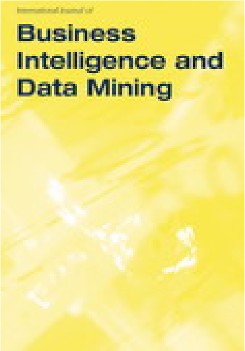


International Journal of Business Intelligence and Data Mining

ISSN online: 1743-8195 - ISSN print: 1743-8187
https://www.inderscience.com/ijbidm

An efficient missing value imputation and evaluation using GK-KH means and HTR-RNN

C.V.S.R. Syavasya, A. Lakshmi Muddana

DOI: 10.1504/IJBIDM.2023.10049909

Article History:

Received:	27 December 2021
Accepted:	08 February 2022
Published online:	01 December 2023

An efficient missing value imputation and evaluation using GK-KH means and HTR-RNN

C.V.S.R. Syavasya* and A. Lakshmi Muddana

Department of Computer Science and Engineering,
Gitam Deemed University,
Rudraram, Hyderabad 502329, India
Email: syavasyadba@gmail.com
Email: amuddana@gitam.edu
*Corresponding author

Abstract: The accuracy of the data mining (DM) outcomes might be affected by mining and analysing incomplete datasets with missing values (MV). Thus, a complete dataset is created by the imputation of MV, which makes the analysis easier. An effectual missing values imputation (MVI) is proposed and evaluated utilising Gaussian kernel-K harmonic means (GK-KH means) and hyperbolic tangent radial-recurrent neural networks (HTR-RNN) to combat this issue. At first, preprocessing is performed on the input data as of the CKD dataset wherein the duplicate form of the data gets eradicated. Next, the missing data are handled by ignoring them; and utilising GK-KH means, the MV is imputed. Next, the data are rationalised into a structured format. Then, SDRM-DHO selects the most optimal features as of the extracted features. Lastly, the HTR-RNN classifier accepts these chosen features as input. Proposed work performed well in more accurate missing value imputation.

Keywords: missing value imputation; K harmonic means; Gaussian kernel function; recurrent neural network; swap displacement reversion operation.

Reference to this paper should be made as follows: Syavasya, C.V.S.R. and Muddana, A.L. (2024) ‘An efficient missing value imputation and evaluation using GK-KH means and HTR-RNN’, *Int. J. Business Intelligence and Data Mining*, Vol. 24, No. 1, pp.25–46.

Biographical notes: C.V.S.R. Syavasya received his MTech in Software Engineering Specialisation from Gitam University, Visakhapatnam. He is currently pursuing his part-time PhD in field of Machine learning. His research area of interests include: machine learning, deep learning, and software cost estimation.

A. Lakshmi Muddana is a Professor from Department of Computer Science and Engineering from Gitam University, Hyderabad. Her research areas of interest include: machine learning and information security.

1 Introduction

Information mining procedures are the most normally utilised answer for dynamic frameworks (Andiojaya and Demirhan, 2019). To make information mining results more viable and significant, guarantee the nature of the gathered information. Yet, the gathered information generally contains some missing qualities or characteristics (Mostafa et al., 2020). While missing qualities happen for various reasons relying upon the information type, any surmising task on such a fragmented informational index should accept the missing passages into account to stay away from one-sided ends (De Bodt et al., 2019). The presence missing qualities can be more awful still; the low recurrence in information update gets somewhat when timestamps are missing under free or uncertain information duplicate capacities among sources. These three issues result in the low unwavering quality of information, which adds to the disarray in information applications. Consequently, the bad quality information might result in adverse consequence on many fields (Ding et al., 2020). Consequently, the missing qualities ought to be dealt with or attributed cautiously before the learning or mining process (Turabieh et al., 2019). There are two general ways to deal with manage the issue of missing qualities, one is the ignoration (expulsion) of information and another is the ascription (filled in) of missing qualities with new qualities (Ye et al., 2019; Ispirova et al., 2020). The main arrangement is material just when limited quantities of information are missed. The subsequent methodology is relevant for a lot of missing information (Gu et al., 2019). However, the greater part of the datasets contains somewhat bigger missing qualities; for this situation, there exists the need ascription process (Lai et al., 2019). Be that as it may, much of the time, the informational index credits are not autonomous of one another. Accordingly, the ID of connections among credits or the missing qualities is difficult to decide (Zhang et al., 2021).

To deal with this deficiency and work on the exhibition of missing worth attribution, different AI-based techniques are presented (Khan et al., 2021). These AI-based strategies are regularly administered and vigorously rely upon both the preparation datasets and they chose highlights (Sefidian and Negin, 2018). If the preparation dataset contains clearly data, the attribution yields might contain an unquestionable level of bowing. Since missing worth is normal in true information, we make conspiracy information ascription technique, use GAN construction to create 'counterfeit' dataset in light of 'genuine' dataset and the discriminator with 'genuine' dataset to gain proficiency with the relationship of the generator and the discriminator. After the preparation, we compute the MSE loss of the train dataset and the test dataset and the misfortune work is for the most part diminishing (Lam and Hsiao, 2019). To deal with these imperfections, the work has proposed a proficient missing worth ascription and assessment utilising GK-KH means and HTR-RNN, which productively play out the missing worth attribution process with restricted attribution time.

The remainder of the paper is showcased as follows: Section 2 reviews the related works with respect to the proposed technique, Section 3 clarifies the proposed strategy called missing worth ascription and assessment utilising GK-KH means and HTR-RNN division, and Section 4 outlines the outcomes and conversation for the proposed strategy dependent on execution measurements. At long last, Section 5 finishes up the paper with future work.

2 Literature survey

Migdady et al. (2020) fostered an exploratory test and carried out calculation that is a hereditary calculation with the Bayesian rule as a wellness capacity to direct the looking process in the arrangement space. After the preparation stage is stop what's more the calculation arrives at the union, the testing information vectors were introduced to the merged model, to such an extent that the worth of the third quality from every vector was concealed in request to really take a look at the productivity of the model to foresee the genuine worth of that characteristic in a particular vector. The exact outcomes show that the proposed strategy effectively credited 22% of the missing qualities though it neglected to gauge 78% of them.

Xu et al. (2018) presented the missing worth attribution calculation dependent on the proof chain (MIAEC), which first and foremost mined all the important proof of missing qualities in every information tuple, and afterward, this significant proof was joined and fabricated the proof chain for additional assessment of missing qualities. The MIAEC was reached out for huge scope information handling, in which the guide lessen model was utilised for the circulation and parallelisation of MIAEC. Exploratory outcomes showed that the created approach supported higher ascription exactness as contrasted and the other existing missing information attribution calculations. Nonetheless, the plan neglected to consider the affecting elements of the transient and spatial data.

Li et al. (2018) presented a multi-view learning strategy for the assessment of missing qualities for traffic-related time series information. The strategy comprised of three generally utilised models, for example, long-momentary memory (LSTM), support vector relapse (SVR), and synergistic separating (CF) procedures. These three models thought about the neighbourhood and worldwide variety in transient and spatial perspectives and caught additional huge data from the current information. The assessments of missing qualities from four perspectives were amassed together and acquired a last worth with a part work. The test investigation expressed that the technique gave strong outcomes diverse missing proportions and outflanked different baselines, particularly for block missing examples with a high missing proportion. Notwithstanding, the plan had not saved the connection between the factors.

Abu-Soud (2019) proposed a novel technique for controlling missing characteristics that was developed in collaboration with ILA and used during the acceptance cycle. ILA4 is the name of the proposed framework. ILA4 has been tested on a number of datasets with varying levels of missing quality. Its findings were also compared to a few common approaches to correcting missing attributes. The results reveal that the ILA4 outcomes appear to be equal to the best examples of a few other prominent strategies for dealing with missing characteristics issues, namely, the most commonly recognised worth, the most widely recognised worth restricted to a concept, and the erase system. The experiments have shown that the proposed system has better results on number and simplicity of the generated rules, execution time, and the induction power than the results obtained by applying these three methods in most cases.

Tsai et al. (2018) introduced a class place based missing worth ascription (CCMVI) approach, which delivered sensibly great attribution results with the restricted ascription time. The CCMVI approach was fundamentally made out of two modules. The first centred around the distinguishing proof of attribution edge dependent on the distances between the class places and their comparing information tests, while the subsequent one utilised the recognised limit for missing worth ascription. The trial results showed that the CCMVI approach beat the other MVI approaches for both mathematical and blended datasets. What's more, it required substantially less attribution time than the AI MVI techniques. However, the plan was more fitting for little datasets.

Hosseinzadeh et al. (2021) developed a model for CKD prediction and early CKD severity level determination to aid clinicians and complex clinical groups in certain countries. The advantages of sensor innovation in the IoT stage were used to develop a model. Impacting imperative signs and clinical indications, which were derived from CKD expectation and doctors' clinical perceptions, were included in a list of capabilities with affecting CKD boundaries. In addition to the CKD severity level, the model also predicted the CKD severity level. The model had five stages: IoT data collection, information pre-handling, highlight selection, CKD order, and outcomes analysis. The results revealed that the used dataset with the selected highlights produced 97% precision, nearly 100% affectability and 95% recall. The explicitness by means of applying choice tree (J48) classifier in contrast with help vector machine (SVM), multi-facet discernment (MLP), and Guileless Bayes classifiers.

Ma et al. (2020) introduced a heterogeneous altered fake neural organisation (HMANN) for the early recognition, division, and conclusion of persistent renal disappointment on the web of clinical things (IoMT) stage. Moreover, the HMANN was named a help vector machine and multi-facet perceptrons (MLP) with a back propagation (BP) calculation. The calculation worked dependent on an ultrasound picture, which was indicated as a pre-handling step and the area of kidney interest was sectioned in the ultrasound picture. In kidney division, the HMANN strategy accomplished high precision and essentially diminishing an opportunity to outline the shape.

Arulanthu and Perumal (2020) introduced a web-based clinical choice emotionally supportive network (OMDSS) for CKD expectation. The introduced model included a bunch of stages, in particular information gathering, pre-handling, and characterisation of clinical information for the expectation of CKD. For characterisation, the strategic relapse (LR) model was applied for arranging the information occasions into CKD and non-CKD. Moreover, for tuning the boundaries of LR, versatile second assessment (Adam), and versatile learning rate streamlining calculations were applied. The exhibition of the model was analysed utilising a benchmark CKD dataset. The result noticed the unrivaled qualities of the introduced model on the applied dataset.

Abdelaziz et al. (2018) introduced a half breed clever model for foreseeing CKD-based cloud-IoT by utilising two savvy procedures, which were direct relapse (LR) and neural organisation (NN). LR was utilised to decide basic factors that impact CKD. NN was utilised to anticipate CKD. The outcomes showed that the precision of the mixture shrewd model in foreseeing CKD was 97.8%. What's more, a mixture clever model was applied on windows sky blue to act as an illustration of a distributed computing climate to anticipate CKD for supporting patients in brilliant urban communities. The model was better than the majority of the models alluded to in the connected works by 64%.

Khamparia et al. (2020) proposed a major learning framework for CKD representation based on a stacked autoencoder model using blended media data and a softmax classifier. After removing the supporting components from the dataset with the stacked autoencoder, a softmax classifier was used to predict the final class. It had researched the UCI dataset, which contained starting periods of 400 CKD patients with 25 credits, which was an equal request issue. Precision, survey, expresses, and F1-score were used as evaluation estimations for the examination of the proposed network. It was seen that this multimodal model outflanked the other customary classifiers utilised for CKD with a grouping precision of 100%.

Sarkar and Sinhababu (2019) cultivated a framework that can detect and group neighbouring tweets regardless of the drug utilised. Changing all tweets over to English, we discard the 'script versus language' issue. The new procedure we arranged involves Content ID, Language examination, and grouped mining. Considering English and the super two Indian vernaculars, we saw that the proposed structure gives favoured exactness over the prevalent systems. Online data recuperated from electronic media, for instance, Twitter has offered monstrous potential for getting critical information that can be used to assess and handle various huge real issues. In this paper, we contemplated simply Twitter data.

This paper employs an intriguing chrominance surface model (DiCTP) technique for strong part extraction (Betty et al., 2020). Instances of two categories are supplied from the image's buried channel information, which removes the data's shaded surface information. Surprising data is generated from the image's RGB and BRG planes, conveying a piece of extended chromatic component vectors. The LBP code (nearby twofold model) is constructed and added close to the component vector, which aids in displaying the image's greyscale information. The preliminary results are sorted using the CASIA Face Picture Informational Index Interpretation 5 (DB1) and Indian Face Informational Collection (DB2), which provide significant improvements over the existing framework. The additional effect on the part vector is added by the three models in progression which is added upon different sections and the effect of the isolated part is imagined most noteworthy with the extension of pad regard. The test keeps an eye on DB1 and DB2 shows ideal updates in results when differentiated and the state of workmanship techniques (Arunkumar and Kannimuthu, 2020).

This paper features the significance of information stream mining and investigates two significant open-source systems, specifically huge web-based examination (MOA) and versatile progressed huge on the web investigation (SAMOA). The ramifications of both the devices foreshadows well for additional consultations in huge information research local area. Business data framework (BIS) models can arrive at exceptional statures with the multiplication of these business examination apparatuses. Real-time analytics offers 'nearzero' latency solutions and hence such stream analytics can stop the effects such as stock exchange collapse and defects in manufacturing sector. Also, routine business operations can be monitored effectively.

3 Problem statement

Activation functions like sigmoid or softmax functions have drawback of vanishing gradient that usually occurs in the back-propagation process, which results in the considerable slowdown of learning and also degrades the classification performance (Agarwal, 2019).

Existing missing value imputation algorithms like KMeans, KMediod, FCM, etc. has its limitation of accurate imputation because, if the training dataset contains noisy data, the imputation outputs may contain a high level of distortion. If the training dataset contains the missing data, then the prediction of missing values (MV) is done on the basis of the incomplete dataset. Also, these hybridised techniques are usually more computationally expensive and consume an excessive amount of time for execution (Emmanuel et al., 2021).

Some drawbacks are encompassed by the prevailing research methodologies that are described below:

- Incorrect imputation on the streaming data is brought about by the lack of online parametrical as well as structural adaptation.
- The evolving models-based imputation brings about the increased time consumption when there is the arrival of the data with the quick changes over the resource-constrained devices.

Predicting the disease at an earlier stage is a vital task. Nevertheless, it is too difficult for the healthcare management system to accurately predict centred on symptoms. The data generated as of connected devices is augmenting at a high rate; however, most big data systems' storage capacity is confined. Storing and managing a large extent of data has become a considerable challenge.

4 Research methodology

The main goal of this paper is to predict whether the patient is having chronic kidney disease (CKD) or not by efficiently performing the missing value imputation using K harmonic mean and HTR-RNN. We have taken the CKD Dataset and 80% of data has been trained and 20% of data has been tested. Initially, we will perform data pre-processing following the steps:

Step 1 In this step, duplicated data will be removed by using the block level data deduplication technique.

Step 2 Next, the de-duplicated data will be sent to the missing data imputation process. This phase contains two main steps. One is to ignore the data, (i.e.,) when the data has quite large and multiple MV, in which case, the whole data or row will be ignored.

Then, in the second phase, the missing value will be imputed by using the Gaussian kernel – K harmonic means (GK-KH means) algorithm. In this, the Gaussian kernel activation function will be included in the existing K harmonic means (GK-KH means) algorithm to improve the clustering accuracy.

In this step, the proposed algorithm will be compared with existing KMeans clustering, partition clustering, and KMediod clustering with result parameters, such clustering time, etc.

Step 3 In this step, the data transferring will be done by using: numeralisation and normalisation that is done by using the log scaling normalisation technique.

Step 4 Once the features are extracted from the structured data, then the optimised features are selected by using swap, displacement, and reversion based mean deer hunting algorithm (SDRM-DHO).

To evaluate the performance of this step, we will compare the results of the proposed algorithm with the existing crow search algorithm, dragonfly algorithm, mayfly algorithm, and deer hunting algorithm with result parameters, such selection time, etc.

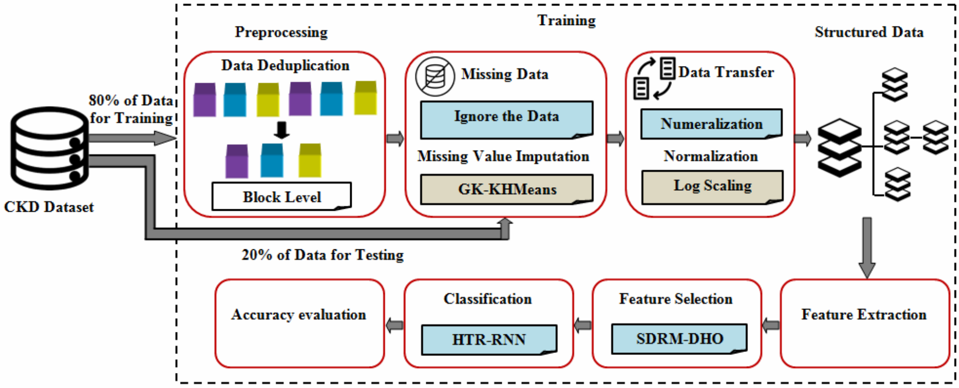
Step 5 Finally, the selected features are trained by using hyperbolic tangent radial – recurrent neural networks (HTR-RNN) algorithm. In this, modified activation will be used in the recurrent neural network algorithm to reduce memory usage and training time.

The proposed algorithm is compared with existing convolutional neural network deep Boltzmann machine, deep belief networks, and recurrent neural networks with result parameters, such as precision, recall, training time, memory usage and accuracy.

5 Proposed methodology

The difficulty of DM and other big data analysis applications is augmented by the presence of MV on incomplete datasets. Therefore, the MV should be handled carefully before the mining or learning process. So, the work has proposed an efficient missing value imputation and evaluation using GK-KH means and HTR-RNN. Initially, the input data are subjected into the pre-processing function, in which the duplicated data are removed, and then the missing data are handled efficiently by ignoring the data and missing value imputation. Once the missing value gets imputed, the data transformation process is taken place, in which the data gets numeralised and normalised. Then the transformed data are converted into a structured format. From the structured data, the most important features are extracted. Thereafter, the most optimal features are selected from the extracted features. Finally, these selected are given to the classifier called HTR-RNN. This classification phase efficiently evaluates the accuracy of the missing value imputation process. The proposed framework's architecture is exhibited in Figure 1.

Figure 1 Architecture of the proposed framework (see online version for colours)



5.1 Input source

In the proposed work, the input data are taken from the CKD dataset, which is publically available on the internet. The CKD dataset was gathered from 400 patients and includes 24 highlights that are partitioned into 11 numeric elements and 13 clear cut elements.

Here, 80% of the information is surrendered to the preparation stage and 20% of the information is surrendered to the testing stage.

5.2 Pre-processing

Initially, the input data are subjected to the pre-processing function. The pre-processing step transforms the raw input data into a predictable and analysable format by neglecting unwanted parts of the data. Furthermore, it also improves the quality of the text. So that, the machine can understands the text data easily. In the proposed work, the pre-processing step concentrated on data deduplication. The pre-processing function is mathematically derived as.

$$P_r = \alpha_{pr}\left(In_t\right) \tag{1}$$

where P_r is the output of pre-processing function, In_t is the input data, α_{pr} is the pre-processing function and the pre-processing function is expressed by.

$$\alpha_{pr} = \left\{\alpha_{Dp}\right\} \tag{2}$$

where α_{Dp} denotes the data deduplication function

5.2.1 Data deduplication

Data deduplication is the process that eliminates duplicate data and avoids the repetition of the same data stored in multiple locations. Therefore, the training time, as well as the execution time is considerably reduced. In this work, the duplicated data are removed by using the Block Level data deduplication technique. The relation for the data deduplication α_{Dp} is given as follows.

$$\alpha_{Dp} = \left\{B_l(d)\right\} \tag{3}$$

where $B_l(d)$ indicates the block-level data deduplication function.

5.2.1.1 Block level data deduplication technique

In the block-level data deduplication technique, the whole data are divided into multiple numbers of blocks. Then, each block is compared with another block and removes the duplicate or repeated data efficiently. Hence, the computation burden of the processing of whole data can be drastically reduced by using the block-level data deduplication technique. The expression for this function is given by.

$$\lambda_{Op}^{B} = \left[B_l(d)\left\{In_t\right\}\right] \tag{4}$$

where λ_{Op}^{B} denotes the outcome of block Level data deduplication technique, and $[B_l(d)\{In_t\}]$ indicates the block-level data deduplication function of the input data. Thus, the relation for the pre-processed data P_r is given by.

$$P_r = \left\{d_1, d_2, d_3, \ldots\ldots, d_n\right\} \tag{5}$$

where $\{d_1, d_2, d_3, \ldots\ldots, d_n\}$ represents the number of deduplicated data.

5.3 Missing data imputation process

Once the data gets pre-processed, the missing data imputation process is carried out. The missing data imputation process checks whether the data are missed or not. If any of the data are missed, then such data are replaced or rearranged. This phase contains two important steps namely ignoring the data and missing value imputation. The mathematical representation of the missing data imputation process is given by.

$$M_{ip} = \{ig_f, mv_f\} \tag{6}$$

where M_{ip} denotes the missing data imputation process, ig_f and mv_f indicates ignore data function and missing value imputation function respectively.

5.3.1 Ignore the data

When the large numbers of data or multiple values are missed from the single row or column, then such a whole row or column gets ignored. This function is mathematically defined as follows.

$$\lambda_{IR} = ig_f\{P_r\} \tag{7}$$

where λ_{IR} denotes the outcome of ignoring of data phase, and $ig_f\{P_r\}$ denotes the ignore data function of the pre-processed data.

5.3.2 Missing value imputation

Missing value imputation is converse to that of the data ignoration step. In the missing value imputation process, the values which are missed are replaced by the relevant data. These relevant data are obtained on the basis of the nearby data. In order to perform the imputation process more precisely, the work uses a technique called GK-KH means. In KH means, the quality of the weight values of each data point is determined by using the Gaussian kernel function. In doing so, the data points with better weight values are obtained. These obtained data points are considered as the MV. Hence, the GK-KH means accurately imputes the MV. The steps which are involved in the GK-KH Means are explained as follows.

5.3.2.1 Missing value imputation using GK-KH means

The K symphonious means (KHM) is like that of the K means calculation. The KHM is likewise a middle-based parcel bunching calculation that haphazardly chooses introductory centroids initially. The significant distinction among KHM and KM is that KHM utilises consonant midpoints of the good ways from every information highlight the focuses as parts of its exhibition work. The algorithmic strides of the GK-KH Means are clarified as follows.

Step 1 Select the k initial cluster centres randomly from the n number of data points. Thus, the selected centroids and the data points are formulated as follows.

$$Ct = \{Ct_1, Ct_2, Ct_3, ..., Ct_k\} \tag{8}$$

$$Y = \{y_1, y_2, y_3, \text{........}, y_n\} \tag{9}$$

Step 2 Calculate the objective function value as follows.

$$khm(Y, Ct) = \sum_{i=1}^{n} \frac{k}{\sum_{j=1}^{k} \frac{1}{\left\|y_i - Ct_j\right\|^p}} \tag{10}$$

Step 3 Calculate the membership of each data point y_i to centroids Ct_i for $i = 1, 2, 3, \ldots\ldots, n$, and $j = 1, 2, 3, \ldots\ldots, k$ and by using the following relation.

$$M\left(Ct_j, y_i\right) = \frac{\left\|y_i - Ct_j\right\|^{-p-2}}{\sum_{i=1}^{k} \left\|y_i - Ct_j\right\|^{-p-2}} \tag{11}$$

Step 4 Calculate the weight of each data point y_i for $i = 1, 2, 3, \ldots\ldots, n$ by using the following relation.

$$w\left(y_i\right) = \frac{\sum_{j=1}^{k} \left\|y_i - Ct_j\right\|^{-p-2}}{\left(\sum_{j=1}^{k} \left\|y_i - Ct_j\right\|^{-p}\right)^2} \bullet \frac{1}{\sqrt{2\pi}} \exp\left[-\frac{Ct_j^2}{2}\right] \tag{12}$$

Step 5 Calculate the new centroid Ct_i for $j = 1, 2, 3, \ldots\ldots, k$ by using the below mentioned equation.

$$Ct_j = \frac{\sum_{i=1}^{n} M\left(Ct_j, y_i\right) w\left(y_i\right) y_i}{\sum_{i=1}^{n} M\left(Ct, y_i\right) w\left(y_i\right)} \tag{13}$$

Step 6 Repeat the step 2 and 4 until the stopping criteria is satisfied, and the final outcome of the missing value imputation function mv_f is represented as follows.

$$mv_f = \left\{im\left(d_1\right), im\left(d_2\right), im\left(d_3\right), \text{........}, im\left(d_n\right)\right\} \tag{14}$$

where $\{im(d_1), im(d_2), im(d_3), \ldots\ldots, im(d_n)\}$ describes the numbers of imputed data.

5.4 Data transformation

After the missing data imputation process, the data transformation phase is implemented. Data transformation is the process in which data gets converted from one format into another format. These transformed formats of data are more convenient for the machine learning process and also increase the efficiency of the analytic process. In the proposed work, the numeralisation and normalisation are the two important steps in the data transformation phase and that are mathematically generated as.

$$D_{dt} = \left\{Nu_f, Nom_f\right\} \tag{15}$$

where D_{dt} signifies the data transformation function, Nu_f denotes the numeralisation function and Nom_f denotes the normalisation function.

5.4.1 Numeralisation

Numeralisation is the process that is used to transform the string values or characters into a numerical format. The numeralisation function is formulated as follows.

$$N(d) = Nu_f[im(d)] \tag{16}$$

where $N(d)$ represents the numeralised data, and $Nu_f[im(d)]$ denotes the numeralisation function on the imputed data.

5.4.2 Normalisation

The normalisation method is used to organise or scale the data by adjusting the data values into a specific range, such as between 0 to 1. Thus, the normalisation technique is used for more efficient access to data. In the proposed work, the data are normalised by using the log scaling technique. Mathematically, the log scaling normalisation is represented as.

$$Norm(im(d)) = Log(im(d)) \tag{17}$$

where $im(d)$ denotes the imputed data and $Norm(im(d))$ signifies the normalised data.

5.5 Structured data

Finally, the outcome from the data transformation phase is again rationalised into a structured format. This structured format provides a well-defined structure to the data. Therefore, this type of structured data improves classification accuracy.

5.6 Feature extraction

From the structured form of data, the most important and information-rich features are extracted out. The main motive of the feature extraction process is to reduce the number of features and form a new set of features. These new sets of reduced features summarised most of the information contained in the original set of features. Mathematically, the feature extraction X_F process is given by.

$$X_F = \{X_1, X_2, X_3, ..., X_n\} \tag{18}$$

where $X_1, X_2, X_3, ..., X_n$ denotes the number of extracted features.

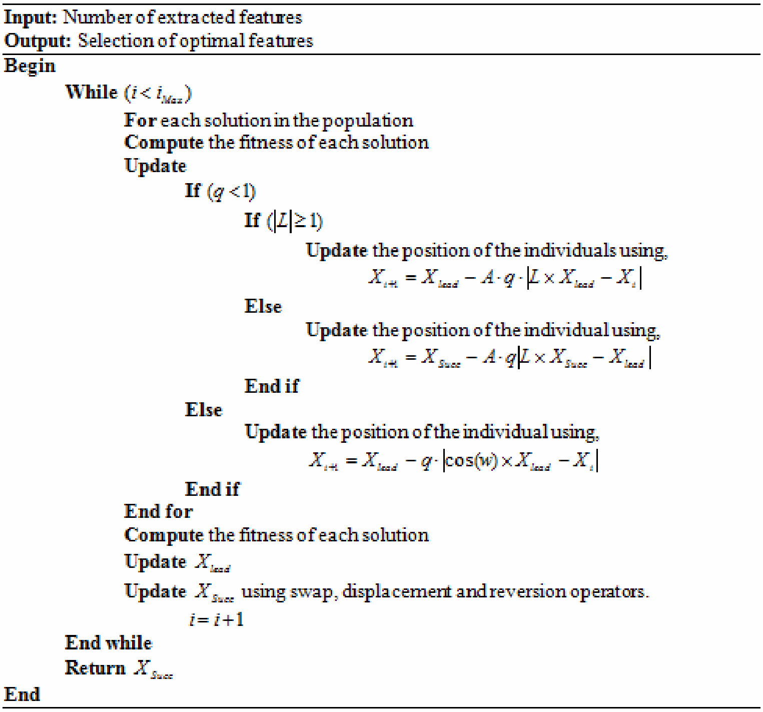
5.7 Feature selection

Thereafter, as of the extracted features, the most appropriate as well as required numbers of features are chosen. SDRM-DHO takes care of this section. The computational intricacy is mitigated as well as the classification's performance is improved with limited time together with cost by the SDRM-DHO.

5.7.1 Feature selection using SDRM-DHO

The deer hunting optimisation (DHO) algorithm is enthused via the hunting behaviour of the humans in the direction of the deer. It is basically a novel meta-heuristic algorithm. Centred on some strategies, the hunters encircle a deer and move in the direction of it to hunt. Disparate parameters say the wind angle, deer position, et cetera were included. Another imperative criterion is the cooperation amongst the hunters, which makes hunting effectual. Lastly, centred on the leader and successor's position, they reach the target. The swap, displacement, and reversion (SDM) operators are utilised for choosing the successor position in a more suitable way. This modification aids in finding the optimal outcomes in a better way. Figure 2 exhibits the SDRM-DHO pseudo-code. And the SDRM-DHO's algorithmic steps are elucidated as:

Figure 2 Pseudo-code for SDRM-DHO algorithm



```
Input: Number of extracted features
Output: Selection of optimal features
Begin
    While ($i < i_{Max}$)
        For each solution in the population
        Compute the fitness of each solution
        Update
            If ($q < 1$)
                If ($|L| \geq 1$)
                    Update the position of the individuals using,
                        $X_{i+1} = X_{lead} - A \cdot q \cdot |L \times X_{lead} - X_i|$
                Else
                    Update the position of the individual using,
                        $X_{i+1} = X_{Succ} - A \cdot q|L \times X_{Succ} - X_{lead}|$
                End if
            Else
                Update the position of the individual using,
                    $X_{i+1} = X_{lead} - q \cdot |\cos(w) \times X_{lead} - X_i|$
            End if
        End for
        Compute the fitness of each solution
        Update $X_{lead}$
        Update $X_{Succ}$ using swap, displacement and reversion operators.
            $i = i + 1$
    End while
    Return $X_{Succ}$
End
```

5.7.1.1 Population initialisation

The initialisation of the populace of hunters is the initial step. The total extracted features are stated by the hunters' population, which is signified as:

$$X = \{X_1, X_2, X_3, ..., X_n\}; \quad 1 < i \leq n \tag{19}$$

wherein *n* signifies the total number of hunters, which are the solutions, on the populace *X*.

5.7.1.2 Parametric instatement

Here, the wind angle in addition to the deer's position angle is initialised. These are the most imperative parameters on the determination of the hunters' best positions. Here, the search space is regarded as a circle. The wind angle θ follows the circumference of a circle, which is rendered as:

$$\theta_i = 2\pi r \tag{20}$$

wherein r signifies the random number with a value in the gamut [0, 1] and i signifies the current iteration. In the interim, the deer's position angle φ_i is rendered as:

$$\varphi_i = \theta + \pi \tag{21}$$

5.7.1.3 Position propagation

In an initial iteration, finding the best solution intended for the algorithm is typically impossible. However, the best integer is regarded as the optimal solution subsequent to generating an arbitrary integer and evaluating the cost function as of it. Here, '2' parameters are regarded:

1 leader position – which is the initial best location of the hunter

2 successor position – which signifies the subsequent hunter position.

5.7.1.3.1 Propagation through a leader's position

The position updation process begins subsequent to defining the best positions since each individual on the populace endeavours to achieve the best position. Therefore, the 'encircling behaviour' is formulated mathematically as:

$$X_{i+1} = X_{lead} - A \cdot q \cdot \left| L \times X_{lead} - X_i \right| \tag{22}$$

wherein X_i signifies the position at the current iteration, X_{i+1} implies the position at next iteration, A and L implies the coefficient vectors, and q signifies an arbitrary number developed via considering the wind speed, whose values are gamut as of 0 to 2. The coefficient vectors are gauged as:

$$A = \frac{1}{4} \log\left(i + \frac{1}{i_{Max}} \right) a \tag{23}$$

$$L = 2 \cdot c \tag{24}$$

wherein i_{Max} signifies the maximum iteration, a implies the parameter that encompasses the value betwixt –1 and 1, and c signifies an arbitrary number that exists within the interval [0, 1]. (X, Y) represents the initial position of the hunter, which gets updated grounded on the prey's position. For attaining the best position (X_{best}, Y_{best}), the '2' coefficient vectors A and L are adjusted. The position updation process takes place if the value of $q < 1$. It signifies that the hunter can arbitrarily move in varied directions devoid of regarding the position angle.

5.7.1.3.2 Propagation through position angle

The conception is extended via regarding the position angle in the update rule for ameliorating the search space. For ascertaining the hunter's position, the angle calculation is necessary so that the prey is oblivious of the attack, which makes the hunting process effectual. The visualisation angle of the deer or the prey u_i is gauged as.

$$u_i = \frac{\pi}{8} \times r \tag{25}$$

A parameter v_i is computed those aids in updating the position angle grounded on the difference betwixt the deer's wind angle and the visual angle.

$$v_i = \theta_i - u_i \tag{26}$$

wherein θ signifies the wind angle. Next, the position angle is updated for the subsequent iteration as:

$$\varphi_{i+1} = \varphi_i + v_i \tag{27}$$

By regarding the position angle, the position updation is ascertained as:

$$X_{i+1} = X_{lead} - q \cdot \left| \cos(w) \times X_{lead} - X_i \right| \tag{28}$$

wherein $B = \varphi_{i+1}$, $X_{best(i)}$ signifies the best position and q implies the arbitrary number. The individual position is found to be exceptionally contrary to the position angle. Thus, the prey is not aware of the hunter.

5.7.1.3.3 Propagation through the position of the successor

At the hour of investigation, the vector is presented inside the circling conduct. At first, the irregular pursuit process is completed by considering the worth of as under 1. At last, the position updation process happens based on a replacement position rather than thinking about the best position. To advance the replacement position, the trade, removal, and inversion activity is utilised. In SDM, the trade administrator trades the upsides of two irregular focuses, the inversion administrator switches an arbitrary position, and the inclusion administrator communicates the worth of an arbitrary highlight another point. Then, at that point, the worldwide inquiry is performed utilising.

$$X_{i+1} = X_{Succ} - A \cdot q \left| L \times X_{Succ} - X_{lead} \right| \tag{29}$$

5.7.1.4 Termination

The position updation process is proceeded until the best position is gotten (for example halting standards). This got best arrangement gives the ideal elements; hence, the grouping system is acted in an exact way with less blunder and intricacy. The numerical portrayal of the chose highlights is given by.

$$S(x) = \{x_1, x_2, x_3, ..., x_n\} \tag{30}$$

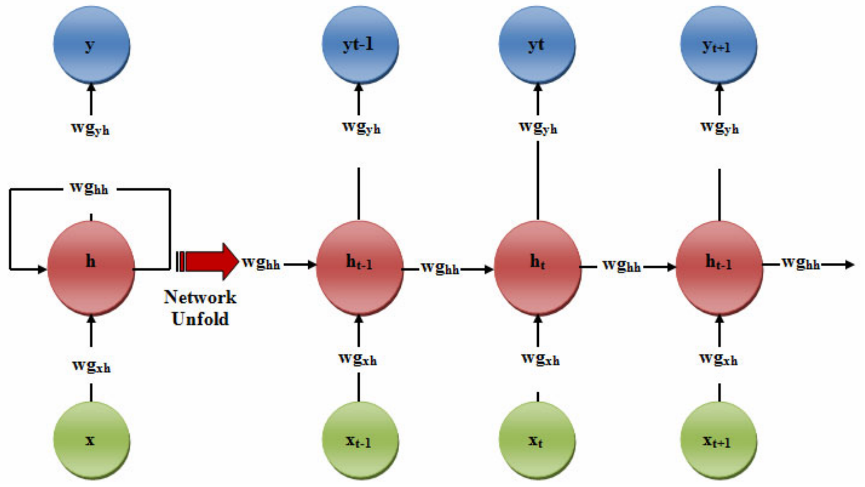
where $\{x_1, x_2, x_3, ..., x_n\}$ denotes the number of optimal features.

5.8 Classification

Finally, the selected features are given as the input to the classifier called hyperbolic tangent radial – recurrent neural networks (HTR-RNN), which efficiently determines whether the MV are imputed accurately. The conventional RNN uses the sigmoid or softmax activation function for classification. But these activation functions have the drawback of vanishing gradient that usually occurs in the back-propagation process, which results in the considerable slowdown of learning and also degrades the classification performance. To overcome such drawbacks, the hyperbolic tangent radial (HTR) function is incorporated with the RNN algorithm. The HTR function overcomes the aforementioned problems, and it is very suitable for deeper networks. Moreover, it also reduces the error rate, so that the optimal result is found with minimal iterations. Thus, the HTR-RNN achieves better classification results with limited time.

An RNN is the sort of NN. The output as of the preceding step is inputted to the subsequent step. The hidden state is basically the key and most vital feature of RNN. It remembers some information concerning a sequence. Every hidden layer (HL) encompasses its own set of weights and biases in conventional NN. For an instance, the weights as well as biases are (W_1, B_1) in HL 1, the weights as well as biases are (W_2, B_2) for the '2nd' HL, together with (W_3, B_3) for the '3rd' HL. Each of these layers is autonomous of one another; therefore, they do not memorise the preceding outputs. However, via rendering the same weights and biases to every layer, the RNN converts the autonomous activations into dependent ones. Thus, it brings about the diminution of the intricacy of augmenting parameters and memorising every previous output via rendering every output as input to the succeeding HL. The RNN's general structure is depicted in Figure 3.

Figure 3 General structure of the RNN classifier (see online version for colours)



The procedures of HTR-RNN are rendered as:

Step 1 The HTR-RNN is done on the input data $x_{IN} = \{x_1, x_2, x_3, \ldots\ldots, x_t\}$ that comprises a hidden vector sequence $h_{Lyr} = \{h_1, h_2, h_3, \ldots\ldots, h_t\}$ along with the output vector sequence $y_{out} = \{y_1, y_2, y_3, \ldots\ldots, y_t\}$ via iterating the following sequence as of $t = 1$ to T. The HL is gauged as:

$$h_{Lyr} = \Im_{act}\left[wg_{xh}x_t + wg_{hh}h_{t-1} + Bais\right] \tag{31}$$

wherein wg signifies the weight matrices (e.g., wg_{xh} is the input-hidden weight value as well as wg_{hh} is the HL weight value), the $Bais$ implies the bias vectors and $\Im_{act}$ signifies the HTR function. The expression for the HTR function is provided by:

$$\Im_{act} = \sum_{i=1}^{n} \frac{\left\|\varepsilon^{xi} - \varepsilon^{-x^i}\right\|}{\left\|\varepsilon^{x^i} + \varepsilon^{-x^i}\right\|} \tag{32}$$

where || || describes the radial function.

Step 2 Then, the output layer is responsible to determine whether the model accurately performs the missing value imputation process. The output layer was activated by the sigmoid activation function (σ_S) and the output layer can be determined by.

$$y_{out} = \sigma_S\left[wg_{xy}h_t + Bais\right] \tag{33}$$

$$x = wg_{xy}h_t + Bais \tag{34}$$

Step 3 The sigmoid activation function is calculated by using the following relation.

$$\sigma_S(x) = \frac{1}{1+\varepsilon^{-\alpha}} \tag{35}$$

Step 4 Then, via computing the difference betwixt the actual value x_a and the predicted value $\hat{x}_p$, the loss value is estimated. The error value can well be gauged as.

$$Loss = \left(x_a - \hat{x}_p\right)^2 \tag{36}$$

If the loss value or the error value of the model is zero ($Loss = 0$), this indicates that the model accurately performed the missed value imputation process. In case, the loss value $Loss \neq 0$ then the back-propagation is taken place by updating the weight values.

Finally, the HTR-RNN classifier efficiently determines whether the model performs the missing value imputation process correctly.

6 Result and discussion

The detailed analysis of the final outcome of the proposed framework is explained in this section. The performance analysis (PA) and the comparative analysis are carried out for proving the work's effectiveness. The implementation of the proposed methodology is done by using PYTHON, and the data are obtained from the CKD dataset, which is publically available on the internet.

6.1 PA of the proposed GK-KH means

PA of the proposed GK-KH means is validated with respect to missing value imputation time and the obtained results are compared with various existing techniques, such as K-means, clustering large application (CLARA), K-medoid, and fuzzy C-means (FCM) in order to prove its effectiveness.

Figure 4 Comparative analysis of the proposed GK-KH means in terms of missing value imputation time (see online version for colours)

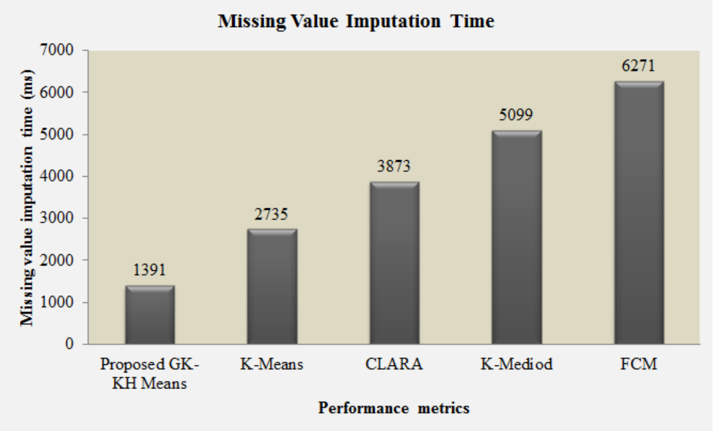


Figure 4 compares the missing value imputation time of the proposed work with the existing works. The graphical analysis states that the proposed GK-KH means requires 1,391 ms to complete the missing value imputation process. But for the completion of the missing value imputation process, the existing work like K-means, CLARA, K-medoid, and FCM requires 2,735 ms, 3,873 ms, 5,099 ms, and 6,271 ms respectively. Hence the time taken by the proposed work is literally low as compared to the existing works. Thus, the comparative analysis concludes that the proposed work consumes a very little amount of time and also imputes the data more precisely and efficiently.

6.2 PA of the proposed HTR-RNN

Concerning disparate performance metrics, say sensitivity, accuracy, specificity, precision, recall, CPU memory usage, F-measure, in addition to training time, the HTR-RNN is analogised to RNN, deep neural networks (DNN), deep belief networks (DBN), together with convolution neural networks (CNN) to state the model's worthiness.

The PA of the HTR-RNN with RNN, DBN, DNN, together with CNN concerning the accuracy, sensitivity, as well as specificity is illustrated in Table 1. The proposed one achieves 97.56% of accuracy, 97.57% of sensitivity, together with 97.54% of specificity. Nevertheless, the prevailing works attain the accuracy rate of those overall ranges betwixt 90.65%–95.76%, sensitivity rate that ranges betwixt 90.65%–95.84%, together with specificity rates that overall range betwixt 90.65%–95.68%. Better performance metrics rates are attained by the HTR-RNN. Thus, the proposed one performs the MVI process more accurately.

Table 1 PA of proposed HTR-RNN based on accuracy, sensitivity, and specificity

Techniques	*Performance metrics (%)*		
	Accuracy	*Sensitivity*	*Specificity*
Proposed HTR-RNN	97.56	97.57	97.54
RNN	95.76	95.84	95.68
DBN	93.34	93.42	93.26
DNN	91.54	91.69	91.39
CNN	90.65	90.65	90.65

Figure 5 Graphical representation of the proposed HTR-RNN based on accuracy, sensitivity, and specificity (see online version for colours)

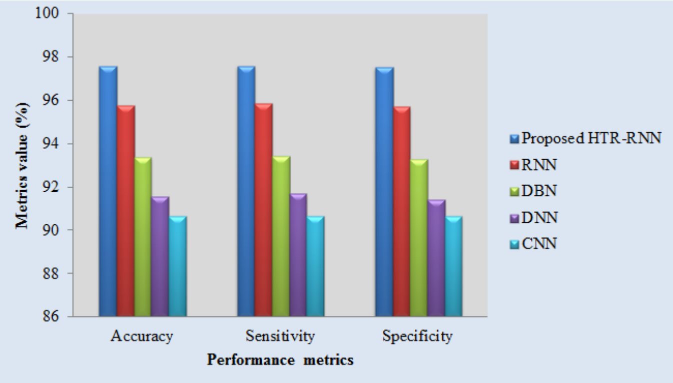


Table 2 PA of proposed HTR-RNN based on precision, recall, and F-measure

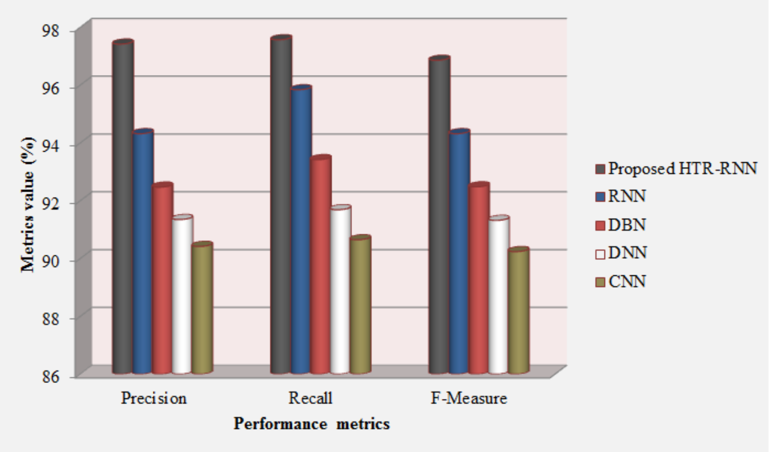
Techniques	*Performance metrics (%)*		
	Precision	*Recall*	*F-measure*
Proposed HTR-RNN	97.42	97.57	96.86
RNN	94.32	95.84	94.32
DBN	92.47	93.42	92.48
DNN	91.35	91.69	91.32
CNN	90.42	90.65	90.24

The comparison analysis of the metrics value attained by the HTR-RNN and also other prevailing works is exhibited in Figure 5. The model's metrics value ought to remain high as possible if it is alleged to be more robust in addition to effective. The HTR-RNN obtained high accuracy, sensitivity, along with specificity rates. Nevertheless, RNN, DBN, DNN, along with CNN attained the metrics value that is very low analogised to the proposed work. Thus, the proposed work proffers more propitious outcomes in an MVI process.

The recall, precision, together with the F-measure value of the HTR-RNN and also the other prevailing works, like RNN, DBN, DNN, and CNN is exhibited in tabulation 2. The higher rate of recall, precision, in addition to F-measure determines the model's significance. As per the statement, the proposed work achieves 97.42% precision, 97.57%

recall, along with 96.86% F-measure. Nonetheless, the prevailing work attains the precision, recall, along with F-measure rate at the average of 92.14%, 92.9%, as well as 92.09% correspondingly. This is literally low as analogised to the proposed work. Thus, the proposed HTR-RNN mitigates disparate complexities and ameliorates the missing value imputation's reliability.

Figure 6 Comparative analysis of proposed HTR-RNN based on precision, recall, and F-measure (see online version for colours)



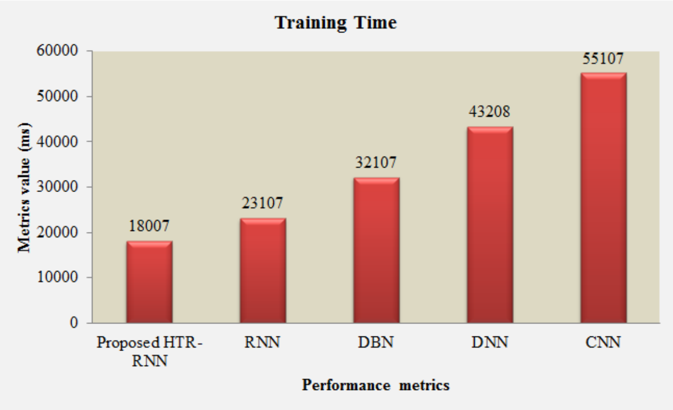
A clear view of tabulation 2 is rendered in Figure 6. The comparative analysis of the proposed work is displayed in Figure 6. The proposed HTR-RNN attains higher precision, recall, as well as F-measure values that range betwixt 94.34%–98.98%. RNN, DBN, DNN, and CNN attain the metrics values that overall range betwixt 90.24%–95.84%, which is lower when weighed against the HTR-RNN. Therefore, the HTR-RNN trounces the other top-notch methods and delivers more prominent outcomes under disparate intricate circumstances.

Table 3 PA of proposed HTR-RNN on the basis of CPU memory usage

Techniques	*CPU memory usage*
Proposed HTR-RNN	415,068
RNN	590,221
DBN	638,194
DNN	789,666
CNN	888,244

Table 3 represents the CPU memory usage of the proposed HTR-RNN and the other exiting works. The CPU memory usage defines the amount of memory consumed during the execution. If the model consumed high memory space, then the entire processing gets slowed down, this affects the performance of the model. As per the data mentioned in Table 3, the proposed model consumes 415,068 kb of CPU memory. But the existing works, such as RNN, DBN, DNN, and CNN consume 590,221 kb, 638,194 kb, 789,666 kb, and 888,244 kb of CPU memory space. Therefore, the proposed work successfully executes the task with limited memory consumption.

Figure 7 Comparative analysis of the proposed HTR-RNN with respect to training time (see online version for colours)



The training time of the HTR-RNN with the existent works is exhibited in Figure 7. The model should complete the whole training process with a minimum time for it to be more robust. As per that, the HTR-RNN takes 67,512 ms to finish the training process. However, RNN, DBN, DNN, along with CNN, took 23,107 ms, 32,107 ms, 43,208 ms, in addition to 55,107 ms, correspondingly to finish the training process. When contrasted with the other prevailing techniques, the proposed one completes the whole process as quickly as possible. Therefore, the time intricacy of the work is alleviated by the proposed work effectively.

7 Conclusions and future scope

The work has proposed an efficient missing value imputation and evaluation using GK-KH means and HTR-RNN. The proposed framework comes with several operations, like pre-processing, missing data, data transformation, structured data, feature extraction, feature selection, classification, and finally, accuracy evaluation of the missing value imputation process. After that, the experimentation analysis is performed in which the PA and the comparative analysis of the proposed techniques are done concerning some performance metrics. The developed approach can handle various uncertainties and render more promising results. The publically available CKD dataset is used for the analysis, in which the proposed method achieves 97.56% of accuracy, 97.57% of sensitivity, and 97.54% of specificity. Overall, the proposed missing value imputation framework outperforms the existing state-of-art methods and remains to be more reliable and robust. In the future, the work will be extended with some advanced NN and perform the imputation process for more complicated datasets.

References

Abdelaziz, A., Salama, A.S., Riad, A.M. and Mahmoud, A.N. (2018) 'A machine learning model for predicting of chronic kidney disease based internet of things and cloud', *Security in Smart Cities: Models, Applications, and Challenges, Part of the Lecture Notes in Intelligent Transportation and Infrastructure Book Series (LNITI)*, Springer, Nature, Switzerland, AG, pp.93–114.

Abu-Soud, S.M. (2019) 'A novel approach for dealing with missing values in machine learning datasets with discrete values', *International Conference on Computer and Information Sciences (ICCIS)*.

Agarwal, A. (2019) *The Problem of Vanishing Gradients* [online] https://towardsdatascience.com/the-problem-of-vanishing-gradients-68cea05e2625 (accessed 20 November 2021).

Andiojaya, A. and Demirhan, H. (2019) 'Abagging algorithm for the imputation of missing values in timeseries', *Expert Systems with Applications*, Vol. 129, No. 3, pp.10–26.

Arulanthu, P. and Perumal, E. (2020) 'An intelligent IoT with cloud centric medical decision support system for chronic kidney disease prediction', *International Journal of Imaging Systems and Technology*, Vol. 30, No. 3, pp.815–827.

Arunkumar, P.M. and Kannimuthu, S. (2020) 'Mining bigdata streams using business analytics tools:a bird's eye view on MOA and SAMOA', *Int. J. Business Intelligence and Data Mining*, Vol. 17, No. 2, pp.226–236.

Betty, P., Geetha, D.M. and Jacob, I.J. (2020) 'An efficient feature extraction forbiometric authentication', *Int. J. Business Intelligence and Data Mining*, Vol. 16, No. 4, pp.480–489.

De Bodt, C., Mulders, D. and Verleysen, M. (2019) 'Nonlinear dimensionality reduction with missing data using parametric multiple imputations', *IEEE Transactions on Neural Networks and Learning Systems*, Vol. 30, No. 4, pp.1166–1179.

Ding, X., Wang, H., Su, J., Wang, M., Li, J. and Gao, H. (2020) 'Leveraging currency for repairing inconsistent and incomplete data', *IEEE Transactions on Knowledge and Data Engineering*, 1 March, Vol. 34, No. 3, p.1.

Emmanuel, T., Maupong, T. et al. (2021) 'A survey on missing data in machine learning', *Journal of Big Data*, Article Number: 140, Vol. 8, pp.1–37.

Gu, B., Li, Z., Liu, A., Xu, J., Zhao, L. and Zhou, X. (2019) 'Improving the quality of web-based dataimputation with crowd intervention', *IEEE Transaction on Knowledge and Data Engineering*, Vol. 33, No. 6, pp.2534–2547.

Hosseinzadeh, M., Koohpayehzadeh, J., Bali, A.O., Asghari, P., Souri, A., Mazaherinezhad, A., Bohlouli, M. and Rawassizadeh, R. (2021) 'A diagnostic prediction model for chronic kidney disease in internet of things platform', *Multimedia Tools and Applications*, Vol. 80, No. 11, pp.16933–16950.

Ispirova, G., Eftimov, T. and Seljak, B.K. (2020) 'Evaluating missing value imputation methods for food composition databases', *Food and Chemical Toxicology*, July, Vol. 141, pp.1–19.

Khamparia, A., Saini, G., Pandey, B., Tiwari, S., Gupta, D. and Khanna, A. (2020) 'KDSAE: Chronic kidney disease classification with multimedia data learning using deep stacked autoencoder network', *Multimedia Tools and Applications*, Vol. 79, No. 47, pp.35425–35440.

Khan, H., Wang, X. and Liu, H. (2021) 'Missing value imputation through shorter interval selection driven by fuzzy c-means clustering', *Computers and Electrical Engineering*, July, Vol. 93, No. C, pp.1–16.

Lai, X., Wu, X., Zhang, L., Lu, W. and Zhong, C. (2019) 'Imputations of missing values using a tracking-removed autoencoder trained with incomplete data', *Neurocomputing*, 13 November, Vol. 366, pp.54–65.

Lam, L.T. and Hsiao, S-W. (2019) 'AI-based online P2P lending risk assessment on social network data with missing value', *IEEE International Conference on Big Data (Big Data)*, Los Angeles, CA, USA, pp 6113-6115.

Li, L., Zhang, J., Wang, Y. and Ran, B. (2018) 'Missing value imputation for traffic-related time series data based on a multi-view learning method', *IEEE Transactions on Intelligent Transportation Systems*, Vol. 20, No. 8, pp.2933–2943.

Ma, F., Sun, T., Liu, L. and Jing, H. (2020) 'Detection and diagnosis of chronic kidney disease using deep learning-based heterogeneous modified artificial neural network', *Future Generation Computer Systems*, October, Vol. 111, pp.17–26.

Migdady, H., Alrabaiah, H. and Al-Talib, M. (2020) 'Note about bias in Bayesian genetic algorithms for discrete missing values imputation', *International Arab Conference on Information Technology (ACIT)*, pp.87–90.

Mostafa, S.M., Eladimy, A.S., Hamad, S. and Amano, H. (2020) 'CBRG a novel algorithm for handling missing data using Bayesian ridge regression and feature selection based on gain ratio', *IEEE Access*, 2 December, Vol. 8, pp.216969–216985.

Sarkar, B. and Sinhababu, N. (2019) 'Mining multilingual and multiscript Twitter data: unleashing the language and script barrier', *International Journal of Business Intelligence and Data Mining*, Vol. 16, No. 1, pp107–127.

Sefidian, A.M. and Negin, D. (2018) 'Missing value imputation using a novel grey based fuzzy C-means, mutual information based feature selection and regression model', *Expert Systems with Applications*, January, Vol. 115, pp.68–94.

Tsai, C-F., Li, M-L. and Lin, W-C. (2018) 'A class center based approach for missing value imputation', *Knowledge-Based Systems*, March, Vol. 151, pp.124–135.

Turabieh, H., Mafarja, M. and Mirjalili, S. (2019) 'Dynamic adaptive network-based fuzzy inference system (D-ANFIS) for the imputation of missing data for internet of medical things applications', *IEEE Internet of Things Journal*, Vol. 6, No. 6, pp.9316–9325.

Xu, X., Chong, W., Li, S., Arabo, A. and Xiao, J. (2018) 'MIAEC Missing data imputation based on the evidence chain', *IEEE Access*, 21 February, Vol. 6, pp.12983–12992.

Ye, C., Wang, H., Lu, W. and Li, J. (2019) 'Effective Bayesian-network-based missing value imputation enhanced by crowdsourcing', *Knowledge-Based Systems*, DOI: 10.1016/j.knosys.2019.105199.

Zhang, Y., Zhou, B., Cai, X., Guo, W., Ding, W. and Yuan, X. (2021) 'Missing value imputation in multivariate time series with end-to-end generative adversarial networks', *Information Sciences*, April, Vol. 551, pp.67–82.