Rethinking the Effect of Sparse Data Completion on Sparse Mobile Crowdsensing Tasks

Yuanbo Xu, *Member, IEEE*, Jiawei Liu, En Wang[†], Bo Yang[†], *Member, IEEE*, Dongming Luan, Yongjian Yang, and Jing Deng, *Fellow, IEEE*

***Abstract*—Mobile crowdsensing (MCS) is a powerful technique that enables a variety of urban tasks, including temperature monitoring, location-based services, and urban path recommendations. However, these tasks often face the challenge of sparse and incomplete sensing data, undermining their effectiveness and reliability. Sparse data completion (SDC) methods have been developed to infer missing or unobserved data by leveraging spatio-temporal correlations to tackle this issue. This forms the core concept of the sparse mobile crowdsensing problem (SMCS), which aims to improve the performance of downstream tasks through inferred data. Despite the potential benefits, most existing SMCS methods fail to consider the trade-off between the cost of SDC and the benefits for downstream tasks. These methods often treat SDC and downstream tasks as independent modules, resulting in suboptimal outcomes. In this paper, we investigate the impact of SDC on the SMCS paradigm, both qualitatively and quantitatively. We establish the upper bound of performance achievable when applying SDC in SMCS under different levels of sensing data sparsity. Based on these studies and findings, we propose a practical and flexible framework called SDC-EVA, Sensing Data Completion EVAluation framework. This framework allows for applying different SDC methods in SMCS, considering factors such as computing complexity, storage space, and associated costs. Our proposed framework allows researchers to assess the necessity and feasibility of integrating SDC into SMCS systems before designing and deploying them in real-world scenarios. This assessment can be tailored to specific data sparsity and contextual information. To validate the effectiveness of our proposed evaluation framework, we conduct experiments in various real-world scenarios involving different combinations of SDC and downstream tasks. The results demonstrate the superiority of our framework in improving the performance of SMCS. By presenting these findings, we aim to contribute to developing SMCS techniques and provide valuable insights for researchers and practitioners.**

***Index Terms*—Spatio-temporal analysis; Data completion; Sparse Crowdsensing**

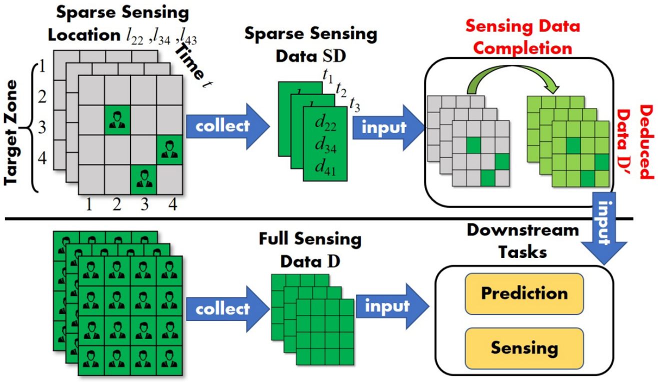

Fig. 1. An example to illustrate the challenges in SMCS. 1) the efficiency trade-off between the SDC computing cost and the full sensor deployment cost, and 2) the data quality of deduced data *D*' and the full sensing data *D* for supporting downstream tasks.

I. Introduction

Mobile crowdsensing (MCS) has emerged as a promising data collection paradigm in urban computing, thanks to the widespread adoption of mobile devices equipped with various sensors like accelerometers, gyroscopes, GPS, cameras, and microphones [1], [2]. MCS leverages collaboration among users to share local knowledge, environmental context, or traffic conditions, enabling large-scale, complex global tasks using mobile devices for sensing, collection, and computation. The key factors that make MCS unique are the presence of many sensing participants and the low deployment costs [3]. These factors have made MCS a subject of exploration in academia and industry. Compared to traditional wireless sensor networks (WSNs) that rely on specific network architectures and designed protocols, MCS has advantages in addressing complex time- and space-aware downstream tasks [4]. For example, MCS enables temperature monitoring [5], location-based services [6], and urban recommendation tasks [4] with its crowd flexibility and dynamic self-organization [3]. The ability to tap into the collective power of the crowd allows MCS to overcome the limitations of traditional WSNs and provide efficient solutions for a wide range of urban computing challenges.

MCS tasks can be broadly categorized into two main types: Prediction and Sensing [7]. The Prediction task focuses on predicting the future actions of a target based on the sensed data. In MCS, for instance, POI recommendation can be considered as a Prediction task, where the destination location of a mobile user can be deduced from their historical trajectory and the trajectories of their neighbors collected by mobile devices. On the other hand, the Sensing task aims to obtain a global understanding of a specific area or zone. Examples of sensing tasks in MCS include temperature monitoring and monitoring traffic conditions. However, we notice that MCS scenarios

Yuanbo Xu, Jiawei Liu, En Wang, Dongming Luan, Yongjian Yang, are with Mobile Intelligent Computing Lab (MIC Lab), the Department of Computer Science and Technology, Jilin University, Changchun, Jilin 130012, China. (E-mail: yuanbox@jlu.edu.cn; jiawei23@jlu.edu.cn; wangen@jlu.edu.cn; luandm20@mails.jlu.edu.cn; yyj@jlu.edu.cn.)

Bo Yang is with the Department of Computer Science and Technology, Jilin University, Changchun, Jilin 130012, China. (E-mail: ybo@jlu.edu.cn)

Jing Deng is with the Department of Computer Science, University of North Carolina at Greensboro, NC 27412, USA. (E-mail:jing.deng@uncg.edu)

[†]The corresponding authors are En Wang and Bo Yang.

usually do not consider the trade-off between the cost of SDC and the benefits for downstream tasks. Introducing SDC may result in more energy costs for specific tasks without ensuring promising performance. To the best of our knowledge, there is no prior work in MCS to formulate or explore this trade-off.

In practice, both prediction and sensing tasks face challenges related to **Sensing Data Quality Issue** and **Energy Cost Issue** associated with sensing [8]–[10]. These challenges have been identified as the bottlenecks of traditional MCS. To mitigate the energy costs of sensing, a variant of MCS called Sparse MCS (SMCS) has been proposed when facing certain constraints. In SMCS, mobile devices collect data only when they are in specific locations or when certain conditions are met, reducing the amount of data that needs to be transmitted and processed, and the energy consumption of mobile devices. Existing research on SMCS has explored various aspects, such as energy saving, data completion, and privacy concerns. Researchers have noted that SMCS has the potential to overcome the energy cost issue in traditional MCS, and in certain tasks, the SMCS framework has demonstrated superior performance.

In SMCS, the classic framework is two-stage: sparse data completion and downstream task (as indicated in Figure 1). Specifically, sparse data completion is the core module for SMCS deployment in a real-world scenario, which utilizes the sensed/observed data to infer the unsensed/unobserved data with the help of spatio-temporal correlations to enhance the data quality. Because SMCS employs limited participants or senses limited sub-zones of the global zone, it naturally solves the energy cost issue. Thus, most existing SMCS methods claim that using limited sensed/observed data to deduce unsensed/unobserved data can address data quality and energy cost issues. However, it notes that SDC consumes computing energy in SMCS, especially when large models such as generative and deep models are employed in the SDC module. These models may require more computing resources to infer data than directly sensing them. Therefore, the energy cost issue cannot be directly tackled by employing SDC, and enhancing prediction accuracy may require the incorporation of supplementary prediction modules, thereby escalating costs, and leading to two important challenges in the SMCS framework:

- How to define the qualitative and quantitative measures of sensed data and inferred data to support specific downstream tasks? (Sensing Data Quality Issue)
- How to strike a balance between the energy-saving benefits of utilizing the SDC module and the option of deploying more sensors directly? (Energy Cost Issue)

To the best of our knowledge, our work is the first to explore both challenges in mobile crowdsensing tasks, particularly from a data quality perspective. In this study, we investigate the various conditions that affect the performance of crowdsensing tasks, including data sparsity and the accuracy of inferred data. Additionally, we consider the computing costs associated with existing SDC methods and establish a performance upper bound for optimal performance when utilizing these methods in SMCS. To address these challenges, we propose a practical and flexible framework, SDC-EVA, for applying different SDC methods in SMCS. This framework considers the computing complexity, storage space, and structure costs, allowing researchers to assess the necessity and feasibility of introducing SDC into their SMCS systems based on their specific data sparsity and context information. Our framework serves as a heuristic deployment guide for SDC in SMCS. We validate the effectiveness of our proposed framework through experiments conducted on real-world scenarios involving various SDC and downstream tasks, such as global monitoring and POI recommendation. The results demonstrate the superiority of our evaluation framework in improving SMCS.

In summary, the contributions of this paper are as follows:

- We highlight the importance of **S**parse **D**ata **C**ompletion in sparse mobile crowdsensing scenarios, largely overlooked by existing research. To validate the SDC module, we establish the performance upper bound for different SMCS situations and downstream tasks.
- We introduce a practical and flexible framework that serves as a novel SMCS paradigm, considering the trade-off between complexity, storage space, and structure cost. This framework can serve as a guide for deploying SDC in various SMCS applications.
- We validate the effectiveness of our proposed framework through experiments conducted on real-world scenarios involving various SDC and downstream tasks, including global monitoring and POI recommendation. The results demonstrate the effectiveness of our evaluation framework for applying SDC in SMCS.

II. Preliminaries

We introduce the basic definitions of SMCS scenarios and the problem definitions of SMCS issues. We conducted the pilot validation to support our motivation and solution.

A. Basic Definitions

In SMCS scenarios, we extract the collected data in a tensor style, where we treat each geo-map as a target zone matrix $\mathbf{M}\in \mathbb{R}^{m\times n}$, m, n are the scales of the indexes of longitude and latitude, respectively. MCS tasks, like global monitoring and POI recommendation, require the MCS framework to collect data in a continuous time period t $= \{t_1, t_2, ...t_z\}$, in which z is the scale of the time dimension. Thus, the whole sensing space can be represented as a tensor $\mathbf{\Gamma}\in \mathbb{R}^{m\times n\times z}$. For each entity d_{ijk} in $\mathbf{\Gamma}$, there is a mask $c_{ijk} \in \{0, 1\}$ to represent whether the specific location in longitude i, latitude j at time k is collected $\{c_{ijk}$=1$\}$ or not $\{c_{ijk}$=0$\}$, where $i \in \{1, 2, ..., m\}$, $j \in \{1, 2, ..., n\}$, $k \in \{t_1, t_2, ...t_z\}$. Specifically, we utilize the following formulation to evaluate the collected data quality Q:

$$\mathrm{Q}_k = \frac{\sum\limits_{i,j} c_{ijk}}{|\mathbf{M}|} = \frac{\sum\limits_{i,j} c_{ijk}}{m \cdot n}. \tag{1}$$

$$\mathrm{Q} = \frac{\sum\limits_{i,j,k} c_{ijk}}{|\mathbf{\Gamma}|} = \frac{\sum\limits_{i,j,k} c_{ijk}}{m \cdot n \cdot z}. \tag{2}$$

Note that $|\mathbf{\Gamma}|$ represents the scale of $\mathbf{\Gamma}$. Q_k represents the collected data quality in time k. And Q is the global collected data quality.

For the SDC module, the input of SDC could be the tensor $\mathbf{\Gamma}$, or several specific subsets of it. The purpose of the SDC model is to utilize the limited collected data to deduce the unobserved ones, which could be formulated as follows:

$$\left\{\widehat{d}_{ijk}|c_{ijk}=0\right\} = \mathrm{SDC}(\mathrm{INITIAL}(\mathbf{\Gamma}, \mathrm{Q}), \alpha_{\mathrm{SDC}}), \tag{3}$$

where the $\left\{\widehat{d}_{ijk}|c_{ijk}=0\right\}$ are the predicted values for unobserved locations, and INITIAL() is the proper function to tackle tensor $\mathbf{\Gamma}$ for input. α_{SDC} is the hyper-parameters of SDC. In this work, we focus on the efficiency of the SDC module. Thus, we need to calculate SDC's storage cost SSTC, computing complexity SCC, and the correlations between them and energy E. To make the trade-off more reasonable, we also need to link these metrics to the performance of downstream tasks P_{task}, as follows:

$$\mathrm{EVA}_{\mathrm{SDC}} \sim \{\mathrm{P}_{\text{task}}, \mathrm{E}_{\mathrm{SSTC}}, \mathrm{E}_{\mathrm{SCC}}\}. \tag{4}$$

where $\mathrm{E}_{\mathbf{SSTC}}$ is the storage energy cost, $\mathrm{E}_{\mathrm{SCC}}$ is the computing complexity energy cost, and $\mathrm{EVA}_{\mathrm{SDC}}$ is the overall evaluation for a SDC method. To apply SDC in SMCS scenarios, we should understand the most important problems: 1) whether the energy saving from collecting limited data can cover the energy cost of SDC. And 2) whether the collected data (sensed by sensors) or predicted data (calculated by SDC) can support the downstream tasks.

B. Problem Definitions

Problem Definition 1: Global sensing *vs* Sparse Data Completion (Trade-Off problem): Given downstream tasks TASK, we can either utilize global sensing to obtain the whole data sensor $\mathbf{\Gamma}$, which may consume sensing energy $\mathrm{E}_{\mathrm{GSSC}}$ and storage cost $\mathrm{E}_{\mathrm{GSTC}}$, or utilize SDC to predict the whole data with the limited collected data (with different Q). Note that for global sensing, Q_*=1 and Q=1, which means that at each time t_*, all the data are collected by sensors. Moreover, we divide the trade-off into two aspects: energy and performance. From an energy aspect, we should consider $\mathrm{E}_{\mathrm{GSSC}}$ and $\mathrm{E}_{\mathrm{GSTC}}$ of global sensing, and the $\mathrm{E}_{\mathrm{SSSC}}$, $\mathrm{E}_{\mathrm{SSTC}}$ and $\mathrm{E}_{\mathrm{SCC}}$ of SDC:

$$\mathrm{E}_{\mathrm{GSSC}}, \mathrm{E}_{\mathrm{GSTC}} \sim \mathrm{E}_{\mathrm{SCC}}, \mathrm{E}_{\mathrm{SSSC}}, \mathrm{E}_{\mathrm{SSTC}}. \tag{5}$$

Note that the SMCS framework utilizing SDC needs to be able to sense and store limited data. Thus, the cost comparison can be treated as the linear formulation with Q as the weight:

$$\begin{aligned} \mathrm{E}_{\mathrm{SSSC}} &\sim \mathrm{E}_{\mathrm{GSSC}} \cdot \mathrm{Q}; \\ \mathrm{E}_{\mathrm{SSTC}} &\sim \mathrm{E}_{\mathrm{GSTC}} \cdot \mathrm{Q}. \end{aligned} \tag{6}$$

Considering real-world situations, including time and space, more details about the Trade-Off problem are given in the following sections.

TABLE I
NOTATION LIST.

Notation	Description
SDC	Sparse data completion
SMCS	Sparse mobile crowdsensing
$\mathbf{M}$	collected data in the target zone (matrix)
$\mathbf{\Gamma}$	collected data in the target zone (tensor)
m, n	geo-scales of the target zone
z	time-scale of the collected data
d_{ijk}	collected data at location (i, j) at time k
c_{ijk}	data indicator at location (i, j) at time k
Q_k	collected data quality at time k, calculated by Eq. (1)
Q	collected data quality (sparsity), calculated by Eq. (2)
Q_0, Q_1	collected data quality thresholds
$\widehat{d}_{ijk}$	predicted data by SDC
$\mathrm{E}_{\mathrm{GSSC}}$,$\mathrm{E}_{\mathrm{GSTC}}$	global sensing cost and storage cost
$\mathrm{E}_{\mathrm{SSSC}}$,$\mathrm{E}_{\mathrm{SSTC}}$	SDC sensing cost and storage cost
$\mathrm{E}_{\mathrm{SCC}}$	SDC computing cost
θ_{SDC}	accuracy of SDC for predicting data, calculated by Eq. (7)
P_{task}	performance of downstream task task
α, τ, β	hyper-parameters of SDC

Problem Definition 2: Effect of SDC module on downstream tasks (Effectiveness problem): Given the SDC module, the predicted unobserved data $\widehat{d}_{ijk}|c_{ijk}=0$ has the bias from the real collected data $d_{ijk}|c_{ijk}=0$, which may affect the performance of downstream tasks. Specifically, the predicted accuracy of SDC modules can measure this bias:

$$\theta_{\mathrm{SDC}} = 1 - \sum_{i,j,k} \frac{\left|\{\hat{d}_{ijk}|c_{ijk}=0\} - \{d_{ijk}|c_{ijk}=0\}\right|}{\left|\{\hat{d}_{ijk}|c_{ijk}=0\}\right|(i \cdot j \cdot k)}. \tag{7}$$

From a performance aspect, the bias may affect the downstream tasks. In this paper, we aim to find the correlations between the SDC performance and the downstream tasks, formulated as follows:

$$\mathrm{P}_{\text{task}} \sim \mathrm{Q}, \theta_{\mathrm{SDC}}, \tag{8}$$

which means that the P_{task} may be affected by the data quality and the predicted accuracy of SDC, respectively. Note that there are different downstream tasks in SMCS scenarios. We aim to propose a general analysis method to give practical and theoretical insurance for applying SDC.

With the above definitions, we summarize our key problem with considerations about the Trade-off problem and Effectiveness problem: we aim to design an analysis methodology to give a practical framework for evaluating the necessity of applying SDC in SMCS scenarios and assessing SDC ($\mathrm{EVA}_{\mathrm{SDC}}$). Important notations are listed in Table I.

C. Pilot Validation

We conduct two pilot validations to validate our proposed method: First, we select different data sparsity as the input to the downstream tasks/SDC methods. This validation aims to indicate the effect of limited data on the performance of downstream tasks/SDC methods. Without loss of generality, we utilize POI recommendation [11] and Global monitoring [12] as the downstream tasks, DMF [13] and basic MF [14] as the SDC methods. Then we randomly select sensed data (`Changchun City Traffic Data`) as the downstream tasks' input. Note that it means that the Q is the indicator of

TABLE II
DESCRIPTION OF SPARSITY OF CHANGCHUN CITY TRAFFIC DATA

Data	sensor data 1	sensor data 2	GPS points
Data Quality Q	0.64	0.57	0.87

data sparsity. We select different evaluation metrics, MAPE for the global monitoring, HR@10 for the POI recommendation, and θ_{SDC} for SDC methods, respectively.

First of all, without loss of generality, we list the statistics of the data quality Q of (`Changchun City Traffic Data`), including sensed GPS points and sensor data 1 and 2 (Table II). Note that the data quality Q is not similar to data sparsity, which is calculated by Formulation 2. From the statistics, we notice that in SMCS scenarios, low data quality is a common situation. The pilot validation 1 is reported in Figure 2. We notice that when the data is relatively sparse (Q< 0.4), the performance of both downstream tasks is unacceptable. When the Q rises to 0.7, the POI recommendation and global monitoring are close to the state-of-the-art performance. Besides, the performance of the SDC methods is also affected by the different Qs. Thus, we validate the first intuition:

Proposition 1: the data sparsity affects both downstream tasks and SDC methods' performance.

For the second validation, we aim to explore the effect of the SDC's prediction accuracy (data inference quality) on the downstream task's performance. However, because of the accuracy limitations of the SDC methods, we cannot cover all the situations with some specific SDC methods. Without loss of generality, we add random bias N(μ, σ) to the ground truth collected global data to simulate the inaccurate prediction of SDC, where μ is the average value of data and σ is the controllable parameter. We utilize θ_{bias} (calculated as the Eq. (7)) as the metric to evaluate the data accuracy.

The pilot validation 2 is reported in Figure 3. We notice that when the data inference quality is low, especially below 0.5 of θ_{bias}, the performance of downstream tasks is severely damaged. In the extreme situation, when the θ_{bias}=0.1, all the downstream tasks are down. Considering that the traditional SMCS framework utilizes the SDC's results as the input of downstream tasks as an end-to-end solution and validates the whole framework with the accuracy of the downstream tasks. They omit analyzing the effect of SDC methods' performance on the downstream tasks. From Figure 3, we conclude that if the SDC's accuracy is low, the performance of the whole SMCS framework cannot be ensured.

Based on the above two pilot validations, to achieve an explainable and efficient SMCS framework, we should consider both SDC's accuracy and the data quality. Hence, we have the second proposition:

Proposition 2: SDC's performance accuracy should be considered for ensuring SMCS's performance.

III. METHODOLOGY FOR APPLYING SDC IN SMCS

A. Qualitative and Quantitative Analysis

We should pay attention to the fact that the performance of the SMCS framework is related to 1) the data quality and 2) if utilizing SDC, SDC's output accuracy, as we indicate in Eq. (8) qualitatively. Thus, we analyze it quantitatively from both performance and energy perspectives. Without loss of generality, for applying SDC, we should consider the following two extreme situations:

- the collected data is too sparse that even if we utilize the SDC methods, the performance of SDC for supporting the downstream tasks cannot be guaranteed.
- the collected data is so dense that even if we utilize the SDC methods, the contribution of SDC for enhancing SMCS performance and the energy cost that SDC introduces are not quantitatively calculated.

To derive the upper performance bound of utilizing SDC, we define the optimal performance of SMCS, which utilizes the global sensed data (Q=1), is P_{opt}. So, the SDC's upper performance bound can be formulated as follows:

$$\text{P} = \lambda_{\text{P}}\text{P}_{\text{opt}} = \theta_{\text{SDC}} \cdot \frac{\beta}{1+e^{-\alpha_{\text{Q}}\text{Q}}} \cdot \text{P}_{\text{opt}}, \tag{9}$$

where β is the model-specific parameter of SDC, which can be calculated by AIC (Akaike Information Criterion) and BIC (Bayesian Information Criterion) [15]:

$$\begin{aligned} \text{AIC} &= 2k - 2ln(\text{LL}_{\text{SDC}}); \\ \text{BIC} &= ln(\text{Q} \cdot |\mathbf{\Gamma}|) - 2ln(\text{LL}_{\text{SDC}}), \end{aligned} \tag{10}$$

where k is the parameter number of SDC, which is related to E_{SCC}; LL_{SDC} is the likelihood function. Note that when Q=1, SDC is unnecessary for SMCS framework, so we set β=1+$e^{-\alpha_{\text{Q}}}$ and θ_{SDC}=1, to ensure P=P_{OPT}. Though the performance upper bound is formulated explicitly, it cannot be easily achieved according to the other factors, including structural risk and empirical risk, which is inevitable. Thus, we introduce α_{Q}=2 to consider both risks' effects on performance. We follow the classic proof of [15], [16], and we conclude that if we apply the SDC model, the ideal upper bound of SMCS performance is $\text{P} = (1/1+\text{e}^{-3})\text{P}_{\text{opt}} \approx 0.9527\text{P}_{\text{opt}}$. However, the ideal situation cannot even be approximated in real-world scenarios. Hence, instead of utilizing the upper bound of utilizing SDC, we need to use λ_{P} in Eq. (9) for choosing the proper SDC model for the specific SMCS scenarios without considering the energy costs.

As we indicate in Eq. (6), Q is the cooperative factor for E_{GSSC} and E_{SSSC}, E_{GSTC} and E_{SSTC}. We could formulate the energy cost as follows:

$$\begin{aligned} \text{E}_{\text{GSSC}} &= e_{\text{unit}} \cdot m \cdot n \cdot z; \\ \text{E}_{\text{GSTC}} &= e_{\text{store}} \cdot m \cdot n \cdot z. \end{aligned} \tag{11}$$

where $m \cdot n \cdot z$ is the scale of tensor $|\mathbf{\Gamma}|$, and e_{unit} and e_{store} are the energy cost for collecting and storing one data value d_{ijk}, respectively. Thus the global sensing (Q=1) energy cost can be formulated as:

$$\text{E}_{\text{G}} = (e_{\text{unit}} + e_{\text{store}}) \cdot |\mathbf{\Gamma}|; \tag{12}$$

Considering the situation that SDC models only require limited collected data, we could formulate SDC's energy cost as follows:

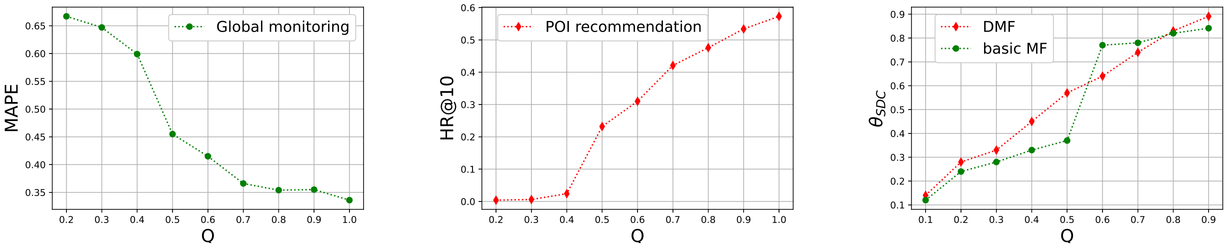

Fig. 2. Pilot validations with different Q for SDC methods and downstream tasks.

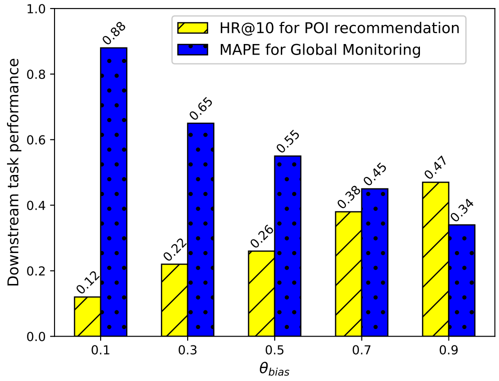

Fig. 3. Pilot validation for data inference quality. We add random noise $\mathrm{N}(\mu, \sigma)$ to simulate the data inference bias. Thus we use θ_{bias} as the metric to evaluate the effect of added noise.

$$\begin{aligned} \mathrm{E}_{\mathrm{SSSC}} &= \mathrm{E}_{\mathrm{GSSC}} \cdot \mathrm{Q}; \\ \mathrm{E}_{\mathrm{SSTC}} &= \mathrm{E}_{\mathrm{GSTC}} \cdot \mathrm{Q}. \end{aligned} \tag{13}$$

Utilizing the SDC module introduces the additional cost, $\mathrm{E}_{\mathrm{SCC}}$, related to the computing complexity. Specifically, we can treat $\mathrm{E}_{\mathrm{SCC}}$ as a stable weight, which is also related to the input data's sparsity Q, which is formulated as:

$$\mathrm{E}_{\mathrm{INTRO}} = \mathrm{E}_{\mathrm{SCC}} = (e_{\text{compute}}) \cdot |\mathbf{\Gamma}| \cdot \mathrm{Q}, \tag{14}$$

where e_{compute} is the energy cost of each computing. Hence, by utilizing SDC, we could reduce the collecting and storing energy costs:

$$\mathrm{E}_{\mathrm{SAVE}} = (e_{\text{unit}} + e_{\text{store}}) \cdot |\mathbf{\Gamma}| \cdot (1 - \mathrm{Q}). \tag{15}$$

We could evaluate the energy cost by utilizing the following metric λ_{E}:

$$\begin{aligned} \lambda_{\mathrm{E}} &= \mathrm{E}_{\mathrm{SAVE}} / \mathrm{E}_{\mathrm{INTRO}} \\ &= (e_{\text{unit}} + e_{\text{store}}) \cdot (1 - \mathrm{Q}) / (e_{\text{compute}}) \cdot \mathrm{Q}. \end{aligned} \tag{16}$$

If the $\lambda_{\mathrm{E}} > 1$, SDC benefits the SMCS framework by saving energy costs, which means that $\mathrm{Q} < (e_{\text{unit}} + e_{\text{store}})/(e_{\text{unit}} + e_{\text{store}} + e_{\text{compute}})$, and vice versa.

Cooperating with the pilot validations, especially Proposition 1 and Proposition 2, we notice that if Q is lower than 0.4, the downstream tasks and SDC's performance cannot be guaranteed, not even close to the theoretical optimal λ_{P}. Still, if the Q is higher than $(e_{\text{unit}} + e_{\text{store}})/(e_{\text{unit}} + e_{\text{store}} + e_{\text{compute}})$, the introduced energy cost is higher than the saved energy, which violates the goal of introducing SDC into the SMCS framework. Hence, we give a simple but practical principle for applying SDC to the SMCS framework:

Principle 1: In real-world SMCS scenarios, to enhance the performance of the whole framework, 1) if the collected data quality (Q) is lower than Q_0, we need to deploy more sensors to collect data; 2) if the Q is in the range (Q_0, Q_1), we could introduce SDC models for better performance and lower energy cost; 3) if the Q is higher than Q_1, SDC is not the proper choice for SMCS models, we could utilize more sensors or modify the models of downstream tasks. Q_0 is an empirical parameter which is pre-defined and $\mathrm{Q}_1 = (e_{\text{unit}} + e_{\text{store}})/(e_{\text{unit}} + e_{\text{store}} + e_{\text{compute}})$.

Without loss of generalization, we set $e_{\text{unit}} = e_{\text{store}} = e_{\text{compute}}$ in Eq. (16):

$$\lambda_{\mathrm{E}} = 2 \cdot (1 - \mathrm{Q})/\mathrm{Q}. \tag{17}$$

When $\lambda_{\mathrm{E}} > 1$, which means $\mathrm{Q} < 2/3 \approx 0.667$, SDC benefits the SMCS framework by saving energy cost; when $\lambda_{\mathrm{E}} < 1$, which means $\mathrm{Q} > 2/3 \approx 0.667$, SDC introduces more computing energy cost than the ones saving. To this end, we set Q_0=0.4, Q_1=0.667 as the initial threshold of Q in the experiment.

B. Metrics for selecting SDC models

With the above qualitative and quantitive analysis, we ensure the data requirement conditions ($\mathrm{Q}_0 < \mathrm{Q} < \mathrm{Q}_1$) for utilizing SDC. However, when selecting SDC methods for enhancing SMCS performance, we still need to analyze the correlations between SDC accuracy θ_{SDC} and downstream task performance P_{task}.

Thus, we design a novel metric for applying SDC in the SMCS framework. To cooperate with the SDC accuracy and downstream task performance, we formulate the overall performance for SDC in the SMCS framework:

$$\lambda^{\mathrm{SDC}} = (1 - \tau) \cdot \lambda_{\mathrm{E}}^{\mathrm{SDC}} + \tau \cdot \lambda_{\mathrm{P}}^{\mathrm{SDC}}, \tag{18}$$

where τ is the weight balance of the energy costs and the performance enhancement. Without specific requirements, we set τ=0.5. Note that the SDC selections should take the effects of SDC on downstream tasks. We modify the calculation of λ_{P} by replacing the likelihood:

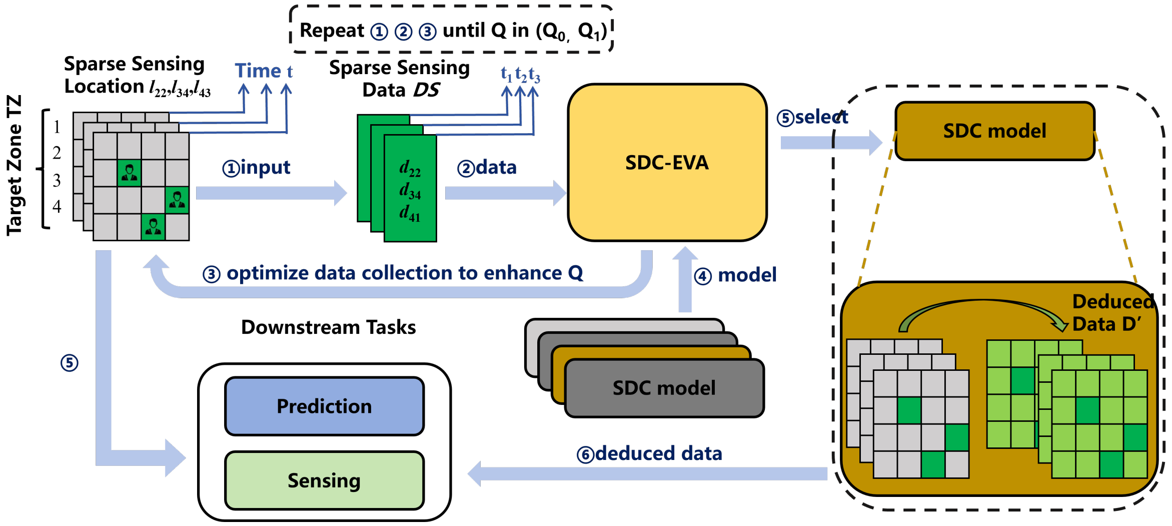

Fig. 4. SDC-EVA framework, a practical, flexible framework for applying different SDC methods in SMCS, considering computing complexity, storage space, and structure cost. The right circle is the processing step order when applying this framework, and the black line separates the whole framework into two parts: with/without SDC.

$$\mathrm{LL}_{\mathrm{SDC}} - \mathrm{LL}_{\mathrm{SDC,\ tasks}}. \tag{19}$$

The calculation of likelihood is not the key question of this work, which can be found in [15]. Thus we introduce the pre-heuristic stage into SMCS: before we design a practical SMCS framework, especially whether to utilize the SDC modules, we should do the following estimation:

- Check the collected data quality Q.
- If the Q is in (Q_0, Q_1), calculate λ^{SDC} for selecting the proper SDC models.

IV. SDC-EVA Framework for Applying SDC in SMCS

With the proposed metric λ^{SDC}, we introduce a module into traditional two-stage SMCS frameworks: the SDC-EVA module, which extends the two-stage, one-way SMCS framework to an explainable, heuristic framework, which is indicated in Figure 4. The SDC-EVA module has the following functions:

- Calculate the collected data quality (Q).
- Decide whether to employ SDC or not (**YES/NO**).
- Calculate the metric λ^{SDC} for candidate SDC methods for specific downstream tasks (if **YES**).
- Redeploy the sensor networks/input the collected data into downstream models (if **NO**).

Note that if Q is lower than Q_0, the SDC-EVA module requires the SMCS framework to redeploy the sensor networks to improve the Q because the poor data quality cannot be remedied by introducing SDC modules. If Q is higher than Q_1, introducing SDC may add more energy cost with limited performance, the SDC-EVA module may refine the data collected procedure to enhance Q or directly input the collected data SD into downstream tasks.

By adding the SDC-EVA module, the proposed SMCS framework could achieve several goals:

- Decide whether to employ SDC or not (**YES/NO**) according to the data conditions.
- Choose the proper SDC methods for specific tasks.
- Adjust the proposed SMCS framework with reasonable explanations.
- Give guidance for data collection strategy.

We give the whole procedure of SDC-EVA in Algorithm 1:

V. Experiment and Validation

This section introduces the experimental settings, including datasets, baselines, and other details. Subsequently, we present extensive experiments to answer the following research questions:

RQ1: How does the hyper-parameter affect the performance of SDC-EVA? Which are the optimal values?

RQ2: What is the effectiveness of SDC-EVA? Can it provide an explainable, practical way to decide which SDC should be employed?

RQ3: What is the effect of each module in SDC-EVA? How does SDC-EVA work in the SMCS framework?

A. Data, task, baselines, and settings

1) Datasets: We self-collect two raw datasets of Changchun City:

(a) *Trajectory dataset*: It contains billions of raw trajectories collected by GPS devices in smartphones from July to December 2017. (b) *POI dataset*: It covers 3,402 POIs with 159 sub-categories of 12 main categories. We delete the POIs with less than 200 check-ins in six months and the trajectory without mobility in 24 hours as data pre-filtering. We present these datasets' visualization in Figure 5.

Without loss of generality, we also employ the benchmark dataset `Gowalla`[1], the details of both datasets are listed in Table III.

[1] https://snap.stanford.edu/data/loc-gowalla.html

Algorithm 1 SDC-EVA framework for SMCS scenarios

```
Require: collected data $\mathbf{\Gamma}$, geo-scales $m, n$, time scale $z$, candidate sparse data completion models $SDCs$, downstream tasks $tasks$, threshold $\mathrm{Q}_0$, $\mathrm{Q}_1$, parameters $\alpha, \tau, \beta$.
Ensure: downstream tasks performance $\mathrm{P}_{\text{task}}$.
Initialization: data initialization function INITIAL() (Eq. (3)). hyperparameter $\alpha_{\text{SDC}}$ (Eq. (3)) $\alpha_{\text{Q}}$ (Eq. (9)), SDC metric weight $\tau$ (Eq. (18))
Step 1
Collect data at Target Zone $\mathbf{M}$;
Preprocess the collected data to form the collected data tensor $\mathbf{\Gamma}$;
Step 2
Input the collected data tensor $\mathbf{\Gamma}$ to SDC-EVA module.
Step 3
Calculate Q with Eq. (2);
if $\mathrm{Q} < \mathrm{Q}_0$ or $\mathrm{Q} > \mathrm{Q}_1$ then
    Jump to Step 4;
end if
for each SDC in candidate SDC set do
    Calculate $\lambda_{\mathrm{E}}, \lambda_{\mathrm{P}}$, respectively.
    Calculate $\lambda^{\text{SDC}}$
end for
Select the proper SDC with the largest $\lambda^{\text{SDC}}$
Jump to Step 5;
Step 4
Refine the data collection strategy;
Step 5
if $\mathrm{Q} > \mathrm{Q}_1$ then
    Input $\mathbf{\Gamma}$ into downstream tasks, Jump to return;
end if
if $\mathrm{Q}_0 < \mathrm{Q} < \mathrm{Q}_1$ then
    Input $\mathbf{\Gamma}$ into selected SDC module, Jump to Step 6;
end if
Step 6
Input predicted data into downstream tasks;
return Output downstream tasks performance $\mathrm{P}_{\text{task}}$.
```

TABLE III
DESCRIPTION OF DATASETS

Datasets	Changchun	Gowalla
#Users	2,239,529	373
#locations	2,185	131,329
#Check-in actions	49,716,815	2,963,373
Sparsity	98.9%	99.9%

2) Downstream tasks: Based on the collected data, we divide the map into (m=64) · (n=64), and (m=32) · (n=32) for Changchun and Gowalla, respectively. We employ two general downstream tasks: *Global Monitoring*: utilizes limited collected data to monitor the global situation of the target zone $m \cdot n$. The validation metric of global monitoring is θ_{GB}, which is calculated by Eq. (7). *POI recommendation*: utilizes the limited collected check-in data to deduce the next POI. The validation metric of POI recommendation is θ_{PR}, which is similar to [11].

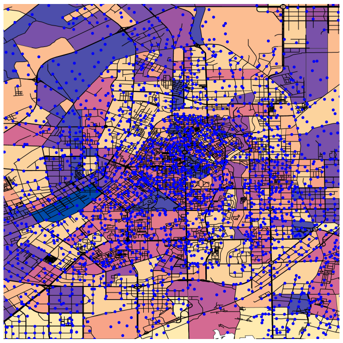

Fig. 5. Changchun trajectory dataset and target user dataset. The blue points in the Changchun trajectory dataset are the POI distribution. The black lines are the road networks.

3) Employed Baselines: For SDC methods, we employ three candidates:

basic MF [13], is widely used for data completion tasks, particularly in sparse mobile crowdsensing scenarios. By decomposing a spatiotemporal matrix into low-rank matrices, MF effectively captures the underlying patterns in the data, enabling accurate completion and prediction,

DMF [14], is a non-linear neural network approach that enhances the capabilities of traditional matrix factorization for data completion. By leveraging the power of non-linear modeling, DMF improves the accuracy of sparse data completion in various applications.

STDMF [5], is a powerful data-completion approach in spatiotemporal datasets. By simultaneously considering the spatial and temporal dimensions of the data, STDMF captures the complex relationships and patterns, resulting in accurate and comprehensive completion results.

Note that the θ_{SDC} of three methods is escalating, so is the computing complexity.

For global monitoring, we employ two baselines:

UrbanPy [17], combines fine-grained data analysis, network modeling, and machine learning algorithms and offers a comprehensive solution for understanding and predicting urban traffic dynamics.

UrbanSG [18], is a state-of-the-art technique for inferring fine-grained urban flow information. Using conditional generative adversarial networks, UrbanSG effectively captures the intricate patterns and relationships in urban traffic data, enabling accurate prediction and analysis for urban transportation planning and management.

For POI recommendation, we employ four baselines:

ToP [11], is an advanced approach for providing personalized and explainable recommendations for points-of-interest (POI). By incorporating time-dependent zone information and embedding techniques, ToP effectively captures the spatio-temporal characteristics of POIs, enhancing the accuracy and interpretability of the recommender system.

STiSAN [19], is a powerful technique for modeling spatio-temporal interactions in sequential data. By incorporating self-attention mechanisms, STiSAN effectively captures the dependencies and dynamics between elements, enabling accurate predictions and analysis in tasks such as video understanding, action recognition, and trajectory forecasting.

TABLE IV
OVERALL PERFORMANCE FOR APPLYING SDC-EVA IN GLOBAL MONITORING TASKS. ACC=MAPE

Datasets			Changchun						Gowalla					
Q			0.4	SDC-EVA	0.5	SDC-EVA	0.6	SDC-EVA	0.4	SDC-EVA	0.5	SDC-EVA	0.6	SDC-EVA
UrbanPy	MF	acc	0.5514	0.74	0.5314	**0.81**	**0.4332**	**0.91**	**0.5661**	**0.88**	0.5511	0.84	0.5472	0.74
		cost	374		386		394		346		344		435	
	DMF	acc	**0.4751**	**0.82**	**0.4641**	0.79	0.4413	0.81	0.5714	0.71	**0.5431**	**0.88**	**0.5401**	**0.83**
		cost	549		581		597		547		554		564	
	STDMF	acc	0.5211	0.71	0.5142	0.61	0.4371	0.78	0.5742	0.70	0.5612	0.64	0.5417	0.79
		cost	878		894		895		833		839		841	
UrbanSG	MF	acc	0.4613	0.66	0.4536	0.78	0.4435	0.78	0.4431	0.77	0.4325	0.77	0.4216	0.79
		cost	1430		1434		1466		1344		1364		1378	
	DMF	acc	0.3916	0.79	0.3811	**0.81**	0.3714	**0.79**	0.3887	**0.81**	0.3771	0.84	0.3664	0.77
		cost	1540		1548		1577		1514		1522		1533	
	STDMF	acc	**0.3764**	**0.81**	**0.3644**	0.80	**0.3517**	0.78	**0.3641**	0.76	**0.3632**	**0.88**	**0.3521**	**0.80**
		cost	1740		1744		1840		1701		1722		1756	

LSPSL [20], introduces two self-supervised optimization objectives to improve the long- and short-term preference modeling, which is the state-of-the-art POI recommendation method.

CTLE [21], is a bi-directional attention pre-trained location embedding model incorporating the spatial-temporal context in trajectories.

MAPE and HR@10 are the metrics for downstream methods' performance, following the definition in references of POI recommendation and global monitoring, respectively, and *cost* is the computing time unit count.

4) Settings: In our work, for all the baselines and our proposed methods, we have adopted a multi-step strategy to systematically configure the parameters: 1) Initial Setup: Based on prior knowledge and literature review, we started with a set of reasonable initial values for each parameter. 2) Automated Tuning: To refine these initializations, we employed an automated hyperparameter optimization technique called Bayesian optimization using the Optuna. This method efficiently explores the parameter space by balancing exploration and exploitation.3) Cross-Validation: During the tuning process, we utilized k-fold cross-validation to ensure that the selected parameters generalize well across different subsets of the data. For data completion, POI recommendation, and Global Monitoring tasks, we tune models' parameters and search the hyper-parameters to achieve their best performance for the specific target. Then, we fix the models' parameters. For our proposed framework SDC-EVA, we set $\alpha = 2, \tau = 0.5, \beta$=1+$e^{-\alpha_{\text{Q}}}$. We conduct all experiments on a server with 64GB RAM, a 12-core AMD 9 Ryzen 5900X CPU, and Nvidia RTX 3090 GPU.

B. Overall performance (RQ1)

We utilize MAPE, HR@10 for accuracy, and *cost* for efficiency to evaluate the different combinations of SDC methods and the downstream tasks. Note that we record the computing time under the same computing ability as the cost of each combination. We hope to see whether the SDC-EVA could select the most proper combination under different Qs. The results are reported in Table IV and Table V. We have the following discussion according to the results:

- In both downstream tasks, each method's performance (*acc*) becomes better when Q increases. Specifically, UrbanPy+DMF and Urban+STDMF achieve the best performance on Gowalla and Changchun in Global monitoring task, respectively; STiSAN+STDMF achieves the best performance on Gowalla and Changchun in POI recommendation task. We notice that no combination could dominate others on accuracy and efficiency, and there is no simple and clear rule to follow for picking SDC for specific downstream models.
- SDC-EVA can select the competitive combination for different downstream tasks with different downstream models with different Qs **before SMCS framework deploys and trains SDC models to test their effect, which reduces the cost and deployment difficulty greatly**. Note that there is no dominating combination for each scenario (Q, downstream models, candidate SDC models). SDC-EVA does solve the data quality and energy cost issues, which could be a guide for sparse mobile crowdsensing framework design.
- For both downstream tasks, we only test the Q in (0.4,0.5,0.6), which is validated in our prior experiments. However, the different real-world situations may lead to different thresholds for utilizing SDC (maybe not in the range of (0.4,0.667)). However, all the researchers could utilize our proposed methodology to customize their personal SDC-EVA module.

C. Ablation Study (RQ2)

For the ablation study, we build several variants of the SDC-EVA module. Note that the SDC-EVA considers both energy cost and downstream tasks' performance; we treat its result as ground truth, and we build the EVA-* series variants:

1) EVA-P: an evaluation module that considers only downstream tasks' performance, which means that $\lambda^{\text{SDC}} = \lambda_{\text{P}}$.

2) EVA-E: an evaluation module that considers only energy costs, which means that $\lambda^{\text{SDC}} = \lambda_{\text{E}}$.

3) EVA-P$_{\text{SDC}}$: an evaluation module that considers only the SDC model's performance, which means that we consider only LL_{SDC} in Eq. (10).

4) EVA-E$_{\text{compute}}$: an evaluation module that considers the computing energy cost is more important, e_{compute}=e_{unit}+e_{store}, which means that the Q range that we could combine with SDC is (0.4, 0.5) according to Eq. (17).

5) EVA-E$_{\text{unit}}$: an evaluation module that considers the sensing energy cost is more important, e_{unit}=2e_{compute}=2e_{store},

TABLE V
OVERALL PERFORMANCE FOR APPLYING SDC-EVA IN POI RECOMMENDATION TASKS. ACC=HR@10

Datasets			Changchun						Gowalla					
Q			0.4	SDC-EVA	0.5	SDC-EVA	0.6	SDC-EVA	0.4	SDC-EVA	0.5	SDC-EVA	0.6	SDC-EVA
ToP	MF	acc	0.1456	0.64	0.2146	**0.81**	0.3579	0.87	0.1567	**0.84**	0.1974	0.81	0.2111	0.79
		cost	647		666		679		584		591		594	
	DMF	acc	0.1974	**0.74**	0.2165	0.78	0.3666	**0.85**	0.1664	0.71	0.2464	**0.82**	0.2554	0.83
		cost	774		787		801		647		694		741	
	STDMF	acc	**0.2001**	0.72	**0.2324**	0.80	**0.3671**	0.77	**0.1674**	0.68	**0.2564**	0.79	**0.2877**	**0.87**
		cost	994		1004		1104		894		901		974	
STiSAN	MF	acc	0.1882	0.69	0.2104	0.77	0.2224	0.74	0.1977	0.76	0.2197	0.74	0.2233	0.77
		cost	1904		1914		1922		1764		1774		1801	
	DMF	acc	0.2194	0.71	0.2547	0.80	0.2234	0.80	0.2447	**0.81**	0.2644	**0.88**	0.2647	0.81
		cost	2041		2086		2111		1914		1923		1955	
	STDMF	acc	**0.2444**	**0.88**	**0.2679**	**0.82**	**0.3847**	**0.84**	**0.2547**	0.80	**0.2710**	0.87	**0.3910**	**0.89**
		cost	2146		2188		2246		2041		2077		2104	
LSPSL	MF	acc	0.1456	0.64	0.2146	**0.81**	0.3579	0.87	0.1567	**0.84**	0.1974	0.81	0.2111	0.79
		cost	647		666		679		584		591		594	
	DMF	acc	0.1974	**0.74**	0.2165	0.78	0.3666	**0.85**	0.1664	0.71	0.2464	**0.82**	0.2554	0.83
		cost	774		787		801		647		694		741	
	STDMF	acc	**0.2001**	0.72	**0.2324**	0.80	**0.3671**	0.77	**0.1674**	0.68	**0.2564**	0.79	**0.2877**	**0.87**
		cost	994		1004		1104		894		901		974	
CTLE	MF	acc	0.1882	0.69	0.2104	0.77	0.2224	0.74	0.1977	0.76	0.2197	0.74	0.2233	0.77
		cost	1904		1914		1922		1764		1774		1801	
	DMF	acc	0.2194	0.71	0.2547	0.80	0.2234	0.80	0.2447	**0.81**	0.2644	**0.88**	0.2647	0.81
		cost	2041		2086		2111		1914		1923		1955	
	STDMF	acc	**0.2444**	**0.88**	**0.2679**	**0.82**	**0.3847**	**0.84**	**0.2547**	0.80	**0.2710**	0.87	**0.3910**	**0.89**
		cost	2146		2188		2246		2041		2077		2104	

which means that the Q range that we could utilize SDC is (0.4, 0.75) according to Eq. (17).

The results are reported in Table VI.

TABLE VI
ABLATION STUDY FOR SDC-EVA

Variant	Q	Global Mon Task	POI Rec Task
EVA-P	0.4	F	T
	0.5	F	F
	0.6	F	T
	0.7	-	-
EVA-E	0.4	F	F
	0.5	T	F
	0.6	T	F
	0.7	-	-
EVA-P$_{\text{SDC}}$	0.4	T	T
	0.5	T	F
	0.6	T	F
	0.7	-	-
EVA-E$_{\text{compute}}$	0.4	T	T
	0.5	T	T
	0.6	-	-
	0.7	-	-
EVA-E$_{\text{unit}}$	0.4	T	T
	0.5	T	T
	0.6	T	T
	0.7	T	T
SDC-EVA	0.4	T	T
	0.5	T	T
	0.6	T	T
	0.7	-	-

From the ablation study, we have the following discussion:

- All variants can choose the right combination (marked by T) for different tasks. However, the performance of all the variants is not as stable as SDC-EVA. Specifically, when we only consider P$_{\text{SDC}}$, EVA-P$_{\text{SDC}}$ achieves the best correct rate among variants (4/6). It indicates we should consider energy cost and task performance when selecting SDC in SMCS.
- Note that for different settings of $e_{\text{compute}}, e_{\text{unit}}$ (EVA-E$_{\text{compute}}$ and EVA-E$_{\text{unit}}$), the correct rate of choosing SDC is not affected, which means that our proposed SDC-EVA can handle the different situations of computing and sensing energy cost, validating its generalization ability for various SMCS scenarios.

D. Parameter Study and A Case Study (RQ3)

We set different parameters to study their effects on SDC-EVA. Specifically, we focus on several important parameters: metric weight τ and performance parameter β={AIC, BIC}. Without loss of generality, we set the Q from 0.4 to 0.6, with each step 0.04. We focus on DMF+ToP (POI recommendation) to check the effect of parameters on λ^{SDC} on both datasets.

The results are reported in Figure 6 and Figure 7. From the results, we have the following discussion:

- The different values of τ force SDC-EVA to focus on different aspects of candidate SDC methods. With a small τ, SDC-EVA pays more attention to energy cost where a low-complexity method is benefitted, and vice versa.
- For β, by analyzing the floating, we conclude that AIC is proper for large datasets while BIC is proper for small datasets. The reason is that AIC calculates the parameter number of the candidate methods, while BIC is more sensitive to the scale of datasets, which is indicated in Eq. (10).

Besides, we give an example of how to utilize SDC-EVA in real-world scenarios. In this scenario, the city planners try to achieve a downstream task T, with collected data D. In the traditional SMCS framework, if the D is sparse, city planners would utilize the SDC module to predict the missing data and finish the task T without any consideration of energy cost. However, when we introduce SDC-EVA, we should employ $e_{\text{unit}}, e_{\text{store}}$, and e_{compute} to calculate the additional energy cost **before** we utilize SDC. Note that to utilize SDC-EVA, we should deduce the Q$_0$ and Q$_1$. Specifically, Q$_0$ is an empirical parameter that is pre-defined (in this experiment, we set it to

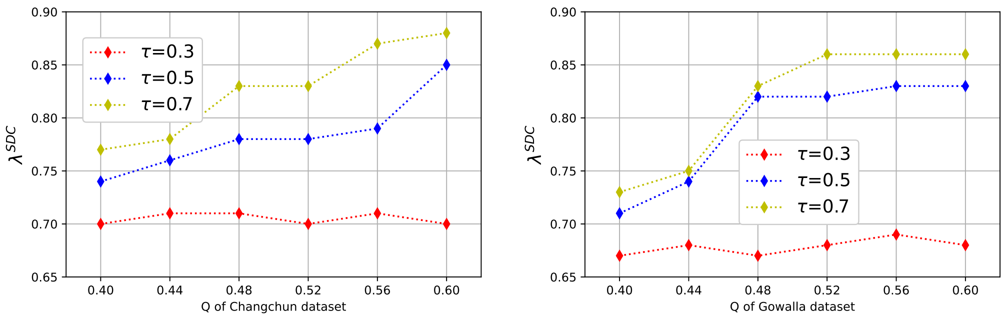

Fig. 6. Parameter study of τ of ToP+DMF on both datasets.

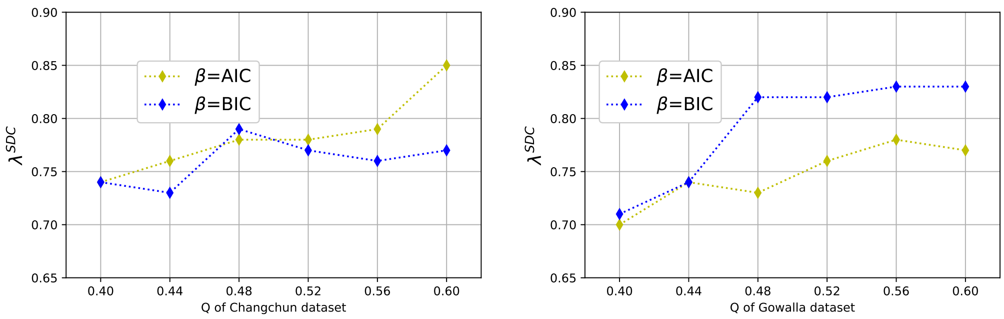

Fig. 7. Parameter study of β of ToP+DMF on both datasets

0.4) and $Q_1=(e_{\text{unit}}+e_{\text{store}})/(e_{\text{unit}}+e_{\text{store}}+e_{\text{compute}})$. The SDC-EVA model could help city planners decide whether to utilize the SDC module or collect more data to achieve task T without introducing additional energy costs, with the calculation of Q, which is efficient for computing, as shown in Figure 8.

Besides, our method could be extended to different SDC methods when we could calculate their accurate energy costs. Note that utilizing SDC-EVA in distributed mode may introduce different issues for calculating the cost of the SDC module; we should carefully design the modified SDC-EVA framework for a distributed computing environment. However, we think that, with their accurate energy costs, a better trade-off could be achieved. We may explore this in our next work.

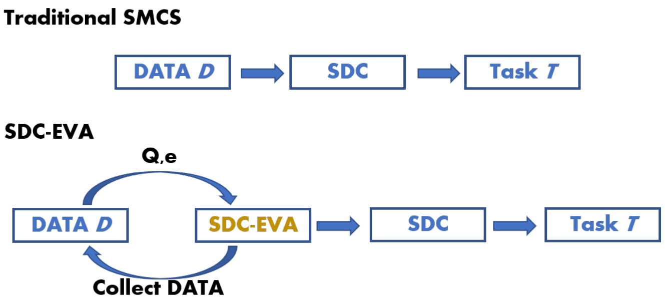

Fig. 8. An example to utilize SDC-EVA in real-world scenarios.

E. Experiment Conclusion

The above three experiment results validate that:

- The results of SDC-EVA could be the heuristic module before the SMCS deployment (solve sensing data quality issue).
- SDC-EVA could trade off the energy-saving benefit and the deployment SDC costs for applying SDC in the SMCS framework (relieve energy cost issue).

VI. Related Work

Our work is closely related to sparse mobile crowdsensing and data completion models.

A. Sparse Mobile Crowdsensing

Sparse mobile crowdsensing is a variant of mobile crowdsensing. This popular technology leverages the ubiquity of mobile devices equipped with various sensors to perform urban crowdsensing tasks [6], [18], [22], [23]. Compared to traditional wireless sensor networks (WSNs), MCS offers unique advantages in terms of scalability and flexibility. However, the cost associated with recruiting many participants can be prohibitive. To address this limitation, SMCS is proposed as a solution that leverages limited conditions, such as sensing specific areas or collecting a subset of data, to reduce energy consumption. Several studies have been conducted to optimize SMCS frameworks. Li et al. [12] focused on task allocation and achieved a diverse and spatially optimized coverage within a limited budget for different application scenarios. Wang et al. [7] proposed a decentralized matrix factorization algorithm to enable sparse MCS without the need for location or data aggregation to a central server. Bian et al. [24] addressed the problem of selecting data instances to maximize the performance of trained models under budget constraints. However, existing SMCS frameworks are typically two-stage and do not consider the energy consumption and impact of the SDC module on downstream tasks.

B. Data Completion Models

Sparse data completion aims to solve the problem of completing a sparse spatiotemporal matrix [4], [25], [26]. In real-world SMCS scenarios, neighboring time slots or subareas often exhibit close values, implying that the rank of the spatiotemporal matrix should be low. Matrix factorization (MF) has been widely adopted for data inference, as it effectively handles spatiotemporal data with strong linear characteristics. However, MF may introduce significant errors when processing non-linear data. To address this limitation, Fan and Cheng [13] proposed Deep Matrix Factorization (DMF), which leverages non-linear neural networks to enhance the capabilities of MF. Wang et al. [7] extended DMF with an end-to-end model for sparse industrial sensing and prediction, enabling high-precision future predictions in addition to current time slot sensing. The emergence of graph neural networks (GNNs) has also contributed to data completion models. Zhang et al. [27] introduced Inductive Graph-based Matrix Completion (IGMC), a novel approach that leverages GNNs for matrix completion. IGMC has demonstrated superior performance compared to DMF for certain datasets. Xie et al. [28] proposed a two-phase matrix completion-based data recovery scheme that exploits the inherent characteristics of environmental data to recover missing values. However, existing SDC models primarily focus on improving the accuracy of data completion and often overlook the trade-off between computing cost and sensing cost, which is crucial for the practical application of SDC in SMCS.

VII. Concluding Remarks

In this paper, we qualitatively and quantitatively investigate the impact of SDC on the SMCS paradigm. We initially establish an upper bound (λ_{P}) for performance when utilizing SDC in SMCS under different levels of sensing data sparsity (Q). Subsequently, we propose a practical and flexible framework (SDC-EVA) for applying various SDC methods in SMCS while considering computing complexity, storage space requirements, and other costs involved. Notably, our proposed framework enables researchers to assess the necessity and feasibility of introducing SDC into SMCS before designing and deploying their systems based on specific data sparsity levels. We conduct experiments using real-world scenarios involving diverse combinations of SDC techniques with downstream tasks; our results demonstrate that SDC-EVA significantly improves SMCS as a heuristic module.

In the future, we will explore how to bridge the downstream tasks and the whole SMCS framework, which may be built upon the basic SDC-EVA framework. Besides, the data distribution, including the relationship between missing and observed data, may become the next key factor in developing SMCS application scenarios. Moreover, researchers could combine side information and NLP technology with the proposed SDC-EVA module for specific MCS tasks.

VIII. Acknowledgment

This work is supported by the Natural Science Foundation of China No. 62472196, Jilin Education Science Foundation JJKH20221010KJ, Jilin Science and Technology Research Project 20230101067JC.

References

[1] R. K. Ganti, F. Ye, and H. Lei, "Mobile crowdsensing: current state and future challenges," *IEEE Commun. Mag.*, vol. 49, no. 11, pp. 32–39, 2011.

[2] S. Zhao, G. Qi, T. He, J. Chen, Z. Liu, and K. Wei, "A survey of sparse mobile crowdsensing: Developments and opportunities," *IEEE Open J. Comput. Soc.*, vol. 3, pp. 73–85, 2022.

[3] H. Wang, Y. Yang, E. Wang, W. Liu, Y. Xu, and J. Wu, "Truthful user recruitment for cooperative crowdsensing task: A combinatorial multi-armed bandit approach," *IEEE Trans. Mob. Comput.*, vol. 22, no. 7, pp. 4314–4331, 2023.

[4] Y. Xu, E. Wang, Y. Yang, and Y. Chang, "A unified collaborative representation learning for neural-network based recommender systems," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 11, pp. 5126–5139, 2022.

[5] E. Wang, M. Zhang, Y. Xu, H. Xiong, and Y. Yang, "Spatiotemporal fracture data inference in sparse urban crowdsensing," in *IEEE INFOCOM 2022 - IEEE Conference on Computer Communications, London, United Kingdom, May 2-5, 2022*. IEEE, 2022, pp. 1499–1508.

[6] Y. Xu, X. Cai, E. Wang, W. Liu, Y. Yang, and F. Yang, "Dynamic traffic correlations based spatio-temporal graph convolutional network for urban traffic prediction," *Inf. Sci.*, vol. 621, pp. 580–595, 2023.

[7] E. Wang, M. Zhang, X. Cheng, Y. Yang, W. Liu, H. Yu, L. Wang, and J. Zhang, "Deep learning-enabled sparse industrial crowdsensing and prediction," *IEEE Trans. Ind. Informatics*, vol. 17, no. 9, pp. 6170–6181, 2021.

[8] E. Wang, M. Zhang, W. Liu, H. Xiong, B. Yang, Y. Yang, and J. Wu, "Outlier-concerned data completion exploiting intra- and inter-data correlations in sparse crowdsensing," *IEEE/ACM Trans. Netw.*, vol. 31, no. 2, pp. 648–663, 2023.

[9] X. Wei, Z. Li, Y. Liu, S. Gao, and H. Yue, "SDLSC-TA: subarea division learning based task allocation in sparse mobile crowdsensing," *IEEE Trans. Emerg. Top. Comput.*, vol. 9, no. 3, pp. 1344–1358, 2021.

[10] F. Liu, B. Zhu, S. Yuan, J. Li, and K. Xue, "Privacy-preserving truth discovery for sparse data in mobile crowdsensing systems," in *IEEE Global Communications Conference, GLOBECOM 2021, Madrid, Spain, December 7-11, 2021*. IEEE, 2021, pp. 1–6.

[11] E. Wang, Y. Xu, Y. Yang, F. Yang, C. Liu, and Y. Jiang, "Top: Time-dependent zone-enhanced points-of-interest embedding-based explainable recommender system," in *40th IEEE Conference on Computer Communications, INFOCOM 2021, Vancouver, BC, Canada, May 10-13, 2021*. IEEE, 2021, pp. 1–10.

[12] J. Li, J. Wu, and Y. Zhu, "Data utility maximization when leveraging crowdsensing in machine learning," in *26th IEEE/ACM International Symposium on Quality of Service, IWQoS 2018, Banff, AB, Canada, June 4-6, 2018*. IEEE, 2018, pp. 1–6.

[13] J. Fan and J. Cheng, "Matrix completion by deep matrix factorization," *Neural Networks*, vol. 98, pp. 34–41, 2018.

[14] D. Xu, C. Ruan, E. Körpeoglu, S. Kumar, and K. Achan, "Rethinking neural vs. matrix-factorization collaborative filtering: the theoretical perspectives," in *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, ser. Proceedings of Machine Learning Research, vol. 139. PMLR, 2021, pp. 11 514–11 524.

[15] S. I. Vrieze, "Model selection and psychological theory: a discussion of the differences between the akaike information criterion (aic) and the bayesian information criterion (bic)." *Psychological methods*, vol. 17, no. 2, p. 228, 2012.

[16] A. M. Robertson and P. Willett, "An upperbound to the performance of ranked-output searching: Optimal weighting of query terms using a genetic algorithm," *J. Documentation*, vol. 52, no. 4, pp. 405–420, 1996.

[17] K. Ouyang, Y. Liang, Y. Liu, Z. Tong, S. Ruan, Y. Zheng, and D. S. Rosenblum, "Fine-grained urban flow inference," *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 6, pp. 2755–2770, 2022.

[18] X. Zhang, Y. Xu, Y. Li, and Y. Yang, “Fine-grained urban flow inferring via conditional generative adversarial networks,” in *Web and Big Data - 6th International Joint Conference, APWeb-WAIM 2022, Nanjing, China, November 25-27, 2022, Proceedings, Part III*, ser. Lecture Notes in Computer Science, vol. 13423. Springer, 2022, pp. 420–434.

[19] E. Wang, Y. Jiang, Y. Xu, L. Wang, and Y. Yang, “Spatial-temporal interval aware sequential POI recommendation,” in *38th IEEE International Conference on Data Engineering, ICDE 2022, Kuala Lumpur, Malaysia, May 9-12, 2022*. IEEE, 2022, pp. 2086–2098.

[20] S. Jiang, W. He, L. Cui, Y. Xu, and L. Liu, “Modeling long- and short-term user preferences via self-supervised learning for next POI recommendation,” *ACM Trans. Knowl. Discov. Data*, vol. 17, no. 9, pp. 125:1–125:20, 2023. [Online]. Available: https://doi.org/10.1145/3597211

[21] Y. Lin, H. Wan, S. Guo, and Y. Lin, “Pre-training context and time aware location embeddings from spatial-temporal trajectories for user next location prediction,” in *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*. AAAI Press, 2021, pp. 4241–4248. [Online]. Available: https://doi.org/10.1609/aaai.v35i5.16548

[22] R. K. Ganti, F. Ye, and H. Lei, “Mobile crowdsensing: current state and future challenges,” *IEEE Commun. Mag.*, vol. 49, no. 11, pp. 32–39, 2011.

[23] D. Zhang, L. Wang, H. Xiong, and B. Guo, “4w1h in mobile crowd sensing,” *IEEE Commun. Mag.*, vol. 52, no. 8, pp. 42–48, 2014.

[24] J. Bian, H. Xiong, Y. Fu, and S. K. Das, “CSWA: aggregation-free spatial-temporal community sensing,” in *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*. AAAI Press, 2018, pp. 2087–2094.

[25] Y. Xu, E. Wang, Y. Yang, and H. Xiong, “GS-RS: A generative approach for alleviating cold start and filter bubbles in recommender systems,” *IEEE Trans. Knowl. Data Eng.*, vol. 36, no. 2, pp. 668–681, 2024.

[26] Y. Xu, Y. Yang, J. Han, E. Wang, F. Zhuang, J. Yang, and H. Xiong, “Neuo: Exploiting the sentimental bias between ratings and reviews with neural networks,” *Neural Networks*, vol. 111, pp. 77–88, 2019.

[27] M. Zhang and Y. Chen, “Inductive matrix completion based on graph neural networks,” in *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020.

[28] K. Xie, X. Ning, X. Wang, D. Xie, J. Cao, G. Xie, and J. Wen, “Recover corrupted data in sensor networks: A matrix completion solution,” *IEEE Trans. Mob. Comput.*, vol. 16, no. 5, pp. 1434–1448, 2017.

Yuanbo Xu received his B.E. degree in computer science and technology from Jilin University, Changchun, in 2012, his M.E. degree in computer science and technology from Jilin University, Changchun, in 2015, and his Ph.D. in computer science and technology from Jilin University, Changchun, in 2019. He is currently an associate professor in the Department of Artificial Intelligence at Jilin University, Changchun. He is also a visiting scholar in the Management Science and Information Systems Department at Rutgers, the State University of New Jersey. His research interests include data mining applications, recommender systems, and mobile computing. He has published research papers in journals such as TKDE, TMC, TMM, TVT, TNNLS, TKDD and conferences such as ICDE, INFOCOM, CIKM, IWQoS, and ICDM.

Jiawei Liu received the ME degree in software engineering from Jilin University, Changchun, in 2021, where he is currently pursuing the PhD degree in computer science and technology. His research interests include applications of reinforcement learning and navigation for AUVs.

En Wang received his B.E. degree in software engineering from Jilin University, Changchun, in 2011, his M.E. degree in computer science and technology from Jilin University, Changchun, in 2013, and his Ph.D. in computer science and technology from Jilin University, Changchun, in 2016. He is currently a Professor in the Department of Computer Science and Technology at Jilin University, Changchun. He is also a visiting scholar in the Department of Computer and Information Sciences at Temple University in Philadelphia. His current research focuses on the efficient utilization of network resources, scheduling and drop strategy in terms of buffer-management, energy-efficient communication between human-carried devices, and mobile crowdsensing.

Bo Yang is currently a professor in the College of Computer Science and Technology, Jilin University. He is also the director of the Key Laboratory of Symbolic Computation and Knowledge Engineering, Ministry of Education, China. His current research interests are in the areas of data mining, complex network analysis, self-organized and self-adaptive multi-agent systems, with applications to knowledge engineering and intelligent health informatics.

Dongming Luan received his B.E. degree in Software Engineering from Jilin University, Changchun, Jilin, China, in 2017. Now, he is a postgraduate student in College of Computer Science and Technology in Jilin University, Changchun, Jilin, China. His current research interest is Mobile Crowdsensing.

Yongjian Yang received his B.E. degree in automatization from Jilin University of Technology, Changchun, Jilin, China in 1983; his M.E. degree in computer communication from Beijing University of Post and Telecommunications, Beijing, China in 1991; and his Ph.D. in software and theory of computer from Jilin University, Changchun, Jilin, China in 2005. He is currently a professor and a PhD supervisor at Jilin University, the Vice Dean of the Software College of Jilin University, Director of Key lab under the Ministry of Information Industry, Standing Director of the Communication Academy, and a member of the Computer Science Academy of Jilin Province. His research interests include: network intelligence management, wireless mobile communication and services, and wireless mobile communication.

Jing Deng (Fellow, IEEE) is the Bank of America Distinguished Professor and Head of the Department of Computer Science at UNC Greensboro, U.S.A. Dr. Deng visited the Department of Electrical Engineering at Princeton University and the Department of Electrical and Computer Engineering, WINLAB at Rutgers University in Fall of 2005. He was with the Department of Computer Science at the University of New Orleans from 2004 to 2008. He served as a Research Assistant Professor in the Department of Electrical Engineering and Computer Science at Syracuse University from 2002 to 2004. He received his Ph.D. degree from School of Electrical and Computer Engineering at Cornell University, Ithaca, New York, USA in January, 2002. He received his M.E. and B.E. degrees in Electronic Engineering at Tsinghua University in 1994 and 1997, respectively. Dr.Deng is an associate editor of IEEE Transactions on Mobile Computing. He served as an editor of IEEE Transactions on Vehicular Technology between 2008-2018. He received the Test-of-Time Award presented by the ACM Special Interest Group on Security, Audit and Control (SIGSAC) in 2013. Dr. Deng's research interests include online social networks, wireless networks, and network security. His research webpage is at http://www.uncg.edu/~j_deng/