IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.DOI

Multivariate Time Series Imputation: A Survey on available Methods with a Focus on hybrid GANs

ANNA RICHTER*[1], JYOTIRMAYA IJARADAR*[1,2], ULF WETZKER[1], Dr. VINEETA JAIN[1,3], Dr. ANDREAS FROTZSCHER[1]

[1]Fraunhofer Institute for Integrated Circuits, Division for Development of adaptive Systems, Dresden, 01187 Germany (e-mail: anna.richter, jyotirmaya.ijaradar, ulf.wetzker, vineeta.jain, andreas.frotzscher@eas.iis.fraunhofer.de)
[2]Technical University of Dresden, Department Computer Science, Dresden, 01187 Germany (e-mail: jyotirmaya.ijaradar@mailbox.tu-dresden.de)
[3]LNM Institute of Information Technology Jaipur, India

Corresponding author: Anna Richter (e-mail: anna.richter@eas.iis.fraunhofer.de).

This work was supported by the Bavarian Ministry of Economic Affairs, Regional Development and Energy through the Center for Analytics – Data – Applications (ADA-Center) within the framework of BAYERN DIGITAL II (20-3410-2- 9-8) .

ABSTRACT

Multivariate time series (MTS) are captured in a great variety of real-world applications. However, analysing and modelling the data for classification and forecasting purposes can become very challenging if values are missing in the data set. The need for imputation methods, to fill the gaps in MTS, is well known. Thus, a great variaty of algorithms for solving this task has been proposed in the literature. However, research community is constantly working on the development of advanced algorithms, that fulfill the special requirements of multidimensional temporal data, since most of the existing imputation methods treat MTS as ordinary structured data and fail to model the temporal relationships within and between sequences of observations. The main emphasis of MTS imputation research is currently put on deep learning (DL) models, especially models making use of generative adversarial networks (GANs). In our survey, we present a general categorization of imputation algorithms and introduce groups of hybrid GAN-models used for the MTS imputation task, which we investigate and discuss in detail. A quantitative comparison of the hybrid GANs' performance regarding MTS imputation is presented based on our findings in the literature.

INDEX TERMS Deep Learning, Generative Adversarial Networks, Hybrid GANs, Imputation, Missing Values, Multivariate Time Series

I. INTRODUCTION

THE goal of time series analysis is to create a model that accurately depicts the series' structure and can be used to predict and classify future events based on past observations. Time series analysis is becoming increasingly popular in a variety of real-world applications, including environmental modeling [1], [2], traffic forecasting [3], health monitoring [4], and autonomous driving [5]. Because of recent extensive research, there has been an advancement in time series modeling, reaching from simple linear models to more powerful deep learning (DL) networks. Nonetheless, most models focus on simple time series data sets [4].

However, time series observations in the real world are usually not limited to a single independent variable. Furthermore, even if all variables are sampled at a constant rate, it is very common that some data is missing due to data transmission issues or broken sensors [6]. Because of manifold measurement strategies and data acquisition devices, missing values for one or more variables are quite common. For some data sets, the missing rate can reach up to 90 % [7]. The PT08.S1 data set [8] on Italian air quality, for example, has a missing rate of 34 percent [9], while the Physionet 2012 data set [10] on medical data has a missing rate of 80 percent [11]. Numerous approaches to handle incomplete MTS data have been developed and can be separated into two superordinate classes: deletion and imputation [12]. **Deletion** is carried out either listwise or pairwise [13], where mainly samples or features are removed that are only partially observed. It has to be considered that deletion leaves gaps in the data set, possibly resulting in erroneous parameter estimations

[14]. By contrast, **imputation** means substituting missing values in a data set with estimated values [15]. Burgess et al. explained two main advantages for imputation over deletion, also called complete case method:

Power: notably, partial missingness in a single variable can still be informative for an estimate. Therefore, a thorough investigation should contain all relevant data.

Bias: while a deletion procedure may add bias, properly stated imputation estimates are unbiased [16] [17].

Several methods have been proposed to perform imputation tasks for incomplete MTS. However, there was significant bias and loss of precision found in mathematical imputation approaches such as mean/median averages [18], last observations [19], or linear regression [20]. Statistical models like autoregressive moving average (ARIMA) [21], Gaussian Process, Bayesian Network [22] along with conventional machine learning (ML) models like support vector regression (SVR) [23] and k-nearest neighbours [24] have also been applied to the MTS imputation problem. These methods are limited in covering complex temporal dependencies between observations [25].

Based on promising results and increasing popularity, a variety of DL methods has been proposed. One kind of neural network (NN), the recurrent neural network (RNN), gained special attention due to its ability to represent temporal dependencies. GRU-Decay (GRU-D) [26] is an early attempt by Che et al. to use DL to impute missing values into MTS using the gated recurrent unit (GRU) (see section 2) [27]. A bidirectional RNN structure, Bidirectional Recurrent Imputation for Time Series (BRITS), based on Long Short Term Memory (LSTM) [28] was presented by Cao et al. instead of the GRU-D to improve training accuracy [29]. Besides bi-directional RNN (bi-RNN), which only imputes values within the data streams, Yoon et al. introduced the multi-directional RNN (m-RNN) [30], which also performs imputation across data streams.

GANs were introduced by Goodfellow et al. [31]. The idea behind GAN as well as their working principle are described in section two. The idea of GANs has been used to impute missing values in MTS like it is presented in [32]–[38]. Since GANs were not designed for sequential data, research has been directed towards the development of hybrid models that use a GANs as the underlying, global concept. For example, Luo et al. proposed GRU for Imputation (GRUI)-GAN, where a modified GRU-cell called GRUI has been used to design the generator and discriminator in order to model the irregularity of time lags [32]. They further improved their work in End-to-End Generative Adversarial Network (E^2GAN) [33] by using an auto-encoder (AE). Besides, Liu et al. proposed Non-autoregressive Multiresolution Sequence Imputation (NAOMI) [39], based on a bi-RNN, to consider both future and historical data for imputation tasks. As these models are just exemplary mentioned, there are more hybrid GAN imputation models to be introduced and discussed.

In this paper, we mainly survey the group of models briefly mentioned above to achieve a generalized view on MTS imputation using GANs. We organised our paper as following: In section 2, we introduce theoretical concepts and common theories related to the survey. In section 3, we systematically present our findings on MTS imputation methods, where we focus on hybrid GANs, and suggest a categorization of the methods. Additionally, we provide a quantitative performance comparison between those algorithms. In section 4, we focus on the discussion of the following aspects related to the work on GAN-based MTS imputation: *comparability*, *autoregressive vs. non-autoregressive modelling*, *temporal dynamics and irregularity*, *interdependencies in multivariate time series* as well as *general applicability and interpretability*. Section 5 offers a summary and conclusion of the presented survey.

II. PREPARATORY KNOWLEDGE

In this section, we introduce some important theoretical concepts and common theories related to the following parts of our survey.

A. MULTIVARIATE TIME SERIES

A multivariate time series is made of two or more time-dependent variables captured at equal timestamps, which may not only depend on their past values but show dependencies on the other variables, too. Given a timestamp list $T = (t_0, t_1, t_2, ..., t_{n-1})$, we can look at $X = (x_{t_0}, x_{t_1}, x_{t_2}, ..., x_{t_{n-1}})^T$ as a series of n observations at the given timestamps. in the multivariate case, the i^{th} observation X_{t_i} of a time series X consists of d attributes $(x^0_{t_i}, x^1_{t_i}, x^2_{t_i}, ..., x^d_{t_i})$, where d is the number of variables.

B. MISSING VALUES

Donald B. Rubin developed the missing data theory in 1976, which is divided into three basic processes for missing data, each characterized by the observed and missing data [15]. For example, in terms of time series, let X be a time series that degenerates into X_0 as observed data and X_m as missing data. So the missing value matrix Y defined as –

$$Y^j_{t_i} = \begin{cases} 0, & \text{if X is missing} \\ 1, & \text{if X is not missing} \end{cases}$$

Three types of missingness are commonly defined –

1) **Missing Completely at Random (MCAR):** MCAR describes an absence of data that is independent of other observed variables as well as the missing variable itself. Let z be a set of values indicating the relationship between the absence of data in Y and X, then the probability of a certain value of Y is denoted as $p(Y|z)$.
2) **Missing at Random (MAR):** MAR describes that there is a relationship of missingness and some observed data, but not with the missing variable itself. The probability of Y is denoted as $p(Y|X_0, z)$.
3) **Missing Not at Random (MNAR):** The missingness is called MNAR if it is dependent on the unobserved (missing) values itself. The probability of Y in MNAR

is written as $p(Y|X_0, X_m, z)$. MNAR is the most critical type of missingness, as you can not observe the data it correlates with.

The categorisation of missingness into one of the three types is not always definite. Further, it was not explicitely developed for time series data. In [15] especially, a socio-economic study with one follow up measurement has been investigated as an example.

C. GENERATIVE ADVERSARIAL NETWORKS

A GAN (see figure 1), as introduced by Goodfellow et al. in 2014 [31], is a combination of two deep learning models, designed to produce high-quality artificial data from any probability distribution, that are indistinguishable from real data. The two parts of a GAN are briefly explained here:

a) **Generator:** The generator G gets a random input vector z, which is sampled from a distribution (like normal or uniform), to create artificial data. z is treated as a latent representation that has to be decoded towards meaningful data, like a certain kind of image or a time series. The size of z can be seen as a hyperparameter. To learn how to transform z into real-looking data, the generator needs the guidance of the discriminator. G can be trained via backpropagation. For this, the generator loss is used, which penalizes G for not being able to output real-looking data.

b) **Discriminator:** The discriminator D has the task to investigate the output from G to figure out if it is real or artificially created. D works as a classifier. While the the generator loss is fed to G, the discriminator loss is used to update the weights of D itself. D is penalized by its loss function for not being able to distinguish between real and artificial samples and can also be trained by backpropagation.

G and D perform a two-player-game – G tries to maximize the classification error between real and generated data while D tries to minimze it.

The minimax-optimization problem that arises out of this game can be formulated as:

$$\min_G \max_D \mathbb{E}_{x\sim p_{\text{data}}(x)}[\log D(x)] + \mathbb{E}_{z\sim p_z(z)}[1 - \log D(G(z))]$$

where $D(x)$ is the output value of the Discriminator, indicating the likelihood that the data is real. To fool the discriminator, the generator tries to minimize the right side of the expression, so that $D(x)$ gets as large as possible. For a detailed mathematical explanation, we refer to [40]

Next to the original version of GAN, which is build of two multi-layer perceptrons (MLPs) and comes with two loss functions, other versions of this generative algorithm have been developed, showing deviations from the original concept regarding constraints on the discriminators output and the corresponding loss [41] as well as regarding the architecture of generator and discriminator. Two versions are mentioned here, as they occur in some of the imputation models explained later on.

a) **Wasserstein-GAN (WGAN):** WGANs differ from original GANs in the way they measure the distance between the model distribution and the real distribution. Instead of Jensen-Shannon divergence, WGAN uses the Wasserstein divergence. This approach tends to offer a higher training stability and reduce mode collapse. For more details, we refer to [42].

b) **Deep Convolutional GAN (DCGAN):** Convolutional GANs replace the MLP with a CNN architecture in both the generator and discriminator. In DCGAN, the authors applied some modification to the CNN architectures that were previously used to scale up GANs, but without success. They used strided convolutions instead of maxpooling, removed full connections in deep hidden layers and applied batch normalization [43].

D. RECURRENT NEURAL NETWORK (RNN) CELLS

The artificial neural network (ANN) based MTS imputation models that are further described in the survey mostly rely on RNN architectures. Besides the so called vanilla RNN cell, there are two widely used cell-types that turn a standard RNN into a gated RNN. This modification addresses RNNs' difficulties in learning long-term dependencies due to vanishing gradients [44].

In a **Long Short-Term Memory (LSTM)** cell, we can find three different gates:

The **forget gate** f decides, which piece of information is removed from the cell-memory by using a sigmoid function. The cell memory is formed by recurrent hidden states at a time step t, denoted as h_t. The gate outputs a value f_t between 0 and 1, which is calculated by:

$$f_t = \sigma\left(W_{f_h}\left[h_{t-1}\right], W_{f_x}\left[x_t\right], b_f\right)$$

where W_{f_h} and W_{f_x} are weight matrices, x_t is the value of the input sequence at time step t and b_f is a constant bias.

The **input gate** i decides if new information is added to the memory or not, using a *sigmoid layer* and a *tanh layer*. Thus, two outputs have to be computed:

$$i_t = \sigma\left(W_{i_h}\left[h_{t-1}\right], W_{i_x}\left[x_t\right], b_i\right)$$

$$\tilde{c}_t = \tanh\left(W_{c_h}\left[h_{t-1}\right], W_{c_x}\left[x_t\right], b_c\right)$$

where i_t tells if a value has to be updated or not, while c_t and $\tilde{c}_t$represent vectors for the cell state and for the candidate values that would be added to the memory. Combining the two layers leads to the following formulation of the input gate's output:

$$c_t = f_t * c_{t-1} + i_t * \tilde{c}_t$$

where f_t denotes the result of the forget gate, which is multiplied with the memory's old value.

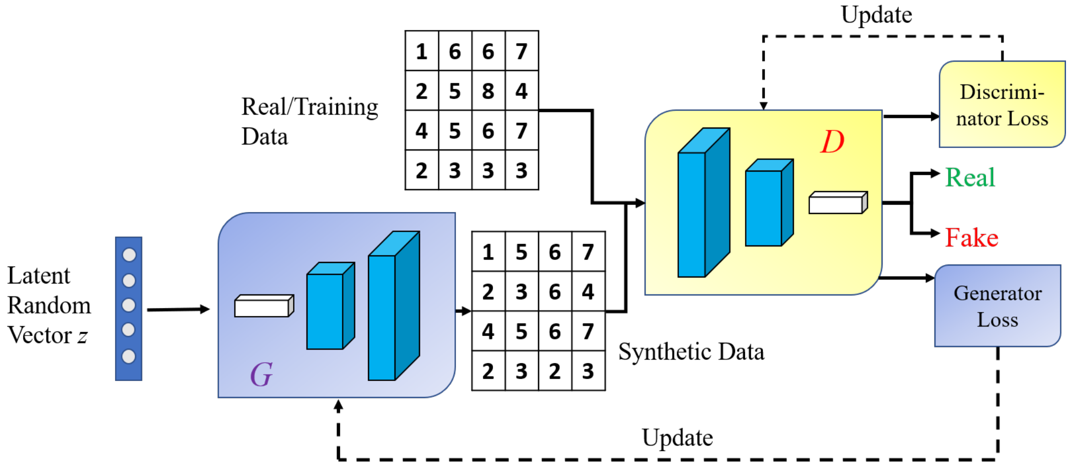

Figure 1: Generative Adversarial Network, proposed by Goodfellow et al. [31]

The **output gate** o computes the output of the LSTM cell by, again, using a *sigmoid layer* and a *tanh layer*. The results are multiplied and we get the output o_t as well as h_t, the mapping of o_t to a value between -1 and 1:

$$o_t = \sigma \left(W_{oh} \left[h_{t-1} \right], W_{o_x} \left[x_t \right], b_o \right)$$

$$h_t = o_t * \tanh \left(c_t \right)$$

The **Gated Recurrent Unit (GRU)** can be seen as a reduced version of the LSTM unit, having only two gates. Different to LSTM, it has no separate memory cells and always exposes the full memory content to other units. While LSTM controls the new content to be added to memory independently from the forget gate, GRU links the information flow control from the previous activation to the computation of the candidate activation via the update gate [45].

The **update gate** z decides to which degree the activation of a unit is updated by past information. Trough the use of a sigmoid function, the values of z lie between 0 and 1.

$$z_t = \sigma \left(W_z x_t + U_z h_{t-1} + b_z \right)$$

Through the **reset gate** r it is possible to ignore past information that are not relevant for the next time steps, by re-evaluating the combined performance of the former and new inputs [46].

$$r_t = \sigma \left(W_r x_t + U_r h_{t-1} + b_r \right)$$

The Output of the GRU at time t is formulated as:

$$h_t = z * h_{t-1} + (1 - z) * \tilde{h}$$

where $\tilde{h}$ is the intermediate memory:

$$\tilde{h} = \tanh \left(W_h x_t + r * U_h h_{t-1} + b_z \right)$$

III. CLASSIFICATION OF MTS IMPUTATION METHODS

In this section, we give an overview of methods for handling missing values in an MTS data set, focussing on the imputation task. Therefore, we introduce our categorization of the different imputation algorithms.

A. STATISTICAL METHODS

We classify statistical imputation methods into two categories based on their working principles.

Model-free imputation methods are simple, non-parametric and mainly designed for single value imputations as shown in [47]. The most common techniques are shortly explained here.

Mean or median imputation replaces a missing value with the arithmetic mean or median of all available values of a variable, e.g. a special measurement size, while *last observation carried forward (LOCF)* only uses the previous non-missing value of a variable and duplicates it. [47]. *Linear interpolation* assumes a constant change rate between two adjacent available samples, so that in theory any missing point between them can be reconstructed. *Spline interpolation* works with a piecewise polynomial fit instead, in order to smoothen the imputed sequence [48]. Since those strategies assume the absence of any significant change of statistical parameters in the data, either over the whole period of observation, since the last observation or between two observations, they can only work for stationary time series with low variance. Furthermore, they will fail imputing a bigger number of missing values between two observations, as all values within a gap would be the same. What is also not considered by most model-free imputation methods, are possible correlations between multiple variables, meaning that multivariate characteristics of a time series cannot be captured and used for imputation. The same is true for temporal relations within one variable, namely autocorrelation, which cannot be adressed. *Weighted k-nearest-neighbour (k-NN)* can be applied to multivariate time series. If enough values of a vector at one time step are observed, the nearest neighbours of this vector can be found by calculating the Euclidian distances. The missing entry in a vector is estimated out of its selected neighbours' corresponding entries [49]. As it behaves with the other methods, k-NN cannot address autocorrelation, since it does not consider the time domain at all. However, model-free methods in general offer the advantage of low computational complexity and do not require any assumptions about the underlying distribution of the data.

Model-based approaches cover parametric methods, especially machine learning algorithms applied to the MTS imputation task. Either the time series itself or the distribution of the data are modeled. We explain a selection of this group of methods here.

Autoregressive time series models predict missing values based on historical data. Thus, they can work on significantly autocorrelated time series. In case of multivariate time series, vector-autoregression can be applied. The ARIMA (autoregressive integrated moving average) model is well-known for time series forecasting. An imputation problem can be interpreted as a forecasting problem, if missing values are calculated one by one based on the previous values of the time series. In case of multiple missing values, the previously imputed value is considered as known and is used for the next imputation step [50]. *Expectation maximization (EM)* can be seen as general algorithm for modeling incomplete data [51]. As an example, it can be applied within a method based on probabilistic principal component analysis (PPCA), as [49] described. PPCA assumes a dependency of the observed data on latent variables. EM is used to find the set of data, which fits the presumed distribution of the latents best. *Matrix factorization (MF)* as a more advanced statistical model makes use of potential redundancies, regularities or correlations within a matrix to fill in the gaps. The multidimensional time series is modeled in the form of a matrix. In order to capture temporal dependencies, the standard MF algorithm can be extended by graph-based regularization [52]. In general, statistical imputation models can capture global characteristics of a data set better than model-free statistical methods. Nonetheless, these methods have not been developed for time series data and cannot consider characteristics like temporal dependencies, time decay and feature correlations in MTS properly [38].

B. DEEP-LEARNING METHODS

To circumvent the difficulties of the formerly described approaches in the modeling and completion of multivariate temporal data, DL methods can be applied to the task of MTS imputation. DL methods can be classified into two major categories based on their way of modeling distributions [53].

- **Discriminative models:** modeling the joint distribution of an observable variable and a target variable
- **Generative models:** modeling the conditional distribution of a target variable, given an observable variable

RNNs (based on LSTM or GRU) as an example of discriminative models are capable of capturing long-term temporal dependencies of MTS without previous assumptions on the data. They can learn complex temporal dynamics and are therefore also suited for irregular time series. variational auto-encoders (VAEs) and GANs fall into the class of generative models. A VAE is a probabilistic model, which uses an auto-encoder-like neural network to perform probabilistic inference on the posterior probability distribution [54]. In contrast, a GAN uses adversarial training via a combination of a generator and a discriminator network, as explained in detail in the previous section.

In the following part of the survey, we present eight DL imputation models, that globally incorporate a **GAN-structure**. Since original GANs were not designed for sequential data analysis and cannot handle the time series imputation task natively, the presented models are **hybrid GANs** which adopt different architectures in the design of generators and discriminators to process MTS data.

Based on their architecture, we further grouped the **hybrid GAN imputation models** into four categories:

- Deep convolutional GAN models
- Gated recurrent GAN models
- Bidirectional GAN models
- VAE-GAN models

1) Deep Convolutional GAN models

At first, the usage of DCGANs on MTS data was limited to anomaly detection, like it is presented in [55]–[58]. The CNN utilized in the DCGAN architecture [43] is specialized in extracting features from images rather than MTS. Therefore, the development of multi-channel CNN (MC-CNN) [59] enabled capturing information from each dimension of an MTS and, using a fully connected MLP, also learning the coupling relationship among the variables. MTS-GAN proposed by [60] is the first published work on MTS missing value imputation with a convolutional GAN. It is designed by replacing the CNN architecture of DCGAN with the multi-channel architecture of MC-CNN.

In figure 2, architecture and functionality of the MTS-GAN are displayed: The generator gets a latent random vector sampled from a uniform distribution and performs a deconvolution separately within every channel to model each dimension of the MTS. The discriminator extracts the features of each single time series in a separate channel using multi-layer 1D-kernel-based convolution, whereas the relationships between the channels are modeled by an MLP, which outputs a scalar.

The MTS imputation is considered a constrained MTS generation task by the authors. The best representation in latent space has to be found for an incomplete time series, after the MTS-GAN has been trained with the subset of complete time series data. This process can be considered as an optimization problem, which can be solved by backpropagation. Out of the found representation, the incomplete MTS can be reconstructed to its complete form. For imputation itself,

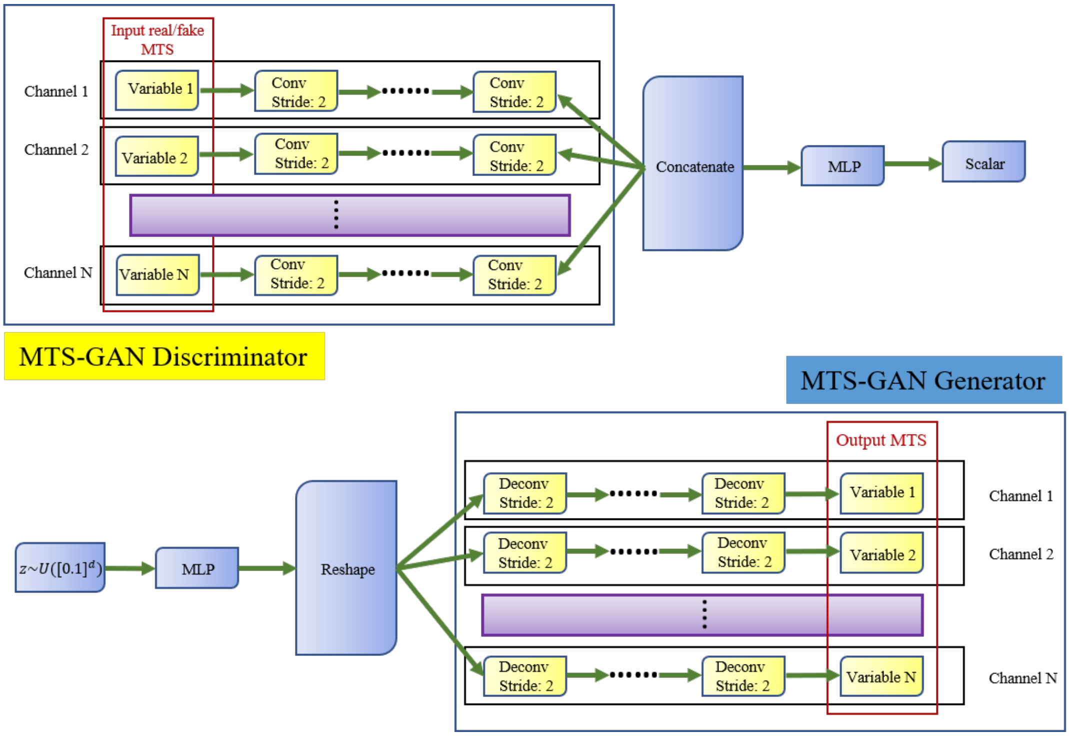

Figure 2: Multivariate Time Series - Generative Adversarial Network (MTS-GAN) proposed by Guo et al. [60]

only the generator loss is considered as the loss function, what reduces the number of hyper-parameters which have to be backpropagated. To find the closest latent encoding of incomplete time series, backpropagation has to be performed multiple times with different initial values, so that the most probable value can be selected.

Huang et al. proposed the Traffic Sensor Imputation Generative Adversarial Network (TSDIGAN) [35] to impute missing traffic sensor data. To deal with temporal dependencies in time series data, they proposed a Gramian Angular Summation Field (GASF), converting the MTS imputation task to an image imputation task. The innovation in their work is the utilization of the GASF to learn the pointwise temporal relation between the time series data while the DCGAN architecture remains unchanged. The working principle of the imputation task is illustrated in figure 3. At first, the time series data is converted into a GASF matrix image, maintaining temporal correlations. This allows the training of a deep convolution-based GAN capable of generating realistic synthetic data. Searching the replacement for missing values is similar to the imputation process in MTS-GAN.

2) Gated Recurrent GAN Models

Che et al. [26] have used the GRU for MTS with missing values. The novelty of their model GRU-D consists in the detection of the time intervals between the observed values as a representation of the missingness pattern. This allows to model "informative missingness", meaning that values are not missing at random. They introduced two trainable decay rates, for input and hidden states, to deal with a vanishing influence of input values, which occurs when a variable has been missing for a longer time. Luo et al. [32] followed the concept of decay rates by introducing GRUI-GAN, which combines a modified GRU-network with a GAN structure for missing value imputation. The generator tries to map a random noise input to a realistic complete MTS, while the discriminator produces a mapping to the probability of the input data to be real (see figure 4). Through the utilization of a WGAN structure instead of original GAN, an increase in stability as well as in the ease of optimization should be achieved.

E²GAN is the follow-up work from Luo et al. [33], where the generator is formed by a denoising auto-encoder (DNAE) based on the GRUI-cell [32]. Not random noise like in GRUI-GAN but incomplete MTS are fed to the generator. But instead of dropping the missing values of the incomplete time series, authors added a random noise to the original samples. In addition to the samples, their time lag matrix representing the temporal gaps between the observed values is fed to the generator. The same MTS is given to the discriminator. The GRUI-based decoding discriminator design is similar to the GRUI-GAN discriminator. Figure 5 illustrates the architecture of E²GAN, where the DNAE compresses and reconstructs the incomplete MTS, before the discriminator tries to distinguish real and reconstructed data. E²GAN, using a modified WGAN loss-function, can be seen as the evolution of the authors' previously introduced GRUI-GAN due to better performance and training time.

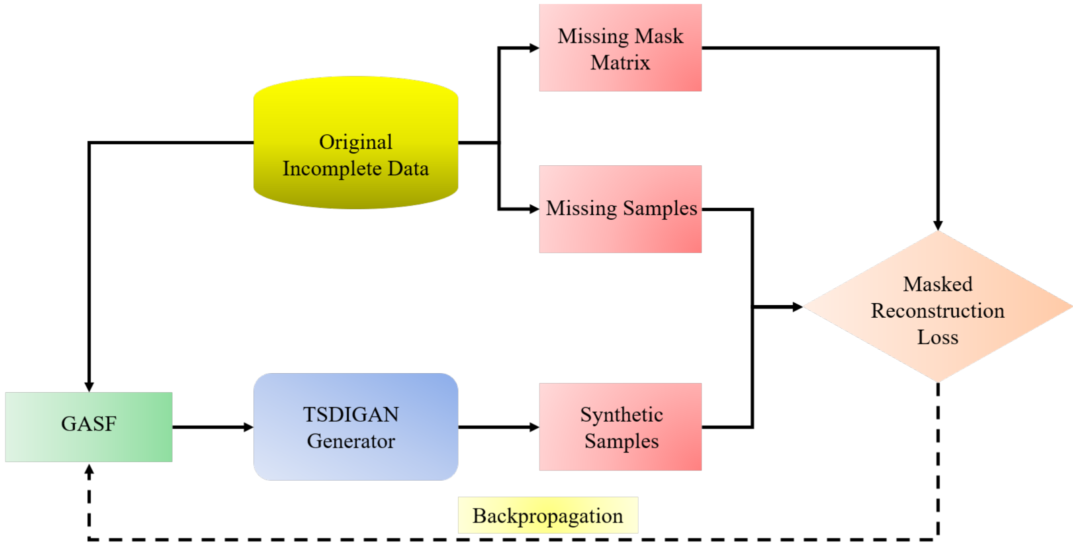

Figure 3: Traffic Sensor Imputation Generative Adversarial Network (TSDIGAN) proposed by Huang et al. [35]

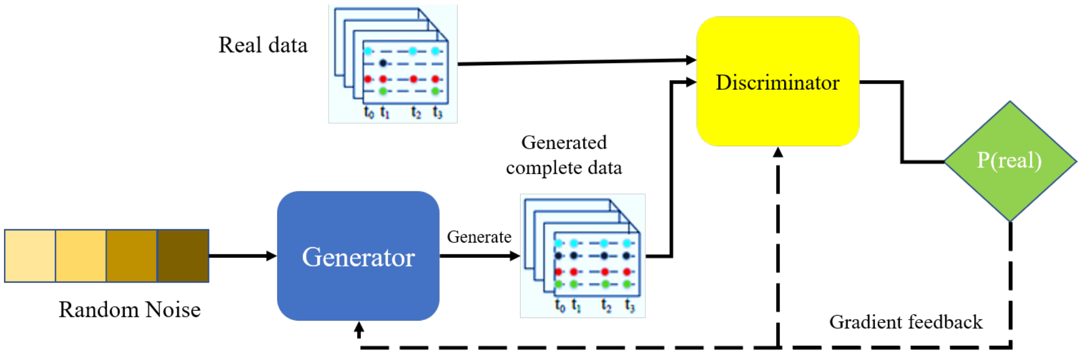

Figure 4: GRU for Imputation - Generative Adversarial Network (GRUI-GAN) proposed by Luo et al. [32]

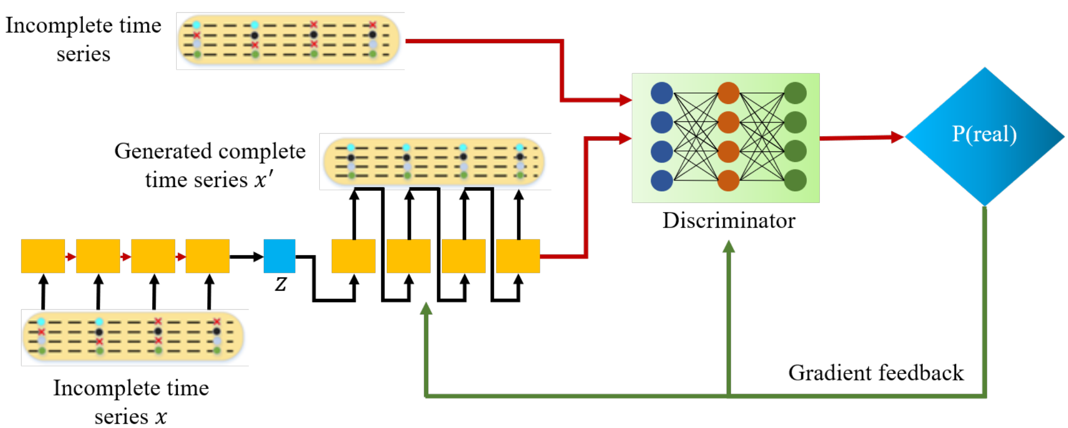

Figure 5: End-to-End Generative Adversarial Network (E²GAN) proposed by Luo et al. [33]

3) Bidirectional GAN models

BRITS [29], a state of the art LSTM-based model for imputation, delivered basic ideas for the development of bidirectional GAN models. BRITS performs imputation using bidirectional dynamics and a temporal decay factor. It consists of two unidirectional LSTM-architectures, one trained in forward and one trained in backward direction. NAOMI, standing for non-autoregressive multiresolution imputation, is a model proposed by Liu et al. [39]. It is also designed bidirectionally, more precisely with a forward-backward encoder and a multiresolution decoder. In case of modeling stochastic dynamics, NAOMI is extended to a GAN-model, as a discriminator is applied instead of a simple loss function. In figure 6, f_f and f_b represent the forward and backward recursive networks. The hidden state h_t is concatenated out of the forward and backward hidden state and is updated one or two times. For prediction of the hidden states h_t, the 'divide and conquer' procedure is used, as depicted in figure 6: for example, h_3 is predicted out of h_1 and h_5, further h_2 and h_4 are predicted from h_3 and h_1 or h_5, respectively. A GAN structure is set up to improve the prediction performance of the bidirectional LSTM-based model.

Miao et al. [61] proposed the semi-supervised generative adversarial network model (SSGAN). Similar to NAOMI and BRITS, it adopts a bidirectional structure to capture the temporal information. The authors added a classifier to address the missing label issue in partially labeled data sets. The components of SSGAN are the mentioned classifier, a generator and a discriminator, all made up from BiRNN cells [61] mainly. The architecture is illustrated in figure 7. SSGAN takes up incomplete MTS, a data label matrix, a mask matrix and a time-lag matrix. The generator predicts the missing components in the given data based on observable components and labels and outputs a completed MTS matrix. The classifier, trained with labeled time series, is used to predict labels for unlabeled samples and sends feedback in form of the cross entropy L_{CE} to the generator. This enables it to focus on samples with the same label while imputing an incomplete time series. The discriminator then outputs a discriminative matrix, which shows the probability of being original or reconstructed for every component of the matrix and is fed back to the generator.

Oh et al. [38] proposed the STING model for MTS imputation, making use of WGAN as well as the concept of attention. It includes two generators, one working in forward and one in backward direction. Since the generators and discriminator both adopt GRU cells, one might also put STING into our category of gated recurrent GAN models. Next to the data matrix, a random matrix, a mask matrix and a time-gap matrix are fed into the generator, which itself includes two types of attention layers, that aim to find the most important parts of the input data. The self-attention module, producing so called context vectors out of the input sequences, calculates attention scores (see [38] for the mathematical procedure) at different positions and captures quantitative correlations within the whole sequence. The temporal attention module represents the correlations between context vectors and hidden states of the network. Motivated by the work of Luo et al. [32], Oh et al. [38] additionally included a temporal decay layer into the generators. The GRU-discriminator has a simple design compared to the generator and outputs a probability of being original or reconstructed for all components of the matrix, like in SSGAN [61]. Next to the completed data matrix, a hint matrix, which enforces attention on special components, is provided to the discriminator.

4) VAE-GAN models

The researchers in [62]–[64] propose the use of non-linear dimensionality reduction to handle missing values in time series using variational autoencoders (VAE). GP-VAE proposed by [36] is especially focused on MTS. Their work includes a VAE to map incomplete time series onto a low dimensional latent space and models temporal dynamics via a Gaussian process (GP). A standard VAE is not capable of an exact inference on latent variables corresponding to a data point, as it only optimizes the lower bound of the data's log-likelihood. Considering this fact, Kingma et al. [65] introduced 'Glow', a generative flow model with invertible convolutions, for log-likelihood evaluation. Liu et al. [37] took advantage of Glow by including it to their GlowImp model 9 next to a VAE and combining it with a WGAN. The Glow components enable exact latent variable inference and log-likelihood evaluation via a sequence of invertible transformations (also called a 'normalizing flow'). A decoder-generator outputs a completed MTS from the Glow-VAE's latent representation and feeds it to the discriminator. Glow-VAE and WGAN are trained jointly using gradient descent.

C. COMPARATIVE PERFORMANCE EVALUATION

To assess and compare the quality and performance of different machine learning models for MTS imputation, the type of data set being modeled as well as the applied evaluation methods found in the corresponding publications have to be taken into account. In most cases, publicly available data sets are used to test a new model. MTS data are created in a variety of domains, but for the imputation task mainly the follwing data sets have served as benchmarks so far:

a) **PhysioNet:** The PhysioNet [10] data set is a publicly accessible electronic medical record data set that was generated as part of the 2012 PhysioNet Challenge. This dataset contains statistics on 12,000 intensive care unit (ICU) admissions. Each ICU stay is a time series of around 48 hours with 37 variables like heart rate, cholesterol and glucose level.

b) **KDD:** The KDD CUP 2018 dataset [66] is a publicly available dataset on air quality that was used in the 2018 KDD CUP Challenge. The KDD dataset contains historical values on Beijing's and London's air quality level between 2017 and 2018 which is represented in multiple metrics. It contains over 1400 samples with

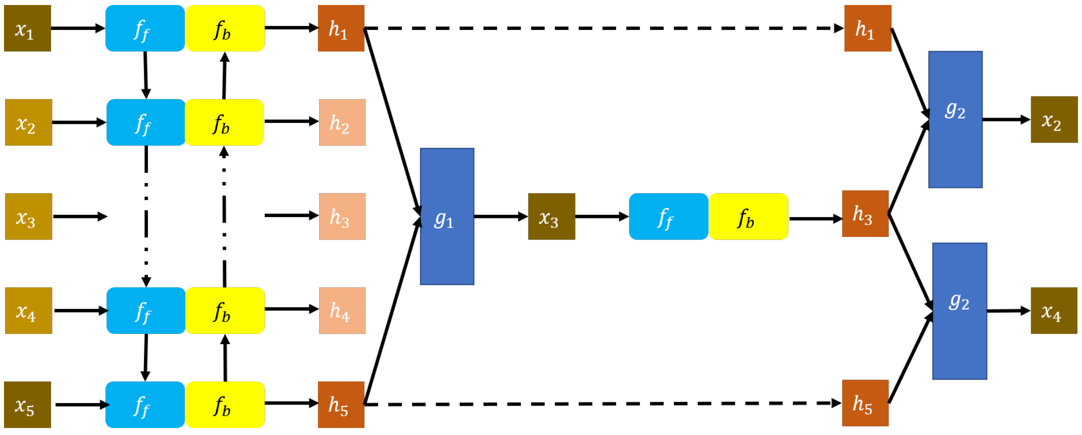

Figure 6: Divide-and-conquer strategy in Non-autoregressive Multiresolution Sequence Imputation (NAOMI), proposed by Liu et al. [39]

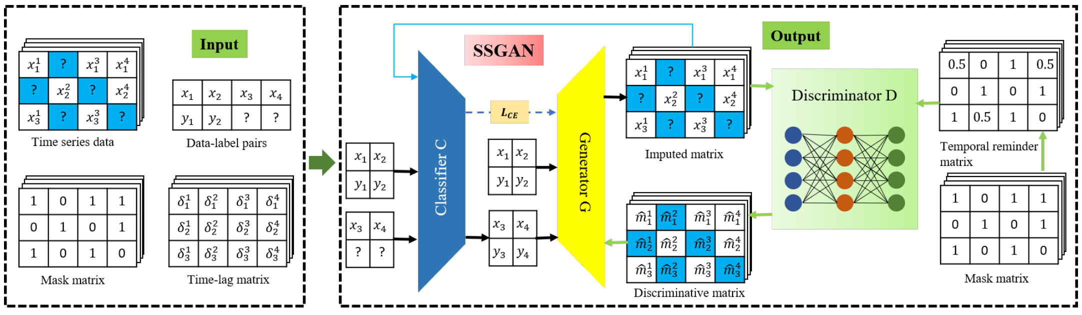

Figure 7: Semi-Supervised Generative Adversarial Network (SSGAN), proposed by Miao et al. [61]

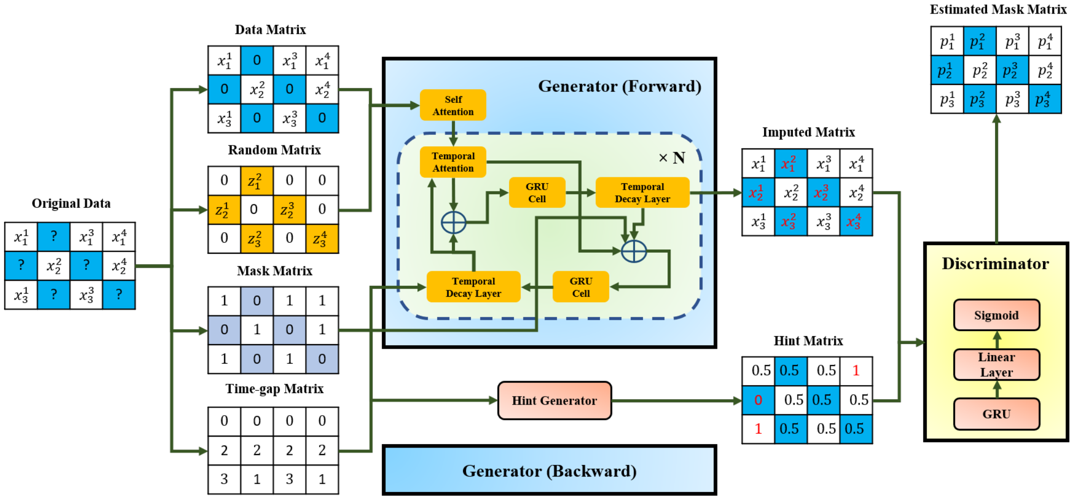

Figure 8: Self-attention-based Time Series Imputation Networks using GAN (STING), proposed by Oh et al. [38]

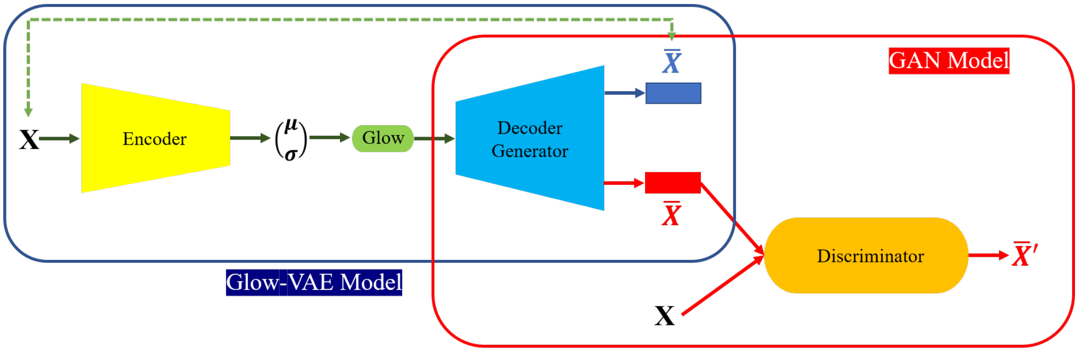

Figure 9: GLOW and GAN fur Multivariate Time Series Imputation (GlowImp), proposed by Liu et al. [37]

hourly captured values and the length of one sequence is 24.

To evaluate the imputation performance in hybrid GAN-models, common regression based evaluation metrics, such as Mean Absolute Error (MAE), Mean Squared Error (MSE), Root Mean Squared Error (RMSE) and Mean Relative Error (MRE), are used. An overview over all mentioned evaluation metrics can be found in [67]. When dealing with a real data set, a direct evaluation of missing value imputation accuracy is impossible as the ground truth is missing. Therefore, one would randomly discard a percentage of true observations and calculate the imputation accuracy on this subset of values, representing the ground truth.

As most of the authors of hybrid GAN imputation models have compared their results to those of the BRITS [29] model, BRITS could be considered as a benchmark standard for the imputation performance of deep neural networks. For this reason, we used BRITS as a baseline and compared the imputation performance of the hybrid GANs against it, as far as we could find the corresponding evaluation metrics in the literature. In table 1, we present our findings for the imputation of PhysioNet and KDD with different percentages of missing values. Except for TSDIGAN, all models have been compared to BRITS at least for the KDD data set. For every calculated percentage, where a positive sign means improvement and a negative sign means deterioration, we have referenced the publication which the underlying RMSE values are taken from. Not in all cases could the comparison to BRITS be found in the original publications on the models. According to the results we found, there are two hybrid GANs which outperformed BRITS on both data sets: SSGAN and STING. However, the results are clearer in the case of KDD. GlowImp showed the best results regarding the KDD data set and GRUI-GAN achieved slightly better results on KDD, too. For NAOMI, we found slightly worse results on PhysioNet, and mixed results on KDD. On average, NAOMI's performance seems to be quite similar to the performance of BRITS. MTS-GAN performed worse than BRITS on both data sets. However, the most interesting results are associated with E²GAN, as we found quite contradictory values for the KDD data set here. From the first source we found E²GAN performing around 20 % better than BRITS, but according to the second source it performs about 20 % worse. Regarding TSDIGAN, we can at least state that it clearly outperformed the baseline models, like ARIMA, k-NN and PPCA, which it was compared to in [35]. For TSDIGAN, none of the introduced data sets was used.

IV. DISCUSSION

In this section, we would like to focus on the discussion of various aspects related to the work on GAN-based MTS imputation presented in the previous sections of this paper. As part of this debate, we refer to the differences that the models show regarding those aspects.

A. COMPARABILITY

In the former section, we performed a quantitative comparison of the hybrid GANs regarding their MTS imputation capability, taking BRITS as a baseline. Regardless of the use of the same standard data sets and evaluation metric (RMSE), a comparison of RMSE values could only be made within the individual publications, not between them. This is for two reasons.

First, using the same data set does not guarantee consistent experimental data, as various subsets of data or versions of the data set and different experimental settings may have been used. Additionally, manually created loss patterns for the purpose of ground truth availability are likely to be different.

Second, the authors applied different kinds of normalization to their data, which leads to differences in the RMSE values. For full comparability of imputation algorithms, the benchmark datasets would have to be expanded by a standardized ground truth for missing values, covering different missingness rates. Furthermore, a consistent normalization would have to be applied to the data.

B. AUTOREGRESSIVE VS. NON-AUTOREGRESSIVE MODELING

Looking at the different architectures, we can distinguish autoregressive and non-autoregressive modeling. A model is inherently autoregressive when it needs the previous latent

		Imputation performance (RMSE) of hybrid GANs compared to BRITS (%)						
Data set	Missing (%)	MTS-GAN	GRUI-GAN	E2GAN	NAOMI	SSGAN	STING	GlowImp
PhysioNet	10	-13.1 [61]	–	-11.1 [61]	-3.4 [61]	+2.1 [61]	–	–
	20	–	–	-15.4 [38]	–	–	+4.0 [38]	–
	30	-16.4 [61]	–	-10.0 [61]	-4.6 [61]	+0.3 [61]	–	–
	50	-4.2 [61]	–	-1.3 [61]	-0.5 [61]	+2.3 [61]	–	–
	70	-2.5 [61]	–	-1.7 [61]	-3.2 [61]	+3.3 [61]	–	–
	90	+0.4 [61]	–	-0.4 [61]	-1.8 [61]	+3.8 [61]	–	–
KDD	10	-28.4 [61]	+8.7 [37]	+21.9 [37], -23.2 [61]	+1.7 [61]	+18.1 [61]	+21.0 [38]	+30.6 [37]
	20	–	+7.9 [37]	-25.9 [38], +20.0 [37]	–	–	+21.0 [38]	+31.0 [37]
	30	-34.0 [61]	+9.0 [37]	+23.5 [37], -20.3 [61]	+0.5 [61]	+17.8 [61]	+21.0 [38]	+30.9 [37]
	50	-34.8 [61]	+1.7 [37]	+18.4 [37], -24.8 [61]	-3.6 [61]	+15.7 [61]	+22.0 [38]	+28.6 [37]
	70	-24.2 [61]	+0.0 [37]	+16.6 [37], -19.8 [61]	-9.4 [61]	+5.9 [61]	+23.0 [38]	+27.6 [37]
	90	-15.1 [61]	–	-9.7 [61]	-5.8 [61]	+8.3 [61]	+25.0 [38]	–

Table 1: Comparison of models' performance

representation h_{t-1} to learn and infer the current representation h_t of a time series [68]. In general, autoregressive networks require longer training times due to longer backpropagation paths. In addition, autoregressive training is not parallelizable. This kind of networks still represents the state-of-art regarding time series modeling and prediction. However, researchers tend to adopt non-autoregressive models for tackling time series analysis more and more. Weber et al. [69] directly compared autoregressive and non-autoregressive implementations of a GRU as well as of a TCN (temporal convolutional network). They observed that the significantly faster non-autoregressive models achieved at least as accurate results as their autoregressive partners. Similarly, Martínez-González et al. [70] used a non-autoregressive model for motion prediction, reaching competitive results with SotA-RNN-approaches. The danger of error accumulation in multistep forecasts, e.g. in case of large gaps in an imputation scenario, also has to be considered. NAOMI [39], for instance, uses a non-autoregressive approach while applying the divide-and-conquer technique. Deep convolutional GAN-models also do not use autoregression. It is important to investigate the influence of (not) using autoregressive techniques on the quality of MTS imputation results in the future.

C. TEMPORAL DYNAMICS AND IRREGULARITY

Another interesting fact is, that the NAOMI [39] is the only one of the considered models which performs a multiresolution analysis on the given time series. This feature can be advantageous if patterns in a time series occur at different time scales, which the other models might fail to capture. Considering temporal dynamics in the latent space, Fortuin et al. [36], who delivered the foundation for the introduced VAE-GAN model, stand out with their model by using deep Gaussian Process (GP) modeling. However, all reviewed GAN-based models that convert time series into a latent representation do not explicitly consider latent temporal dynamics but only represent multidimensional frequency distributions, which might be inaccurate. Another imputation-specific concern is the irregularity of time lags due to the occurring missingness patterns. The authors of GRUI-GAN [32] and E²GAN [33] have addressed this with the introduction of decay rates in their models, which limit the influence of an input value in case of a large time lag. In the other reviewed publications, vanishing impacts are not explicitely considered.

D. INTERDEPENDENCIES IN MULTIVARIATE TIME SERIES

When it comes to modeling MTS in order to predict them or impute their values, dependencies between single time series play an important role. Mathematically, we have to look at the modeling of multidimensional probability distributions. Generative models like GANs and VAEs are characterized by modeling joint distributions, whereas they do not inherently model conditional distributions. For instance, an encoder-decoder generator can only impute the values based on a joint distribution, in which the the single time series can still be stochastically independent. It cannot make decisions based on conditional distributions, where the value of a single time series at time t is stochastically dependent on another time series at time t or $t - x$. The latter case is called a lagged dependency. Multidirectional imputation [30], which happens within and across channels, directly addresses the dependency between time series in the non-lagged case. To our knowledge, multidirectional imputation has not yet been adopted in a GAN-based MTS imputation model. Lagged dependencies between time series have also not yet been considered in this group of algorithms.

E. GENERAL APPLICABILITY AND EXPLAINABILITY

The application of new models to a very limited set of data would lead to a more general problem besides comparability, namely the uncertainty about general applicability. Although a model might perform well on a special medical or environmental data set it may not show precise results on data sets from other domains as the data might show different scale levels, dependency structures and temporal dynamics. With the mentioned benchmark data sets, domain specific comparisons of imputation models are possible. However, if the models' general capability should be compared, they would all have to be applied to a great variety of data sets collected from different domains. This could form a future research task on hybrid GAN imputation models. Of course, the described challenges are not only relevant for the group of algorithms reviewed in this paper, but for all kinds of deep learning algorithms which should perform a certain task without dependencies on a certain domain. The explainability of imputation results, i.e. the clarification about which part of the given information was used in which way by the model to predict the values, is another challenge that complex DL models like hybrid GANs face. It can be stated that attention-based models like [38] provide an intrinsic explainability, as they highlight the parts of the input data that are considered being especially important for the model predictions. Apart from that, e.g. interpreting the imputation process in VAE-based models is quite difficult because the latent representations often do not match the actual structure or distribution of the data. Generally, in most of the reviewed models, additional methods would have to be included for the purpose of explainability.

V. SUMMARY AND CONCLUSION

In our survey, we have presented our categorization of imputation models in general, referring to basic statistical (model-free), machine learning (model-based) and deep learning algorithms. After this, we have reviewed and discussed the state of the art regarding GAN-based MTS imputation methods. In particular, we classified the hybrid GAN models into four categories based on their design and functionality: deep convolutional GAN models, gated recurrent GAN models, bidirectional GAN models and VAE-GAN models. A quantitative performance comparison of hybrid GANs was carried out, based on numerical results we found in the reviewed publications. Further, we discussed different aspects related to the reviewed models, including comparability, (non-)autoregressive modeling, temporal dynamics, MTS characteristics as well as applicability and explainability. Hybrid GAN models are surely a powerful tool for the challenging task of MTS imputation, as it has been shown by comparison using example applications on two domain-specific data sets. According to our findings, most of the recently developed algorithms performed on the level of state of the art MTS imputation methods or even outperformed them, while only a few hybrid GAN models could not keep up with the considered baseline model in the reported experiments. Still, more comparative research using a bigger variety of data sets from different domains is needed to further strengthen the statement on their performance ranking.

References

[1] R. R. Stein, V. Bucci, N. C. Toussaint, C. G. Buffie, G. Rätsch, E. G. Pamer, C. Sander, and J. B. Xavier, "Ecological modeling from time-series inference: insight into dynamics and stability of intestinal microbiota," PLoS Comput. Biol., vol. 9, p. e1003388, Dec. 2013.

[2] C. Taylor, D. Pedregal, P. Young, and W. Tych, "Environmental time series analysis and forecasting with the captain toolbox," Environ. Model. Softw., vol. 22, pp. 797–814, June 2007.

[3] Y. Hua, Z. Zhao, R. Li, X. Chen, Z. Liu, and H. Zhang, "Deep learning with long short-term memory for time series prediction," IEEE Commun. Mag., vol. 57, pp. 114–119, June 2019.

[4] C. Xiao, E. Choi, and J. Sun, "Opportunities and challenges in developing deep learning models using electronic health records data: a systematic review," J. Am. Med. Inform. Assoc., vol. 25, pp. 1419–1428, Oct. 2018.

[5] C. Roesener, F. Fahrenkrog, A. Uhlig, and L. Eckstein, "A scenario-based assessment approach for automated driving by using time series classification of human-driving behaviour," in 2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC), IEEE, Nov. 2016.

[6] D. Neil, M. Pfeiffer, and S.-C. Liu, "Phased lstm: Accelerating recurrent network training for long or event-based sequences," in Advances in Neural Information Processing Systems (D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, eds.), vol. 29, Curran Associates, Inc., 2016.

[7] P. J. García-Laencina, P. H. Abreu, M. H. Abreu, and N. Afonoso, "Missing data imputation on the 5-year survival prediction of breast cancer patients with unknown discrete values," Comput. Biol. Med., vol. 59, pp. 125–133, Apr. 2015.

[8] S. De Vito, E. Massera, M. Piga, L. Martinotto, and G. Di Francia, "On field calibration of an electronic nose for benzene estimation in an urban pollution monitoring scenario," Sens. Actuators B Chem., vol. 129, pp. 750–757, Feb. 2008.

[9] N. Fouladgar and K. Främling, "A novel LSTM for multivariate time series with massive missingness," Sensors (Basel), vol. 20, p. 2832, May 2020.

[10] I. Silva, G. Moody, D. J. Scott, L. A. Celi, and R. G. Mark, "Predicting in-hospital mortality of icu patients: The physionet/computing in cardiology challenge 2012," in 2012 Computing in Cardiology, pp. 245–248, 2012.

[11] Q. Li and Y. Xu, "VS-GRU: A variable sensitive gated recurrent neural network for multivariate time series with massive missing values," Appl. Sci. (Basel), vol. 9, p. 3041, July 2019.

[12] J. R. Cheema, "A review of missing data handling methods in education research," Rev. Educ. Res., vol. 84, pp. 487–508, Dec. 2014.

[13] P. E. McKnight, K. M. McKnight, S. Sidani, and A. J. Figueredo, Missing data: A gentle introduction. Guilford Press, 2007.

[14] J. W. Graham, "Missing data analysis: making it work in the real world," Annu. Rev. Psychol., vol. 60, no. 1, pp. 549–576, 2009.

[15] D. B. Rubin, "Inference and missing data," Biometrika, vol. 63, no. 3, pp. 581–592, 1976.

[16] I. R. White and J. B. Carlin, "Bias and efficiency of multiple imputation compared with complete-case analysis for missing covariate values," Stat. Med., vol. 29, pp. 2920–2931, Dec. 2010.

[17] S. Burgess, I. R. White, M. Resche-Rigon, and A. M. Wood, "Combining multiple imputation and meta-analysis with individual participant data," Stat. Med., vol. 32, pp. 4499–4514, Nov. 2013.

[18] I. B. Aydilek and A. Arslan, "A hybrid method for imputation of missing values using optimized fuzzy c-means with support vector regression and a genetic algorithm," Inf. Sci. (Ny), vol. 233, pp. 25–35, June 2013.

[19] S. Song, C. Li, and X. Zhang, "Turn waste into wealth," in Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, (New York, NY, USA), ACM, Aug. 2015.

[20] P. T. von Hippel, "4. regression with missing ys: An improved strategy for analyzing multiply imputed data," Sociol. Methodol., vol. 37, pp. 83–117, Aug. 2007.

[21] V. Layanun, S. Suksamosorn, and J. Songsiri, "Missing-data imputation for solar irradiance forecasting in thailand," in 2017 56th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE), IEEE, Sept. 2017.

[22] S. P. Susanti and F. N. Azizah, "Imputation of missing value using dynamic bayesian network for multivariate time series data," in 2017 International Conference on Data and Software Engineering (ICoDSE), IEEE, Nov. 2017.

[23] S.-F. Wu, C.-Y. Chang, and S.-J. Lee, "Time series forecasting with missing values," in Proceedings of the 1st International Conference on Industrial Networks and Intelligent Systems, ICST, 2015.

[24] E. Acuna and C. Rodriguez, “The treatment of missing values and its effect on classifier accuracy,” in Classification, clustering, and data mining applications, pp. 639–647, Springer, 2004.
[25] S. Siami-Namini, N. Tavakoli, and A. S. Namin, “A comparative analysis of forecasting financial time series using arima, lstm, and bilstm,” arXiv preprint arXiv:1911.09512, 2019.
[26] Z. Che, S. Purushotham, K. Cho, D. Sontag, and Y. Liu, “Recurrent neural networks for multivariate time series with missing values,” Sci. Rep., vol. 8, p. 6085, Apr. 2018.
[27] K. Cho, B. Van Merriënboer, D. Bahdanau, and Y. Bengio, “On the properties of neural machine translation: Encoder-decoder approaches,” arXiv preprint arXiv:1409.1259, 2014.
[28] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” Neural computation, vol. 9, no. 8, pp. 1735–1780, 1997.
[29] W. Cao, D. Wang, J. Li, H. Zhou, L. Li, and Y. Li, “Brits: Bidirectional recurrent imputation for time series,” Advances in neural information processing systems, vol. 31, 2018.
[30] J. Yoon, W. R. Zame, and M. van der Schaar, “Estimating missing data in temporal data streams using multi-directional recurrent neural networks,” IEEE Transactions on Biomedical Engineering, vol. 66, no. 5, pp. 1477–1490, 2018.
[31] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Advances in neural information processing systems,” Curran Associates, Inc, vol. 27, pp. 2672–2680, 2014.
[32] Y. Luo, X. Cai, Y. Zhang, J. Xu, et al., “Multivariate time series imputation with generative adversarial networks,” Advances in neural information processing systems, vol. 31, 2018.
[33] Y. Luo, Y. Zhang, X. Cai, and X. Yuan, “E2gan: End-to-end generative adversarial network for multivariate time series imputation,” in Proceedings of the 28th international joint conference on artificial intelligence, pp. 3094–3100, AAAI Press, 2019.
[34] Y. Zhang, B. Zhou, X. Cai, W. Guo, X. Ding, and X. Yuan, “Missing value imputation in multivariate time series with end-to-end generative adversarial networks,” Information Sciences, vol. 551, pp. 67–82, 2021.
[35] T. Huang, P. Chakraborty, and A. Sharma, “Deep convolutional generative adversarial networks for traffic data imputation encoding time series as images,” International Journal of Transportation Science and Technology, 2021.
[36] V. Fortuin, D. Baranchuk, G. Rätsch, and S. Mandt, “Gp-vae: Deep probabilistic time series imputation,” in International conference on artificial intelligence and statistics, pp. 1651–1661, PMLR, 2020.
[37] C. Liu, H. Zhou, Z. Sun, and G. Cui, “Glowimp: combining glow and gan for multivariate time series imputation,” in International Conference on Algorithms and Architectures for Parallel Processing, pp. 50–64, Springer, 2021.
[38] E. Oh, T. Kim, Y. Ji, and S. Khyalia, “Sting: Self-attention based time-series imputation networks using gan,” in 2021 IEEE International Conference on Data Mining (ICDM), pp. 1264–1269, IEEE, 2021.
[39] Y. Liu, R. Yu, S. Zheng, E. Zhan, and Y. Yue, “Naomi: Non-autoregressive multiresolution sequence imputation,” Advances in neural information processing systems, vol. 32, 2019.
[40] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial nets,” in Advances in Neural Information Processing Systems (Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Weinberger, eds.), vol. 27, Curran Associates, Inc., 2014.
[41] M. Lucic, K. Kurach, M. Michalski, S. Gelly, and O. Bousquet, “Are gans created equal? a large-scale study,” Advances in neural information processing systems, vol. 31, 2018.
[42] M. Arjovsky, S. Chintala, and L. Bottou, “Wasserstein generative adversarial networks,” in International conference on machine learning, pp. 214–223, PMLR, 2017.
[43] A. Radford, L. Metz, and S. Chintala, “Unsupervised representation learning with deep convolutional generative adversarial networks,” arXiv preprint arXiv:1511.06434, 2015.
[44] S. Siami-Namini, N. Tavakoli, and A. S. Namin, “The performance of lstm and bilstm in forecasting time series,” in 2019 IEEE International Conference on Big Data (Big Data), pp. 3285–3292, IEEE, 2019.
[45] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, “Empirical evaluation of gated recurrent neural networks on sequence modeling,” arXiv preprint arXiv:1412.3555, 2014.
[46] S. Russel-Pulerii, “Gated recurrent units explained using matrices: Part 1,” Towards Data Science, 24.02.2019.
[47] Z. Zhang, “Missing data imputation: focusing on single imputation,” Annals of translational medicine, vol. 4, no. 1, 2016.
[48] K. Usman and M. Ramdhani, “Comparison of classical interpolation methods and compressive sensing for missing data reconstruction,” in 2019 IEEE International Conference on Signals and Systems (ICSigSys), pp. 29–33, IEEE, 2019.
[49] Y. Li, Z. Li, and L. Li, “Missing traffic data: comparison of imputation methods,” IET Intelligent Transport Systems, vol. 8, no. 1, pp. 51–57, 2014.
[50] W. Junger and A. P. De Leon, “Imputation of missing data in time series for air pollutants,” Atmospheric Environment, vol. 102, pp. 96–104, 2015.
[51] F. V. Nelwamondo, S. Mohamed, and T. Marwala, “Missing data: A comparison of neural network and expectation maximization techniques,” Current Science, pp. 1514–1521, 2007.
[52] H.-F. Yu, N. Rao, and I. S. Dhillon, “Temporal regularized matrix factorization for high-dimensional time series prediction,” Advances in neural information processing systems, vol. 29, 2016.
[53] L. Deng and D. Yu, “Deep learning: methods and applications,” Foundations and trends in signal processing, vol. 7, no. 3–4, pp. 197–387, 2014.
[54] J. An and S. Cho, “Variational autoencoder based anomaly detection using reconstruction probability,” Special Lecture on IE, vol. 2, no. 1, pp. 1–18, 2015.
[55] J. P. Assendorp, Deep learning for anomaly detection in multivariate time series data. PhD thesis, Hochschule für Angewandte Wissenschaften Hamburg, 2017.
[56] F. Lüer, D. Mautz, and C. Böhm, “Anomaly detection in time series using generative adversarial networks,” in 2019 International Conference on Data Mining Workshops (ICDMW), pp. 1047–1048, IEEE, 2019.
[57] Y. Lyu, Z. Han, J. Zhong, C. Li, and Z. Liu, “A gan-based anomaly detection method for isoelectric line in high-speed railway,” in 2019 IEEE International Instrumentation and Measurement Technology Conference (I2MTC), pp. 1–6, IEEE, 2019.
[58] W. Jiang, Y. Hong, B. Zhou, X. He, and C. Cheng, “A gan-based anomaly detection approach for imbalanced industrial time series,” IEEE Access, vol. 7, pp. 143608–143619, 2019.
[59] Y. Zheng, Q. Liu, E. Chen, Y. Ge, and J. L. Zhao, “Exploiting multi-channels deep convolutional neural networks for multivariate time series classification,” Frontiers of Computer Science, vol. 10, no. 1, pp. 96–112, 2016.
[60] Z. Guo, Y. Wan, and H. Ye, “A data imputation method for multivariate time series based on generative adversarial network,” Neurocomputing, vol. 360, pp. 185–197, 2019.
[61] X. Miao, Y. Wu, J. Wang, Y. Gao, X. Mao, and J. Yin, “Generative semi-supervised learning for multivariate time series imputation,” in Proceedings of the AAAI conference on artificial intelligence, vol. 35, pp. 8983–8991, 2021.
[62] S. K. Ainsworth, N. J. Foti, and E. B. Fox, “Disentangled vae representations for multi-aspect and missing data,” arXiv preprint arXiv:1806.09060, 2018.
[63] C. Ma, S. Tschiatschek, K. Palla, J. M. Hernández-Lobato, S. Nowozin, and C. Zhang, “Eddi: Efficient dynamic discovery of high-value information with partial vae,” arXiv preprint arXiv:1809.11142, 2018.
[64] A. Nazabal, P. M. Olmos, Z. Ghahramani, and I. Valera, “Handling incomplete heterogeneous data using vaes,” Pattern Recognition, vol. 107, p. 107501, 2020.
[65] D. P. Kingma and P. Dhariwal, “Glow: Generative flow with invertible 1x1 convolutions,” Advances in neural information processing systems, vol. 31, 2018.
[66] R. Godahewa, C. Bergmeir, G. Webb, R. Hyndman, and P. Montero-Manso, “Kdd cup dataset (without missing values),” June 2020.
[67] A. Botchkarev, “Performance metrics (error measures) in machine learning regression, forecasting and prognostics: Properties and typology,” arXiv preprint arXiv:1809.03006, 2018.
[68] A. H. Liu, Y.-A. Chung, and J. Glass, “Non-autoregressive predictive coding for learning speech representations from local dependencies,” arXiv preprint arXiv:2011.00406, 2020.
[69] D. Weber and C. Gühmann, “Non-autoregressive vs autoregressive neural networks for system identification,” IFAC-PapersOnLine, vol. 54, no. 20, pp. 692–698, 2021.
[70] A. Martínez-González, M. Villamizar, and J.-M. Odobez, “Pose transformers (potr): Human motion prediction with non-autoregressive transformers,” in Proceedings of the IEEE/CVF International Conference on Computer Vision, pp. 2276–2284, 2021.

ANNA RICHTER received her M.Sc. degree in Sensor Science and Cognition from the Technical University of Chemnitz, Germany, in 2021. Since August 2021, she has been a research scientist with the Division Engineering of Adaptive Systems (EAS), Fraunhofer Institute for Integrated Circuits IIS, Dresden, Germany, where she joined the Industrial Wireless Communication Group. Her research interests are in the areas of data science and artificial intelligence, with a focus on time series analysis, as well as the application of data analytics concepts to the state and failure analysis in wireless communication systems.

JYOTIRMAYA IJARADAR received his B.Tech degree in Electronics and Communication Engineering from Centurion University, India, in 2018. He is currently pursuing his MSc. degree in Computational Modeling and Simulation at TU Dresden, Germany. His research interests include machine learning, deep learning, artificial intelligence, IoT, time series analysis and wireless communication networks.

ULF WETZKER received the Diploma degree in computer science and systems engineering from the Ilmenau University of Technology (TU Ilmenau), Ilmenau, Germany, in 2008. Since October 2008, he has been a research scientist with the Division Engineering of Adaptive Systems (EAS), Fraunhofer Institute for Integrated Circuits IIS, Dresden, Germany, where he joined the Industrial Wireless Communication Group in 2012. His research interests are in the areas of wireless communication systems, data analytics and machine learning, with a special focus on anomaly detection and root cause analysis in wireless networks.

DR. VINEETA JAIN is currently pursuing her postdoctoral study at Fraunhofer Institute of Integrated Circuits IIS Division Engineering of Adaptive Systems EAS, Germany. She received her Ph.D. in Computer Science from Malaviya National Institute of Technology Jaipur, India. Her research interests include the area of security and privacy with a special emphasis on wireless networks and mobile security. She is a student member of IEEE.

DR. ANDREAS FROTZSCHER received his Dipl.-Ing. (MSEE) and PhD degree in Electrical Engineering from University of Technology Dresden, Germany in 2005 and 2010, respectively. From 2005 to 2010, he was a research associate at the Vodafone Chair Mobile Communications Systems at the University of Technology Dresden, Germany. From 2010 to 2011 he worked as research scientist at the Bell Laboratories of Alcatel-Lucent in Stuttgart, Germany. His research focused on physical layer signal processing, transceiver design. Since 2012 he is with the Fraunhofer Institute for Integrated Circuits IIS, Division Engineering of Adaptive Systems EAS. Since 2013 he is heading the group industrial wireless communication, working on monitoring and troubleshooting tools for industrial wireless networks, as well as ultra-reliable and low latency wireless communication systems for industrial automation applications.