Neurocomputing 441 (2021) 161–178

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

A review of irregular time series data handling with gated recurrent neural networks

Philip B. Weerakody *, Kok Wai Wong, Guanjin Wang, Wendell Ela

Murdoch University, 90 South Street, Murdoch, Western Australia, Australia

ARTICLE INFO

Article history:
Received 10 June 2020
Revised 4 October 2020
Accepted 15 February 2021
Available online 3 March 2021
Communicated by Zidong Wang

Keywords:
Irregular time series
Recurrent neural networks
Missing data
Imputation methods

ABSTRACT

Irregular time series data is becoming increasingly prevalent with the growth of multi-sensor systems as well as the continued use of unstructured manual data recording mechanisms. Irregular data and the resulting missing values severely limit the data's ability to be analysed and modelled for classification and forecasting tasks. Often, conventional methods used for handling time series data introduce bias and make strong assumptions on the underlying data generation process, which can lead to poor model predictions. Traditional machine learning and deep learning methods, although at the forefront of data modelling, are at best compromised by irregular time series data sets and fail to model the temporal irregularity of incomplete time series. Gated recurrent neural networks (RNN), such as LSTM and GRU, have had outstanding success in sequential modelling, and have been applied in many application fields, including natural language processing. These models have become an obvious choice for time series modelling and a promising tool for handling irregular time series data. RNNs have a unique ability to be adapted to make effective use of missing value patterns, time intervals and complex temporal dependencies in irregular univariate and multivariate time series data. In this paper, we provide a systematic review of recent studies in which gated recurrent neural networks have been successfully applied to irregular time series data for prediction tasks within several fields, including medical, human activity recognition, traffic monitoring and environmental monitoring. The review highlights the two common approaches for handling irregular time series data: missing value imputation at the data preprocessing stage and modification of algorithms to directly handle missing values in the learning process. Reviewed models are confined to those that can address issues with irregular time series data and does not cover the broader range of models that deal more generally with sequences and regular time series. This paper aims to present the most effective techniques emerging within this branch of research as well as to identify remaining challenges, so that researchers may build upon this platform of work towards further novel techniques for handling irregular time series data.

1. Introduction

Time series analysis aims to use the collected data of past observations to generate an accurate model that reflects the structure of the series, to enable the prediction, and classification of future events. The importance of this predictive capability has made time series analysis an essential part of data modelling in a wide range of industries, including financial predictions, medical diagnostics and environmental monitoring [1–4]. As a result, there has been significant research in the development of more accurate and robust algorithms and models for time series data analytics. Despite the impressive evolution of time series modelling from simple linear models to current cutting-edge deep learning networks, most of these models are only concerned with regular time series data [4].

Due to the numerous types of sensing devices or recording practices that generate data, it is rare for raw data to come with all input variables sampled at a constant rate with common timestamps and consistency across multiple variables [5]. Irregular data can arise for univariate and multivariate datasets. Univariate irregular data is generated from a single feature variable that is measured at a sampling rate without a consistent interval between observations, either due to unstructured manual processes, event-driven recordings, device or signal failure, and intentional omissions based on cost or importance. For multivariate datasets with multiple measuring techniques and instruments, the frequency with which each variable is recorded will often be different

* Corresponding author.
E-mail address: p.weerakody@murdoch.edu.au (P.B. Weerakody).

https://doi.org/10.1016/j.neucom.2021.02.046
0925-2312/

and result in missing values for one or more variables at a given timestamp. Fig. 1 shows an example of irregular multivariate data and its associated time intervals and missing indicators.

Missing data and irregular data are often used interchangeably in research material associated with time series data analysis. In the absence of the knowledge of the exact causes of data irregularity, missing data is generally defined with respect to a fixed interval feature space [6]. For the case of an irregularly sampled variable, there may be no defined expected sampling frequency, such as event-triggered sensors; therefore, it is not always possible to know when an expected value is "missing". This paper covers the broad concept of irregular data from univariate and multivariate systems, where there are irregular sampling intervals and, in the case of a multivariate feature space, where any given timestamped record does not have values for every variable in the feature space.

Irregularity of data is often defined in terms of the missing data percentage of a dataset, which is normally referred to as sparsity. In real-world time series datasets, there can be a considerable difference between the degree of irregularity of data belonging to different domains. For example, medical Intensive Care Unit (ICU) samples, can frequently contain 80% of missing data in multivariate feature space [7], while environmental datasets, such as the PM2.5 Beijing Air Quality, can contain 13.3% missing data [8]. In the case of high-frequency data, such as financial trading, the percentage of missing data can be interpreted as extremely high given that quotes or trades on multiple stocks rarely occur at the same time, and trading is conducted on a sub-second level. The average interval size between irregular samples is also a defining feature of irregular datasets [9]. Interval sizes are often categorised based on the arbitrary lengths of time or the relative lengths with respect to sampling frequency or the periodicity of the variables concerned. As an example, Tian et al. [9] categorised missing traffic observations into short periods which are less than 5 min and long periods which last hours to days.

Numerous methods have been developed over the years to deal with sparse and irregular time series data so that analysis can produce predictive models close to those that deal with regular data. However, many of the simple statistical techniques conventionally used for imputation such as zero, mean, moving average, last observation and simple regression can introduce problems of bias and loss of accuracy [6]. Statistical models such as ARIMA [10], Bayesian Network [11] and Gaussian Processes [12], as well as traditional machine learning models including Support Vector Regression (SVR) [13] K-nearest neighbour (KNN) [14], have been applied to time series problems as well. However, these approaches struggle to capture the complex temporal dependencies between observations in univariate and multivariate time series [15–17].

As will be further discussed in section 2, a number of recent machine learning models which include novel applications of convolutional networks [19], kernel learning [20] and Transformers (Self Attention) [21] have provided promising results in time series modelling. However, gated RNNs and their evolving architectures are still firmly amongst the leading methods for time series modelling. Based on the success of time series data modelling with gated RNNs, the modification and enhancement of these models to handle irregular time series data have been viewed as an important area for researchers.

In the development of this review paper, the search and selection of references were performed according to the following steps:

Step 1. Publication database search: A keyword search was performed on the following databases and search facilities: IEEE, ResearchGate, Pubmed and Google Scholar. The following keywords were used: irregular time series, recurrent neural networks, missing data, and imputation methods.

Step 2. Preliminary screening of articles: The publications were selected for further review if they

- Specifically focused on time series data with irregularity and missing values
- Gated RNN models (i.e. LSTM, GRU) models were directly discussed within the article

Step 3. Bibliography search: Further articles were identified for initial review based on references within articles selected from the previous step. They were also screened as per the criteria in step 2.

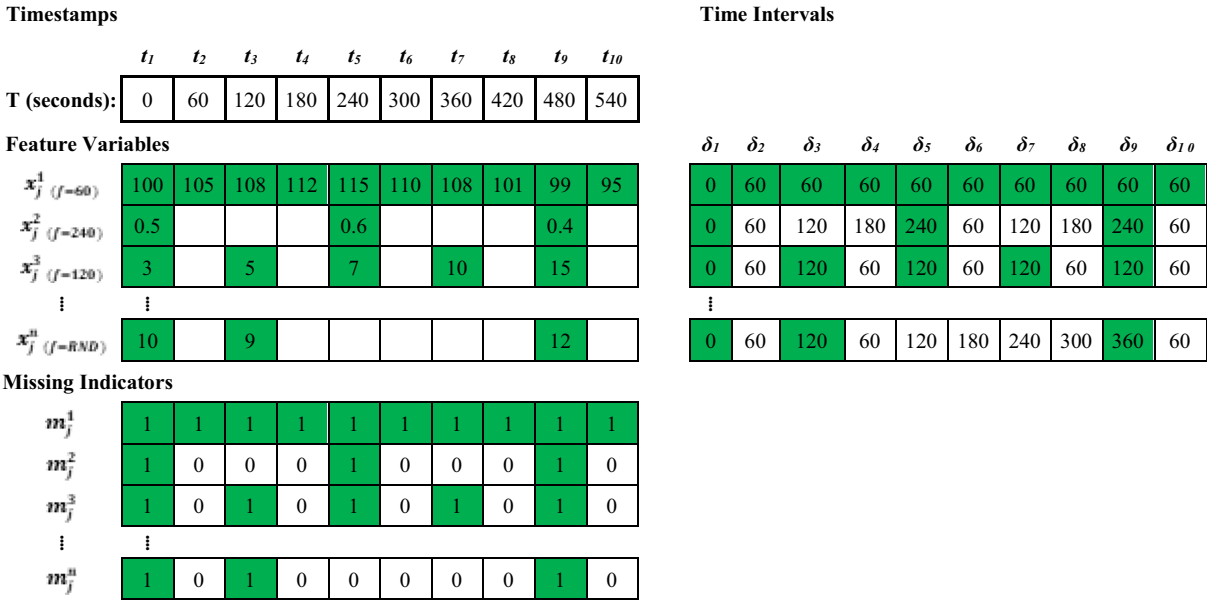

Fig. 1. Multivariate time series x_j^i, with features $i \in \{1, 2, 3, .., n\}$, observations $j \in \{1, 2, 3, .., 10\}$ and sampling frequency (f). m_j^i corresponds to the missing indicators, where $m_j^i = 1$ if x_j^i is observed and $m_j^i = 0$ if x_j^i is missing. δ_j^i corresponds to the time interval between x_j^i and x_k^i, where k is the last true observation before the j^{th} observation.

Step 4. Result filtering: The articles selected to this point were further filtered based on the date and their influence, using the following criteria

- Date: published primarily within the last six years
- Influence: published in high-quality journals/conferences or having high citation numbers

Step 5. Content selection: Papers filtered from the previous step were reviewed and referenced where applicable. For inclusion in the detailed review section of this paper, further selection criteria were imposed to ensure that the articles focused on a solution for handling irregular time series data in which a gated RNN model played a primary role. Based on this condition, a number papers that used gated RNN based models were examined but not included in the detailed review section either because the core technique for handling the irregular time series data was a method other than RNNs or because the model was applicable to regular time series data and did not show an obvious application to irregular time series data. The resulting set of models reviewed is, therefore, a much smaller collection than the set of leading-edge models that have been presented in recent times for sequence modelling or regular time series modelling. However, in order to provide context to gated RNN models which handle irregular time series data, we provide a background description of the state-of-the-art models for handling general time series data and point out the continued leading role of the gated RNN's play within this broader field.

This paper aims to conduct a systematic review of the use of gated RNN models applied to irregular time series data. General reviews of machine learning methods for managing irregular data have tended to provide high-level references to proposed models, given the breadth of methods required to be covered. Alternately, this paper attempts to focus on gated RNN models and techniques proposed by a representative selection of papers on the topic. We are currently not aware of a published review of irregular time series data which specifically focuses on RNN models. The need for a review dedicated to RNN solutions for irregular time series data modelling is because LSTM and GRU based models have been providing outstanding results in the modelling of sequence data, including time series data, which have not only been outperforming advanced statistical techniques but have done so with minimal prior knowledge or assumption of the data. In comparison to other leading deep learning architectures, RNNs are often better suited to temporal data, can handle time series of varying length input [18] and can be adapted for dealing with irregular data with missing values. RNNs have the advantage of learning the complex temporal dynamics involved with time series data and are therefore well equipped for tackling irregular time series data in which there is often temporal information associated with the observations as well as with the missing data.

Given the sheer quantity of irregular time series data present in many real-world domains and the confidence that gated RNNs can address the problems associated with the modelling of this type of data, we believe that a detailed review of gated RNN developments is warranted. We shall provide information on the current state-of-the-art models in this field, which will provide a platform for further research. The main contributions of this paper are:

(1) It summarises research developments in irregular times series handling by gated RNNs and categorises the research into two methodologies: imputation and data generation focused methods, and predictive model focused methods.
(2) Provides the detailed workings of selected models to enable low-level understanding of the methods used.
(3) It highlights common techniques or algorithms applied across a number of proposed models, which may be collectively applied to improve model capabilities.
(4) It suggests emerging models which warrant further development as well as other necessary research directions in this area based on current challenges.

The remainder of this paper is organised into five main sections, commencing with an overview of some of the state-of-the-art machine learning models for modelling regular time series data, in section 2. Section 3 provides a description of the two main categories for handling irregular time series data. The body of the review is provided in Sections 4 and 5, which are further categorised into subsections based on the types of techniques used. Section 6 provides a summary of the evaluation metrics and datasets used by papers within the review, as well as an analysis of the comparative performance and complexity of the models. The conclusion, in Section 7, follows with the analysis of the main findings, areas of ongoing research and some remaining challenges in this field of research.

As an initial point on nomenclature, the term RNN has been used in this paper as well as some referenced papers to refer to the category of recurrent neural networks, which include the vanilla RNN, LSTM, GRU and other associated variants. When referring to the original RNN model based on the work of Rumelhart et al. [22], the term vanilla or conventional RNN shall be used.

2. Background – Machine learning models for time series

Machine learning (ML) techniques are now established at the forefront of time series forecasting, and empirical research has demonstrated that ML algorithms for time series frequently outperform statistical models [23]. Through the rapid developments of sequence modelling in NLP and Computer Vision (CV) applications, time series modelling with ML has benefitted from the resulting algorithms. The field of ML for time series modelling has now been widely used such that state-of-the-art techniques has evolved from base techniques like Kernel methods [20,24], K-nearest neighbour (KNN) [14], Auto-Encoder Networks [1], Convolution Neural Networks (CNNs) [25] and Recurrent Neural Networks (RNNs) [26], to more sophisticated models involving the advancement of these base techniques, as well in combination with other techniques.

Convolution based models such as Wavenet [27], which developed from text to speech modelling have improved the long term dependency requirements of time series modelling by dilated convolutions, which allow the receptive field to increase exponentially with layer depth. Borovykh et al. [28] adapts the WaveNet architecture for time series forecasting, with stacked dilated convolutions and applies the model to noisy financial data in the form of multivariate time series. Lea et al. [19] propose Temporal Convolutional Networks (TCN) for video-based action segmentation which captures long-range time series patterns using a hierarchy of temporal convolutional filters. CNNs have been successfully combined with RNN models for time series forecasting in numerous instances such as Wu et al. [29] which uses a GRU network to encode the temporal patterns of each sequence with a low dimensional representation and then integrates these into a convolutional network to model the interdependencies between sequential patterns with different time resolutions. Other hybrid convolutional architectures such as the convLSTM [30] and MLSTM-FCN [31] have also been successfully applied for time series modelling. The MLSTM-FCN combines CNN and RNN architectures as well as Attention to offer a state-of-the-art method for multivariate time series classification.

Kernel methods and subsequently, Multiple Kernel Learning (MKL) have also provided an important stream of development in time series modelling. Early kernel methods solved time series estimation problems using SVM for regression, Gaussian Process Regression and Radial Based Function neural networks. In recent works, Futoma et al. [32] proposed a Multitask Gaussian Process RNN Classifier which accounted for time series with high uncertainty, frequent missingness, and irregular sampling rates. Shukla et al. [33] proposed an approach with RBF Kernel interpolation layers followed by a GRU prediction layer to address the difficulties of the complexity of Gaussian process interpolation layers of the previous work [32]. In order to prevent reliance on a predefined parametric kernel given a priori, recent models have used Kernel Learning, which aims to learn effective kernels from the data. One of the most widely used kernel learning methods is MKL [34]. Huang et al. [35] employ MKL and deep learning to predict oil price time series, while Sahoo et al. [20] extends MKL to an online learning scheme which sequentially learns a kernel-based regressor and dynamically searches a pool of multiple diverse kernels in order to optimize time series prediction.

Tensor Factorization (TF), which decomposes a tensor into multiple low-rank latent factor matrices, has been used for addressing regular time series modelling as well as time series with sparse and missing values. Yu et al. [36] present a temporal regularized matrix factorization (TRMF) framework which uses a novel temporal regularizer to incorporate temporal structures into a standard Matrix Factorization (MF) formulation. The scalable MF methods can handle high dimensional time series data, even when characterised by a high level of missing values. Wu et al. [37] fuses LSTM and Tensor Factorization into a framework to model dynamic time interactions across several dimensions, in the area of user-item interaction modelling. The authors develop a neural network based Tensor Factorization model (NTF) which inputs a three-way tensor (i.e. user-item-time) and learns the latent embeddings (i.e. factors in TF) for each dimension of the tensor. The LSTM is used in the model to capture dependencies between multi-dimensional interactions based on learned representations at each time slot, which do not require fixed time intervals. Unlike standard factorization techniques of using the dot product of learned representations for predictions, the NTF concatenates the factors together and feeds them into a Multilayer Perceptron (MLP).

Gated variants of RNNs have received considerable interest with respect to time series analysis [38]. RNNs have been increasingly applied to time series problems due to the achievements that Long Short Term Memory (LSTM) and the Gated Recurrent Unit (GRU) have had in improving the state-of-the-art performance in machine translation, speech recognition and other natural language processing (NLP) tasks that require capturing the context of words based on temporal dependencies within a sequence of text [39]. Gated variants of RNNs are inherently designed to maintain an internal state memory through their recurrent feedback mechanism, which makes them highly suited for modelling sequential time series data. Their capacity to capture complex non-linear temporal dependencies which can extend from short to long term as well as across different variables within a multivariate system, position these models at the forefront of time series analysis research [8]. Within the range of RNN based models, sequence to sequence architectures [40,41] have proven powerful in applying multilayered recurrent neural networks (RNN) for time series feature extraction. Malhotra et al.'s Timenet [42] demonstrate the strength of pre-trained deep RNNs for time series classification, using a Sequence Auto-Encoder (SAE) based on sequence-to-sequence models to learn latent representations of time series for subsequent input to various classifier models. Qin et al. [43] successfully combine an LSTM encoder-decoder architecture with dual Attention, applied at the input state to extract relevant driving series at each time step and also a second attention layer to select relevant encoder hidden states across all time steps.

A particular type of time series which frequently arise in applications such as transportation and the environment is the spatial–temporal series which poses particular challenges due to the presence of both spatial patterns, and long and short temporal patterns. Recently, many works have shown the strength of Deep Neural Networks (DNN) in solving spatiotemporal prediction problems, such as Liang et al [44], for the prediction of urban crowd flows from real-time data using convolution and residual layers for feature extraction. Asadi et al. [45] forecasted large scale traffic flows using a DNN framework incorporating convolutional layers, convolutional-LSTM layers and an autoencoder. DNNs in the form of combined CNN and LSTM models have frequently provided a successful solution for this type of time series due to convolutional networks exhibiting outstanding performance on spatial data while recurrent networks perform exceptionally well on temporal data. A convolutional-LSTM (convLSTM) is used by Shi et al. [30] for rainfall prediction for a spatial–temporal series, and Yuan et al. [46] models traffic accident predictions using a convLSTM, in which state to state transitions involve convolutional operations that generate 3d tensors. Models leveraging the strengths of gated recurrent neural networks have also been applied to spatial–temporal problems as per Huang et al. [47] and Pan et al. [48], which present GRU or LSTM based sequence-to-sequence frameworks with Attention for spatial–temporal time series. Lian et al. [49] use a multi-level attention mechanism, incorporating spatial attention and temporal attention within an encoder-decoder LSTM architecture to model dynamic spatial–temporal dependencies. Missing values in spatial–temporal forecasting are addressed by Fan et al. [50], using an LSTM model and three variants to impute missing values, which are based on latest observation, mean with a weighted sum of last observation, and mean using exponential decay weighting.

Transformer models, originally introduced by Vaswani et al. [51] have demonstrated very impressive results on a variety of sequential tasks involving NLP and image processing, and are beginning to make progress in time series forecasting applications. Transformers and more specifically Self-Attention use positional encoding as they have no knowledge of the sequence order. Therefore, they usually incorporate positional information through time step embedding. Song et al. [21] apply Self-Attention on medical time series data, utilising a multi-headed attention operation and a masking mechanism. Ma et al. [52] look at imputing missing values in multivariate time series data using a cross attention method that jointly captures the self-attention across multiple dimensions. Li et al. [53] address the main perceived issues associated with applying Transformers to time series data; their insensitivity to local context and memory intensiveness for long sequences. In response to these issues, the authors propose new attention mechanisms that use convolutional self-attention to enhance awareness of local context and log attention to reduce the existing quadratic time complexity of standard transformers.

Although the prospects for applying Transformers to time series data show much potential, research in this area are still limited, and the results are mixed. Within time series modelling Transformers still struggle with feature extraction due to their lack of true recurrent gradients and difficulty in encoding positional information. The memory intensity of standard transformers is another limitation for very long sequences and their quadratic time complexity in comparison to the linear complexity of RNNs, makes tasks involving 1000 s of timestep extremely challenging. These limitations with transformers along with the continued evolution of gated recurrent networks, confirm that there is still an important place for continued research into gated RNN models for time series data. Gated RNNs with attention as well as innovative new

structures and algorithms to handle time series issues are maintaining gated RNNs as a leading technique for time series modelling which make it very difficult for non-recurrent architectures to replace entirely.

3. Approaches for handling irregular data

There are two conventional approaches for handling irregular datasets: (1) use of imputation strategies to produce a resultant time series without missing values for input into a predictive model or (2) develop a predictive model that can handle irregular times series with minimal pre-processing of input data [33].

3.1. Imputation and data generation focused methods

Numerous imputation methods have been used for time series data, which include: replacement [54], interpolation [3], autoregression [55], resampling [56], Gaussian processes [57], and machine learning methods [3]. One of the most common baseline methods used or tested against is missing value replacement with statistical values such as mean, mode or median values. Although frequently used, these statistical techniques have the obvious limitation of disregarding important temporal information. Machine learning techniques for data imputation, outside of deep learning methods, have frequently applied K-nearest neighbour [58], Matrix Factorization [59], and Expectation Maximisation [60]. Along with the non-machine learning methods stated above, these techniques tend to produce good results when datasets have a small number of missing values but perform poorly with sparse datasets with a high proportion of missing values [17,61]. Also, temporal relationships between observations, and important information from missing values and their patterns are not adequately captured by these techniques.

Deep learning through RNNs provides for more sophisticated strategies of data imputation for multivariate time series data, taking into account temporal correlations within each series as well as exploiting correlations across variables [57]. The missing data can be imputed by values estimated by a proposed RNN model through a number of methods, including where the missing value estimates are the trained variables of a deep network or where they are calculated based on learnt weights within a formula. In addition to estimation of imputation values with RNN models, the generation of a complete time series for input into the prediction layer may also take the form of a data generation, as in the case of Generative Adversarial Networks (GAN) [62], which can incorporate RNNs to produce a time series based on the characteristics of the original time series.

3.2. Predictive model focused methods

Directly modelling time series datasets with irregular data, without resampling nor a significant imputation stage is often done by masking the missing values or simple filling of NaN values (i.e. zero, mean, forward fill) such that models may be used on irregular datasets. A number of these types of models have utilised the set of missing data indicators, rather than disregarding the missing data. As per Lipton et al. [26], this technique has been used in multiple research papers to improve the performance of models for irregular time series data. Predictive models which do not rely on the imputation stage for handling missing time series data are also more focused on identifying observations or segments of observations in a sequence that are more important to the model. This generally involves algorithms which apply focus to observations within a sequence and identify temporal patterns which allow unimportant observations to be skipped.

Vanilla models of the LSTM and GRU have difficulty in handling irregular time series data with missing values, often leading to sub-optimal analyses and predictions [63]. The simple application of these off-the-shelf models on irregular time series data, using missing value filling or temporal smoothing, do not allow the model to distinguish whether an observation is a true value or an imputed value. This fails to allow the model to utilise the rich information associated with the missing data and irregularity. Although concatenating missing value indicators and time interval vectors have addressed this issue to some extent [17,26], more innovative RNN based solutions have been developed, which allow for better understanding of the missing data, and the subsequent patterns and relationships within the data. These novel approaches include modifications to the gating architecture of LSTM or GRU units and updates of their equations to allow for missing value and temporal decay factors.

A selection of RNN models for handling irregular time series data, categorised by their relative focus on an imputation-based approach or predictive model-based approach is presented in Section 4 and 5, respectively. Those models are further sub-categorised based on specific techniques or algorithms used by the proposed models, as listed in Table 1 below.

4. Imputation and data generation focused models

Many of the reviewed recurrent models for handling irregular data have focused on an imputation or data generation stage to estimate missing values and reconstruct a complete time series. This approach converts an irregular time series into a regular one so that a prediction layer can take the resulting series and generate final predictions. The benefit of an approach that concentrates on the construction of a regular time series without missing values is the resulting flexibility of allowing the series to be input into any type of prediction layer for the subsequent regression or classification task.

4.1. Unidirectional RNN based imputation models

Conventional RNNs are an advancement of feedforward deep neural networks to model sequential data such as time series and natural language sequences. Their recurrent structures capture the temporal dynamics of the time series data, through the use of hidden states which work as a memory that is dependent on previous states and the current input. Two leading gated RNN variants are the LSTM and GRU, which are designed to capture long and short term dependencies while overcoming the vanishing gradient problem. The following imputation-focused models use conventional forward directional gated RNN units in several different methods to handle irregular data. These methods include the imputation based on estimated missing value calculation using RNN next step prediction, exponential decay towards the mean or last value, and applying deep learning to produce a higher-order series with latent temporal dynamics.

Table 1
RNN model categories for handling irregular time series data.

Imputation and Data Generation Focused Models	Predictive Model Focused Methods
Unidirectional RNN	Input Augmentation – missing indicators
Bidirectional RNN	Input Augmentation – time intervals
Bidirectional RNN with Attention	Time Decay Factors
Sequence-to-Sequence (Seq2Seq)	Ordinary Differential Equations
Generative Networks	Gated RNN Modified Structure

In one of the simplest models reviewed, conventional RNNs have been used for imputation through prediction of the next value based on previous observations. Nguyen et al. [63] apply this approach for multistep prediction modelling of Alzheimer's disease (AD) progression trained on irregular data. Their main strategy uses the RNN to interpolate and extrapolate the missing data during training to generate a complete time series dataset and predict the next state of the patient. An LSTM model is trained such that gradients of the errors between true values and predicted values are backpropagated to update the model. The loss function is only calculated at time points where true observations are available. The complete imputation process combines linear interpolation with RNN model estimation on a derivation of the Alzheimer's Disease Neuroimaging Initiative (ADNI) database. Yuan et al. [64] apply LSTM in a similar approach in the environmental monitoring domain, for imputation of missing data in the time series of air pollutants to improve the PM2.5 concentration prediction accuracy. Both models tackle missing values but rely on a uniform sampling frequency in order for the method to work and therefore do not address irregular sampling frequencies of multivariate data.

Simple unidirectional RNN imputation can be enhanced for long sequences by leveraging short paths from skip connections, as applied by Residual Networks [65]. Shen et al. [66] apply this concept of short paths to model underlying correlations between missing values and their previous true observations to improve time series imputation. An imputation network called Residual Imputation (RIMP-LSTM) is proposed, using LSTM units and introduction of a residual sum unit (RSU), which connects the LSTM's hidden states and the RSU's previous states, at each time step. The RSU integrates the information from its historical states via residual paths in a similar way to weighted graphs, which utilises previous observations and reduces the impact of missing values. All parameters in the proposed model are learnt via conventional backpropagation through time. Zhang et al. [67] build on this use of residual short connections for RNN networks, with an imputation formula which also includes a decay factor for shortcut connections.

A common approach to develop imputation values for missing values in time series is to use an exponential decay function based on time intervals, such that the longer the interval, the greater the decay. Kim et al. [68] implement this method through a proposed imputation module which is combined with an LSTM, as part of a recurrent structure which systematically imputes missing values in both forward and backward directions and then performs prediction. The proposed Temporal Belief Memory (TBM) model, considers time continuity and missing data patterns for handling irregular data within the medical domain, more specifically the early prediction of septic shock, using a multivariate time series dataset with an average of 81.84% missing data. The TBM is built as a memory module with two gating units, a missing information gate "m" and a belief gate "b". Operationally, the missing gate indicates whether a value is missing or present, and the belief gate decides if the last observation is passed on as the imputed value based on the temporal reliability of the last observation. For missing values, the belief of the last observation is calculated using a time interval based exponential decay, and the value is imputed if the belief is above the threshold otherwise it is set as the mean for the feature variable. The forward and backward evaluation of imputation values in this model does not use a bidirectional RNN. The TBM module outputs to an LSTM, which is trained with inputs with imputed values. The limitation of models such as the TBM is that the LSTM is not able to differentiate between imputed values and observed values, which in comparison to more sophisticated imputation and predictive models such as in [17], valuable missingness information cannot be used by the LSTM.

Deep learning networks with multiple layers have the advantage of being able to identify latent patterns that shallow networks are often not able to recognise. This approach can also be applied in missing value imputation, which can capture hidden temporal dynamics from these higher layers instead of working purely in the input observations feature space. Zhou et al. [69] present a deep neural network called an Iterative Imputing Network (IIN) that captures these latent temporal dynamics of time series data. This model relies on initialising the missing values with simple statistical estimates and then updating these estimates with a multilayer Iterative Imputing Network (IIN). An LSTM network is at the core of the model, which takes the inputs and generates a latent hidden layer. The paper uses the vanilla LSTM for regularly spaced data and the Phased LSTM [5] for irregularly spaced data. Each layer in the model is called an Imputation Network (IN), which captures a summary representation of the context of each missing value by taking the left and right neighbouring observations or imputations with separate forward and backward RNNs, respectively. An output layer takes these representations and learns to impute the current missing value. The complete IIN solution incorporates a multilayer model of Imputing Networks (IN) that share the same set of weights, where the output imputation of an IN block is fed into a higher-level block as input, which is comparable to an iterative process. The IIN model is expected to be particularly effective in imputation for missing blocks of data rather than just single observations; however, its iterative process is likely to be computationally expensive.

4.2. Bidirectional RNN imputation models

Bidirectional RNN models offer a distinct advantage over unidirectional RNNs for missing value imputation, in that they can exploit the long-range context dependencies of the past as well as the future time steps of a missing observation. Bidirectional RNNs allow for training in both directions simultaneously with a separate forward hidden layer and backward hidden layer [70]. The architecture, as shown in Fig. 2, is similar to putting two independent RNNs together, where the input sequence is fed in regular time order for one network and in reverse time order for the second network. The outputs of the two networks are usually concatenated or summed at each time step. Because the backward component provides the ability for the network to see future data and learn its weights accordingly it allows the estimation of missing values to capture specific dependencies which would not have otherwise been identified by the conventional LSTM or GRU. This is especially true in cases where a sequence is long and the time interval between two observations is large, or when the missing values occur within the first few time intervals when previously observed values are not available.

Yoon et al. [71] present a novel multi-directional RNN (M-RNN) model for estimating missing random data, which takes into account the intra-data relationships within a data stream and

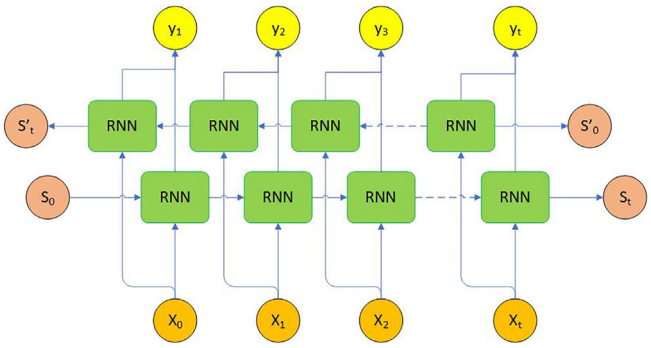

Fig. 2. Bidirectional RNN.

inter-data relationships across data streams. The M-RNN model consists of two parts, an interpolation block (intra-stream) and imputation block (inter-stream), which aim to minimise the error between estimated input values and actual input values. Interpolation is carried out by a bidirectional GRU model, with modified timing such that the inputs into the hidden layers are lagged in the forward direction and advanced in the backward direction. Imputation is implemented independently of the timestamps and involves fully connected layers. The M-RNN model is successfully evaluated on multivariate electronic health records from the MIMIC dataset.

The BRITS (Bidirectional Recurrent Imputation for Time) [8] model treat imputed values as trainable variables which are fully updated during the backpropagation process, unlike the M-RNN's approach of treating missing values as constants. Cao et al. present an imputation focused model which considers the correlations within and between feature variables. BRITS utilises missing data indicators, time intervals and a temporal decay factor, which represents the missing patterns in the irregularly sampled time series. An intermediate vector is built from a combination of a historic estimation based on correlation with past observations within a single variable, and feature estimation based on the correlation between other feature variables at a given timestamp. The vector is input into a bidirectional RNN, where each value in the time series can be estimated in the forward and backward direction. The tested model uses the LSTM which treats a new value in the series as a variable of the RNN graph, such that the missing value to impute is learnt. The BRITS model and M-RNN model are notable for being in a small set of imputation models which consider both relationships within a variable data-stream as well as across data-streams (i.e. inter-feature correlation), which is expected to enhance the accuracy of missing value estimation in comparison to models without this capability.

4.3. Bidirectional RNN with attention

Both unidirectional and bidirectional models can be enhanced by the addition of an Attention mechanism which enables the network to focus on relevant parts of the input more than the unnecessary segments when performing prediction tasks [72]. Attention is commonly used in sequence-to-sequence models, which attempt to encode the entire input sequence in a single hidden state, which is particularly challenging in practice, especially for long input sequences [73]. The attention mechanism addresses this problem by allowing the network to refer back to the input sequence states, instead of forcing it to encode all information into one fixed-length vector. Therefore, each decoder output depends not just on the last decoder state, but on a weighted combination of all the input states. The attention mechanism is effectively another structure to the network and is therefore learned as part of standard model training via backpropagation. RNN models have demonstrated performance improvements with the addition of the attention mechanism for time series modelling tasks [74], with LSTNet [39] being one of the first to implement an LSTM with an attention mechanism for multivariate time series forecasting. The following papers extend the application of RNN with attention mechanism for time series with missing values.

Nguyen et al. [75] use a bidirectional LSTM model which incorporates multiple attention mechanisms to predict mortality outcomes in ICUs. The model employs a bidirectional LSTM and two attention layers. The first layer is a sensing layer which decides whether to observe and incorporate parts of the current measurements, and the second layer is a reasoning layer, in which time steps are selectively weighted and combined. The complete model contains four components: data pre-processing, bidirectional LSTM, attention and classifier. The pre-processing component uses simple mean imputation of missing values which are then input into the LSTM. This produce a sequence of state vectors which go through the attention layers. Final values are max-pooled to give a feature vector, which can be input into any classifier layer to generate predictions. This model illustrates generation of a representative feature vector for final prediction rather than utilising the original input feature space.

Attention is also applied by Sing et al. [76] in the Flexible Irregular Time Series Network (FIT), which uses a fully connected neural network (FCN) and Bidirectional LSTM. Their model imputes missing values into time series, based on representations learned from input values, missing indicators, time intervals between observations and average values. Variants of FIT are also introduced, the Vertical Flexible Irregular Time Series Network (FIT-V), which considers other features correlated with the variable at the same timestamp. Differing sampling frequencies of multivariate data are handled by variants Multi-FIT and Multi-FIT-V, which split signals into dedicated FIT branches which are trained to handle data with a specific frequency range, prior to concatenation before target prediction. The FIT network's resulting series is input into a Bidirectional LSTM (Bi-LSTM) with Attention. The Bi-LSTM's hidden outputs are used by an attention layer to attain the final sequence representation for the time series, which is then used by the prediction layer. Both [75] and the FIT model successfully use attention; however, the FIT based models also explicitly address multifrequency sampling and inter-feature correlations.

4.4. Sequence-to-sequence models

In the application of unidirectional or bidirectional models for time series data, sequence-to-sequence models have been successfully used for time series forecasting tasks and also provide a technique for handling irregular time series data. The Sequence-to-Sequence model was first developed by Sutskever et al. [41] which introduced the concept of using one RNN to read an input sequence one step at a time, producing a fixed dimensional array, which is then fed into another RNN that extracts the output sequence from that array. As shown in Fig. 3, the first RNN is an encoder that compresses the information in the input sequence to a fixed dimensional array, as the last hidden state of the RNN. The second RNN acts as a decoder and calculates the probability of the target sequence. The LSTM was used in Sutskever's work due to its ability to learn data with long temporal dependencies and therefore han-

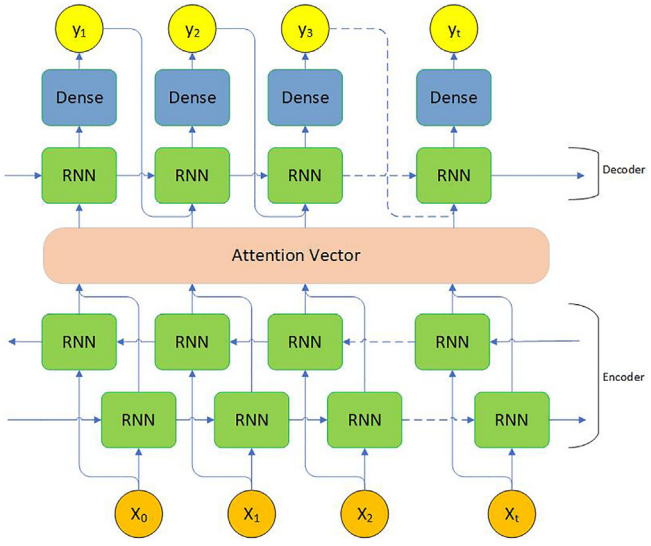

Fig. 3. Sequence to sequence network layers with attention [77].

dle significant time lags between inputs and their corresponding outputs. Sequence-to-Sequence models are potentially well suited for handling incomplete time series, whereby the decoder can generate a new representative time series that can be used by a prediction layer or where the model learns values to impute into the original input time series.

Tang et al. [78] propose a model specifically for time series classification based on the sequence-to-sequence model. The encoder LSTM takes input data from time series of arbitrary lengths and extracts information from the raw data, which is fed into the decoder LSTM to construct fixed-length sequences of automatically extracted features. The proposed model also uses an attention mechanism, so that along with the information from the encoder, the decoder LTSM can view the raw input data and focus attention on the parts that are most relevant to the feature under construction. The LSTM units used in the model are unidirectional. The model can be applied to multivariate data with different sampling rates between each series by expanding each series to a common longest length with gaps automatically filled in with "0″ values. The "0" representing no input for the specified time step. Although the proposed model allows for inputting irregular data it does not use the sequence-to-sequence approach for recovering missing data unlike the following models, which more directly address the problem of missing data and also enhance the model by using bidirectional RNNs.

Zhang et al. [77] use a bidirectional RNN within a sequence-to-sequence model, which allows for consideration of observations before and after the missing data. The proposed Sequence-to-Sequence Imputation Model (SSIM), with a variable-length sliding window algorithm, is used for generation of arbitrary target data from a water quality time series with variable-length input data. Sliding window algorithms are commonly used for generating time series training samples for deep learning models. In this case, a variable sliding length window algorithm is used to process time series data sets with small numbers of measurements. The SSIM architecture also employs an attention mechanism. The encoder takes input time series data and maps it to a higher dimension vector using a Bidirectional LSTM, which is passed on to the LSTM decoder that generates target sequence data. The attention function allows the decoder to focus on a specific range of the input sequence for the different outputs. The final model adds a fully connected layer on top of the LSTM layer to generate predictions with continuous values.

Dabrowski et al. [79] propose an LSTM based sequence-to-sequence model to predict missing values of a time series sequence, using two encoders, a forward encoder and a backward encoder, and bidirectional decoder. The model can handle arbitrary length input and output sequences as well as multivariate data where there are missing values across all features at a given time step. The model also uses a scaling factor which decays as the prediction diverges from the observed data, and is applied to the outputs of the forward and backward decoder RNNs before combining in the final output to the fully connected prediction layer. As an advantage over the SSIM model [77], this model is not required to learn that there is a difference between the observations before and after the missing data. However, unlike SSIM, it does not include an attention mechanism which may allow for further improvement.

4.5. Generative adversarial network

In contrast to the previous methods outlined in this section, generative adversarial networks treat the problem of missing value imputation with a data generation approach. GANs are generative models that create new data instances that resemble training data by training two neural networks, a generator and a discriminator, as shown in Fig. 4. The generator uses random noise as an input to generate samples, while the discriminator inputs the generated samples and real samples in order to learn to distinguish between the two. The two networks are trained alternatively, to drive competition between them and therefore generate more realistic samples. Esteban et al. [80] proposed recurrent generative adversarial networks (R-GAN) to produce realistic medical multivariate time series data (MTS). However, the authors do not address missing values and handled the issue by down-sampling, therefore limiting its ability to cope with MTS data. Several CNN based GAN models [81] have also been proposed for tackling missing data with data generation, but they do not consider the temporal dynamics of time series data. Recurrent GAN methods are, however, capable of learning temporal relationship within the same variable and the associations between variables.

Multivariate time series imputation with Generative Adversarial Networks is proposed by Luo et al. [62], utilising a recurrent GAN model which can learn temporal dependencies and the nature of complex distribution in multivariate time series in order to generate the missing values in the time series. The work is inspired by the success of GAN in image imputation and is extended to time series in order to learn latent relationships between observations with irregular time lags. The design of the GAN involves a GRU as its basis, however, to handle irregular time lags and learn implicit information from the time intervals; the GRU is slightly modified into a new form called GRUI. A time decay vector is introduced to decay the influence of the past observations by multiplying the decay elementwise with the GRU hidden state previous value. The proposed model performs comparatively well on a real-world clinical dataset and air quality dataset and includes achieving a new state-of-the-art mortality prediction result on the frequently tested PhysioNet ICU dataset.

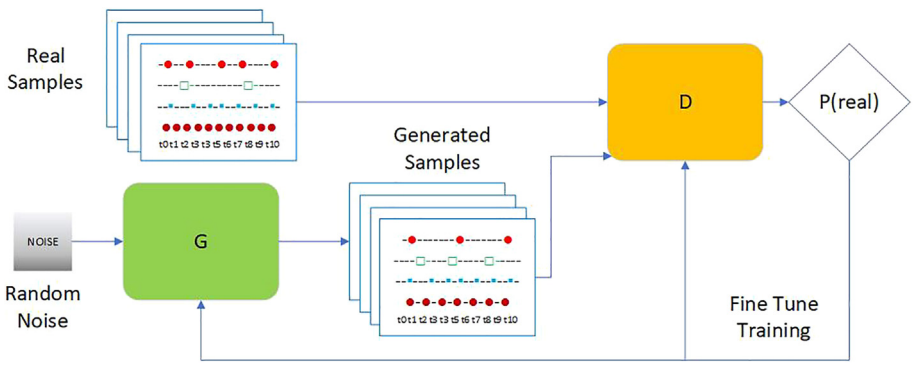

Fig. 4. Generative adversarial network (GAN).

5. Predictive model focused methods

The following models concentrate on building a prediction layer that can input irregular time series with minimal pre-processing for missing value imputation, to generate accurate classification and regression outputs. Unlike the models in the previous section, the main aim of these models is not to develop the most accurate estimations of the missing values. Therefore, simple imputation techniques are often employed, which include replacing missing values with zero, last observed value or mean value. The focus of these models is in the prediction layer's ability to handle the irregular data. Although they may perform some level of imputation, the most important part of a model's characteristics which handle the irregularity and missing values, is considered to be the prediction layer.

5.1. Input augmentation

Augmenting inputs with additional features which provide direct information on missing data, such as a binary variable indicating missingness and the time delta between missing values, are methods of enhancing RNN models for handling irregular time series data. RNN models which are modified to include these new inputs can learn the associated patterns and improve their predictive performance. Little and Rubin [6] were one of the earliest to identify that missing values and their patterns provide rich information on a dataset and can be utilised to minimise the missing data problem raised by irregular time series data. Several RNN based models have proposed modifications to the standard LSTM or GRU equations by augmenting the input time series with a missing indicator vector, which provides information on the relationship between missingness and the observed values of variables in the input vector [26]. Augmentation of inputs with the intervals between observations can also provide information on missing values as well as more explicitly provide information on the relationship between observed values. The premise of these methods is, however, only applicable for the case of informative missing patterns or intervals and not when they occur completely at random, in which case such methods will have limited benefit.

Lipton et al. [26] provide one of the first systematic studies which validate the importance of missing indicators for irregular clinical time series modelling based on RNN. Lipton et al. directly model missing values as features using a binary indicator input. The proposed model uses a conventional LSTM to output to a fully connected layer. A missing indicator augments the input data, which consists of a binary missing value for each input variable. The LSTM's hidden state computation is therefore used to learn both arbitrary functions of the past observations and missingness patterns. This research includes two simple imputation strategies, forward-filling and zero imputation, alongside direct modelling with missing indicator variables. The paper's results show that RNN's make substantial use of binary indicators for missing data and improves the classification of patient ICU episodes while using a conventional LSTM structure. Lipton et al. focus specifically on the benefits of missing indicator augmentation resulting in a relatively simple model, while Che et al. [17] combine this technique with several other methods to develop a more comprehensive approach.

Choi et al.'s [82] "Doctor AI" model inputs medical codes along with their time intervals into a GRU model to predict the next medical event. The model learns the patient status at each time step, through a vector representation based on the weights of the hidden units. The full network architecture contains an initial embedding layer to map high dimensional input vectors into a lower dimensional space, followed by one or more GRU layers which learn the status of the patient at each timestamp. In the final layer, the next event's diagnosis codes and time interval until the next visit are predicted. Although this model deals with events with irregular intervals, it does not explicitly leverage information from the intervals other than to predict the next interval size. In contrast, Baytas et al. [18] develop a similar RNN model which directly utilises the time interval inputs for diagnosis prediction.

5.2. Time decay factors

In addition to the use of time intervals as direct inputs into a model, they are also frequently used to generate time decay factors, which are used to modify conventional gated RNN architectures. Time decay factors reflect the common assumption that previous observations with a long time interval from the current observation, will have less influence on the outcome of the model. Although this broad assumption will not always be true, with the addition of learnt weights on the decay, the mechanism can be moderated and effectively implemented for irregular time series data. As reflected in the following models, negative exponential functions are frequently used to implement time decay, by applying an exponent which incorporate the time interval with associated weight and bias parameters.

Che et al. [17] investigate the application of Gated Recurrent Units (GRU) with trainable decays on multivariate time series with missing data. The research develops a deep learning model, called GRU-D, based on the GRU with an added decay mechanism. It takes two representations of missing patterns, i.e., masking and time interval, and includes them into the model architecture. Masking advises the model which inputs are observed (or missing), while time intervals capture the input observation patterns. The time interval vector is used in an exponential decay calculation with trained parameters. The decay value along with the masking array is used to generate a derived input which imputes values that are close to the last real observation for short time intervals and which decays over time towards the empirical mean. The GRU-D also modifies the conventional GRU model by feeding the masking array directly into gate and hidden state equations. A second trained exponential decay value further modifies the GRU model by decaying the GRU's hidden state to capture richer information on missingness patterns. The paper's experimental results show that the GRU-D utilise the missing patterns when the correlations between labels and missing rates are high and relies on the observed values when the rates are low. The GRU-D model is a case in which both the imputation layer and the prediction layer have considerable influence on the handling of missing data. The imputation method does, however, use a missing value estimation formula which decays towards an empirical mean, like several other proposed models [67,68], which assumes a stationary Markov process and therefore limits the consideration given to the temporal dynamics of the time series.

Tian et al. [9] develop an LSTM based model for traffic flow prediction, which handles missing data and captures the long and short term temporal dependencies of time series observations. The paper considers two general categories of missing data, short period missing values (i.e. 1 s to 5 min) and long period missing values (typically ranging from hours to days). The proposed model utilises a masking vector to represent missing values, as well as a time interval vector used to generate an exponential decay function. The masking vector is directly input into the LSTM equations for each of the three gates as well as the candidate cell state input. The model is closely based on Che et al. [17], with minor modifications which may cater to the specific traffic flow application domain. Other than its use of the LSTM, it differs from the GRU-D [17] model in its derived input, which decays towards past observations rather than the variable's mean value.

Li et al. [7] is another model which is heavily influenced by the GRU-D model and in this case, introduces a missing rate factor to modify the conventional GRU model. The proposed Variable Sensitive GRU (VS-GRU) considers the impact of the missing rate of different variables in multivariate time series. Missing value indicator, time interval and missing rate vectors are inputs to the model. The conventional GRU is modified by augmenting the input data with a weighted missing indicator vector, as well as a weighted missing rate factor, into the GRU gate and hidden state equations. The weights in these equations are used by the model to learn information on the actual observations, rather than the imputed observations. The VS-GRU also adopts the GRU-D's trainable decays for dynamic imputation and decay of the hidden state. The output of the GRU layer goes to a fully connected layer for classification purposes. Although VS-GRU introduces a missing rate factor on top of the characteristics of the GRU-D, the resulting improvement in performance is modest.

5.3. RNN with Ordinary differential equations (ODE)

Neural ODEs [83] are a family of continuous-time models which parameterize the derivative of the hidden state using deep neural networks. In these models, the hidden state is defined as the solution to an ODE initial-value problem, where the hidden state can be calculated at any time (t) using a numerical ODE solver. Unlike traditional RNN based models, these models develop a continuous-time function and are not bound to discrete-time sequences. RNN with ODEs uses the update function of a gated RNN model and ODE differential equation solver to produce RNN models as a function of continuous-time. The resulting RNN models are capable of learning the dynamics between observations and therefore, naturally handle arbitrary time gaps prevalent in irregular and sparse data.

Rubanova et al. [84] develop the ODE-RNN model in which the hidden state between observations is defined as the solution to an ODE. For each observation, the model updates the hidden state using a standard gated RNN (GRU) hidden state update. The authors use neural ODEs to define two continuous-time models, an autoregressive ODE-RNN and a variational autoencoder based Latent ODE model. The Latent-ODE model employs a sequence-to-sequence architecture in which variable lengths sequences are encoded into a fixed dimensional embedding using the ODE-RNN and then decoded into a variable-length sequence. The experimental results included testing against the Physionet clinical ICU database with missing values, achieving similar AUC classification results as the GRU-D model. Testing with a human activity dataset, further demonstrated the Latent ODE model's superior classification accuracy against RNN models including the GRU, GRU-D and Variational Autoencoder RNN. It was noted that because ODE models are required to be continuously solved, independent of the availability of data points, they were found to take 60% more time than the standard GRU to evaluate. Based on the work in [84], Habib et al. [85] aim to address the issue of time complexity as well as to develop a simpler architecture which removes the requirement for an encoder-decoder network. The authors redesign traditional LSTM and GRU models to compute the hidden and cell states at any point in time, while significantly reducing the associated computational overhead. The proposed models, ODE-GRU and ODE-LSTM, train gradients themselves with minimal error using an ODE solver, where all the parameters are learnt during training. The paper demonstrated that these ODE models can address the issues associated with time series data that include irregular sampling rates. The paper also presented an evaluation of model performance against sparse and irregular datasets, including a human activity dataset in which the proposed models are compared with GRU, LSTM and the Latent-ODE model. The authors concluded that the proposed models demonstrate faster training and evaluations times with a simpler architecture without adversely affecting accuracy, however, their conclusions lack comprehensive numerical results to illustrate these points.

5.4. Gated RNN modified structure

For gated RNN variants, different gates and inner connections allow RNNs to take on various internal architectures that achieve different capabilities and efficiencies. Numerous RNN cell variants have been introduced in recent years which have achieved improved performance in terms of speed and accuracy for specific datasets or domains. Zhou et al. [86] reduce the GRU model to a single gate in their Minimal Gated Unit (MG) for faster training time, Nugaliyadde et al. [87] modify the LSTM for handling longer sequences and Nina et al. [88] simplify the LSTM cell by coupling the forget gate and input gate for tasks not requiring recall of very long sequences. Although these models do not attempt to address the issues of irregular time series data, they illustrate the flexibility of the RNN architecture to be modified to address specific dataset issues. In the current literature, only a very small number of modified gated RNN architectures have been proposed to address irregular times series datasets. The following models make structural changes to the base LSTM or GRU models, which go beyond input augmentation and the decay of inputs or hidden states. As shown by the representations of the T-LSTM and Phased-LSTM in Fig. 5, these models may result from the removal or addition of gates or implementing additional functions to the inputs, outputs or existing gates, which have the aim of better capturing the irregularity of the data in terms of missing values and multiple sampling frequencies.

Pham et al. [89] implement a gated RNN with modification of the forget gate by decaying its output and augmenting its inputs with time intervals. The model is used to generate predictions from observations within electronic health records which are irregular in time. In contrast to other RNN based clinical models, a vector representing medical interventions is also augmented with the diagnostic inputs to improve disease progression modelling. The model is built on an LSTM and introduces time parameterisations to handle irregularly timed events by moderating the "forget" mechanism. The study introduces two mechanisms for modifying the forget gate, first by applying a monotonically decreasing time decay on the forget gate and secondly by augmenting the inputs into the forget gate with a time interval vector which is parametrically time-weighted. The overall model has the LSTM outputs aggregated through a time decayed multiscale pooling strategy and at the final layer, "pooled illness" states are passed through an FCN for estimating a future prognosis. The proposed "DeepCare" model effectively considers the decaying influence of time intervals; however, it is limited by its focus on admission interval irregularity rather than feature input irregularity.

Baytas et al. [18] further develop irregular time series modelling, following in a similar line of research to [89], through a proposed time-aware LSTM (T-LSTM) model which modifies the LSTM memory cell rather than the forget gate, as shown in Fig. 5a. The authors develop a modified LSTM architecture that takes the elapsed time into consideration between consecutive elements of a sequence to adjust the memory content of the unit. T-LSTM takes two inputs, the input vector and the elapsed time at the current time step. Elapsed time is transformed into a weight using a time decay function. The memory cell is adjusted such that the longer the elapsed time, the smaller the effect of the previous memory on the current output. This approach does not alter the effect of the current input to the current output but does change the effect of the previous memory on the current output. Similarly to [89], the T-LSTM specifically models input medical diagnosis and proce-

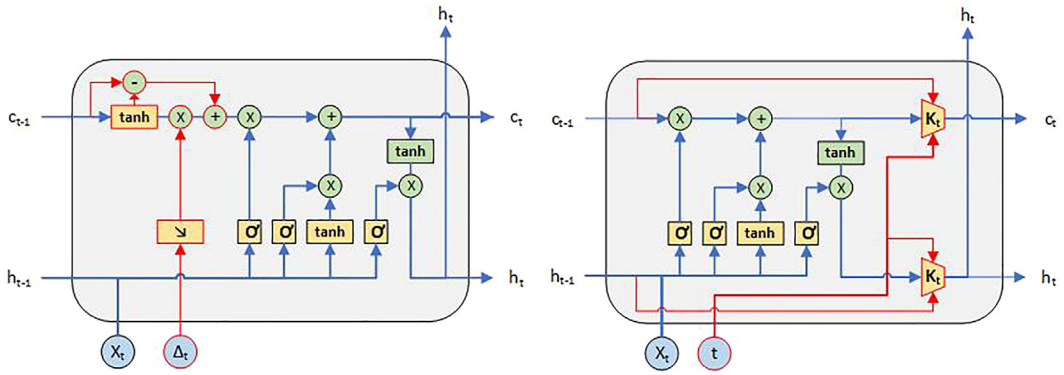

Fig. 5. Gated RNN modified structure examples: (a) T-LSTM [25] (b) Phased-LSTM [5].

dural codes but has not been applied to missing data with real-valued input variables and therefore has limited transferability to different application domains.

In comparison to [18,89], Neil et al. [5] provide the most comprehensive structural change to a conventional gated RNN model for handing asynchronous data. As shown in Fig. 5b, the Phased LSTM model includes two new time gates that are of an oscillatory nature, which allows the update of the memory cell and hidden state during only a fraction of their cycles. The key applications of the model are for handling asynchronous input data with irregular sampling rates and learning long sequences at much faster rates than regular LSTMs. In terms of inputs into the model, it differs from the standard LSTM by requiring the input of event times. The opening and closing of a new time gate are determined by an independent periodic oscillation function defined by three learned parameters. A significant advantage of the Phased LSTM is that updates can be performed at irregularly sampled time points, as required for event-driven asynchronous data. The model's oscillatory nature and sparse updates also enable faster learning convergence. The Phased LSTM has subsequently been applied in several research papers, including [69], which demonstrated the effectiveness of the model on long, sparse time series. For shorter time series, the expected benefit of this model is limited and may result in reduced accuracy from sparse updates for short sequences. The model also inputs timestamps as opposed to time intervals which may implicitly capture the time interval information between observations but may not capture the relationships between observations and events, as well as explicit modelling of the time intervals as per Pham et al. [89].

Vecoven et al. [90] present an innovative recent development to the GRU with the Bistable Recurrent Cell (BRC), which shows significant potential for long, sparse time series which contain noise or missing data. The model is inspired by biological neuron bistability, which embed RNNs with long-lasting memory at the cell level. This allows the bistable neurons to switch between two stable states in response to transient inputs and enables a cellular memory which does not fade with time. The primary design modification to the GRU is in the feedback structure, in which a new feedback parameter modifies the reset gate and subsequent hidden state, which allows the cell to switch between monostability and bistability. The model includes a network structure, where the outputs of other neurons in the layer build a recurrent layer-wise neuromodulation, similar to the GRU. The BRC is tested against three time series and performs regression on the first two and classification on the third, which is an MNIST sequence. Results showed that the Bistable Cell model outperformed the LSTM and GRU for long time series sequence lengths, greater sparsity and noise. Similar to the Phased LSTM, the BRC model has limited benefits for short sequences; however, its ability to handle sparsity and noise provides the potential for application in irregular time series problems.

6. Evaluation and datasets

For modelling time series data with missing values, two important considerations are the method of evaluation and the type of dataset being modelled, which will be reviewed in this section. In addition to selecting the appropriate metrics for model evaluation, the performance of imputation based missing value estimation involves the selection of different methods of simulating missing values which can significantly impact on model performance. In section 6.1, we summarise the evaluation metrics used by the papers within this review and how they vary depending on whether the model is imputation focused or prediction focused. Datasets and the domain which they belong to are also a critical factor when modelling irregular time series data, as data from different domains are often characterised by different types of underlying features and missing data patterns. As will be shown, the current body of research work is skewed towards the medical domain, and therefore some of the models reviewed may be more effective on data prevalent in this domain more than others. Section 6.2 describes the datasets used by the irregular times series models, which gives a clear picture of the domains which have been the focus of studies within this field.

6.1. Evaluation metrics

Time series models for handling missing data are generally evaluated in terms of the prediction task they perform or in terms of the accuracy of the imputation process performed. In the case of imputation focused models, the evaluation metrics are regression-based, which estimate the error between the estimated value for imputation and the ground truth value. As can be seen in Table 2, imputation performance is most commonly evaluated using error-based statistics such as Mean Absolute Error (MAE), Mean Squared Error (MSE) or root MSE (RMSE) and Mean Relative Error (MRE). The Coefficient of Determination (R^2) has also been used to describe imputation accuracy by calculating the correlation of the actual values with the imputed values. When dealing with datasets with missing values, the problem occurs that direct evaluation of missing value imputation accuracy is impossible as there are no ground truth values; therefore, the solution has been to randomly discard a percentage of true observations and calculate the imputation accuracy on this set of discarded values, which represent the ground truth. It should be noted that the problem is not

Table 2
Evaluation metrics used by gated RNN models for irregular time-series data handling.

Forecasting Type	Metric	Referenced Papers
Imputation Accuracy – Regression	MAE	[66,8,77,79]
	MSE, RMSE	[62,66,71,77]
	MRE	[8,79]
	MAPE, SMAPE	[77]
Predictive Model – Classification	Accuracy	[63,68,8,78,5,90]
	AUC	[63,68,62,8,75,26,17,7,18,84]
	F1-score	[68,76,78,26,89]
	Recall	[68,76,82]
	Precision	[68,76,26,89]
Predictive Model – Regression	MAE	[63,69,9]
	MSE, RMSE	[9,18,84,90]
	MRE	[9]
	R^2	[82]

always easily addressed by introducing additional missingness because the cause of missing values is not always completely random and therefore a better understanding of their cause is required to provide an appropriate mechanism for generating missing values in a dataset. In some cases, direct imputation accuracy evaluation is circumvented, and imputation performance is assessed by the overall predictive task of the model, such as time series forecasting values or classification accuracy.

For predictive models that are not imputation focused, the evaluation metrics for irregular time series data are the same as metrics used for regular time series forecasting tasks. For regression tasks, MAE, MSE (or RMSE) and MRE are most frequently used for evaluating the performance of the model. Mean Absolute Percentage Error (MAPE) and Symmetrical MAPE (SMAPE) are also used in some instances of regression evaluation. RMSE and MAE are scale-dependent metrics, MRE has been scaled, and MAPE and SMAPE are percentage-error metrics. The MSE metric is highly sensitive to outliers, while the impact of outliers is reduced by MAE and further reduced by MRE. MAE and MRE cannot, however, indicate the bias of predictions in terms of over predicting or underpredicting. MAPE can differentiate this type of bias and places a penalty on predicted values above the actual values, while SMAPE alternatively penalises predicted values lower than actual values [91].

For predictive models with classification tasks, area under the ROC curve (AUC) was the most common metric used within the papers reviewed. Other common classification metrics included Accuracy, F1-score, Recall and Precision. AUC was frequently used in health care datasets due to their highly unbalanced nature, while the Accuracy metric was adequate for performance evaluation of balanced datasets. For multiclass classification, micro-F1 and macro-F1 scores were used in some cases, as per Lipton et al. [26]. From these summary characteristics of metrics for time series imputation accuracy or prediction model performance evaluation, it can be seen that appropriate metrics are highly dependent on the dataset and the required prediction outcome.

6.2. Data sets

Irregular time series data is generated in a multitude of domains, including medical, environmental, industrial and financial. Table 3 provides a listing of the external datasets used within the reviewed papers and their characteristics, including domain and missing data rate. From the missing data field, we can see that very high missing rates often characterise medical data. In contrast, datasets from other domains, including environmental and human activity recognition (HAR) have lower average missing rates. In some of the papers reviewed, missing rates were not stated although irregular time series datasets were used, which may occur where there are no apparent missing values for variables at any given timestamp, but the timestamps come in at irregular intervals. It can also be seen from Table 3 that the majority of research papers on this topic have focused on medical data and particularly electronic health records (EHR). The focus on EHR, within the field of irregular time series data modelling, has been due to the prevalence of irregular data within this domain and the importance attributed to predictive modelling of patient outcomes. Which in turn, has led to the availability of several high-quality publicly available time series medical datasets, of which Physionet [92] and MIMIC-III [93] are two of the current benchmarks. Harutyunyan [94] proposes a comprehensive clinical prediction benchmark based on the MIMIC-III database. Environmental monitoring data is one of the next most frequently encountered application types, of which the Beijing PM2.5 air quality dataset [95] has become a benchmark dataset for modelling irregular time series data. The absence of a broader set of domains and associated benchmark datasets is apparent in the field of irregular time series modelling, which has made the comparative performance of machine learning techniques difficult to measure.

6.3. Results comparison

As stated in the preceding section, the lack of consistent performance benchmark datasets and task metrics has made an overall results comparison difficult for the set of reviewed models. It must also be noted that reference to a common dataset name across different papers does not necessarily provide for consistent experimental data as different subsets of data or versions of the dataset, along with different experimental conditions may have been used. However, application of a common dataset does provide the opportunity for some level of comparison of numerical results and several common datasets have been used by some of the reviewed models. The Beijing air quality and meteorological data set has been used in several cases for testing imputation using MAE metrics, with the BRITS [8], seq2seqImp [79] and Iterative Imputing Network (IIN) [69] models attaining the following respective results 11:56, 11.13, 10.63. The bidirectional model with feature correlation (BRITS) shows marginally better results than the sequence to sequence model with bidirectional decoder [79].

The MIMIC-III database was used by several reviewed models for the prediction of missing values for imputation or diagnosis classification of the time series. Internally presented RMSE results within [71] show that the M-RNN outperformed the models Doctor-AI [82] (0.0337), MI [26] 0.0295 and GRU-D [17] 0.0292, with an RMSE of 0.0137 that represented more than a 50% gain in performance against each model. Two of the primary differences between the M-RNN network and the compared models was the M-RNN's use of bidirectional RNNs and inter-feature correlation. MIMIC-III diagnosis performance was evaluated using the AUC metric by GRU-D 0.8527, which was marginally outperformed by results from VS-GRU [7] 0.8576. Indicating very similar results from the GRU-D and VS-GRU, as would be expected from two similar models, with the exception of the missing rate feature in the VS-GRU.

The Physionet ICU dataset was the most widely used experimental dataset which was used for tasks including mortality prediction, measured by the AUC metric. The following models presented AUC values: BiLSTM with Attention [75] 0.839, GRU-D 0.8424, BRITS 0:850., VS-GRU 0.8502, GAN [62] 0.8603. The results show that VS-GRU, which was heavily based on the benchmark GRU-D model, provides a marginal improvement of the AUC through the addition of a missing rate factor, while the BRITS model also provides improvement on the GRU-D, possibly through its Bidirectional capability and additional consideration of correla-

Table 3
List of external datasets used for irregular time series data (*Simulated or manually set missing value rate. NA = Not Available).

Model	Domain	External Datasets Used	Uni/Multivariate	Missing Data (%)
RNN, Nguyen et al. [63]	Medical	• Alzheimer's Disease Neuroimaging Initiative (ADNI) database	Multivariate	30–80
RIMP-LSTM, Shen et al. [66]	Varied	• Daily Births, Quebec (1) • Electricity Consumption-MT124 (2) • DSIM - Simulated Diabetes (3) • SCITOS G5 - Robot sensors (4) • Freeway Traffic Volume, China (5)	Univariate (1,2,5), Multivariate (3,4)	5–50 *
Temporal Belief Network, Kim et al. [68]	Medical	• EHR system at Christiana Care Health System data	Multivariate	81.4
Iterative Imputing Network (IIN), Zhou et al. [69]	Environment	• Beijing air quality and meteorological data (PM2.5 air quality dataset)	Multivariate	20–65 *
GAN, Luo et al. [62]	Medical, Environment	• 2012 PhysioNet - ICU (1) • KDD CUP 2018 public air quality (2)	Multivariate	80.67 (1), 20–90 (2) *
Multidirectional-RNN (M-RNN), Yoon et al. [71]	Medical	• Medical Information Mart for Intensive Care (MIMIC-III)	Multivariate	96
BRITS, Cao et al. [8]	Environment, Medical, HAR	• PM2.5 Beijing Air Quality (1) • PhysioNet Challenge 2012 (2) • UCI localisation data for Human Activity (3)	Multivariate (1,2) Univariate (3)	13.3 (1) 78 (2) 10 (3)
Flexible Irregular Time Series (FIT), Singh et al. [76]	HAR, Medical	• Inertial Sensor Dataset by Osaka University -OU-ISIR (1) • Human Motion Primitives(HMP) Detection (2) • 2012 PhysioNet - ICU (3)	Multivariate	60 (1) * 60 (2) * 84 (3)
BiLSTM with Attention, Nguyen et al. [75]	Medical	• 2012 PhysioNet - ICU	Multivariate	82
S2SwA, Tang et al. [78]	Varied, HAR	• 12 UCR datasets (1) • Opportunity Challenge Human Activity (2)	Multivariate (2) Univariate (1,2)	NA (1) 30 (2) *
SSIM, Zhang et al. [77]	Environment	• Great Barrier Reef Catchment Loads Monitoring Program data	Multivariate	NA
Sequence-to-Sequence, Dabrowski et al. [79]	Traffic, Environment	• UCI Metro Interstate Traffic Volume (1) • UCI Birmingham Parking dataset • UCI PM2.5 Beijing Air Quality	Multivariate	NA (1,2) 13.3 (3)
Missing Indicators (MI), Lipton et al. [26]	Medical	• Paediatric intensive care unit (PICU) at Children's Hospital Los Angeles	Multivariate	15.7–93.8
Doctor AI, Choi et al. [82]	Medical	• Palo Alto Medical Foundation - Heart Failure EHR dataset (1) • MIMIC-III (2)	Multivariate	96 (2)
GRU-D, Che et al. [17]	Medical	• UCI Gesture phase segmentation dataset (1) • 2012 PhysioNet – ICU (2) • MIMIC-III (3)	Multivariate	NA (1) 82.2 (2) 96.2 (3)
LSTM-M, Tian et al. [9]	Traffic	• Cal-trans Traffic Performance Measurement System (PeMS) (1) • Locally recorded traffic dataset (2)	Multivariate	NA (1) 30 (2)
Variable Sensitive GRU (VS-GRU), Li et al. [7]	Medical	• 2012 PhysioNet – ICU (1) • MIMIC-III (2)	Multivariate	81.93 (1) 95.6 (2)
ODE-RNN, Rubanova et al. [84]	Medical, HAR	• 2012 PhysioNet – ICU (1) • UCI localisation data for Human Activity (2)	Multivariate	81.93 (1) NA (2)
Deep-Care, Pham et al. [89]	Medical	• Australian Regional Hospital dataset - Diabetes and Mental Health patient Data	Multivariate	NA
Time Aware-LSTM (T-LSTM), Baytas et al. [18]	Medical	• Parkinson's Progression Markers Initiative (PPMI)	Multivariate	NA
Phased LSTM, Neil et al. [5]	Visual and Audio Recognition	• N-MNIST dataset for neuromorphic vision. • GRID multi-sensor dataset	Multivariate	NA
Bistable Recurrent Cell, Vecoven et al. [90]	Visual	• M-NIST Sequential images	Multivariate	NA

tions between feature variables. However, the GAN model provides the highest AUC performance, indicating the strength of GAN based imputation.

The ODE-RNN model [84] was internally compared against the GRU-D in regression tasks and classification tasks on the Physionet dataset, achieving improved results of MSE 2.361, AUC 0.833 against the GRU-D MSE 3.384, AUC 0.818. For external comparison of AUC results from other papers using the Physionet dataset, the ODE-RNN AUC was generally lower than the other reviewed models. However, as noted above, without confirmation of the same conditions for experimental setup, data selection and preprocessing methods, direct comparison of model performance between papers is difficult.

The gated RNN modified structure models Deepcare [89], T-LSTM [18], Phased LSTM [5] and Bistable Recurrent Cell (BRC) model [90], did not utilise any common datasets to enable comparative results against other models reviewed but provided internal comparisons against conventional machine learning models. Deepcare used a Diabetes and Mental Health patient dataset to predict unplanned readmissions and achieved a superior F-score (79) in comparison to base methods SVM (66.7), Random Forests (71.4), Plain RNN (75.1) and LSTM (75.9). The phased LSTM model included comparative results against a number of sequence datasets, including the N-MNIST dataset for neuromorphic vision, which showed improved accuracy (97.27) against a CNN model (94.99) and batch-normalized LSTM model (96.55). The BRC model showed improved performance accuracy (0.9760) against the LSTM (0.1124) and GRU (0.1081) when the sequence length was appended with long blank periods. In order to provide relative performance measures of these types of modified gated RNN struc-

tures in handling irregular time series data, future work would require their application against common benchmark datasets such as Physionet 2012 or Beijing PM2.5 Air Quality.

6.4. Computational complexity

In the absence of reference to computation complexity in all but very few of the papers reviewed, a general comparison of the models with respect to each other and standard gated RNN models can be considered. This shall take into account the broad types of complexity, such as linear and quadratic complexity, associated with the structure of the model's dominant components and disregarding model and data characteristics such as regularization, dropout, hidden cell sizes or sample lengths.

The LSTM is known to be local in space and time [96,97], meaning that the input sequence length does not impact on the storage requirements of the network, and its time complexity per time step and weight is O(1) constant. Hence the overall complexity of an LSTM per time step is equal to O(w), where w is the number of weights. Similarly, for other RNN variants, the time complexity is O(w). Given that w is dependent on the number of memory cells, number of input units, number of output units and layers, we can generally compare the computational complexity of the reviewed gated RNN models, taking into account the number of recurrent layers, bidirectional properties and other mechanisms that will add to the number of parameters.

With regard to attention, this mechanism adds to the computational complexity of a model and requires more computation when a deep network wants to retrieve relevant information over a longer period of past information, or from a longer sequence of data [98]. An efficient attention mechanism will generally have linear computational complexity O(n). In the case of some neural network models utilising Self-Attention, the computational complexity of the attention mechanism is proportional to the square of the search targets, resulting in $O(n^2)$ complexity.

Of the imputation models considered, RNN filling [63] and TBM [68] use a relatively standard gated RNN structure, while the Iterative Imputation Network [69] adds complexity with its forward and backward networks and multiple iterative layers. The RIMP-LSTM [66] with its residual connections does not increase the computational complexity of LSTM networks. The M-RNN [71] and BRITS [8] models increase their complexity through the application of bidirectional RNNs, while the FIT [76] and BI-LSTM with Attention [75] further increase their complexity by applying bidirectional RNNs and Attention. Sequence-to-Sequence networks increase complexity by virtue of involving separate encoder and decoder networks, as per the SSIM [77] which combines the additional complexity of bi-direction, sequence-to-sequence and Attention.

Within the predictive models, augmentation of input values with missing indicators [26] or time intervals [82] increases complexity from standard gated RNN models to a small degree due to larger weight matrices associated with a larger number of input features. In addition to input augmentation, the GRU-D [17], LSTM-M [9] and VS-GRU [7] increase the complexity of standard RNN models by including a new decay rate with learnable weights into the hidden state equation. Latent ODEs [84] is one of very few models which comment on computational complexity, and the authors acknowledge that although the complexity is similar to RNN models, the ODE is continuously solved even when no observations occur which adds to computational cost, as reflected in experiments where this model took twice the amount of time as a standard GRU.

In the modified structure gated RNN models, each of the modifications introduce new parameters which marginally increase the complexity of the standard LSTM or GRU model. The T-LSTM's [18] memory cell modification with time decay includes additional parameters and the Phased LSTM's [5] two additional time gates and associated parameters, both result in increased computational complexity over the standard LSTM. The Bistable Recurrent Cell [90] similarly introduces new parameters with its new cellular feedback mechanism for enabling bi-stability and raises the computational complexity over the standard GRU.

7. Conclusion

This study presents a systematic review of a selection of current leading methods for modelling irregular time series data, specifically focusing on the application of gated RNNs. We have explained how models dealing with irregular data can be separated into two broad categories, those which focus on developing a complete time series prior to prediction and those which focus on dealing with irregularity as part of the prediction layer. On presentation of the different models, we have provided detailed descriptions of their methodology and highlighted a number of key techniques used for enhancing accuracy. The resulting RNN based forecasting methods have been found to provide superior performance against many statistical techniques as well as other machine learning techniques and therefore play an important role in the development of more efficient and accurate models that can cope with the ever-increasing levels of irregular data.

In addressing the challenge of building better models to handle irregular time series data, RNNs are a critical stream of development due to their inherent capability to extract patterns from univariate and multivariate temporal sequences and retain memory. In the field of deep learning for sequence modelling, RNNs have proven to be one of the foremost techniques in many applications, including NLP and time series. Even in comparison to state-of-the-art models such as Transformers, evolving gated RNN architectures still maintain certain advantages for time series applications. The models reviewed in this paper, therefore represent an important view of the current developments of a leading technique in addressing the problems associated with modelling irregular time series data.

Within the class of models focussing on imputation and data generation, the reviewed papers [75,76] indicate that the use of bidirectional RNNs with Attention can lead to higher accuracy in missing value estimation, in comparison with unidirectional RNNs or RNNs without Attention. This is achieved by exploiting the long-range context dependencies of past as well as future time steps and also identifying which parts of the sequence to focus on. Several models have also used bidirectional RNNs with attention within a Sequence-to-Sequence (Seq2Seq) structure for accurate imputation of irregular time series data. We have also seen that making use of missing indicators and decay functions based on weighted time intervals can be used to optimise estimated imputation values. Recurrent GAN networks are a promising and underutilised technique in this category, as reflected by its application in [62] which demonstrated its effective use for modelling irregular time series datasets. This review also identified that predictive model focused methods had been less widely applied than imputation focused methods; however, forecasting results presented within the associated papers have shown that these models can directly address the issue of irregular time series data. These predictive models have clearly shown the benefit of direct augmentation of inputs with missing indicators, which has been further enhanced in some cases by using time intervals for decay factors on inputs, hidden states or gate outputs. Systematic modification of a gated RNN network architecture to handle irregular data has been applied by very few proposed models but has proved to be an extremely efficient method of tackling this problem, as reflected

by the success of the Phased LSTM, which handles long asynchronous sequences exceedingly well. The Bistable Recurrent Cell (BRC) network is another recent modified gated RNN network which shows potential to handle sparse, irregular data in long time series with its new bistable cell structure.

A summary of the key techniques used by gated RNN models for dealing with irregular time series data, as presented in the reviewed papers, are shown in Table 4. The column headings reflect the technique categories introduced in section 3, with the additional inclusion of two methods, Missing Rate and Inter-Feature Correlation (Inter-Feature), which did not appear in a specific section category within this paper but were however important methods to individually highlight in the summary table. From the analysis of the results comparison and key techniques in Table 4, some general techniques can be identified for the best performing models. Based on comparable results on a common dataset, for regression tasks on the MIMIC-III database, the M-RNN showed superior performance which can be attributed to its use of bidirectionality in its gated RNN and also inter-feature correlation. Classification tasks on the Physionet database showed the GRU-D, BRITS and VS-GRU had similarly impressive AUC results between 0.84 and 0.85. As indicated in Table 4, all three models incorporated missing indicators and time decay functions into the gated RNN models, while the VS-GRU also incorporates a missing rate feature. The ODE model showed promising results for classification on the Physionet dataset with internal results indicating outperformance of the benchmark GRU-D model. However, the recurrent GAN model produced the state-of-the-art mortality prediction results for this common dataset. In many of the modified gated RNN structure models shown in the final rows of Table 4, there were insufficient comparative results on common datasets or against other leading irregular time series models to benchmark their performance, however, characteristics of their novel architectures show much potential for handling sparse and irregular datasets.

Despite the recent achievements of gated recurrent neural networks for irregular time series modelling applications, there still exist a number of outstanding challenges within this field, which are summarised as follows:

1. The overall level of research into machine learning for handling irregular time series is very low given the abundance of this data in the real world, so there is a need for more research to further exploit the capabilities of leading methods such as gated RNNs to model this type of data. In particular, modified RNN architecture models have rarely been proposed in the literature but represent a very elegant and promising solution to this problem.
2. Inter-feature correlations provide richer information on temporal relationships when combined with relationships within each variable. Yet few models explicitly utilise this technique, as reflected by the limited number of cases shown in summary Table 4.
3. Random missing value generation for imputation accuracy evaluation often fails to consider the nature of the missing data, which can contain important information in the patterns of missingness.

Table 4
Common techniques for the application of gated RNNs in handling irregular time series data.

Model	Uni-RNN	Bi-RNN	Attention	Seq-2-Seq	GAN	Missing Indicator	Time Interval	Time Decay	Inter-Feature	Missing Rate	ODE	Modified Structure
Imputation & Data Generation Focused Methods												
RNN, Nguyen et al. [63]	X											
RIMP-LSTM, Shen et al. [66]	X											
Temporal Belief Network, Kim et al. [68]	X					X		X				
Iterative Imputing Network (IIN), Zhou et al. [69]	X											
GAN, Luo et al. [62]	X				X	X		X				
Multidirectional-RNN (M-RNN), Yoon et al. [71]		X				X	X		X			
BRITS, Cao et al. [8]		X				X		X	X			
BI-LSTM with Attention, Nguyen et al. [75]		X	X									
Flexible Irregular Time (FIT), Singh et al. [76]		X	X			X			X			
S2SwA, Tang et al. [78]	X			X								
SSIM, Zhang et al. [77]		X	X	X								
Sequence-to-Sequence, Dabrowski et al. [79]		X		X								
Predictive Model Focused Methods												
Missing Indicators (MI), Lipton et al. [26]	X					X						
Doctor AI, Choi et al. [82]	X						X					
GRU-D, Che et al. [17]	X					X		X				
LSTM-M, Tian et al. [9]	X					X		X				
Variable Sensitive GRU(VS-GRU), Li et al. [7]	X					X		X		X		
ODE-RNN, Rubanova et al. [84]	X										X	
Deep-Care, Pham et al. [89]	X						X	X				X
Time Aware-LSTM (T-LSTM), Baytas et al. [18]	X							X				X
Phased LSTM, Neil et al. [5]	X											X
Bistable Recurrent Cell, Vecoven et al. [90]	X											X

4. Current research is highly skewed towards medical data, which has helped develop novel techniques in this field but there is a need for more applications and benchmark datasets in domains such as environmental monitoring, industrial systems and financial forecasting.
5. Transferability of models for different types of data has had a limited focus in the existing literature. Characteristics of the time series need to be better understood and measured in terms of sequence length, volatility and underlying process, as well as the characteristics of missing data such as sparsity, randomness and average missing interval size.

Based on the existing challenges it is believed that further empirical studies are required, where leading gated RNN models and associated techniques for handling irregular data are experimented against different types of time series datasets, in terms of the underlying time series signal and the type of missing data. This should be performed with simulated and real-world data from different domains. By understanding the characteristics of the time series data and its irregularities, it is also believed that more effective models can be built with modified gated architectures. Design of novel customised structures, gate activation functions and memory retention functions, based on the nature of the time series data will lead to more efficient recurrent units which can directly handle different types of irregularity.

In conclusion, it is hoped that this review provides researchers with a view of state-of-the-art methods provided by gated RNNs, which may be individually or collectively applied to handle irregular time series data. Consideration of the techniques used, relative advantages, as well as existing challenges will provide a platform for further research on this increasingly important topic.

CRediT authorship contribution statement

Philip B. Weerakody: Conceptualization, Investigation, Writing - original draft, Writing - review & editing, Visualization. **Kok Wai Wong:** Writing - review & editing. **Guanjin Wang:** Writing - review & editing. **Wendell Ela:** Supervision.

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 paper.

References

[1] W. Bao, J. Yue, Y. Rao, A deep learning framework for financial time series using stacked autoencoders and long-short term memory, PLoS ONE 12 (7) (2017) e0180944, https://doi.org/10.1371/journal.pone.0180944.

[2] T. Lin, T. Guo, K. Aberer, Hybrid neural networks for learning the trend in time series, in: IJCAI International Joint Conference on Artificial Intelligence, 2017, pp. 2273–2279. 10.24963/ijcai.2017/316.

[3] M. Lepot, J.-B. Aubin, F. Clemens, Interpolation in time series: an introductive overview of existing methods, their performance criteria and uncertainty assessment, Water 9 (2017) 796, https://doi.org/10.3390/w9100796.

[4] C. Xiao, E. Choi, J. Sun, Opportunities and challenges in developing deep learning models using electronic health records data: a systematic review, J. Am. Med. Inform. Assoc. 25 (10) (2018) 1419–1428, https://doi.org/10.1093/jamia/ocy068.

[5] D. Neil, M. Pfeiffer, S.-C. Liu, Phased LSTM: accelerating recurrent network training for long or event-based sequences, in: Neural Information and Processing Systems (NIPS), 2016, pp. 3889–3897. http://papers.nips.cc/paper/by-source-2016-1928.

[6] R. Little, D. Rubin, Statistical Analysis with Missing Data, 2nd ed., Wiley, 2014, pp. 200–220. 10.1002/9781119013563.ch10.

[7] Q. Li, Y. Xu, VS-GRU: a variable sensitive gated recurrent neural network for multivariate time series with massive missing values, Appl. Sci. 9 (2019) 3041, https://doi.org/10.3390/app9153041.

[8] W. Cao, D. Wang, J. Li, H. Zhou, L. Li, Y. Li, BRITS, Bidirectional recurrent imputation for time series, in: NIPS'18: Proceedings of the 32nd International Conference on Neural Information Processing Systems, 2018, pp. 6776–6786.

[9] Y. Tian, K. Zhang, J. Li, X. Lin, B. Yang, LSTM-based traffic flow prediction with missing data, Neurocomputing 318 (2018) 297–305, https://doi.org/10.1016/j.neucom.2018.08.067.

[10] E.R. Ziegel, G. Box, G. Jenkins, G. Reinsel, Time series analysis, forecasting, and control, Technometrics 37 (1995) 238, https://doi.org/10.2307/1269640.

[11] D.J.C. MacKay, Bayesian interpolation, Neural Comput. 4 (1992) 415–447, https://doi.org/10.1162/neco.1992.4.3.415.

[12] S.J. Roberts, M.A. Osborne, M. Ebden, S. Reece, N. Gibson, S. Aigrain, Gaussian processes for timeseries modelling, Philos. Trans. Royal Soc. A: Math. Phys. Eng. Sci. 371 (2013) 20110550–20110550. 10.1098/rsta.2011.0550.

[13] V. Vapnik, S.E. Golowich, A.J. Smola, Support vector method for function approximation, regression estimation and signal processing, in: Advances in Neural Information Processing Systems 9, MIT Press, 1997, pp. 281–287. http://papers.nips.cc/paper/1187-support-vector-method-for-function-approximation-regression-estimation-and-signal-processing.pdf.

[14] N. Ahmed, A. Atiya, N. Gayar, H. El-Shishiny, An empirical comparison of machine learning models for time series forecasting, Econometr. Rev. 29 (2010) 594–621, https://doi.org/10.1080/07474938.2010.481556.

[15] E. Choi, A. Schuetz, W. Stewart, J. Sun, Using recurrent neural network models for early detection of heart failure onset, J. Am. Med. Inform. Assoc. 24 (2016) 361–370, https://doi.org/10.1093/jamia/ocw112.

[16] S. Siami Namini, N. Tavakoli, A. Siami Namin, A comparative analysis of forecasting financial time series using ARIMA, LSTM, and BiLSTM, ArXiv. abs/1911.0 (2019). https://arxiv.org/abs/1911.09512v1.

[17] Z. Che, S. Purushotham, K. Cho, D. Sontag, Y. Liu, Recurrent neural networks for multivariate time series with missing values, Sci. Rep. 8 (2016) 6085, https://doi.org/10.1038/s41598-018-24271-9.

[18] I.M. Baytas, C. Xiao, X. Zhang, F. Wang, A.K. Jain, J. Zhou, Patient subtyping via time-aware LSTM networks, in: Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, New York, NY, USA, 2017, pp. 65–74. 10.1145/3097983.3097997.

[19] C. Lea, M. Flynn, R. Vidal, A. Reiter, G. Hager, Temporal convolutional networks for action segmentation and detection, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (2017) 1003–1012. 10.1109/CVPR.2017.113.

[20] D. Sahoo, S.C.H. Hoi, B. Li, Large scale online multiple kernel regression with application to time-series prediction, ACM Trans. Knowl. Discovery Data 13 (9) (2019) 1–33, https://doi.org/10.1145/3299875.

[21] H. Song, D. Rajan, J.J. Thiagarajan, A. Spanias, Attend and diagnose: clinical time series analysis using attention models, in: 32nd AAAI Conference on Artificial Intelligence, AAAI 2018, AAAI Press, 2018, pp. 4091–4098. https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/viewFile/16325/16790.

[22] D.E. Rumelhart, G.E. Hinton, R.J. Williams, Learning internal representations by error propagation, in: Parallel Distributed Processing: Explorations in the Microstructure of Cognition, MIT Press, 1986, pp. 318–362.

[23] J. Lago, F. De Ridder, B. De Schutter, Forecasting spot electricity prices: deep learning approaches and empirical comparison of traditional algorithms, Appl. Energy 221 (2018) 386–405, https://doi.org/10.1016/j.apenergy.2018.02.069.

[24] F. Perez-Cruz, O. Bousquet, Kernel methods and their potential use in signal processing, IEEE Signal Process Mag. 21 (2004) 57–65, https://doi.org/10.1109/MSP.2004.1296543.

[25] M. Binkowski, G. Marti, P. Donnat, Autoregressive convolutional neural networks for asynchronous time series, in: Proceedings of the 35th International Conference on Machine Learning, 2017, pp. 580–589.

[26] Z. Lipton, D. Kale, R. Wetzel, Modeling missing data in clinical time series with RNN, in: Proceedings of the 1st Machine Learning for Healthcare Conference, 2016, pp. 6776–6786.

[27] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, K. Kavukcuoglu, WaveNet, A Generative model for Raw Audio, ArXiv (2016).

[28] A. Borovykh, S. Bohte, C.W. Oosterlee, Conditional time series forecasting with convolutional neural networks, in: Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2017. 10.1007/978-3-319-68612-7.

[29] X. Wu, B. Shi, Y. Dong, C. Huang, L. Faust, N. V Chawla, RESTFul: resolution-aware forecasting of behavioral time series data, in: Proceedings of the 27th ACM International Conference on Information and Knowledge Management, Association for Computing Machinery, New York, NY, USA, 2018, pp. 1073–1082. 10.1145/3269206.3271794.

[30] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W. Wong, W. Woo, Convolutional LSTM network: a machine learning approach for precipitation nowcasting, in: NIPS'15: Proceedings of the 28th International Conference on Neural Information Processing Systems, 2015, pp. 802–810, https://doi.org/10.1093/toxsci/kfr046.

[31] F. Karim, S. Majumdar, H. Darabi, S. Harford, Multivariate LSTM-FCNs for time series classification, Neural Netw. 116 (2018), https://doi.org/10.1016/j.neunet.2019.04.014.

[32] J. Futoma, S. Hariharan, K. Heller, Learning to detect sepsis with a multitask Gaussian process RNN classifier, in: ICML'17: Proceedings of the 34th International Conference on Machine Learning, 2017, pp. 1174–1182.

[33] S.N. Shukla, B.M. Marlin, Interpolation-prediction networks for irregularly sampled time series, in: International Conference on Learning Representations, 2019. https://openreview.net/forum?id=r1efr3C9Ym.

[34] G.R.G. Lanckriet, N. Cristianini, P. Bartlett, L. El Ghaoui, M.I. Jordan, Learning the Kernel matrix with semidefinite programming, J. Mach. Learn. Res. 5 (2004) 27–72. https://www.jmlr.org/papers/volume5/lanckriet04a/lanckriet04a.ps.
[35] S.-C. Huang, C.-F. Wu, Energy commodity price forecasting with deep multiple kernel learning, Energies 11 (2018) 1–16, https://doi.org/10.3390/en11113029.
[36] H.-F. Yu, N. Rao, I.S. Dhillon, Temporal regularized matrix factorization for high-dimensional time series prediction, in: Proceedings of the 30th International Conference on Neural Information Processing Systems, Curran Associates Inc., Red Hook, NY, USA, 2016, pp. 847–855. http://papers.nips.cc/paper/6160-temporal-regularized-matrix-factorization-for-high-dimensional-time-series-prediction.pdf.
[37] X. Wu, B. Shi, Y. Dong, C. Huang, N. V Chawla, Neural tensor factorization for temporal interaction learning, in: Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining, Association for Computing Machinery, New York, NY, USA, 2019, pp. 537–545. 10.1145/3289600.3290998.
[38] M. Längkvist, L. Karlsson, A. Loutfi, A review of unsupervised feature learning and deep learning for time-series modeling, Pattern Recogn. Lett. 42 (2014) 11–24, https://doi.org/10.1016/j.patrec.2014.01.008.
[39] G. Lai, W.-C. Chang, Y. Yang, H. Liu, Modeling long- and short-term temporal patterns with deep neural networks, in: The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval, 2018, pp. 95–104, https://doi.org/10.1145/3209978.3210006.
[40] K. Cho, B. van Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, Y. Bengio, Learning phrase representations using RNN encoder-decoder for statistical machine translation, ArXiv:1406.1078 [Cs, Stat]. (2014).
[41] I. Sutskever, O. Vinyals, Q. V Le, Sequence to sequence learning with neural networks, in: Proceedings of the 27th International Conference on Neural Information Processing Systems. 2 (2014) 3104–3112. http://papers.nips.cc/paper/5346-sequence-to-sequence-learning-with-neural-networks.pdf.
[42] P. Malhotra, T. Vishnu, L. Vig, P. Agarwal, G. Shroff, TimeNet: pre-trained deep recurrent neural network for time series classification, in: ESANN 2017: European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, 2017. http://www.elen.ucl.ac.be/Proceedings/esann/esannpdf/es2017-100.pdf.
[43] Y. Qin, D. Song, H. Cheng, W. Cheng, G. Jiang, G.W. Cottrell, A Dual-stage attention-based recurrent neural network for time series prediction, in: Proceedings of the 26th International Joint Conference on Artificial Intelligence, AAAI Press, 2017, pp. 2627–2633. https://www.ijcai.org/Proceedings/2017/0366.pdf.
[44] Y. Liang, K. Ouyang, L. Jing, S. Ruan, Y. Liu, J. Zhang, D.S. Rosenblum, Y. Zheng, UrbanFM: inferring fine-grained urban flows, in: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Association for Computing Machinery, New York, NY, USA, 2019, pp. 3132–3142. 10.1145/3292500.3330646.
[45] R. Asadi, A.C. Regan, A spatio-temporal decomposition based deep neural network for time series forecasting, Appl. Soft Comput. 87 (2020) 105963, https://doi.org/10.1016/j.asoc.2019.105963.
[46] Z. Yuan, X. Zhou, T. Yang, Hetero-ConvLSTM, A deep learning approach to traffic accident prediction on heterogeneous spatio-temporal data, in: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2018, pp. 984–992, https://doi.org/10.1145/3219819.3219922.
[47] C. Huang, C. Zhang, J. Zhao, X. Wu, D. Yin, N. Chawla, MiST: a multiview and multimodal spatial-temporal learning framework for citywide abnormal event forecasting, in: WWW '19: The World Wide Web Conference, Association for Computing Machinery, New York, NY, USA, 2019, pp. 717–728. 10.1145/3308558.3313730.
[48] Z. Pan, Y. Liang, W. Wang, Y. Yu, Y. Zheng, J. Zhang, Urban traffic prediction from spatio-temporal data using deep meta learning, in: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Association for Computing Machinery, New York, NY, USA, 2019, pp. 1720–1730. 10.1145/3292500.3330884.
[49] Y. Liang, S. Ke, J. Zhang, X. Yi, Y. Zheng, GeoMAN: multi-level attention networks for geo-sensory time series prediction, in: Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, {IJCAI-18}, International Joint Conferences on Artificial Intelligence Organization, 2018, pp. 3428–3434. 10.24963/ijcai.2018/476.
[50] J. Fan, Q. Li, J. Hou, X. Feng, H. Karimian, S. Lin, A spatiotemporal prediction framework for air pollution based on deep RNN, ISPRS annals of photogrammetry, Rem. Sens. Spat. Inf. Sci. 44W2 (2017) 15–22, https://doi.org/10.5194/isprs-annals-IV-4-W2-15-2017.
[51] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, U. Kaiser, I. Polosukhin, Attention is all you need, in: Proceedings of the 31st International Conference on Neural Information Processing Systems, Curran Associates Inc., Red Hook, NY, USA, 2017, pp. 6000–6010. https://papers.nips.cc/paper/7181-attention-is-all-you-need.pdf.
[52] J. Ma, Z. Shou, A. Zareian, H. Mansour, A. Vetro, S. Chang, CDSA: cross-dimensional self-attention for multivariate, geo-tagged time series imputation, ArXiv Preprint. arXiv, 1905 (2019). https://arxiv.org/abs/1905.09904.
[53] S. Li, X. Jin, Y. Xuan, X. Zhou, W. Chen, Y.-X. Wang, X. Yan, Enhancing the locality and breaking the memory bottleneck of transformer on time series forecasting, Adv. Neural Inform. Process. Syst. (NeurIPS) (2019).
[54] Y. Bengio, M. Hc-j, F. Gingras, Recurrent neural networks for missing or asynchronous data, in: NIPS'95: Proceedings of the 8th International Conference on Neural Information Processing Systems, 1995, pp. 395–401.
[55] C.F. Ansley, R. Kohn, On the estimation of ARIMA models with missing values, in: E. Parzen (Ed.), Time Series Analysis of Irregularly Observed Data, Springer, New York, New York, NY, 1984, pp. 9–37, https://doi.org/10.1007/978-1-4684-9403-7_2.
[56] F. Cismondi, A. Fialho, S. Vieira, S. Reti, J. Sousa, S. Finkelstein, Missing data in medical databases: impute, delete or classify?, Artif Intell. Med. 58 (2013) 63–72, https://doi.org/10.1016/j.artmed.2013.01.003.
[57] V. Fortuin, D. Baranchuk, G. Rätsch, S. Mandt, GP-VAE: deep probabilistic time series imputation, 23rd International Conference on Artificial Intelligence and Statistics, 2020.
[58] M. Kulesh, M. Holschneider, K. Kurennaya, Adaptive metrics in the nearest neighbours method, Physica D 237 (2008) 283–291, https://doi.org/10.1016/j.physd.2007.08.019.
[59] W. Shi, Y. Zhu, P. Yu, T. Huang, C. Wang, Y. Mao, Y. Chen, Temporal dynamic matrix factorization for missing data prediction in large scale coevolving time series, IEEE Access 4 (2016) 6719–6732, https://doi.org/10.1109/ACCESS.2016.2606242.
[60] P.J. García-Laencina, J.-L. Sancho-Gómez, A.R. Figueiras-Vidal, Pattern classification with missing data: a review, Neural Comput. Appl. 19 (2010) 263–282, https://doi.org/10.1007/s00521-009-0295-6.
[61] X. Tang, H. Yao, Y. Sun, C. Aggarwal, P. Mitra, S. Wang, Joint modeling of local and global temporal dynamics for multivariate time series forecasting with missing values, in: American Association for Artificial Intelligence, 2019. 10.1609/aaai.v34i04.6056.
[62] Y. Luo, X. Cai, Y. ZHANG, J. Xu, Y. Xiaojie, Multivariate time series imputation with generative adversarial networks, in: Advances in Neural Information Processing Systems 31 (NIPS 2018), Curran Associates, Inc., 2018, pp. 1596–1607. http://papers.nips.cc/paper/7432-multivariate-time-series-imputation-with-generative-adversarial-networks.pdf.
[63] M. Nguyen, N. Sun, D. Alexander, J. Feng, B.T.T. Yeo, Modeling Alzheimer's disease progression using deep recurrent neural networks, in: 2018 International Workshop on Pattern Recognition in Neuroimaging (PRNI), Singapore, 2018, pp. 1–4. 10.1109/PRNI.2018.8423955.
[64] H. Yuan, G. Xu, Z. Yao, J. Jia, Y. Zhang, Imputation of missing data in time series for air pollutants using long short-term memory recurrent neural networks, in: ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers, 2018, pp. 1293–1300, https://doi.org/10.1145/3267305.3274648.
[65] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2016 (2016) 770–778, https://doi.org/10.1109/CVPR.2016.90.
[66] L. Shen, Q. Ma, S. Li, End-to-end time series imputation via residual short paths, in: J. Zhu, I. Takeuchi (Eds.), Proceedings of The 10th Asian Conference on Machine Learning, 2018, pp. 248–263.
[67] J. Zhang, X. Mu, J. Fang, Y. Yang, Time series imputation via integration of revealed information based on the residual shortcut connection, IEEE Access 7 (2019) 102397–102405, https://doi.org/10.1109/ACCESS.2019.2928641.
[68] Y.J. Kim, M. Chi, Temporal belief memory: imputing missing data during RNN training, in: IJCAI'18: Proceedings of the 27th International Joint Conference on Artificial Intelligence, 2018, pp. 2326–2332.
[69] J. Zhou, Z. Huang, Recover missing sensor data with iterative imputing network, Workshops of the 32 AAAI Conference on Artificial Intelligence, 2018.
[70] A. Graves, S. Fernández, J. Schmidhuber, Bidirectional LSTM networks for improved phoneme classification and recognition, in: Artificial Neural Networks: Formal Models and Their Applications - ICANN 2005, 2005, pp. 799–804. 10.1007/11550907_126.
[71] J. Yoon, W.R. Zame, M. van der Schaar, Multi-directional recurrent neural networks: a novel method for estimating missing data, in: International Conference on Machine Learning (ICML) Time Series Workshop, 2017. https://icml.cc/Conferences/2019/ScheduleMultitrack?event=3525.
[72] J. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho, Y. Bengio, Attention-based models for speech recognition, in: Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 1, MIT Press, Cambridge, MA, USA, 2015, pp. 577–585. https://papers.nips.cc/paper/5847-attention-based-models-for-speech-recognition.pdf.
[73] D. Bahdanau, K. Cho, Y. Bengio, Neural machine translation by jointly learning to align and translate, in: Y. Bengio, Y. LeCun (Eds.), 3rd International Conference on Learning Representations, {ICLR} 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings, 2015. http://arxiv.org/abs/1409.0473.
[74] Y. Tang, J. Xu, K. Matsumoto, C. Ono, Sequence-to-sequence model with attention for time series classification, IEEE International Conference on Data Mining Workshops, ICDMW (2017), https://doi.org/10.1109/ICDMW.2016.0078.
[75] P. Nguyen, T. Tran, S. Venkatesh, Deep learning to attend to risk in ICU, CEUR Workshop Proc. 2017 (1891) 25–29. http://ceur-ws.org/Vol-1891/paper4.pdf.
[76] B.P. Singh, I. Deznabi, B. Narasimhan, B. Kucharski, R. Uppaal, A. Josyula, M. Fiterau, Multi-resolution networks for flexible irregular time series modeling (Multi-FIT), ArXiv. abs/1905.0 (2019). https://arxiv.org/pdf/1905.00125.pdf.
[77] Y. Zhang, P. Thorburn, W. Xiang, P. Fitch, SSIM - a deep learning approach for recovering missing time series sensor data, IEEE Internet Things J. 6 (2019) 6618–6628, https://doi.org/10.1109/JIOT.2019.2909038.

[78] Y. Tang, J. Xu, K. Matsumoto, C. Ono, Sequence-to-sequence model with attention for time series classification, IEEE International Conference on Data Mining Workshops, ICDMW (2017) 503–510, https://doi.org/10.1109/ICDMW.2016.0078.
[79] J. Dabrowski, A. Rahman, Sequence-to-sequence imputation of missing sensor data, in: Australasian Conference on Artificial Intelligence, 2019, pp. 265–276. 10.1007/978-3-030-35288-2_22.
[80] C. Esteban, S. Hyland, G. Rätsch, Real-valued (Medical) time series generation with recurrent conditional GANs, ArXiv. abs/1706.0 (2017). https://arxiv.org/abs/1706.02633.
[81] S.C. Li, B. Jiang, B. Marlin, Learning from incomplete data with generative adversarial networks, in: International Conference on Learning Representations, 2019. https://openreview.net/forum?id=S1lDV3RcKm.
[82] E. Choi, T. Bahadori, J. Sun, Doctor AI: predicting clinical events via recurrent neural networks, in: Proceedings of the 1st Machine Learning for Healthcare Conference 56 (2016) 301–318. http://proceedings.mlr.press/v56/Choi16.html.
[83] R.T.Q. Chen, Y. Rubanova, J. Bettencourt, D. Duvenaud, Neural ordinary differential equations, in: Proceedings of the 32nd International Conference on Neural Information Processing Systems, Curran Associates Inc., Red Hook, NY, USA, 2018, pp. 6572–6583. https://papers.nips.cc/paper/7892-neural-ordinary-differential-equations.pdf.
[84] Y. Rubanova, R.T.Q. Chen, D. Duvenaud, Latent ODEs for irregularly-sampled time series, Adv. Neural Inform. Process. Syst. (NeurIPS) (2019).
[85] M. Habiba, B.A. Pearlmutter, Neural ordinary differential equation based recurrent neural network model, in: 2020 31st Irish Signals and Systems Conference (ISSC), 2020, pp. 1–6. 10.1109/ISSC49989.2020.9180182.
[86] G. Zhou, J. Wu, C. Zhang, Z.-H. Zhou, Minimal gated unit for recurrent neural networks, Int. J. Autom. Comput. 13 (2016) 226–234, https://doi.org/10.1007/s11633-016-1006-2.
[87] A. Nugaliyadde, F. Sohel, K.W. Wong, H. Xie, Language modeling through long-term memory network, International Joint Conference on Neural Networks (IJCNN) 2019 (2019) 1–6, https://doi.org/10.1109/IJCNN.2019.8851909.
[88] O. Nina, A. Rodriguez, Simplified LSTM unit and search space probability exploration for image description, in: 2015 10th International Conference on Information, Communications and Signal Processing (ICICS), 2015, pp. 1–5. 10.1109/ICICS.2015.7459976.
[89] T. Pham, T. Tran, D. Phung, S. Venkatesh, DeepCare: a deep dynamic memory model for predictive medicine, in: PAKDD 2016: Proceedings, Part II, of the 20th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining, Springer International Publishing, Cham, 2016, pp. 30–41. 10.1007/978-3-319-31750-2_3.
[90] N. Vecoven, D. Ernst, G. Drion, A bio-inspired bistable recurrent cell allows for long-lasting memory, ArXiv. abs/2006.0 (2020). https://arxiv.org/pdf/2006.05252.pdf.
[91] N.G. Reich, J. Lessler, K. Sakrejda, S.A. Lauer, S. Iamsirithaworn, D.A.T. Cummings, Case study in evaluating time series prediction models using the relative mean absolute error, Am. Statist. 70 (3) (2016) 285–292, https://doi.org/10.1080/00031305.2016.1148631.
[92] [Dataset], A.L. Goldberger, L.A.N. Amaral, L. Glass, J.M. Hausdorff, P.C. Ivanov, R. G. Mark, J.E. Mietus, G.B. Moody, C.-K. Peng, H.E. Stanley, PhysioBank, PhysioToolkit, and PhysioNet: components of a new research resource for complex physiologic signals, Circulation 101 (n.d.) e215–e220. https://www.ahajournals.org/doi/full/10.1161/01.cir.101.23.e215.
[93] A.E.W. Johnson, T.J. Pollard, L. Shen, L.H. Lehman, M. Feng, M. Ghassemi, B. Moody, P. Szolovits, L. Anthony Celi, R.G. Mark, MIMIC-III, a freely accessible critical care database, Sci. Data 3 (2016) 160035, https://doi.org/10.1038/sdata.2016.35.
[94] H. Harutyunyan, H. Khachatrian, D.C. Kale, G. Ver Steeg, A. Galstyan, Multitask learning and benchmarking with clinical time series data, Sci. Data 6 (2019) 96, https://doi.org/10.1038/s41597-019-0103-9.
[95] [Dataset], Beijing PM2.5 Dataset, (n.d.). http://www.bjmemc.com.cn/.
[96] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Comput. (1997), https://doi.org/10.1162/neco.1997.9.8.1735.
[97] H. Sak, A. Senior, F. Beaufays, Long short-term memory based recurrent neural network architectures for large vocabulary speech recognition, ArXiv Preprint. arXiv:1402 (2014). https://arxiv.org/abs/1402.1128.
[98] T.J. Ham, S. Jung, S. Kim, Y. Oh, Y. Park, Y. Song, J.-H. Park, S. Lee, K. Park, J. Lee, D.-K. Jeong, A^3: accelerating attention mechanisms in neural networks with approximation, in: IEEE International Symposium on High Performance Computer Architecture (HPCA), 2020, pp. 328–341, https://doi.org/10.1109/HPCA47549.2020.00035.

Philip B. Weerakody received the B.Eng. degree in Electronic and Electrical Engineering from The University of Western Australia, in 1996. He is a chartered member of the Institution of Engineering and Technology (IET, UK). He is currently working as a systems technology consultant within the oil and gas industry in Western Australia and is a Ph.D. candidate in Information Technology at Murdoch University, Western Australia. His research interests include machine learning and deep learning applied to time series applications.

Kok Wai Wong is an Associate Professor with the Discipline of Information Technology, Mathematics and Statistics at the College of Science, Health, Engineering and Education at Murdoch University in Western Australia. He is the current Vice President (Membership) for The Asia Pacific Neural Network Society (APNNS). He is a Senior Member of Institute of Electrical and Electronics Engineers (IEEE), a Senior member of Australia Computer Society (ACS), and Certified Professional of ACS. He is also the current chapter chair for the Joint Chapter of IEEE Computer Intelligence Society, and Robotic and Automation Society (WA Chapter). His current research interests include Intelligent Data Mining, Artificial Intelligence and Machine Learning

Guanjin Wang received the join Ph.D degree in software engineering from University of Technology Sydney and The Hong Kong Polytechnic University. She is currently a lecturer in Information Technology with Discipline of Information Technology, Mathematics & Statistics, Murdoch University, Perth, Australia. Her current research interest lies in the areas of machine learning and health informatics.

Professor Wendell Ela is the inaugural Chair of Desalination and Water Treatment at Murdoch University. he gained his PhD from Stanford University and was on the faculty of Chemical and Environmental Engineering at the University of Arizona for 15 years before moving to Australia in 2015. Alongside exceptional students and colleagues, he has published widely in scientific journals and co-authored one of the most widely used textbooks in environmental engineering. He has supervised more than 65 degree by research Masters and PhD students, and regularly serves on advisory and review committees for regulatory, water management, and environmental research agencies.