International Journal of Internet, Broadcasting and Communication Vol.13 No.3 63-72 (2021)
http://dx.doi.org/10.7236/IJIBC.2021.13.3.63

IJIBC 21-3-9

A Probabilistic Tensor Factorization approach for Missing Data Inference in Mobile Crowd-Sensing

Shathee Akter, Seokhoon Yoon*

Ph.D Candidate, Department of Electrical, Electronic, and Computer Engineering, University of Ulsan, Ulsan, Korea
Professor, Department of Electrical, Electronic, and Computer Engineering, University of Ulsan, Ulsan, Korea
erittrashathee@gmail.com, seokhoonyoon@ulsan.ac.kr

Abstract

Mobile crowd-sensing (MCS) is a promising sensing paradigm that leverages mobile users with smart devices to perform large-scale sensing tasks in order to provide services to specific applications in various domains. However, MCS sensing tasks may not always be successfully completed or timely completed for various reasons, such as accidentally leaving the tasks incomplete by the users, asynchronous transmission, or connection errors. This results in missing sensing data at specific locations and times, which can degrade the performance of the applications and lead to serious casualties. Therefore, in this paper, we propose a missing data inference approach, called missing data approximation with probabilistic tensor factorization (MDI-PTF), to approximate the missing values as closely as possible to the actual values while taking asynchronous data transmission time and different sensing locations of the mobile users into account. The proposed method first normalizes the data to limit the range of the possible values. Next, a probabilistic model of tensor factorization is formulated, and finally, the data are approximated using the gradient descent method. The performance of the proposed algorithm is verified by conducting simulations under various situations using different datasets.

Keywords: *mobile crowd-sensing, missing data inference, probabilistic tensor factorization, gradient descent*

1. Introduction

Mobile crowd-sensing (MCS), a versatile sensing platform, exploits mobile users and their smart devices to collect sensing data (such as information about traffic conditions, air pollution, crowd and noise level) from their surrounding environments, which is further aggregated and analyzed in a cloud server for large-scale intelligence extraction [1]. In MCS, users need to visit the location of the sensing tasks intentionally or unintentionally to complete the task, i.e., to collect the sensing data [1, 2]. However, there can be scenarios where users do not visit the task location as expected deliberately or accidentally, although they are rewarded (in monetary or entertainment form), resulting in the missing data entries. The missing data values may affect

Manuscript Received: June. 12, 2021 / Revised: June. 18, 2021 / Accepted: June. 22, 2021
Corresponding Author: seokhoonyoon@ulsan.ac.kr
Tel: +82-52-259-1403, Fax: +82-52-259-1687
Associate Professor, Department of Electrical, Electronic, and Computer Engineering, University of Ulsan, Korea

the data quality and intelligence extraction process that is used for specific applications (e.g., monitoring senior citizen's wellness [3], using user experience to improve mobile apps such as travel apps [4], making intelligent business model based on consumer behavior [5], detecting micro-dust level in outdoor facilities [6], and smart farming [7]). Especially, the missing sensing values may bring irrecoverable consequences for emergency applications, where data should be collected and transmitted within a specific time. For example, if information about gas leakage in an area is not received within a specific time or missing completely, it will lead to a disastrous situation. However, data may not be successfully delivered in time or not at all because of asynchronous transmission, relay delay, connection errors, smart devices running out of battery, or mechanical errors. Thus, to avoid these issues and extract better and meaningful information, the missing values need to be recovered, which has been investigated by a very few studies [8, 9] in MCS.

In [8], the authors used the K-Nearest Neighbors approach and spatial-temporal correlation between observed data to deduce the missing entries, which is called K-Nearest Neighbors-Spatio-Temporal (KNN-ST). However, the limitation of the KNN based approaches is that the performance decreases as the number of missing data increases [9]. The authors in [10] aimed to infer the missing value by exploiting the low-rank-based matrices, i.e., they employ compress sensing and matrix factorization techniques to minimize the error between observed and predicted entries.

Matrix factorization (MF) is one of the promising techniques for missing data inference that has been employed by several studies in other research fields. For instance, [11] proposed an extension of the MF, called probabilistic matrix factorization (PMF), to approximate the missing observations in collaborative filtering. The MF uses the inner product of two feature matrices to infer the missing value [12] and can only take two criteria into account, *e.g.,* sensing value collected at a specific venue and time, whereas in many cases, sensing data from multi-criteria viewpoint are available or data may depend on more than three objects (for example, sensing data may change depending on the latitude, longitude, and time). Furthermore, inferring missing data using multi-dimensional factorization, i.e., tensor factorization (TF), is shown to be more accurate than matrix factorization [13]. However, TF tends to overfit the data if regularization parameters are not tuned properly. To reduces the pain of parameters tuning, TF is extended to the probabilistic tensor factorization (PTF) using the Bayesian technique [14].

Therefore, in this paper, to minimize the error between actual and predicted sensing data in MCS, we propose a missing data recovery algorithm called missing data approximation with probabilistic tensor factorization method (MDI-PTF) based on the PTF. The proposed method uses feature scaling and probabilistic modeling to limit the range of possible values of feature matrices and improve the performance quality of the data inference.

The rest of this paper is organized as follows: Section 2 presents the modeling of the system and problem definition. In Section 3, the missing data recovery algorithm with probabilistic tensor factorization is described. The simulation results and the parameters used for the simulation are presented. Finally, Section 5 concludes the paper.

2. Problem Definition

Assume that there are N points of interest (PoIs) from which sensing data will be collected by the mobile users. i and j denote the location of the PoI according to the x-axis and y-axis, respectively, where $1 \leq i,\ j \leq N$. From each location or PoI (i, j), data is collected periodically, and the total sensing period is divided into total T_{ij} number of timeslots. In this paper, T_{ij} varies according to locations since the sensing period and length of a timeslot τ_{ij} can be different in different locations. Thus, the total monitoring period contains K timeslots,

where the length of each timeslot k $(1 \leq k \leq K)$ is equal to $\min_{ij} \tau_{ij}$. Note that at each timeslot, sensing data is collected at most once. x_{ijk} denotes the sensing data collected from location (i, j) at timeslot k. Then, the definition of the data tensors and the problem formulation used in this paper are given below.

Definition 1 Complete Sensing tensor: a tensor where each data point is successfully collected, i.e., no data is missing, and it is denoted by $X \in R^{N \times N \times K}$.

Definition 2 Selection or Binary Tensor: a binary tensor that indicates which data points are missing and denoted as $B \in B^{N \times N \times K}$. B is defined as:

$$B = [b_{ijk}]_{N \times N \times K} = \begin{cases} 0, & if\ x_{ijk}\ is\ missing \\ 1, & Otherwise \end{cases} \quad (1)$$

Definition 3 Observed Data Tensor: an $N \times N \times K$ tensor, which contains the raw data collected by the users at each location and timeslot, and denoted by S, where $S = B \circ X$.

Definition 4 Reconstructed Tensor: the tensor which is reconstructed by deducing the missing data points using collected sensing data and defined as $\hat{X}$.

Problem Definition: Given S, the missing data recovery (MDR) problem is to find tensor $\hat{X}$ such that the error between $\hat{X}$ and X is minimized, i.e.,

$$\min \left\| X - \hat{X} \right\|_F \quad (2)$$

where $\|.\|_F$ is the Frobenius norm and calculated as follows: $\|X\|_F = \sqrt{\sum_{i,j,k} (x_{ijk})^2}$

3. Missing Data Approximation with Probabilistic Tensor Factorization

In this section, the proposed PTF-based missing data recovery method, namely MDI-PTF, is presented. PTF, which is the probabilistic modeling of the tensor factorization technique [15], uses low-rank matrices and their linear combinations to reconstruct the missing values, where the approximated values show high precision [16]. However, the execution time of PTF is usually high. Thus, we first normalize or rescale the data within the range of 0 (lowered bound of the possible measured data) and 1 (upper bound of the possible measured data), which reduces the complexity by limiting the range of possible values.

After data scaling, assume that the tensor S has low-rank features and can be estimated by using outer-product of low-rank matrices $U \in R^{D \times N}$, $V \in R^{D \times N}$, and $T \in R^{D \times T}$, i.e.,

$$S \approx [\![U, V, T]\!] = \sum_{d=1}^{D} U_d \otimes V_d \otimes T_d \quad (3)$$

where U_d, V_d, and T_d represent the d^{th} row of U, V, and T respectively. Each entry of tensor S can be expressed as:

$$S_{ijk} \approx < U_i, V_j, T_k > \equiv \sum_{d=1}^{D} U_{di} V_{dj} T_{dk} \quad (4)$$

where U_i, V_j, and T_k are the i^{th}, j^{th}, and k^{th} columns of U, V, and T, respectively. Now, the problem is to find

$U,\ V$, and T, which will enable recovering the missing entries.

A solution to this problem is to estimate $U,\ V$, and T by using a probabilistic approach, *e.g.,* by maximizing a posteriori estimate. Therefore, the prior distributions on $U,\ V$, and T are given by:

$$P(U|\mu_U, \sigma_U^2) = \prod_{i=1}^{N} N(U_i|\mu_U, \sigma_U^2) \tag{5}$$

$$P(V|\mu_V, \sigma_V^2) = \prod_{j=1}^{N} N(V_j|\mu_V, \sigma_V^2) \tag{6}$$

$$P(T|\mu_t, \sigma_t^2) = \prod_{k=1}^{K} N(T_k|\mu_t, \sigma_t^2) \tag{7}$$

where $N(.|.,.)$ is the Gaussian distribution. $\mu_U,\ \mu_V$, and μ_t are the mean and $\sigma_U^2,\ \sigma_V^2$, and σ_t^2 are the variance. Then, the conditional distribution over observed sensing entries can be represented as follows:

$$P(S|U,V,T,\sigma^2) = \prod_{i=1}^{N}\prod_{j=1}^{N}\prod_{k=1}^{K}[N(S_{ijk}| < U_i,\ V_j,\ T_K >,\ \sigma^2)]^{b^{ijk}} \tag{8}$$

where $< U_i,\ V_j,\ T_K >$ is mean and σ^2 is the variance.

Given distribution over observed entries and prior distribution, we can approximate latent feature matrices $U,\ V$, and T by maximizing the log-posterior distribution. Maximizing the log-posterior is usually equivalent to minimizing the following regularized loss function:

$$L = \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{K} b_{ijk}\,(S_{ijk} - < U_i, V_j, T_k >)^2 + \frac{\lambda_u}{2}\sum_{i=1}^{N}\|U_i - \mu_U\|_F^2 + \frac{\lambda_V}{2}\sum_{j=1}^{N}\|V_j - \mu_V\|_F^2 + \frac{\lambda_k}{2}\sum_{k=1}^{K}\|T_k - \mu_t\|_F^2 \tag{9}$$

where $\lambda_u = \frac{\sigma^2}{\sigma_U^2}$, $\lambda_v = \frac{\sigma^2}{\sigma_V^2}$, and $\lambda_t = \frac{\sigma^2}{\sigma_t^2}$.

The loss function L is optimized by using a gradient descent algorithm in the proposed method. For easier learning, parameters of the loss function are set to the fixed value, and $U,\ V$, and T are updated as follows:

$$U_i = U_i + \eta\frac{\delta L}{\delta U_i} \tag{10}$$

$$V_j = V_j + \eta\frac{\delta L}{\delta V_j} \tag{11}$$

$$T_k = T_k + \eta\frac{\delta L}{\delta T_k} \tag{12}$$

where η is the learning step.

The flow chart of the proposed method is given in Fig. 1 and explained below. The proposed method, namely MDI-PTF, starts with normalizing the observed data in tensor S, as shown in step 1 in Fig. 1. After the normalization, in step 2, three matrices U, V, and T are generated using given Gaussian prior distribution (given in Eqs. (5), (6), and (7) respectively), which satisfies the Eqs. (3) and (4). Then, the root mean square error (RMSE) is calculated between original tensor S and predicted tensor $\hat{X}$ in step 3, and the RMSE is given by:

$$RMSE = \sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{K} b_{ijk}\,(S_{ijk} - < U_i, V_j, T_k >)^2 \quad (3).$$

Note that the predicted tensor $\hat{X}$ is constructed by using the estimated U, V, and T, i.e., $\hat{X} = [\![U, V, T]\!]$. In step 4, the RMSE between tensor S and $\hat{X}$ is compared with a threshold value, i.e., if RMSE is higher than the threshold ϵ, U, V, and T are updated using gradient descent algorithm following the gradients from Eqs. (10), (11), and (12) respectively (step 4.1); otherwise, the original tensor S is completed with the missing values obtained from $\hat{X}$ (step 4.2). The main loop (from step 3 to 4.2) is repeated until it attains a given number of iteration or RMSE is less than ϵ.

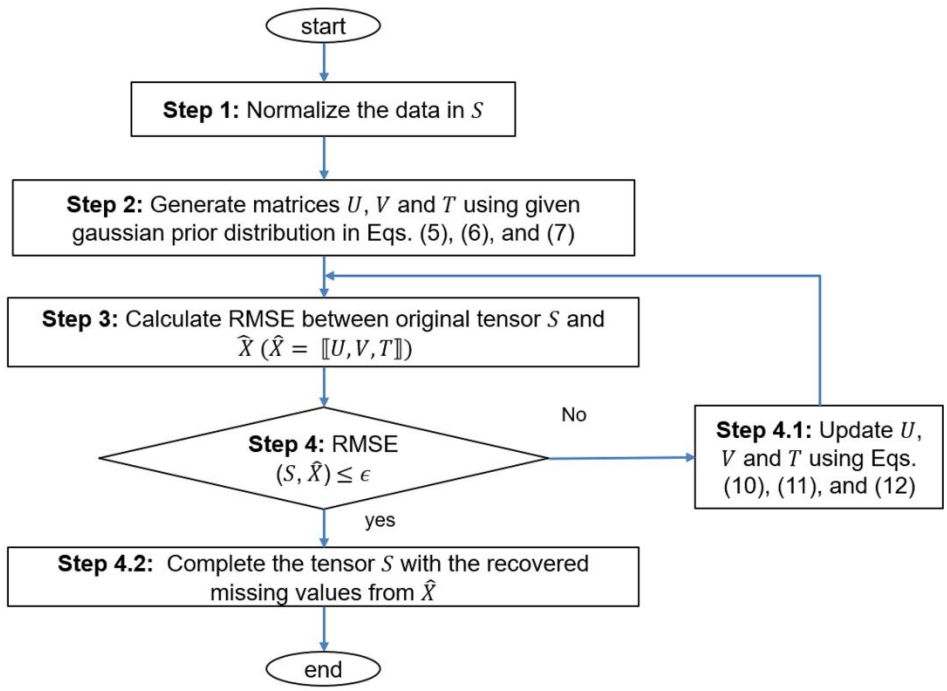

Figure 1. Flow-chart of the MDI-PTF

4. Result and Analysis

In this section, the simulation setup used for evaluating the performance of the MDI-PTF, and the results are presented. The proposed method is evaluated using three different datasets (i.e., temperature, light, and humidity dataset), which are obtained from the Intel Lab dataset [17]. Intel Lab dataset contains weather data (such as temperature, light, and humidity) and is collected by sensors deployed in the Intel Berkeley Research

Laboratory from February 28, 2004, to April 5, 2004. The data are collected by 54 sensors every 30 seconds. We have extracted temperature, light, and humidity data of one day from 1 am to 7 am and made three corresponding datasets.

Using each dataset, the effect of different parameters such as different number of locations [5, 10, 15, 20, 30], minimum length of timeslots in minutes [10, 20, 30, 40, 50], and loss probability (loss probability 0.8 means 80% data are missing) [0.5, 0.6, 0.7, 0.8, 0.9] are studied, where the underlined values are default values. We have used different learning rates for different datasets (i.e., 0.0001 for the light dataset and 0.000001 for the temperature and humidity dataset) since the same learning rate for all the datasets may lead to poor performance in some cases. The value of μ_U, μ_V, and μ_t is the mean of the observed data, and the variances σ_U^2, σ_V^2, and σ_t^2 are set to the variance of the observed data. Furthermore, the proposed algorithm is compared with two other approaches (i.e., K-nearest neighbor-spatio-temporal (KNN-ST) [9] and a probabilistic matrix factorization (PMF) [10]) using a performance matrix RMSE, and each simulation result is obtained averaging over three runs. Note that, for fairness, in PMF, we have normalized the data similar to the MDI-PTF.

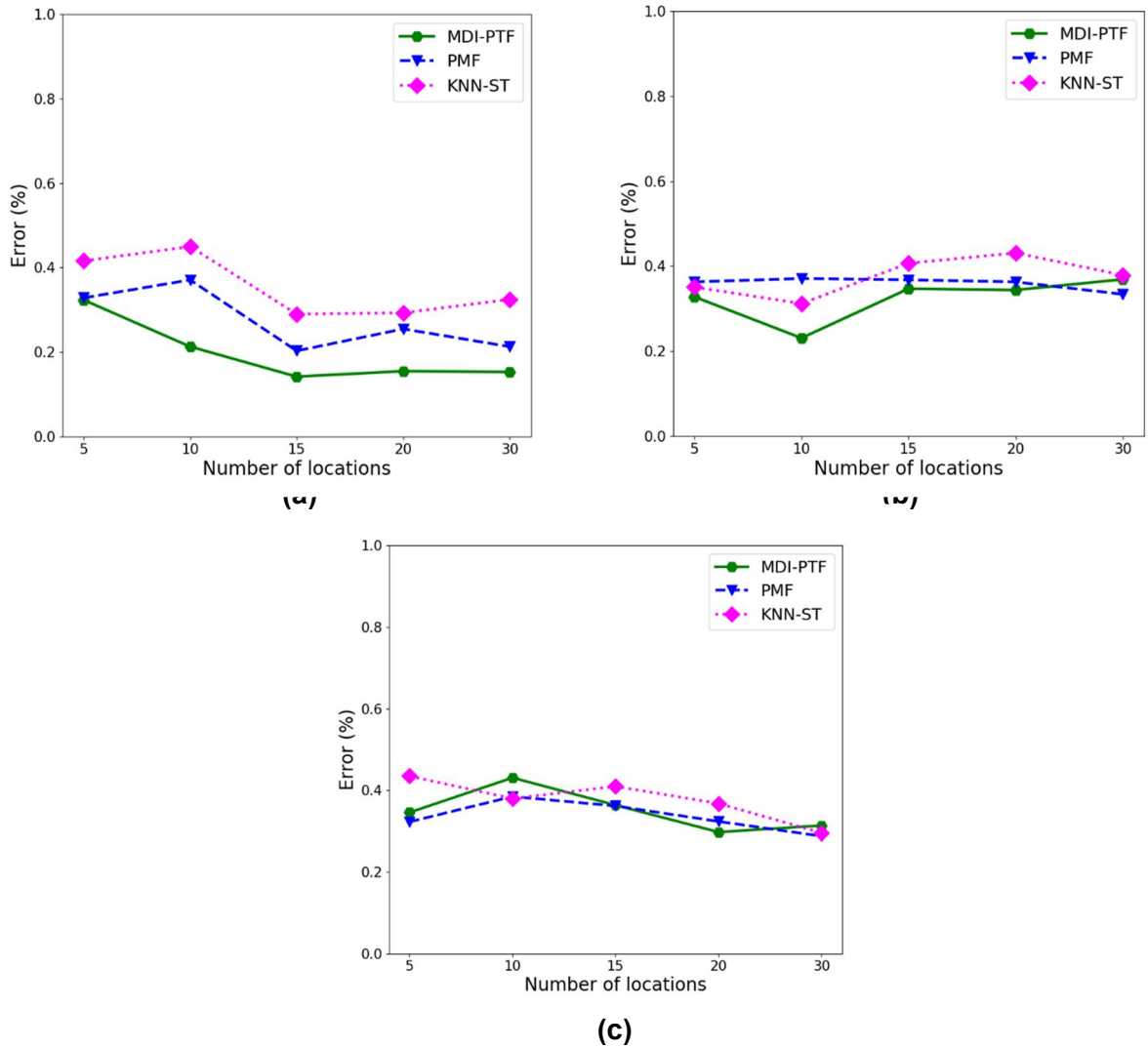

Figure 2. Effect of the different number of locations in (a) light, (b) temperature, and (c) humidity dataset

Fig. 2 depicts the performance of the three algorithms (MDI-PTF, PMF, and KNN-ST) when the number of locations changes using three datasets. Fig. 2 (a) shows the result obtained using the light dataset, where data mostly changes depending on the location rather than time (*e.g.,* mean and standard deviation of data of all locations at a specific time are 58.81 and 43.08, whereas at a specific location, mean and standard deviation are 44.39 and 3.81). Thus, MDI-PTF shows better performance than other approaches as it uses two latent feature matrices to capture the spatial changes. Furthermore, it can be seen that the error ratio between actual and predicted values decreases as the number of locations increases because the distribution of data becomes less dense. In addition, there is more sample for training the gradient descent in MDI-PTF and PMF. In KNN-ST, a higher number of locations increases the probability of selecting neighbors relatively closer to the missing data points; hence, the decreasing trend.

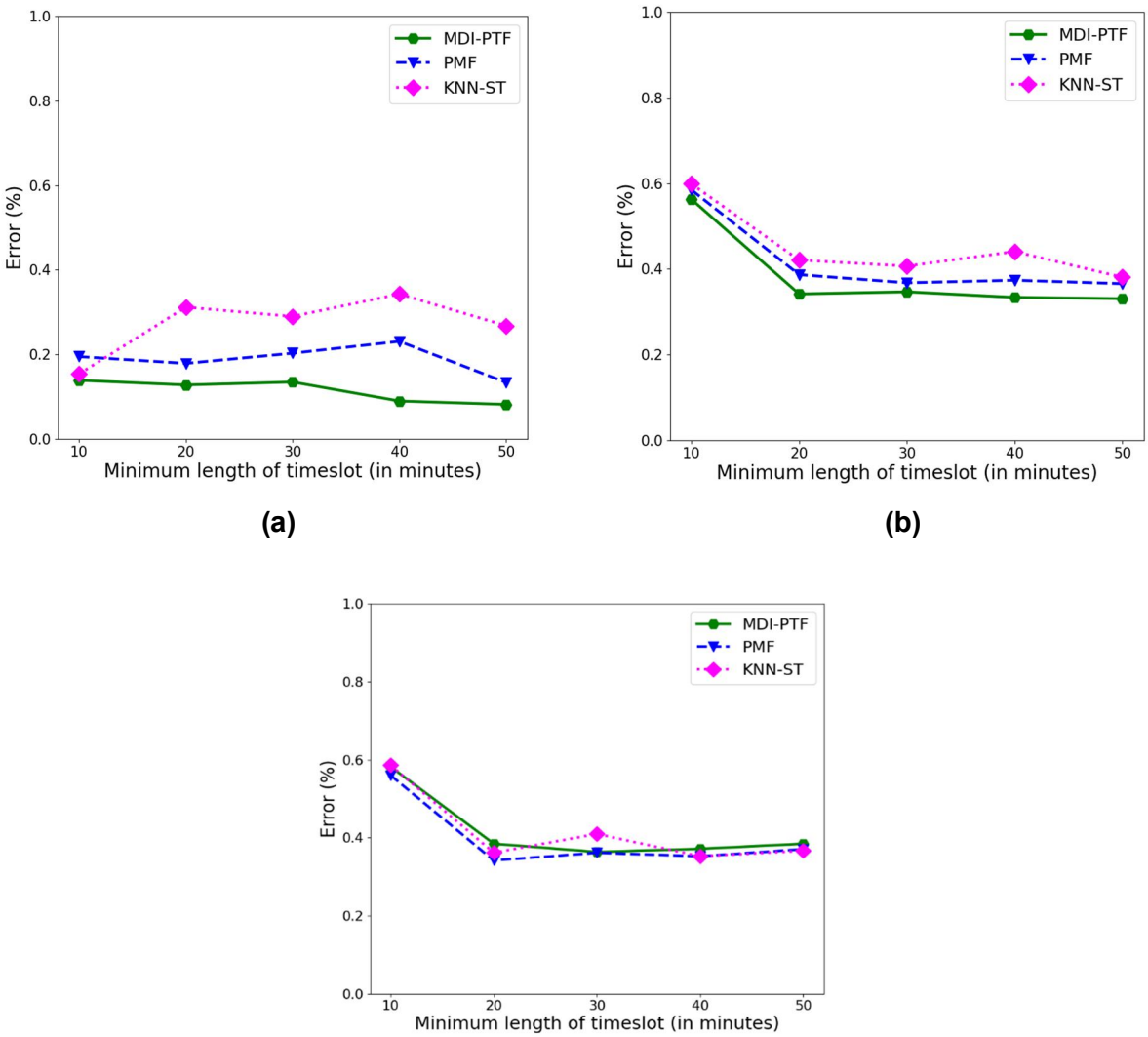

Figure 3. Effect of the various minimum length of timeslots in (a) light, (b) temperature, and (c) humidity dataset

In Fig. 2(b) and 2 (c), the temperature and humidity dataset, respectively, are used to show the performance of the three methods, where MDI-PTF shows average performance, i.e., sometimes outperformed by other approaches (PMF and KNN-ST). This is because data are in these datasets are very compactly distributed both spatially and temporally (for example, at a given location, temperature data has a mean of 18.20 and standard deviation of 0.50, whereas humidity data has 39.07 and 0.13). After normalization, the sensing data become closer than before and smaller, making it hard for gradient descent to approximate the value close to the original, i.e., the algorithm diverges or falls into local optima. The data distribution becomes more compact in these two datasets when the number of locations increases resulting in the haphazard plot (*e.g.,* Fig. 2(b)).

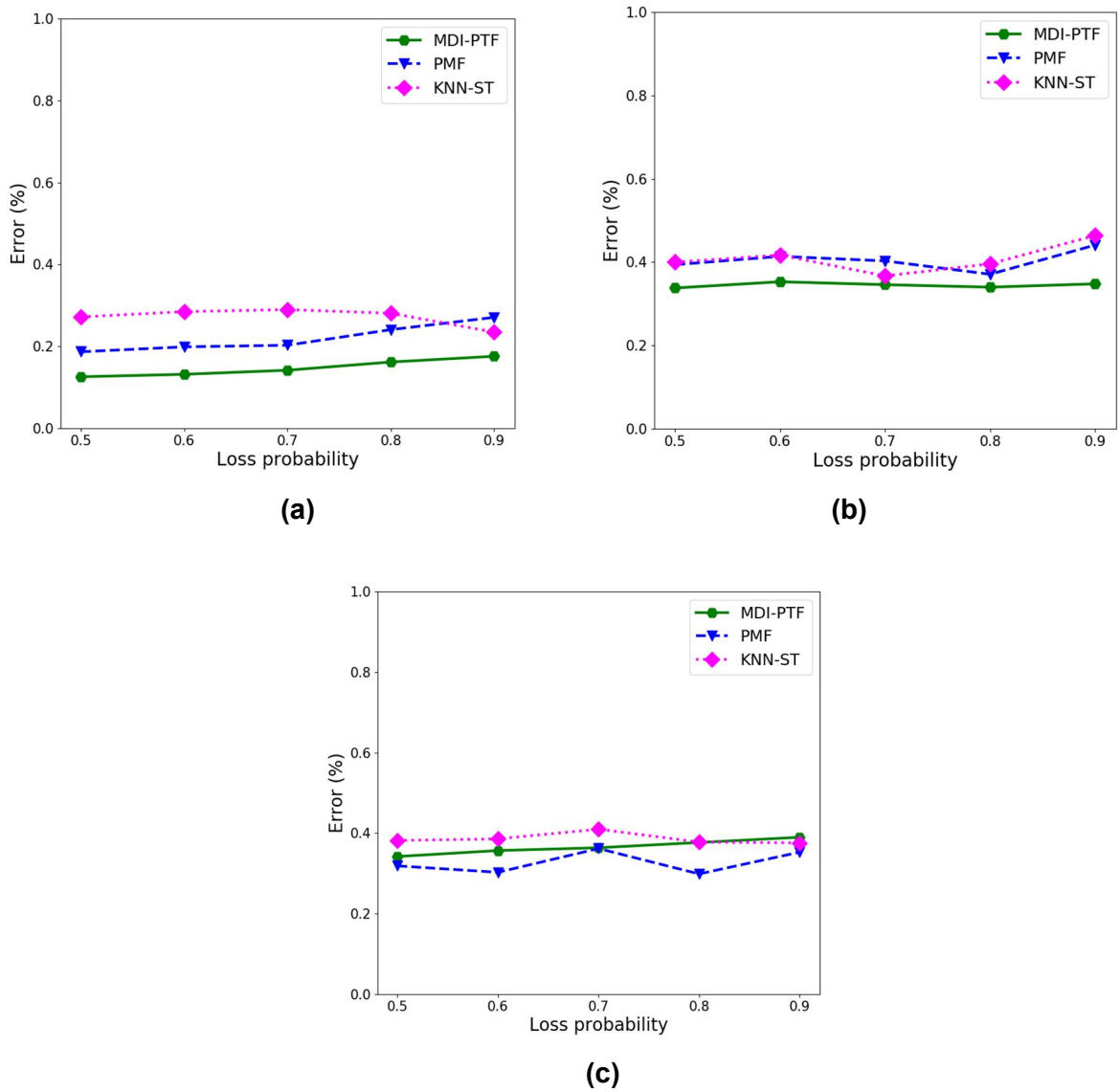

Figure 4. Effect of the different loss probabilities in (a) light, (b) temperature, and (c) humidity dataset

In Fig. 3, the results obtained using the various minimum length of timeslots and dataset are shown. It can be seen from the figure that when the minimum length of timeslots increases, all three methods maintain a

downward trend because data become relatively less compact as the duration between two consecutive data collection times increases. Furthermore, KNN-ST shows relatively better performance when data are more compact (*e.g.,* Fig. 3(a), when minimum timeslot length is 10 minutes) as it estimates the missing value based on the spatial-temporal distance.

Fig. 4 presents the effect of loss probability using different datasets. In all three figures, i.e., Fig. 4(a), 4 (b), and 4(c), when the loss probability increases, the error ratio between ground truth and inferred data increases in the case of all three approaches because a higher number of missing values affect the interpolation as there are less observed data training the gradient descent in MDI-PTF and PMF and for learning the value from neighbors in KNN-ST. However, sometimes KNN-ST obtains a lower error rate even though the loss probability is high, possibly because the spatial-temporal correlation between nearest neighbors and missing points is comparatively higher than previous scenarios as data are considered to be missing randomly.

5. Conclusion

In this paper, we have focused on the missing data inference problem in MCS, where sensing tasks or data collection may not be completed or completed in time because of users, connection, or mechanical errors. Therefore, to infer the missing data, a tensor factorization (TF) technique based on the probabilistic approach is proposed, namely MDI-PTF. MDI-PTF exploits the feature scaling technique along with the probabilistic model of TF to improve the execution time and the quality of data recovery. Evaluation under different datasets using various parameters has shown that MDI-PTF performs better when the data distribution is more scattered, i.e., less compact. Thus, the performance of the proposed algorithm needs further investigation, and we will focus on this issue in our future work.

Acknowledgement

This research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science and ICT (2019R1F1A1058147).

References

[1] B. Guo *et al*., ‘‘Mobile crowd sensing and computing: The review of an emerging human-powered sensing paradigm,’’ *ACM Computing Surveys*, Vol. 48, No. 1, pp. 1–31, Aug. 2015.
DOI: https://doi.org/10.1145/2794400

[2] R. Ganti, F. Ye, and H. Lei, ‘‘Mobile crowdsensing: Current state and future challenges,’’ *IEEE Communications Magazine*, Vol. 49, No. 11, pp. 32–39, Nov. 2011.
DOI: https://doi.org/10.1109/MCOM.2011.6069707

[3] S. M. Lee, J. U. Kim, and Y. M. Kim, “On the Physical Function Evaluation, Prevention Training, and Cognitive Ability Improvement through the Design of a Healthcare Independence Support System based on Emotional Satisfaction of Senior Users,” *International Journal of Internet, Broadcasting and Communication*, Vol. 13, No. 1, pp. 37–46, Feb. 2021.
DOI: https://doi.org/10.7236/IJIBC.2021.13.1.37

[4] Y. Kim and H. Kim, “Usability Evaluation and Improvements of Mobile Travel Apps,” *International Journal of Internet, Broadcasting and Communication*, Vol. 12, No. 1, pp. 27–36, Feb. 2020.
DOI: https://doi.org/10.7236/IJIBC.2020.12.1.27

[5] M. Song, “A Case Study on Energy focused Smart City, London of the UK: Based on the Framework of ‘Business Model Innovation,” *International journal of advanced smart convergence*, Vol. 9, No. 2, pp. 8–19, Jun. 2020.
DOI: https://doi.org/10.7236/IJASC.2020.9.2.8

[6] S. K. Kim, V. Mariappan, and J. S. Cha, "A Study on Environmental Micro-Dust Level Detection and Remote Monitoring of Outdoor Facilities," *International journal of advanced smart convergence*, Vol. 9, No. 1, pp. 63–69, Mar. 2020. DOI: https://doi.org/10.7236/IJASC.2020.9.1.63

[7] G. Kim, "A Case Study on Smart Concentrations Using ICT Convergence Technology," *International journal of advanced smart convergence*, Vol. 8, No. 1, pp. 159–165, Mar. 2019. DOI: https://doi.org/10.7236/IJASC.2019.8.1.159

[8] N. Marchang and R. Tripathi, "KNN-ST: Exploiting Spatio-Temporal Correlation for Missing Data Inference in Environmental Crowd Sensing", *IEEE Sensors Journal*, Vol. 21, No. 3, pp. 3429-3436, Sept. 2020. DOI: https://doi.org/10.1109/JSEN.2020.3024976

[9] L. Kong *et al*., "Data loss and reconstruction in wireless sensor networks*," IEEE Transaction of Parallel and Distribution Systems*, Vol. 25, No. 11, pp. 2818–2828, 2014. DOI: https://doi.org/10.1109/TPDS.2013.269.

[10] L. Wang *et al*., "CCS-TA: Quality-guaranteed online task allocation in compressive crowdsensing," *in Proc. ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pp. 683–694, 2015. DOI: https://doi.org/10.1109/JSEN.2020.3024976

[11] R. Salakhutdinov and A. Mnih, "Probabilistic Matrix Factorization" *in Proc. 20th International Conference on Neural Information Processing Systems*, pp. 1257–1264, 2007.

[12] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems" *Computer*, Vol. 42, NO. 8, pp. 30–37, Aug. 2009. DOI: https://doi.org/10.1109/MC.2009.263

[13] H. Morise, S. Oyama, and M. Kurihara, "Collaborative filtering and rating aggregation based on multicriteria rating", *in Proc. 2017 IEEE International Conference on Big Data*, pp.4335-4340, Dec. 2017. DOI: https://doi.org/10.1109/bigdata.2017.8258477

[14] L. Xiong *et al.*, "Temporal Collaborative Filtering with Bayesian Probabilistic Tensor Factorization", *in Proc. 2010 SIAM International Conference on Data Mining,* pp. 211–222, Dec. 2010. DOI: https://doi.org/10.1137/1.9781611972801.19

[15] F. L. Hitchcock, "The Expression of a Tensor or a Polyadic as a Sum of Products", *Journal of Mathematics and Physics*, Vol. 6, No. 1, pp. 164–189, Apr. 1927.

[16] F. Yang *et al.*, "LFTF: A Framework for Efficient Tensor Analytics at Scale", *in Proc. VLDB Endowment*, Vol. 10, No. 7, pp. 745–756, Mar. 2017. DOI: https://doi.org/10.14778/3067421.3067424

[17] Intel Berkeley Research Lab Data. http://db.csail.mit.edu/labdata/labdata.html