Exploiting Context and Attention using Recurrent Neural Network for Multivariate Time Series Forecasting *

Rashmi Dutta Baruah[1,2][0000−0002−6756−8157] and Mario Muñoz-Organero[1][0000−0003−4199−2002]

[1] Department of Telematic Engineering, Universidad Carlos III de Madrid, Avda. de la Universidad, 30, Leganés, Madrid 28911, Spain
{rdutta, munozm}@it.uc3m.es

[2] Department of Computer Science and Engineering, Indian Institute of Technology Guwahati, Guwahati - 781039, Assam, India

Abstract. In the current era of Internet of Things, typically data from multiple sources are captured through various sensors yielding Multivariate Time Series (MTS) data. Sensor MTS prediction has several real-life applications in various domains such as healthcare, manufacturing, and agriculture. In this paper, we propose a novel Recurrent Neural Network (RNN) architecture that leverages contextual information and attention mechanism for sensor MTS prediction. We adopt the notion of primary and contextual features to distinguish between the features that are independently useful for learning irrespective of other features, and the features that are not useful in isolation. The contextual information is represented through the contextual features and when used with primary features can potentially improve the performance of the model. The proposed architecture uses the contextual features in two ways. Firstly, to weight the primary input features depending on the context, and secondly to weight the hidden states in the alignment model. The latter is used to compute the dependencies between hidden states (representations) to derive the attention vector. Further, integration of the context and attention allows visualising temporally and spatially the relevant parts of the input sequence which are influencing the prediction. To evaluate the proposed architecture, we used two benchmark datasets as they provide contextual information. The first is NASA Turbofan Engine Degradation Simulation dataset for estimating Remaining Useful Life, and the second is appliances energy prediction dataset. We compared the proposed approach with the state-of-the-art methods and observed improved prediction results, particularly with respect to the first dataset.

Keywords: Recurrent Neural Network · Gated Recurrent Unit· Context · Attention · Multivariate Sensor Time Series · Remaining Useful Life · Appliance energy prediction

* This work is supported by CONEX-Plus programme funded by Universidad Carlos III de Madrid and the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement No. 801538.

1 Introduction

The Internet of Things (IoT), driven by advanced sensors, computing and communication technologies, has enabled capturing data from various sources and utilise them to realise various 'smart' environments such as smart homes, smart cities, smart factories. The data captured through various sensors can be considered as Multivariate Time Series (MTS) [25]. Sensor MTS can be used for learning predictive models, thereby innovating various applications for such environments. These data rich environments often provide contextual information that can be leveraged while learning the predictive models to improve the performance [26]. For example, an automated fault detection and diagnosis agent for a HVAC system in a smart building can utilise the environmental factors such as indoor and outdoor temperature and humidity (contextual information) along with the current and voltage data from the HVAC system [11]. However, most of the machine learning algorithms do not explicitly take into account the available contextual information [13].

We adopt the definitions of *primary* and *contextual* features to distinguish between the features that are independently useful for learning irrespective of other features, and the features that are not useful in isolation [26]. The contextual data available in terms of contextual features may influence the performance by improving the model but may not be involved directly in learning. We also emphasize that the contextual data is available from the environment where primary data is captured and is a MTS itself. Over the past decade, Recurrent Neural Networks (RNNs) including Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRU) have been widely used for sequential or time series data modeling. They are well known for capturing temporal contexts implicitly due to their internal memory. It is worth mentioning here that in this paper, the focus is not on the temporal context or the contexts that are generated within the network from input and/or output signals. Here, the focus is on *explicit* contexts, which is in the form of additional data available from the problem domains. The current RNN architectures do not explicitly exploit the contextual data. Recently, in [8, 9], a context integrated RNN (CiRNN) which uses GRU as basic unit is proposed. CiRNN, takes both primary and contextual features as input. The contextual features are used to weight the primary features depending on the context such that the input to hidden layer weights are function of contextual features. With CiRNN, a significant improvement in performance is observed when compared to stat-of-the art methods for the task of remaining useful life prediction in machine prognostics.

On the other hand, recently, attention mechanism has received a great deal of attention mainly due to the work of Bahdanau et al. [1] in the area of neural machine translation (NMT). Typically, NMT models are based on encoder-decoder approach where the encoder is for the source language and the decoder is for the target language. For the source language encoder, usually RNNs are used where the necessary information of the source sentence is compressed in to a fixed length vector. For longer sentences, this conventional approach, gives poor performance. Attention mechanism allows to capture the information from all

or few source positions in the encoder thereby alleviating the problem with conventional encoder-decoder approach.

In this paper, we propose a novel RNN architecture that exploits context and attention for sensor MTS prediction. The architecture primarily consists of CiRNN with attention layer and finally a fully connected (FCN) layer. In addition to CiRNN, attention layer uses the contextual features to weight the hidden sates of the alignment model [17]. The alignment model is used to compute the dependencies among the hidden states or representations to derive the attention vector. Further, adding contextual attention to CiRNN, provides interpretations at two levels. First, the input features weighted by the context indicates which of the features are relevant in a given context for prediction. Second, the attention weights show which parts of the time series, apart from the last time step, the network is attending to prior to the prediction.

To demonstrate the effectiveness of the proposed approach, it is applied to two benchmark datasets. The first task is in the domain of engine health prognostics where we considered the widely used NASA Turbofan Engine Degradation Simulation dataset (C-MAPSS dataset) for estimating RUL [20]. The dataset contains information from 21 sensors and 3 operational settings. The operational settings have a substantial effect on engine performance and represent the contextual information required for the proposed model. The second task is to predict household appliance energy usage where Appliances energy prediction (AEP) dataset from UCI repository is used [2]. The results of the proposed model is compared with baseline models and also state-of-the art methods. The results show an improvement in performance in prediction results.

The rest of the paper is organised as follows. In the next section, we briefly present the related work. In section 3, we discuss the architecture and learning in the proposed architecture . Section 4, first describes the datasets and then discusses the experiments and results . Finally, section 5 concludes the paper.

2 Related work

Considering the increase in amount and dimensionality of time series data, particularly data from ubiquitous sensors, deep learning methods have been applied to a great extent to extract features and to recognize complex latent patterns [10]. In this paper, we limit the scope of related work to prediction models that use RNNs and attention mechanism for MTS prediction. The work related to context integration to RNN is largely done in the area of natural language processing (NLP) domain and it has been discussed in [8, 9].

In [5], three extensions of content attention [1] are provided that use the relative positions in input and output to capture the pseudo-periods in time series. Several experiments with MTS data showed that for multi-horizon forecasting the proposed approach is significantly better than RNN with attention and baseline methods based on ARIMA and Random Forests (RF). In [19], a dual-stage attention-based recurrent neural network (DA-RNN) is proposed which consists of an encoder with an input attention mechanism and a decoder

with a temporal attention mechanism. It is tested for predicting indoor temperature and for predicting the index value of the NASDAQ 100 using stock dataset. A temporal pattern attention mechanism for multivariate time series is presented in [22]. The focus is on extracting relevant input features rather than time steps through attention. CNN filters are applied to the row vectors of RNN (encoder) hidden units before deriving the attention vector. They tested the approach with six MTS datasets that include various domains such as energy, music, traffic, and finance, and achieved good results. In [7], temporal attention-based encoder–decoder model is proposed for MTS multi-step forecasting tasks. It uses a Bidirectional-LSTM (Bi-LSTM) with attention mechanism to encode the hidden representations of MTS data as the temporal context vector. Another LSTM is used to decode the hidden representation for prediction. Experiments on five MTS datasets showed that the proposed model is effective in multi-step forecasting. Cheng et al. [3] proposed a model that uses dual stage attention with Bi-LSTM as encoder and LSTM decoder. The experimental results with MTS data related to energy and finance showed better performance for single step and multi-step prediction. However, for longer time steps, the prediction performance of the model reduces.

To summarise, the existing approaches discussed here leverage attention mechanism to deal with longer time sequences which LSTM or GRU alone is not able to handle. Also, the attention mechanism is tailored for MTS data such that relevant input features is taken into consideration while computing the attention vector. None of the previous studies, to the best of our knowledge, investigated the possibility of utilizing contextual information to weight the hidden states as well as the input features through CiRNN to realise a context sensitive attention based model for improving sensor MTS prediction.

3 Proposed Approach

In this section, we first present the overall framework and finally the details of each of the units is provided[3].

3.1 Proposed Framework

Fig. 1 shows the proposed context sensitive attention-based RNN model for the prediction of sensor MTS. It consists of Context Integrated Gated Recurrent Units (CiGRU) [8] which have recurrent connections and takes the primary and contextual input. The learned sequential features (hidden states of CiGRU) are provided as input to the attention layer. The output of the attention layer is an attention vector ($\mathbf{a}_t$) which is computed using the temporal context vector (TCV). Note that, conventionally the TCV is referred to as *context vector*. Here, to make a distinction between temporal context and explicit context we are using the term TCV. The TCV ($\mathbf{c}_t$) is computed using the attention weights

[3] The code is available at https://github.com/rduttabaruah/CiRNN

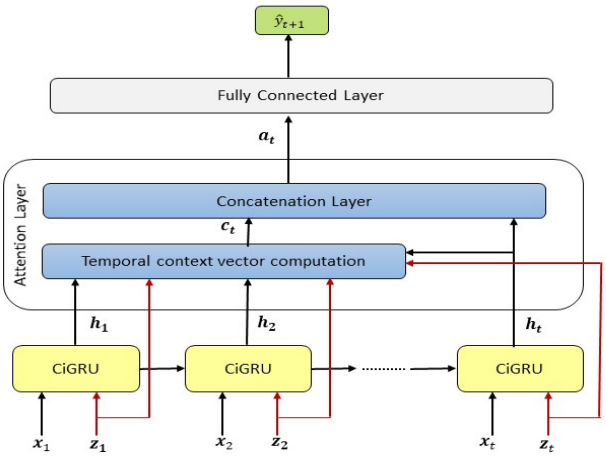

Fig. 1: Proposed framework with CiGRU and attention.

computed in the attention layer. The target hidden state (h_t) is concatenated with the TCV through a concatenation layer to produce the attention vector. Finally, the attention vector is passed as input to fully connected layers (FCLs) in the network to predict the target at time step $(t+1)$.

3.2 Context Integrated Gated Recurrent Unit

The RNN is composed of CiGRU units [8, 9], which are fundamentally GRUs [4] with an additional context input. In CiGRU, the input to hidden unit connection weights are dependent on the context variables. Fig. 2 shows the architecture of a single CiGRU. The output ($\hat{\mathbf{y}}_t \in \Re^{n_y \times 1}$) at time step t in CiGRU is computed in similar manner as in GRU. However, the candidate hidden state ($\tilde{\mathbf{h}}_t \in \Re^{n_h \times 1}$), update gate ($\mathbf{s}_t \in \Re^{n_h \times 1}$), and reset gate ($\mathbf{r}_t \in \Re^{n_h \times 1}$) values are determined based on context $\mathbf{z}_t$ as shown below:

$$
\begin{aligned}
\hat{\mathbf{y}}_t &= f(\mathbf{V}\mathbf{h}_t + \mathbf{b}_y) \\
\mathbf{h}_t &= \mathbf{s}_t \odot \mathbf{h}_{t-1} + (1 - \mathbf{s}_t) \odot \tilde{\mathbf{h}}_t \\
\tilde{\mathbf{h}}_t &= tanh(\mathbf{W}^h(\mathbf{z}_t)\mathbf{x}_t + \mathbf{U}^h(\mathbf{r}_t \odot \mathbf{h}_{t-1})) \\
\mathbf{s}_t &= \sigma(\mathbf{W}^s(\mathbf{z}_t)\mathbf{x}_t + \mathbf{U}^s\mathbf{h}_{t-1}) \\
\mathbf{r}_t &= \sigma(\mathbf{W}^r(\mathbf{z}_t)\mathbf{x}_t + \mathbf{U}^r\mathbf{h}_{t-1})
\end{aligned}
\tag{1}
$$

where n_x, n_y, n_z, n_h are the input, output, context, and hidden unit dimensions, $\mathbf{h}_t \in \Re^{n_h \times 1}$ is the hidden unit activation at time step t, $\mathbf{U} \in \Re^{n_h \times n_h}$, $\mathbf{V} \in$

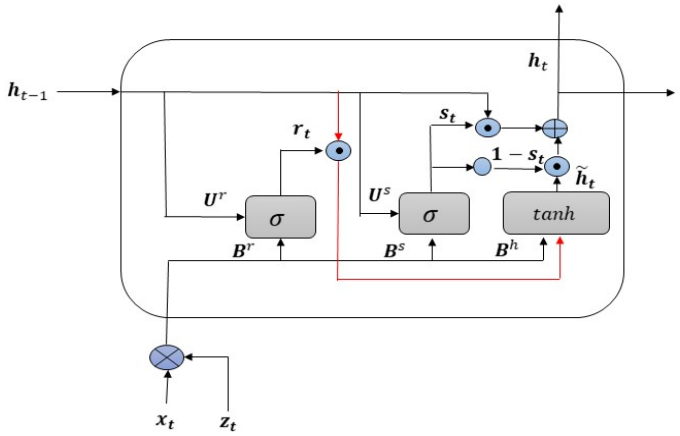

Fig. 2: A Context Integrated Gated Recurrent Unit.

$\Re^{n_y \times n_h}$, $\mathbf{W} \in \Re^{n_h \times n_x}$ are the parameter (weight) matrices, and $\mathbf{b}_y \in \Re^{n_y \times 1}$ is the bias vector.

In equation 1, the weights associated with the input ($\mathbf{x}_t$) are dependent on the vector of contextual variables ($\mathbf{z}_t$). Let us consider one of the parameters, $\mathbf{W}^h(\mathbf{z}_t)$. The parameters $\mathbf{W}^s(\mathbf{z}_t)$ and $\mathbf{W}^r(\mathbf{z}_t)$ can be expressed in a similar way. The matrix $\mathbf{W}^h(\mathbf{z}_t)$ is of dimension $n_h \times n_x$ and each of the components can be given as:

$$\mathbf{W}^h(\mathbf{z}_t) = \begin{bmatrix} w^h_{11}(\mathbf{z}_t) & w^h_{12}(\mathbf{z}_t) & \cdots & w^h_{1n_x}(\mathbf{z}_t) \\ w^h_{21}(\mathbf{z}_t) & w^h_{22}(\mathbf{z}_t) & \cdots & w^h_{2n_x}(\mathbf{z}_t) \\ \vdots & \vdots & \cdots & \vdots \\ w^h_{n_h 1}(\mathbf{z}_t) & w^h_{n_h 2}(\mathbf{z}_t) & \cdots & w^h_{n_h n_x}(\mathbf{z}_t) \end{bmatrix} \tag{2}$$

where each element of the matrix can be expressed as:

$$\begin{aligned} &w^h_{ki}(\mathbf{z}_t) = \mathbf{B}^h_{ki}\mathbf{G}(\mathbf{z}_t), \quad k = 1, .., n_h, \quad i = 1, .., n_x \\ &\mathbf{B}^h_{ki} = [b^h_{ki1}, b^h_{ki2}, \cdots, b^h_{kim}] \end{aligned} \tag{3}$$

where $\mathbf{G}(\mathbf{z}_t) = [g_1(\mathbf{z}_t), g_2(\mathbf{z}_t), ..., g_m(\mathbf{z}_t)]^T$ is a vector of basis functions that can be chosen at the time of design. $\mathbf{B}^h_{ki}$ is a vector of coefficients that specify the dependence of weights on context variables. We can define a matrix $\mathbf{B}^h$ where each row $\mathbf{B}^h_k$ can be formed by concatenating coefficient vectors $\mathbf{B}^h_{ki}$ as shown below. Therefore, $\mathbf{B}^h$ is of dimension $(n_h \times n_x m)$.

$$\mathbf{B}^h_k = [\mathbf{B}^h_{k1}, \mathbf{B}^h_{k2}, \cdots, \mathbf{B}^h_{kn_x}], \quad k = 1, 2, ..., n_h \tag{4}$$

Using $\mathbf{B}^h$ and similarly $\mathbf{B}^s$ and $\mathbf{B}^r$, the candidate hidden state $\tilde{\mathbf{h}}_t$, the update gate $\mathbf{s}_t$, and reset gate $\mathbf{r}_t$ in equation (1) can be expressed as:

$$\begin{aligned}\tilde{\mathbf{h}_t} &= tanh[\mathbf{B}^h(\mathbf{x}_t \otimes \mathbf{G}(\mathbf{z}_t)) + \mathbf{U}^h(\mathbf{r}_t \odot \mathbf{h}_{t-1})] \\ \mathbf{s}_t &= \sigma[\mathbf{B}^s(\mathbf{x}_t \otimes \mathbf{G}(\mathbf{z}_t)) + \mathbf{U}^s\mathbf{h}_{t-1}] \\ \mathbf{r}_t &= \sigma[\mathbf{B}^r(\mathbf{x}_t \otimes \mathbf{G}(\mathbf{z}_t)) + \mathbf{U}^r\mathbf{h}_{t-1}]\end{aligned} \tag{5}$$

where the symbol $\otimes$ represents Kronecker product.

Learning of the vector of coefficients $\mathbf{B}^h_{ki}$ with m elements is similar to RNN. For RUL estimation, L2 loss function and back propagation through time (BPTT) is used. Finally, the parameters can be optimized using any suitable optimization algorithm such as stochastic gradient descent (SGD), Adam or RMSProp. The details are available in [9].

3.3 Attention Mechanism

The attention used here is global attention [17], at each time step t, the hidden states of CiGRU is used to compute the TCV ($\mathbf{c}_t$) which captures relevant information about the next target $\mathbf{y}_t$. The vector $\mathbf{c}_t$ is defined as:

$$\mathbf{c}_t = \sum_{i=1}^{t} \alpha_{ti}\mathbf{h}_i \tag{6}$$

where α_{ti} is the attention weight. So, the context vector considers all the hidden states of the CiGRU weighted by attention weights. The attention weight is given as:

$$\alpha_{ti} = \frac{exp(f(\mathbf{h}_t, \mathbf{h}_i))}{\sum_{i=1}^{t} exp(f(\mathbf{h}_t, \mathbf{h}_i))} \tag{7}$$

The function f is given by, $f(\mathbf{h}_t, \mathbf{h}_i) = \mathbf{h}_t^T\mathbf{W}^a(\mathbf{z}_t)\mathbf{h}_i$. Here, the weight matrix $\mathbf{W}^a \in \Re^{n_h \times n_h}$ depends on the context $\mathbf{Z}$. As discussed in section 3.2, f can further be expressed as,

$$f(\mathbf{h}_t, \mathbf{h}_i) = \mathbf{h}_t^T[\mathbf{B}^a(\mathbf{h}_i \otimes G(\mathbf{z}_t))] \tag{8}$$

where $\mathbf{B}^a$ is of dimension $(n_h \times n_h m)$.

Finally, the TCV and the hidden state $\mathbf{h}_t$ is combined in the concatenation layer through a fully connected layer to get the attention vector as given by the following equation.

$$\mathbf{a}_t = tanh(\mathbf{W}^c[\mathbf{c}_t, \mathbf{h}_t]) \tag{9}$$

4 Experiments and Results

In this section, we first describe the datasets, and then discuss the experiments and the results achieved with the proposed model. The results from the proposed model are compared with baseline models and also with the state-of-the-art methods.

4.1 Dataset description

For evaluation of the proposed model, we considered two benchmark datasets where contextual information is available. The first dataset is the widely used, NASA Turbofan Engine Degradation Simulation Data Set (TEDS) [20]. The dataset is generated using Commercial Modular Aero-Propulsion System Simulation (C-MAPSS) tool. The dataset consists of four distinct datasets that contain information from 21 sensors (such as Total temperature at fan inlet, Total temperature at Low Pressure Compressor outlet), 3 operational settings (flight altitude, Mach number, and throttle resolver angle). In addition to these, engine identification number, and cycles of each engine is also available. We considered the dataset (FD002) which has six operating conditions. Due to the presence of different working conditions, it is suitable for the proposed model. The operating working conditions can be treated as contextual features while training the model. The dataset provides separate training and test sets. In the training set, the sensor data is captured until the system fails. Whereas in the test set it is captured up to a certain time prior to the failure. The test sets also provide true Remaining Useful Life (RUL) values. The FD002 dataset has 260 and 259 number of engines, 53759 and 33991 data samples in train and test set, respectively, and has one fault mode.

The second dataset is Appliance Energy Prediction (AEP) dataset [2]. The dataset comprises of measurements of house temperature and humidity with a 10 minute interval for a period of 4.5 months. The indoor data was merged with weather data from the nearest airport weather station (Chievres Airport, Belgium) using date and time column. The weather data was retrieved from a public data set from Reliable Prognosis (rp5.ru). Two random variables are also included in the data set for testing the regression models and to filter out non-predictive attributes (parameters). It consists of 19735 data samples and 29 features including the random variables. The dataset has features like, energy use of light fixtures in the house, Temperature in kitchen area (T1), Humidity in kitchen area (RH_1), Appliances energy usage, and weather data such as outside temperature, humidity, pressure, wind speed etc.

4.2 Data Preprocessing

For TEDS dataset, the data from the 21 sensors are analysed. First, univariate and bivariate analyses are performed and the trend of sensor data is also analyzed. We selected 6 sensors ($s_1, s_2, s_8, s_{13}, s_{14}, s_{19}$) considering scatter plot observations and correlation analysis [8]. As the training data does not have the true RUL values, piece-wise degradation model [15, 6] is used to get the values. With this degradation model, initially for a specific period, the RUL values remain constant and after that as the number of time cycles progress, the RUL values reduces linearly [12]. For our experiments, 125 is selected as the initial constant RUL based on existing works that have used C-MAPSS dataset. The data is normalized using min-max normalization and then it is clustered into

6 clusters based on operational regimes and then normalized again using cluster mean and range. Finally, the data is smoothed using moving average with window size of 3 while excluding the target. The target variable is RUL.

For AEP dataset, first the two random variables are removed and then the data is normalized using min-max normalization. The outside temperature, pressure, humidity, wind speed and hour of the day is considered as context variable and the remaining indoor variables are used as primary features. The target variable is Appliances energy usage.

4.3 Performance Metrics

The performance of the proposed model is measured using three metrics, RMSE (Root Mean Squared Error), MAE (Mean Absolute Error), and score from a asymmetric scoring function.

The scoring function is specific to the problem of RUL estimation and was proposed by by Saxena et al. [20]. The score metric given in equation (10) is formulated in such a way that late predictions (positive errors) draw more penalty compared to early predictions (negative errors). In either case, the penalty increases exponentially with error.

$$score = \begin{cases} \sum_{i=1}^{n} e^{-\frac{d_i}{a_1}} - 1, \text{ if } d_i < 0 \\ \sum_{i=1}^{n} e^{\frac{d_i}{a_2}} - 1, \text{ if } d_i \geq 0 \end{cases} \tag{10}$$

where $a_1 = 10$, $a_2 = 13$, and $d_i = \hat{RUL}_i - RUL_i$ is the difference between predicted RUL and actual RUL values, n is the number of samples in the test data .

4.4 Training and Validation

To train the models, a validation set is created from the available training dataset of TEDS. From each engine unit the last l samples, where l is multiple of sequence length, are kept for validation. Thus, the validation set consists of samples from each engine unit as in the test set. For the experiments, l is set to 2. The AEP dataset is split into 80% training and 20% testing and for another set of experiments it is divided as 75% training and 25% testing. From the training set, 10% data is used as validation set. This ratio is selected to compare the results with existing works. The following hyperparameters are used for tuning the model, number of hidden units (RNN): 15-30, step 5, number of hidden units (FCL): 5-30, step 5, sequence (window) length: {10, 15, 20}, learning rate: $loguniform(1e-5, 1e-3)$, oprimizer: SGD, Adam, RMSProp. The number of CiGRU layers is fixed to 1 and the batch size is set to 128. For the contextual inputs, polynomial basis functions of degree 2 are used. The proposed model is implemented using Python 3.10 with PyTorch 1.12 library in a Dell Precision 3650 workstation with Ubuntu 20.04 OS. The hyperparameters are optimized using Optuna with Tree-Structured Parzen Estimater sampler (TPESampler) and Median Pruner.

Table 1: Model configurations and Hyperparameters

Dataset	Model	Hyperparameter	Optimizer
TEDS (FD002)	GRU [9]	$15, 10, 64, 9 \times 10^{-3}$	RMSProp
	CiGRU [9]	$20, 15, 64, 5 \times 10^{-3}$	RMSProp
	CiGRU + A	$30, 5, 15, 128, 3 \times 10^{-3}$	Adam
	CiGRU + CxA (Proposed)	$20, 20, 10, 128, 6 \times 10^{-3}$	RMSProp
AEP	GRU	$15, 15, 128, 2 \times 10^{-3}$	RMSProp
	CiGRU	$10, 15, 128, 5 \times 10^{-5}$	RMSProp
	CiGRU + A	$25, 10, 20, 128, 2 \times 10^{-3}$	Adam
	CiGRU + CxA (Proposed)	$15, 15, 20, 128, 1 \times 10^{-3}$	Adam

We considered two models as baseline to compare with the proposed model, for which we use the acronym as CiGRU + CxA (CiGRU with contextual attention). The first baseline model is RNN with GRUs, and the second is RNN with CiGRU and attention (CiGRU + A). All the models are trained with primary as well as contextual features, however, in the first model (GRU) contextual features are concatenated with primary features. The latter way of using contextual features with primary features is also referred to as contextual expansion [26]. The models and the best hyperparameter values achieved after tuning the models using TEDS (FD002) dataset and AEP dataset is presented in Table 1. The hyperparameters shown in the Table 1 are: number of hidden units (GRU/CiGRU), number of hidden units in fully connected layer of CiGRU with attention, sequence length, batch-size, and learning rate.

4.5 Results

Table 2 presents the results obtained from CiGRU + CxA and the baseline models with the test dataset of TEDS and AEP. For TEDS, the testing is performed for each engine unit separately and the average RMSE and average score is reported. It is apparent from Table 2 that CiGRU, CiGRU + A, and CiGRU + CxA performed similar in terms of RMSE with CiGRU + CxA model's RMSE marginally better. On the other hand, the scores of CiGRU + A and CiGRU + CxA are comparable and significantly better than CiGRU. So, CiGRU + CxA is able to lower the number of late predictions. Fig. 3 shows the predicted RUL values versus actual RUL values for a selected engine from the test data. It can be observed from the figure that for the constant part, the error is negative which is contributing towards low score. Similar trend is observed in other engines as well.

For AEP dataset, CiGRU + CxA performance is better than all other models in terms of MAE. Considering RMSE metric, CiGRU + CxA performed slightly better than CiGRU + A but significantly better than other models. Fig. 4 shows the predicted and actual Appliance energy usage for first 300 samples in the test data which is almost 2 days of data. It can be observed from the figure that the model can predict appliance energy but underestimates the peaks. One of the reasons could be that for certain days of the week the energy consumption

Table 2: Comparison of proposed model with baseline models

Dataset	TEDS-FD002		AEP	
Model	RMSE	Score	RMSE	MAE
GRU [9]	25.83	4122.89	75.81	38.42
CiGRU [9]	11.97	363.03	76.40	40.30
CiGRU + A	12.57	**299.75**	60.11	30.58
CiGRU + CxA (Proposed)	**11.80**	306.23	**59.11**	**26.55**

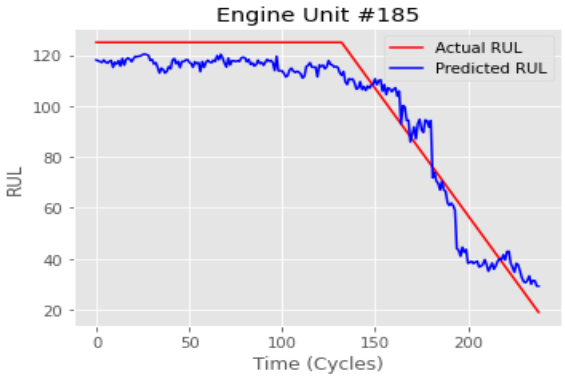

Fig. 3: Actual and Predicted RUL values.

is higher compared to the other days which is not captured by the model. Incorporating additional features such as day of the week and holidays potentially can improve the model. It is to be noted here, CiGRU + CxA is CiGRU + A and context in attention, CiGRU + A is CiGRU with attention, and CiGRU is GRU with context as separate input. The results show that adding context and attention to the baseline GRU provides a significant improvement in terms of given performance metrics.

A comparison of results achieved from CiGRU + CxA and results from state-of-the art deep learning models applied to TEDS dataset is presented in Table 3. The models that are selected for comparison are sequential models based on LSTM, sequential models with attention, and additionally CNN-based models are considered. The best values from the existing approaches and the values from the proposed model is highlighted in bold. It is evident from the table that CiGRU + CxA performed better compared to all other models both in terms of RMSE and score. The percentage decrease is 25.41% and 69.62% in RMSE and score, respectively.

Table 4 shows the comparison of CiGRU + CxA with existing approaches for AEP dataset. It is to be noted that there are several other approaches [30] that used the AEP dataset, however, only three approaches are compared here. The reason is that there is inconsistency in selection of test data in the existing approaches. The original paper [2] that published the dataset used 25% of the

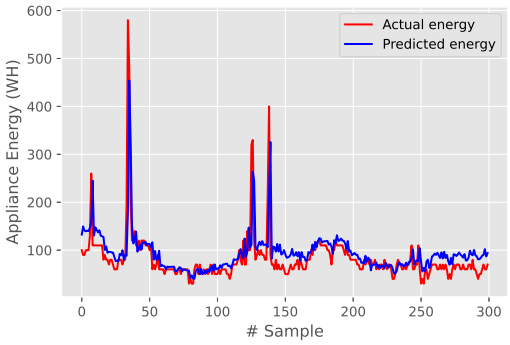

Fig. 4: Actual and Predicted values of Appliance energy usage.

Table 3: Comparison of proposed model with state-of-the-art -TEDS

Model	RMSE	Score
LSTM + FNN [32]	24.49	4,450.00
CNN + FNN [14]	22.36	10,412.00
RBM + LSTM [15]	22.73	3,366.00
LSTM + Attention [6]	17.65	2,102.00
MS-DCNN [23]	19.35	3,747.00
DA-CNN [24]	16.95	1,842.38
Attention Bi LSTM [21]	16.59	1,223.00
DA architecture [16]	17.08	1,575.00
Transformer Encoder + Attention [28]	**15.82**	**1,008.08**
CiGRU + CxA (this paper)	**11.80**	**306.23**

data as test set and showed that Gradient Boosting Machines (GBM) achieved the best results. Similarly, [31] used a 25% data as test set with XGBoost. Finally, [18] considered 20% data for testing with Adaptive Input Selection RNN (AIS-RNN). As shown in the Table 4, we tested CiGRU + CxA with two test sets one is 25% of available data and the other is 20% of the data for comparison. It is apparent from the results that CiGRU + CxA performed better or at par with existing approaches in terms of RMSE. However, the MEA is comparatively little higher than other models.

The experimental results show that RNN model with CiGRU and contextual attention performed significantly better than other models in presence of context, particularly in case of TEDS dataset where multiple operating conditions are explicitly present. Also, in comparison to other models, the proposed model achieved the given performance with relatively simpler model with 1 layer, 20 hidden units in RNN and 20 in FCL for TEDS and 1 layer 15 hidden units in RNN and 15 in FCL for AEP dataset. For TEDS dataset, the results are also influenced by selected features and normalization based on clustering. It is also worth mentioning here that each of the existing approaches had considered the

Table 4: Comparison of proposed model with state-of-the-art-AEP

Model	RMSE	MAE
GBM [2]	66.21	35.24
XGBoost[31]	59.69	**26.67**
CiGRU + CxA (this paper)	**58.82**	29.33
AIS-RNN [18]	59.81	**23.42**
CiGRU + CxA (this paper)	**59.11**	26.55

operating conditions (contextual features) in a different way. For example, Zheng et al. [32], in their approach, clustered the operating conditions and use one-hot encoding for their representation and then it is included as a primary feature. On the other hand, for AEP dataset, the existing approaches consider both the weather and indoor conditions as primary inputs.

Next, we analyse the attention weights and contextual weights. For RUL prediction model, Fig. 5 shows the attention weights for the same engine unit as in Fig. 3. It can be seen that prediction at time steps 25 to 190 mainly relied on early as well as recent time windows (5-15) whereas during the last time steps the network focuses at the last time window. In Fig. 6 the contextual weights ($\mathbf{B}^s$) associated with only one primary feature (demanded corrected fan speed) is shown which has mostly positive values. Similarly, two other features, for which heatmaps are not shown here, associated with fan speed have higher weights compared to other primary features. This indicates that the fan speed has more impact in prediction of RUL compared to other features. We performed similar analysis with AEP dataset. However, we are not providing the heatmaps for the weights due to space constraint. We observed that the heatmaps for the primary features are not significantly different except two heatmaps, temperature in the kitchen area and temperature in laundry room. In comparison to these two features, other features have more positive weights. Thus, the attention and contextual weights allow understanding the impact of features and time steps on the predicted output. However, as the size of this weight matrices grow the interpretation becomes challenging.

5 Conclusion and Future work

In this paper, we proposed a novel RNN architecture which has CiGRU as basic units and additionally incorporates the contextual attention mechanism. CiGRU allows integrating explicit contexts available from the problem domain and attention mechanism helps in retaining information from long sequences. Further, attention weights are learnt in a way that they are influenced by the context. The contextual weights in CiGRU and attention weights can be utilized for interpreting the model by visualising which feature and time steps are affecting the predictions. The experimental results with two benchmark datasets showed that the architecture achieves better performance or at par with the existing approaches. In future, we intend to perform more experiments and analysis with

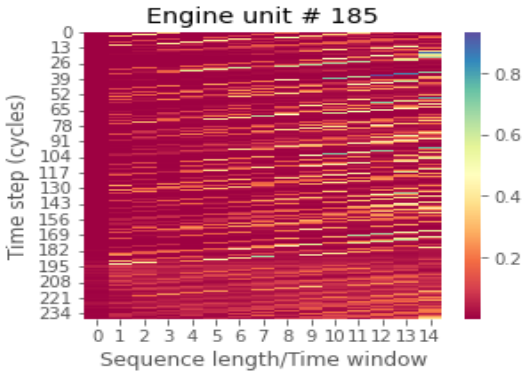

Fig. 5: Attention weights at each time step.

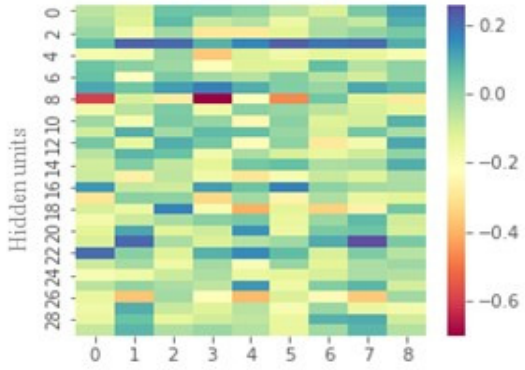

Fig. 6: Contextual input-hidden weights (part of matrix $\mathbf{B}^s$ associated with one input, *demanded corrected fan speed*, with dimension $n_h \times m$)

other benchmark datasets, and also apply it to the applications for smart environments where context data can be acquired. Recently, the success of transformers [27] in NLP and computer vision has attracted researchers and practitioners from time series community and there is a surge in transformer-based solutions for time series forecasting [29]. Investigating the pertinence of transformers to sensor MTS prediction and also the relevance of external contextual information for such models will further be considered in future.

References

1. Bahdanau, D., Cho, K., Bengio, Y.: Neural machine translation by jointly learning to align and translate (2014), https://arxiv.org/abs/1409.0473

2. Candanedo, L.M., Feldheim, V., Deramaix, D.: Data driven prediction models of energy use of appliances in a low-energy house. Energy and Buildings **140**, 81–97 (2017)
3. Cheng, Q., Chen, Y., Xiao, Y., Yin, H., Liu, W.: A dual-stage attention-based bi-lstm network for multivariate time series prediction. The Journal of Supercomputing **78**(14), 16214–16235 (2022)
4. Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., Bengio, Y.: Learning phrase representations using RNN encoder–decoder for statistical machine translation. In: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 1724–1734. Association for Computational Linguistics, Doha, Qatar (Oct 2014)
5. Cinar, Y., Mirisaee, H., Goswami, P., Gaussier, E., Aït-Bachir, A., Strijov, V.: Position-based content attention for time series forecasting with sequence-to-sequence RNNs. In: International Conference on Neural Information Processing. pp. 533–544 (10 2017). https://doi.org/10.1007/978-3-319-70139-4_54
6. da Costa, P., Akçay, A.E., Zhang, Y., Kaymak, U.: Attention and long short-term memory network for remaining useful lifetime predictions of turbofan engine degradation. IJPHM Special Issue on PHM Applications of Deep Learning and Emerging Analytics **10**(4), 1–12 (2019)
7. Du, S., Li, T., Yang, Y., Horng, S.J.: Multivariate time series forecasting via attention-based encoder–decoder framework. Neurocomputing **388**, 269–279 (2020)
8. Dutta Baruah, R., Muñoz Organero, M.: Integrating explicit contexts with recurrent neural networks for improving prognostic models. In: IEEE Aerospace Conference (2023), accepted.
9. Dutta Baruah, R., Organero, M.M.: Explicit context integrated recurrent neural network for sensor data applications (2023), https://arxiv.org/abs/2301.05031
10. Han, Z., Zhao, J., Leung, H., Ma, K.F., Wang, W.: A review of deep learning models for time series prediction. IEEE Sensors Journal **21**(6), 7833–7848 (2021)
11. Haruehansapong, K., Roungprom, W., Kliangkhlao, M., Yeranee, K., Sahoh, B.: Deep learning-driven automated fault detection and diagnostics based on a contextual environment: A case study of hvac system. Buildings **13**(1) (2023)
12. Heimes, F.O.: Recurrent neural networks for remaining useful life estimation. In: 2008 International Conference on Prognostics and Health Management. pp. 1–6 (2008)
13. Kinch, M.W., Melis, W.J., Keates, S.: The benefits of contextual information for speech recognition systems. In: 2018 10th Computer Science and Electronic Engineering (CEEC). pp. 225–230 (2018)
14. Li, X., Ding, Q., Sun, J.Q.: Remaining useful life estimation in prognostics using deep convolution neural networks. Reliability Engineering System Safety **172**, 1–11 (2018)
15. Listou Ellefsen, A., Bjørlykhaug, E., Æsøy, V., Ushakov, S., Zhang, H.: Remaining useful life predictions for turbofan engine degradation using semi-supervised deep architecture. Reliability Engineering System Safety **183**, 240–251 (2019)
16. Liu, L., Song, X., Zhou, Z.: Aircraft engine remaining useful life estimation via a double attention-based data-driven architecture. Reliability Engineering & System Safety **221**, 108330 (2022)
17. Luong, T., Pham, H., Manning, C.D.: Effective approaches to attention-based neural machine translation. In: Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing. pp. 1412–1421. Association for Computational Linguistics, Lisbon, Portugal (Sep 2015)

18. Munkhdalai, L., Munkhdalai, T., Park, K.H., Amarbayasgalan, T., Batbaatar, E., Park, H.W., Ryu, K.H.: An end-to-end adaptive input selection with dynamic weights for forecasting multivariate time series. IEEE Access **7**, 99099–99114 (2019). https://doi.org/10.1109/ACCESS.2019.2930069

19. Qin, Y., Song, D., Cheng, H., Cheng, W., Jiang, G., Cottrell, G.W.: A dual-stage attention-based recurrent neural network for time series prediction. In: Proceedings of the 26th International Joint Conference on Artificial Intelligence. p. 2627–2633. IJCAI'17, AAAI Press (2017)

20. Saxena, A., Goebel, K., Simon, D., Eklund, N.: Damage propagation modeling for aircraft engine run-to-failure simulation. In: 2008 International Conference on Prognostics and Health Management. pp. 1–9 (2008)

21. Shah, S.R.B., Chadha, G.S., Schwung, A., Ding, S.X.: A sequence-to-sequence approach for remaining useful lifetime estimation using attention-augmented bidirectional lstm. Intelligent Systems with Applications **10**, 200049 (2021)

22. Shih, S.Y., Sun, F.K., Lee, H.y.: Temporal pattern attention for multivariate time series forecasting. Machine Learning **108**(8), 1421–1441 (2019)

23. Song, Y., Gao, S., Li, Y., Jia, L., Li, Q., Pang, F.: Distributed attention-based temporal convolutional network for remaining useful life prediction. IEEE Internet of Things Journal **8**(12), 9594–9602 (2020)

24. Song, Y., Gao, S., Li, Y., Jia, L., Li, Q., Pang, F.: Distributed attention-based temporal convolutional network for remaining useful life prediction. IEEE Internet of Things Journal **8**(12), 9594–9602 (2020)

25. Sun, L., Zhong, Z., Zhang, C., Zhang, Y., Wu, D.: TESS: Multivariate sensor time series prediction for building sustainable smart cities. ACM Transactions on Sensor Networks (Dec 2022), just Accepted

26. Turney, P.D.: The management of context-sensitive features: A review of strategies (2002), https://arxiv.org/abs/cs/0212037

27. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L.u., Polosukhin, I.: Attention is all you need. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (eds.) Advances in Neural Information Processing Systems. vol. 30. Curran Associates, Inc. (2017)

28. Wang, X., Li, Y., Xu, Y., Liu, X., Zheng, T., Zheng, B.: Remaining useful life prediction for aero-engines using a time-enhanced multi-head self-attention model. Aerospace **10**(1) (2023)

29. Wen, Q., Zhou, T., Zhang, C., Chen, W., Ma, Z., Yan, J., Sun, L.: Transformers in time series: A survey (2023)

30. Yang, Y., Jinfu, F., Zhongjie, W., Zheng, Z., Yukun, X.: A dynamic ensemble method for residential short-term load forecasting. Alexandria Engineering Journal **63**, 75–88 (2023)

31. Zhang, T., Liao, L., Lai, H., Liu, J., Zou, F., Cai, Q.: Electrical energy prediction with regression-oriented models. In: Krömer, P., Zhang, H., Liang, Y., Pan, J.S. (eds.) Proceedings of the Fifth Euro-China Conference on Intelligent Data Analysis and Applications. pp. 146–154. Springer International Publishing, Cham (2019)

32. Zheng, S., Ristovski, K., Farahat, A., Gupta, C.: Long short-term memory network for remaining useful life estimation. In: 2017 IEEE International Conference on Prognostics and Health Management (ICPHM). pp. 88–95 (2017)

Ethical Statement

This research work does not involve any human subjects or personal information pertaining to them. It neither has potential policing or military use.