%.**

Keywords*—anomaly detection, compression, federated learning, multidimensional time series

I. Introduction

In recent years, the Internet of Things (IoT) has revolutionized the industry discipline, paving the way for better efficiency, safety, and security in the manufacturing processes [1]. Industry 4.0 emerged recently, as an efficient paradigm to handle the need of the inter-connectivity of Industrial IoT (IIoT), enabling the access to real time datasets derived from dispersed edge devices, which can sense the environment and process data in an autonomous and decentralized manner [1]. Nonetheless, the deployment of IoT devices in the Industrial domain introduces some crucial challenges. To be more specific, the quality of the derived multidimensional data is often degraded by various factors e.g., faulty sensors and communication failures, thus introducing various types of anomalies (e.g. data instances that significantly deviate from the majority of data instances such as missing values and/or outliers) [2] and affecting heavily the performance of various IIoT tasks, such as classification [3], prediction [4]. To address the problem of anomaly detection and restoration numerous centralized solutions has been developed over the years. The considered problem has been studied under various scenarios and settings. Especially, utilizing the advances in deep learning, data-driven models including RNN [5], GAN [6] and CNN [7] have been getting attention achieving state-of-the-art results in the considered problem. Nonetheless, the above deep learning models require massive amounts of training data and significant computing and storage resources, thus rendering them unsuitable for IIoT edge devices. More importantly, in several cases the available data produced from a single industrial site may be insufficient for learning accurate machine/deep learning models to address efficiently Anomaly detection, a.k.a. outlier detection or novelty detection [8], [9]. An intuitive solution to tackle the above issue is to gather data collected from different parties and/or design at the same time more compact NN models. However, due to privacy constraints [1], the industrial entities may be reluctant to expose their owned dataset, while still being interested in an AI model trained using their own and similar privately owned datasets of others.

Towards this direction, Federated learning (FL) , as a secure distributed framework is capable of addressing the above challenges [10], enabling the clients to collaboratively learn a machine/deep learning model without sharing their own private datasets [11], [12]. The privacy-preserving and distributed nature of the FL promotes such solutions at the network edge, where each IIoT device can be considered as a client containing one or more sensors that measure different physical quantities, thus recording a part of the generated multivariate data instances [13]. Both the FL principle, that is based on the frequent exchange of trained models between the edge devices and the server, but also the limited computational and storage resources in IIoT devices render the compression and acceleration of the models an imperative requirement.

Thus our work focuses on providing an efficient framework for anomaly detection by utilizing compressed NNs at the edge following the FL paradigm, allowing the training of the models from a small subset of data instances. In literature, only several studies have explored this perspective limited to i.i.d. scenarios where the edge devices (or clients) contain sensors that measure the same physical quantities, thus sharing the same feature space [14], [15]. Different from the above works,

This paper has received funding from the European Union's H2020 research and innovation programme EnerMan under grant agreement No 958478.

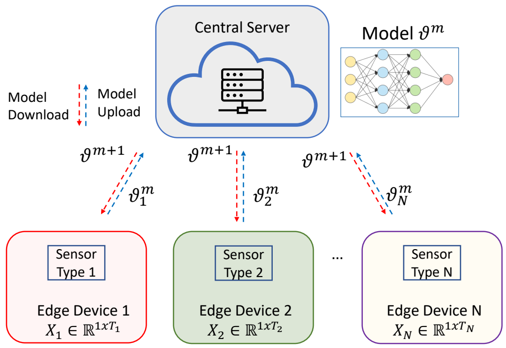

Fig. 1. The proposed resource-efficient federated learning protocol, where each edge device contains sensors that measure different physical quantities, thus having different feature space. In this illustration, each edge device has access to only one sensor ($\boldsymbol{M_i} = \mathbf{1}$).

in this study, we explore a more realistic federated learning non i.i.d. scenario considering that the edge devices measure only a part of multidimensional (multivariate) time series data. Furthermore, inspired by the compression and acceleration techniques aiming to reduce the size of deep learning networks [16]–[19] and considering the limitations of the IIoT edge devices in terms of computational and power resources [1], [20], [21], a novel compression-based optimization problem is proposed at the server-side in order to fuse the local models broadcast by the edge devices, thus deriving a compressed global model with reduced number of weights without simultaneously affecting its performance accuracy, providing **compression rates (weight pruning) greater than 99%**. To sum up, the key contributions of this paper are:

- A realistic federated learning scenario is proposed considering that the edge devices contain sensors that measure only a part of multidimensional time series data.
- A novel compressed based fusion rule is proposed at the server-side to combine the local models of the edge devices, providing compressed global models with high compression rates (more than 99%) and no performance accuracy losses.

II. Proposed Federated Learning in IIoT Architecture

To formulate the examined federated learning framework, a network with N edge devices is considered. Each edge device $i \in \{1, 2, \ldots, N\}$ consists of $\boldsymbol{M_i}$ sensors measuring different physical quantities e.g., temperature, humidity, energy consumption, e.t.c.. Particularly, each device i has locally a time series dataset $\boldsymbol{Y_i} = \{y_i^1, y_i^2, \ldots, y_i^{T_i}\}$ comprised of a sequence of T_i measurements. The $t-$th measurement $y_i^t \in \mathbb{R}_i^K$ contain M_i features measured at the time step t.

Nonetheless, in IIoT applications, the local time series data $\boldsymbol{Y}_i$ of each edge device may contain anomaly measurements due to faulty sensors and communication failures. To address this challenging task and surmount the communication and privacy drawbacks of the centralized solutions, a novel distributed approach is proposed that pushes all the involved computations toward to the edge. In particular, taking into consideration, the computational and power limitations of the edge devices [1], [20], a resource efficient federated learning architecture is derived, allowing the edge devices to learn a compressed global model without sharing their local datasets. Note that in this scenario the edge devices do not share the same feature space (i.e., each edge device may contain different sensors), hence the considered federated learning approach constitutes a non i.i.d. problem. Figure 1 illustrates the proposed federated architecture, in the case where each edge device consists of only one sensor, thus containing univariate data (i.e., $\boldsymbol{M_i} = \mathbf{1}$).

III. Proposed Resource Efficient Federated Learning Methodology

In this section, details of the proposed resource efficient federated learning approach is provided, describing the main operations of the involved entities, that is the centralized server and the dispersed edge devices.

A. Server-side

On the server-side, the server aims to compute a global model by utilizing a fusion rule that combines all the received local models from the dispersed edge devices. In vanilla FL methodology [22], the server employs an average update rule of the edge devices models derived by the following optimization problem.

$$\theta_g^m = \arg\min_{\theta_g^m} \sum_{i=1}^{N} \left\|\theta_g^m - \theta_i^m\right\|_{\mathrm{F}}^2 . \tag{1}$$

where θ_g^m, θ_i^m denote the global model and the local model of the i$-th$ edge device at the m$-th$ communication round.

However, considering that the limited computational and power resources of the edge devices impose major restrictions during the training and more importantly the inference time, in this study, a model compression fusion rule is proposed, which aims to combine the local models by calculating a compressed global model and thus achieving model compression with negligible accuracy loss. Hence, the proposed optimization problem is described by the following cost function

$$\arg\min_{\theta_g^m} \frac{1}{2} \sum_{i=1}^{N} \left\|\theta_g^m - \theta_i^m\right\|_{\mathrm{F}}^2 + \lambda \left\|\theta_g^m\right\|_1 . \tag{2}$$

where λ is a positive scalar constants that controls the relative importance of the $l-1$ sparsity imposed regularizer, promoting sparsity in the global model. After solving the proposed optimization problem, the server conveys the derived compressed global model back to all edge devices, and the next communication round is performed.

1) Efficient ADMM solver : The proposed compression fusion rule in (2, although convex requires special treatment due to the non-smooth $l-1$ term. Hence, the ADMM methodology [23] is employed by introducing an auxiliary variable Z in order to decouple the original problem into two individual subproblems. The corresponding augmented Lagrangian function of problem (2) is

$$\mathcal{L} = \frac{1}{2} \sum_{i=1}^{N} \left\|\theta_g^m - \theta_i^m\right\|_{\mathrm{F}}^2 + \lambda \left\|Z\right\|_1 + \frac{b}{2} \left\|Z - \theta_g^m + U/b\right\| \tag{3}$$

where U denotes the Lagrange multiplier matrix associated with the constraint [23], and $b > 0$ stands for the penalty parameter. Hence, a sequence of individual sub-problems emerges,

$$\begin{aligned}\theta_g^{m,k+1} &= \arg\min_{\theta_g^m} \mathcal{L}(\theta_g^m, Z^k, U^k) \\ Z^{k+1} &= \arg\min_{Z} \mathcal{L}(\theta_g^{m,k+1}, Z, U^k) \\ U^{k+1} &= \arg\min_{U} \mathcal{L}(\theta_g^{m,k+1}, Z^{k+1}, U)\end{aligned} \tag{4}$$

The solutions of the above problems are

$$\begin{aligned}\theta_g^{m,k+1} &= \arg\min_{\theta_g^m} \mathcal{L}(\theta_g^m, Z^k, U^k) \\ Z^{k+1} &= soft\,(\theta_g^{m,k+1} - U^k/b, \lambda/b) \\ U^{k+1} &= U^k - b(Z^{k+1} - \theta_g^{m,k+1})\end{aligned} \tag{5}$$

where the $soft(.,\tau)$ denotes the soft-thresholding operator $x = sign(x)max(\mid x \mid -\tau, 0)$. The above solutions are repeated recursively until convergence is reached.

B. Edge Devices-side

Focusing on the edge-device side, at every communication round m, each device i receives the compressed global model θ_g^m and aims to update its local model θ_i^m by employing its private local time series dataset $\boldsymbol{Y}_i$. To ensure that the local model will remain close to the global model during the training procedure, a regularized objective function is utilized

$$\arg\min_{\theta_i^m} L_i(Y_i, \theta_i^m) + \mu \left\| \theta_g^m - \theta_i^m \right\|^2 \tag{6}$$

where $L_i(\cdot)$ denotes a general definition of the loss function describing any supervised/unsupervised learning problem, where its parameters are the local time series dataset and the local model. Additionally, the second term known as proximal regularization term [24] restricts the local model to diverge significantly from the compressed global model. It should be highlighted that the above optimization problem is equivalent to the original neural network training plus a $L-2$ regularizer, thus it can be solved employing the stochastic gradient descent, since both terms are differentiable. After the local updates, the participated devices broadcast their models back to the centralized server.

C. Masked fine-tuning process

The proposed framework for federated learning may require a significant number of communication rounds to obtain a compressed and precise global model. To ensure the model achieves the desired compression rate, a masked retraining step is utilized. This step involves additional iterations (or communication rounds) between the server and edge devices until the global model reaches the required compression rate as defined by equation (2). After deriving a global model with the desired compression rate, the focus shifts to the edge devices. During their local training procedure, the devices are only permitted to update the non-zero weights. To achieve this, gradients of zero weights of the local models are masked, preventing them from updating. On the server-side, the aggregation fusion rule defined in equation (1) is applied to compute the new global model. Since the local models have only updated the non-zero weights while keeping the other weights at zero, the aggregation rule is straightforward. These masked retrained communication rounds are repeated until the performance accuracy of the compressed global model achieves a satisfactory level.

D. Autoencoder-based model for anomaly detection and restoration

To tackle the anomaly detection and restoration problem in multidimensional time series data, an autoencoder-based model is proposed, deployed by the participated edge devices. In general the autoencoder aims to copy its input to its output, by projecting the data into a low dimensional latent space [25]. To increase the receptive field of the autoencoder, thus capturing the strong time dependencies among the time series data, the sliding window methodology [26] is employed. In more detail, the local time series data $\boldsymbol{Y}_i$ of each edge device i is processed into overlapping sequences with time length w, $\{X_i^q\}_{q=1}^Q$. In other words, each derived sequence $X_i^q = \{y_i^t, y_i^{t+1}, \cdots, y_i^{t+w-1}\} \in \mathbb{R}^{M_i \times w}, q = 1, \cdots Q$ consists of w measurements of the dataset Y_i.

Having derived the pre-processed local datasets, each edge-device i aims to train a local autoencoder-based model utilizing the regularized optimization problem in (6) and employing as loss function in optimization problem (6), the following,

$$L_i(X_i, \theta_i^m) = \sum_{p=1}^{P} \|x_i^q - \hat{x}_i^q\|_2^2 = \sum_{q=1}^{Q} \|x_i^q - \mathcal{D}(\mathcal{E}(x_i^q))\|_2^2, \tag{7}$$

where $x_i^q \in \mathbb{R}^l, l = M_i \times w$ denotes the vectorized version of the pre-processed local data X_i^q, $\mathcal{E}(\cdot)$ denotes the encoder network aiming to to compute the intrinsic hidden representation of the input data and $\mathcal{D}(\cdot)$ is the decoder network that targets to decode the derived hidden representation of the encoding process back to the input data.

Detect Anomalies: Focusing on the anomaly detection task, the above loss function enables the autoencoder-based models of the edge devices to learn the distribution of the normal data inside the local training datasets. Thus, once the models are trained, they are capable of estimating data points very similar to the training normal data distribution. During the inference stage, the estimated values will follow the distribution of the normal data. Hence, if an anomalous measurement occurs, the trained models will fail to reconstruct it accurately. In more detail, when the reconstruction error (anomaly score) exceeds a certain threshold E, the corresponding data point is determined as an anomaly [2]. To define a proper threshold for the anomaly detection approach, we employ the following relation:

$$E = \mu + c \cdot \sigma \tag{8}$$

where μ, σ denote the mean and variance values of the training data reconstruction error and c is a user-defined parameter that controls the sensitivity of the threshold.

IV. Experimental Part

To highlight the efficiency and applicability of the proposed resource efficient federated learning framework, experiments

TABLE I. ANOMALY DETECTION PERFORMANCE AND COMPRESSION RATE OF THE PROPOSED FEDERATED LEARNING APPROACH (I.E., FL-NON IID-24) WITH N = 24 EDGE DEVICES (EACH EDGE DEVICE HAS ONLY ONE SENSOR $M_i = 1$) COMPARED TO THE I.I.D. FEDERATED LEARNING SCENARIO WITH N = 5 EDGE DEVICES (EACH EDGE DEVICE HAS $M_i = 24$ SENSORS) AND THE CENTRALIZED SCHEME.

Anomaly value rate		Centralized	FL-iid	FL-iid compressed	**FL-non-iid**	**FL-non-iid compressed**
10%	Recall	0.9985	0.9952	0.9951	0.9947	0.991
	Prec	0.9932	0.9901	0.9942	0.9921	0.9903
	Acc	0.9996	0.9919	0.9905	0.9929	0.9910
	No. Para.	325K	325K	19.5K	10K	700
	No. features (sensors) per device	-	$M_i = 24$	$M_i = 24$	$M_i = 1$	$M_i = 1$
	Compress. rate	-	-	94%	**96.2%**	**99.78%**
30%	Recall	0.9924	0.9917	0.9864	0.9832	0.9806
	Prec	0.9883	0.9838	0.9825	0.9803	0.9804
	Acc	0.9892	0.9845	0.9814	0.9825	0.9800
	No. Para.	325K	325K	19.5K	10K	500
	No. features (sensors) per device	-	$M_i = 24$	$M_i = 24$	$M_i = 1$	$M_i = 1$
	Compress. rate	-	-	**94%**	**96.2%**	**99.84%**

were curried out on a real-world multidimensional time series dataset in the context of anomaly detection problem.

Dataset [27]: The considered multivariate time series dataset consists of 24 features derived from dispersed wireless sensors measuring various physical quantities e.g., temperature, humidity, pressure, energy consumption from a building. The derived measurements were recorded every 10 minutes over several months, thus leading to a time series dataset $\boldsymbol{Y}$ with 24 features and T=19000 measurements.

Compared Methods: To tackle the problem of anomaly detection and restoration, we considered the following approaches: (a) *Centralized scenario*: The server has access to all local time series datasets of the participated edge devices, (b) *FL-iid scenario*: In this case, the whole time series dataset is i.i.d distributed to the edge devices. In more detail, we assume that each edge device contains the same types of sensors (i.e., $\boldsymbol{M_i = 24}$ sensors), thus deriving a local time series dataset with d = 24 features and, (c) *Proposed FL approach - FL-non-iid*: In this case, we consider a scenario with more practical value, where the devices have different sensors, and thus measuring different physical quantities. Hence, since the considered dataset contains 24 features, we employed $N = 24$ edge devices with $\boldsymbol{M_i = 1}$ sensor.

Parameter Settings: Concerning the dataset, a time window with size $w = 70$ was employed to split the time series data into overlapping sequences. For the centralized and the FL-iid scenario, since they share the same feature space (i.e., the same number of features) an autoencoder with two layers of size $\{128, 64, 64, 128\}$ was used. Additionally, for the FL-non_iid scenario an autoencoder with size $\{64, 32, 32, 64\}$ was employed.

A. Anomaly Detection and Restoration

Detect Outliers: In this application, the goal is to detect the anomaly measurements in the time series dataset. To this end, we split the dataset into training, validation and test set introducing randomly outlier points. Specifically, we explored two anomaly rates i.e., $\{10\%, 30\%\}$.

Table I summarizes the anomaly detection results in terms of precision , recall and accuracy metrics. Considering Table I, it can be deduced that utilizing the proposed FL-iid scenario, we can reduce the number of parameters of the global model by **96.2%** compared to models derived by the centralized and FL-iid solutions, since the local datasets of the edge devices have smaller feature space (one feature/sensor) compared to the other two approaches. Additionally, by combining the FL-non-iid method with the proposed compressed-based strategy (termed as FL-non-iid compressed method), a **compression rate greater than 99% can be achieved without sacrificing any performance accuracy.** In more detail, in the anomaly detection task, the performance degradation of the FL-non-iid -compressed scheme is negligible (less than 1.18%) as compared with the performances of the centralized solution. Another great benefit that stems out from the proposed FL-non-iid and FL-non-iid compressed schemes is the fact that these approaches are able to provide competitive results even if the edge devices train their local models using only a subset of the multidimensional time series data (they have access to univariate data derived from only one sensor), contrary to the centralized and FL-iid solutions that have access to the whole feature space of the data.

V. CONCLUSIONS

In this work, the problem of anomaly detection and restoration was studied under a resource efficient federated learning perspective. To overcome the limitations of the centralized solutions, the the proposed federated learning scheme pushes all the involved computations at the edge, where each edge device measures only a part of a multidimensional time series. In addition, considering the limited computational and power resources of the IIoT edge devices, a novel compression-based optimization problem is proposed at the server-side in order to fuse the local models broadcast by the edge devices, thus deriving a compressed global model with reduced number of weights without simultaneously affecting its performance accuracy. Extensive experiments were performed on a real-world time series dataset, examining the anomaly detection problem to highlight the efficiency and applicability of the proposed compressed-based federated learning framework. The proposed framework is able to achieve **compression rates greater than 99**% without any degradation to its performance.

REFERENCES

[1] E. Sisinni, A. Saifullah, S. Han, U. Jennehag, and M. Gidlund, "Industrial internet of things: Challenges, opportunities, and directions," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 11, pp. 4724–4734, 2018.

[2] K. Choi, J. Yi, C. Park, and S. Yoon, "Deep learning for anomaly detection in time-series data: Review, analysis, and guidelines," *IEEE Access*, vol. 9, pp. 120 043–120 065, 2021.

[3] A. Gupta, H. P. Gupta, B. Biswas, and T. Dutta, "An unseen fault classification approach for smart appliances using ongoing multivariate time series," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 6, pp. 3731–3738, 2021.

[4] N. Jin, Y. Zeng, K. Yan, and Z. Ji, "Multivariate air quality forecasting with nested long short term memory neural network," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 12, pp. 8514–8522, 2021.

[5] Z. Che, S. Purushotham, K. Cho, D. Sontag, and Y. Liu, "Recurrent Neural Networks for Multivariate Time Series with Missing Values," *Scientific Reports*, vol. 8, no. 1, pp. 1–12, dec 2018.

[6] D. Li, D. Chen, B. Jin, L. Shi, J. Goh, and S. K. Ng, "MAD-GAN: Multivariate Anomaly Detection for Time Series Data with Generative Adversarial Networks," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 11730 LNCS, pp. 703–716, 2019.

[7] T. Wen and R. Keyes, "Time series anomaly detection using convolutional neural networks and transfer learning," 2019. [Online]. Available: https://arxiv.org/abs/1905.13628

[8] W. Zhang, X. Li, H. Ma, Z. Luo, and X. Li, "Federated learning for machinery fault diagnosis with dynamic validation and self-supervision," *Knowledge-Based Systems*, vol. 213, p. 106679, feb 2021.

[9] Y. Liu, S. Garg, J. Nie, Y. Zhang, Z. Xiong, J. Kang, and M. S. Hossain, "Deep anomaly detection for time-series data in industrial iot: A communication-efficient on-device federated learning approach," *IEEE Internet of Things Journal*, vol. 8, no. 8, pp. 6348–6358, 2021.

[10] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas, "Communication-efficient learning of deep networks from decentralized data," in *AISTATS*, 2017.

[11] J. Ding, E. Tramel, A. K. Sahu, S. Wu, S. Avestimehr, and T. Zhang, "Federated learning challenges and opportunities: An outlook," in *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 8752–8756.

[12] F. Sattler, K.-R. Müller, T. Wiegand, and W. Samek, "On the byzantine robustness of clustered federated learning," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 8861–8865.

[13] A. Gkillas and A. S. Lalos, "Missing data imputation for multivariate time series in industrial iot: A federated learning approach," in *2022 IEEE 20th International Conference on Industrial Informatics (INDIN)*, 2022, pp. 87–94.

[14] W. Zhang, X. Li, H. Ma, Z. Luo, and X. Li, "Federated learning for machinery fault diagnosis with dynamic validation and self-supervision," *Knowledge-Based Systems*, vol. 213, p. 106679, feb 2021.

[15] X. Zhou, X. Liu, G. Lan, and J. Wu, "Federated conditional generative adversarial nets imputation method for air quality missing data," *Knowledge-Based Systems*, vol. 228, p. 107261, 2021.

[16] Y. Cheng, D. Wang, P. Zhou, and T. Zhang, "Model compression and acceleration for deep neural networks: The principles, progress, and challenges," *IEEE Signal Processing Magazine*, vol. 35, no. 1, pp. 126–136, 2018.

[17] S. Nousias, E. V. Pikoulis, C. Mavrokefalidis, and A. S. Lalos, "Accelerating deep neural networks for efficient scene understanding in automotive cyber-physical systems," in *2021 4th IEEE International Conference on Industrial Cyber-Physical Systems (ICPS)*, 2021, pp. 63–69.

[18] T. Zhang, S. Ye, K. Zhang, J. Tang, W. Wen, M. Fardad, and Y. Wang, "A systematic dnn weight pruning framework using alternating direction method of multipliers," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 184–199.

[19] B. Liu, M. Wang, H. Foroosh, M. Tappen, and M. Pensky, "Sparse convolutional neural networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2015, pp. 806–814.

[20] J. Tang, D. Sun, S. Liu, and J.-L. Gaudiot, "Enabling deep learning on iot devices," *Computer*, vol. 50, no. 10, pp. 92–96, 2017.

[21] S. Nousias, E.-V. Pikoulis, C. Mavrokefalidis, and A. S. Lalos, "Accelerating deep neural networks for efficient scene understanding in multimodal automotive applications," *IEEE Access*, vol. 11, pp. 28 208–28 221, 2023.

[22] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y. Arcas, "Communication-Efficient Learning of Deep Networks from Decentralized Data," in *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, ser. Proceedings of Machine Learning Research, A. Singh and J. Zhu, Eds., vol. 54. PMLR, 20–22 Apr 2017, pp. 1273–1282.

[23] S. Boyd, N. Parikh, E. Chu, J. Eckstein, S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers," *Foundations and Trends R in Machine Learning*, vol. 3, no. 1, pp. 1–122, 2010.

[24] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," in *Proceedings of Machine Learning and Systems*, I. Dhillon, D. Papailiopoulos, and V. Sze, Eds., vol. 2, 2020, pp. 429–450.

[25] P. V. Ca, L. T. Edu, I. Lajoie, Y. B. Ca, and P.-A. M. Ca, "Stacked Denoising Autoencoders: Learning Useful Representations in a Deep Network with a Local Denoising Criterion Pascal Vincent Hugo Larochelle Yoshua Bengio Pierre-Antoine Manzagol," Tech. Rep., 2010.

[26] H. S. Hota, R. Handa, and A. K. Shrivas, "Time series data prediction using sliding window based rbf neural network," 2017.

[27] L. M. Candanedo, V. Feldheim, and D. Deramaix, "Data driven prediction models of energy use of appliances in a low-energy house," *Energy and Buildings*, vol. 140, pp. 81–97, apr 2017.