JOURNAL OF NORTHEASTERN UNIVERSITY

Volume 25 Issue 04, 2022 ISSN: 1005-3026 https://dbdxxb.cn/ Original Research Paper

MVIBPM: DESIGN OF A MISSING VALUE IDENTIFICATION TECHNIQUE VIA BIOINSPIRED PREDICTIVE MODELING

Dipalika Das [1]*, Maya Nayak[2] , Subhendu Kumar Pani[3]

[1] Dipalika Das, Research Scholar, Department of Computer Science and Engineering, Biju Patnaik University of Technology, Rourkela, Odisha, India; dipalika.das@gmail.com,

[2] Dr. Maya Nayak, Dean School of Computer Studies, Ajay Binay Institute of Technology(ABIT), Cuttack, Biju Patnaik University of Technology(BPUT) Rourkela, Odisha, India; mayanayak3299@yahoo.com,

[3] Dr. Subhendu Kumar Pani, Professor, Krupajal Engineering College(KEC), Bhubaneswar, Biju Patnaik University of Technology (BPUT),Rourkela, Odisha, India; pani.subhendu@gmail.com,

Abstract: Identification of missing values from time-series data samples is a complex signal processing task, that involves pattern analysis, pre-emptive modelling, and regression techniques. A wide variety of models are proposed by researchers to optimize efficiency of missing value identification techniques, but most of them are highly complex, and cannot be used for large-scale information sets. Moreover, the simpler models that are applied to large-scale sets have low efficiency levels, which limits their applicability for real-time applications. To overcome these issues, this text proposes design of a novel Elephant Herding Optimization (EHO) Model for tuning an efficient missing value identification ensemble classifier, which can be used for feature-based data samples. The proposed model uses a combination of Deep Forest (DF), Support Vector Machines (SVM), Naïve Bayes (NB), and k Nearest Neighbour (kNN) classifiers for correlative analysis of missing value samples. The efficiency of proposed classifier is optimized via EHO model, which assists in identification of classifier hyper parameters in order to improve performance of missing value identification process. The EHO model uses an efficient fitness function that combines accuracy, precision, and recall levels obtained when evaluating effectiveness of the missing value identification process. To evaluate its performance, the model was used for multiple large-scale datasets, and an accuracy improvement of 9.5%, with a precision improvement of 8.3%, and recall improvement of 4.5% was observed, when compared with standard regression-based pre-emption models. Due to this, the proposed method was observed to be highly scalable, and can be applied to multidomain use cases.

Keywords: Missing, Value, NB, kNN, SVM, DF, EHO, Accuracy, Precision, Recall, Optimizations

1. Introduction

A time series is generally understood to refer to a collection of measurements that have been obtained at consistent time intervals. The basic objective of time series prediction is to foretell future tendencies in the data by examining past data. This is accomplished via the use of

Submitted: 17/10/2022 **Accepted**: 10/11/2022

historical data. As a result, it plays an important part in the process of decision-making for a variety of applications, including industrial monitoring, business metrics, management of electrical grids, and other applications. The following is a list of probable overarching categories for the challenges with the time series. If we are just concerned with the next one- or two-time steps, as is the case with the overwhelming majority of time series issues, we will create a prediction that is referred to as a one-step or single-step forecast. Predictions that look multiple steps into the future are referred to as multistep predictions since they gaze quite far into the future. When making a prediction that involves a number of stages, you have the option of using either the direct technique or the iterative method. The direct approach requires the creation of a model that can foretell the outcomes of many stages in the future, while the iterative method necessitates the creation of a series of predictions for one step at a time up until the relevant step is reached. The field of time series prediction has seen a recent uptick in the use of artificial intelligence (AI). The support vector machine is a well-known artificial intelligence technology that is used for time series prediction (SVM). During the 1970s, Vapnik and his fellow employees at AT&T Bell Laboratories made important strides in the development of SVM. It was first developed to assist with categorization issues and had practical uses such as optical character recognition at the time. In [1, 2, 3], an improvement was made to the support vector machine in order to address issues with regression. Neural networks must be concerned with local minimums, but support vector machines do not have this worry. However, in order to tackle quadratic programming issues, a significant amount of computing power is required. The Takagi-Sugeno Modeling (TSM), and Least Squares Support Vector Machine, sometimes known as the LSSVM, was presented in reference [4, 5, 6] as a method for converting constraint problems into a linear system. Although the LSSVM is better at cutting down on computational expenses, in the process, the sparsity of the support vectors is lost. The weighted LSSVM is an alternate strategy that was presented to cope with sparsity and also used Local Median-based Gaussian Naive Bayes (LMeGNB) [7, 8, 9]. LSSVM has recently been successful in a number of domains, including time series prediction and financial forecasting, among others. Since the data for time series are obtained from real-world scenarios, it is common for there to be values that are missing. Failures of the sensors or mistakes made by humans might explain for the missing data. [10, 11, 12, 13] In order to address the issue of missing data, a variety of ad hoc approaches have been tried out throughout the years. Among them are techniques and deletion processes that work toward the goal of finding a solitary replacement for each value that was lost. Ad hoc approaches have the potential to have an effect on both standard errors and biases in estimates [14, 15]. Despite this, research demonstrates that they are nevertheless used on a regular basis [16, 17, 18]. The maximum likelihood method [19, 20] and the multiple imputation methods [21, 22] are two of the most well-known and successful approaches to the imputation of missing data. When using multiple imputation, copies of the missing information are first generated, and then those duplicates are separately imputed. The ultimate judgment was arrived at by compiling the results of several parameter estimates as well as standard errors, one of each kind for every copy that was examined. Also, maximum likelihood with generative adversarial network

(GAN) and bi-directional long short-term memory (Bi-LSTM) [23, 24, 25] takes into account all of the data that is available and produces estimates with the greatest probability. Considering that multiple imputation and maximum probability often provide the same results, choose one over the other is a very subjective decision. It is difficult to make predictions based on time series since there is missing information. When compared to other types of data analysis, time series prediction stands out due to the temporal relevance of its predictions.

These findings imply that academics have developed a broad range of models in an effort to improve the efficiency of missing value detection techniques. Despite this, the great majority of these models are very complex to implement and cannot be used to enormous information sets. In addition, the low levels of efficiency that simpler models have when applied to large-scale datasets are a hurdle to the development of real-time applications. This paper provides a unique Elephant Herding Optimization (EHO) Model for optimizing an efficient ensemble classifier for missing value detection that is applicable to feature-based data sets. The goal of this model is to overcome the problems that have been identified. In section 3 of this text, using a wide range of datasets, an evaluation of the usefulness of the model was carried out. This performance was evaluated in comparison to industry standards in order to demonstrate its superiority over contemporary models. The investigation comes to a close with some general thoughts on the work that was offered as well as some recommendations for broadening its applicability under different use cases.

2. Proposed Missing Value Identification technique via Bioinspired Predictive Modeling

Based on the review of existing missing value identification models, it can be observed that most of these models are highly complex, and cannot be used for large-scale information sets. Moreover, the simpler models that are applied to large-scale sets have low efficiency levels, which limits their applicability for real-time applications.

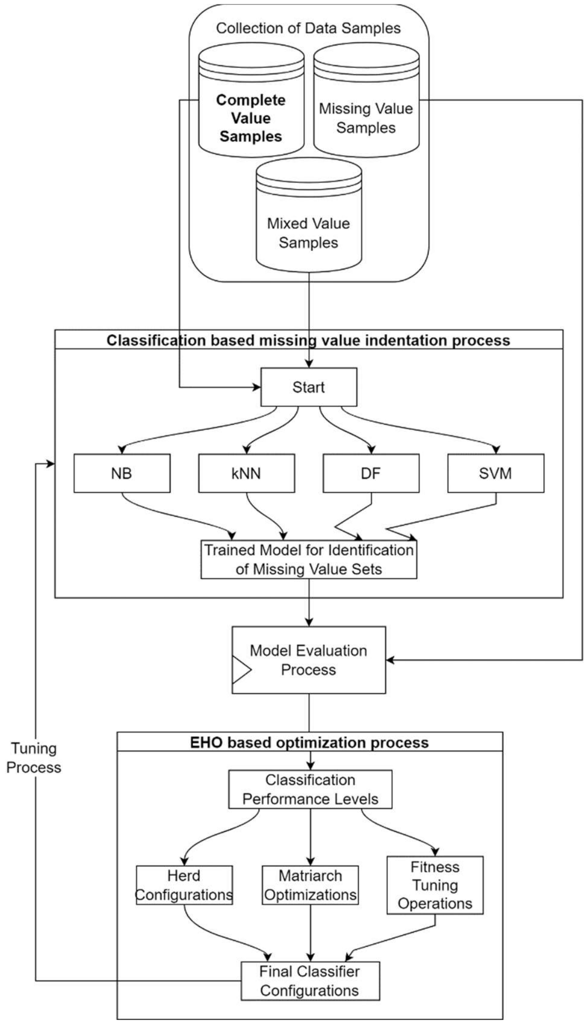
Collection of Data Samples
Complete Value Samples
Missing Value Samples
Mixed Value Samples
Classification based missing value indentation process
Start
NB
kNN
DF
SVM
Trained Model for Identification of Missing Value Sets
Model Evaluation Process
Tuning Process
EHO based optimization process
Classification Performance Levels
Herd Configurations
Matriarch Optimizations
Fitness Tuning Operations
Final Classifier Configurations

Figure 1. Flow of the proposed missing value identification process

To overcome these issues, this section proposes design of a novel Elephant Herding Optimization (EHO) Model for tuning an efficient missing value identification ensemble classifier, which can be used for feature-based data samples. Overall flow of the proposed model is depicted in figure 1, where it can be observed that the proposed model uses a combination of Deep Forest (DF), Support Vector Machines (SVM), Naïve Bayes (NB), and k Nearest Neighbour (kNN) classifiers for correlative analysis of missing value samples. The efficiency of proposed classifier is optimized via EHO model, which assists in identification of classifier hyper parameters in order to improve performance of missing value identification process. The EHO model uses an efficient fitness function that combines accuracy, precision, and recall levels obtained when evaluating effectiveness of the missing value identification process.

Based on the flow, it can be observed that training datasets are used to train Naïve Bayes (NB), k Nearest Neighbour (kNN), Deep Forest (DF), and Support Vector Machine (SVM) classifiers. These classifiers and their initial parameter sets can be observed from table as follows,

Classifier	**Parameter Sets**
Naïve Bayes	*Priors* These are set to variance levels of training set samples Smoothing Value (S_v), is initially set to 1, and tuned by the EHO process
kNN	k = 1, and tuned by the EHO process
Deep Forest	Number of Estimators (N_{est}), initially set as number of features, and tuned by the EHO process Max Depth (M_{dep}), initially set as 1, and later modified by the EHO process
SVM	Regularization Coefficient (C), initially setup as 1, and modified by the EHO process Tolerance (tol), initially setup as 0.0001, later modified by the EHO process

Table 1. Classifiers along with their parameter sets

Based on these parameter sets, missing value samples are classified into 1 of N categories. The average value of missing parameters (MPV) is evaluated via equation 1,

$$MPV = \sum_{i=1}^{N_c} \frac{NMVP_i}{N_c} \dots (1)$$

Where, $NMVP$ & N_c represents values of parameters that are not missing in the dataset, and total samples present in the identified class. Based on this value of MPV, accuracy of classifier is estimated, and kept for future reference purposes. If the accuracy is observed to be lower than a specified threshold, then an EHO based optimization model is activated, which works as per the following process,

- To initialize the optimization model, setup following EHO constants,
  - EHO iterations for which Herds will be checked and reconfigured (N_i)
  - EHO Herds which will be used for optimization process (N_h)
  - Rate at which Herds will learn from each other (L_r)
  - Current parameters for each of the classifier obtained from table 1, which has to be optimized by EHO process
- Once these parameters are setup, then generate N_h solutions as per the following process,
  - Stochastically modify values for each of the classifier parameters via equations, 2, 3, 4, 5, 6 and 7 as follows,

$$S_v = S_v(Old) \pm STOCH\left(\frac{L_r}{2}, L_r\right) \dots (2)$$

Where, $STOCH$ represents a Markovian process used for generation of stochastic number sets.

$$k = k(old) \pm 1 \dots (3)$$

Where, increment (+), and decrement (-) operators are selected stochastically for individual solution sets.

$$N_{est} = N_{est}(Old) * STOCH\left(\frac{L_r}{2}, 2 * L_r\right) \dots (4)$$

$$M_{depth} = M_{depth}(Old) \pm 1 \dots (5)$$

$$C = C(Old) * STOCH\left(\frac{L_r}{2}, L_r\right) \dots (6)$$

$$tol = tol(old) \pm STOCH\left(tol * \frac{L_r}{2}, tol * L_r\right) \dots (7)$$

- Based on these values of classifier parameters, fitness levels are estimated for each Herd via equation 8,

$$f = \frac{A + P + R}{3} \dots (8)$$

Where, A, P, R represents accuracy, precision & recall levels for each of the classifier entities, and is estimated via equations 9, 10 and 11,

$$A = \frac{t_p + t_n}{t_p + t_n + f_p + f_n} \dots (9)$$

$$P = \frac{t_p}{t_p + t_n} \dots (10)$$

$$R = \frac{t_p + f_p}{t_p + t_n + f_p + f_n} \dots (11)$$

Where, t_p, t_n, f_p & f_n represents values of true positive, true negative, false positive, and false negative under real-time use cases.

○ Repeat this process for all Herds, which assists in generation of N_h different solution sets.

- Once all Herd configurations (solution sets) are generated, then estimate Herd solution fitness threshold via equation 12,

$$f_{th} = \frac{1}{N_h} \sum_{i=1}^{N_h} f_i * L_r \dots (12)$$

- Herds that showcase $f < f_{th}$ are reconfigured via equations, 2, 3, 4, 5, 6, & 7; while other Herds are not modified during consecutive iterations.
- At the end of each iteration, Herd with maximum fitness level is marked as 'Matriarch' Herd, and is used to modify the learning rate via equation 13,

$$L_r(New) = L_r(Old) \pm \frac{f(Matriarch)}{\sum_{i=1}^{N_h} f_i} \dots (13)$$

Where, $f(Matriarch)$ represents highest fitness levels, and the rate is incremented if current solution is better than previous, while rate is reduced if current solution has lower performance than previous. Once all iterations are completed, then select solution with maximum fitness levels, and use its configurations if accuracy with this solution is higher than current accuracy levels. Due to which, the proposed model is able to improve classification performance under different use cases. This performance is evaluated, and compared with standard models in the next section of this text.

3. Result & Comparison

The proposed model uses a combination of Naïve Bayes (NB), k Nearest Neighbours (kNN), Support Vector Machine (SVM), and Deep Forest (DF) classifiers in order to estimate correct class for missing value samples. The values of this class are averaged to estimate current missing value sets. Performance of this classifier is improved via a EHO based optimization process, which assists in identification of optimal hyperparameters that can achieve higher accuracy under different data samples. This accuracy was estimated for Missing Value Dataset from Kaggle (https://www.kaggle.com/code/alexisbcook/missing-values/data), Brittleness

Index Dataset (https://openmv.net/info/brittleness-index), Class Grades Dataset (https://openmv.net/info/class-grades), and Raw Material Properties Datasets (https://openmv.net/info/raw-material-properties) samples. Each of these sets were aggregated to form a total of 500k data samples, out of which 70% were used for training, while 15% each were used for validation & testing purposes. Based on this strategy, the accuracy of missing value identification was estimated w.r.t. Test Set Samples (TSS), and compared with TSM [4], LME GNB [9], and GAN Bi LSTM [25] in table 2 as follows,

TSS	**A (%)** **TSM [4]**	**A (%)** **LME GNB [9]**	**A (%)** **GAN Bi LSTM [25]**	**A (%)** **MVI BPM**
833	79.94	79.45	81.15	85.32
1250	81.24	80.64	82.39	86.61
1667	82.29	81.60	83.38	87.66
2500	83.14	82.40	84.20	88.52
2917	83.88	83.13	84.95	89.31
3333	84.63	83.91	85.74	90.14
3750	85.48	84.78	86.62	91.07
4167	86.40	85.69	87.56	92.05
4583	87.33	86.60	88.49	93.02
5000	88.23	87.48	89.38	93.97
5417	89.13	88.36	90.27	94.92
5833	90.09	89.28	91.13	95.86
6250	91.10	90.25	91.95	96.74
6667	92.11	91.21	92.79	97.64

7083	93.07	92.12	93.71	98.60
7500	93.92	92.96	94.64	99.57

Table 2. Accuracy evaluation of different missing value models

Based on this estimation, and figure 2, it was observed that the proposed model showcased 5.5% higher accuracy than TSM [4], 6.4% higher accuracy than LME GNB [9], 4.9% higher accuracy than GAN Bi LSTM [25] under different use cases. The reason for this accuracy enhancement is use of accuracy during tuning the classifier hyperparameter sets.

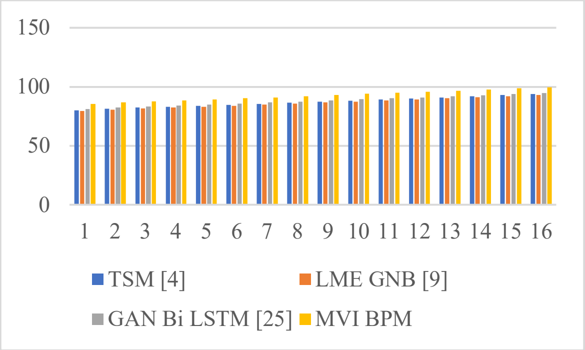

Figure 2. Accuracy evaluation of different missing value models

Similar performance was evaluated for precision levels, and can be observed from table 3 as follows,

TSS	P (%) TSM [4]	P (%) LME GNB [9]	P (%) GAN Bi LSTM [25]	P (%) MVI BPM
833	75.90	76.48	78.60	81.23
1250	77.09	77.63	79.82	82.45
1667	78.05	78.56	80.81	83.44
2500	78.83	79.34	81.63	84.27
2917	79.53	80.04	82.36	85.02

3333	80.26	80.79	83.11	85.81
3750	81.07	81.62	83.95	86.70
4167	81.95	82.50	84.86	87.63
4583	82.82	83.37	85.76	88.55
5000	83.67	84.22	86.64	89.46
5417	84.52	85.07	87.52	90.38
5833	85.41	85.96	88.44	91.35
6250	86.36	86.90	89.42	92.36
6667	87.30	87.83	90.40	93.36
7083	88.19	88.71	91.31	94.29
7500	88.99	89.52	92.15	95.15

Table 3. Precision evaluation of different missing value models

Based on this estimation, and figure 3, it was observed that the proposed model showcased 6.5% higher precision than TSM [4], 5.5% higher precision than LME GNB [9], 2.9% higher precision than GAN Bi LSTM [25] under different use cases. The reason for this precision enhancement is use of this parameter during the EHO tuning process which assists in identification of efficient parameters for each of the classifiers.

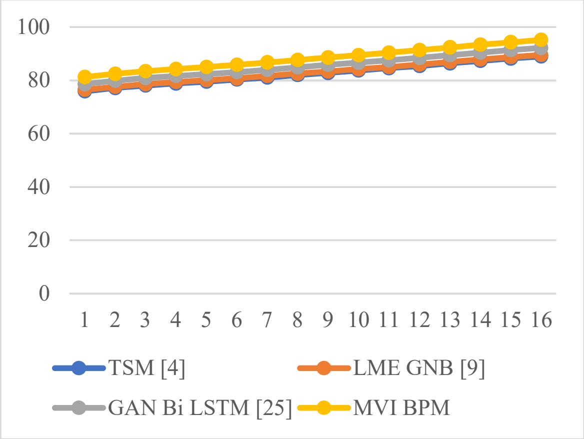

Figure 3. Precision evaluation of different missing value models

Similar performance was evaluated for recall levels, and can be observed from table 4 as follows,

TSS	**R (%)** **TSM [4]**	**R (%)** **LME GNB [9]**	**R (%)** **GAN Bi LSTM [25]**	**R (%)** **MVI BPM**
833	77.92	77.96	79.87	83.27
1250	79.16	79.13	81.10	84.53
1667	80.17	80.08	82.09	85.55
2500	80.99	80.87	82.91	86.39
2917	81.71	81.59	83.66	87.17
3333	82.45	82.35	84.42	87.98
3750	83.28	83.20	85.29	88.88
4167	84.18	84.10	86.21	89.84
4583	85.08	84.99	87.12	90.79

5000	85.95	85.85	88.01	91.72
5417	86.82	86.71	88.90	92.66
5833	87.75	87.62	89.84	93.66
6250	88.73	88.57	90.83	94.70
6667	89.70	89.52	91.81	95.72
7083	90.63	90.42	92.74	96.68
7500	91.45	91.24	93.59	97.56

Table 4. Recall evaluation of different missing value models

Based on this estimation, and figure 4, it was observed that the proposed model showcased 5.9% higher recall than TSM [4], 6.2% higher recall than LME GNB [9], 4.5% higher recall than GAN Bi LSTM [25] under different use cases. The reason for this recall enhancement is use of ensemble classifier, and inclusion of recall during the EHO tuning process which assists in identification of efficient parameters for each of the classifiers.

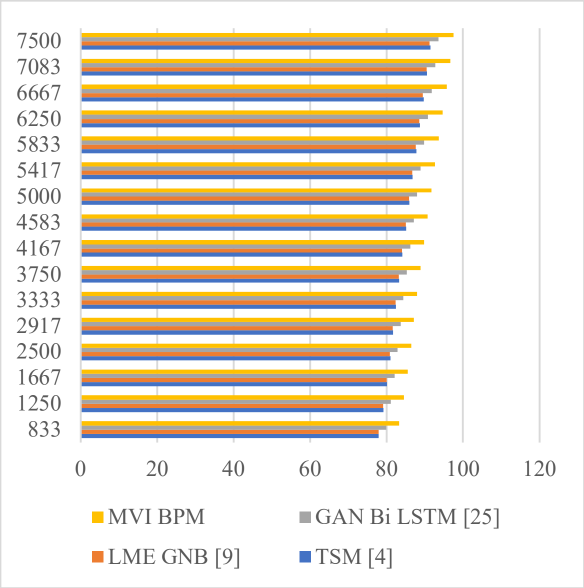

Figure 4. Recall evaluation of different missing value models

Similar performance was evaluated for computational delay levels, and can be observed from table 5 as follows,

TSS	**D (ms)** **TSM [4]**	**D (ms)** **LME GNB [9]**	**D (ms)** **GAN Bi LSTM [25]**	**D (ms)** **MVI BPM**
833	1.60	1.60	1.56	1.33
1250	1.97	1.97	1.92	1.63
1667	2.33	2.34	2.28	1.92
2500	2.70	2.70	2.64	2.21
2917	3.08	3.08	3.00	2.51
3333	3.49	3.49	3.41	2.85
3750	3.97	3.97	3.88	3.25
4167	4.53	4.53	4.42	3.70
4583	5.15	5.16	5.03	4.21
5000	5.81	5.82	5.68	4.72
5417	6.45	6.46	6.30	5.21
5833	7.01	7.02	6.85	5.63
6250	7.48	7.49	7.30	5.99
6667	7.89	7.90	7.70	6.32
7083	8.31	8.31	8.11	6.66

7500	8.79	8.79	8.58	7.05

Table 5. Delay evaluation for different missing value models

Based on this estimation, and figure 5, it was observed that the proposed model showcased 10.5% higher speed than TSM [4] and LME GNB [9], and 9.5% higher speed than GAN Bi LSTM [25] under different use cases. The reason for this delay enhancement is selection of optimal tuning parameters, and use of ensemble classifier which assists in identification of efficient parameters for each of the classifiers.

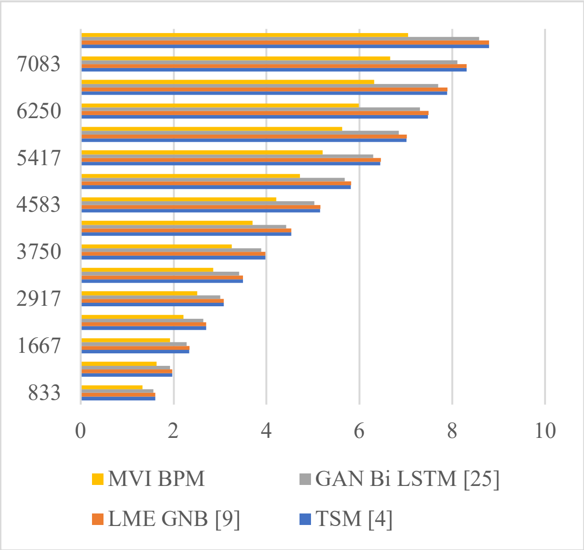

Figure 5. Delay evaluation for different missing value models

Due to these optimizations, the proposed model is capable of low-error, high-speed, high precision, and better recall performance, which makes it useful for a wide variety of missing value identification applications.

4. Conclusion & future scope

The proposed model uses a combination of Naïve Bayes (NB), k Nearest Neighbours (kNN), Support Vector Machine (SVM), and Deep Forest (DF) classifiers in order to estimate correct class for missing value samples. The values of this class are averaged to estimate current missing value sets. Performance of this classifier is improved via a EHO based optimization process, which assists in identification of optimal hyperparameters that can achieve higher accuracy under different data samples. The model's performance was evaluated on different datasets, and it was observed that the proposed model showcased 5.5% higher accuracy than TSM [4], 6.4% higher accuracy than LME GNB [9], 4.9% higher accuracy than GAN Bi LSTM [25] under different use cases. The reason for this accuracy enhancement is use of accuracy during tuning the classifier hyperparameter sets. Based on precision estimation, it was observed that the proposed model showcased 6.5% higher precision than TSM [4], 5.5% higher precision than LME GNB [9], 2.9% higher precision than GAN Bi LSTM [25] under different use cases. The reason for this precision enhancement is use of this parameter during the EHO tuning process which assists in identification of efficient parameters for each of the classifiers. Based on recall evaluation, it was observed that the proposed model showcased 5.9% higher recall than TSM [4], 6.2% higher recall than LME GNB [9], 4.5% higher recall than GAN Bi LSTM [25] under different use cases. The reason for this recall enhancement is use of ensemble classifier, and inclusion of recall during the EHO tuning process which assists in identification of efficient parameters for each of the classifiers. Based on computational delay estimation, it was observed that the proposed model showcased 10.5% higher speed than TSM [4] and LME GNB [9], and 9.5% higher speed than GAN Bi LSTM [25] under different use cases. The reason for this delay enhancement is selection of optimal tuning parameters, and use of ensemble classifier which assists in identification of efficient parameters for each of the classifiers. Due to these optimizations, the proposed model is capable of low-error, high-speed, high precision, and better recall performance, which makes it useful for a wide variety of missing value identification applications.

5. References

[1] T. Huamin, D. Qiuqun and X. Shanzhu, "Reconstruction of time series with missing value using 2D representation-based denoising autoencoder," in Journal of Systems Engineering and Electronics, vol. 31, no. 6, pp. 1087-1096, Dec. 2020, doi: 10.23919/JSEE.2020.000081.

[2] P. C. Chiu, A. Selamat, O. Krejcar, K. K. Kuok, S. D. A. Bujang and H. Fujita, "Missing Value Imputation Designs and Methods of Nature-Inspired Metaheuristic Techniques: A Systematic Review," in IEEE Access, vol. 10, pp. 61544-61566, 2022, doi: 10.1109/ACCESS.2022.3172319.

[3] M. Jena and S. Dehuri, "An Integrated Novel Framework for Coping Missing Values Imputation and Classification," in IEEE Access, vol. 10, pp. 69373-69387, 2022, doi: 10.1109/ACCESS.2022.3187412.

[4] X. Lai, L. Zhang and X. Liu, "Takagi-Sugeno Modeling of Incomplete Data for Missing Value Imputation With the Use of Alternate Learning," in IEEE Access, vol. 8, pp. 83633-83644, 2020, doi: 10.1109/ACCESS.2020.2991669.

[5] D. Li, H. Zhang, T. Li, A. Bouras, X. Yu and T. Wang, "Hybrid Missing Value Imputation Algorithms Using Fuzzy C-Means and Vaguely Quantified Rough Set," in IEEE Transactions on Fuzzy Systems, vol. 30, no. 5, pp. 1396-1408, May 2022, doi: 10.1109/TFUZZ.2021.3058643.

[6] B. Foggo and N. Yu, "Online PMU Missing Value Replacement Via Event-Participation Decomposition," in IEEE Transactions on Power Systems, vol. 37, no. 1, pp. 488-496, Jan. 2022, doi: 10.1109/TPWRS.2021.3093521.

[7] S. J. Fernstad and J. J. Westberg, "To Explore What Isn't There—Glyph-Based Visualization for Analysis of Missing Values," in IEEE Transactions on Visualization and Computer Graphics, vol. 28, no. 10, pp. 3513-3529, 1 Oct. 2022, doi: 10.1109/TVCG.2021.3065124.

[8] T. Wang, H. Ke, A. Jolfaei, S. Wen, M. S. Haghighi and S. Huang, "Missing Value Filling Based on the Collaboration of Cloud and Edge in Artificial Intelligence of Things," in IEEE Transactions on Industrial Informatics, vol. 18, no. 8, pp. 5394-5402, Aug. 2022, doi: 10.1109/TII.2021.3126110.

[9] L. Jia, Z. Wang, S. Lv and Z. Xu, "PE_DIM: An Efficient Probabilistic Ensemble Classification Algorithm for Diabetes Handling Class Imbalance Missing Values," in IEEE Access, vol. 10, pp. 107459-107476, 2022, doi: 10.1109/ACCESS.2022.3212067.

[10] S. M. Mostafa, A. S. Eladimy, S. Hamad and H. Amano, "CBRG: A Novel Algorithm for Handling Missing Data Using Bayesian Ridge Regression and Feature Selection Based on Gain Ratio," in IEEE Access, vol. 8, pp. 216969-216985, 2020, doi: 10.1109/ACCESS.2020.3042119.

[11] Y. Liu, T. Dillon, W. Yu, W. Rahayu and F. Mostafa, "Missing Value Imputation for Industrial IoT Sensor Data With Large Gaps," in IEEE Internet of Things Journal, vol. 7, no. 8, pp. 6855-6867, Aug. 2020, doi: 10.1109/JIOT.2020.2970467.

[12] R. Razavi-Far, D. Wan, M. Saif and N. Mozafari, "To Tolerate or To Impute Missing Values in V2X Communications Data?," in IEEE Internet of Things Journal, vol. 9, no. 13, pp. 11442-11452, 1 July1, 2022, doi: 10.1109/JIOT.2021.3126749.

[13] Y. Yu, J. J. Q. Yu, V. O. K. Li and J. C. K. Lam, "A Novel Interpolation-SVT Approach for Recovering Missing Low-Rank Air Quality Data," in IEEE Access, vol. 8, pp. 74291-74305, 2020, doi: 10.1109/ACCESS.2020.2988684.

[14] A. Liu, J. Lu and G. Zhang, "Concept Drift Detection: Dealing With Missing Values via Fuzzy Distance Estimations," in IEEE Transactions on Fuzzy Systems, vol. 29, no. 11, pp. 3219-3233, Nov. 2021, doi: 10.1109/TFUZZ.2020.3016040.

[15] L. Chen, G. Li, G. Huang and P. Shi, "A Missing Type-Aware Adaptive Interpolation Framework for Sensor Data," in IEEE Transactions on Instrumentation and Measurement, vol. 70, pp. 1-15, 2021, Art no. 2510515, doi: 10.1109/TIM.2021.3089783.

[16] Q. Li, H. Tan, Y. Wu, L. Ye and F. Ding, "Traffic Flow Prediction With Missing Data Imputed by Tensor Completion Methods," in IEEE Access, vol. 8, pp. 63188-63201, 2020, doi: 10.1109/ACCESS.2020.2984588.

[17] D. Xu, J. Q. Sheng, P. J. -H. Hu, T. -S. Huang and C. -C. Hsu, "A Deep Learning–Based Unsupervised Method to Impute Missing Values in Patient Records for Improved Management of Cardiovascular Patients," in IEEE Journal of Biomedical and Health Informatics, vol. 25, no. 6, pp. 2260-2272, June 2021, doi: 10.1109/JBHI.2020.3033323.

[18] J. Fan, P. Zhang, J. Chen, B. Li, L. Han and Y. Zhou, "Quantitative Estimation of Missing Value Interpolation Methods for Suomi-NPP VIIRS/DNB Nighttime Light Monthly Composite Images," in IEEE Access, vol. 8, pp. 199266-199288, 2020, doi: 10.1109/ACCESS.2020.3035408.

[19] C. Garcia, D. Leite and I. Škrjanc, "Incremental Missing-Data Imputation for Evolving Fuzzy Granular Prediction," in IEEE Transactions on Fuzzy Systems, vol. 28, no. 10, pp. 2348-2362, Oct. 2020, doi: 10.1109/TFUZZ.2019.2935688.

[20] A. Wang, J. Yang and N. An, "Regularized Sparse Modelling for Microarray Missing Value Estimation," in IEEE Access, vol. 9, pp. 16899-16913, 2021, doi: 10.1109/ACCESS.2021.3053631.

[21] X. Liu, N. Li, G. Shu and L. Min, "Generation of High-Quality Spaceborne Interrupted FMCW SAR Images via Singular Value Threshold-Based Matrix Completion," in IEEE Geoscience and Remote Sensing Letters, vol. 19, pp. 1-5, 2022, Art no. 4505905, doi: 10.1109/LGRS.2022.3157466.

[22] X. Zhu, J. Yang, C. Zhang and S. Zhang, "Efficient Utilization of Missing Data in Cost-Sensitive Learning," in IEEE Transactions on Knowledge and Data Engineering, vol. 33, no. 6, pp. 2425-2436, 1 June 2021, doi: 10.1109/TKDE.2019.2956530.

[23] M. Niemelä, S. Äyrämö and T. Kärkkäinen, "Toolbox for Distance Estimation and Cluster Validation on Data With Missing Values," in IEEE Access, vol. 10, pp. 352-367, 2022, doi: 10.1109/ACCESS.2021.3136435.

[24] Q. Ma et al., "End-to-End Incomplete Time-Series Modeling From Linear Memory of Latent Variables," in IEEE Transactions on Cybernetics, vol. 50, no. 12, pp. 4908-4920, Dec. 2020, doi: 10.1109/TCYB.2019.2906426.

[25] L. Guo, L. Renze, L. Xingyu, T. Juanjuan, C. Lei and Z. Yang, "Logging Data Completion Based on an MC-GAN-BiLSTM Model," in IEEE Access, vol. 10, pp. 1810-1822, 2022, doi: 10.1109/ACCESS.2021.3138194.