IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 3 May 2023, accepted 29 May 2023, date of publication 2 June 2023, date of current version 26 July 2023.
Digital Object Identifier 10.1109/ACCESS.2023.3282545

SURVEY

Spatio-Temporal Forecasting: A Survey of Data-Driven Models Using Exogenous Data

SAFAA BERKANI[1], BASSMA GUERMAH[1], MEHDI ZAKROUM[1,2], (Member, IEEE), AND MOUNIR GHOGHO[1,3], (Fellow, IEEE)
[1]TICLab Research Laboratory, International University of Rabat, Rabat 11103, Morocco
[2]CNRS, Inria, LORIA, Universite de Lorraine, 54000 Nancy, France
[3]School of Electronic and Electrical Engineering, University of Leeds, LS2 9JT Leeds, U.K.

Corresponding author: Safaa Berkani (safaa.berkani@uir.ac.ma)

ABSTRACT Forecasting Spatio-Temporal processes has been attracting a great deal of interest within the research community. In this context, there is an increasing trend of proposing and improving methodologies to gather and use vast amounts of Spatio-Temporal data. Spatio Temporal Forecasting (STF) problems present complex interactions and non-linearities as the temporal dimension and the spatial one are usually entangled. To address these problems, statistical, Machine Learning based, and Deep Learning based models are introduced and developed. The use of exogenous data has proven to be beneficial in many STF models. Various techniques of incorporating exogenous data in STF problems have been proposed in the literature. This survey aims at providing a systematic review of the data-driven STF models, with a focus on those that incorporate exogenous data. We first investigate the data properties, including their dynamics, types and representations. Next, we propose a new taxonomy of the reviewed models and inspect the different complexities of STF problems. Exogenous data incorporation techniques are then presented and analyzed. We conclude our paper by highlighting the current open challenges and future research directions.

INDEX TERMS Exogenous data, deep learning, forecasting, machine learning, spatio-temporal data, statistics.

I. INTRODUCTION

Over time, scholars have attempted to understand our world. By observing, describing, modeling and analyzing the different phenomena, it is not only possible to comprehend the world, but also reach a certain level of control over it. This has been facilitated thanks to the increasing availability of data and computation resources.

As the number of sensors and connected objects has exponentially increased in the last few years, data in their different formats are gathered in massive amounts, offering the opportunity to generate insight. Time series, for example, encompass information about when each data element was collected. Spatial data on the other hand comprise information about where each data element was collected. Spatio-Temporal (ST) data involve both types of information. In this survey, we focus on the latter.

The associate editor coordinating the review of this manuscript and approving it for publication was Vicente Alarcon-Aquino.

Spatio-temporal series refer to data that exhibit fluctuations in both the spatial and temporal dimensions. These series can be seen as sequences of objects described in a multidimensional space, where there is a dependence between the represented objects. It is crucial to note that throughout the remainder of the paper, ''location'' does not necessarily refer to the physical position of the described data instance. Rather, it refers to the data representation in the given multidimensional feature space. For instance, it might refer to different road segments, neighborhoods, users' identifiers (IDs), or networks' IDs, depending on the specific application in question.

Extracting valuable knowledge from ST data has become very important in addressing various problems such as classification, clustering, kriging, anomaly detection, and forecasting. However, the forecasting task has attracted the most interest among researchers [1]. It consists of building a data model that explains the spatio-temporal variability of data with the aim of predicting future data values. Spatio-Temporal

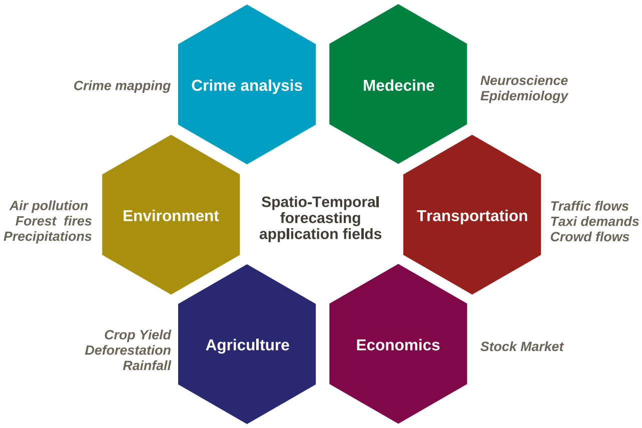


FIGURE 1. Examples of applications for Spatio-Temporal forecasting.

Forecasting (STF) plays a key role in a wide range of domains of application, including but not limited to, agriculture [2], [3], crime analysis [4], [5], transportation [6], [7], environment [8], [9], economics [10], [11], and medicine [12], [13]. Fig. 1 depicts some of these fields.

However, STF is still a challenging issue due to multiple factors. Indeed, ST data have some complex characteristics such as the non-identical distribution of data across space (**heterogeneity**), across time (**non-stationarity**) and the **non linear** dependencies within data [14]. Other challenges include:

- **Complexity:** since ST data fluctuate along both the spatial and temporal dimensions, the patterns within them are complex, implicit and challenging to capture.
- **Dynamicity:** in most real-life processes, the spatial dependencies are dynamic, meaning that they change through time. This entanglement between dimensions implies that they should be modeled simultaneously rather than individually.
- **Interdisciplinary:** this challenge may be encountered when integrating exogenous data in the forecasting models, since such data may come from domains that are very different from that of the endogenous data. Therefore, knowledge and techniques from various fields may be required.

Numerous data-driven models for ST forecasting have been introduced and developed in the literature. These models aim at finding relationships between the entangled factors in the data by understanding the underlying substructures within the data. They are characterized by their ability of using historical evidence to predict spatio-temporal dynamics. They can be classified into three categories: statistical models, conventional Machine Learning (ML) models and Deep Learning (DL)-based models.

Forecasting may sometimes greatly benefit from external features, formally called exogenous variables, that are correlated with the endogenous variables. For example, the urban traffic flow data can be significantly affected by some external factors such as weather, social events, and holidays. Several studies have explored ways of integrating exogenous data into forecasting models [15]. We differentiate between two categories of spatio-temporal forecasting models in the literature: explanatory models and predictive ones. Explanatory models leverage the explicit temporal and spatial characteristics embodied in the exogenous variables, and are therefore highly interpretable. During the modeling phase, endogenous data autocorrelations are not considered. For example, Bansak et al. and Kang et al. predict the refugees' employment probabilities in a specific time and region based on the characteristics of these individuals rather than the historical spatial information [16], [17]. On the other hand, a predictive model focuses more on the spatial and temporal lags of endogenous data, from which ST properties are derived, with the possibility of incorporating exogenous variables. The scope of this survey is limited to predictive models, with an emphasis on the incorporation of exogenous data.

A. RELATED SURVEYS AND CONTRIBUTIONS

A few relevant surveys have examined Spatio-Temporal Forecasting. Each of these surveys tackle the literature from a different perspective, as described next.

Hamdi et al. [14] and Atluri et al. [18] both outline the challenges that researchers face when mining information from ST data. They categorize these challenges according to different aspects including the data properties, the tackled task and the application field. Although both of these surveys are relatively recent and do discuss the STF problem, their scope is broad. Since they cover the different tasks and problems included in the Spatio-Temporal Data Mining (STDM) field, they do not give sufficient details on the ST forecasting task and its related models.

Providing more model-related details, Wang et al. reviewed recent progress, up to 2019, in applying DL techniques to address several STDM tasks in different domains [1]. They introduced a framework outlining the employment of these models and categorized the existing literature based on the types of ST data, the mining tasks and the employed DL models. Shi et al. on the other hand, led a systematic review for ML applications for Spatio-Temporal Sequence Forecasting (STSF) [19]. Both surveys concluded by pinpointing the limitations of the reviewed work and suggesting future research directions. These surveys emphasize the forecasting task, but none of them covers the three classes of data-driven STF models mentioned earlier. Indeed, the former, along with [20] and [21] only included the Deep Learning based models, while the latter focused on the conventional Machine Learning-based models.

Several surveys covered the three classes of data-driven models: Ermagun and Levinson [22], Boukerche and Wang [23] and Yuan and Li [24] reviewed the models employed in traffic forecasting; Xie et al. [25] covered the models used in urban flow prediction; and Tascikaraoglu [26] reviewed the models exploited in smart city applications. However, these surveys solely focus on specific application domains, and thus did not include the models that were not used in the addressed field of application. To comprehensively review the state-of-the-art data-driven models, it is crucial not to limit the research investigation to a specific application field.

Xu et al. reviewed statistical, Machine Learning, and Deep Learning based models. However, little attention was paid to the incorporation of exogenous data [27].

Our survey paper stands out from others in the field by providing a thorough examination of the STF pipeline. We begin by analyzing the data types, dynamics, and representations, and offer examples of exogenous variables used in the literature. This lays the foundation for the study. Next, we discuss various problem formulations and provide an up-to-date overview of state-of-the-art models. We offer insights into the strengths, weaknesses, and selection criteria of these models. Unlike previous work, our study is not limited to specific fields or model categories. We then shed light on the exogenous variables incorporation phase, by introducing a novel full-view taxonomy that organizes and synthesizes existing incorporation approaches and levels. We also identify open challenges and future research directions to stimulate further research in this area.

Overall, the main contributions of our study are summarized as follows:

- We provide a thorough examination of data properties.
- We categorize the different problem formulations of the STF task.
- We propose a new categorization of data-driven models used in STF.
- We examine a wide range of application fields to track STF research progress.
- We propose the model selection criteria to assist researchers in selecting the adequate model for their STF task.
- We propose a novel, full view taxonomy that organizes and synthesizes the exogenous data incorporation approaches and levels.

B. SURVEY ORGANIZATION

As illustrated in Fig. 2, we organize the remainder of the paper according to the Spatio-Temporal Forecasting pipeline. In section II, we focus on the data properties by highlighting the data dynamics, data types and the different data representations. Then, in section III, we outline the different problem formulations that lay under the Spatio-Temporal forecasting problem. In section IV, we review the state-of-the-art data-driven models that were developed to tackle the different problems and wrap up the section by shedding light on the model selection criteria. In section V, we explore the existing techniques used to incorporate exogenous data in the models, as well as the possible levels to do so. The last section concludes the survey with a discussion.

II. INPUT DATA PROPERTIES

As a result of the digital revolution and the widespread of smartphones and Internet of Things (IoT) devices, data has become more widely available for analysis. Data collection and analysis have become tremendously important in addressing a variety of problems such as recommendation, anomaly detection, classification, and forecasting. It is the latter that has captured the researchers' attention the most [1]. This is mostly owing to the fact that accurately predicting the future of ST processes is critical in a wide range of applications. Various real-world phenomena are generated by a complex combination of factors in which spatial and temporal information play a significant role.

Forecasting entails analyzing previous observations and building a model to predict future values. The model must capture the spatio-temporal patterns of the data. Data that explicitly describes the studied phenomenon is referred to as endogenous data. Aside from the endogenous data under study, there are often additional features, technically known as exogenous variables, that are significantly correlated with the endogenous variables. When such factors are combined with ST data, the prediction performance is generally improved [15]. For example, Rong et al. have demonstrated that external factors such as weather, holiday events, and map query features can have a major impact on understanding the underlying substructures within the parking availability data [28].

A. DATA DYNAMICS

In the context of ST data analysis, the examined data can fluctuate along both spatial and temporal dimensions. For each of the variables, we use the notation $\mathrm{ST}(\mathbb{1}_S(\mathbb{D}), \mathbb{1}_T(\mathbb{D}))$ to describe data dynamics in the feature space $\mathbb{D}$, where

$$\mathbb{1}_S(\mathbb{D}) = \begin{cases} 1 & \text{if } \mathbb{D} \text{ is spatially variant} \\ 0 & \text{else} \end{cases}$$

and

$$\mathbb{1}_T(\mathbb{D}) = \begin{cases} 1 & \text{if } \mathbb{D} \text{ is temporally variant} \\ 0 & \text{else} \end{cases}$$

Accordingly, and as illustrated in Fig. 3, we can classify the features under study as follows:

ST(1,0): This class represents features that remain constant over time, but are spatially variant, i.e. the values of the features differ across spatial points but do not change throughout the observation period as illustrated in Fig.3a. This is for instance observed in the field of transportation. Road condition features such as the number of intersections, the number of lanes, the road function, and the road curve, in addition to the coordinates of different spatial locations are time invariant and spatially variant features [29], [30]. Other commonly used spatially variant exogenous features are the location weighting, and the fixed adjacency matrices, which facilitate the extraction of the proper spatial characteristics from the endogenous data [2].

ST(0,1): This class represents features that are time variant and space invariant, i.e. the observed time series are independent of the location, as shown in Fig. 3b. Numerous work have

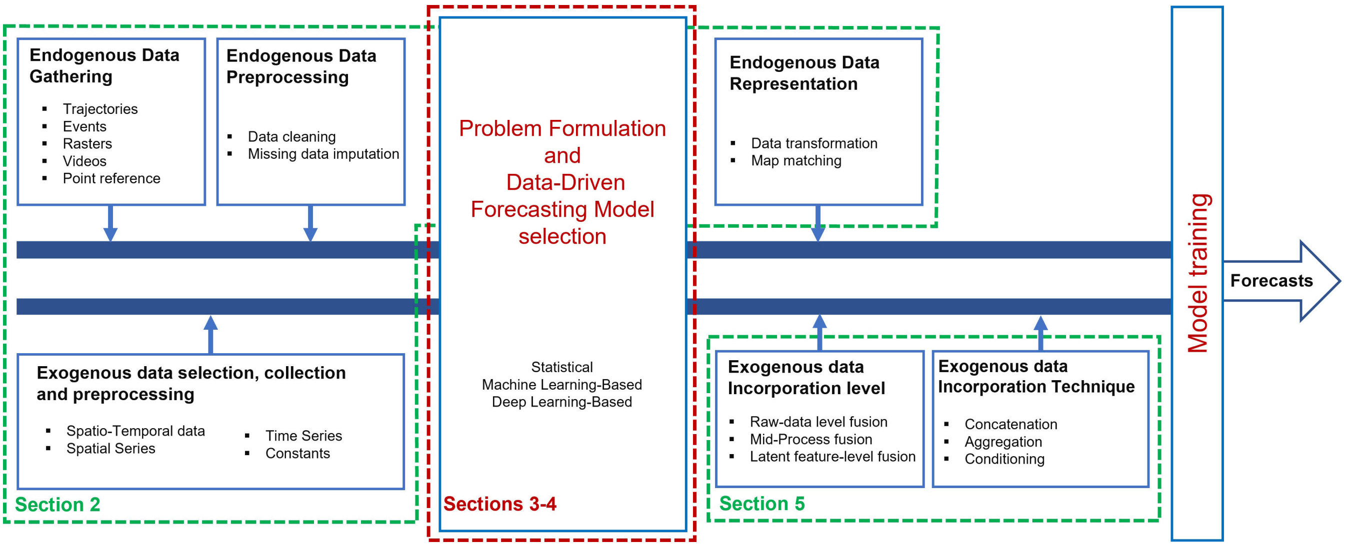
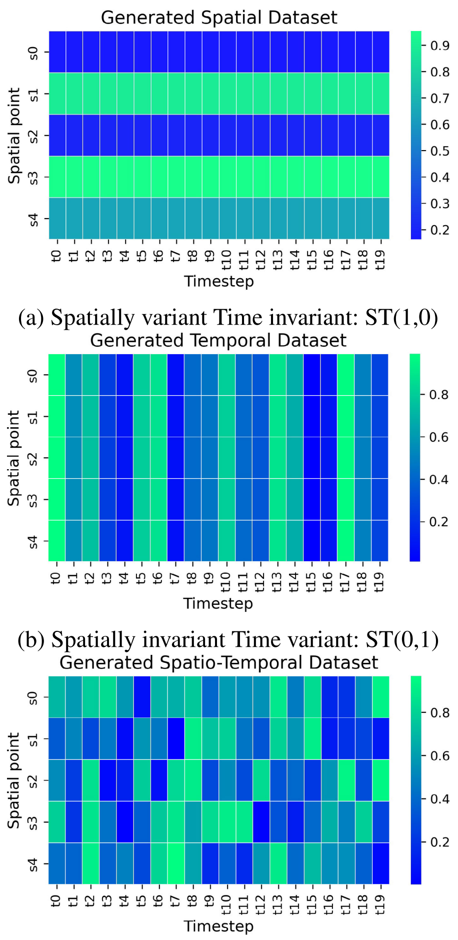


FIGURE 2. Spatio-Temporal Forecasting pipeline.



(a) Spatially variant Time invariant: ST(1,0)

(b) Spatially invariant Time variant: ST(0,1)

(c) Spatio-Temporal: ST(1,1)

FIGURE 3. Randomly generated data sets showing the dynamics of data.

incorporated this type of variables. In particular, calendar features such as: day of week, day of year, month of year and the binary feature indicating whether or not it is a holiday, are commonly used [28], [30], [31].

ST(1,1): This class represents features that are both time and space variant, as illustrated in Fig. 3c. Data fluctuate in both dimensions allowing each spatial point to have different values at different time instances. In the literature, there are numerous cases where exogenous variables vary both in space and time. For example, Andayani et al. used the ST price of dry grain data to predict the rice price [2], [3]. Likewise, Alajali et al. used the ST accident and roadwork data sets to forecast traffic volume [32]. Their experimental results show that incorporating nearby event information yields significant improvements. Cui et al. also used ST-data sets containing the traffic volume and the occupancy to forecast traffic speed [33].

ST(0,0): This class represents constants. Even though these are rarely used, they can still be useful. In particular, these constants can be valuable for fine-tuning models' parameters. Examples are statistical summaries of ST relations [30] and temporal period lengths [34].

Exogenous data with varying dynamics can be incorporated into the same model simultaneously. Several examples of exogenous features are presented in Table 1.

B. DATA TYPES

Besides their dynamics, we can categorize data according to their types. There are five major types:

Event data: Spatio-Temporal event data comprise discrete events, each characterized by a tuple (e_i, s_i, t_i) where e_i is the event type and s_i and t_i are respectively its location and time of occurrence. Event data are quite common in real world applications. For instance, in criminology it is common to exploit crime events history [4], and in traffic analysis, accident events [29], taxi [30] and car-hailing [36] demands are frequently analyzed. On the other hand, spatially constant events are collected without taking into consideration their location and are characterized by a tuple (e_i, t_i) denoting the event type and its occurrence time.

TABLE 1. Examples of exogenous features used in STF.

Paper	Target variable	Exogenous variables ST(1,0)	Exogenous variables ST(0,1)	Exogenous variables ST(1,1)
[29]	Accident events	Number of intersections Number of lanes Road curve	Day of year Day of week Month of year	Rainfall feature Temperature and wind speed Traffic volume data
[35]	Traffic intensity	–	–	Hourly temperature, Solar radiation Wind speed, Wind direction
[30]	Taxi requests history	Longitude and latitude of the region center	Av. demand value of the last 4 periods Events: Holidays	Weather conditions
[33]	Traffic speed	–	–	Volume and occupancy
[31]	Taxi-Bike trajectories	–	Holidays and weather conditions	Temperature Wind speed
[4]	Crime events	–	Holidays	Weather
[2], [3]	Rice price	Location weighting matrices	–	Dry grain price
[36]	Car-hailing supply-demand	–	Weather conditions: Weather type, temperature and $PM_{2.5}$	Traffic condition
[28]	Parking availability data	–	Holiday event features	Geolocation features, Navigation features Map query features Temperature, Weather, Wind speed

Trajectory data: Represent time-stamped paths of moving objects. Each trajectory is represented as a sequence $\{(s_1, t_1), (s_2, t_2) \ldots (s_n, t_n)\}$, where t_i represents the time stamp at which the moving object was at location s_i. This type of data is most common in the transportation domain, where accurate predictions are based on historical evidence of vehicles' trajectories [31], [37], [38].

Point reference data: Consist of measurements of moving reference points over a continuous ST field. Point reference data can usually be represented as a set of tuples $\{(r_1, s_1, t_1), (r_2, s_2, t_2), \ldots, (r_n, s_n, t_n)\}$. Each tuple (r_i, s_i, t_i) denotes the measurement of sensor r_i at location s_i of the ST field at time t_i. For example, Yu et al. [39] use data from floating cars that are equipped with Global Positioning System (GPS) devices. Each data instance is represented by a tuple that includes the coordinates, the timestamp, the measurement (vehicle speed), and extra information (direction).

Raster data: Refer to the measurements of a continuous or discrete ST field that are recorded at fixed locations in space and at fixed time points. Given m fixed locations $S = \{s_1, s_2, \ldots s_m\}$ and n timestamps $T = \{t_1, t_2, \ldots t_n\}$, the raster data can be represented as a matrix $Z \in \mathbb{R}^{m \times n}$, where each entry z_{ij} is the measurement at location s_i at time stamp t_j. This type of data is the most common and can be seen in multiple application fields such as urban traffic [32], [35] and agriculture [2].

Videos: Consist of a sequence of images, called frames, and are also considered as a type of ST data. A video can be generally conceived as a three- or four-dimensional tensor with one dimension representing time and the remaining dimensions representing a 2D or 3D image respectively. With the increasing availability of cameras and video recording devices, videos have become widely accessible for analysis and data mining. Shi et al. [40] and Wang et al. [41] used different video data sets to build models for next frame prediction.

C. DATA REPRESENTATION

In order to extract information from input data sets, data must be structured in a way that the model can handle. In the context of ST-forecasting, four different forms of data representations are commonly used: sequence, graph, matrix, and tensor. Data representation is determined by the input data type as well as the model to be used.

Sequence: All types of ST-data may be represented as sequences by merging all spatial information at each time step. Grigsby et al. [42], for example, embedded the different ST features in the data and represented it as a sequence before feeding it to the forecaster. Although this representation is the least utilized because it does not preserve spatial deformations, it still holds significant value in extracting temporal relations [35]. Formally, $\mathbf{z} = (z(1), z(2), \ldots, z(n))$ represents the data at n discrete points in time, where $z(i)$ is the value at time i, for $i \in \{1, 2, \ldots, n\}$.

2D Matrix: In the 2D matrix representation, rows indicate the different spatial points while columns represent timestamps. It is the second most often used representation and also the easiest to visualize. For example, De Medrano and Aznarte [35] and Andayani et al. [2] represented their endogenous data in the form of 2D matrices. In this case, the variable collected in m locations at n discrete points in time is represented as follows:

$$Z = (\mathbf{z}(1), \mathbf{z}(2), \ldots, \mathbf{z}(n)) = \begin{pmatrix} \mathbf{z}_1 \\ \mathbf{z}_2 \\ \vdots \\ \mathbf{z}_m \end{pmatrix}$$

$$= \begin{pmatrix} z_1(1) & \dots & z_1(n) \\ z_2(1) & \ddots & \vdots \\ \vdots & \ddots & z_{m-1}(n) \\ z_m(1) & \dots & z_m(n) \end{pmatrix} \in \mathbb{R}^{m\times n} \tag{1}$$

where $\mathbf{z}_j = \big(z_j(1), z_j(2), \dots, z_j(n)\big)$ for $j \in \{1, 2, \dots, m\}$,

and $\mathbf{z}(i) = \begin{pmatrix} z_1(i) \\ z_2(i) \\ \vdots \\ z_m(i) \end{pmatrix}$ for $i \in \{1, 2, \dots, n\}$.

Graph: Graph representation enhances the discovery of similar spatial points that are not necessarily close in terms of the Euclidean distance. It aids in leveraging semantic similarities between locations and thus retain the real topology in non-Euclidean spaces. Graphs are especially popular in Intelligent Transportation Systems (ITS) applications. For instance, [30], [43], [44] explicitly integrated the transportation network topology by representing it as a graph in which the nodes denote the road links. Khodayar et al. [45] also modeled their endogenous ST data as an undirected graph where each node represents a site and the edges reflect the correlation between the nodes. The graph representation of a variable collected at n points in time at m locations is given by: $\mathcal{G} = (V, E, Z)$ where V is the vertices' set, E is the edges' set, and $Z \in \mathbb{R}^{m\times n}$ is the features matrix.

Tensor: This is the most used representation since it maintains both the spatial variations and temporal fluctuations. As defined in [46], a tensor is an array of numbers arranged on a regular grid with a variable number of axes. It can be viewed as a higher-dimensional matrix where one dimension denotes the time axis, while the remaining dimensions refer to the spatial points' coordinates. Accordingly, to facilitate the utilization of their employed models, [31], [37], [38], [47] represented their input trajectories as 3D tensors. After grid-mapping the investigated region, they constructed their tensors by map-matching the trajectories using the techniques introduced in [34]. Zhang et al. represented their data in the shape of a 4D tensor since their spatial coordinates also included the altitude of the data instances [48].

For instance, the 3D tensor representation of a variable collected at n points in time over a $m_1 \times m_2$ grid is:

$$T = (\mathbf{Z}(1), \mathbf{Z}(2), \dots, \mathbf{Z}(n)) \in \mathbb{R}^{m_1\times m_2\times n} \tag{2}$$

where, for $\forall i \in \{1, 2, \dots, n\}$:

$$\mathbf{Z}(i) = \begin{pmatrix} z_{1,1}(i) & \dots & z_{1,m_2}(i) \\ z_{2,1}(i) & \ddots & \vdots \\ \vdots & \ddots & z_{m_1-1,m_2}(i) \\ z_{m_1,1}(i) & \dots & z_{m_1,m_2}(i) \end{pmatrix} \in \mathbb{R}^{m_1\times m_2}$$

Regardless of the data structure and representation, mining the most pertinent information from ST data is quite challenging. Due to the complex interactions, non-linearities, and non-stationarities within the ST processes, in addition to the entanglement of the spatial and temporal dimensions, analyzing ST data has emerged an area of intense study. This has led to the development of various data-driven models aiming at simultaneously capturing the spatial correlations and temporal dependencies within data.

The complexity of the forecasting models grows with that of the addressed ST-forecasting problem, which range from predicting the value of a unique variable in a single location at one time stamp ahead, to predicting multiple variables in several locations at multiple time stamps ahead. In the next section, we discuss the different forecasting problem complexities.

III. STF PROBLEMS CATEGORIZATION

Before delving into discussing the state-of-the-art data-driven approaches employed in the literature for ST forecasting, it is fundamental to specify the different problem formulations and complexities that fall within this scope. Hence, we categorize them according to their type, forecasting time horizon, and spatial coverage.

A. FORECASTING TASK

In terms of the prediction types, we can classify ST-forecasting problems into two categories: regression and classification. The studies that address the regression task aim at forecasting the continuous value of the target variable. For example, Cui et al. predict the traffic speed [33], Liang et al. forecast air and water quality [49], and Yu et al. predict soil temperature [50].

On the other hand, work conducted to perform the classification task intend to predict the probability of a certain event's occurrence based on ST inputs. For instance, some studies predict the occurrence of weather events such as storms [48], [51], while others predict urban anomalies [52]. Additionally, Liao et al. aim to identify traffic speed hotspots [53].

B. FORECASTING TIME HORIZON

We can categorize the STF problems conforming to their forecasting horizon. Although prediction horizons differ from an application to another and mainly depend on the granularity of the data, we can distinguish between two main categories: Monostep forecasting and Multi-step forecasting.

Monostep forecasting, also known as nowcasting, aims at forecasting the target variable at only one step ahead. Various studies address this short term forecasting problem, including [54], [55], [56], [57], and [58]. While problems that fall into the multi-step forecasting category aim to predict the values of the target variable in multiple steps ahead [59], [60], [61], [62]. It is especially advantageous in applications that require prediction enhanced planning. There are in fact two common approaches to tackle this long-term forecasting problem, the Iterated Mutli-Step (IMS) approach and the Direct Multi-Step (DMS) approach.

The IMS approach consists of learning a one-step-ahead forecaster and recursively applying it to yield the multi-step prediction. It trains a short-term forecasting model and

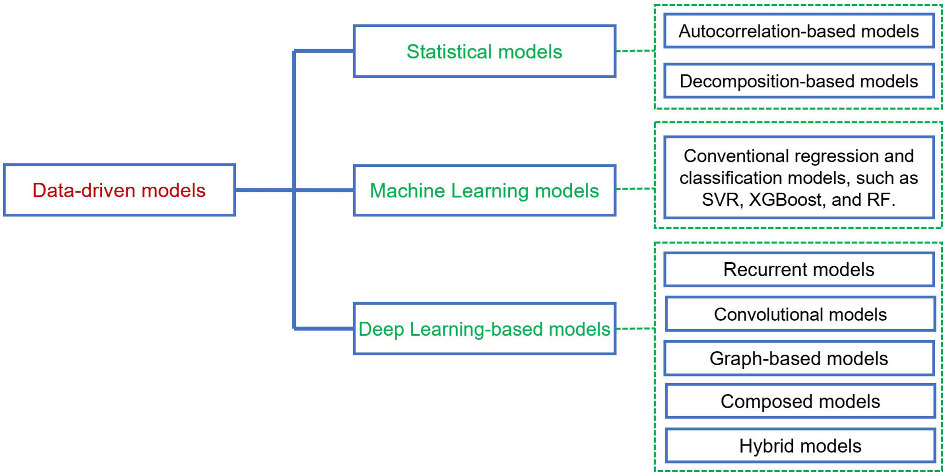


FIGURE 4. Categorization of data-driven models.

iteratively applies it while feeding the predictions of the early timestamps to generate the predictions of the later ones. Although this approach can theoretically yield predictions of arbitrary lengths, it rises the error propagation issue as the horizon extent increases. On the other hand, the DMS strategy directly forecasts the values at multiple future time steps. By doing so, it mitigates the problem of error accumulation and is particularly suitable for long-term forecasting tasks that require high computational costs. The DMS approach has been used in various applications, such as traffic prediction [63], soil moisture prediction [64], and air quality forecasting [65], [66].

C. FORECASTING SPATIAL COVERAGE

Last but not least, we can classify the ST forecasting problems conforming to their spatial coverage. Some experimental studies aim at predicting the target variable at one spatial location, which is referred to as the point wise prediction, like forecasting traffic volume at one target road [37], [38], or predicting the taxi demand at a specific area [30]. While in other studies, problems are formulated as an image prediction problem, focus on collectively predicting the target variable at all the spatial locations. Such is the case for predicting traffic conditions network-wide [28], [29], [67] or the rice price at all studied locations [2].

IV. STF MODELS

Several models have been proposed and developed in the literature to address the different complexities of the STF problem. In this section, we discuss the state-of-the-art data-driven approaches employed for STF and cover the criteria to consider when choosing the model to employ. As illustrated in Fig. 4, we classify them into three primary categories: Statistical models, conventional Machine Learning models, and Deep Learning-based models.

A. STATISTICAL MODELS

Spatio-Temporal series can be seen as multiple time series that involve a spatial correlation within them. Many researchers took advantage of the success of classical time series models such as AutoRegressive Integrated Moving Average (ARIMA) [68] and exponential smoothing (ETS) [69] and employed them to forecast the spatio-temporal processes. This involves the construction of several models, each corresponding to a different location. This approach has been adopted in forecasting shopping behaviors [70], traffic flow conditions [71], homicide rates [72], COVID-19 (Coronavrus Disease 2019) outbreak [73] and several other applications. However, it has two major drawbacks; it does not capture the spatial dependencies between the series and requires high computational power for large data sets.

1) AUTOCORRELATIONS-BASED MODELS

The expansion of the aforementioned models into the ST domain was initiated in the literature by Cliff and Ord [74] who introduced a space-time model by extending the Autoregressive (AR) models into the Space-Time Autoregressive (STAR) models. Pfeifer and Deutsch [10], [75] further developed the Space-Time auto-regressive integrated moving average (STARIMA) as an extension of the univariate ARIMA models into the spatial domain.

STARIMA is used to model a single variable collected in m locations at n discrete points in time by expressing each observation $z_s(t)$ at time t and location s as a weighted linear combination of past observations and errors lagged in both space and time. Specifically letting $z(t)$ be the $m \times 1$ vector of observations at time t, the STARIMA model class is expressed as

$$\nabla^d z(t) = \sum_{k=1}^{p} \sum_{l=0}^{\lambda_k} \boldsymbol{\phi}_{kl}^{(l)} \mathbf{W}^{(l)} \nabla^d z(t-k) + \boldsymbol{\varepsilon}(t) - \sum_{k=1}^{q} \sum_{l=0}^{\alpha_k} \boldsymbol{\theta}_{kl} W^{(l)} \boldsymbol{\varepsilon}(t-k) \tag{3}$$

where ∇ is the $m \times m$ difference operator matrix. p, d, q, λ_k and α_k are the model's orders, ϕ_{kl} and θ_{kl} are the ST parameters, $W^{(l)}$ is the weight matrix and $\boldsymbol{\varepsilon}(\boldsymbol{t})$ is the random normally distributed error vector at time t.

The STARIMA model, which is a three stage iterative procedure, starts with a tentative identification by preliminary designating the model's orders. Followed by identifying the model's parameters by solving (3). Then examining the residuals from the fitted model. It is considered robust in modeling and forecasting Spatio-temporal series. It is widely applied; Pfeifer and Deutsch [10] used it to forecast the spread of the percentage of farms with tractors, Lin et al. [76] and Kamarianakis et al. [77] adopted it to forecast traffic flow conditions, and Awwad et al. employed it to forecast the outbreaks of COVID-19 in Makkah [73]. One major deduction is that, despite the low number of parameters in the STARIMA model, it outperformed the use of several ARIMAs. That being stated, the STARIMA model can be extremely beneficial when dealing with datasets with high spatial and temporal dimensions. However, the STARIMA model is considered to have deficiencies in capturing heterogeneity of characteristics in each location. That is mainly because the model assumes same values of the parameters for all locations. In fact, each location usually may have a different influence, which makes the model only suitable for locations with homogeneous characteristics.

To address the heterogeneous inter-location characteristics, Borovkova et al. [78] introduced the Generalized Space Time Autoregressive (GSTAR) model. The model only involves autoregressive effects but is considered more realistic as the parameters are assumed to be different for each location. The inter-location heterogeneity is quantified by a weighted location matrix. Setiawan et al. applied the GSTAR model to forecast the foreign tourist arrival [79] and Suhartono et al. used it to forecast oil production [80]. GSTAR evolved afterward in non-stationary time series and moving average pattern, resulting in the Generalized STARIMA (GSTARIMA) model [81]. Still, the GSTARIMA is limited to only accommodating one observed variable in each location.

GSTARIMA models that include exogenous variables are known as the GSTARIMA-X models. The GSTARIMA-X model was introduced by Andayani et al. [2] who employed the transfer function approach. This study demonstrated that GSTARIMA-X was capable of modeling space-time data with exogenous variables on rice price data in six Indonesian provinces. Despite the capacity of exogenous data to improve the performance of a model, they are sometimes considered unnecessary. Consequently, it is important to compare the performance of models that do not incorporate exogenous variables with those that do, such as the work established in [3]. Andayani et al. applied both GSTARIMA and GSTARIMA-X models to further highlight the impact of incorporating the exogenous variables in the models. They have indeed concluded that GSTARIMA-X outperformed GSTARIMA which proves that in their case, the exogenous data enhanced the model. The model was further applied in several studies [82], [83], [84].

Significant efforts have been made recently to cope with the deficiencies of the STARIMA-based models. To mitigate the non stationary temporal problem found in the seasonal ST data, Zhao et al. proposed the Seasonal Difference STARIMA (SD-STARIMA) by bringing seasonal difference calculations into the model [85]. Implementing the proposed model to forecast the trends in the Hemorrhagic Fever with Renal Syndrome (HFRS), confirmed the model's capability to depict the seasonal characteristics in the non stationary ST data.

Another limitation of STARIMA-based models is that they assume that the spatial dependencies are fixed over time, whereas they may be dynamic. This entanglement between the spatial and temporal dimensions highly impacts the efficiency of the models. In some applications, like traffic forecasting, the spatial dependencies are highly dynamic leading to variations in the correlated spatial points. One plausible technique to alleviate this issue is to construct multiple STARIMA models for different time periods of the day. This, however, increases the computational complexity and does not take into account the temporally shifting periods. To this end, Duan et al. [56] proposed a unified STARIMA model which explicitly incorporates the impact of the physical factors and thus captures the intricate Spatio-Temporal correlation structure.

Coping with the STARIMA-based models' deficiencies at forecasting multi-step ahead, other statistical models, like the Spatial Beta convergence forecasting method, were exploited in the literature. Santos-Marquez extended the beta convergence forecasting method [86] to include spatial effects using a spatially lagged regression framework and applied it to forecast the crime rate in different municipalities in Columbia [72].

2) DECOMPOSITION-BASED MODEL: mSSA

Methods that were initially developed to forecast multivariate time series were also adopted in forecasting ST processes. That is the case for the multivariate singular spectrum analysis (mSSA)-based methods [87].

The mSSA was first proposed as an extension of the non-parametric Singular Spectrum Analysis (SSA) method [88], a powerful one used to analyze, impute, and forecast time series, into a vector forecasting method. In fact, the empirical studies in [87] show that when there is a spatial correlation within the data set, the mSSA yields better results than the conventional SSA method. Unlike the auto-correlation based models, mSSA consists of decomposing the temporal series into the sum of a number of interpretable components namely trend, periodicity and random noise.

The methodology of mSSA is closely similar to the principal component analysis (PCA) method. It involves three major stages, each containing different steps; decomposition, reconstruction and forecasting. The first step of the **decomposition** stage requires mapping the ST series $Z \in \mathbb{R}^{m \times n}$ into a multidimensional matrix, formally called the trajectory matrix. In the original mSSA, the chosen format is the Hankel matrix $H(Z,n,p)$ corresponding to the m series. In this step, two approaches can be employed. The induced trajectory matrices can either be stacked row wise, resulting in the **VMSSA** where $H(Z,n,p) \in \mathbb{R}^{mp\times K}$, p the window size and $K = n - p + 1$, or column wise, resulting in the **HMSSA** where $H(Z,n,p) \in \mathbb{R}^{p\times mK}$. The main parameter in this step is the window size p as it determines the shape of the final stacked trajectory matrix.

The second step requires applying the Singular Value decomposition on the stacked trajectory matrix to decompose it into the sum different elementary matrices (4), each representing a component of the original series.

$$H(Z,n,p) = H_1(Z,n,p) + H_2(Z,n,p) + .. + H_d(Z,n,p) \tag{4}$$

where d is the number of singular values of the elementary matrices $H(Z,n,p)$, $\boldsymbol{\varepsilon(t)}_i = \sqrt{\lambda_i}U_iV_i^\top$, $(\lambda_1 \geq \lambda_2 \geq \ldots, \lambda_d \geq 0)$ are the eigenvalues of $H \times H^\top$ arranged in a decreasing order, U is the right eigenvectors matrix and V the left eigenvectors matrix.

In the reconstruction stage, the number $r \leq d$ of components to keep is selected. It is a critical parameter in this step as, it should not be too large to avoid including the noise, nor too small to avoid information loss. The eigentriple grouping is then performed by clustering the elementary matrices into a k-set of disjoint subsets according to their singular values: $H'_1, H'_2, \ldots, H'_k$.

$$H'(Z,n,p) = H'_1(Z,n,p) + H'_2(Z,n,p) + .. + H'_k(Z,n,p) \tag{5}$$

$H'(Z,n,p)$ is the Hankel representation of the reconstructed denoised series. Next, the matrices are Hankelized by averaging their skew diagonals to obtain the reconstructed unidimensional series. Last but not least, the final stage consists of learning a linear model between the $p-1$ lines of the trajectory matrices and the p-th one.

Other variants of the original mSSa were proposed in the literature. Agarwal et al. [89] for instance, levitated the increasing size of the trajectory matrix problem by employing the Page matrix in the embedding step. In addition, they also discussed several other variants of the imputation mSSA such as the tensor SSA (tSSA). This latter's embedding step consists of considering each trajectory matrix of the temporal series as a slice of the final trajectory tensor, $T(Z,n,p) \in \mathbb{R}^{p\times n/p\times m}$. While the proposed representation was primarily used for missing data imputation, leveraging tensor estimation advancements, it can also be used for forecasting purposes in future studies.

Even though the mSSA is a relatively novel technique in Spatio-Temporal forecasting, it has been proven to be powerful in various previous studies thanks to its strong statistical basis. It has been applied to various real-life problems across different fields, such as burned forest area prediction [90], mortality rate forecasting [91], and industrial production forecasting [92]. It is especially used for noisy and short series since it also provides a method of data pre-processing. However, as aforementioned, singular value decomposition of the trajectory matrix is the key step in mSSA. That being said, when stacking the trajectory matrices of the multiple series, the spatial structure among the time series is lost. Furthermore, as the dimension of the stacked matrix increases by increasing the temporal history and/or the spatial coverage, computing its singular values becomes computationally expensive.

B. CONVENTIONAL MACHINE LEARNING MODELS

Classic statistical models are usually uncomplicated in their implementation all while keeping a high prediction accuracy. Still, these methods require strong assumptions, such as the linearity, stationarity and independence within data samples which are violated by most of the real life ST processes. The use of traditional Machine Learning methods was necessary given their ability to model non-linearity and extract more complex dependencies within the data. For convenience, we refer to the conventional Machine Learning-based models as Machine Learning-based models in the remainder of the paper.

The most frequent approach for forecasting time series using ML models consists in converting the original problem into a multiple regression task. In this case, the target variable is the future value of the series, while the predictors are previous past values of the series up to a certain p-length time window. To extend this simple strategy to the ST dimension, Spatio-Temporal indicators can further be incorporated in the exploited forecasting models [93]. In order to do so, Ohashi et al. first introduce the Spatio-Temporal distance. In fact, let $z_s(t)$ be the value of the endogenous variable at time t in location s. For (i,i') and (j,j'), two points in the spatio-temporal domain of interest, the values are $z_{i'}(i)$ and $z_{j'}(j)$ respectively. The authors define the Spatio-Temporal distance as $D_{((i,i'),(j,j'))} = d_{i',j'} \times \alpha + t_{i,j} \times (1-\alpha)$ where $d_{i',j'}$ and $t_{i,j}$ are the spatial distance and temporal one respectively, and α is a weighing factor. Next, they define the Spatio-Temporal neighborhood of a point (i,i') as $\mathcal{N}^{\beta}_{(i,i')} = \{(k,k') \in \mathbb{D} \mid D_{(i,i'),(k,k')} < \beta \text{ and } k \leq i\}$ where $\mathbb{D}$ is the available Spatio-Temporal data set. The ST indicators are then defined as summaries of certain properties of the time series and the spatial correlations between them, such as the average of the values within a neighborhood or the ratio between the moving averages of two different neighborhoods. Thus, the forecasting problem can be formulated as:

$$z_s(t+\tau) = f\left(z_s(t), z_s(t-1), \ldots, z_s(t-p), STI(\mathcal{N}_{(s,t)})\right) \tag{6}$$

where τ is the forecasting horizon, p is the historical window, f is the regression function to be learned and $STI(\mathcal{N}_{(s,t)})$ are the used Spatio-Temporal Indicators calculated on the set of neighborhoods $\mathcal{N}_{(s,t)} = \{\mathcal{N}^{\beta_1}_{(s,t)}, \mathcal{N}^{\beta_2}_{(s,t)}, \ldots, \mathcal{N}^{\beta_r}_{(s,t)}\}$. This incorporation allows the models to extract information not only from previous values of the series but also from the dynamics of series within the neighborhood [93], [94].

Support Vector Machine (SVM) and its variants are the most commonly exploited ML-models in STF literature [95], [96], [97], [98]. For instance, Pozdnoukhov et al. [99] formulated their problem as a classification one and applied the vanilla SVM to forecast snow avalanches. As for Wu at al. the Support Vector Regression (SVR) model was exploited since they performed short-term traffic flow forecasting, which they formulated as a regression problem [100]. They first assume that the traffic flow at a certain road is affected by the upstream roads, so in order to identify the series that will be input to the model, they estimate time lags between the aforementioned roads. A global SVR is first constructed to make the predictions, and based on the error analysis, local SVRs are used to improve the performance at intervals where the traffic flow is dramatically fluctuating. The final multiple SVR model, which combines the global and local SVR models is then adopted for predictions.

In order to yield good predictions, ML models require sufficient and reliable data especially when predicting phenomena with complex fluctuations. However, in some real life fields, the challenging data collection processes can lead to insufficient data and missing entries, as occurs in meteorology. To mitigate this insufficiency issue, Song et al. [101] introduced a spatial data incremental support vector regression (SaIncSVR) and employed it to forecast Particulate Matter - $PM_{2.5}$ concentrations. Through the inclusion of spatial domain data, the proposed model is able to further enhance the prediction of the SVR-based models by facilitating the learning procedure. Their experimental results suggest that the proposed model is robust to the relatively small data set and to missing samples.

Since ensemble learning enhances the performance of single predictors, tree-based ensemble models were widely employed in the STF literature. In particular, the bootstrap aggregating model Random Forest (RF) was exploited as can be seen in [96], [98], and [102]. On the other hand, boosting models such as Extreme Gradient Boosting Trees (XGBoost) and Gradient Boosting Regression Trees (GBRT) were proven to be stable and powerful in predicting Spatio-Temporal processes [102], [103], [104], [105]. Typically, Ashwini et al. exploited the predicting capabilities of XGBoost to forecast the short term bus arrival time [106], and Zhang et al. employed a GBRT-based model to predict travel times [107]. The proposed Spatio-Temporal Gradient–boosted regression tree (STGBRT), which incorporates spatio-temporal correlations induced from the ST data, demonstrated empirical success in fitting to complex nonlinear relationships with little data pre-processing required.

It is worth noting that most scholars apply various ensemble models in order to address STF problems. For instance, after pre-processing their raw data and obtaining high resolution gridded population projections, Chen et al. applied several boosting and bagging algorithms to predict the population from 2015 to 2030 at 5-year intervals [108]. In addition to the historical neighborhood endogenous population information, some exogenous environmental factors were also incorporated.

Although vanilla ML-models showed reliable forecasting results, if new data samples are provided, learning has to restart using all data samples. Alajali et al. investigated two types of learning schemes; Batch learning and Online learning. Within the former, three different ensemble models were adopted, as for the latter, the Fast Incremental Model Trees with Drift Detection (FIMT-DD) model was used [32]. The online FIMT-DD method shows promising performance compared with the batch methods taking considerably less time in both training and testing. Likewise, Ghaemi et al. [109] adopted the SVM-based learning system: LaSVM [110]. The online learning scheme allows the SVM to involve streaming data and update the model accordingly to continuously predict the air pollution.

C. DEEP LEARNING-BASED MODELS

Despite the ability of traditional ML methods to address the non-linearity issue, they remain insufficient owing to their shallow architecture. Moreover, although they can model complex ST data, they can not consider the temporal and spatial dependencies within the data simultaneously. Thanks to the capacity of DL-based models to capture the complex interactions and non-linearities, they were exploited in the STF literature. In this vein, Recurrent Neural Networks (RNNs), Convolutional Neural Networks (CNNs), Graph Convolutional Networks (GCNs) and their variants are the main building blocks of the state-of-the-art DL-based ST-forecasting models. The primary distinction is the chosen architecture and the complementary mechanisms used.

1) RECURRENT MODELS

Recurrent neural networks were initially introduced to capture the temporal patterns of sequential data; data that represent a significance in their order [111]. They are artificial neural networks with recurrent layers that can recognize and model non-linear temporal dependencies within the data. RNNs use a relatively small number of parameters, and thus they were exploited in handling ST data. Experimental results reveal that even the elementary Elman Recurrent Neural network shows promising performance compared to the statistical STARIMA model [112], implying that the RNN-based models yield better forecasts than the statistical ones. However, they appear to have a major drawback; their sensitivity to vanishing/exploding gradients impedes them to learn long-term dependencies.

To overcome short-term memory issue, extensions of the recurrent neural networks were employed [33], [41], [113], [114], [115], [116]. In particular, Long Short-Term Memory (LSTM) is a specific type of RNN which was introduced by Hochreiter and Schmidhuber [117], and is mainly designed to enable RNNs to learn long term dependencies. LSTMs have the same chain-like structure as RNNs, but instead of having a basic state cell, they rely on a special four-layered memory unit. This unit allows the network to perform three main operations: Decide which previous state can be forgotten/deleted through the Forget gate, decide which new information to store in the cell state through the Input gate, and decide what parts of the cell state to output through the Output gate.

To predict the wind speed at a target wind turbine, Shi et al. [114] adopted the LSTM in which the effective information no longer relies on the current neuron, but is transmitted directly to the next layer. They further enhanced the model by introducing the hybrid SC-LSTM model. It consists of analyzing the dynamic spatial correlation between the target wind turbine and its nearby turbines using the Wavelet Coherence Transformation (WCT). Historical data from the adjacent wind turbines are then input accordingly. The WCT has been proven to help depict the implicit spatial information and hence improve the model performance.

More complex architectures were introduced to deal with Spatio-Temporal data. In fact Wang et al. [41] proposed a multi layered RNN-architecture, called PredRNN, in which the different layers are dependent. They proposed a new ST-LSTM unit which allows the memory cell to zigzag between the layers, and thus gain the ability to not only capture the temporal dependencies, but also to accommodate the spatial variations as well. The performance of this latter inspired the introduction of more complex models, such as the ST-LSTM-Self Attention model proposed in [115].

Furthermore, Cui et al. [33] exploit the potential of LSTMs by building a stacked bidirectional and unidirectional LSTM (SBU-LSTM) neural network. To capture both spatial features and temporal dependencies, the Bidirectional-LSTM (BDLSTM) layer, which considers both forward and backward dependencies in time series, is employed. The LSTM layer is then used to generate the predictions. In order to study the influence of external features, they were concatenated with the predicted values before being fed to the forecasting model. The recurrent models have achieved satisfactory results in some Spatio-Temporal forecasting applications. They can especially be exploited in problems that are formulated as an element-per-element long term prediction ones. However, they are time consuming since they are unable to parallelize the forecasting process, as each time step necessitates the output of the preceding one to run. Moreover, they appear to have inadequacies when modeling spatial relations.

2) CONVOLUTIONAL MODELS

Given their recursive structure, RNNs have a privileged nature for dealing with sequential data and learning the temporal dynamics. However, they seem to have some deficiencies. The vanishing/exploding gradient problem, for example, can be lessened but not entirely solved by employing LSTMs and Gated Recurrent Units (GRU). That being stated, more complex modules were needed to capture the long-term dependencies.

Convolutional neural network (CNN) is a class of deep, feed-forward artificial neural networks that have a privileged nature for working with image-like data. They have demonstrated much empirical success in the computer vision field, image recognition tasks, speech processing and so on. Their ability to capture spatial interactions has also increased their usage in dealing with ST series. Furthermore, they are showing state-of-the-art performance in capturing temporal relations on a short term, which makes them suitable for ST data.

Given the out-performance of CNNs over RNNs in various applications, numerous work have employed convolutional networks in sequence modeling and prediction. In fact, to review the constant assignment of RNNs to ordered data, Bai et al. conducted an extensive systematic evaluation on both CNNs and RNNs on various sequence modeling tasks [118]. In this study, convolutional networks were represented by Temporal convolutional networks (TCNs) that are characterized by causal convolutions. These latter benefit from the strength of residual blocks and dilated convolutions to take sequences of any length. The comparison indicates that TCN does not only outperform RNN and its variants in terms of accuracy, but is also simpler and cleaner.

To further benefit from TCN's parallel calculations and extend it to ST series, Fan et al. [119] proposed a novel Parallel Spatio-Temporal Attention based TCN (PSTA-TCN). The proposed architecture combines stacked TCN backbones to extract temporal features with a parallel ST-attention mechanism to extract dynamic internal correlations. The spatial correlations in this study refer to the correlation between the target time series and the exogenous ones. To employ this architecture to ST data, series from adjacent locations to the target area can be considered as the exogenous series.

In addition to the temporal variations that can be obtained by the 1D CNN, the spatial correlation can be revealed by 2D CNN. Hong and Satriani proposed a robust design-based convolutional neural network and applied it to forecast the day ahead wind speed [120]. The multi-layer convolutional model showed a promising capacity in capturing both the spatial and temporal variations.

Instead of merely applying convolutional networks to forecast ST data, some research propose novel CNN-based architectures that are more sophisticated. For instance, Zhang et al. [34] proposed a Deep Learning-based prediction model for Spatio-Temporal data (Deep-ST) in which, the spatial dependencies of the data are learned by multiple layers of convolutional NN. The temporal dependencies are leveraged by generating three sets of input, each preserving a temporal property. Each set is passed through a Convolution layer that captures the spatial closeness. The convolution output is then fused with metadata before generating the final prediction.

Experimental results show that adding convolutions can capture spatially-far dependencies, but adding too many compromises the accuracy of the model. To mitigate this issue, the same authors [31], [121] introduced a ST neural network (ST-ResNet), based on residual units, and applied it to forecast citywide flow of crowds. The ST-ResNet aims at modeling the temporal properties of the data by combining three different residual CNNs. The model is then further consolidated by leveraging exogenous variables. Wang et al. [4] later adapted the same architecture to collectively predict crime distribution over the Los Angeles area.

In 3D-Tensor shaped data, the convolutional kernels in the aforementioned CNN-based models only slide in the spatial dimension. Alternatively, to preserve the temporal information of the input data, the use of convolution kernels that also slide in the temporal dimension was required. In fact, 3D convolutions have recently demonstrated empirical success in the literature, especially in video prediction [122], [123]. In the latter for instance, Djilali et al. proposed cubic convolution model to predict video saliency. The proposed Conv3D-based model depicts both spatial and temporal features simultaneously. It thus predicts the upcoming frames based on the current and the accumulated knowledge from the previous ones.

In this vein, Yu et al. [50] applied both Conv2D and Conv3D to predict the ST soil temperature. In the Conv2D, the spatial information is captured through the convolutions and the continuous sequences in the time dimension are preserved by the channels. Alternatively, the Conv3D performs the convolutions by sliding through both the temporal and spatial dimensions and jointly learns the features in one module.

Besides incorporating the temporal dimension, 3D CNN-based models were particularly valuable when handling 4D tensors. In fact, in [48], storm data instances are 3D radar data, outlining the latitude, the longitude and the altitude. Zhang et al. presented a multi-channel 3D-cube successive convolution network (3D-SCN). Within the 3D-SCN the spatial dependencies, as well as ancillary information, are captured by the cubic convolutions. The kernels slide along the three spatial dimensions, and the temporal variations are captured thanks to the stacked architecture.

The performance of CNN-based models can be further enhanced by pre-processing the input data. One popular pre-processing method is decomposition, which involves decomposing the input sequences into their main components, thereby facilitating temporal feature extraction. For example, Ahajjam et al. employed different levels of Variational Mode Decomposition (VMD) and Wavelet Decomposition to pre-process the data before feeding it to the forecasting models [124]. Their findings indicated that appropriate decomposition levels can lead to improved model performance. Similarly, Yu et al. utilized the Empirical Ensemble Mode Decomposition (EEMD) method [125], and their experimental results demonstrated that the EEMD-Conv3D model outperformed the contrasted models in terms of accuracy, suggesting that it better captured the input data characteristics. Overall, these studies showcase the utility of decomposition methods for pre-processing data and enhancing the performance of CNN-based forecasting models.

3) GRAPH MODELS

When contrasted with other deep learning models, the conventional CNNs proved successful in abstracting spatial correlations with a reduced number of parameters. However, they are most appropriate for image-like data in which the spatial structure is Euclidean. That being said, to feed data into CNNs, it should be structured in a shape that can be processed by it. Yet, the grid conversion of graph structured data requires considerable effort and leads to spatial information loss. To mitigate this issue, the convolution operator was extended to handle graph-structured data leading to the introduction of Graph Convolutional Networks [126], [127]. Graph convolutions can be directly applied on graph-structured data which reinforces the extraction of significant patterns and features in the non-Euclidean space domain.

To forecast road traffic speed, Yu et al. proposed a graph-based learning model that expanded the existing Graph Convolutional Network [43]. To depict the dynamic spatial dependencies multiple parameterized adjacency matrices were used. These latter allowed the model to account for different intensities of connectivity between neighbors, unlike the conventional GCN that gives equal weights to each neighboring road. As for the temporal patterns, they were extracted by stacking different numbers of graph convolutions corresponding to each time lag, inducing more graph convolutions for older traffic states.

One of the primary challenges in adopting graph-based forecasting models is their heavy reliance on the graph structure, where the nodes represent spatial locations, and the edges embed the strength of the spatial connections. As a result, there has been a significant focus in the literature on the graph learning phase. A variety of approaches have been explored to construct adjacency matrices for spatio-temporal data and multivariate time series, including distance based, correlation based and information theory based methods.

Distance-based approaches rely on the first law of geography, which states that "everything is related to everything else, but near things are more related than distant things". These approaches use different distance measures, such as the Euclidean and Manhattan distances between nodes, as well as dynamic time warping.

Correlation-based methods assume that sequences that are highly correlated are related and should be linked in the graph. For example, Gu et al. [128] proposed constructing the graph based on the Pearson correlation coefficient, resulting in a dynamic correlation adjacency matrix.

Information theory-based techniques leverage mutual information and transfer entropy to construct the graph. For instance, Khodayar and Wang [57] used mutual information between time series to build the adjacency matrix, while Duan et al. utilized transfer entropy to build directed graphs that emphasize the causal relationships between nodes [129].

Although predefined graph structures can lead to good forecasting performance, they might limit the model to the initial state of the graph. To mitigate this limitation, multiple studies have incorporated ML based graph learning layers into their models resulting in end-to-end models that adaptively learn the graph structure from the input data [130], [131], [132], [133], [134].

Instead of relying on a single adjacency matrix, Wang et al. [135] proposed the Multivariate Time Series Forecasting via Heterogeneous Graph Neural Network (MTHetGNN) model. MTHetGNN leverages multiple, heterogeneous graph embeddings to forecast spatio-temporal data. These approaches offer promising avenues for addressing the challenges of graph-based forecasting models and improving their accuracy and reliability.

4) CONVOLUTIONAL-RECURRENT MODELS

To benefit from RNNs' ability to model the long-range temporal dependencies as well as CNNs' effectiveness in capturing the spatial variations, composed architectures that exploit different classes of neural networks have been proposed in the literature [136]. For instance, Yu et al. [39] proposed a Spatio-Temporal Recurrent Convolutional Network (SRCN) that inherits the capacities of both CNNs and LSTM. In fact, the SRCNs is composed of a DeepCNN that mines the spatial near and far dependencies, followed by LSTM layers that learn the temporal dynamics.

When using the DCNN, the model takes into consideration all the locations to predict the variable at the target location, even those with weak correlation, which can lead to deterioration in model performance. To address this problem, Yao et al. [30] proposed a novel Deep Multi-View ST Network (DMVST-Net) and applied it to predict the Taxi demand. Instead of the conventional CNN, the spatial component of the DMVST-Net employs a **local** CNN, which only captures spatial dependencies within the neighborhood of the target location. The temporal component includes an LSTM which, at every time step, takes as input the corresponding output of the spatial component along with the context features. The third component of the model is a semantic one that improves the spatial features by allowing the model to take into consideration the regions that are not close to the target but still have an impact on it. The output of this latter component is concatenated with the LSTM one, and is then fed to a fully connected layer that generates the final prediction.

The DMVST-Net captures both the spatial and temporal variations. However, it does not capture the dynamic spatial similarity nor the periodic temporal shifting. As a result, Yao et al. extended their work by proposing a novel framework in [38]. In fact, they propose a Spatial-Temporal Dynamic Network (STDN). The spatial variations are captured by a local CNN, passed through a flow gated mechanism that further assesses their importance and are then concatenated with external features and fed to the temporal modules. The long-term temporal dependencies are modeled by a Periodically Shifted Attention mechanism that captures the temporal shifting, while the short term information is captured by an LSTM.

To improve the long-term dependency of the model, Wang et al. [37] proposed a multi-scale attention mechanism that uses available hourly, daily, and weekly data to capture temporal features. In fact, rather than feeding the model all of the available historical data, the selection step consists of identifying the intervals that are useful for the prediction. Spatial dependencies in the vicinity of the target location are detected by a local CNN. The BDLSTM, enhanced by a masking mechanism, then reinforces the learning of temporal variations.

Although the aforementioned models yield satisfactory results, the predictions are made element per element. That being said, to predict the values for all the locations, several models should be trained, which is expensive. To alleviate this problem, De Medrano and Aznarte [35] have proposed a Convo-Recurrent Attentional Neural Network (CRANN) that relies on the idea of the classic additive ST series decomposition and collectively forecasts the future values. The model was defined as a combination of several modules, each of which captures a key component of the series, these latter are further aggregated to make the final predictions. In particular, the temporal module consists of two LSTMs stacked in an encoder-decoder architecture and enhanced by an attention mechanism to learn the seasonality and trend of the series. As for the spatial module, it is composed of a conventional CNN followed by a spatio-temporal attention mechanism that captures both the short-term and spatial dependencies. A dense module is then in charge of connecting the modules, the exogenous data, and the auto-regressive terms before making predictions.

5) GRAPH-RECURRENT MODELS

Another architecture, that has received widespread attention for graph-structured Spatio-Temporal data is the GCN-RNN integrated architecture wherein a GCN depicts spatial dependencies and an RNN extracts the temporal correlations [137], [138], [139], [140]. Following this framework, Zhao et al. [44] proposed the T-GCN model, which consists of two parts: GCN and GRU, and implemented it to forecast traffic conditions. The GCN is employed to capture the complex non-Euclidean topological structure in the data, and the Gated Recurrent Unit is used to capture temporal features. The model was further explored to forecast the charging demand of Electrical Vehicles [141].

Similarly, Cui et al. [142] adopted the GCN-RNN architecture, and introduced the Traffic Graph Convolutional LSTM (TGC-LSTM) to collectively forecast traffic states in a given network. They modeled the traffic network as a graph, where the nodes represented the sensor stations, and the edges reflected the roads connecting the stations. For the first module of the model, they extended the conventional spectral based graph convolutional network by incorporating physical properties of roadways. The generated spatial features are then flattened and fed into an LSTM that learns the temporal dependencies within the data. This model, which maintains a fixed adjacency matrix, does not consider the dynamic graph structure and hence overlooks the entanglement of the spatial and temporal dimensions.

Khodayar and Wang [57] proposed a scalable graph-based wind speed forecasting model which combined LSTMs, GCNs and Rough Set theory. They first modeled the wind farms under study as an undirected graph, where the nodes depicted the wind sites and the edges reflected similarities between the nodes in terms of mutual information. For each wind site, an LSTM is required to map the historical data into the temporal feature space. The extracted temporal features are then fed to several spectral-based Graph convolutional layers stacked together with rough feature learning layers. This allows the model to capture the spatial features and provide the forecast wind speed values at each wind site. To optimize the number of parameters, all of the LSTMs in the model share the same parameters. This assumes that all wind sites have the same temporal patterns causing the temporal information to deteriorate.

As a variant of graph neural networks, Xu et al. [143] employ a spatial transformer to dynamically model directed spatial dependencies within the data using self-attention mechanism. As for the temporal dependencies, they are captured through a Temporal Transformer. These two transformers are assembled to form the Spatial-temporal Transformer Network (STTN)'s Spatio-Temporal Block, capable of modeling long-range dynamical spatio-temporal dependencies. The final network is constructed by stacking ST-blocks and then adding a predictive basic convolution-based layer. The resultant STTN captures the dynamic spatial dependencies within data. But rather than learning the spatial connections from this latter, it relies on preset positional embedding holding the dynamic spatial structure.

Inspired by the good performance of transformers, especially in natural language processing, they were further exploited in the literature to handle STF tasks. In fact, to adapt transformers to multivariate series, Grigsby et al. [42] started by a Spatio temporal embedding. A value and time embedding maintained the temporal information in the input data, a variable embedding incorporated the spatial features to distinguish between the different series, and a given embedding to differentiate historical data from the to-be forecast data. After the ST-embedding, data were passed through a standard transformer encoder-decoder architecture strengthened by local attention modules. Transformers were also exploited in the GCN-RNN architecture. For instance Zhang et al. [144] proposed a novel GCN-Transformer model, and employed it to forecast short term passenger flow.

6) GRAPH-CONVOLUTIONAL MODELS

Rather than employing recurrent neural networks, adopting an entire convolutional structure on the time axis yields promising results, leading to increased interest in the GCN-CNN architecture [145]. Yu et al. [146] introduce a novel Deep Learning framework to forecast traffic states; The Spatio Temporal Graph Convolutional Network (STGCN). The proposed architecture consists of several stacked ST convolutional blocks. Each block is composed of two temporal gated convolution layers centered by a spatial graph convolution layer. The experiments that were conducted reveal that the model is quicker during the training phase than state-of-the-art models, which is expected since it uses fewer parameters.

Zhang et al. [147] adopt convolutions on both the temporal and spatial axes. In order to extract the spatial dependencies, they propose a novel graph-based framework named Structure Learning Convolution (SLC). SLC explicitly incorporates the structure information into the convolution operation and employs it to model local and global patterns separately. The SLC modules, which exploit the structure learning convolutions have the ability to leverage both the static and dynamic graph information. To furher depict temporal dependence, the Pseudo-3D convolution is integrated in the model. The model is then built by stacking multiple SLCNN blocks allowing the expansion of the convolutions' receptive field. The composed models outperform the baseline ones in terms of evaluation metrics. However, the use of multiple modules from different natures makes them highly expensive due to the high number of parameters.

In the literature, RNNs, CNNs, and GCNs were also employed simultaneously. The Attention-adjusted Graph Spatio-Temporal Network (AGSTN) [148] can be viewed as a two-module model. For the first module, the Empirical Ensemble Decomposition algorithm generates additional time series along with the raw ones. Using multiple graph-convolutional networks, a ST correlation matrix that embeds the dynamic spatial correlations between the series is generated. Forecasts are then provided by a 1D CNN applied on the STC matrix. The second module, on the other hand, inputs raw time series and produces forecasts based on recurrent architecture. A final prediction is produced by leveraging both forecasts through an attention mechanism.

7) HYBRID MODELS

Instead of stacking different neural networks, Shi et al. [40] introduced ConvLSTMs as a hybrid model able to simultaneously handle the temporal and spatial dimensions. Convolutional LSTMs were defined as a special case of LSTM that keep the standard recurrent structure but use a ConvLSTM unit instead of the regular LSTM one. More precisely, the input-to-state and state-to-state transitions in the ConvLSTM cell involves convolutional operations that output 3D tensors. Instead of using a fully connected layer, as it is the case for a conventional LSTM, convolution operation is applied which allows the model to capture the spatial information all while reducing the number of parameters.

The ConvLSTM captures the temporal auto-correlation in the data through the LSTM part and captures the local spatial properties thanks to the convolution operator. This makes the model especially suitable for problems that can be formulated as a next frame prediction problem [149]. Aidan Curtis [150] applied the ConvLSTM to predict sport videos' dynamics. They proved that a ConvLSTM that lacked depth could not consider all spatio temporal dependencies leading to a decrease of the prediction accuracy as the horizon increased. To mitigate this issue, stacked ConvLSTMs were employed as they have the ability to learn hierarchical representation of the raw data [151], [152].

Ma and Mao [153] employed a Deep ConvLSTM network, which involves two stacked ConvLSTMs and a fully connected layer to predict the remaining useful life of rotational machinery. However, in the field of traditional spatio-temporal, it is important to properly detect the temporal periodicity and seasonality of the data. For this purpose, Wang et al. [154] employed a deep Spatio-Temporal ConvLSTM and proceeded by segmenting the time series. They then built a model that consists of three different branches that have the same structure but do not share weights. Through this segmentation, the temporal feature is extracted from three aspects (closeness, period, trend). Similarly, He et al. [60] introduced the Spatio-Temporal Convolutional Neural Network (STCNN). The model consists of ConvLSTMs built in an encoder-decoder architecture. The encoder leverages the general ST dependencies and uses skip-ConvLSTMs to explicitly detect the periodic patterns. The decoder then decodes the ST dependencies and combines them with the periodical patterns.

When training models that are based on the ConvLSTM blocks, using Mean Squared Errors tends to bias the models towards producing blurry images. To address this problem, different loss functions were used in the literature. Zhao et el. [155] used the CW-SSIM loss function to train applied deep ConvLSTM model. Tan et al. [156] also used a CNN-DeepConvLSTM architecture but introduced a new loss function; Forecaster loss. This latter assigns higher weights to pixels with lower value preventing the model from yielding blurry images.

Although deep ConvLSTMs raise the forecasting time step, as the depth increases the model begins to suffer from vanishing gradients. To alleviate this drawback, Yasuno et al. [157] employed a seemingly forecasting timestamp reduction by fusing multiple frames into one. The fused frames were then fed into the ConvLSTM to generate the predictions. Another drawback of the ConvLSTM is that it does not explicitly address heterogeneity. In order to overcome this challenge and better capture the temporal trends, Yuan et al. [29] incorporated spatial features in the ConvLSTM model by map-matching them with the endogenous data. In fact, instead of training a single ConvLSTM that covers all the studied area, the Hetero-ConvLSTM trains a deep ConvLSTM for each sub-region of the examined area. The ensemble of the outputs is then used as the final prediction.

Despite their groundbreaking abilities in depicting complex ST features, DL models are often criticized for being computationally expensive. Furthermore, forecasts should be made more interpretable and reasonable. Therefore, when selecting a model, the trade-off between the model's performance and its requirements must be considered.

D. MODEL SELECTION

As the **no free lunch** principle affirms [158], no model can be seen as the best or worst model. In fact, since the input data and problem formulations vary vaguely, there is not a "go to" model that can be employed to solve all the STF problems. Thus, to select the model to be adopted, scholars should pay attention to a number of aspects to determine the most adequate one. By adding the accuracy of the model to the four aspects of the multi-dimensional comparison suggested in [112], we propose a synopsis of the main points to consider when choosing the forecaster.

1) EASE OF IMPLEMENTATION

Statistical models usually have limited steps, but the determination of the parameters require experience and good understanding of both the model and the data to avoid errors. Furthermore, the statistical models can only tackle well-structured data. Since the collected ST data in their raw form are often unstructured, much effort is required to arrange data in a way that retains all the information and can be processed by the model.

DL models, on the other hand, usually do not require much pre-processing but tuning the hyper-parameters can sometimes be delicate given their influence on the model. To design a robust model, several approaches exist in the literature and have been widely employed. Ranging from manual like the Grid Search and Random search, automated like the Bayesian optimization or more sophisticated methods like the Orthogonal Array Tuning Method (OATM) [159]. This last-mentioned approach is based on the Taguchi method [160], and was exploited in [120] to determine all the structure parameters and hyper-parameters needed to design a robust ST-forecaster CNN.

2) INTERPRETABILITY

The main goal in some cases is to obtain the predictions, without the need to pinpoint the factors that lead to the variation of the endogenous variables. In [28], Rong et al. aim at predicting the real-time parking availability to provide information to drivers and allow them to find a parking space faster. In this application, the authors neglect the factors that lead to the increase or reduction in the number of available spots. In other case-studies, beyond providing the predictions, it is crucial for the model to explain why it yields them. Understanding those factors is especially useful if a certain event, like an accident [29] or a crime [4], should be prevented. The idea is to mitigate the factors that enhance the likelihood of the event's occurrence. In this context, the interpretability, or transparency of the model is essential.

The vanilla statistical and ML-based models are generally easy to interpret as they provide explicit parameters and highlight which aspect has the most influence on overall prediction result. On then contrary, due to their black-box nature, DL-based models, and even thought they might yield good predictions, usually do not provide insights into the

TABLE 2. Commonly used Regression Evaluation Metrics for Spatio-Temporal Forecasting.

Regression	
Index	**Expression**
Mean Absolute Error	$\frac{1}{n}\sum_{i=1}^{n}\lvert\hat{z}_i - z_i\rvert$
Mean Squared Error	$\frac{1}{n}\sum_{i=1}^{n}(\hat{z}_i - z_i)^2$
Root Mean Squared Error	$\sqrt{\frac{1}{n}\times\sum_{i=1}^{n}(\hat{z}_i - z_i)^2}$
Correlation Coefficient	$R^2 = 1 - \frac{\sum_{i=1}^{n}(\hat{z}_i - z_i)^2}{\sum_{i=1}^{n}(z_i - \bar{z})^2}$
Mean Absolute Percentage Error	$\frac{1}{n}\sum_{i=1}^{n}\left\lvert\frac{\hat{z}_i - z_i}{z_i}\right\rvert \times 100\%$

TABLE 3. Commonly used Classification Evaluation Metrics for Spatio-Temporal Forecasting.

Classification	
Index	**Expression**
Precision	$\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}}$
Recall	$\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}}$
Threat score	$\frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}+\mathrm{FP}}$
Equitable threat score	$\frac{\mathrm{TP}-\mathrm{TP}_{\mathrm{random}}}{\mathrm{TP}+\mathrm{FN}+\mathrm{FP}-\mathrm{TP}_{\mathrm{random}}}$

reasons behind them. However, since the interpretability has become a must-have characteristic in different applications, many researchers have adopted several methodologies that are able to mitigate this issue. For example, widely used in the STF task solving, the attention mechanism does not only help the models yield better results in terms of precision, but also helps add an intrinsic interpretability layer to the DL-based models [161].

The attention scores, output of the attention mechanism, can be interpreted as the importance of different feature groups for prediction. They are used to understand how the model utilizes the different inputs when forecasting. Thus, different scholars adopted this mechanism, like [35] and [162]. Furthermore, as they typify the importance of every input when predicting, they can be visualized to enhance the interpretability of the model, like it was performed in [49] and [42].

The saliency-based SHAP values method [163] is another strategy used to improve the model's interpretability. The SHAP values, or SHapley Additive exPlanations quantify the contribution of each predictor to the forecast value, either negatively or positively. It should be noted, however, that this technique necessitates high computational power since it requires the implementation of all the possible scenarios.

3) PERFORMANCE INDEXES

Although the accuracy of the model might seem like the main factor to consider when adopting a forecaster, we must pay attention to the trade-off between it and the aforementioned aspects. The commonly used regression performance indexes for STF are listed in Table 2. These indexes are calculated using n data points with z_i representing the observed values and $\hat{z}_i$ representing the predicted values.

It is worth mentioning that Mean Absolute Percentage Error (MAPE) values are usually misleading for count-regression applications, in which the values may be null, so it is recommended to avoid using it in these scenarios.

In applications that are formulated as a next frame prediction problem, the aforementioned accuracy metrics are not the most adequate. A low Mean Absolute Error (MAE) or Mean Squared Error (MSE) for example does not necessarily reflect a good prediction as the picture can be blurry or lacks details. Consequently, other performance metrics were proposed in the literature. For example, Wang et al. used the Structural Similarity (SSIM) score and Peak Signal to Noise (PSNR) [41].

Based on three comparison functions, namely luminance l, contrast c, and structure s, the SSIM [164] score attempts to assess the similarity between windows of the ground truth frame Z and the forecast frame $\hat{\mathbf{Z}}$ as follows:

$$\mathrm{SSIM}(\mathbf{Z},\hat{\mathbf{Z}}) = f(l(\mathbf{Z},\hat{\mathbf{Z}}), c(\mathbf{Z},\hat{\mathbf{Z}}), s(\mathbf{Z},\hat{\mathbf{Z}})) \tag{7}$$

where $f(.)$ is the combination function. The overall structural similarity index is then calculated as the average of the similarity between theses windows.

$$\mathrm{MSSIM}(\mathbf{Z},\hat{\mathbf{Z}}) = \frac{1}{\mathrm{M}}\sum_{j=1}^{M}\mathrm{SSIM}\left(\mathbf{Z}_j,\hat{\mathbf{Z}}_j\right) \tag{8}$$

where M is the number of windows.

The Peak Signal to Noise Ratio (PSNR), on the other hand, is based on the MSE, and is determined by the following expression:

$$\mathrm{PSNR} = 10\log_{10}\left(\frac{V}{\sqrt{\mathrm{MSE}}}\right) \tag{9}$$

where V is the maximum variation in input image data; 255 in the RGB images.

As for STF problems formulated as classification problems, several performance metrics are commonly used. Table 3 summarizes the most frequently used among them. TP, TN, FP, and FN stand for True Positive, True Negative, False Positive, and False Negative, respectively. $\mathrm{TP}_{\mathrm{random}}$ is the number of TP expected by random chance.

In the literature, especially in the meteorological field, Recall is also referred to as Probability of Detection (POD), while Threat Score (TS) is referred to as Critical Success Index (CSI). Other metrics that are often used include False Alarm Rate (FAR) $= 1-$ Precision, and the F1-score $= \frac{2\times(\mathrm{Recall}\times\mathrm{Precision})}{\mathrm{Recall}+\mathrm{Precision}}$.

Precision, Recall, and the F1-score do not take into account the true negative rate. On the other hand, FAR, TS, and Equitable TS can evaluate both deterministic and probabilistic forecasts, but can be affected by imbalanced datasets, especially for rare events. Schaefer et al. suggested that CSI is a better choice for evaluating low-frequency events [165]. In general, the choice of metrics to use should depend on the specific problem and the goals of the evaluation. It is recommended to use multiple metrics to have a more comprehensive evaluation.

In addition to these prior aspects, it is also important to consider the models' running time, instability and memory usage. For instance, a model with a large number of parameters may have a better ability to capture deeper ST features from data but will require more time and memory. It is also necessary to determine whether the model can handle exogenous data and how difficult it will be to inject them. In the next section, we propose a taxonomy of current approaches used to incorporate these latter, as well as the possible levels.

V. EXOGENOUS DATA FUSION PHASE

Exogenous data can be regarded as a form of prior knowledge that lends structure to the information contained in the endogenous data set. As a result, the models would require less historical data while still yielding good predictions. This is particularly advantageous when employing DL-based models. In this vein, several studies have attempted to incorporate both exogenous attributes and endogenous variables into the forecasting models. The merge allows the model to jointly learn features and capture the correlations among them. However, it is critical to test whether exogenous variables are beneficial. Otherwise, they may just hurt the model by adding unnecessary or redundant information.

In addition to selecting the appropriate exogenous variable and its type, when and how it is incorporated in the model both play a significant role in the improvement that it may bring to the forecasting model. In this section, we explore these two aspects and propose a taxonomy of the existing incorporation approaches.

A. FUSION LEVEL

The fusion level can play a considerable influence on the impact of exogenous data on the forecasting models. Based on this level, we can categorize the incorporation approaches, as illustrated in Fig. 5, to three main categories.

1) RAW DATA-LEVEL FUSION

In this case, we incorporate the exogenous data in the model at the initial step of the forecasting process, before dealing with endogenous data. To rephrase it, exogenous data are fused with the raw endogenous variables before being jointly input into the ST-forecasting model.

This fusion approach is particularly used when the ancillary data play a highly discriminative role in the prediction. In [48] the raw 3D radar data are insufficient to forecast the storm initialization, growth and advection. Thus, two exogenous variables, output of a re-analysis system where also included in the input tensor. Last but not least, to take into account the temporal trend of both the endogenous and exogenous variables, their point-to-point difference were also considered as initial inputs of the model. Thus, at each time stamp, the input tensor is 4-dimensional. The three first dimensions refer to the spatial coordinates and the fourth one denotes the number of channels, in this case, the total number of input variables. Authors in [29], [33], and [43] also incorporated the exogenous data in the raw data-level, by feeding all the engineered exogenous features along with the endogenous variable to the forecasting models.

2) MID PROCESS FUSION

In this situation, exogenous variables are not introduced to the model until certain features are learned from the endogenous variables. That being said, the extracted features are fused with the exogenous data and then fed into the rest of the model's modules for further learning.

For instance, in the DMVST-Net model proposed in [30], instead of directly merging the exogenous and endogenous variables at their raw state, the endogenous data are passed through a spatial component. The output vectors are then fused with the exogenous variables. Finally, the resulting vector is fed to a temporal component to extract the sequential dependencies and yield the predictions. In [36], Wang et al. begin by passing the endogenous data through the first block of the model. The output embodies the endogenous feature. It is then fused with the exogenous features and jointly fed to the rest of the model's components to provide the predictions.

3) LATENT FEATURE-LEVEL FUSION

The approaches that lay under this section are characterized by the fact that exogenous data are exclusively introduced to the ST-forecasting model when the endogenous ST feature extraction procedure is achieved. In other words, the exogenous variables are presented at the final step; right before making the final predictions.

Within the CRANN model proposed by De Medrano and Aznarte [35], different modules are employed to capture the spatial and temporal variations from the endogenous data. Then, before yielding the prediction, a dense module is used to fuse the extracted ST features along with exogenous data and autoregressive terms. Similarly, in the Deep-ST [34] and ST-ResNet [31] models, the spatio-temporal features are modeled independently and only fused before applying the activation function and providing the final prediction.

As far as the STARIMA-based models are concerned, the exogenous and endogenous impacts are both modeled independently, then fused together using the transfer function model approach. We preserve the same notations introduced in the models' section, and let $\mathbf{x}(t)$ be the $m \times 1$ exogenous vector of observations at time t. Incorporating exogenous data in (3) results in the STARIMA-X Transfer

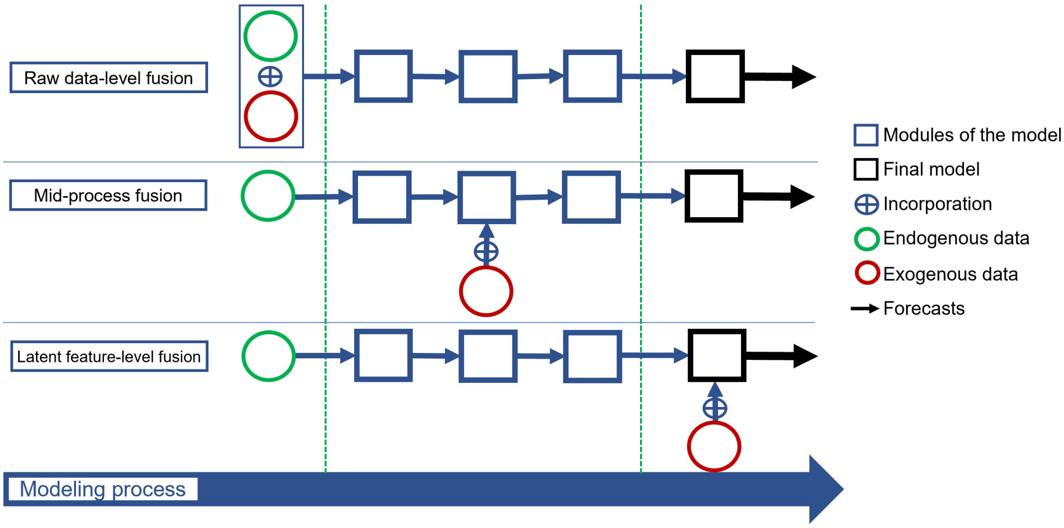


FIGURE 5. **Incorporation approaches taxonomy: It is possible to incorporate exogenous data at the raw data level (fusing them with the endogenous data before modeling), mid-process (fusing them with the features extracted from the endogenous data) or at the latent feature level (incorporating them right before yielding results).**

Function Model [2] (10).

$$\nabla^d \mathbf{z}(t) = \sum_{k=1}^{p}\sum_{l=0}^{\lambda_k} \boldsymbol{\phi}_{kl}\mathbf{W}^{(l)}\nabla^d \mathbf{z}(t-k) + \sum_{k=0}^{s}\sum_{l=0}^{\gamma_k} \boldsymbol{\Omega}_{kl}\mathbf{W}^{(l)}\nabla^d \mathbf{x}(t-b-k) + \boldsymbol{\varepsilon}(t) - \sum_{k=1}^{q}\sum_{l=0}^{\alpha_k} \theta_{kl}\mathbf{W}^{(l)}\boldsymbol{\varepsilon}(t-k) \quad (10)$$

where b is the impact delay parameter, γ_k is the spatial order of the k^{th} time lag of the exogenous variable, s is the exogenous temporal order, and $\boldsymbol{\Omega}_{kl}$ is exogenous parameter at temporal lag k and spatial lag l.

4) MULTI-LEVEL FUSION

When predicting Spatio-Temporal processes, it is possible to incorporate multiple exogenous variables. Each of these features may have a different impact on the prediction accuracy, thus instead of incorporating all the exogenous data at once, some work tend to fuse them at different stages of the modeling process.

Wang et al. for example, used various exogenous features to predict the traffic volume such as weather, holidays and traffic flow information [37]. In their work, flow information was raw data-level fused, and fed to the model along with the endogenous feature. The rest of the exogenous features were concatenated at the latent-feature level. Yao et al. [38] also adopted a multi-stage fusion to incorporate the exogenous data. They used a Flow Gating Mechanism, at the raw-data level, to properly model the spatial dependency of the traffic volume based on the exogenous flow feature, then further fused other external features mid-process.

B. FUSION TECHNIQUES

In addition to specifying the fusion phase, it is necessary to depict the incorporating approaches conforming to the manner they introduce the exogenous data to the model.

1) CONCATENATION

As the most common incorporation technique, concatenation merely entails joining the exogenous data along one of the endogenous data's axes. For example, in both [29] and [50], data instances were transformed from three-dimensional tensors to four-dimensional ones after the concatenation of exogenous features. This incorporation technique can be applied at any phase of the modeling. For instance, Cui et al. [33] employed it at the raw data-level, Yao et al. [30] used it mid process and De Medrano and Aznarte [35] utilized it at the latent-feature level.

2) AGGREGATION

The aggregation approach simply involves adding up endogenous and exogenous values. Accordingly, the ST-ResNet [31] depicted endogenous and exogenous features separately. In fact, a set of three convolution-based modules was used to extract the ST features from the endogenous data, denoted as Z_{Res} and an external component was used to extract the exogenous features from the metadata, denoted as X_{Ext}. To yield the final prediction, the two features were added and then passed through an activation function

$$\hat{Z}(t) = \tanh(Z_{Res} + X_{\text{Ext}}) \quad (11)$$

where tanh is the hyperbolic tangent.

Weighted Aggregation Slightly different from the vanilla aggregation, the weighted one consists of associating weights to the features before adding them. For instance, in the DeepST [34] model, which is similar to the ST-ResNet in the feature extraction phase, instead of merely adding the values, two weight matrices were utilized. That being said, and keeping the same notations as the ST-Resnet, the final prediction was given by:

$$\hat{Z}(t) = \tanh(W_{\text{Res}} \cdot Z_{\text{Res}} + W_{\text{Ext}} \cdot X_{\text{Ext}}) \quad (12)$$

The weight matrices W_{Res} and W_{Ext} are learned parameters.

3) CONDITIONING

This approach involves controlling the input of endogenous data, conforming to the exogenous variables. For example, Yu et al. [43] embedded the exogenous data in the adjacency matrix. The adjacency matrix was parameterized to account different intensities in traffic propagation by weighting its parameters based on the distance and road capacity. Furthermore, the traffic flow information reflects the correlation between different forecast regions [37]. In fact, this exogenous variable explicitly embodies the spatial dependencies that are implicit in the endogenous traffic data.

Gating mechanism This approach, which can be seen as a variant of the conditioning one, consists of controlling the input of endogenous data based on the exogenous variables. For instance, to model the spatial dependency within the traffic volume data, Yao et al. [38] used a local CNN. To this end, the traffic volume information is represented as a $S \times S$ image with a start volume and an end volume channels: $Z(t) \in \mathbb{R}^{S\times S\times 2}$. The formulation of the k-th layer of the local CNN is:

$$Z^{(k)}(t) = \text{ReLU}(W^{(k)} * Z^{(k-1)}(t) + b^{(k)}) \quad (13)$$

where $W^{(k)}$ and $b^{(k)}$ are learned parameters and $Z^{(0)}(t) = Z(t)$. One of the major drawbacks of directly applying the local CNN is that the spatial dependency is stationary and does not effectively reflect the relation between the target region and its neighbors. To mitigate this issue, they exploited the fact that having more flows between two regions strengthens the relation. They designed a Gating Mechanism which encompasses the flow information at every layer of the local CNN. That being the case, the formulation of the k-th layer becomes:

$$Z^{(k)}(t) = \text{ReLU}(W^{(k)} * Z^{(k-1)}(t) + b^{(k)}) \otimes \sigma(X^{k-1}(t)) \quad (14)$$

where $\otimes$ denotes the Hadamard product and $Z^{(k)}(t)$ is the output of k-th layer of a CNN applied on the Flow tensor.

Selecting the appropriate exogenous data incorporation approach and level does not only depend on their type and importance, but also on the model to be employed.

C. EXOGENOUS DATA VS MULTIVARIATE FORECASTING

In the context of Spatio-Temporal forecasting, incorporating exogenous data may seem similar to a multivariate STF model, but there are several differences, on multiple levels, as described next.

1) **Dynamics:** as shown in Table 1, exogenous data can be temporal, spatial, or Spatio Temporal series. In contrast, multi-variables have to be Spatio Temporal to be included in the model.
2) **Incorporation level:** as aforementioned, exogenous data can be integrated into the forecasting model at various levels, such as the raw data level, mid-process, or latent-feature level. Multivariate series, on the other hand, require raw data level incorporation.
3) **Incorporation technique:** several techniques are used to incorporate exogenous variables in the model, including concatenation, conditioning, and aggregation whilst multi-variables can only be concatenated.
4) **Training:** even when exogenous data are concatenated at the raw level, there would still be a difference between multivariate forecasting models and ones that includes exogenous data. The loss function of multivariate forecasting models accounts for the errors in all the variables. As for the models that incorporate exogenous only account for the errors in the target variables.

In summary, incorporating exogenous variables is a valuable technique for enhancing forecasting accuracy by integrating external information without modeling them.

VI. DISCUSSION

There has been a tremendous growth in the number of studies that tackle the Spatio-temporal forecasting problem. This growth was driven by the importance of forecasting the Spatio-Temporal processes in our understanding of the world we live in. However, many open challenges still exist and should be addressed in future work to enhance the STF task solving. In this section, we categorize the open challenges into three main subsets: Data related, Model related, and Exogenous data incorporation-related challenges.

A. DATA RELATED ISSUES

1) DATA QUALITY AND QUANTITY

The amount of ST data that are generated and collected is increasing. Yet, there are still some perturbations during the data collection phase owing to technical deficiencies and human error. This leads to the decrease of data quality, which impacts the modeling and therefore forecasting quality. Thus great effort is required to adequately pre-process data by cleaning it and dealing with missing instances. Moreover, since some complex models require massive amounts of data, especially when merging the exogenous features with the endogenous ones, large data sets should be available. However, to the best of our knowledge, open-access public data sets are very limited. More work should be devoted to providing such data sets to help researchers propose and develop novel models.

2) DATA REPRESENTATION AND VISUALIZATION

As sources of ST data vary vaguely, different types of these latter can be used in solving a single forecasting problem. There is still a lack of in-depth research on how to select the appropriate representation, especially that it also depends on the models to be exploited. Furthermore, when addressing a STF problem, fully understanding the data that are being handled is crucial. An effective way to do so is visualizing the data as it provides interpretations and visual summaries. Many researchers came up in the recent years with techniques and tools to visualize spatial data especially in the scope of GIS (Geographic Information Systems), but visualizing the dynamics of ST data has not been thoroughly studied and needs more research attention in future work.

B. MODEL RELATED ISSUES

1) MODEL SELECTION

Among the papers that we have reviewed, we noticed a significant imbalance in the distribution of the models that are exploited. Deep Learning-based models were used in the majority of the studies. However, it should be noted that while DL models outperform the conventional ones in various data mining tasks, they are not the go-to models for tackling the STF problem. In fact, the empirical results presented in [166] demonstrate that conventional models can be superior for simple data such as univariate time-series forecasting. This finding proves that many aspects should be taken into consideration when selecting the model to be employed, and that Statistical and vanilla ML models should not be disregarded. Furthermore, we have noticed that little focus has been granted to the employment of hybrid models.

2) EXPLOITING NEW MECHANISMS

Despite the advances in the STF models that are developed in the literature, several aspects should get further attention. Namely, enhancing the models' interpretability, speed, especially for applications requiring real time predictions, and ease of implementation to make them more accessible to STF practitioners. Thus, it is necessary to employ techniques and mechanisms that are shown to be effective in other data mining tasks.

3) COVERING NEW APPLICATION FIELDS

Although STF models have been widely implemented in different application areas, it is necessary to widen the scope of its uses by covering new application fields. It is also important to increase the number of studies in areas where it is rarely employed, namely medicine and sport. Indeed, throughout our investigation, we have noticed that the primary field benefiting from the advancements in STF is the transportation field.

C. EXOGENOUS DATA INCORPORATION ISSUES

Although there has been a considerable growth in studies that use exogenous data when forecasting Spatio-Temporal processes, there is still a shortage of study on the selection of the adequate exogenous data and their incorporation. In fact, it is crucial to only add data that will not harm the model, and to pay great attention to the trade-off between model accuracy and speed. Moreover, more focus should be granted to how and when to fuse the exogenous data. To the best of our knowledge, there are no specific guidelines that emphasize the choice of the incorporation technique and level. As a result, it is important to set a framework that outlines how to choose the relevant exogenous features, how to incorporate them and how to assess their impact on the models.

VII. CONCLUSION

In this work, we conducted a comprehensive survey of the up-to date literature on data-driven Spatio-Temporal Forecasting models. We first investigated the data properties to establish the groundwork for our survey and identified the several problem formulations that fall within the STF task. Next, we extensively reviewed data-driven STF models applied in various application fields. We categorized these models into three major categories: Statistical, Machine Learning-based and Deep Learning-based models. We also emphasized the model selection criteria in order to guide researchers in selecting the appropriate model. In addition, we proposed a novel full view taxonomy that organizes and synthesizes the incorporation approaches used in the literature. Finally, we discussed the open STF challenges and highlighted promising future research directions.

DECLARATION OF COMPETING INTEREST

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this article.

REFERENCES

[1] S. Wang, J. Cao, and P. S. Yu, "Deep learning for spatio-temporal data mining: A survey," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 8, pp. 3681–3700, Aug. 2022.

[2] N. Andayani, I. M. Sumertajaya, B. N. Ruchjana, and M. N. Aidi, "Development of space time model with exogenous variable by using transfer function model approach on the Rice price data," *Appl. Math. Sci.*, vol. 11, no. 36, pp. 1779–1792, 2017.

[3] N. Andayani, I. M. Sumertajaya, B. N. Ruchjana, and M. N. Aidi, "Comparison of GSTARIMA and GSTARIMA-x model by using transfer function model approach to Rice price data," *IOP Conf. Ser., Earth Environ. Sci.*, vol. 187, Nov. 2018, Art. no. 012052.

[4] B. Wang, D. Zhang, D. Zhang, P. J. Brantingham, and A. L. Bertozzi, "Deep learning for real time crime forecasting," 2017, *arXiv:1707.03340*.

[5] A. R. Khairuddin, R. Alwee, and H. Harun, "Comparative study on artificial intelligence techniques in crime forecasting," *Appl. Mech. Mater.*, vol. 892, pp. 94–100, Jun. 2019.

[6] S. Reza, M. C. Ferreira, J. J. M. Machado, and J. M. R. S. Tavares, "A multi-head attention-based transformer model for traffic flow forecasting with a comparative analysis to recurrent neural networks," *Expert Syst. Appl.*, vol. 202, Sep. 2022, Art. no. 117275.

[7] R. Asadi, "Deep learning models for spatio-temporal forecasting and analysis," Ph.D. thesis, Dept. Comput. Sci., Univ. California Irvine, 2020.

[8] Y. Kim, B. Kim, and S. Ahn, "Application of spatiotemporal transformer model to improve prediction performance of particulate matter concentration," *J. Intell. Inf. Syst.*, vol. 28, no. 1, pp. 329–352, 2022.

[9] F. Amato, F. Guignard, S. Robert, and M. Kanevski, "A novel framework for spatio-temporal prediction of environmental data using deep learning," *Sci. Rep.*, vol. 10, no. 1, pp. 1–11, Dec. 2020.

[10] P. E. Pfeifer and S. J. Deutsch, "A STARIMA model-building procedure with application to description and regional forecasting," *Trans. Inst. Brit. Geographers*, vol. 5, no. 3, p. 330, 1980.

[11] A. E. Gelfand, S. K. Ghosh, J. R. Knight, and C. F. Sirmans, "Spatio-temporal modeling of residential sales data," *J. Bus. Econ. Statist.*, vol. 16, no. 3, p. 312, Jul. 1998.

[12] A. Ziat, E. Delasalles, L. Denoyer, and P. Gallinari, "Spatio-temporal neural networks for space-time series forecasting and relations discovery," in *Proc. IEEE Int. Conf. Data Mining (ICDM)*, Nov. 2017, pp. 705–714.

[13] J. M. K. Aheto, B. M. Taylor, T. J. Keegan, and P. J. Diggle, "Modelling and forecasting spatio-temporal variation in the risk of chronic malnutrition among under-five children in Ghana," *Spatial Spatio-Temporal Epidemiol.*, vol. 21, pp. 37–46, Jun. 2017.

[14] A. Hamdi, K. Shaban, A. Erradi, A. Mohamed, S. K. Rumi, and F. D. Salim, "Spatiotemporal data mining: A survey on challenges and open problems," *Artif. Intell. Rev.*, vol. 55, no. 2, pp. 1441–1488, Feb. 2022.

[15] A. Zagouras, H. T. C. Pedro, and C. F. M. Coimbra, "On the role of lagged exogenous variables and spatio-temporal correlations in improving the accuracy of solar forecasting methods," *Renew. Energy*, vol. 78, pp. 203–218, Jun. 2015.

[16] K. Bansak, J. Ferwerda, J. Hainmueller, A. Dillon, D. Hangartner, D. Lawrence, and J. Weinstein, "Improving refugee integration through data-driven algorithmic assignment," *Science*, vol. 359, no. 6373, pp. 325–329, Jan. 2018.

[17] J. Kang, X. Guo, L. Fang, X. Wang, and Z. Fan, "Integration of internet search data to predict tourism trends using spatial–temporal XGBoost composite model," *Int. J. Geographical Inf. Sci.*, vol. 36, no. 2, pp. 236–252, Feb. 2022.

[18] G. Atluri, A. Karpatne, and V. Kumar, "Spatio-temporal data mining: A survey of problems and methods," *ACM Comput. Surv.*, vol. 51, no. 4, pp. 1–41, Jul. 2019.

[19] X. Shi and D.-Y. Yeung, "Machine learning for spatiotemporal sequence forecasting: A survey," 2018, *arXiv:1808.06865*.

[20] X. Yin, G. Wu, J. Wei, Y. Shen, H. Qi, and B. Yin, "Deep learning on traffic prediction: Methods, analysis, and future directions," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 6, pp. 4927–4943, Jun. 2022.

[21] S. Aslam, H. Herodotou, S. M. Mohsin, N. Javaid, N. Ashraf, and S. Aslam, "A survey on deep learning methods for power load and renewable energy forecasting in smart microgrids," *Renew. Sustain. Energy Rev.*, vol. 144, Jul. 2021, Art. no. 110992.

[22] A. Ermagun and D. Levinson, "Spatiotemporal traffic forecasting: Review and proposed directions," *Transp. Rev.*, vol. 38, no. 6, pp. 786–814, Nov. 2018.

[23] A. Boukerche and J. Wang, "Machine learning-based traffic prediction models for intelligent transportation systems," *Comput. Netw.*, vol. 181, Nov. 2020, Art. no. 107530.

[24] H. Yuan and G. Li, "A survey of traffic prediction: From spatio-temporal data to intelligent transportation," *Data Sci. Eng.*, vol. 6, no. 1, pp. 63–85, Mar. 2021.

[25] P. Xie, T. Li, J. Liu, S. Du, X. Yang, and J. Zhang, "Urban flow prediction from spatiotemporal data using machine learning: A survey," *Inf. Fusion*, vol. 59, pp. 1–12, Jul. 2020.

[26] A. Tascikaraoglu, "Evaluation of spatio-temporal forecasting methods in various smart city applications," *Renew. Sustain. Energy Rev.*, vol. 82, pp. 424–435, Feb. 2018.

[27] L. Xu, N. Chen, Z. Chen, C. Zhang, and H. Yu, "Spatiotemporal forecasting in Earth system science: Methods, uncertainties, predictability and future directions," *Earth-Sci. Rev.*, vol. 222, Nov. 2021, Art. no. 103828.

[28] Y. Rong, Z. Xu, R. Yan, and X. Ma, "Du-parking: Spatio-temporal big data tells you realtime parking availability," in *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Jul. 2018, pp. 646–654.

[29] 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.

[30] H. Yao, F. Wu, J. Ke, X. Tang, Y. Jia, S. Lu, P. Gong, J. Ye, and Z. Li, "Deep multi-view spatial–temporal network for taxi demand prediction," in *Proc. AAAI Conf. Artif. Intell.*, vol. 32, 2018, pp. 2588–2595.

[31] J. Zhang, Y. Zheng, and D. Qi, "Deep spatio-temporal residual networks for citywide crowd flows prediction," in *Proc. AAAI Conf. Artif. Intell.*, vol. 31, 2017, pp. 1655–1661.

[32] W. Alajali, W. Zhou, S. Wen, and Y. Wang, "Intersection traffic prediction using decision tree models," *Symmetry*, vol. 10, no. 9, p. 386, Sep. 2018.

[33] Z. Cui, R. Ke, Z. Pu, and Y. Wang, "Deep bidirectional and unidirectional LSTM recurrent neural network for network-wide traffic speed prediction," 2018, *arXiv:1801.02143*.

[34] J. Zhang, Y. Zheng, D. Qi, R. Li, and X. Yi, "DNN-based prediction model for spatio-temporal data," in *Proc. 24th ACM SIGSPATIAL Int. Conf. Adv. Geographic Inf. Syst.*, Oct. 2016, pp. 1–4.

[35] R. de Medrano and J. L. Aznarte, "A spatio-temporal spot-forecasting framework for urban traffic prediction," 2020, *arXiv:2003.13977*.

[36] D. Wang, W. Cao, J. Li, and J. Ye, "DeepSD: Supply-demand prediction for online car-hailing services using deep neural networks," in *Proc. IEEE 33rd Int. Conf. Data Eng. (ICDE)*, Apr. 2017, pp. 243–254.

[37] J. Wang, W. Zhu, Y. Sun, and C. Tian, "An effective dynamic spatiotemporal framework with external features information for traffic prediction," *Int. J. Speech Technol.*, vol. 51, no. 6, pp. 3159–3173, Jun. 2021.

[38] H. Yao, X. Tang, H. Wei, G. Zheng, and Z. Li, "Revisiting spatial–temporal similarity: A deep learning framework for traffic prediction," in *Proc. Conf. Artif. Intell.*, vol. 33, 2019, pp. 5668–5675.

[39] H. Yu, Z. Wu, S. Wang, Y. Wang, and X. Ma, "Spatiotemporal recurrent convolutional networks for traffic prediction in transportation networks," *Sensors*, vol. 17, no. 7, p. 1501, Jun. 2017.

[40] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-k. Wong, and W.-c. Woo, "Convolutional LSTM network: A machine learning approach for precipitation nowcasting," 2015, *arXiv:1506.04214*.

[41] Y. Wang, H. Wu, J. Zhang, Z. Gao, J. Wang, P. S. Yu, and M. Long, "PredRNN: A recurrent neural network for spatiotemporal predictive learning," in *Proc. 31st Conf. Neural Inf. Process. Syst. (NIPS)*, 2017, pp. 879–888.

[42] J. Grigsby, Z. Wang, N. Nguyen, and Y. Qi, "Long-range transformers for dynamic spatiotemporal forecasting," 2021, *arXiv:2109.12218*.

[43] B. Yu, Y. Lee, and K. Sohn, "Forecasting road traffic speeds by considering area-wide spatio-temporal dependencies based on a graph convolutional neural network (GCN)," *Transp. Res. C, Emerg. Technol.*, vol. 114, pp. 189–204, May 2020.

[44] L. Zhao, Y. Song, C. Zhang, Y. Liu, P. Wang, T. Lin, M. Deng, and H. Li, "T-GCN: A temporal graph convolutional network for traffic prediction," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 9, pp. 3848–3858, Sep. 2020.

[45] M. Khodayar, S. Mohammadi, M. E. Khodayar, J. Wang, and G. Liu, "Convolutional graph autoencoder: A generative deep neural network for probabilistic spatio-temporal solar irradiance forecasting," *IEEE Trans. Sustain. Energy*, vol. 11, no. 2, pp. 571–583, Apr. 2020.

[46] E. Akleman, "Deep learning," *Computer*, vol. 53, no. 9, p. 17, Sep. 2020.

[47] Z. Ma, H. Zhang, and J. Liu, "MS-RNN: A flexible multi-scale framework for spatiotemporal predictive learning," 2022, *arXiv:2206.03010*.

[48] W. Zhang, L. Han, J. Sun, H. Guo, and J. Dai, "Application of multi-channel 3D-cube successive convolution network for convective storm nowcasting," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Dec. 2019, pp. 1705–1710.

[49] Y. Liang, S. Ke, J. Zhang, X. Yi, and Y. Zheng, "GeoMAN: Multi-level attention networks for geo-sensory time series prediction," in *Proc. 27th Int. Joint Conf. Artif. Intell.*, Jul. 2018, pp. 3428–3434.

[50] F. Yu, H. Hao, and Q. Li, "An ensemble 3D convolutional neural network for spatiotemporal soil temperature forecasting," *Sustainability*, vol. 13, no. 16, p. 9174, Aug. 2021.

[51] K. Zhou, Y. Zheng, B. Li, W. Dong, and X. Zhang, "Forecasting different types of convective weather: A deep learning approach," *J. Meteorolog. Res.*, vol. 33, no. 5, pp. 797–809, Oct. 2019.

[52] X. Wu, Y. Dong, C. Huang, J. Xu, D. Wang, and N. V. Chawla, "UAPD: Predicting urban anomalies from spatial–temporal data," in *Proc. Joint Eur. Conf. Mach. Learn. Knowl. Discovery Databases*, Skopje, Macedonia: Springer, Sep. 2017, pp. 622–638.

[53] B. Liao, J. Zhang, M. Cai, S. Tang, Y. Gao, C. Wu, S. Yang, W. Zhu, Y. Guo, and F. Wu, "Des.-ResNet: A deep spatiotemporal residual network for hotspot traffic speed prediction," in *Proc. 26th ACM Int. Conf. Multimedia*, Oct. 2018, pp. 1883–1891.

[54] M. Zakroum, J. François, I. Chrisment, and M. Ghogho, "Monitoring network telescopes and inferring anomalous traffic through the prediction of probing rates," *IEEE Trans. Netw. Service Manage.*, vol. 19, no. 4, pp. 5170–5182, Dec. 2022.

[55] H. Ge, S. Li, R. Cheng, and Z. Chen, "Self-attention ConvLSTM for spatiotemporal forecasting of short-term online car-hailing demand," *Sustainability*, vol. 14, no. 12, p. 7371, Jun. 2022.

[56] P. Duan, G. Mao, W. Liang, and D. Zhang, "A unified spatio-temporal model for short-term traffic flow prediction," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 9, pp. 3212–3223, Sep. 2019.

[57] M. Khodayar and J. Wang, "Spatio-temporal graph deep neural network for short-term wind speed forecasting," *IEEE Trans. Sustain. Energy*, vol. 10, no. 2, pp. 670–681, Apr. 2019.

[58] J. Liang and W. Tang, "Ultra-short-term spatiotemporal forecasting of renewable resources: An attention temporal convolutional network-based approach," *IEEE Trans. Smart Grid*, vol. 13, no. 5, pp. 3798–3812, Sep. 2022.

[59] W. Shao, Z. Jin, S. Wang, Y. Kang, X. Xiao, H. Menouar, Z. Zhang, J. Zhang, and F. Salim, "Long-term spatio-temporal forecasting via dynamic multiple-graph attention," 2022, *arXiv:2204.11008*.

[60] Z. He, C. Chow, and J. Zhang, "STCNN: A spatio-temporal convolutional neural network for long-term traffic prediction," in *Proc. 20th IEEE Int. Conf. Mobile Data Manage. (MDM)*, Jun. 2019, pp. 226–233.

[61] S. Modi, J. Bhattacharya, and P. Basak, "Multistep traffic speed prediction: A deep learning based approach using latent space mapping considering spatio-temporal dependencies," *Expert Syst. Appl.*, vol. 189, Mar. 2022, Art. no. 116140.

[62] Z. Li, Y. Zhang, D. Guo, X. Zhou, X. Wang, and L. Zhu, "Long-term traffic forecasting based on adaptive graph cross strided convolution network," *Int. J. Speech Technol.*, vol. 53, no. 4, pp. 3672–3686, Feb. 2023.

[63] A. Oluwasanmi, M. U. Aftab, Z. Qin, M. S. Sarfraz, Y. Yu, and H. T. Rauf, "Multi-head spatiotemporal attention graph convolutional network for traffic prediction," *Sensors*, vol. 23, no. 8, p. 3836, 2023.

[64] L. Li, Y. Dai, W. Shangguan, N. Wei, Z. Wei, and S. Gupta, "Multistep forecasting of soil moisture using spatiotemporal deep encoder–decoder networks," *J. Hydrometeorol.*, vol. 23, no. 3, pp. 337–350, Jan. 2022.

[65] K. Zhang, J. Thé, G. Xie, and H. Yu, "Multi-step ahead forecasting of regional air quality using spatial–temporal deep neural networks: A case study of Huaihai Economic Zone," *J. Cleaner Prod.*, vol. 277, Dec. 2020, Art. no. 123231.

[66] Y. Qi, Q. Li, H. Karimian, and D. Liu, "A hybrid model for spatiotemporal forecasting of $PM_{2.5}$ based on graph convolutional neural network and long short-term memory," *Sci. Total Environ.*, vol. 664, pp. 1–10, May 2019.

[67] R. He, Y. Liu, Y. Xiao, X. Lu, and S. Zhang, "Deep spatio-temporal 3D densenet with multiscale ConvLSTM-resnet network for citywide traffic flow forecasting," *Knowl.-Based Syst.*, vol. 250, Aug. 2022, Art. no. 109054.

[68] G. E. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung, *Time Series Analysis: Forecasting and Control*. Hoboken, NJ, USA: Wiley, 2015.

[69] R. Hyndman, A. B. Koehler, J. K. Ord, and R. D. Snyder, *Forecasting With Exponential Smoothing: The State Space Approach*. Cham, Switzerland: Springer, 2008.

[70] J. Cartlidge, S. Gong, R. Bai, Y. Yue, Q. Li, and G. Qiu, "Spatio-temporal prediction of shopping behaviours using taxi trajectory data," in *Proc. IEEE 3rd Int. Conf. Big Data Anal. (ICBDA)*, Mar. 2018, pp. 112–116.

[71] Y. Kamarianakis and P. Prastacos, "Forecasting traffic flow conditions in an urban network: Comparison of multivariate and univariate approaches," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 1857, no. 1, pp. 74–84, Jan. 2003.

[72] F. Santos-Marquez, "Spatial beta-convergence forecasting models: Evidence from municipal homicide rates in Colombia," *J. Forecasting*, vol. 41, no. 2, pp. 294–302, Mar. 2022.

[73] F. A. Awwad, M. A. Mohamoud, and M. R. Abonazel, "Estimating COVID-19 cases in Makkah region of Saudi Arabia: Space-time ARIMA modeling," *PLoS ONE*, vol. 16, no. 4, Apr. 2021, Art. no. e0250149.

[74] A. D. Cliff and J. K. Ord, "Space-time modelling with an application to regional forecasting," *Trans. Inst. Brit. Geographers*, vol. 64, p. 119, Mar. 1975.

[75] P. E. Pfeifer and S. J. Deutsch, "Identification and interpretation of first order space-time ARMA models," *Technometrics*, vol. 22, no. 3, pp. 397–408, Aug. 1980.

[76] S.-L. Lin, H.-Q. Huang, D.-Q. Zhu, and T.-Z. Wang, "The application of space-time ARIMA model on traffic flow forecasting," in *Proc. Int. Conf. Mach. Learn. Cybern.*, vol. 6, Jul. 2009, pp. 3408–3412.

[77] Y. Kamarianakis and P. Prastacos, *Spatial Time Series Modeling: A Review of the Proposed Methodologies*. Urbana, IL, USA: Regional Economics Applications Laboratory (2003).

[78] S. Borovkova, H. Lopuha, and B. Ruchjana, "Generalized s-tar with random weights," in *Proc. 17th Int. Workshop Stat. Modeling*, 2002, pp. 143–151.

[79] S. Setiawan and M. Prastuti, "S-GSTAR-SUR model for seasonal spatio temporal data forecasting," *Malaysian J. Math. Sci.*, vol. 10, pp. 53–65, Mar. 2016.

[80] S. Suhartono, D. D. Prastyo, H. Kuswanto, and M. H. Lee, "Comparison between VAR, GSTAR, FFNN-VAR and FFNN-GSTAR models for forecasting oil production," *MATEMATIKA*, vol. 34, no. 1, pp. 103–111, May 2018.

[81] V. Di Giacinto, "A generalized space-time ARMA model with an application to regional unemployment analysis in Italy," *Int. Regional Sci. Rev.*, vol. 29, no. 2, pp. 159–198, Apr. 2006.

[82] M. A. Novianto, D. D. Prastyo, and A. Suharsono, "GSTARIX model for forecasting spatio-temporal data with trend, seasonal and intervention," *J. Phys., Conf. Ser.*, vol. 1097, Sep. 2018, Art. no. 012076.

[83] S. S. Prasetiyowati, Y. Sibaroni, and S. Carolina, "Prediction and mapping of air pollution in Bandung using generalized space time autoregressive and simple kriging," in *Proc. Int. Conf. Data Sci. Appl. (ICoDSA)*, Aug. 2020, pp. 1–8.

[84] N. Aulia and D. R. S. Saputro, "Generalized space time autoregressive integrated moving average with exogenous (GSTARIMA-X) models," *J. Phys., Conf. Ser.*, vol. 1808, no. 1, Mar. 2021, Art. no. 012052.

[85] Y. Zhao, L. Ge, Y. Zhou, Z. Sun, E. Zheng, X. Wang, Y. Huang, and H. Cheng, "A new seasonal difference space-time autoregressive integrated moving average (SD-STARIMA) model and spatiotemporal trend prediction analysis for hemorrhagic fever with renal syndrome (HFRS)," *PLoS ONE*, vol. 13, no. 11, Nov. 2018, Art. no. e0207518.

[86] R. J. Barro, X. Sala-I-Martin, O. J. Blanchard, and R. E. Hall, "Convergence across states and regions," *Brookings Papers Econ. Activity*, vol. 1991, no. 1, p. 107, 1991.

[87] H. Hassani and R. Mahmoudvand, "Multivariate singular spectrum analysis: A general view and new vector forecasting approach," *Int. J. Energy Statist.*, vol. 1, no. 1, pp. 55–83, Mar. 2013.

[88] D. S. Broomhead and G. P. King, "Extracting qualitative dynamics from experimental data," *Phys. D, Nonlinear Phenomena*, vol. 20, nos. 2–3, pp. 217–236, Jun. 1986.

[89] A. Agarwal, A. Alomar, and D. Shah, "On multivariate singular spectrum analysis and its variants," 2020, *arXiv:2006.13448*.

[90] H. O. Cekim, C. O. Güney, Ö. Şentürk, G. Özel, and K. Özkan, "A novel approach for predicting burned forest area," *Natural Hazards*, vol. 105, no. 2, pp. 2187–2201, Jan. 2021.

[91] R. Mahmoudvand, D. Konstantinides, and P. C. Rodrigues, "Forecasting mortality rate by multivariate singular spectrum analysis," *Appl. Stochastic Models Bus. Ind.*, vol. 33, no. 6, pp. 717–732, Nov. 2017.

[92] A. Zhigljavsky, H. Hassani, and S. Heravi, "Forecasting European industrial production with multivariate singular spectrum analysis," *Int. J. Forecasting*, vol. 25, no. 1, pp. 103–118, 2009.

[93] O. Ohashi and L. Torgo, "Wind speed forecasting using spatio-temporal indicators," in *Proc. ECAI*, 2012, pp. 975–980.

[94] A. Appice, S. Pravilovic, D. Malerba, and A. Lanza, "Enhancing regression models with spatio-temporal indicator additions," in *Congress of the Italian Association for Artificial Intelligence*. Cham, Switzerland: Springer, 2013, pp. 433–444.

[95] A. Agafonov and A. Yumaganov, "Performance comparison of machine learning methods in the bus arrival time prediction problem," in *Proc. 5th Int. Conf. Inf. Technol. Nanotechnol.*, 2019, pp. 57–62.

[96] D. Yang, T. Heaney, A. Tonon, L. Wang, and P. Cudré-Mauroux, "CrimeTelescope: Crime hotspot prediction based on urban and social media data fusion," *World Wide Web*, vol. 21, no. 5, pp. 1323–1347, Sep. 2018.

[97] C.-H. Yu, W. Ding, P. Chen, and M. Morabito, "Crime forecasting using spatio-temporal pattern with ensemble learning," in *Proc. Pacific–Asia Conf. Knowl. Discovery Data Mining*. Cham, Switzerland: Springer, 2014, pp. 174–185.

[98] S. K. Rumi, K. Deng, and F. D. Salim, "Crime event prediction with dynamic features," *EPJ Data Sci.*, vol. 7, no. 1, p. 43, 2018.

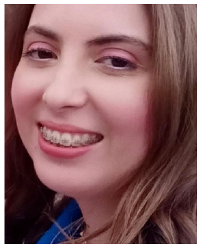
[99] A. Pozdnoukhov, G. Matasci, M. Kanevski, and R. S. Purves, "Spatio-temporal avalanche forecasting with support vector machines," *Natural Hazards Earth Syst. Sci.*, vol. 11, no. 2, pp. 367–382, Feb. 2011.

[100] T. Wu, K. Xie, G. Song, and C. Hu, "A multiple SVR approach with time lags for traffic flow prediction," in *Proc. 11th Int. IEEE Conf. Intell. Transp. Syst.*, Oct. 2008, pp. 228–233.

[101] L. Song, S. Pang, I. Longley, G. Olivares, and A. Sarrafzadeh, "Spatio-temporal $PM_{2.5}$ prediction by spatial data aided incremental support vector regression," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2014, pp. 623–630.
[102] B. P. Ashwini, R. Sumathi, and H. S. Sudhira, "Bus travel time prediction: A comparative study of linear and non-linear machine learning models," *J. Phys., Conf. Ser.*, vol. 2161, no. 1, Jan. 2022, Art. no. 012053.
[103] B. Sun, T. Sun, and P. Jiao, "Spatio-temporal segmented traffic flow prediction with ANPRS data based on improved XGBoost," *J. Adv. Transp.*, vol. 2021, pp. 1–24, May 2021.
[104] B. Qiu and W. Fan, "Machine learning based short-term travel time prediction: Numerical results and comparative analyses," *Sustainability*, vol. 13, no. 13, p. 7454, Jul. 2021.
[105] I. Ahmed, I. Kumara, V. Reshadat, A. S. M. Kayes, W.-J. van den Heuvel, and D. A. Tamburri, "Travel time prediction and explanation with spatio-temporal features: A comparative study," *Electronics*, vol. 11, no. 1, p. 106, Dec. 2021.
[106] B. P. Ashwini, R. Sumathi, and H. S. Sudhira, "A dynamic model for bus arrival time estimation based on spatial patterns using machine learning," 2022, *arXiv:2210.00733*.
[107] F. Zhang, X. Zhu, T. Hu, W. Guo, C. Chen, and L. Liu, "Urban link travel time prediction based on a gradient boosting method considering spatiotemporal correlations," *ISPRS Int. J. Geo-Inf.*, vol. 5, no. 11, p. 201, Nov. 2016.
[108] Y. Chen, X. Li, K. Huang, M. Luo, and M. Gao, "High-resolution gridded population projections for China under the shared socioeconomic pathways," *Earth's Future*, vol. 8, no. 6, Jun. 2020, Art. no. e2020EF001491.
[109] Z. Ghaemi, A. Alimohammadi, and M. Farnaghi, "LaSVM-based big data learning system for dynamic prediction of air pollution in Tehran," *Environ. Monitor. Assessment*, vol. 190, no. 5, pp. 1–17, May 2018.
[110] A. Bordes, S. Ertekin, J. Weston, L. Botton, and N. Cristianini, "Fast kernel classifiers with online and active learning," *J. Mach. Learn. Res.*, vol. 6, no. 9, pp. 1579–1619, 2005.
[111] J. L. Elman, "Finding structure in time," *Cognit. Sci.*, vol. 14, no. 2, pp. 179–211, Mar. 1990.
[112] H. Shi, Y. Yue, and Y. Zhou, "The comparison between two different algorithms of spatio-temporal forecasting for traffic flow prediction," in *Proc. Int. Conf. Comput. Urban Planning Urban Manage.*, Cham, Switzerland: Springer, 2019, pp. 321–345.
[113] A. Ghaderi, B. M. Sanandaji, and F. Ghaderi, "Deep forecast: Deep learning-based spatio-temporal forecasting," 2017, *arXiv:1707.08110*.
[114] X. Shi, S. Huang, Q. Huang, X. Lei, J. Li, P. Li, and M. Yang, "Deep-learning-based wind speed forecasting considering spatial–temporal correlations with adjacent wind turbines," *J. Coastal Res.*, vol. 93, no. sp1, p. 623, Sep. 2019.
[115] J. Liu, L. Xu, and N. Chen, "A spatiotemporal deep learning model ST-LSTM-SA for hourly rainfall forecasting using radar echo images," *J. Hydrol.*, vol. 609, Jun. 2022, Art. no. 127748.
[116] A. Anshuka, R. Chandra, A. J. V. Buzacott, D. Sanderson, and F. F. van Ogtrop, "Spatio temporal hydrological extreme forecasting framework using LSTM deep learning model," *Stochastic Environ. Res. Risk Assessment*, vol. 36, no. 10, pp. 3467–3485, Oct. 2022.
[117] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
[118] S. Bai, J. Z. Kolter, and V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," 2018, *arXiv:1803.01271*.
[119] J. Fan, K. Zhang, Y. Huang, Y. Zhu, and B. Chen, "Parallel spatio-temporal attention-based TCN for multivariate time series prediction," *Neural Comput. Appl.*, vol. 35, no. 18, pp. 13109–13118, Jun. 2023.
[120] Y.-Y. Hong and T. R. A. Satriani, "Day-ahead spatiotemporal wind speed forecasting using robust design-based deep learning neural network," *Energy*, vol. 209, Oct. 2020, Art. no. 118441.
[121] J. Zhang, Y. Zheng, D. Qi, R. Li, X. Yi, and T. Li, "Predicting citywide crowd flows using deep spatio-temporal residual networks," *Artif. Intell.*, vol. 259, pp. 147–166, Jun. 2018.
[122] H. Li, X. Li, L. Su, D. Jin, J. Huang, and D. Huang, "Deep spatio-temporal adaptive 3D convolutional neural networks for traffic flow prediction," *ACM Trans. Intell. Syst. Technol.*, vol. 13, no. 2, pp. 1–21, Apr. 2022.
[123] Y. A. D. Djilali, M. Sayah, K. McGuinness, and N. E. O'Connor, "3DSAL: An efficient 3D-CNN architecture for video saliency prediction," in *Proc. 15th Int. Conf. Comput. Vision Theory Appl.*, Feb. 2020, pp. 1–10.
[124] M. A. Ahajjam, D. B. Licea, M. Ghogho, and A. Kobbane, "Experimental investigation of variational mode decomposition and deep learning for short-term multi-horizon residential electric load forecasting," *Appl. Energy*, vol. 326, Nov. 2022, Art. no. 119963.
[125] S. Gaci, "A new ensemble empirical mode decomposition (EEMD) denoising method for seismic signals," *Energy Proc.*, vol. 97, pp. 84–91, Nov. 2016.
[126] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," 2016, *arXiv:1609.02907*.
[127] M. Wang, Y. Li, Y. Zhang, and L. Jia, "Spatio-temporal graph convolutional neural network for remaining useful life estimation of aircraft engines," *Aerosp. Syst.*, vol. 4, no. 1, pp. 29–36, Mar. 2021.
[128] J. Gu, Z. Jia, T. Cai, X. Song, and A. Mahmood, "Dynamic correlation adjacency-matrix-based graph neural networks for traffic flow prediction," *Sensors*, vol. 23, no. 6, p. 2897, Mar. 2023.
[129] Z. Duan, H. Xu, Y. Huang, J. Feng, and Y. Wang, "Multivariate time series forecasting with transfer entropy graph," *Tsinghua Sci. Technol.*, vol. 28, no. 1, pp. 141–149, Feb. 2023.
[130] S. Guo, Y. Lin, N. Feng, C. Song, and H. Wan, "Attention based spatial–temporal graph convolutional networks for traffic flow forecasting," in *Proc. AAAI*, vol. 33, 2019, pp. 922–929.
[131] Z. Wu, S. Pan, G. Long, J. Jiang, X. Chang, and C. Zhang, "Connecting the dots: Multivariate time series forecasting with graph neural networks," in *Proc. 26th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2020, pp. 753–763.
[132] Y. Xu, Y. Lu, C. Ji, and Q. Zhang, "Adaptive graph fusion convolutional recurrent network for traffic forecasting," *IEEE Internet Things J.*, early access, Feb. 13, 2023, doi: 10.1109/JIOT.2023.3244182.
[133] Z. He, C. Zhao, and Y. Huang, "Multivariate time series deep spatiotemporal forecasting with graph neural network," *Appl. Sci.*, vol. 12, no. 11, p. 5731, Jun. 2022.
[134] C. Weikang, L. Yawen, X. Zhe, L. Ang, and W. Guobin, "Spatial–temporal adaptive graph convolution with attention network for traffic forecasting," 2022, *arXiv:2206.03128*.
[135] Y. Wang, Z. Duan, Y. Huang, H. Xu, J. Feng, and A. Ren, "MTHetGNN: A heterogeneous graph embedding framework for multivariate time series forecasting," *Pattern Recognit. Lett.*, vol. 153, pp. 151–158, Jan. 2022.
[136] Y. Seo, M. Defferrard, P. Vandergheynst, and X. Bresson, "Structured sequence modeling with graph convolutional recurrent networks," in *Proc. Int. Conf. Neural Inf. Process.* Cham, Switzerland: Springer, 2018, pp. 362–373.
[137] X. Zhu, Y. Lin, Y. He, K.-L. Tsui, P. W. Chan, and L. Li, "Short-term nationwide airport throughput prediction with graph attention recurrent neural network," *Frontiers Artif. Intell.*, vol. 5, p. 105, Jun. 2022.
[138] C. Li, W. Zheng, and P. Ge, "Tourism demand forecasting with spatiotemporal features," *Ann. Tourism Res.*, vol. 94, May 2022, Art. no. 103384.
[139] J. Yang, J. Li, L. Wei, L. Gao, and F. Mao, "Spatiotemporal DeepWalk gated recurrent neural network: A deep learning framework for traffic learning and forecasting," *J. Adv. Transp.*, vol. 2022, pp. 1–11, Apr. 2022.
[140] X. Wang, X. Xu, Y. Wu, and J. Liu, "An effective spatiotemporal deep learning framework model for short-term passenger flow prediction," *Soft Comput.*, vol. 26, no. 12, pp. 5523–5538, Jun. 2022.
[141] F. B. Hüttel, I. Peled, F. Rodrigues, and F. C. Pereira, "Deep spatio-temporal forecasting of electrical vehicle charging demand," 2021, *arXiv:2106.10940*.
[142] Z. Cui, K. Henrickson, R. Ke, and Y. Wang, "Traffic graph convolutional recurrent neural network: A deep learning framework for network-scale traffic learning and forecasting," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 11, pp. 4883–4894, Nov. 2020.
[143] M. Xu, W. Dai, C. Liu, X. Gao, W. Lin, G.-J. Qi, and H. Xiong, "Spatial–temporal transformer networks for traffic flow forecasting," 2020, *arXiv:2001.02908*.
[144] S. Zhang, J. Zhang, L. Yang, J. Yin, and Z. Gao, "Spatial–temporal attention fusion network for short-term passenger flow prediction on holidays in urban rail transit systems," 2022, *arXiv:2203.00007*.
[145] Z. Wu, S. Pan, G. Long, J. Jiang, and C. Zhang, "Graph WaveNet for deep spatial–temporal graph modeling," 2019, *arXiv:1906.00121*.
[146] B. Yu, H. Yin, and Z. Zhu, "Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting," 2017, *arXiv:1709.04875*.
[147] J. Tang, T. Qian, S. Liu, S. Du, J. Hu, and T. Li, "Spatio-temporal latent graph structure learning for traffic forecasting," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2022, pp. 1–8.

[148] Y. Lu and C. Li, "AGSTN: Learning attention-adjusted graph spatio-temporal networks for short-term urban sensor value forecasting," in *Proc. IEEE Int. Conf. Data Mining (ICDM)*, Nov. 2020, pp. 1148–1153.
[149] P. Desai, C. Sujatha, S. Chakraborty, S. Ansuman, S. Bhandari, and S. Kardiguddi, "Next frame prediction using ConvLSTM," *J. Phys., Conf. Ser.*, vol. 2161, no. 1, Jan. 2022, Art. no. 012024.
[150] A. Curtis and V. A. Gonzalez, "Short term spatiotemporal video prediction on sports via convolutional LSTMS," 2016.
[151] P. Arora, H. Kumar, and B. K. Panigrahi, "Prediction and analysis of COVID-19 positive cases using deep learning models: A descriptive case study of India," *Chaos, Solitons Fractals*, vol. 139, Oct. 2020, Art. no. 110017.
[152] H. Huang, Z. Zeng, D. Yao, X. Pei, and Y. Zhang, "Spatial–temporal ConvLSTM for vehicle driving intention prediction," *Tsinghua Sci. Technol.*, vol. 27, no. 3, pp. 599–609, Jun. 2022.
[153] M. Ma and Z. Mao, "Deep-convolution-based LSTM network for remaining useful life prediction," *IEEE Trans. Ind. Informat.*, vol. 17, no. 3, pp. 1658–1667, Mar. 2021.
[154] D. Wang, Y. Yang, and S. Ning, "DeepSTCL: A deep spatio-temporal ConvLSTM for travel demand prediction," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2018, pp. 1–8.
[155] C. Zhao, P. Zhang, J. Zhu, C. Wu, H. Wang, and K. Xu, "Predicting tongue motion in unlabeled ultrasound videos using convolutional LSTM neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2019, pp. 5926–5930.
[156] C. Tan, X. Feng, J. Long, and L. Geng, "FORECAST-CLSTM: A new convolutional LSTM network for cloudage nowcasting," in *Proc. IEEE Vis. Commun. Image Process. (VCIP)*, Dec. 2018, pp. 1–4.
[157] T. Yasuno, A. Ishii, and M. Amakata, "Rain-code fusion: Code-to-code ConvLSTM forecasting spatiotemporal precipitation," in *Proc. Int. Conf. Pattern Recognit.* Cham, Switzerland: Springer, 2021, pp. 20–34.
[158] D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization," *IEEE Trans. Evol. Comput.*, vol. 1, no. 1, pp. 67–82, Apr. 1997.
[159] X. Zhang, X. Chen, L. Yao, C. Ge, and M. Dong, "Deep neural network hyperparameter optimization with orthogonal array tuning," in *Proc. Int. Conf. Neural Inf. Process.* Cham, Switzerland: Springer, 2019, pp. 287–295.
[160] G. Taguchi and S. Chowdhury, *Robust Engineering: Learn How to Boost Quality While Reducing Costs & Time to Market*. New York, NY, USA: McGraw-Hill, 1999.
[161] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," 2014, *arXiv:1409.0473*.
[162] A. Rassil, H. Chougrad, and H. Zouaki, "Holistic graph neural networks based on a global-based attention mechanism," *Knowl.-Based Syst.*, vol. 240, Mar. 2022, Art. no. 108105.
[163] L. S. Shapley, *A Value for N-Person Games*. Princeton, NJ, USA: Princeton Univ. Press, 2016.
[164] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.
[165] J. T. Schaefer, "The critical success index as an indicator of warning skill," *Weather Forecasting*, vol. 5, no. 4, pp. 570–575, Dec. 1990.
[166] S. Makridakis, E. Spiliotis, and V. Assimakopoulos, "Statistical and machine learning forecasting methods: Concerns and ways forward," *PLoS ONE*, vol. 13, no. 3, Mar. 2018, Art. no. e0194889.

SAFAA BERKANI received the BSc. degree in fundamental mathematics and the M.Sc. degree in applied analysis and statistics engineering from Abdelmalek Essaadi University, Tetouan, in 2018 and 2020, respectively. She is currently pursuing the Ph.D. degree in data science with the TICLab Laboratory, International University of Rabat (UIR), Morocco. Additionally, she has been appointed as a Temporary Lecturer with ESIN, UIR. Her research interests include data science, deep learning, time series analysis, and forecasting. In 2021, she received the Research Grant from the TICLab Laboratory, for her Ph.D. degree.



BASSMA GUERMAH received the Engineering degree in software engineering from the National Institute of Statistics and Applied Economics (INSEA), in 2014 (Major of Promotion), and the Ph.D. degree in computer science and telecommunications from the National Institute of Posts and Telecommunications (INPT), in 2018. She is currently an Assistant Professor with the Computer Science Engineering School, International University of Rabat (UIR). She is a member of TICLab. Her research interests include machine learning/deep learning (artificial intelligence), signal processing, robotics, context-aware service-oriented computing, ontologies, and semantic web.

MEHDI ZAKROUM (Member, IEEE) received the B.S. degree in mathematics from the University of Montpellier, France, and the M.S./Engineering degree in computer science and data science from the Polytechnic School, University of Lille, France. He is currently pursuing the Ph.D. degree with the TICLab, International University of Rabat, Morocco, and LORIA, France. He is with the RESIST Team, which is a joint research team between Inria Nancy-Grand Est and the University of Lorraine, France. His current research interests include artificial intelligence in support of cybersecurity, for example, network monitoring and cyber-threat inference and predictability using machine learning techniques.

MOUNIR GHOGHO (Fellow, IEEE) received the M.Sc. and Ph.D. degrees from the National Polytechnic Institute of Toulouse, France, in 1993 and 1997, respectively. He was an EPSRC Research Fellow with the University of Strathclyde, Scotland, from September 1997 to November 2001. In December 2001, he joined the School of Electronic and Electrical Engineering, University of Leeds, England, where he was promoted as a Full Professor, in 2008. In 2010, he joined the International University of Rabat, where he is currently the Dean of the College of Doctoral Studies and the Director of ICT Research Laboratory (TICLab). He is currently the Co-Founder and the Co-Director of the CNRS-Associated International Research Laboratory DataNet, in the field of big data and artificial intelligence. He has coordinated around 20 research projects and supervised more than 30 Ph.D. students in the U.K. and Morocco. His research interests include machine learning, signal processing, and wireless communication. He is a fellow of the Asia–Pacific AI Association (AAIA). He was a recipient of the 2013 IBM Faculty Award and the 2000 U.K. Royal Academy of Engineering Research Fellowship. He served as an Associate Editor for many journals, including the *IEEE Signal Processing Magazine* and the IEEE Transactions on Signal Processing.