Data Fusion Method for Multi-Sensor Internet of Things Systems Including Data Imputation

Saugat Sharma , Grzegorz Chmaj , Henry Selvaraj *

Posted Date: 8 January 2026

doi: 10.20944/preprints202601.0520.v1

Keywords:

data fusion; IoT; synthetic data

 doi:10.20944/preprints202601.0520.v1

Article

Data Fusion Method for Multi-Sensor Internet of Things Systems Including Data Imputation

Saugat Sharma, Grzegorz Chmaj and Henry Selvaraj *

University of Nevada, Las Vegas

* Correspondence: henry.selvaraj@unlv.edu

Abstract

In the age of the Internet of Things (IoT), IoT devices scattered across various locations gather and store data in a decentralized manner to improve computational efficiency. Nevertheless, within IoT networks, factors such as fragile devices, challenging deployment conditions, and unreliable data transmission are raising the likelihood of data gaps, potentially having a substantial impact on the subsequent data processing resulting in failure of the system. Conventional imputation approach relies on using historical trend or sensor fusion techniques to combine information from different sensors to fill in the gaps in where information is missing. Historical trend struggles to capture new or emerging patterns, whereas using sensor fusion, even though it shows promising results, relies on information from multiple sensors from same target environment, making it vulnerable to single-point failures. This article presents an alternative strategy: using sensor-based fusion, but in this case, multiple sensors gather data from different targets independently. The architecture intelligently looks and gathers the sensor information from other location/target (multiple locations), sensing the same environmental information, learns the distribution and correlation and employ algorithm to generate synthetic data for imputing missing information. The study conducted experiments by fusing weather station data from various US locations and comparing the effectiveness of this approach to conventional methods. Further, the proposed synthetic data generation approach outperformed other algorithms when applied to the fused weather station dataset. This innovative approach mitigates the risk of single-point failures and offers a more robust solution for dealing with missing data in IoT networks.

Keywords: data fusion; IoT; synthetic data

1. Introduction

The concept of the Internet of Things (IoT) revolves around linking physical objects, including embedded systems, to enable the collection and exchange of data. IoT serves as the foundation for seamlessly integrating sensors, actuators, and communication devices, facilitating real-time data collection, and remote control of actuators [1]. This interconnected network of physical objects has created a vast ecosystem in which these objects communicate with one another to enable a wide range of applications. This has unlocked opportunities across various sectors, including smart industries, smart transportation, smart agriculture, smart healthcare, and many more [2].

All these applications of IoT in various sectors have increasingly contributed to generation of large amounts of data [3,4]. For example, in smart waste management large volumes of data are generated from various distributed IoT devices, such as cameras, RFIDs, and odor sensors and these generated data are transmitted to the cloud for further analysis [5,6]. Similarly, in smart traffic prediction systems, huge data information – vehicles speed and location, traffic data from surveillance cameras and so on – are transmitted continuously from generating source to the cloud for analysis [7,8].

However, due to delicate IoT devices and severe environmental conditions, these raw data might get lost during the transmission and storage process [9]. Moreover, this loss can also be result of

 doi:10.20944/preprints202601.0520.v1

failure of sensors/actuators, processing unit, embedded software, or from service and application levels [10]. Furthermore, growing use of renewable energy resources to power IoT devices can induce discontinuous data collection and missing data [11].

This data loss can result in significant losses, sometimes leading to serious failure. For example, in smart healthcare, if technology fails to work as intended, a patient could be injured, or sensitive personal health information may be exposed [12]. Thus, missing data results in availability of insufficient data for performing meaningful processes and analysis for the corresponding applications. Furthermore, a lack of sufficient data can result in analysis that lacks statistical significance, potentially leading to erroneous conclusions or flawed decision-making when the missing data includes crucial and sensitive features [11]. Therefore, as the effectiveness of numerous statistical and machine learning algorithms depends on having complete data, it is vital to address missing data appropriately.

There are various types of missing data mechanisms and identifying the type of missing data is crucial to find solutions to address them. Missing data has been categorized into three different types: missing completely at random (MCAR), missing at random (MAR) and missing not at random (MNAR).

MCAR signifies that the absence of data occurs in a completely random pattern, independent of any observed or unobserved values, i.e., probability of the missingness depends neither on the observed values in any variable of the dataset nor on unobserved part of dataset [11,13]. MAR indicates that missing values are solely dependent on the observed values and are unrelated to the missing values themselves, i.e., probability depends on observed value but not on unobserved values [14]. MNAR suggests that data is missing in a non-random manner, with the missing values depending on both observed and unobserved values i.e., the probability that a data point is missing depends on the value of that data point or other unobserved variables [14].

There are many ways that have been developed to handle the missing data scenarios: 1) Discard-based and 2) imputation-based methods. In discard-based approach, the missing data is directly removed from datasets [15]. Although it is simple and easy to implement, it is not applicable when missing values are large [16]. Furthermore, when the application is sensitive to time and values generated as specific time, discard method will not solve the problem. In imputation-based method, the missing value is predicted, and the predicted value is used for analysis process [16]. In other words, the synthetic data is generated in case of missing value. Conventionally, the imputation method is divided into two types 1) statistics-based imputation and 2) model-based imputation. In statistics-based imputation, usually missing data is imputed using mean, median, mode, and linear regression methods [17,18] In model-based imputation, appropriate machine learning algorithms are used to impute missing data.

All this imputation method depends on two types of datasets to generate synthetic data: 1) historical dataset and 2) dataset from multi-sensor data fusion of same target. But using historical data for data fusion is unreliable as it struggles to capture new or emerging trends and patterns that have developed since the data was collected. Multi-sensor data fusion of same target shows promising results; however, these multi-sensor data of same target sometime may not be present in all applications. Moreover, all these multi-sensors may experience single point of failure scenarios and in this case, sensors could not generate data and hence failure recovery (imputation) become difficult.

In this article, we propose an efficient data fusion method utilizing fusion of data from different target sources, to tackle the above-mentioned issues. To the best of our knowledge, proposed approach is the first ever data fusion approach. Moreover, along with data fusion approach, we have also proposed efficient data imputation methods. The major contributions of this article are summarized as follows:

1) We propose an efficient data fusion method where multi-sensor data from different targets are fused to facilitate the imputation method in case of single point of failure. Whenever the application encounter data missing and it does not have any redundant sensor from its

 doi:10.20944/preprints202601.0520.v1

application area, then it looks for similar data from other network (outside its application reach) and utilizes it if found necessary.

2) We propose a KNN with Iterative PCA based imputation method and compared its efficiency with other imputation approaches.
3) We experimented on weather station datasets from 8 different US states, performed data fusion of weather values from these states, and result showed that this approach performs on par with the conventional approach.

2. Related Work

In this section, we review some data fusion methods for data imputation – historical data-based data fusion, and Sensor/feature-based fusion from the same target location – and some data imputation methods from the perspective of statistics-based imputation, model-based imputation, K nearest neighbors-based imputation.

2.1. Historical Data-Based Data Fusion with Statistics-Based Imputation

Utilizing historical data-based data fusion methods include the use of past data for data imputation. Various statistic-based imputation methods – mean, median, mode, linear regression, and others – are then applied to historical dataset [19]. Author in [20] has applied mean and hot-decking imputations method to impute data in a real breast cancer problem. Author in [17] has compared the mean, median and mode-based imputation on gas emission data set, where mean outperformed other approaches. Mean, median and mode-based imputation replaces the missing value with the mean, median and mode of the observed values respectively. Regression based imputation replaces the missing values for each variable with the values predicted from a regression of that variable on other variables. Hot deck imputation replaces each missing value with a random draw from a 'donor pool' consisting of observed values of those variables [14][19].

The above statistics-based imputation on historical dataset imputes the missing data by taking only the missing feature into account. Although this approach favors in terms of single point of failure scenarios, they tend to produce more bias results and even sometime changing the distribution of data for dataset having larger number of gaps [17] Moreover, these statistics approaches applied to historical dataset gave poor results compared to model-based imputation approaches. Thus, utilizing only the statistics-based imputation method with historical dataset could not provide better solution in terms of accuracy and induces biases in data.

2.2. Multi Sensor-Based Data Fusion from Same Target with Model-Based Imputation

In multi sensor-based data fusion different features of same target location are collected and utilized to make a decision. Thus, various kinds of features of an environment are analyzed together to make a prediction. Since there exists correlation among these features as they represent different parameters of same environment, when any one of those features goes missing, then the remaining features of environment can be utilized to impute the missing features. For this kind of multi sensor-based data fusion technique, model-based imputation is popular since it can analyze the various environmental parameters together to decide.

Author in [21] presents that using MICE has boosted recognition accuracy from 87% to 98%. Author in [22] has proposed a framework to improve the popular multivariate imputation by chained equations (MICE) method for dealing with large data gaps. They have demonstrated the efficiency of their framework using data from continuous water quality monitoring stations in Vermont. Author in [23] has presented model selection to improve multiple imputations for handling high rate missingness in a water quality dataset. Authors have proposed a robust method for selecting the best algorithm to combine MICE to handle multiple relationships between a high number of features of interest concerned with a high rate of missingness. Thus, they express their main contribution as to improve MICE, by taking advantage of the ML models such as Random Forest (RF), K Nearest

 doi:10.20944/preprints202601.0520.v1

Neighbors (KNN), Support Vector Regression (SVR), Boosted Regression Trees (BRT) by hybridizing with MICE. They obtained that MICE-SVR gives a good trade-off in terms of performance and computing time.

Author in [24] has utilized random forest (RF) method to impute the missing weather data facilitating in critical agricultural decision making. Authors in [24] have used 8.5 years of air temperature, relative humidity, wind, and solar radiation features from Washington state to impute the missing data. Authors in [25] have predicted the corn variety yield having attribute missing data by graph neural network. Various ecological zones in China were considered to gather corn trait features – corn variety, corn strain, corn cob type and so on. Thus, corn features such as corn variety, corn strain, corn cob type, axis color, seeding lead, sheath color, and stay-green trait were chosen and labeled to create the complete dataset. Similarly, author in [26] has presented the imputation of missing precipitation data using KNN, RF, self-organizing maps (SOM), and feed-forward neural network (FNN). The missing precipitation data from Cacher watershed, Assam state of India was taken for imputation. Precipitation from various places of assam was used to create a complete dataset. Author in [27] has presented the example of imputing missing water quality data by using Deep Neural Network (DNN). The water quality data consists of features such as water temperature, pH, electric conductivity, dissolved oxygen, chlorophyll-a, and nitrate. There is some missing data in these feature sets which was imputed using DNN. Author in [28] proposes a comprehensive method to forecast AQIs. Initially, they predicted hourly ambient concentrations of $PM_{2.5}$ and PM_{10} using artificial neural network. Later it was extended to the prediction of other criteria pollutants, i.e., O_3, SO_2, NO_2, and CO to predict the AQIs. Moreover, these features consist of missing gaps, which was handled using missForest, a machine learning-based imputation technique which employed the random forest (RF).

All these methods use multi sensor data features from the same target. Thus, if there is missing data but no redundant sensor features due to single point of failure or any other possible reason, then imputation from above methods will be almost impossible.

2.3. KNN-Based Imputation

KNN-based imputation is a method used to fill in missing values in a dataset based on the K-nearest neighbors' algorithm. In this approach, for each missing value, the algorithm identifies the K nearest data points (neighbors) with known values for the feature(s) of interest. Then, it imputes the missing value by calculating a weighted average or a majority vote of the known values from these nearest neighbors. The weights are typically determined based on the distance or similarity between the missing data point and its neighbors. KNN-based imputation is commonly used in data preprocessing and is particularly effective for datasets with numerical or categorical features where missing values need to be addressed before further analysis or modeling. Authors in [29] has presented combined KNN and BiLSTM methods to address noise and variance challenges in data imputation. Initially, wind power dimension influence is assessed using Pearson correlation, and CNN extracts information. BiLSTM learns time-series data representation. Missing data is imputed using CNN-BiLSTM and KNN models, forming the final dataset. It shows the combined model improves imputation, increasing R^2 by 0.02. Authors in [30] has compared KNN, Sequential KNN (SKNN) and other statistical based imputation method to impute the missing air quality value collected from peninsular Malaysia. KNN and SKNN showed better results compared to statistical based imputation. Similarly, author in [31] has performed KNN based imputation to impute the missing weather data. The missing weather data has the various weather attributes of various weather station from Pakistan, and these attributes are utilized together to predict the missing data. Authors study in [32] assesses categorical variable imputation methods using Ugandan maternal health records. KNN imputation stands out for predicting missing values. Results reveal KNN's superior precision at multiple levels of missingness, with RF also performing well at lower missing data proportions. This study highlighted the importance of method selection based on data characteristics for effective imputation strategies.

 doi:10.20944/preprints202601.0520.v1

In our proposed KNN with iterative Principal Component analysis (PCA) approach, we have used regular KNN based imputation approach and after that Iterative PCA is applied. Thus, we will implement KNN with Iterative PCA to impute the missing data in our proposed data fusion method and compare the result with other algorithms implementing conventional data fusion method.

2.4. Proposed Data Fusion Approach

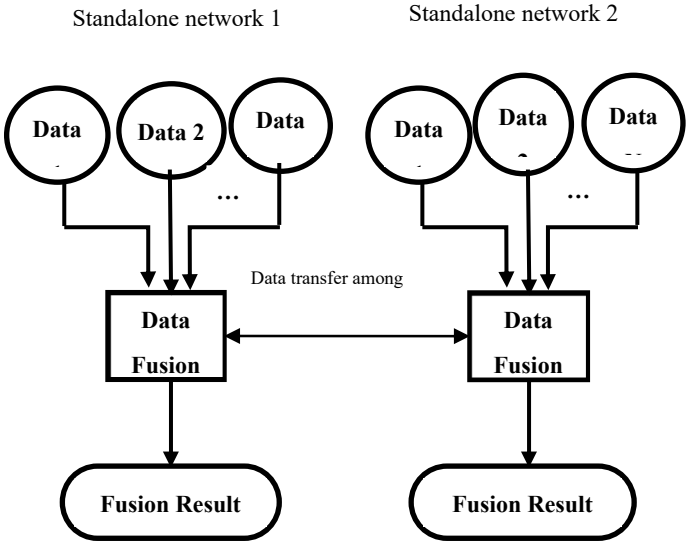
This article aims to design an imputation architecture by using proposed MICE with Iterative PCA on proposed fused dataset. We first describe the proposed data fusion method implementation.

2.5. Proposed Data Fusion Method

Sensor based data fusion method collecting information from the same target environment will have an IoT architecture composed of gateways, nodes, sensors, and other components based on the requirement of application. Sensors will be responsible for collecting various features of same target environment and when if there is data gap in one of the feature, then remaining multiple features collected from multiple sensors will be analyzed to generate the synthetic data for the missing data (feature).In our proposed system we assume these multiple features are not only available from multiple sensor sensing from same target environment but they can be located in different geographical location sensing features of different environment.

Let g_{ns} represent gateway n at location s, where $n = 1, 2, \ldots, N$; $s = 1, 2, \ldots, S$. For a given gateway g_{ns}, each gateway has p number of sensors features of type t where p can vary among gateways and the collection of these p number of features for each gateway g_{ns} can be defined as set $F_n = \{f_{nt1}, f_{nt2}, \ldots, f_{ntp}\}$, where $t = 1,2, \ldots, T$. The collection of all feature sets across all the gateways can be represented as $X_n = \{F_1, F_2, \ldots, F_N\}$. Thus, f_{122} represents feature number 2 of feature type 2 from gateway 1. In our approach, the system will look for similar feature type t from the network making sure it has significant correlation with the missing feature data. Let us assume that data feature f_{122} (f_{ntp}) i.e., feature number 2 of type 2 from dataset X_1 and gateway 1 located in geographical location 1 (g_{11}) is missing. In this case, in our proposed framework, the system will look for similar feature type 2 in dataset X_n where $n \neq 1$ and gateway g_{ns} where $n = s \neq 1$. Then these similar features ($f_{ntp,}$ if and only if $n \neq 1$ and $t == 2$) are transmitted to $g_{11.}$

For example, from the above Figure 1, when Data 1 in the first standalone network experiences missing data, the first network will search for a comparable feature in another IoT network (in this case, standalone network 2). If a matching feature is located, for instance, if Data 2 in standalone network 2 bears a resemblance to Data 1 in standalone network 1, it will be transferred from standalone network 2 to standalone network 1 to facilitate fusion.



 doi:10.20944/preprints202601.0520.v1

Figure 1. Data fusion in interconnected network architecture.

2.6. Correlation Estimation

Correlation estimation involves approximating the level of statistical association or correlation between two variables without utilizing the complete dataset. The goal of correlation estimation is to provide a reasonably accurate estimation of the correlation coefficient by relying on summary statistics or a subset of the available data, thus without having to send the complete dataset to check the correlation. In our specific scenario, we have employed summary statistics techniques.

To elaborate on summary statistics techniques, let's consider a situation where the feature f_{143} i.e., feature number 3 of type 4 from gateway 1 is not available. In this case, we compute summary statistics, namely the mean and standard deviation, for feature f_{143}. These summary statistics are then transmitted to other gateways g_{ns}. If these other gateways, for example, g_{22}, g_{33}, and g_{55}, possess similar feature types, then the summary statistics of these corresponding features are calculated within their respective gateways. Subsequently, correlation estimation is performed by comparing the summary statistics of feature f_{143} with those g_{22}, g_{33}, and g_{55}, respectively. If a correlation is identified, then the entire feature is sent to g_{11}. We have used Pearson correlation coefficients to estimate the correlation.

The Pearson correlation coefficient (commonly denoted as ϱ or r) can be estimated using summary statistics, specifically the means (μ) and standard deviations (σ) of two variables, X and Y, as follows:

$$\rho(X,Y) = \frac{\sum_{i=1}^{n}[(X_i - \mu_X) * (Y_i - \mu_Y)]}{[n * \sigma_X * \sigma_Y]} \quad (1)$$

- X_i *and* Y_i *are individual data points.*
- μ_X *and* μ_Y *are the means (averages) of X and Y, respectively.*
- σ_X *and* σ_Y *are the standard deviations of X and Y, respectively.*
- n *is the number of data points.*

2.7. KNN with Iterative PCA For Synthetic Data Generation

2.7.1. Dataset Preparation

Data fusion method described above and Figure 2 will have the fusion of same data features from multiple networks or gateways. The initial summary statistics-based correlation estimation was used to decide whether the correlation exists or not. After this, another round of correlation estimation will be conducted to select only the top 4 highly correlated features which will be later used to generate synthetic data. For simplicity, let us consider the fused dataset consists of the *N* features fused from outer gateway and missing feature *M* for which synthetic data is to be generated. Let *X* be the set of all *N* features. Let *C* be the correlation coefficient between features. The Pearson correlation coefficient same as equation (1), is used to estimate the pairwise correlation between the features. The correlation coefficient for each feature pair is calculated as shown in equation (2).

 doi:10.20944/preprints202601.0520.v1

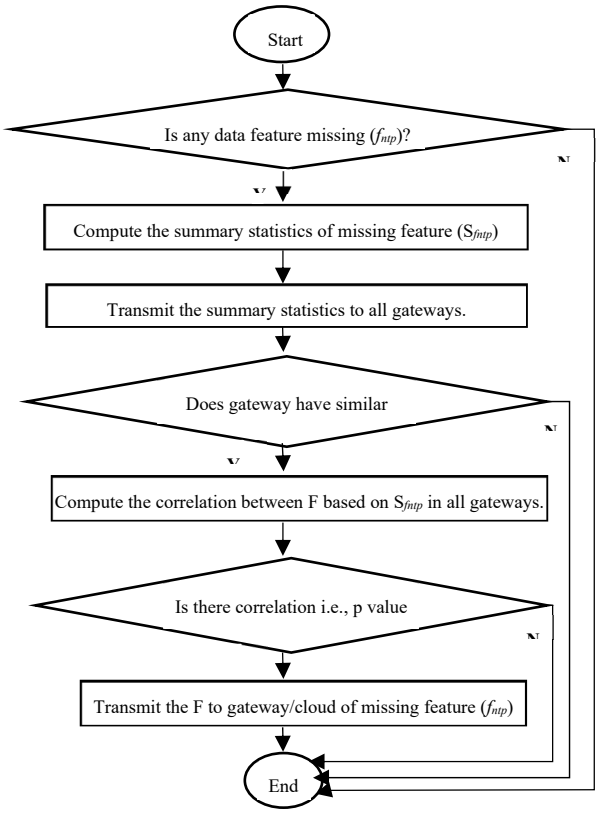


Figure 2. Data fusion flowchart.

$$C_j = \text{corr}(M, X_j) \quad \text{for } j = 1,2,\dots,N \text{ and } M_i \neq X_j \ (2)$$

After the correlation coefficient for all features with *M* is calculated, the second step includes selecting the top 4 features that exhibit the highest correlations. These top 4 features are the ones with the strongest linear associations among them. The notation for this selection can be represented as equation (3).

$$\text{Selected Features} = \arg\max_j C_j \quad \text{for } j = 1,2,\dots,4 \ (3)$$

The selected features from equation (3) represent the set of features that we want to use to generate synthetic data. The arg max$_j$ is used to find the indices of the top 4 highest correlation coefficients, denoted by C_j but only considering features other than *M*. In other words, feature *M* is excluded from this selection process.

2.7.2. Synthetic Data Generation Approach

After selecting the highly correlated features, these features will be trained to generate synthetic data for the missing feature. The algorithm for synthetic data generation is described as per algorithm 1. Let us consider our dataset *X* containing both complete and incomplete variables. We initially impute the missing value using KNN method. The dataset is divided into two parts: one that contains complete data and one that contains the missing data. Complete data is used to build imputation models. For each variable with missing data, a regression model is built using the other variables in the dataset (both complete and imputed) as predictors. The missing values for that variable are then imputed based on the predictions from the regression model. These processes are repeated for each variable with missing data in the dataset. This process continues for several iterations until convergence is achieved or a predefined stopping criteria is met. These imputed datasets are combined to create a final imputed dataset.

 doi:10.20944/preprints202601.0520.v1

Algorithm 1: Imputing missing values using KNN based iterative PCA

Require: Receive the data set X after initial KNN imputation.
Ensure: Imputed data set with missing values filled in.
1: Initialize X_0 with initial imputed values from KNN.
2: **while** not converged **do**
3: Use current estimate of X_t and compute Y_t and Z_t using PCA decomposition.
4: Y_t, $Z_t \leftarrow$ PCA Decomposition(X_t)
5: Compute the residuals between the observed values in X and the predicted values based on Y_t and Z_t.
6: $E \leftarrow X - Y_t \cdot Z_t$
7: Update the missing values in X_t by setting the missing values to their corresponding values in $Y_t \cdot Z_t$, and the observed values to their corresponding values in X.
8: $X_t \leftarrow M \cdot (Y_t \cdot Z_t) + (\neg M) \cdot X \triangleright$ Apply the mask matrix M.
9: **end while**

After initially imputing by KNN method, we then perform Iterative Principal Component Analysis (I-PCA) to run imputation only for the initially missing variables before running KNN. We initialize I-PCA by using one of the imputed datasets as our working dataset for PCA, denoted as X_{knn}. Let x_i be the *ith* row of X and let x_{ij} be the jth element of x_i. Let N be the number of rows in X and let p be the number of features in X. Let Y be the PCA decomposition of X such that $X = Y^*Z$, where Y is an N x k matrix of the k principal components of X, and Z is a k * p matrix of the corresponding loadings. Let y_{ij} be the jth element of the ith row of Y and let zij be the jth element of the ith column of Z. Let M be a binary mask matrix of the same dimensions as X, where $M_{ij} = 1$ is x_{ij} is observed and *0* if it is missing. Let X_t be the imputed dataset at iteration t.

2.7.3. Evaluation

The evaluation is based on the implementation of proposed synthetic data generation algorithm on both proposed fused datasets and with unfused datasets. For the simplicity of representing the evaluation method, let us consider a dataset, can be of both fused or unfused dataset, D has R – row and C – columns with a missing value. d_{rc} ($1 \le r \le R$, $1 \le c \le C$) is the value of the r-th row and c-th column in D. Similarly, $\widehat{D}$ denotes the imputed data for D, and $\hat{d}_{rc}$ ($1 \le r \le R$, $1 \le c \le C$) is the r-th row and c-th column value in $\widehat{D}$. However, if d_{rc} is not a missing value, then $\hat{d}_{rc} = d_{rc}$ holds. Thus, the goal is to minimize the difference (error) between $\hat{d}_{rc}$ and d_{rc}. We have used the root-mean square error (RMSE) to describe the error which is shown in equation (4).

The performance of the data fusion method and synthetic data generation method are both compared with RMSE value. The synthetic data will be generated using proposed KNN with Iterative PCA approach and its error will be obtained in both types of datasets to compare the data fusion method efficiency. Further, RMSE value will be evaluated in different synthetic data generation algorithms to compare the efficiency of proposed synthetic data generation algorithm.

$$\text{RMSE} = \sqrt{\frac{1}{RC}\sum_{r=1}^{R}\sum_{c=1}^{C}\left(d_{rc} - \widehat{d_{rc}}\right)^2} \quad (4)$$

3. Dataset

Since our goal is to analyze the proposed data fusion method with the conventional unfused data fusion method, hence, two different types of datasets must be evaluated but both facilitate to generate synthetic data for same feature. Thus, if synthetic data must be generated for missing features, let's say, F, then we will have conventional style fused dataset and the proposed style fused dataset, that will work to generate synthetic data for missing feature F. Moreover, we have datasets of both conventional and proposed approaches, working on generating synthetic data for all missing features. However, the proposed KNN with Iterative PCA is evaluated with any one type of dataset –fused or unfused.

 doi:10.20944/preprints202601.0520.v1

The weather data was gathered from a total of eight distinct locations across various U.S. states and cities, specifically Colorado, California, Arizona, Las Vegas, Washington, Salt Lake City, Texas, and Oregon. These states exhibit differences in weather patterns even when considering the same time of year. This diversity allows us to assess the effectiveness of our cross-network data fusion technique in a broad context. We obtained these datasets individually, covering the period from February to April 2023, spanning three months.

3.1. Dataset Before Fusion (DS1)

Tables 1 and 2 show a glimpse of the conventional style fused dataset under consideration. Similar datasets exist for other states as well. As observed in these tables, each dataset comprises a total of 9 features –temperature, feels like temperature, dew point, humidity, wind gust, wind speed, wind direction, cloud cover, and visibility – obtained from same target locations, i.e., Arizona and Las Vegas respectively. All these weather station datasets from all eight states share an identical number of features and same feature types. In aggregate, there are 93 data points available for each of these features. The choice to select these states was driven by the noticeable variations in weather conditions exhibited across them.

If any one of these features goes missing, then, the remaining features will be used to generate synthetic data, a conventional data fusion technique to impute missing value.

Table 1. Arizona weather dataset.

Temperature	Feels Like	Dew	Humidity	Wind Gust	Windspeed	Wind-Direction	Cloud Cover	Visibility
12	11.4	1.7	51.4	66.5	40.3	198.1	88.1	14.7
8.1	7.2	0.8	61.4	37.1	21.1	227.6	82.6	16
11.2	10.8	2.4	56.1	14.8	15.3	188.6	32.3	16
13.7	13.1	2.6	51.3	16.6	16	109.1	20.9	16
15.3	14.9	2.7	45.9	29.5	17.5	154.8	65.7	16

Table 2. Las Vegas weather dataset.

Temperature	Feels Like	Dew	Humidity	Wind Gust	Windspeed	Wind-Direction	Cloud Cover	Visibility
7.3	4.1	-0.9	57.5	55.4	31.9	200.5	68.1	15.3
9.7	7.5	-6.5	33	67.1	40.9	339.7	8.8	16
9.1	8.6	-4.6	39.2	25.9	12.5	25.5	7.1	16
11.7	11.4	-3.1	39.1	64.8	41.7	201.9	15.3	16
11.4	10.1	-3.4	35.9	64	38.7	202.3	19.3	16

3.2. Dataset After Fusion (DS2)

Tables 3 and 4 show a glimpse of the proposed fused dataset for “feels like” and “average temperature” respectively. For example, from Table 3, whenever the “feels like” feature is missing from the, let’s say gateway from Arizona, then the proposed fusion approach will gather the “feels like” feature from the other gateways from different locations. Similarly, Table 4 shows the fused dataset for average temperature.

Table 3. “Feels like” data after fusing with all 8 states and cities.

Arizona	California	Colorado	Las Vegas	Oregon	Salt Lake	Texas	Washington
11.4	4.5	-6.7	4.1	2	-2.4	24	8.3
7.2	6.5	-2.7	7.5	1.5	-2.4	21.9	12.7
10.8	8.5	-2	8.6	0.7	-1.3	15.7	4.8

 doi:10.20944/preprints202601.0520.v1

13.1	6.6	-4.7	11.4	0.8	-1.8	16.8	8.2
14.9	6.2	-1.2	10.1	0.5	-1.2	18.5	9.7

Table 4. Average temperature" dataset after fusing with all 8 states and cities.

Arizona	California	Colorado	Las Vegas	Oregon	Salt Lake	Texas	Washington
12	7	-2	7.3	3.4	2	23.7	9
8.1	8	-0.5	9.7	5.1	0.5	21.8	12.9
11.2	9.2	1.1	9.1	4.3	1.9	15.7	7.4
13.7	8.6	-0.2	11.7	4.9	2.9	16.8	9.6
15.3	8	1.7	11.4	2.6	2.5	18.5	10.2

3.3. Experiments

We performed experiments on two parts: (1) evaluating the effectiveness of the proposed data fusion technique, and (2) evaluating the effectiveness of KNN + Iterative PCA with other imputation approach. At first the effectiveness of KNN + Iterative PCA is shown and later the data fusion technique is compared based on KNN + Iterative PCA values.

3.4. Correlation Statistics Analysis

For both DS1 and DS2, we obtained the top 4 and 5 correlation of each feature with as shown in Figure 3 and 4. For both figures, correlation for "Temperature" and "Feels like" covers the highest value. Similarly, "Wind Direction" has the lowest correlation for both datasets. Likewise, "Humidity" has the moderate correlation in both datasets. The comparison of synthetic data generated is evaluated based on how effectively synthetic data was generated on proposed data fusion method when it has high. For conventional data fusion method, except "visibility", "humidity" and "feels like" feature, it shows higher correlation than proposed data fusion method. Likewise, for "cloud cover", "wind direction" and "dew" feature, conventional data fusion method shows significantly higher correlation value. For rest, it is almost on par.

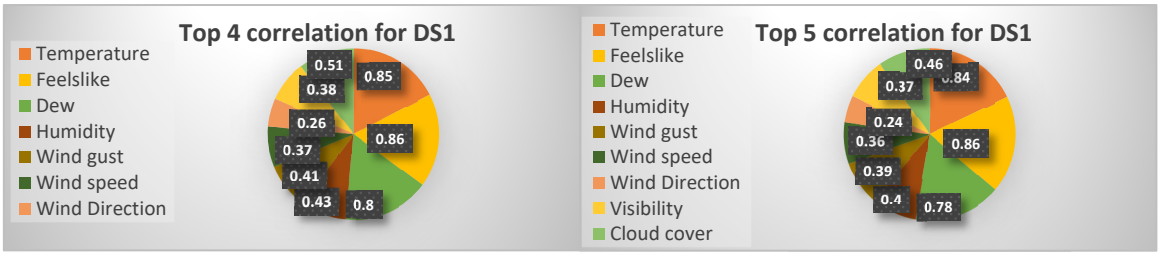


Figure 3. Top 4 and 5 correlation values for each feature for DS1.

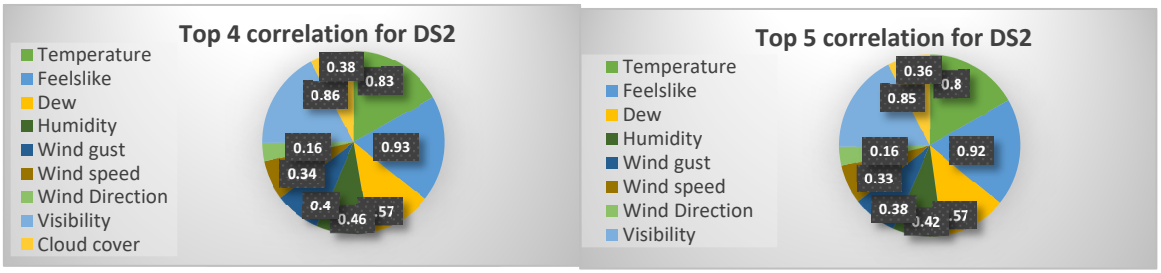


Figure 4. Top 4 and 5 correlation values for each feature for DS2.

 doi:10.20944/preprints202601.0520.v1

3.5. KNN + Iterative PCA with Other Approaches

In our study, we assessed the RMSE of generating synthetic data for missing data in our new fusion dataset, DS2. We compared the performance of KNN + Iterative PCA with various machine learning and statistical methods. Among the machine learning approaches – KNN, Random Forest (RF), RF + Iterative PCA, Decision Tree (DT), and DT + Iterative PCA. The statistical methods evaluated included MICE, MICE + Iterative PCA, Mean, and PMF. We chose "Temperature", "Humidity", and "Wind direction" features based on their correlation metrics, which ranged from highest to lowest. Our comparison covered all correlation levels, and we specifically analyzed the effectiveness of the KNN + Iterative PCA method against the other approaches.

Figure 5 illustrates the Root Mean Square Error (RMSE) comparison of the KNN + Iterative PCA method for the "Temperature" feature against all other approaches. It consistently displayed the lowest RMSE value across all geographical locations. Conversely, PMF exhibited the highest RMSE values across all geographic areas.

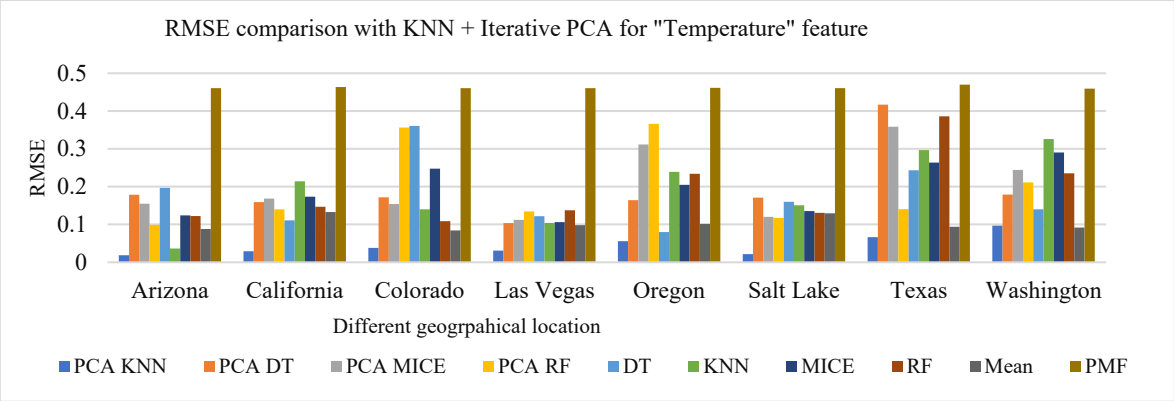
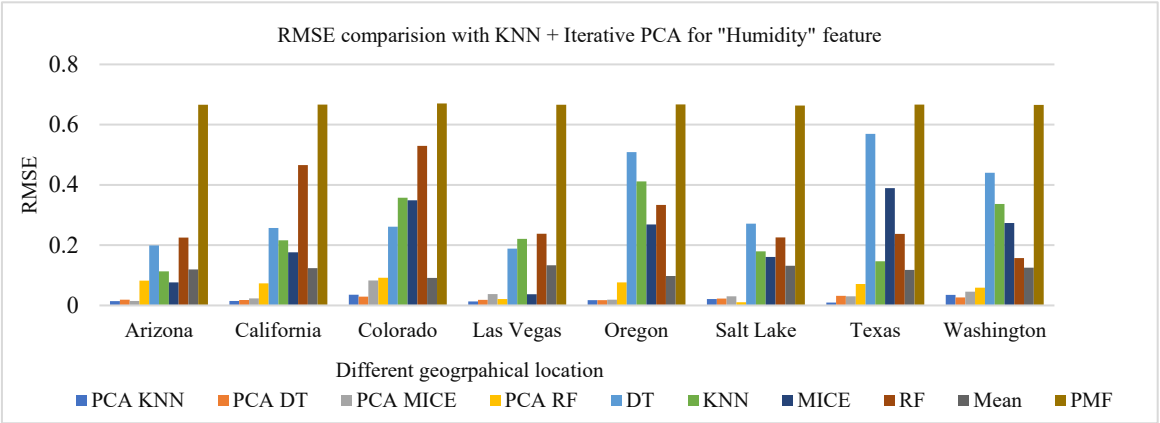


Figure 5. RMSE comparison of KNN + Iterative PCA with various Machine Learning and statistical method for "Temperature" feature.

Similarly, Figures 6 and 7 depict the RMSE comparison of KNN + Iterative PCA for the "Humidity" and "Wind direction" features, respectively, against all other methods. In both cases, KNN + Iterative PCA demonstrated lower RMSE values compared to all other methods across all geographical locations. Indeed, based on these results, it is evident that our proposed KNN + Iterative PCA approach consistently outperforms all other methods across all correlation levels. The lower RMSE values obtained for "Temperature", "Humidity", and "Wind direction" features in all geographical locations indicate the superiority of the KNN + Iterative PCA method in generating synthetic data with greater accuracy compared to alternative approaches.



 doi:10.20944/preprints202601.0520.v1

Figure 6. RMSE comparison of KNN + Iterative PCA with various Machine Learning and statistical method for "Humidity" feature.

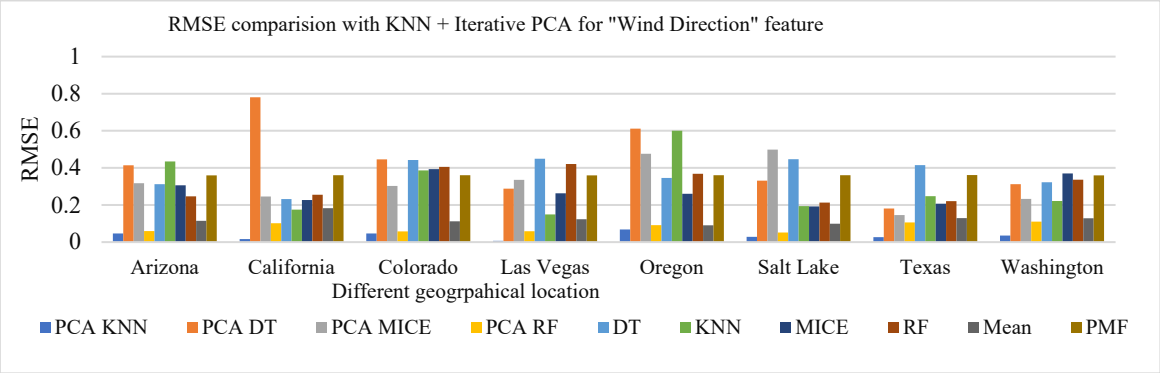


Figure 7. RMSE comparison of KNN + Iterative PCA with various Machine Learning and statistical method for "Wind direction" feature.

3.6. Comparison of Proposed Data Fusion Methods with Other Fusion Methods

The experiment includes comparison of proposed data fusion method (DS2) with conventional data fusion method (DS1). The KNN + Iterative PCA was implemented to generate synthetic data using both DS1 and DS2. The RMSE value obtained was used to compare the effectiveness of DS1 versus DS2.

We performed the experiment by randomly missing 10% and 20% of data missing and generating synthetic data for these missing values with top 4 and top 5 correlation respectively – we considered higher correlation number when the total missing percentage was high. Figure 8 to 16 shows the comparison of proposed data fusion versus conventional data fusion method for generating synthetic data for "Temperature", "Feels Like", "Dew", "Humidity", "Wind Gust", "Wind Speed", "Wind Direction", "Visibility" and "Cloud Cover" features for 10% missing data respectively. Similarly, figures 17 to 25 shows for 20% missing data for similar features. All these features have different top correlation values. For feature "Feels Like" and "Visibility", the correlation is high in proposed data fusion method than conventional data fusion method and thus proposed data fusion method shows less RMSE error. Likewise, "dew" and "cloud cover" feature has significantly lower correlation then conventional data fusion method thus it shows higher RMSE error compared to the conventional data fusion method. Whereases, in proposed data fusion method, for the rest of the feature where the correlation is higher or on par with the conventional method, it shows the lower RMSE error compared to conventional data fusion method.

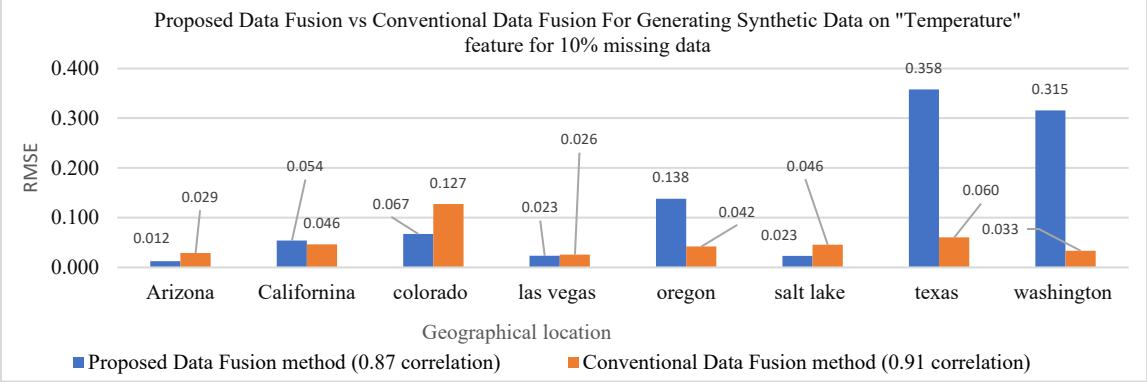


Figure 8. RMSE comparison of proposed data fusion method with conventional data fusion method for generating synthetic data for "Temperature" feature for 10% missing data for various geographical location.

 doi:10.20944/preprints202601.0520.v1

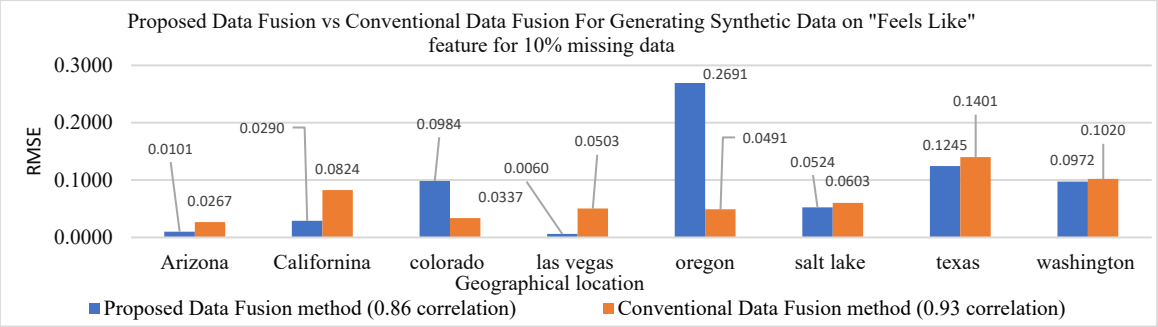


Figure 9. RMSE comparison of proposed data fusion method with conventional data fusion method for generating synthetic data for "Feels Like" feature for 10% missing data for various geographical location.

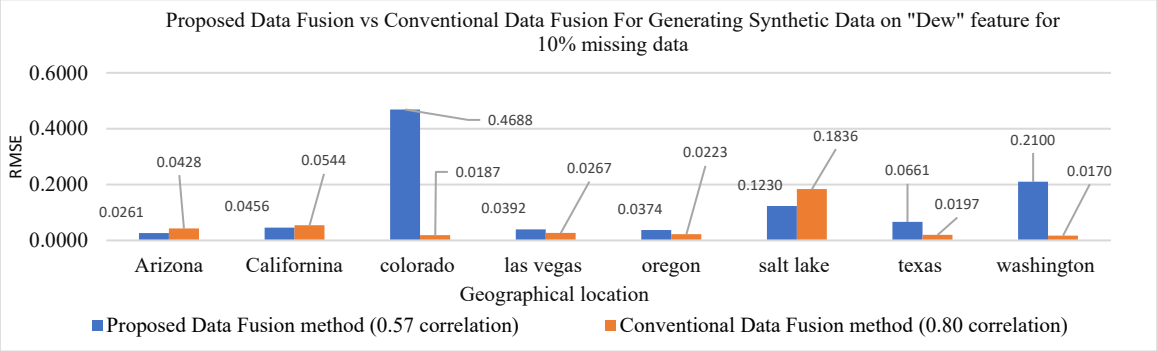


Figure 10. RMSE comparison of proposed data fusion method with conventional data fusion method for generating synthetic data for "Dew" feature for 10% missing data for various geographical location.

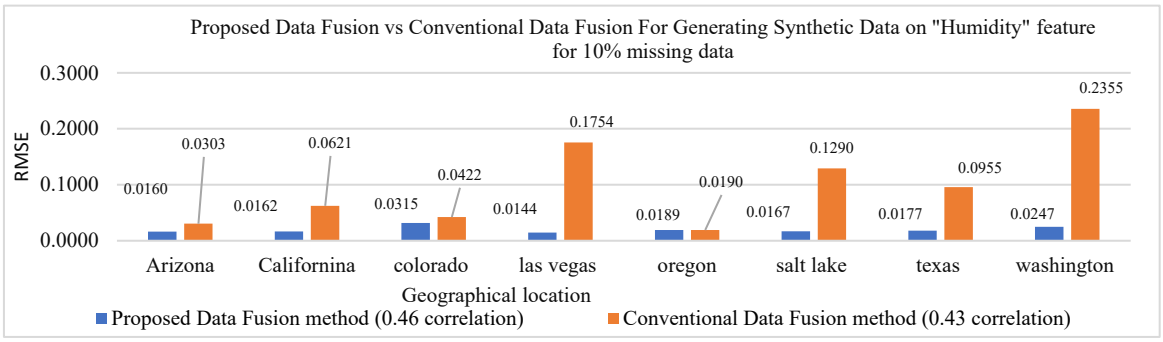
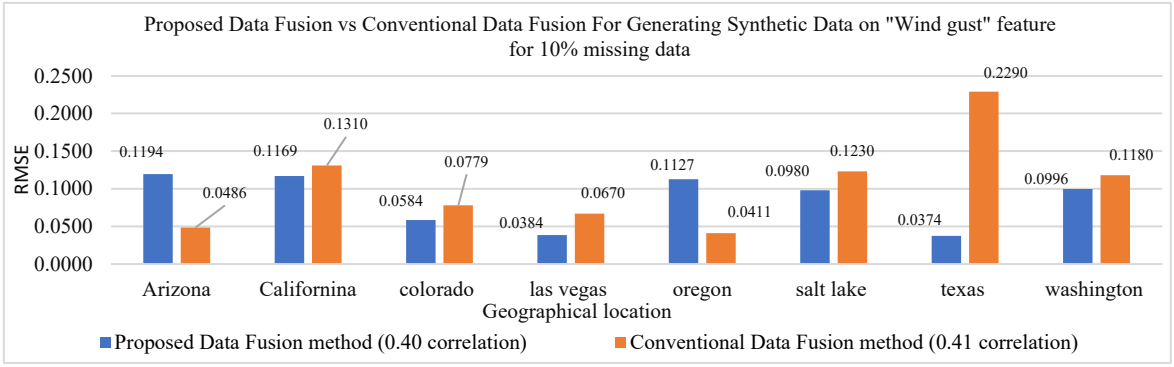


Figure 11. RMSE comparison of proposed data fusion method with conventional data fusion method for generating synthetic data for "Humidity" feature for 10% missing data for various geographical location.



 doi:10.20944/preprints202601.0520.v1

Figure 12. RMSE comparison of proposed data fusion method with conventional data fusion method for generating synthetic data for "Wind Gust" feature for 10% missing data for various geographical location.

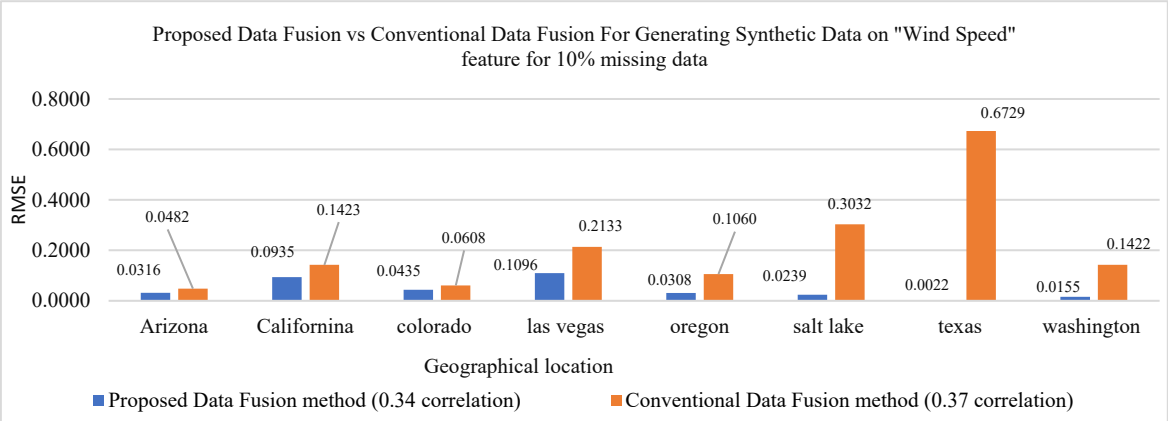


Figure 13. RMSE comparison of proposed data fusion method with conventional data fusion method for generating synthetic data for "Wind Speed" feature for 10% missing data for various geographical location.

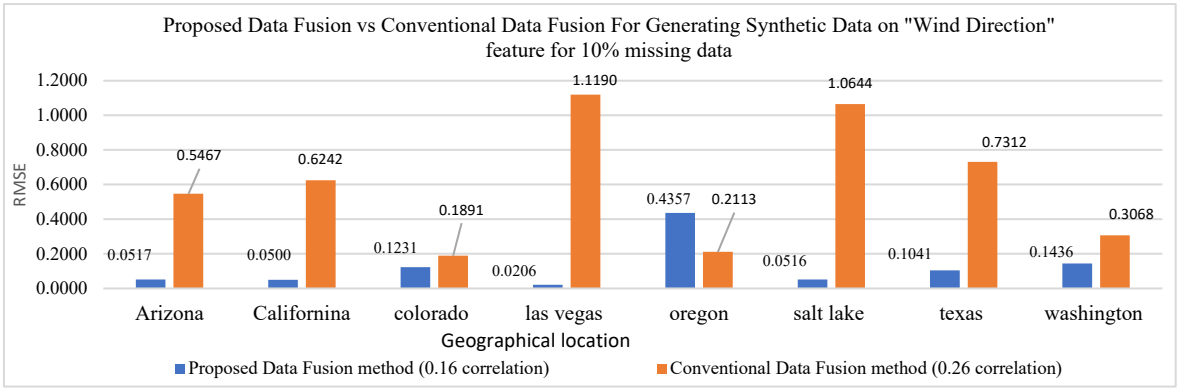


Figure 14. RMSE comparison of proposed data fusion method with conventional data fusion method for generating synthetic data for "Wind Direction" feature for 10% missing data for various geographical location.

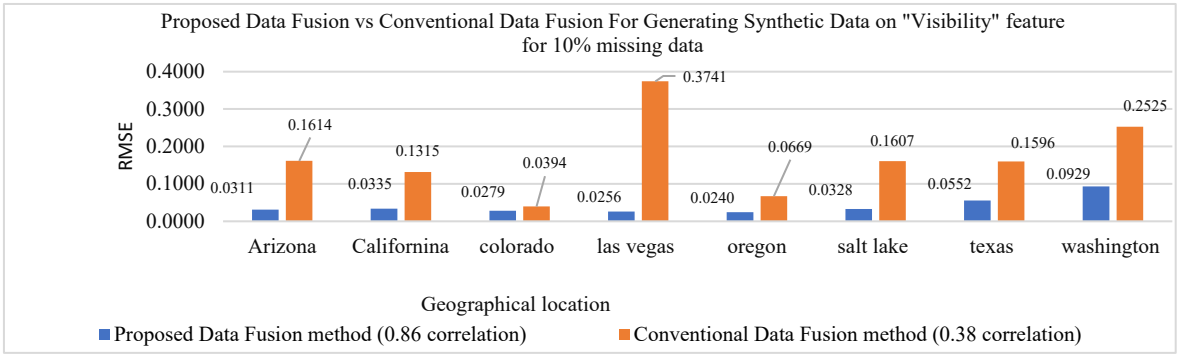


Figure 15. RMSE comparison of proposed data fusion method with conventional data fusion method for generating synthetic data for "Visibility" feature for 10% missing data for various geographical location.

 doi:10.20944/preprints202601.0520.v1

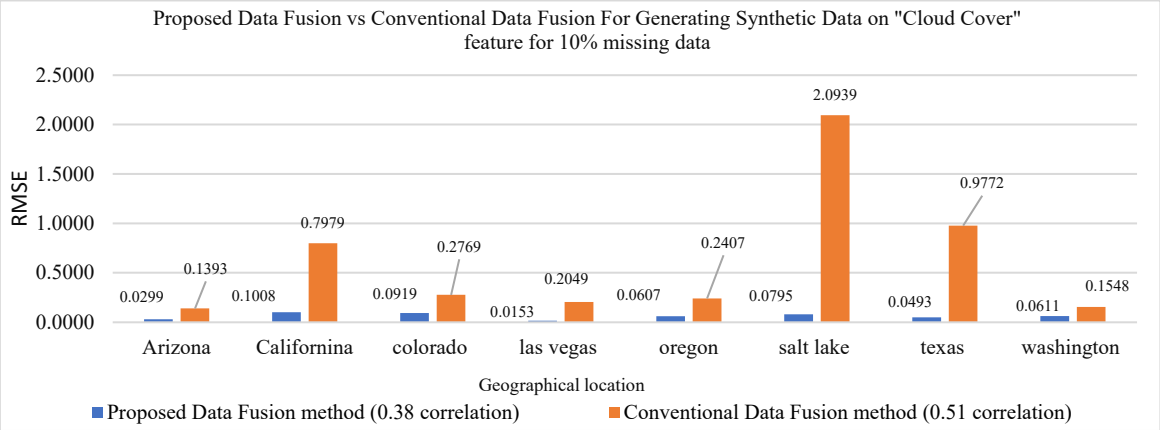


Figure 16. RMSE comparison of proposed data fusion method with conventional data fusion method for generating synthetic data for "Cloud Cover" feature for 10% missing data for various geographical location.

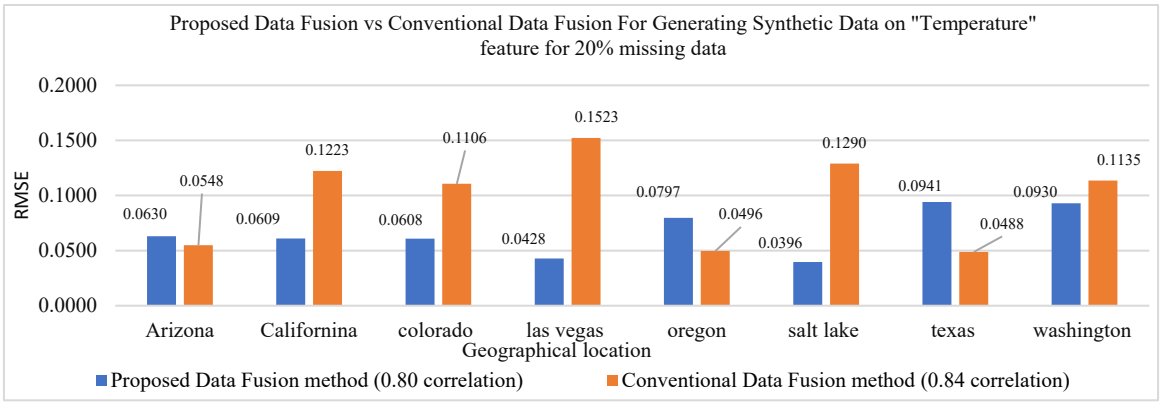


Figure 17. RMSE comparison of proposed data fusion method with conventional data fusion method for generating synthetic data for "Temperature" feature for 20% missing data for various geographical location.

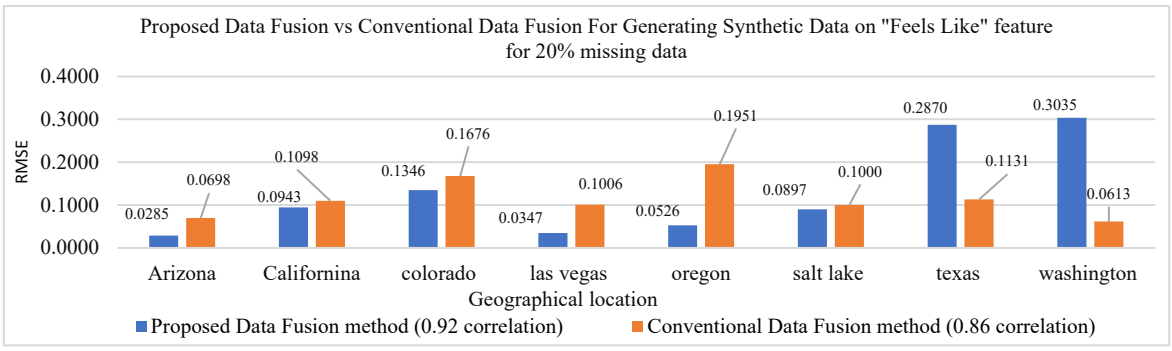


Figure 18. RMSE comparison of proposed data fusion method with conventional data fusion method for generating synthetic data for "Feels Like" feature for 20% missing data for various geographical location.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 January 2026 doi:10.20944/preprints202601.0520.v1

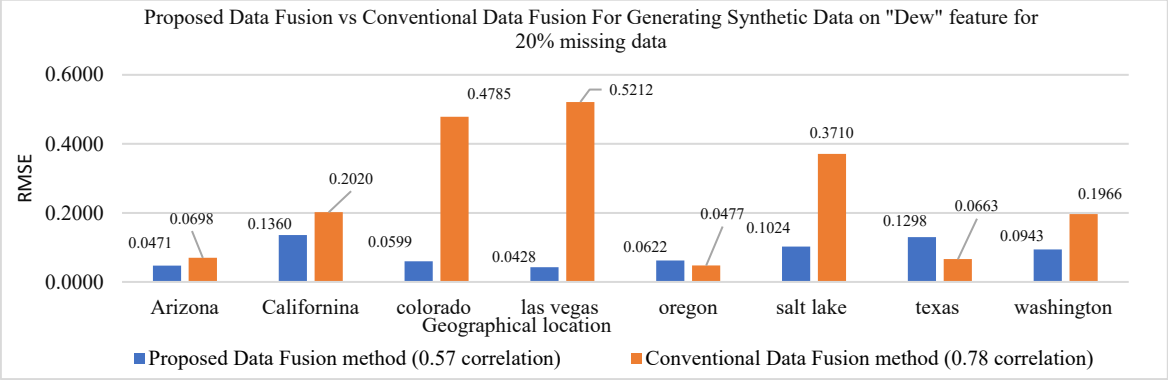


Figure 19. RMSE comparison of proposed data fusion method with conventional data fusion method for generating synthetic data for "Dew" feature for 20% missing data for various geographical location.

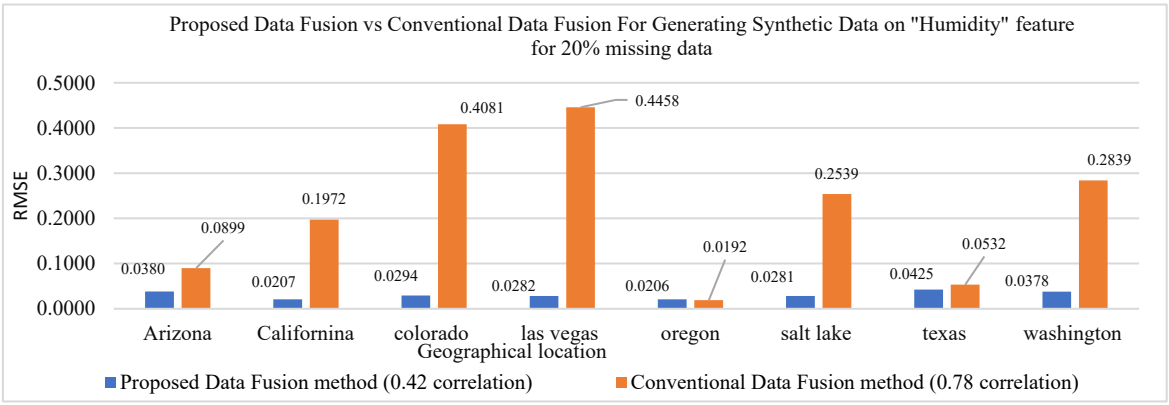


Figure 20. RMSE comparison of proposed data fusion method with conventional data fusion method for generating synthetic data for "Humidity" feature for 20% missing data for various geographical location.

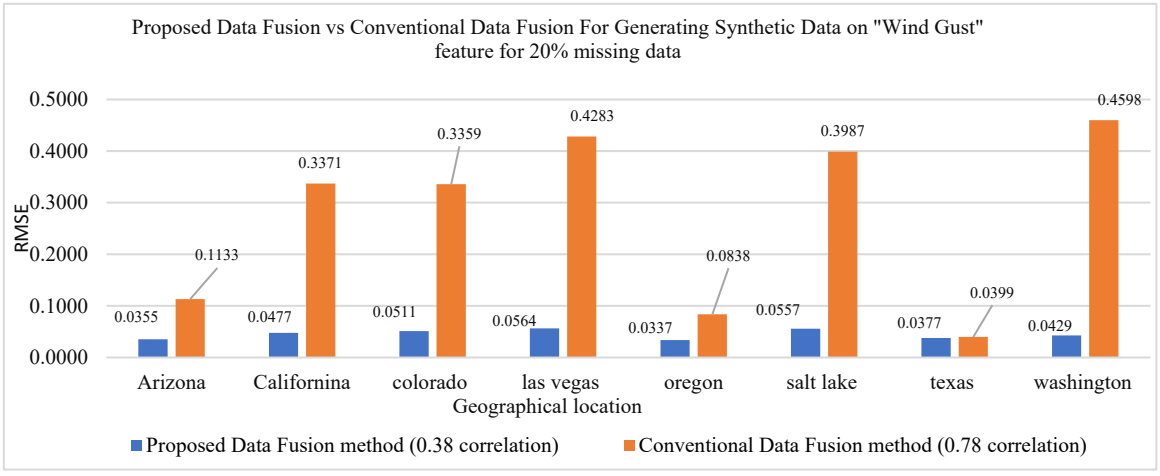


Figure 21. RMSE comparison of proposed data fusion method with conventional data fusion method for generating synthetic data for "Wind Gust" feature for 20% missing data for various geographical location.

 doi:10.20944/preprints202601.0520.v1

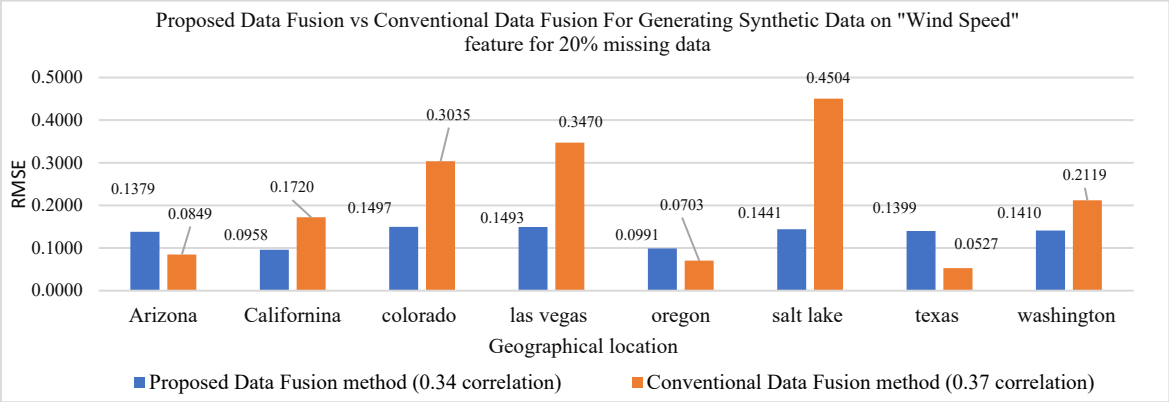


Figure 22. RMSE comparison of proposed data fusion method with conventional data fusion method for generating synthetic data for "Wind Speed" feature for 20% missing data for various geographical location.

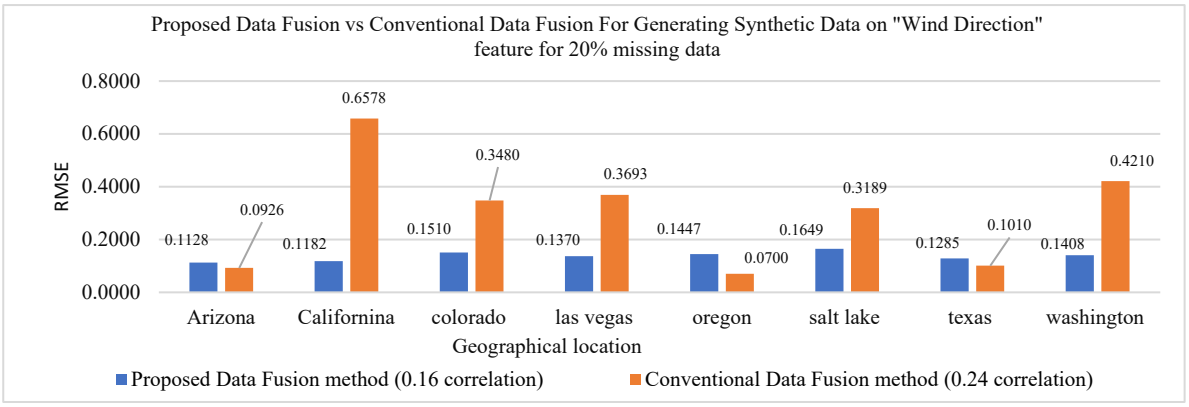


Figure 23. RMSE comparison of proposed data fusion method with conventional data fusion method for generating synthetic data for "Wind Direction" feature for 20% missing data for various geographical location.

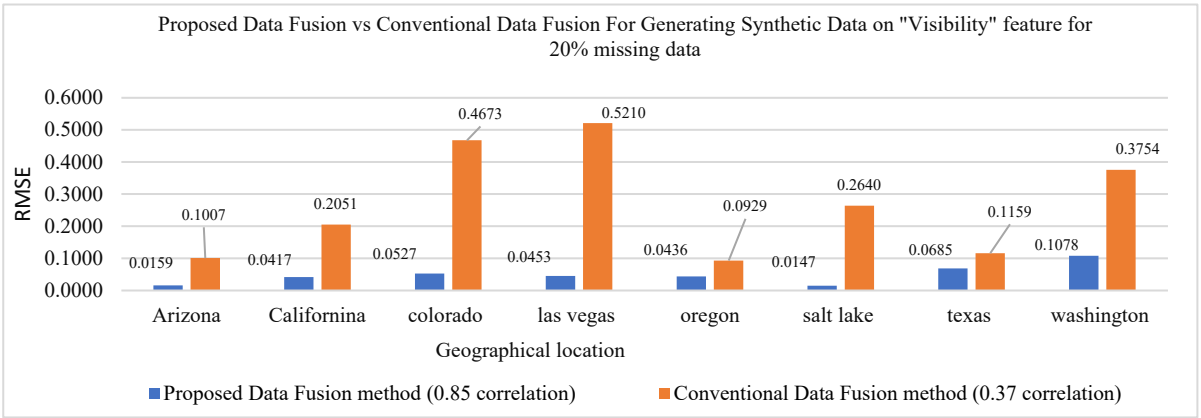


Figure 24. RMSE comparison of proposed data fusion method with conventional data fusion method for generating synthetic data for "Visibility" feature for 20% missing data for various geographical location.

 doi:10.20944/preprints202601.0520.v1

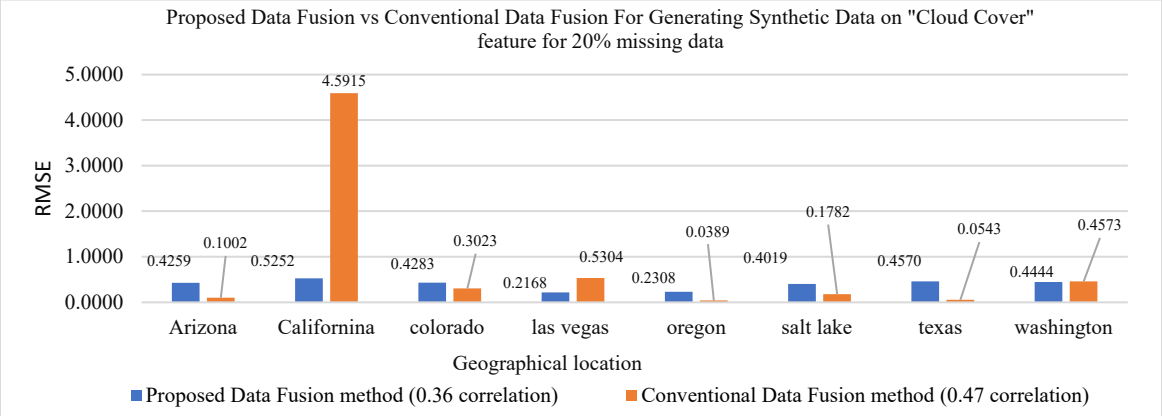


Figure 25. RMSE comparison of proposed data fusion method with conventional data fusion method for generating synthetic data for "Cloud Cover" feature for 20% missing data for various geographical location.

For 20% of missing data, proposed data fusion method outperforms conventional data fusion method where the correlation of proposed data fusion method is high or on par with the conventional method. For the "cloud cover" feature, where the proposed data fusion shows less correlation, it has higher RMSE error compared to conventional data fusion method.

Thus, from this we can say that the proposed data fusion method significantly gives less RMSE error when generating synthetic data compared to conventional methods when the correlation is high and even when it is on par with conventional method.

4. Conclusions

The study assessed the effectiveness of a proposed data fusion technique for generating synthetic data during system failures. It utilized weather data from 8 diverse locations across the US to demonstrate the potential for broader application of the fusion method. Evaluation involved comparing the accuracy of synthetic data produced during sensor data unavailability with conventional fusion methods. Results showed promise, with some instances of superior performance and others on par. Synthetic data could be valuable in scenarios with missing data or where redundant sensors are unnecessary across networks. Moreover, we compared our KNN with Iterative PCA machine learning approach with other methods. Our approach provided better result than other. Other machine learning approaches (KNN, RF, RF with Iterative PCA), were less far away in result with our approach than mean imputation and PMF. PMF was the worst because of its inefficiency to work on time series kind of data.

Further, this study demonstrated the efficiency of the proposed data fusion method through data imputation techniques. The application and benefits of this technique extend to scenarios where predictions are necessary despite the absence of redundant features to aid decision-making. Typically, additional sensors of the same data type are deployed as backups to address failures in IoT sensors or nodes, which increases resource utilization. However, with the proposed data fusion technique, relevant data from other networks can be fused, offering a more resource-efficient solution to support the decision-making process.

Data Availability: The data presented in this study are available from the National Oceanic and Atmospheric Administration (NOAA) National Centers for Environmental Information (NCEI) at https://www.ncei.noaa.gov. These data were derived from public domain resources provided by NOAA.

Acknowledgments: The authors acknowledge the use of ChatGPT (OpenAI) for language editing and grammatical improvement. The AI-assisted tool did not contribute to the study design, data processing, data fusion methodology, synthetic data generation, experimental analysis, or scientific conclusions. Full responsibility for the content of this manuscript rests with the authors.

 doi:10.20944/preprints202601.0520.v1

References

1. O. Arshi and S. Mondal, "Advancements in sensors and actuators technologies for smart cities: a comprehensive review," *Smart Construction and Sustainable Cities*, vol. 1, no. 1, 2023, doi: 10.1007/s44268-023-00022-2.
2. G. and S. H. Sharma Saugat and Chmaj, "Machine Learning Applied to Internet of Things Applications: A Survey," in *Advances in Systems Engineering*, H. and Ś. J. Borzemski Leszek and Selvaraj, Ed., Cham: Springer International Publishing, 2022, pp. 301–309.
3. R. K. Kodali, G. Swamy, and B. Lakshmi, "An implementation of IoT for healthcare," in *2015 IEEE Recent Advances in Intelligent Computational Systems, RAICS 2015*, 2016. doi: 10.1109/RAICS.2015.7488451.
4. A. Gaur, B. Scotney, G. Parr, and S. McClean, "Smart city architecture and its applications based on IoT," in *Procedia Computer Science*, Elsevier B.V., 2015, pp. 1089–1094. doi: 10.1016/j.procs.2015.05.122.
5. A. Hussain et al., "Waste management and prediction of air pollutants using IoT and machine learning approach," *Energies (Basel)*, vol. 13, no. 15, Aug. 2020, doi: 10.3390/en13153930.
6. A. Medvedev, P. Fedchenkov, A. Zaslavsky, T. Anagnostopoulos, and S. Khoruzhnikov, "Waste management as an IoT-enabled service in smart cities," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Springer Verlag, 2015, pp. 104–115. doi: 10.1007/978-3-319-23126-6_10.
7. M. Mangla, S. Satpathy, B. Nayak, and S. Mohanty Nandan, "Integration of Cloud Computing with Internet of Things." [Online]. Available: https://onlinelibrary.wiley.com/doi/
8. G. Fusco, C. Colombaroni, L. Comelli, and N. Isaenko, MT-ITS : 2015 International Conference on Models and Technologies for Intelligent Transportation Systems : Budapest University of Technology and Economics (BME), Faculty of Transport Engineering and Vehicle Engineering, Department of Transport Technology and Economics : 3-5 June 2015, Budapest. Institute of Electrical and Electronics Engineers, 2015. doi: 10.1109/MTITS.2015.7223242.
9. D. Turner, K. Levchenko, A. C. Snoeren, and S. Savage, Proceedings of the ACM SIGCOMM 2010 conference. ACM Digital Library, 2013. https://doi.org/10.1145/1851182.1851220
10. M. Melo and G. Aquino, "The Pathology of Failures in IoT Systems," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Springer Science and Business Media Deutschland GmbH, 2021, pp. 437–452. doi: 10.1007/978-3-030-87013-3_33.
11. L. Li, Y. Wang, H. Wang, S. Hu, and T. Wei, "An Efficient Architecture for Imputing Distributed Data Sets of IoT Networks," *IEEE Internet Things J*, vol. 10, no. 17, pp. 15100–15114, Sep. 2023, doi: 10.1109/JIOT.2023.3264609.
12. E. Ismagilova, L. Hughes, N. P. Rana, and Y. K. Dwivedi, "Security, Privacy and Risks Within Smart Cities: Literature Review and Development of a Smart City Interaction Framework," *Information Systems Frontiers*, vol. 24, no. 2, pp. 393–414, Apr. 2022, doi: 10.1007/s10796-020-10044-1.
13. A. Jadhav, D. Pramod, and K. Ramanathan, "Comparison of Performance of Data Imputation Methods for Numeric Dataset," *Applied Artificial Intelligence*, vol. 33, no. 10, 2019, doi: 10.1080/08839514.2019.1637138.
14. S. Rässler, D. B. Rubin, and E. R. Zell, "Imputation," Jan. 2013. doi: 10.1002/wics.1240.
15. F. Barzi and M. Woodward, "Imputations of missing values in practice: Results from imputations of serum cholesterol in 28 cohort studies," *Am J Epidemiol*, vol. 160, no. 1, 2004, doi: 10.1093/aje/kwh175.
16. T. Aljuaid and S. Sasi, "Proper imputation techniques for missing values in data sets," in *Proceedings of the 2016 International Conference on Data Science and Engineering, ICDSE 2016*, 2017. doi: 10.1109/ICDSE.2016.7823957.
17. S. Sharma, G. Chmaj, and H. Selvaraj, "Sensor Data Restoration in Internet of Things Systems Using Machine Learning Approach," in *Lecture Notes in Networks and Systems*, 2023. doi: 10.1007/978-3-031-27470-1_3.
18. S. Sharma, G. Chmaj, and H. Selvaraj, "Applying Machine Learning to Minimize the Impact of Sensor Failures to RTOS Based Internet of Things Systems," in *Lecture Notes in Networks and Systems*, 2023. doi: 10.1007/978-3-031-40579-2_14.
19. Z. Zhang, "Missing data imputation: Focusing on single imputation," *Ann Transl Med*, vol. 4, no. 1, 2016, doi: 10.3978/j.issn.2305-5839.2015.12.38.

 doi:10.20944/preprints202601.0520.v1

20. J. M. Jerez et al., "Missing data imputation using statistical and machine learning methods in a real breast cancer problem," *Artif Intell Med*, vol. 50, no. 2, 2010, doi: 10.1016/j.artmed.2010.05.002.
21. S. Guguloth, A. Telu, U. Sairam, and S. Voruganti, "Activity Recognition in Missing Data Scenario Using MICE Algorithm," in *Lecture Notes in Networks and Systems*, Springer Science and Business Media Deutschland GmbH, 2023, pp. 844–851. doi: 10.1007/978-3-031-27524-1_82.
22. R. Wu, S. D. Hamshaw, L. Yang, D. W. Kincaid, R. Etheridge, and A. Ghasemkhani, "Data Imputation for Multivariate Time Series Sensor Data With Large Gaps of Missing Data," *IEEE Sens J*, vol. 22, no. 11, pp. 10671–10683, Jun. 2022, doi: 10.1109/JSEN.2022.3166643.
23. R. Ratolojanahary, R. Houé Ngouna, K. Medjaher, J. Junca-Bourié, F. Dauriac, and M. Sebilo, "Model selection to improve multiple imputation for handling high rate missingness in a water quality dataset," *Expert Syst Appl*, vol. 131, pp. 299–307, Oct. 2019, doi: 10.1016/j.eswa.2019.04.049.
24. J. P. Boomgard-Zagrodnik and D. J. Brown, "Machine learning imputation of missing Mesonet temperature observations," *Comput Electron Agric*, vol. 192, Jan. 2022, doi: 10.1016/j.compag.2021.106580.
25. F. Yang et al., "Prediction of corn variety yield with attribute-missing data via graph neural network," *Comput Electron Agric*, vol. 211, Aug. 2023, doi: 10.1016/j.compag.2023.108046.
26. Y. Zhang and P. J. Thorburn, "Handling missing data in near real-time environmental monitoring: A system and a review of selected methods," *Future Generation Computer Systems*, vol. 128, pp. 63–72, Mar. 2022, doi: 10.1016/j.future.2021.09.033.
27. H. Alkabbani, A. Ramadan, Q. Zhu, and A. Elkamel, "An Improved Air Quality Index Machine Learning-Based Forecasting with Multivariate Data Imputation Approach," *Atmosphere (Basel)*, vol. 13, no. 7, Jul. 2022, doi: 10.3390/atmos13071144.
28. C. Wang, B. Ren, X. Li, and L. Chen, "A CNN-BiLSTM and KNN based missing data imputation for wind power generation forecasting," in *2023 IEEE 6th International Electrical and Energy Conference, CIEEC 2023*, 2023. doi: 10.1109/CIEEC58067.2023.10166993.
29. N. A. Zainuri, A. A. Jemain, and N. Muda, "A comparison of various imputation methods for missing values in air quality data," *Sains Malays*, vol. 44, no. 3, pp. 449–456, Mar. 2015, doi: 10.17576/jsm-2015-4403-17.
30. H. Nida, M. Kashif, M. I. Khan, and M. Ghamkhar, "Comparison of missing data imputation methods using weather data," *Pak J Agric Sci*, vol. 60, no. 2, pp. 327–336, 2023, doi: 10.21162/PAKJAS/23.228.
31. S. M. Memon, R. Wamala, and I. H. Kabano, "A comparison of imputation methods for categorical data," *Inform Med Unlocked*, vol. 42, 2023, doi: 10.1016/j.imu.2023.101382.