We propose an adaptation of the unsupervised batch-SOFM competitive learning algorithm to pre-process the raw data set of LoRa I/Qs into SOFMs, thereby attaining the objectives of enhanced ML model performance resulting in accurate localization; and, an expeditiousness in device authentication that is measured in the order of the LoRa frame duration (i.e., $\leq \mathcal{O}\{32 \cdot T_S\}$, in-turn attained by dimensionality reduction in the order of $\mathcal{O}\{10^4\}$ per LoRa device; computed as a ratio of the number of raw LoRa I/Q samples to the number of neurons in the ANN, i.e., $\frac{M}{U \times V} = \frac{2 \times 10^6}{10 \times 10} = 2 \times 10^4$. Illustrated in Fig. 1 are the steps of the algorithm. Here, the input layer of every LoRa I/Q in the raw data set $\boldsymbol{X} = [\boldsymbol{x}_1^T, \boldsymbol{x}_2^T, \cdots, \boldsymbol{x}_6^T] \in \mathbb{C}^{M \times S}$ is first indexed serially. Then, a hidden layer of weight matrices are randomly initialised and batched as given by $\boldsymbol{Z}(j=1) = \boldsymbol{Z}_{\boldsymbol{x}_1}(j=1), \boldsymbol{Z}_{\boldsymbol{x}_2}(j=1), \cdots, \boldsymbol{Z}_{\boldsymbol{x}_6}(j=1)]$, where $\boldsymbol{Z}(j=1) \in \mathbb{R}^{M \cdot U \times S \cdot V}$ and where any $s$-th matrix $\boldsymbol{Z}_{\boldsymbol{x}_s}(j=1) \in \boldsymbol{Z}(j=1)$ associates every LoRa I/Q sample in the $s$-th LoRa I/Q vector $\boldsymbol{x}_S$ to the ANN matrix (output layer) $\boldsymbol{A}(j=1) \in \mathbb{R}^{U \times V}$, that is created such that $U, V \ll M$ (thus ensuring that the SOFMs are of low rank which also results in dimensionality reduction [1]; on the $j = 1$-st epoch, $j \in \{1, \cdots, J\}$, as shown in Fig. 1. Next, the matrix of $M \cdot S$ Euclidean norms $\boldsymbol{D}(j=1) = \boldsymbol{d}_{\boldsymbol{x}_1}^T(j=1), \boldsymbol{d}_{\boldsymbol{x}_2}^T(j=1), \cdots, \boldsymbol{d}_{\boldsymbol{x}_6}^T(j=1)]$ are computed, in which any $s$-th vector of $M$ Euclidean norms are given by $\boldsymbol{d}_{x_{s \in \mathcal{S}}}^T(j=1) = \boldsymbol{x}_{\boldsymbol{s}}^T(j=1) - \boldsymbol{Z}_{\boldsymbol{x}_s}(j=1)$; from which the batch of $S = 6$ ``winning" neurons in the ANN $\boldsymbol{A}(j=1)$, known as the best matching units (BMUs), given by $[a_{u,v}(j=1) \leftarrow I_{\boldsymbol{x}_1}(j=1) = min\{\boldsymbol{d}_{\boldsymbol{x}_1}(j=1)\}, a_{u',v'}(j=1) \leftarrow I_{\boldsymbol{x}_2}(j=1) = min\{\boldsymbol{d}_{\boldsymbol{x}_2}(j=1)\}, \cdots, a_{u'''''',v'''''''}(j=1) \leftarrow I_{\boldsymbol{x}_6}(j=1) = min\{d_{\boldsymbol{x}_6}(j=1)\}]$ are selected which minimize the Euclidean norms. Utilizing this batch of BMUs, 6 batch update matrices $[\delta \boldsymbol{A}_{\boldsymbol{x}_1}(j=1) = \mu_j(I_{\boldsymbol{x}_1}(j=1)), \delta \boldsymbol{A}_{\boldsymbol{x}_2}(j=1) = \mu_j(I_{\boldsymbol{x}_2}(j=1)), \cdots, \delta \boldsymbol{A}_{\boldsymbol{x}_6}(j=1) = \mu_j(I_{\boldsymbol{x}_6}(j=1))]$ are computed on the $j = 1$-st epoch; where the hyperparameter $\mu_j(\cdot)$ represents the learning rate with any $s$-th batch update matrix $\delta A_{\boldsymbol{x}_{s \in \mathbb{S}}}(j=1)$ having dimensionality $\mathbb{R}^{U \times V}$. At the end of the $j = 1$-st epoch, each of the 6 batch update matrices are added to the original ANN $\boldsymbol{A}(j=1)$ to produce a batch of 6 offspring ANNs denoted as $\boldsymbol{A}(j=1) = [\boldsymbol{A}_{\boldsymbol{x}_1}(j=1) = \delta \boldsymbol{A}_{\boldsymbol{x}_1}(j=1) + \boldsymbol{A}(j=1), \boldsymbol{A}_{\boldsymbol{x}_2}(j=1) = \delta \boldsymbol{A}_{\boldsymbol{x}_2}(j=1) + \boldsymbol{A}(j=1), \cdots, \boldsymbol{A}_{\boldsymbol{x}_6}(j=1) = \delta \boldsymbol{A}_{\boldsymbol{x}_6}(j=1) + \boldsymbol{A}(j=1)]$. This procedure is repeated from the $j \geq 2$-nd epoch onward, with the exception that the 6 batch update matrices obtained at the end of any $j \geq 2$-nd epoch are added to the batch of 6 offspring ANNs derived at the previous $j - 1 \geq 2$ epoch instead of the original ANN; as also depicted in Fig. 1. Lastly, the batch of 6 offspring ANN matrices generated at the $j \leq J$-th epoch denoted by $\mathcal{A}(J) = [\boldsymbol{A}_{\boldsymbol{x}_1}(J), \boldsymbol{A}_{\boldsymbol{x}_2}(J), \cdots, \boldsymbol{A}_{\boldsymbol{x}_6}(J)]$ characterize the pre-processed SOFM data sets of the original LoRa I/Qs; with each SOFM being associated with a specific LoRa device.

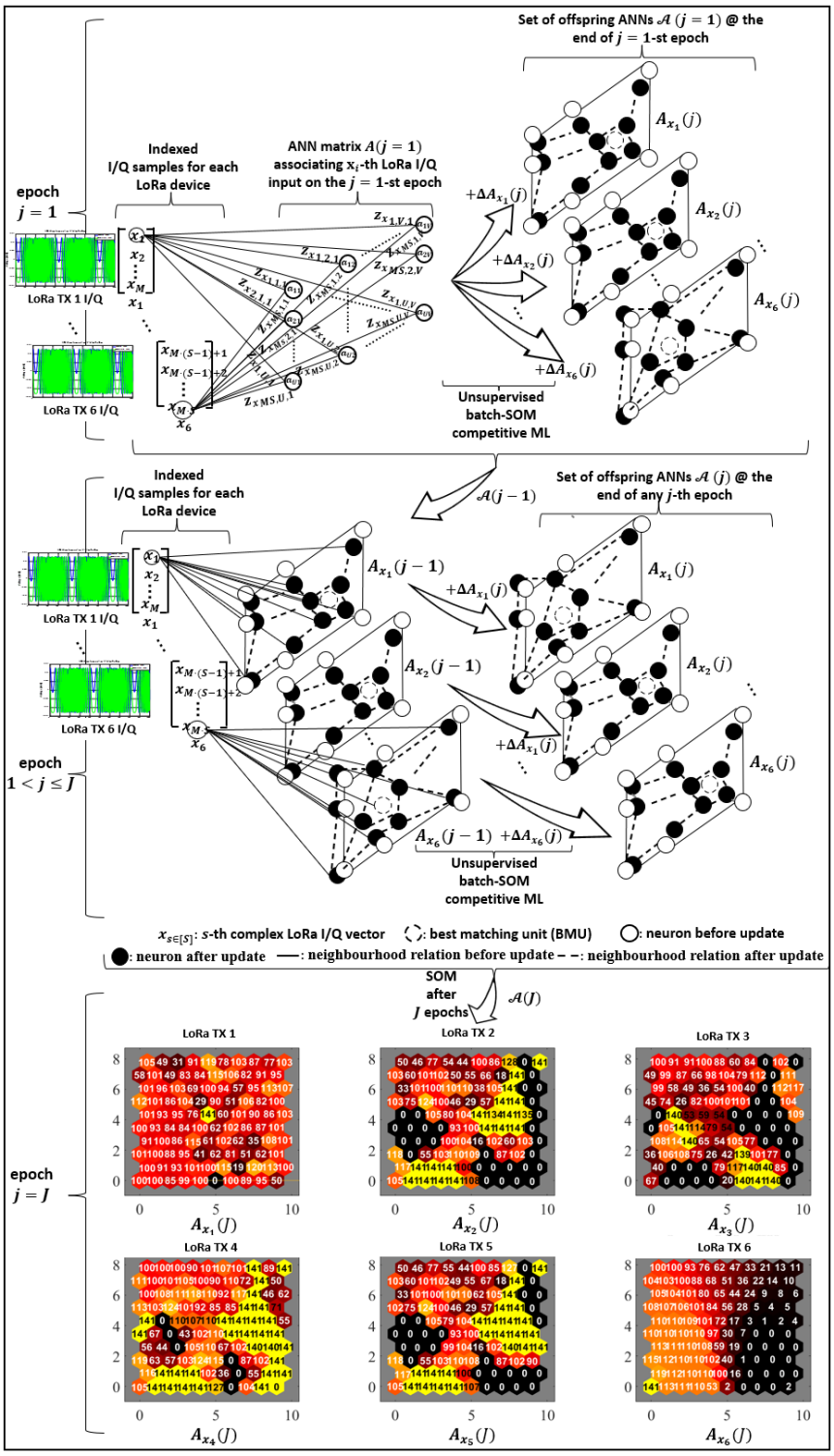


Figure 1: Conceptual illustration of SOFM generation from the data set of LoRa I/Q vectors $\boldsymbol{X}=[\boldsymbol{x}_1^T,\cdots,\boldsymbol{x}_6^T]$ acting on ANN $\boldsymbol{A}(j=1)$ initialized on the $j=1$-st epoch, by an unsupervised ML, over $J$ epochs. White circles are neurons of the ANN, black circles are neurons after ANN update, dashed circle is BMU and dashed lines are the Euclidean norm. The set of offspring ANNs $\boldsymbol{A}(J)$ after $J$-th epoch characterizes the pre-processed SOFM images. The integers on the neurons constituting each of the SOFMs in $\mathcal{A}(J)$ are their extent of cluster by the $J$-th epoch. In each SOFM, a particular neuron with the lowest integer value (dark color-map), representing the extent of greatest cluster by the $J$-th epoch, is the possible BMU.