Uncertainty-Aware Deep Learning Architectures for Highly Dynamic Air Quality Prediction

Ichrak Mokhtari, Walid Bechkit, Hervé Rivano, Mouloud Riadh Yaici

IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received November 26, 2020, accepted January 4, 2021. Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2021.3052429

Uncertainty-Aware Deep Learning Architectures for Highly Dynamic Air Quality Prediction

ICHRAK MOKHTARI[1], WALID BECHKIT[1], HERVÉ RIVANO[1], AND MOULOUD RIADH YAICI[1,2]

[1]CITI Laboratory, INSA Lyon, F-69621 Villeurbanne, France
[2]LMCS laboratory, Ecole Nationale Supérieure d'Informatique ESI, Ouer Smar 16270, Algeria

Corresponding author: Ichrak Mokhtari (ichrak.mokhtari@insa-lyon.fr)

This work was supported in part by the LABEX IMU of Université de Lyon under Grant ANR-10-LABX-0088, and in part by the "Investissements d'Avenir" Program operated by the French National Research Agency (ANR) under Grant ANR-11-IDEX-0007.

ABSTRACT Forecasting air pollution is considered as an essential key for early warning and control management of air pollution, especially in emergency situations, where big amounts of pollutants are quickly released in the air, causing considerable damages. Predicting pollution in such situations is particularly challenging due to the strong dynamic of the phenomenon and the various spatio-temporal factors affecting air pollution dispersion. In addition, providing uncertainty estimates of prediction makes the forecasting model more trustworthy, which helps decision-makers to take appropriate actions with more confidence regarding the pollution crisis. In this study, we propose a multi-point deep learning model based on convolutional long short term memory (ConvLSTM) for highly dynamic air quality forecasting. ConvLSTM architectures combines long short term memory (LSTM) and convolutional neural network (CNN), which allows to mine both temporal and spatial data features. In addition, uncertainty quantification methods were implemented on top of our model's architecture and their performances were further excavated. We conduct extensive experimental evaluations using a real and highly dynamic air pollution data set called Fusion Field Trial 2007 (FFT07). The results demonstrate the superiority of our proposed deep learning model in comparison to state-of-the-art methods including machine and deep learning techniques. Finally, we discuss the results of the uncertainty techniques and we derive insights.

INDEX TERMS Conv-LSTM, spatio-temporel prediction, highly dynamic air quality, accidental pollutant release, uncertainty, FFT-07, WSN.

I. INTRODUCTION

Air pollution is a global human hazard, causing every year considerable damage to human health, the environment and the worldwide economy. According to World Health Organization (WHO), 9 out of 10 people breath air that exceeds the limits of WHO guidelines in terms of pollution levels, resulting in 4.2 million annual deaths related directly to pollution [1]. Therefore, there is a tremendous pressure on decision-makers to develop effective pollution maps that would allow management plans with an emphasis on prevention. This need is even greater in emergency situations where real-time forecast maps are highly required to set up crisis management strategies and evacuation models. In addition, these maps can be particularly useful for designing anticipatory mobility models of drones to track pollution plumes and to better characterize the dispersion phenomenon.

In recent years, a rising number of natural (volcanic eruption, etc) and man-made pollution disasters (transport of hazardous materials, terrorist attacks, etc) have caused considerable damages to human health and the environment. As seen in these emergency situations such as Fukushima explosion in Japan (March 2011), the Lubrizol accident in France (October 2019) or more recently the Beyrouth harbour explosion in Lebanon (August 2020), very large amounts of pollutants are released and quickly transported over the air. In these cases, the strong and unpredictable dynamics of the pollution plumes, the lack of knowledge of pollution sources and emission rates make the phenomenon complex to model. Furthermore, the strong need for reactivity and

The associate editor coordinating the review of this manuscript and approving it for publication was Zhe Xiao.

rapid responses, make the use of traditional monitoring solutions (conventional air quality monitoring stations, dispersion models alone) difficult and usually inefficient.

At the same time, the recent democratisation of low-cost wireless sensor networks (WSNs) gives new impetus to the dynamic air quality monitoring. Indeed, the small size of sensors and their large number enable data collection with a high spatial and temporal resolution. Moreover, these sensors can be easily mounted on mobile vehicles such as drones, allowing precise mapping of gas plumes in hard-to-reach areas, where human intervention is not an option and the need for rapid response is primordial. Drones have already been used to this purpose; for example, the authors of [2] used a helicopter UAV to measure volcanic gases at La Fossacrater Volcano (Italy). Coupled with adequate spatio-temporal learning models, measurements made by these sensors would enable a precise mapping and better forecasting of plume dispersion and thus make it possible to anticipate potential catastrophes and mitigate their consequences. Thereby, the use of such solutions would have undeniable advantages in terms of plume knowledge quality, time and cost which are crucial parameters in emergency situations.

For several years, great efforts have been devoted to the study of chronic air pollution forecasting where innovative solutions including machine and deep learning techniques were proposed to address this issue [3], [4] [5]. However, the spatio-temporal analysis of punctual and highly dynamic pollution emissions in emergency situations has attracted less attention, although predicting pollution plume in these particular situations is essential as mentioned earlier. This interest is even more important in the case of toxic, explosive or irradiating pollutant releases, following natural disasters, industrial accidents or terrorist attack. Considering the criticality and severity of the risks during these situations, it is important to produce fairly accurate real-time plume forecast maps with estimations of prediction uncertainties; this last aspect has often been neglected when designing pollution prediction models. In fact, providing such information, especially in an emergency situation, makes the models more reliable and allows better crisis management.

This research aims at designing an adequate spatio-temporal forecasting model for high dynamic air pollution. The main points that guided our design are:

- Proposing a rapid and efficient multi-point framework able to forecast concentrations in several locations (nodes) at the same time when taking spatio-temporal data dependencies.
- Taking into account the uncertainty of the predictions. Indeed, this parameter is crucial in the case of monitoring dynamic phenomena with mobile vehicles as it may guide their mobility to improve the quality of the monitoring.
- Validating our architecture on highly dynamic real data sets. This objective was not obvious to reach given the lack of such real data.

The main contributions of our work can be summarized as follows:

- We investigate the state of the art of air pollution forecasting methods in which top-level articles are reviewed. We also propose a classification scheme to analyze the existing literature.
- We designed a multi point spatio-temporal deep learning model based on ConvLSTM able to mine efficiently spatio-temporal data pollution for high dynamic pollution prediction.
- We incorporate uncertainty quantification techniques to our architecture to make the model more trustworthy. Numerous experiments have been conducted when varying model's parameters.
- We compared our solution to a great set of machine and deep learning techniques through extensive experiments. These latter has demonstrated the superiority of our model over the considered baselines. For that end, we have identified and taken over a highly dynamic real data set namely Fusion Field Trials (FFT07). To the best of our knowledge, no learning based solution has been tested on real data in such dynamic environment.

The remainder of this paper is organized as follows: Section II presents the literature review. Section III shows some preliminaries of deep learning models as well as uncertainty background and it then presents in details our solution's design. Section IV describes the comparative experiments: the effectiveness of the developed model is analyzed and evaluated and then we conclude this section by the evaluation of the uncertainty techniques. Finally, Section V summarizes the results of this work and draws conclusions.

II. LITERATURE REVIEW

The importance of intelligent prediction systems for short and long-term pollution forecasting is critical to alleviate damages. Nonetheless, designing such systems can be extremely complex due to the dynamic nature of pollution plume and the lack of knowledge about the pollution sources and their emission rates. As a result, mimicking the complex mathematical expressions of the corresponding physical process can be challenging. Two categories of methods can be identified to tackle the air pollution forecasts issue, namely: physical models and data-driven models.

Physically based models, also called chemical transport models were widely used in the past on pollution field [6]. These models provide both spatial and temporal estimations of air quality by simulating the pollutants dispersion process. In the literature, several kind of models have been developed depending on the length of the scale considered to characterize the pollution dispersion. Three main scales can be identified: the regional scale (5-50 km), the urban scale (1-5 km) and the local scale (up to 1 km) [7]. Among the regional scale models CHIMERE [8] and CamX [9] are widely used while SIRANE [6] and ADMS-Urban [10] are dedicated models for urban scale. SIRANERISK [11] is another chemical transport

model dedicated to the dispersion of pollutants in urban areas in unstable situations due to unexpected releases. Despite the undeniable usefulness of these physical models, these latter suffer from various drawbacks. Indeed, chemical transport models need different input data including pollution sources, meteorological data (temperature, wind speed and direction, etc.) as well as land use data (buildings, roads, etc.). However, it is sometimes difficult to know precisely the exact sources of pollution, especially during sudden pollution events, and hence resulting on a poor predictability. Moreover, previous researches have shown the latter's shortcomings regarding short-term forecasting [12], [13]. Finally, some studies highlighted the expertise and the in-depth knowledge needed to develop such models besides the computation capacities required for run [14]. On the other hand, combining observations to numerical models in what's called data assimilation (DA) have shown great potential to correct and improve these latter [15]. Nonetheless, Denby et al have pointed out the need of unbiased models for DA techniques in order to meet assumptions made in their formulation and hence achieve better results [16].

Data-driven techniques have also a long tradition with air pollution forecasting. These methods assimilate available meteorological indices and air quality measurements to provide better insights without relying on physical or chemical processes. Within this category, approaches like autoregressive integrated moving average (ARIMA) [17] or generalized additive models (GAMs) [18] were among the early statistical methods for predicting air pollution. That is, these techniques were reported to be significantly limited in term of performance due to the non-realistic hypothesis taken, and therefore need more improvements. Later, the shortcomings of numerical and statistical methods encouraged the use of more advanced data-driven approaches, e.g., machine learning (ML). A further reason for the wide use of these techniques is their capacities to formulate the non-linearity air pollution constraints, entirely relying on historical data, and without any primary knowledge on the physical equations. In that context, Ochando *et al.* have proven the suitability of ML algorithms as random forest (RF) over conventional techniques for all the conducted experiments [19]. A bit later, significant advances in hardware and the symbiosis of data collection opened the doors to the use of more complex ML models based on neural networks. Many works have reviewed recent solutions using artificial neural networks (ANNs) for air pollution predictions and have claimed their ability to extract efficient representations of relevant functionalities and characteristics from large amounts of data [20], [21]. In comparison to traditional approaches, deep learning solutions has shown greater accuracy [22]. In addition, their hybridization with other ML techniques as K nearest neighbours for instance has been shown to significantly improve their performances [3], [23]. However, certain characteristics of ML algorithms must be highlighted to be carefully taken into consideration. Among them, the data dependence of these algorithms. Indeed, the data fed to these latter, must be sufficiently varied and broad to allow their generalization. Regarding the scarcity of data, some work in the literature has demonstrated the effectiveness of using the transfer learning approach [24]. The principle of this latter is to use a model already trained for one task as a starting point for another task.

In what follows, we review some interesting works in the field. We start with solutions exploring the spatial prediction, then temporal one before discussing solutions addressing spatio-temporal forecasting.

The authors of [25] propose a general framework to compare between regression approaches and data assimilation ones for annual NO_2 forecasts in urban area. The study included the following regression methods: random forest, extreme gradient boosting (XGBoost), land use regression (LUR), KNN and best linear unbiased estimator (BLUE) as data assimilation technique. Results have demonstrated the superiority of BLUE over other approaches when the simulation and sensing errors are well characterized.

While the former research work focus on spatial air quality prediction, some effort has been devoted to the study of temporal prediction. In [26], a novel variant of support vector machine (SVM) combined with Principal component analysis (PCA) was proposed to predict daily $PM_{2.5}$. The PCA is first used to extract relevant features and reduce the inputs dimension and then the prediction is performed by a least squares SVM model. The experiments have demonstrated the superiority of the developed model over the baseline techniques considered: general regression neural network (GRNN) and least square support vector regression (SVR) with default parameters.

In a recent work [24], Ma *et al.* tackle the forecasting pollution problem at different temporal resolution (hourly, daily and weekly) using a bidirectional LSTM. The proposed model were compared with other commonly used methods including ARIMA, gradient boost decision tree (GBDT), SVR, RNN, gated recurrent unit (GRU) and LSTM. The authors evaluate also the effectiveness of transfer learning from smaller temporal resolutions to larger ones. A case study on $PM_{2.5}$ in Guangdong (China) have been conducted and the results have shown that the developed methodology can effectively improve the prediction accuracy for larger temporal resolutions.

Authors of [27], present a deep neural network model named ApNets to predict $PM_{2.5}$ concentrations using a combination of CNN and LSTM network. The objective is to forecast the concentration of $PM_{2.5}$ for the next hour based on the previous concentrations, wind speed and cumulative hours of rain over the last 24 hours. ApNets was compared to many traditional approaches like multiple layer perceptron, decision tree and random forest. Results showed that APNet enhances the prediction accuracy compared to the other considered solutions.

Others research works exploited the spatial correlations between stations besides the temporal dependencies to provide better performances. In [28], Soh *et al.* propose a model that combine an artificial neural network with CNN and

TABLE 1. Summary of research works on forecasting air pollutants using different methods.

Reference	Location	Pollutants	Forecast Type	Method	Resolution
[24]	Guangdong (China)	$PM_{2.5}$	Temporal	bidirectional LSTM	Hourly
[25]	Lyon (France)	NO_2	Spatial (20 m)	BLUE, KNN, RF, XGBoost, LUR	Annual
[26]	Hebei (China)	$PM_{2.5}$	Temporal	Hybrid (PCA, LSTM)	Daily
[27]	Beijing (China)	$PM_{2.5}$	Temporal	Hybrid (1D CNN, LSTM)	Hourly
[28]	Taiwan, Bejing (China)	$PM_{2.5}$	Spatio-temporal	Hybrid (CNN, LSTM, ANN	Hourly
[3]	Beijing (China)	$PM_{2.5}$	Spatio-temporal	Hybrid (CNN, bidirectional LSTM)	Hourly
[5]	Bejing (China)	$PM_{2.5}$	Spatio-temporal	Ensemble Deep Learning Model using LSTM	Hourly
[16]	Europe	PM_{10}	Spatio-temporal	Data Assimilation:Ensemble Kalman filtering, residual kriging	Daily, annual

LSTM in order to extract spatio-temporal relations and to predict air quality for up to 48 h. The proposed predictive model considers different meteorological data from the previous few hours as well as information related to the elevation space to extract terrain impact on air quality. Results showed that the developed model achieves good performance and outperforms current state-of-the-art methods.

Du *et al.* develop a deep forecasting air quality model named DAQFF for $PM_{2.5}$ single step forward and multi-step forward prediction. The framework is based on two components : a multiple one dimension convolution layers (CNN) and a bidirectional LSTM [3]. The former is used to extract spatial features between stations and the latter to capture the temporal features in the time series data. Results showed that DAQFF outperforms a great set of baseline models.

A deep spatial-temporal ensemble (STE) model is proposed to predict air quality using historical air quality observations and meteorological data in [5]. The framework is divided into three components. The first component consists of an ensemble method with a partitioning strategy based on weather pattern. It trains multiple models and dynamically combines them. The objective of the second component is to discover spatial correlation between stations by analyzing Granger causality and select relative areas and relative stations to add spatial properties to the predictor. The last one is a temporal predictor that utilizes a deep LSTM to learn both long-term and short-term dependencies of air quality data. For comparison purposes, we summarize studies regarding different techniques in Table 1.

In the above, several solutions for forecasting atmospheric pollution have been studied. Nevertheless, there are still some interesting and relevant problems to be addressed. First, most of the designed solutions focus on predicting chronic pollution in cities or at a larger scale. However, when considering the emergency situations, the pollution plume is strongly dynamic and the spatial and temporal resolution needed is much higher to efficiently cover the phenomenon in real time. Little researches consider the numerous challenges in such situations. In addition, very few work take into consideration the measurement uncertainties of sensors and even less the uncertainties of their forecasting models. The latter can be useful, for example, during sudden pollution events to guide drone fleets when tracking pollution plumes. Finally, most research works that consider the spatio-temporal relationship between the various inputs, propose complex models to take into account the information from neighboring stations and thus improve the prediction of a target one. However, these predict the outputs of a single point at a time and therefore require several models to forecast all the points. In this work, we aim to provide solutions that address the identified cutting-edge gaps.

III. BACKGROUND, PROBLEM FORMULATION AND OUR ARCHITECTURE

In this section, we present our designed deep learning model for high dynamic air pollution forecasting. Before diving deeper into this latter, we go over some preliminaries. We first introduce the LSTM, CNN and ConvLSTM models. Next, background on uncertainty is discussed. Finally, we provide a formulation of our problem before describing the detailed architecture of our uncertainty aware ConvLSTM framework.

A. LONG SHORT TERM MEMORY NETWORK

LSTM is an improved version of recurrent neural network (RNN) proposed by Hochreiter and Schmidhuber [29]. It uses the concepts of cell memory and gates to bypass the problem of RNN's exploding gradients. The cell represents the memory of the block, it stores the parameters over arbitrary time intervals while the gates regulate the flow of information into and out of the cell. By doing that, LSTM can pass important information down the long chain of sequences to make predictions. Thus, it is well suited to process time series data with relatively long delays and intervals. The LSTM architecture is given by the following equations:

$$i_t = \sigma(W_{x_i}x_t + W_{h_i}h_{t-1} + b_i) \quad (1)$$
$$f_t = \sigma(W_{x_f}x_t + W_{h_f}h_{t-1} + b_f) \quad (2)$$
$$g_t = tanh(W_{x_g}x_t + W_{h_g}h_{t-1} + b_c) \quad (3)$$
$$c_t = f_t * c_{t-1} + i_t * g_t \quad (4)$$
$$o_t = \sigma(W_{x_o}x_t + W_{h_o}h_{t-1} + b_o) \quad (5)$$
$$h_t = o_t * tanh(c_t) \quad (6)$$

In these equations σ and tanh are sigmoid and hyperbolic tangent activation functions. x_t, c_t and h_t represent current input, cell memory state and hidden state respectively.

i_t, f_t, o_t and g_t represent input gate, forgetting gate, output gate and input modulation gate across which the network can

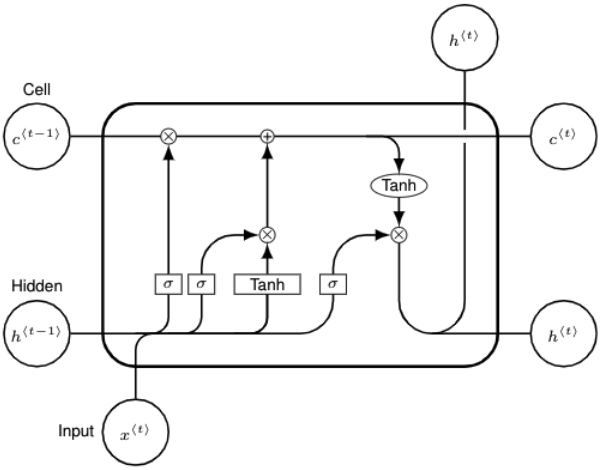

FIGURE 1. Architecture of LSTM Cell.

selectively retain and forget past and present information. W and b are the model parameters (weights and bias). The outputs of the LSTM cell are o_t and c_t. Finally $*$ denotes the Schur product. The internal architecture of an LSTM cell is presented in Figure 1. For further information, one can refer to [30].

B. CONVOLUTIONAL NEURAL NETWORK

CNN is a subclass of artificial neural networks, usually used to process images. The input is a vector with three dimensions namely height, width and channel (depth). Channel is equal to three if the input is an RGB image and one in the case of Grayscale image. CNN's architecture is made up of two blocks. The former works as a feature extractor, it filters the image with several convolution kernels and returns "feature maps", which are then normalized. The latter block exploits the previous layer's result to make a prediction.

CNNs have strong abilities to extract spatial correlation features. Therefore, they have been widely explored in the image recognition and computer vision fields, offering improvements over deep neural networks (DNNs) on many tasks [31], [32]. Recently, CNNs have been exploited for air pollution forecasting [33], [34].

C. CONVOLUTIONAL LSTM

ConvLSTM was introduced for the first time by Shi *et al.* [35] to tackle the precipitation nowcasting problem. ConvLSTM is a variant of the standard LSTM model where the convolution operation is embedded at each gate of the LSTM cell resulting to a strong ability to process spatial and temporal information simultaneously. Mathematically, the computation of ConvLSTM can be described by the following equations:

$$i_t = \sigma(W_{x_i} \circledast x_t + W_{h_i} \circledast h_{t-1} + b_i) \quad (7)$$
$$f_t = \sigma(W_{x_f} \circledast x_t + W_{h_f} \circledast h_{t-1} + b_f) \quad (8)$$
$$g_t = tanh(W_{x_g} \circledast x_t + W_{h_g} \circledast h_{t-1} + b_c) \quad (9)$$
$$c_t = f_t * c_{t-1} + i_t * g_t \quad (10)$$

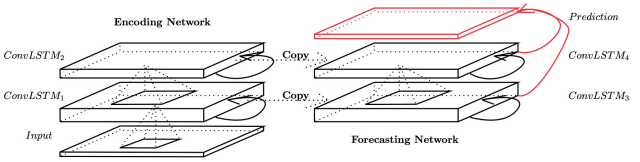

FIGURE 2. Encoding-Forecasting ConvLSTM structure for spatio-temporal sequence predicting [35].

$$o_t = \sigma(W_{x_o} \circledast x_t + W_{h_o} \circledast h_{t-1} + b_o) \quad (11)$$
$$h_t = o_t * tanh(c_t) \quad (12)$$

where $\circledast$ denotes the matrix vector operation. As shown in Figure 2, the convLSTM cell is composed of two networks. The former is the encoding network and the latter is the forecasting network.

This network has since shown its effectiveness in various spatio-temporal problems [36].

D. BACKGROUND ON UNCERTAINTY

In many applications, it is not enough that a model performs well on average, but the uncertainty of each estimate must also be quantified. This can be particularly crucial when there is a considerable downside to an incorrect or imprecise forecast, such as the case of high dynamic pollution events that occur suddenly. Unfortunately, little works have considered uncertainty in the pollution field as seen in section 2.

One can distinguish two types of uncertainties: epistemic uncertainty and aleatory uncertainty [37]. The former uncertainty also called model uncertainty is the uncertainty resulting from a lack of input data and knowledge. The latter represent the natural randomness of the real data generating process.

A diverse set of methods are proposed in the literature to cope with neural network uncertainty. Among these approaches, we count the fully bayesian NNs [38], Monte Carlo (MC) dropout method [39], bootstrap [40], quantile regression (QR) [41], gaussian process inference, maximum likelihood estimation [42] or deep ensemble [43]. In what follows, we briefly review the theoretical background of MC dropout and QR techniques. The remaining methods require either strong assumptions on sample distribution, modifications of the neural network or of the way it is trained or it requires training several networks (computationally expensive), they are therefore not taken into consideration in this work.

MC dropout was introduced as a technique to estimate uncertainty prediction for the first time in [39]. The authors described it as an approximation for a bayesian approach. The idea behind it is to mask (drop) randomly a certain number of the network's nodes. Concretely this is done by setting the outputs of the selected nodes to zero. To estimate the uncertainty, the authors of [39] suggest to activate the dropout during the inference phase (prediction time), thus getting multiple outputs for every single input. We call a single prediction step when activating dropout a stochastic forward

pass. This procedure allows to have an ensemble of sub-networks within one single network, that have marginally distinctive views of the data. The objective is to get a distribution of the target variables and therefore an estimation of uncertainty. The main drawback of MC dropout is it requires to make multiple predictions for every new data point to figure out the variance of the predictions.

Quantile Regression is a particular kind of regression introduced by [41]. This approach aims to estimate the conditional quantiles y. Therfore the QR is capable of modeling the entire conditional distribution of y without any prior assumption of residuals as opposed to linear regression. This is done by minimizing the quantile loss defined as follows:

$$L_\tau(y, \hat{y}) = \sum_{i|y_i<\hat{y}_i} (1-\tau)|y_i - \hat{y}_i| + \sum_{i|y_i\geq\hat{y}_i} \tau|y_i - \hat{y}_i| \quad (13)$$

One of the shortcomings of quantile regression is that it produces non symmetric intervals from the median estimation.

E. FORMULATION OF THE PROBLEM

Given the previously observed pollution data, our study aims to predict the upcoming concentration of pollution simultaneously in different nodes. The problem corresponds to a sequence spatio-temporal forecasting issue.

We divide the node coverage rectangle into a width $\times$ height grid and assign the collected air pollution data to the cells in the grid. The value in a cell represent the pollution concentration of a single node in a given timestamp. We consider that the region of interest can be approximated by a set of N discrete points. Thus, the observation at a given time can be represented as a 2D Tensor $\chi \in R^{P\times N}$ where P is the measurements that vary over time and R is the air pollution domain. Hence, the problem can be mathematically formulated as follows:

$$\hat{\chi}_{t+1}, \ldots, \hat{\chi}_{t+K} = \arg \max_{\chi_{t+1},\ldots,\chi_{t+K}} p(\chi_{t+1}, \ldots, \chi_{t+K} \mid \hat{\chi}_{t-J+1}, \hat{\chi}_{t-J+2}, \ldots, \hat{\chi}_t) \quad (14)$$

As a first step, our objective is to build a predictive model producing estimates as close as possible to real observations (ground truth). Nonetheless, sometimes point forecasts are not sufficient and a measure of uncertainty is needed to make the model more trustworthy. Uncertainty estimation allows considering a range of possible outputs rather than single values as in the case of point estimates. Therefore, as a second goal, our aim is to quantify uncertainty for pollution forecasts. Concretely this could be done by building up prediction intervals (PIs). A prediction interval communicate uncertainty by affording a lower and upper bound that bracket a future estimation with a prescribed probability called a confidence level $[(1-\alpha)\%]$. Note that PIs are different from confidence intervals. In fact, the confidence interval gives an idea of the model uncertainty whereas the PI takes also into account data variance noise.

One way to construct such intervals is to consider quantiles. For example, estimating the 0.9 and 0.1 quantiles is equivalent to say that 80% of the target values fall within this interval. Ideally, we would like PIs to be as narrow as possible, with a low mean width.

Let's consider a neural network regressor which processes an input $x \in X$ and output $y \in R$. Let $(x_i, y_i) \in X \times R$ be a data point representing an input and its corresponding output. Let U_i and L_i be the upper and lower bounds of the prediction interval corresponding to the ith sample. With this said, we want to build up intervals $[L_i, U_i]$ for every sample such that the probability $P(L_i \leq y_i \leq U_i) \geq 1-\alpha$ with $1-\alpha$ the confidence level of the prediction interval.

F. PROPOSED ARCHITECTURE

In this study, the problem involves predicting high dynamic air pollution using historical concentration data. We resolve the problem by proposing an adapted architecture based on ConvLSTM. The motivation behind this choice is to benefit from the pattern recognition of Convolution Networks and the memory properties of pure LSTM networks.

Concretely, our scheme consists of two modules: A feature extractor module and a prediction module. The former is constructed by stacking multiple ConvLSTM layers. Within these layers, the values of each cell's grid are determined by current and historical values of neighboring cells. That's what gives ConvLSTM layers the strong ability to mine spatio-temporal features between inputs. Each ConvLSTM block has five hidden states to extract features from historical data. Dropout and recurrent dropout are added in each one of them to prevent overfitting and batch normalization is applied between ConvLstm Layers to speed up the training process [44]. Finally, the output of the feature extraction's module is fed to a Conv2D with 1×1 filter (1×1 convolutional layer) to output the final estimation. This latter layer is also called a feature map pooling layer. It creates a one to one projection of the feature maps to pool features across channels whilst retaining their salient spatio-temporel features. Our model processes a sequence of grayscale pollution images as 2D input tensors with M$\times$N dimension. It outputs the estimated pollution image of the corresponding study area for the next timestamp. The pollution inputs are first converted to images before being processed by the neural network. This step could correspond to a simple projection if the input data are uniformly distributed in the space. Otherwise a more complex image construction could be done, for instance by aggregating data points. The model's structure is depicted in Figure 3.

It is worth mentioning that in our model's architecture, we have used variational dropout proposed by Gal and Ghahramani in [45]. The motivation behind this choice is that ConvLSTM and RNN in general are prone to overfitting, especially when the number of parameters in the network is large, and the amount of training data is small. Besides, variational dropout technique applies both dropout and recurrent-dropout inside the ConvLSTM cell-block. In this way, not only the weights of forward connections can be regularized, but also the weights of recursive connections can be as well.

More concretely, we regularize the subnetworks inside the cell block by dropping the gates input on both current x_t and previous hidden h_{t-1} states, using masks sampled at each timestep. As shown in the enlarged diagram of Figure 3 (ConvLSTM Cell Block, left), coloured connections represent different dropout masks applied to the gated architecture that are: input gate (i), forget gate (f), output gate (o) and input modulation gate (g). The equation that describes this technique is as follows:

$$\begin{pmatrix} i \\ f \\ o \\ g \end{pmatrix} = \begin{pmatrix} \sigma \\ \sigma \\ \sigma \\ tanh \end{pmatrix} \left(\begin{pmatrix} x_t \circ z_x \\ h_{t-1} \circ z_h \end{pmatrix} W \right) \tag{15}$$

where W is the weight matrix, z_x is the current state dropout mask and z_h represent the hidden state dropout mask.

Regarding the uncertainty estimations, our goal is to construct PIs for pollution forecasts considering different confidence levels. Both methods presented in section 3 were considered and applied to our forecasting model namely Monte Carlo Dropout and quantile regression. To perform prediction intervals with dropout, two hyper parameters should be taken into consideration: the dropout rate and the number of stochastic forward passes. We should also highlight the fact that in our network, two types of dropouts are used in every convLSTM layer (standard dropout and recurrent dropout), so globally we can choose which dropout to apply and in which layer of the model. For constructing uncertainty intervals with the quantile regression, the lower and upper bound of the PI is directly estimated through the network using the equation 13. Each time, two separate regression models are trained, one for the lower percentile and the other for the upper percentile to produce PIs following a confidence level.

IV. VALIDATION

For the effective evaluation, two trials from the Fusion Field Trial 2007 real data set were used [46]. The experimentation part is divided into two parts. In the former one, the objective is to predict one step ahead given an input sequence. Different tests were conducted to demonstrate the effectiveness of our forecasting model. The proposed model was compared to various state of the art techniques ranging from shallow machine learning algorithms to performing deep learning ones. It's worth to mention that all these latter methods are trained to forecast only one point at a time in contrast to our method which predict the output of all nodes simultaneously. In the second experimentation part, we investigate the differences between the MC dropout and quantile regression methods to quantify uncertainty. For the majority of the experiments and for both trials, we considered around 80% of the data for training and 20% for tests. Before going deeper in experimentation, we present the data sets used and the preprocessing made to the collected data. Next, we describe the evaluation measures and the baseline models considered in this work as well as the implementation details.

A. DATA COLLECTION AND PREPROCESSING

The Fusion Field Trial 2007 (FFT07) is a set of short range dispersion experiments that were conducted by the U.S Army's Dugway Proving Ground (DPG), Utah in September 2007, over a flat area of 450m × 475m [46]. Within this terrain, a total of 100 sensors were placed in a rectangular staggered grid at a height of 2 meters, uniformly so that the horizontal and vertical distance between each two sensors is 100 and 50 meters respectively (see Figure 4). The experiments involved both continuous and instantaneous releases from single and multiple sources (1, 2, 3, 4). For almost all the trials, the propylene C_3H_6 is released as a gas tracer continuously over a period of 10 min with a release rate of 377.5 L min^{-1}; the sensors record the concentrations at a frequency of 2 ms. The propylene was chosen as a pollution gas for its low toxicity. Meteorological measurements including wind speed, wind direction, temperature, relative humidity and pressure were taken from three ultrasonic tower with five levels (2m, 4m, 8m, 16m and 32 m). In the data set, the locations of the sources and the sensors are given in the latitude and longitude.

The present study utilizes only the observations measured during multiple continuous releases (three sources in trial 28 and four sources in trial 55) as data in other conditions are not available for us. In total the trial 28 and trial 55 contains 4910000 and 4405000 records respectively. various flags were provided to describe the status of the measurements. For both trials, the dysfunctional sensors were removed entirely (10 sensors for trial 28 and 7 sensors for trial 55) as shown in Figure 4 and Figure 5.

The corrupt and missing data were dropped and were not imputed. Indeed, we observed that the vast majority of these latter data were recorded towards the end of the experiment, and as the pollution plume at this point completely disappeared, the sensors logically measured zero concentration values. Finally the concentrations data were aggregated to a second scale.

Regarding meteorological data, to assign them into the grid cells, we first aggregate the data to a second scale and then we affected the values using the nearest neighbours method. Finally both pollution and meteorological data were normalized and scaled to [0,1] by the MinMaxScaler method and de-normalization were applied to the predicted data outputs.

To the best of our knowledge FFT07 is one of the rare real data set where punctual releases were observed and measured by dozens of sensors simultaneously, this makes this data set well suited to study high dynamic air pollution.

B. EVALUATION METRICS

To evaluate the models performances, we used Root Mean Squared Error (RMSE) and Mean Absolute Error(MAE). Their corresponding formulas are as follows:

$$RMSE = \sqrt{\frac{1}{n}\Sigma_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{16}$$

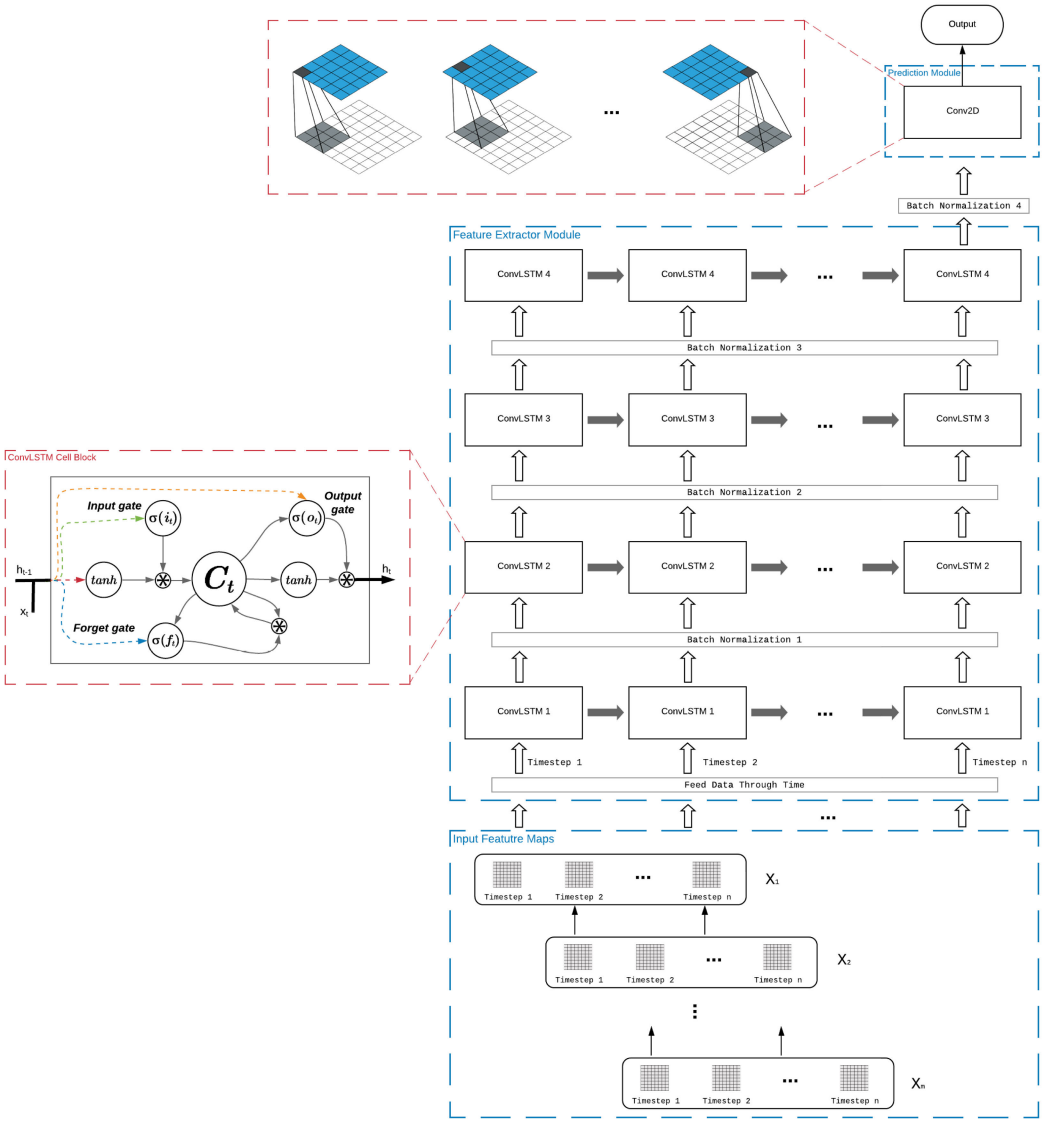

FIGURE 3. Illustration of the Uncertainty-Aware ConvLSTM model architecture, with four stacked ConvLSTM layers. Each vertical and horizontal arrow represents dropout and recurrent dropout input, respectively. The enlarged diagram on the left represents a single ConvLSTM cell block, with a representation of dropouts, where coloured connections correspond to different dropout masks.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|y_i - \hat{y}_i\right| \tag{17}$$

where y_i and $\hat{y}_i$ are the measured and predicted propylene concentration respectively, and n is the size of the test sample. Note that RMSE is scale-dependent and sensitive to outliers, which means that it can't be compared over data sets with different scale levels.

For uncertainty evaluation, we considered two known metrics for PIs: prediction interval coverage probability (PICP) and mean prediction interval width (MPIW).

Indeed, a good prediction interval should be as tight as possible but still have the desired coverage probability. Their mathematical equations are as follows [47]:

$$PICP = \frac{1}{n}\sum_{i=1}^{n} c_i \; with \begin{cases} c_i = 1, & \text{if } y_i \in [L_i, U_i]. \\ c_i = 0, & \text{else.} \end{cases} \tag{18}$$

$$MPIW = \frac{1}{n}\sum_{i=1}^{n}(U_i - L_i) \tag{19}$$

y_i denotes the ground truth, L_i and U_i represent the lower and upper bound respectively.

C. BASELINE MODELS

Different baseline models were considered to evaluate and compare the performances of our model. The baselines

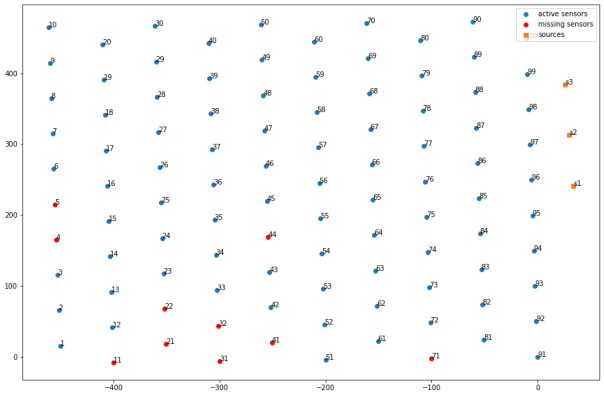

FIGURE 4. FFT grid for trial 28.

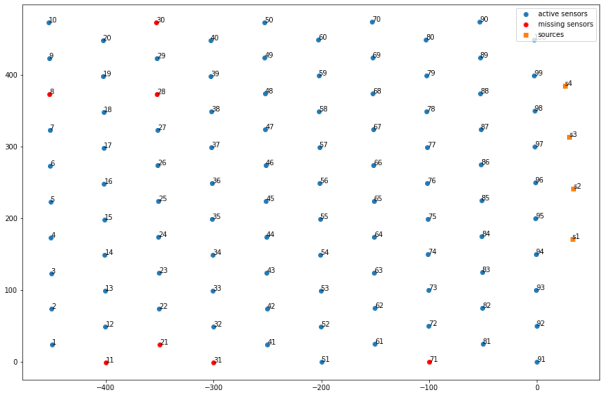

FIGURE 5. FFT grid for trial 55.

included support vector regression, multiple layer perceptron, convolutional neural network, two variants of LSTM and GRU. In what follows we describe briefly these methods.

SVR is a non parametric machine learning method based on kernels. It was introduced for the first time by [48] and it is often regarded as a basic technique used for comparison purposes with more complex models.

The MLP can be considered as a precursor of a large set of neural networks. LSTM and GRU belong to Recurrent Neural Networks and has both proven their ability to process sequential data and extract relevant temporal features. More precisely, a stacked LSTM was considered in this work to enable more complex representation of the time series data. As for CNN, it is generally a good tool to capture the spatial relations present in input data.

For LSTM and GRU models, we implemented two versions for both of them: a univariate and a multivariate version. For multivariate models, only the wind speed and the wind direction were considered as explanatory variables. The other variables as temperature, humidity and pressure were not taken into consideration as they don't change much over time and hence they don't provide additional information to the model.

D. IMPLEMENTATION DETAILS

The open source deep learning library Keras [49] with TensorFlow backend [50] are used to build up our forecasting model and the other deep learning baseline models. The sickit-learn python library was used to construct shallow learning methods.

For all the experiments, the time horizon l was set to 5. The loss function used for all DNNs models is the mean square error (MSE) loss function. Adam [51] was used as optimizer with a learning rate of 0.00005. This value was selected from the set {0.00001, 0.00005, 0.0001, 0.0005, 0.001, 0.05, 0.1}. All the weights in the Deep Learning models were initialized according to the Xavier uniform initialization [52]. The batch size was set to three and the number of epoch to 60. The parameters were chosen following a grid search in the parameters space.

E. PROOF OF CONCEPT

In this experiment, we aim to verify the feasibility of our forecasting model when considering two aspects: the temporal and the spatial stability. First, we tested the temporal stability to get an idea of the model performances over time. In Figure 6 and Figure 7, we depicted the predicted values versus the recorded values of propylene for 16 randomly selected sensors on test data for both trials. As noticed, our model shows consistent performance for almost all nodes. Indeed, we observe that the model manages to simultaneously follow the general trend of the different sensors and can follow the fluctuations of actual values during the test set successfully. It is also able to detect the sudden changes most of the time. However, we observed that the model underestimates the peaks a bit.

Next, we evaluated the spatial stability. Mean absolute error was computed for every sensor and then averaged over time. The results are depicted in Figure 8 and Figure 9. We observe that our forecasting model showed different predictive performances for the different nodes. In detail, the MAE varied from 0.00085 to 43.5 for the trial 28 and from 0.00065 to 10.05 for trial 55, the RMSE varied from 0.0016 to 58.135 for trial 28 and from 0.0010 to 24.07 for trial 55. The highest errors were observed for the sensors the closest to the sources (sensor 96 for trial 28 and sensor 97 for trial 55). These latter recorded larger concentration values, resulting in more important errors. To investigate further the impact of the sources proximity on the prediction errors, we plotted the MAE in function of the euclidean distance that separate the sensors from the sources (Figure 10 and Figure 11). It is obvious that the further the sensors are from the sources, the smaller the errors and vice versa. In fact, the sensors close to the sources are the first exposed to the pollution plume, and this latter moves in the grid according to the direction and speed of the wind (from the right to the left in the x axis) and as our model can predict the next sensor concentrations from their neighboring sensors, the prediction error becomes smaller and smaller for nodes far from the sources.

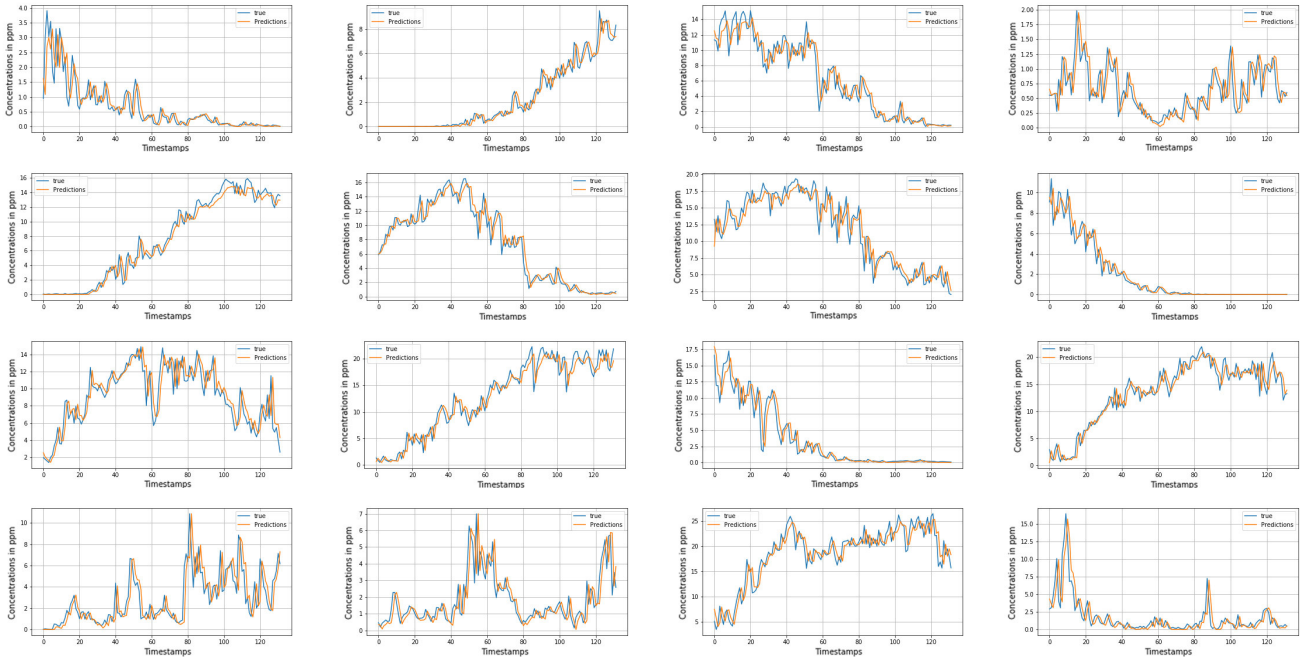

FIGURE 6. Forecasting performance on 16 stations for trial 28 data set. Blue lines represent real values and orange ones are forecast values. Horizontal axis shows timestamps and vertical axis shows concentration pollution of propylene (ppm).

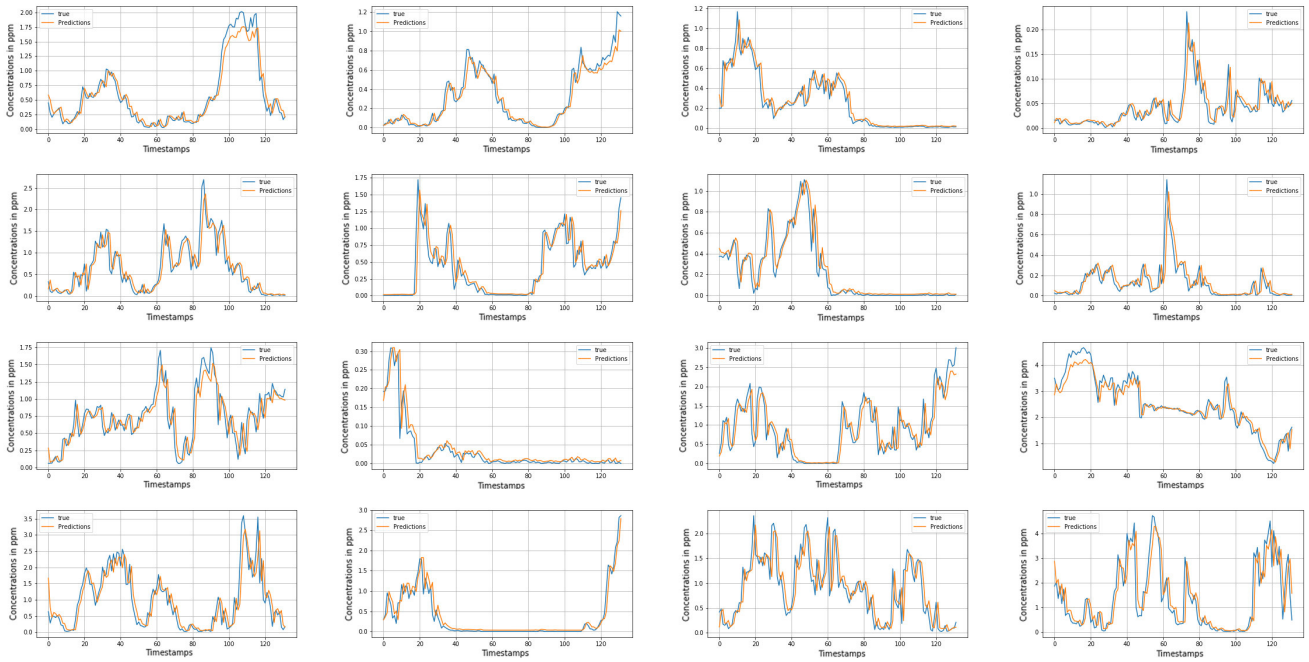

FIGURE 7. Forecasting performance on 16 stations for trial 55 data set. Blue lines represent real values and orange ones are forecast values. Horizontal axis shows timestamps and vertical axis shows concentration pollution of propylene (ppm).

F. EVALUATION OF THE FORECASTING MODEL

This section compares our forecasting model with 7 state of the art methods. We evaluated the prediction performances of these models for all the nodes (90 nodes for trial 28 and 93 nodes for trial 55). It's worth to mention that all the baseline models considered in this work are trained to forecast one sensor at a time in the contrast to our model which can predict the whole grid in one step. The results for all sensors were averaged and depicted in Figure 12, the quantitative results for the two trials are reported in Table 2, which give RMSE, MAE and processing time comparative analysis of all the considered models.

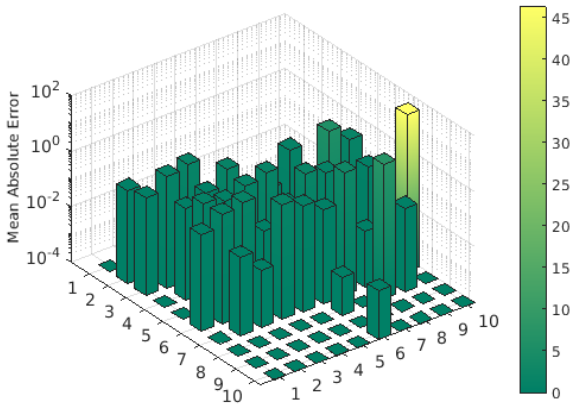

FIGURE 8. Mean Absolute Error for all the sensors for trial 28.

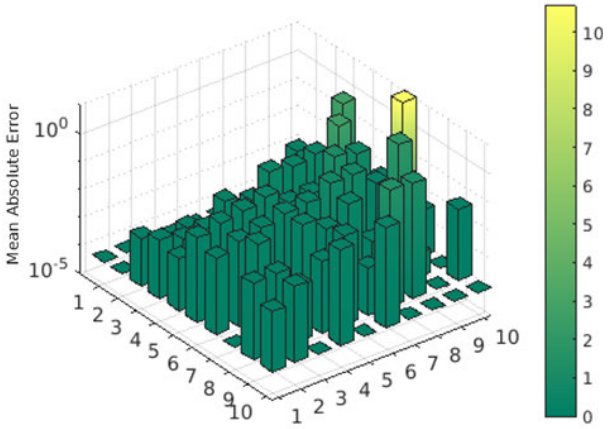

FIGURE 9. Mean Absolute Error for all the sensors for trial 55.

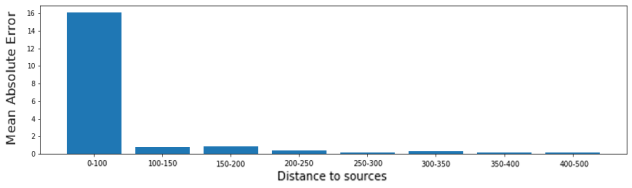

FIGURE 10. Mean Absolute Error in function of distance to sources for trial 28.

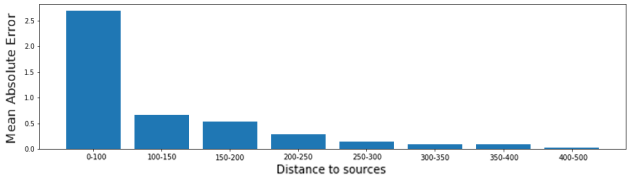

FIGURE 11. Mean Absolute Error in function of distance to sources for trial 55.

According to Figure 12 and Table 2, our model achieves the best performance on both trials (trial 28 and trial 55) in terms of MAE. Indeed, it can learn efficiently the useful information for every node and anticipate their behaviour pretty well. In addition, we observe that classical recurrent

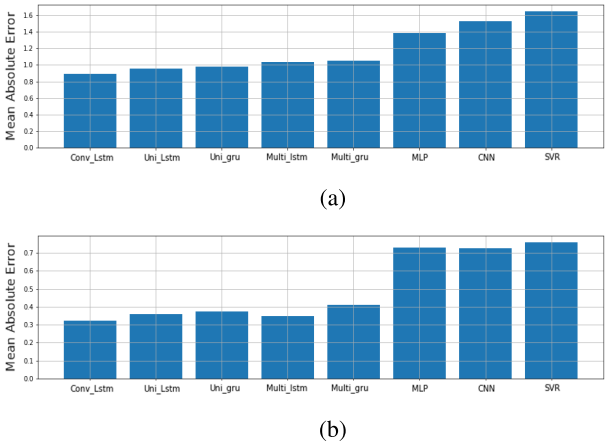

FIGURE 12. Mean Absolute Error in function of the implemented methods using (a): trial 28 and (b): trial 55.

deep learning models as LSTM and GRU performs much better than other shallow models (CNN, MLP and SVR). The univariate LSTM and univariate GRU obtained scores pretty close to our model's scores. However, to achieve such performance with these latter models we have to train as many models as there are nodes, which result on an important processing time as shown in Table 2. Moreover, following Table 2, one can notice that in terms of RMSE our model isn't the best for trial 28 but still obtain a score quite close to LSTM and GRU models.

In order to further investigate the differences between the models, we evaluate the impact of the training size on the prediction accuracy. We considered various training sizes ranging from 2 to 10 minutes. From the results in Figure 13 and 14, it is shown that with the increase of the training size the prediction errors becomes smaller for all the models. Compared to the baseline models, our model performs better for almost all the considered training sizes (except for 8 minutes) in terms of MAE and RMSE. Our model is followed by the univariate LSTM and GRU models which mostly give acceptable results and outperform their multivariate versions. This means that the meteorological inputs were not relevant for LSTM and GRU models in our case study. For training sizes greater than 8 minutes, we notice that models based on LSTM and GRU maintains performances close to our model. For example, when the training size is set to 10 min, our model reduces the MAE by 0.1 compared to the univariate LSTM. In contrast, for small training sizes the difference in performance is more significant (our model performs 58% in terms of MAE better than multivariate LSTM for a training size of 2min). Finally, we should highlight that almost all deep learning baseline models are computationally expensive as they require many models to train to predict all the nodes.

G. UNCERTAINTY COMPARISON

In this section the performance of MC dropout and quantile regression methods are quantitatively assessed on real data.

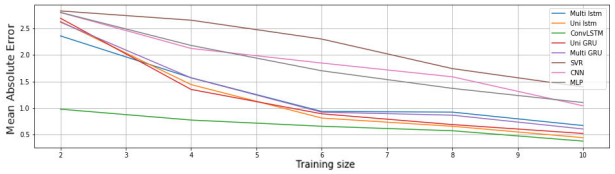

FIGURE 13. Mean Absolute Error in function of training size for trial 28.

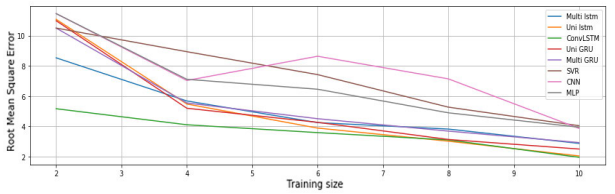

FIGURE 14. Root Mean Square Error in function of training for trial 28.

TABLE 2. Models performance for predicting C_3H_6 of all nodes averaged. Optimal scores are given in bold.

SVR	1.648	6.103	0.759	2.602	0.006	0.001
CNN	1.531	6.758	0.725	3.186	1.321	1.401
MLP	1.387	5.565	0.729	3.028	6.887	6.917
Uni LSTM	0.955	5.028	0.360	1.730	15.230	15.870
Multi LSTM	1.036	5.263	0.346	1.742	15.350	16.007
Uni GRU	0.981	**5.015**	0.374	1.813	15.750	15.850
Multi GRU	1.051	5.224	0.410	2.145	15.430	16.368
Our model	**0.896**	5.284	**0.326**	**1.712**	2.297	2.434

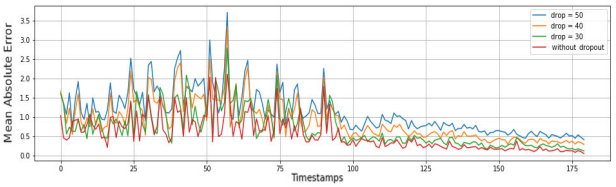

FIGURE 15. Mean Absolute Error over time considering both cases: when enabling and disabling dropout during the test phase for trial 28.

The uncertainty estimation was performed on our forecasting model (Figure 3) using PIs. The number of stochastic forward pass was set to 200 for all the experiments. As recommended in [39], this number could be set to 100 or greater.

As a first step, we investigate the prediction quality obtained when enabling dropout in the inference phase for both trials. To that end, we considered different dropout rates $r \in \{0.3, 0.4, 0.5\}$ and we computed for each rate 200 stochastic forward passes that we averaged to get the mean prediction. The resulting prediction errors were compared to the results obtained with our model using settings in the previous section (optimal dropouts, disabled dropout in test phase). As shown in Figure 15 and Figure 16, forecast performance is obviously impacted by the dropout rate. Indeed, larger dropouts lead to larger prediction errors and on the contrary, the smaller the dropouts, the smaller the errors. This can be explained as follows: large dropouts lead to large variances in predictions, which make the model unstable.

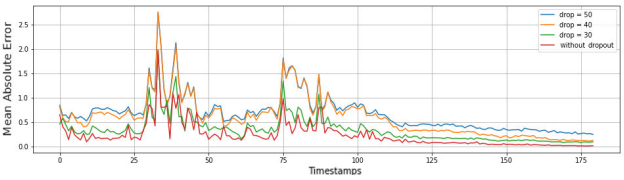

FIGURE 16. Mean Absolute Error over time considering both cases: when enabling and disabling dropout during the test phase for trial 55.

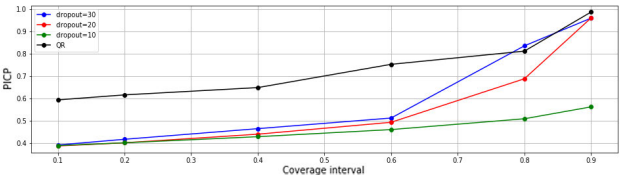

FIGURE 17. PICP comparison in function of the coverage interval for the quantile regression and MC dropout considering three rate for dropout: 0.1, 0.2, 0.3.

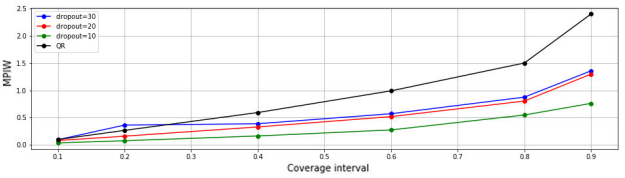

FIGURE 18. MPIW comparison in function of the coverage interval for the quantile regression and MC dropout considering three rate for dropout: 0.1, 0.2, 0.3.

As a second step, we inspected the quality of the generated PIs using QR and MC dropout considering different coverage intervals ranging from 10% to 90% and different dropout rates $r \in \{0.1, 0.2, 0.3\}$. The PICP and MPIW were computed and results were depicted in Figure 17 and Figure 18. Ideally, we would like the PICP to be as large as possible and the MPIW as small as possible. As shown in Figure 17 and Figure 18, increasing the dropout rate causes an increase in both PICP and MPIW. The larger the number of neurons being dropped out of the network, the higher the variance is in the output. This would yield to larger average intervals that would capture our output values. In addition, we observe that even with low coverage rates, the MC dropout still provide satisfactory results, for example, considering a coverage interval of 20% and a dropout rate of 0.3, lead to a PI that brackets around 43% of the target values.

For MC dropout method, the PICP varied from 0.38 to 0.96 and the MPIW from 0.037 to 1.3 while for QR technique the PICP fluctuates from 0.59 to 0.989 and the MPIW from 0.09 to 2.4. Comparing to MC dropout, the QR method obtained the highest PICP for almost all the coverage intervals. However the MPIW values of this latter is also greater than those of MC dropout method whatever the dropout rate considered. Concretely, this means that QR produces larger PIs that captures more real observations within it. In addition, the results shows that MC dropout can generate tight PIs that

are close to those of QR in terms of coverage. For instance, the PI obtained using MC dropout considering a dropout rate of 0.3 and a coverage rate of 90% captures 96% of measurements with a width of 1.3 while QR's PIs brackets 98% of the observations with a width equal to 2.4.

In conclusion, the results have shown that with both methods it is possible to generate quite tight intervals with a majority of observed values within this interval and with minor changes in the architecture. Moreover, when considering low coverage intervals (less than 60%) QR performs better, it obtains PIs with higher PICP and MPIW quite close to the ones of MC dropout. In the contrary, for higher coverage intervals (greater than 60%), the PIs produced by MC dropout are better in terms of coverage/width compromise. Nonetheless, one obvious inconvenient of MC dropout is that it can be computationally expensive (when the number of stochastic forward is elevated) which makes it not well adapted for real time applications as monitoring dynamic pollution in emergency situations, in the contrast of QR that gives an estimate of uncertainty from single prediction.

V. CONCLUSION

In this paper, we have tackled the highly dynamic air-pollution forecasting problem. First, we comprehensively reviewed state of the art methods for air pollution prediction where many top-level articles were discussed and compared. Second, we presented a spatio-temporal deep learning model based on ConvLSTM for high dynamic air pollution prediction, as the case of sudden pollution events. The developed model is able to forecast the output of several nodes at the same time using only one framework. Moreover, it learns which nodes are more important to forecast one specific node without any prior knowledge on the relationships between nodes. Third, we compared our model's performance to seven state of the art methods ranging from shallow machine learning techniques to conventional deep learning ones. The tests were performed on two real world high dynamic data sets. The experiment results has demonstrated the superiority of our forecasting model over the baseline techniques and has proven its effectiveness for short-term air pollution forecasting whatever the training size considered. Lastly, two techniques namely MC dropout and quantile regression were used to compute uncertainty estimates on the top of our forecasting model. Results have shown that the quantile regression technique outperforms MC dropout for low coverage intervals, offering better prediction intervals and in the contrast, the MC dropout technique is better for high coverage intervals. Moreover, it has been shown that the performance of MC dropout highly depends on dropout rates. Hence this latter has to be carefully chosen.

ACKNOWLEDGMENT

The authors would like to thank M. Eugene Yee for his precious help, particularly in handling the FFT-07 data set.

REFERENCES

[1] Word Health Organization. *9 Out of 10 People Worldwide Breathe Polluted Air, But More Countries Are Taking Action*. [Online]. Available: https://www.who.int/health-topics/air-pollution#tab=tab_1

[2] T. Hyttinen, "Service concepts based on unmanned aerial vehicle systems," *J. Operations Manage.*, vol. 20, no. 2, pp. 135–157, 2002.

[3] S. Du, T. Li, Y. Yang, and S.-J. Horng, "Deep air quality forecasting using hybrid deep learning framework," *IEEE Trans. Knowl. Data Eng.*, early access, Nov. 20, 2020 doi: 10.1109/TKDE.2019.2954510.

[4] V.-D. Le, T.-C. Bui, and S.-K. Cha, "Spatiotemporal deep learning model for citywide air pollution interpolation and prediction," in *Proc. IEEE Int. Conf. Big Data Smart Comput. (BigComp)*, Feb. 2020, pp. 55–62.

[5] J. Wang and G. Song, "A deep spatial-temporal ensemble model for air quality prediction," *Neurocomputing*, vol. 314, pp. 198–206, Nov. 2018.

[6] L. Soulhac, P. Salizzoni, F.-X. Cierco, and R. Perkins, "The model SIRANE for atmospheric urban pollutant dispersion; part I, presentation of the model," *Atmos. Environ.*, vol. 45, no. 39, pp. 7379–7395, Dec. 2011.

[7] P. Thunis, A. Miranda, J. M. Baldasano, N. Blond, J. Douros, A. Graff, S. Janssen, K. Juda-Rezler, N. Karvosenoja, G. Maffeis, A. Martilli, M. Rasoloharimahefa, E. Real, P. Viaene, M. Volta, and L. White, "Overview of current regional and local scale air quality modelling practices: Assessment and planning tools in the EU," *Environ. Sci. Policy*, vol. 65, pp. 13–21, Nov. 2016.

[8] L. Menut, B. Bessagnet, D. Khvorostyanov, M. Beekmann, N. Blond, A. Colette, I. Coll, G. Curci, G. Foret, A. Hodzic, S. Mailler, F. Meleux, J.-L. Monge, I. Pison, G. Siour, S. Turquety, M. Valari, R. Vautard, and M. G. Vivanco, "CHIMERE 2013: A model for regional atmospheric composition modelling," *Geosci. Mod. Develop.*, vol. 6, no. 4, pp. 981–1028, Jul. 2013.

[9] *Comprehensive Air Quality Model With Extensions (CAMX) Version 4.50*, Environ Int. Corp., Novato, CA, USA, 2008.

[10] D. Carruthers, H. Edmunds, C. McHugh, and R. Singles, "Development of adms-urban and comparison with data for urban areas in the UK," in *Air Pollution Modeling and its Application*. Cham, Switzerland: Springer, 199, pp. 467–475.

[11] L. Soulhac, G. Lamaison, F.-X. Cierco, N. Ben Salem, P. Salizzoni, P. Mejean, P. Armand, and L. Patryl, "SIRANERISK: Modelling dispersion of steady and unsteady pollutant releases in the urban canopy," *Atmos. Environ.*, vol. 140, pp. 242–260, Sep. 2016.

[12] M. Lei, L. Shiyan, J. Chuanwen, L. Hongling, and Z. Yan, "A review on the forecasting of wind speed and generated power," *Renew. Sustain. Energy Rev.*, vol. 13, no. 4, pp. 915–920, May 2009.

[13] P. Costabile and F. Macchione, "Enhancing river model set-up for 2-D dynamic flood modelling," *Environ. Model. Softw.*, vol. 67, pp. 89–107, May 2015.

[14] P. Zannetti, *Air Pollution Modeling: Theories, Computational Methods and Available Software*. Cham, Switzerland: Springer, 2013.

[15] E. Aristodemou, R. Arcucci, L. Mottet, A. Robins, C. Pain, and Y.-K. Guo, "Enhancing CFD-LES air pollution prediction accuracy using data assimilation," *Building Environ.*, vol. 165, Nov. 2019, Art. no. 106383.

[16] B. Denby, M. Schaap, A. Segers, P. Builtjes, and J. Horálek, "Comparison of two data assimilation methods for assessing PM10 exceedances on the European scale," *Atmos. Environ.*, vol. 42, no. 30, pp. 7122–7134, Sep. 2008.

[17] U. Kumar and V. K. Jain, "ARIMA forecasting of ambient air pollutants (O_3, NO, NO_2 and CO)," *Stochastic Environ. Res. Risk Assessment*, vol. 24, no. 5, pp. 751–760, Jul. 2010.

[18] F. Dominici, "On the use of generalized additive models in time-series studies of air pollution and health," *Amer. J. Epidemiol.*, vol. 156, no. 3, pp. 193–203, Aug. 2002.

[19] L. C. Ochando, C. I. F. Julián, F. C. Ochando, and C. F. Ramirez, "Airvlc: An application for real-time forecasting urban air pollution," in *Proc. ICML*, 2015, pp. 1–5.

[20] S. M. Cabaneros, J. K. Calautit, and B. R. Hughes, "A review of artificial neural network models for ambient air pollution prediction," *Environ. Model. Softw.*, vol. 119, pp. 285–304, Sep. 2019.

[21] B. Liu, S. Yan, J. Li, G. Qu, Y. Li, J. Lang, and R. Gu, "A sequence-to-sequence air quality predictor based on the n-step recurrent prediction," *IEEE Access*, vol. 7, pp. 43331–43345, 2019.

[22] Q. Tao, F. Liu, Y. Li, and D. Sidorov, "Air pollution forecasting using a deep learning model based on 1D convnets and bidirectional GRU," *IEEE Access*, vol. 7, pp. 76690–76698, 2019.

[23] M. Yang, H. Fan, and K. Zhao, "PM2.5 prediction with a novel multi-step-ahead forecasting model based on dynamic wind field distance," *Int. J. Environ. Res. Public Health*, vol. 16, no. 22, p. 4482, Nov. 2019.
[24] J. Ma, J. C. P. Cheng, C. Lin, Y. Tan, and J. Zhang, "Improving air quality prediction accuracy at larger temporal resolutions using deep learning and transfer learning techniques," *Atmos. Environ.*, vol. 214, Oct. 2019, Art. no. 116885.
[25] M. A. Fekih, I. Mokhtari, W. Bechkit, Y. Belbaki, and H. Rivano, "On the regression and assimilation for air quality mapping using dense low-cost wsn," in *Proc. Int. Conf. Adv. Inf. Netw. Appl.* Cham, Switzerland: Springer, 2020, pp. 566–578.
[26] W. Sun and J. Sun, "Daily PM 2.5 concentration prediction based on principal component analysis and LSSVM optimized by cuckoo search algorithm," *J. Environ. Manage.*, vol. 188, pp. 144–152, Mar. 2017.
[27] C.-J. Huang and P.-H. Kuo, "A deep CNN-LSTM model for particulate matter (PM2.5) forecasting in smart cities," *Sensors*, vol. 18, no. 7, p. 2220, Jul. 2018.
[28] P.-W. Soh, J.-W. Chang, and J.-W. Huang, "Adaptive deep learning-based air quality prediction model using the most relevant spatial-temporal relations," *IEEE Access*, vol. 6, pp. 38186–38199, 2018.
[29] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
[30] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.
[31] Y. LeCun, F. Jie Huang, and L. Bottou, "Learning methods for generic object recognition with invariance to pose and lighting," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, 2004, pp. 1–5.
[32] S. Lawrence, C. L. Giles, A. Chung Tsoi, and A. D. Back, "Face recognition: A convolutional neural-network approach," *IEEE Trans. Neural Netw.*, vol. 8, no. 1, pp. 98–113, Jan. 1997.
[33] C. Zhang, J. Yan, C. Li, X. Rui, L. Liu, and R. Bie, "On estimating air pollution from photos using convolutional neural network," in *Proc. 24th ACM Int. Conf. Multimedia*, Oct. 2016, pp. 297–301.
[34] S. Li, G. Xie, J. Ren, L. Guo, Y. Yang, and X. Xu, "Urban PM2.5 concentration prediction via attention-based CNN–LSTM," *Appl. Sci.*, vol. 10, no. 6, p. 1953, Mar. 2020.
[35] S. Xingjian, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-C. Woo, "Convolutional LSTM network: A machine learning approach for precipitation nowcasting," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 802–810.
[36] Z. Yuan, X. Zhou, and T. Yang, "Hetero-ConvLSTM: A deep learning approach to traffic accident prediction on heterogeneous spatio-temporal data," in *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Jul. 2018, pp. 984–992.
[37] A. D. Kiureghian and O. Ditlevsen, "Aleatory or epistemic does it matter?" *Struct. Saf.*, vol. 31, no. 2, pp. 105–112, 2009.
[38] D. J. C. MacKay, "A practical Bayesian framework for back propagation networks," *Neural Comput.*, vol. 4, no. 3, pp. 448–472, May 1992.
[39] Y. Gal and Z. Ghahramani, "Dropout as a Bayesian approximation: Representing model uncertainty in deep learning," in *Proc. Int. Conf. Mach. Learn.*, 2016, pp. 1050–1059.
[40] R. Tibshirani, "A comparison of some error estimates for neural network models," *Neural Comput.*, vol. 8, no. 1, pp. 152–163, Jan. 1996.
[41] R. Koenker, "Regression quantiles," *Econometrica, J. Econ. Soc.*, vol. 4, pp. 33–50, Dec. 1978.
[42] D. A. Nix and A. S. Weigend, "Learning local error bars for nonlinear regression," in *Proc. Adv. Neural Inf. Process. Syst.*, 1995, pp. 489–496.
[43] B. Lakshminarayanan, A. Pritzel, and C. Blundell, "Simple and scalable predictive uncertainty estimation using deep ensembles," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 6402–6413.
[44] I. Goodfellow, Y. Bengio, and A. Courville, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Conf. Comput. Sci.*, 2016, pp. 173–177.
[45] Y. Gal and Z. Ghahramani, "A theoretically grounded application of dropout in recurrent neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2016, pp. 1019–1027.
[46] D. Storwold, *Detailed test plan for the fusing sensor information from observing networks Ground*, document WDTC-TP-07-078, 2007.
[47] A. Khosravi, S. Nahavandi, D. Creighton, and A. F. Atiya, "Lower upper bound estimation method for construction of neural network-based prediction intervals," *IEEE Trans. Neural Netw.*, vol. 22, no. 3, pp. 337–346, Mar. 2011.
[48] V. N. Vapnik, *The Nature of Statistical Learning Theory*. Berlin, Germany: Springer-Verlag, 1995.
[49] F. Chollet. *keras. Github Repository*. Accessed: May 25, 2017. [Online]. Available: https://github.com/fchollet/keras
[50] M. Abadi *et al.*, "TensorFlow: Large-scale machine learning on heterogeneous distributed systems," 2016, *arXiv:1603.04467*. [Online]. Available: http://arxiv.org/abs/1603.04467
[51] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*. [Online]. Available: http://arxiv.org/abs/1412.6980
[52] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proc. 13th Int. Conf. Artif. Intell. Statist.*, 2010, pp. 249–256.

ICHRAK MOKHTARI received the master's and Engineering degrees in computer science from the École Nationale Supérieure d'informatique, Algiers, in 2019. She is currently pursuing the Ph.D. degree with INSA-Lyon/INRIA, France. She is also a member of the INRIA Agora Team, CITI Laboratory. Her main research interests include machine and deep learning, optimization, UAVs deployment, and path planning.

WALID BECHKIT received the Engineering degree in computer science from the École Nationale Supérieure d'Informatique, Algiers, in 2009, and the Ph.D. degree in system and information technology from the Compiegne University of Technology (UTC), France, in 2012. He is currently an Associate Professor with the Telecommunications Department, INSA-Lyon, France. He is also a member of the INRIA Agora Team, CITI Laboratory. His main research interests include wireless sensor networks, the IoT for smart cities applications, air quality monitoring, data analysis, machine and deep learning, LPWAN technologies, and security.

HERVÉ RIVANO graduated from the Ecole Normale Supérieure de Lyon, France. He received the Ph.D. degree from the University of Nice-Sophia Antipolis, in November 2003. He was an INRIA and CNRS Researcher, from October 2004 to August 2017. He is currently a Full Professor with INSA-Lyon, France, and the Head of the INRIA Agora Team, which focuses on the use of wireless networks in smart cities. His research interests include combinatorial optimization and approximation algorithms applied to network design and provisioning.

MOULOUD RIADH YAICI is currently pursuing the Engineering degree and the master's degree in computer science from the École Nationale Supérieure d'informatique, Algiers. He is currently a Research Intern with INRIA and a member of the INRIA Agora Team, CITI Laboratory. His research interests include machine learning and combinatorial optimization problems.