Participation Selection in Mobile Crowd Sensing with Diversity Ensures

Jiahui Jin
Computer Science Department
Emory University
helen.jin@emory.edu

Dr. Ting Li
Computer Science Department
Emory University
ting.li@emory.edu

***Abstract*—Mobile Crowd Sensing (MCS) systems are reshaping the data collection process by utilizing the capabilities of smart devices. Traditional MCS models employ cost-effective algorithms, treating their participants as unit workers. However, this approach overlooks the nuance of the MCS system, which is not limited to collecting quantitative data such as weather or traffic conditions. Instead, MCS also has the potential to perform opinionative tasks. By selecting diverse participants to offer opinionative data, we can gain unique but easily overlooked perspectives. This paper proposes a model to improve diversity in participant selection to ensure that diverse social identities and voices are considered, thereby promoting equality. We first divided participants into distinct similarity groups, each characterized by shared behavior patterns, responses, interests, etc., and then selected participants by prioritizing diversity. The simulations presented in this paper demonstrated the effectiveness of our proposed model.**

***Index Terms*—mobile crowd sensing, task assignment, participation diversity, POS tagging, opinion mining**

I. Introduction

The MCS system can be divided into three stages: (1) An agent publishes sensing tasks online, providing information such as the target area and duration; (2) A group of appropriate MCS participants collects sensing data from their surroundings using mobile devices and uploads them to a cloud server; (3) The data contributed by the selected group of participants are integrated, producing a spatiotemporal view of the phenomenon of interest in a city.

Selection is a vital part of the above process. In stage two, a group of appropriate participants needs to be selected. In stage three, relevant data is collected to answer the task. It is important to ask: "Who is selected? And whose data is collected?" MCS is a powerful information-collecting network that has the potential to influence future city planning and development. Hence, it is vital to understand whose voice is heard and who is excluded from the system. These questions are essential to ensure that all groups in current society can be part of the system, especially those traditionally marginalized.

Previously, in data collection, not enough attention was given to the more vulnerable groups, such as women, children, the elderly, the LGBTQ community, people with disabilities, people of color, and many others. Hence, this project aims to design an MCS selection system to collect diverse needs, interests, and concerns, ultimately ensuring diversity and equality in city planning.

In most existing MCS system studies, the participant design leans towards utilitarian principles. Most algorithms focus on minimizing cost, maximizing time efficiency, or improving data quality [8], [9]. This approach works for quantitative interests but overlooks qualitative interests needing varied perspectives. Nevertheless, a few MCS studies do take participants' identities into account. In [5], the authors compared different selection algorithms. Through simulations, the authors proposed UR-GAT (Unit Reward-based Greedy algorithm by type) as the best to maximize user type diversity under a set task budget and time constraints. In [6], the authors examined the intersection of user diversity and social effects in MCS, proposing a reward mechanism that encourages data diversity by utilizing users' social relationships.

In both [5] and [6], the foundations of the studies relied on the presupposition that pre-defined participant groupings already exist. However, in a more realistic setting, the system starts with a raw data set provided by participants. The challenge is to discern the social attributes of participants from their opinionated data, as well as establish participant groupings. To ensure diverse opinion inclusion in MCS data collection, a mechanism must be developed to extract participant attributes and group them. These challenges are addressed in this project. A model is trained based on an open Yelp data set, assuming that MCS is used for collecting feedback for local restaurants. In the future, this can be applied to other parts of the city, offering diverse feedback for public facilities.

Fig 1 illustrates the general process of the new MCS system model. Firstly, participants are divided into similarity groups according to their attributes analyzed from historical data. We utilized some topic extraction and clustering techniques in the natural language processing field to achieve this purpose. Secondly, tasks are classified as efficiency- or diversity-dominated. For the latter, We designed a selection logic that prioritizes diverse participant identities.

To group participants into similarity groups, we need to extract participants' attributes from their prior history. In the context of this project, we need to group participants by analyzing their previous restaurant reviews.

In the field of Natural Language Processing (NLP), there

979-8-3503-7099-7/24/$31.00 ©2024 IEEE

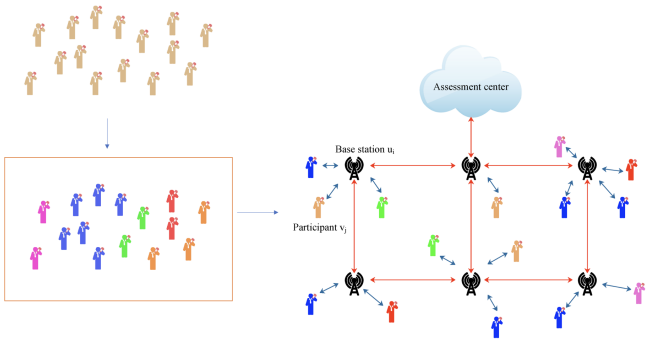

Fig. 1: Model Development

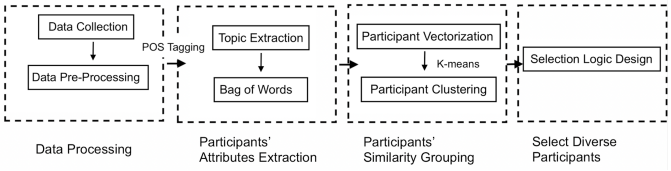

Fig. 2: Frame Work of The Project

already exist some prior works focusing on analyzing short comments and texts. Most revolve around e-commerce and social platforms, where some developed techniques and models can serve as references for this project.

In [1], the authors focused on e-commerce review summarizing. They used sentence and word tokenization to break down reviews into individual words. The words are then Part of Speech (POS) tagged to identify product features and associated sentiments (positive, negative, or neutral). These feature-opinion pairs are subsequently utilized to create a summarized graphical representation of user opinions, facilitating quick decision-making for users and insightful feedback for producers. The process primarily targeted reviews on Amazon for a range of products like smartphones and clothing.

Both [2], [4] studied Latent Dirichlet Allocation (LDA) as a common NLP method for extracting hidden topics. In [2], the author applied LDA to analyze reviews from 18,067 Airbnb users. Alongside, sentiment analysis is employed using the Chinese language processing library, SnowNLP, to qualify user satisfaction levels associated with these topics. In [4], the authors addressed the issues in traditional LDA, integrating Word Embeddings, specifically the Skip-gram model, into LDA to improve the topic modeling process.

In [3], the authors adapted the RFM (Recency, Frequency, Monetary) customer value model to evaluate user comments on e-commerce platforms. This is extended to the RFM-RES model, adding Responsiveness, Effectiveness, and Status. The K-means clustering algorithm is used to analyze user comments, clustering reviews into high-value and low-quality groups. This method is designed to facilitate decision-making and improve efficiency in handling customer feedback.

As discussed in the above examples, there exist mature studies around analyzing the content of text: applying POS tagging to mark the word's attributes, performing sentimental analysis, and using various clustering algorithms to group valuable comments and topics.

As illustrated in Fig 2, in this project, techniques including POS tagging and K-means clustering is applied in the MCS context for extracting topics, predicting participant attribute accordingly, and clustering participant into similarity group. Finally, a selection algorithm is designed, allowing for adjustments regarding the number of participants wanted and the number of similarity groups selected.

Overall, this is the first system that provides a complete MCS Selection System, extracting participants' social attributes from raw data, considering diversity in MCS participant selection, and handling both efficiency-dominated tasks and diversity-dominated tasks.

II. Problem Definition and Solution

A. Problem Definition

Let $P = \{p_1, p_2, \ldots, p_N\}$ be the set of participants, where N is the total number of participants. A person's opinion, interest, and preference could be reflected in his comments/speech. Hence, we classify these contextual data into a set of categories as $C = \{c_1, c_2, \ldots, c_M\}$, where M is the system parameter representing the total number of categories and c_i represents a distinct interest. Each participant p_i (for $1 \leq i \leq N$) has a vector $\mathbf{v_i} = (x_{i1}, x_{i2}, \ldots, x_{iM})$, which is a quantitative representation on M categories. Of course, we can cluster all participants by clustering algorithms (like k-mean, etc.) into different groups based on their $\mathbf{v_i}$s and let g_i denote the group which the participant p_i belongs to, and g_i is one of the G groups. Each participant also has a set of locations L_i standing for where this participant ep_i could be. Overall, each participant is denoted as $p_i = L_i, g_i$, where the L_i is a set and g_i is an integer. We formulate our current problem regardless of the time scale since it only affects our problem size for locations and participant groupings, which should be easily handled. In MCS, we also have a task $T = t_1, t_2, \ldots, t_K$, where K denotes the total number of locations that this task requests participants to visit. Each subtask t_k has two parameters f_k and l_k, where f_k represents the number of participants that fulfill this task, and l_k for this task's location. We could easily calculate the minimum distance from a participant to a subtask by their locations, and we name it d_{ik}. Besides, each task has another two requirement diversity requirements D_k, which denotes the participants that fulfill this task have to come from at least D_k different groups, and the set of selected participants, SP_k. Overall, our goal for this work is to select a subset of participants, such as all the locations where task request could be visited, and these participants come from at least D_k different groups. It could be formulated as below.

$$\begin{aligned} &\min \sum_{P,T} d_{ik} \\ &\text{s.t.} f_k > |SP_k| \textit{ for each subtask } k. \\ &|G|_{p_i \in SP_k} > D_k \textit{ per subtask } k \end{aligned} \tag{1}$$

Here, we use the distance as the efficiency indicator. It could be easily extended to other metrics. It's worth noting that this model will become the traditional participation selection problem if the diversity requirement is smaller than or equal to 1. In the following subsection, we will resolve the above optimization problem by elaborating the details, including the construction of the vector $\mathbf{v_i}$ for each participant, grouping participants, and participant selection with diversity guarantee. These steps have been illustrated in Fig 2.

B. Participant Vectorization

We conduct participant vectorization based on the Yelp review open data set. Each participant provides multiple comments for various local restaurants in the great New York area. We first clean those data to better analyze participants' attributes by removing punctuation, uniforming the test by converting all text to UTF-8 encoding lowercase, and eliminating users who contributed less than five comments. Even though we care about participant diversity and want to pay attention to minorities, these participants who provide less than five comments will lead to a huge bias with insufficient data. Inspired by [1], to ensure the accuracy of topic extraction, only nouns are considered potential topics. Each sentence in the comments is broken down into individual words, and the stopwords have been removed. Consider an example comment from the data set:

"Give this place a try, the banana bread french toast was delicious." After cleaning, the sentence is marked as "place, banana, bread, french, toast." We have examined some popular topic extraction models in natural language processing, including LDA, TF*IDF [4], [7]. Numerically comparing the quality of topic extraction between algorithms is challenging. Furthermore, due to paragraph constraints, LDA and TF*IDF result will not be presented here. In short, LDA and TF*IDF algorithm failed to generate satisfying topic extractions.

This is caused by the particular context of our project. In most popular topic extraction models, words that are mentioned more frequently are more likely to be extracted as a topic. However, the purpose of this project is to focus on the needs of minority groups, where their needs might not be mentioned as frequently. For example, high chairs for children, wheelchair access for the disabled, or dietary options for vegetarians might not be the most talked-about aspects in reviews but are crucial to certain groups.

Hence, we set a minimum frequency threshold for word frequency and decide to manually extract word topics for all words above this threshold. This process ensures the inclusion of topics that may be significant for certain groups, even if they're not frequently mentioned in reviews.

As a result, we manually organize akin words into 13 distinct categories. Here, 13 is not fixed, and it could be changed to other values. We choose 13 for better performance with the data set we have. Each of the 13 categories represented an area of interest in dining, respectively 'vegan,' 'cultural foods,' 'dessert,' 'serving speed,' 'beverages,' 'serving speed,' 'baby friendly,' 'locations establishments,' 'days of the week,' 'time of the day,' 'electronic devices and internet,' 'customer related,' 'sport-related.' These categories serve as our criteria for measuring the attributes of the participants.

Given the categories and participants' comments, we construct and normalize each participant vector. This normalization process involved dividing each vector $\mathbf{v_i}$ by its magnitude. The magnitude (or length) of a vector $\mathbf{v_I}$ is given by the square root of the sum of the squares of its components, which can be expressed mathematically as follows:

$$|\mathbf{v_I}| = \sqrt{\sum_{i=1}^{13} x_{ij}^2} \tag{2}$$

Consequently, the normalized (or unit) vector $\mathbf{v_i}$ for each participant p_i is obtained by dividing $\mathbf{v_i}$ by its magnitude $|\mathbf{v_I}|$. This can be expressed as:

$$\mathbf{v_i} = \left(\frac{x_{i1}}{|\mathbf{v_I}|}, \frac{x_{i2}}{|\mathbf{v_I}|}, \ldots, \frac{x_{i13}}{|\mathbf{v_I}|} \right) \tag{3}$$

C. Divide Participants into Similarity Groups

After vectorization, the next step is to group the participants based on their similarity in interests. This is accomplished by using k-means clustering [3], a widely-used technique in machine learning and data mining for partitioning data into distinct, non-overlapping subgroups.

K-means algorithm calculates the Euclidean distance between each vector (participant) in the multi-dimensional space. In the context of this project, vectors that are closer in distance represent participants who are similar to each other. The algorithm optimizes the grouping by minimizing the distance between vectors within the same cluster and maximizing the distance between each cluster. Vectors within the same cluster represent participants within the same similarity group.

The number of clusters can be adjusted according to specific requirements, offering flexibility in adjusting the degree of diversity among participants grouping.

D. Participant Selection Algorithm with the Diversity Guarantee

After establishing similarity groups, a selection algorithm, as shown in as shown in algorithm 1, is designed to account for both diversity and efficiency. This algorithm incorporates two tuneable parameters: the required participant count for MCS tasks, and the minimal diverse groups involved. The algorithm consists of two parts. Firstly, we calculate the distance between each task to all users and put all participants into a min-heap, ordered by the distance. Secondly, participants are selected based on their distance to subtasks if the diversity requirement D is satisfied. Otherwise, we will prioritize the diversity and select the participant who is from a different group. The detailed algorithm is illustrated in algorithm Participant Selection with the Diversity Guarantee

Our algorithm put all the participants into a minimal heap ordered by their distance to the sub task t_k. We initialize the system parameter from Line 2- 4 and start to select

Algorithm 1 Participant Selection with the Diversity Guarantee

Input: a task T with a set of sub-tasks with (f_k, l_k, SP, D_k) for sub-task k, a set of participants with (L_i, g_i) for participant p_i.
Output: A set of participants S.

1: **for all** $t_k \in T$ **do**
2: Initialize an participant min-Heap MP ordered by their distance to t_k.
3: group_Count = 0, f_k = 0.
4: Initialize a empty list $selected\_g$ and an empty min-Heap $deferred_p$.
5: **while** $f_k < SP$ **do**
6: p = get the top participant from MP.
7: **if** group_count $< D_k$ **then**
8: **if** g_i is not existing in the $selected\_g$ list **then**
9: Add g_i to $selected\_g$.
10: Add p to S.
11: increase group_count and f_k by 1, respectively.
12: **else**
13: Add p to $deferred\_p$.
14: **else**
15: Add p from the top of $deferred_p$ to S.
16: increase f_k by 1.
17: reset MP, group_Count, f_k, $selected\_g$ and $deferred_p$
18: **return** S

participant from Line 5 to Line 16. Line 7 -14 ensures the diversity requirement D_k by select participants from different groups and defers the participant whose distance is small but selecting him won't contribute to the diversity requirement. Line 15 - 16 selects the participants who has the smallest distance regardless of their group identify. Line 17 reset all the system parameter for next sub task. And we return the selected participants in Line 18.

III. Simulation Result

A. Data Pre-Processing

We used the user review data from the Yelp open dataset, which has 10,376 entries. Each entry records the participant's id, the restaurants id, and one participant's comments for a local restaurant. After data cleaning, we obtained a data set containing 2749 restaurants, 951 participants, and 7697 comments. Each sentence in comments broken down into individual words. To better classify nouns from others, each word was POS-tagged using udpipe package in R. Consider an example comment from data set:

"Give this place a try, the banana bread french toast was delicious."

After POS tagging, the sentence was marked as:
('give','onix'),('this','SMART'),('place','onix'),
('a','SMART'),('try','SMART'),('the','snowball'),
('banana','noun'),('bread','noun'),('french','noun'),
('toast','noun'),('was','SMART'), ('delicious','onix')

TABLE I: Exemplar Comment From Two Similarity Groups

Participant ID	Comments	Group
215956	No wifi or music at this location right now, this starbucks demonstrates why the change to free wifi for all was a bad idea :-(	A
226566	Taking coffee here now, best coffee in morning, it's run by nyu so no starbucks card purchases or free wifi (you have to be an nyu student.)	A
63432	Awesome cupcakes, blueberry muffins are amazing	B
71504	The chocolate egg cream was underwhelming! The best chocolate egg cream I've ever had!	B

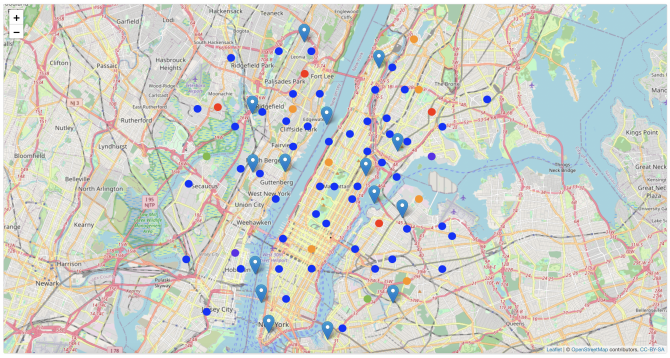

Fig. 3: Pseudo Map of MCS Grouping

Subsequently, we proceeded to count the frequency of each remaining word in the list, eliminating those that occurred less than ten times. Intriguingly, we discovered that words with higher frequencies tended to correspond to commonplace food items, such as "coffee", "chicken", or "cheese". In contrast, terms of greater relevance to our study, including "baby", "parents", and "vegan", were far less prevalent.

Therefore, in our process of arranging words into respective categories, we adopted a bottom-up approach. Starting from the least frequent terms, we progressively worked our way towards the more common ones.

B. Illustration of K-means Clustering Effectiveness in Participant Grouping

To illustrate the effectiveness of K-mean clustering algorithm in participant grouping, four comments are selected from two different similarity group as an example.

As shown in Table I, participants from group A mentioned words such as "wifi," "location," and "free," indicating higher interest in the categories "locations and establishments" and "electronic devices and internet." In comparison, participants from group B mentioned words like "cupcake," "muffins," "chocolate," and "cream," demonstrating higher interest in the category "desserts."

C. Visualizing Participant Groupings in a Simulated Setting

To give a more general picture of grouping, 50 participants are selected from the original data set. Utilizing the previously mentioned algorithm, these participants are segregated into five

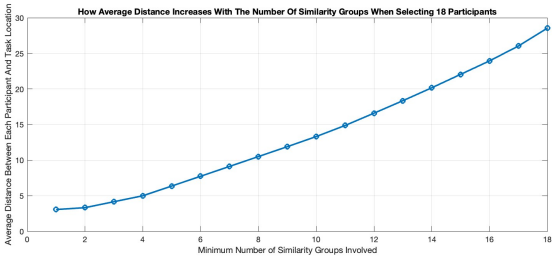

Fig. 4: Diversity-Distance Relationship For Fixed Number of Participants

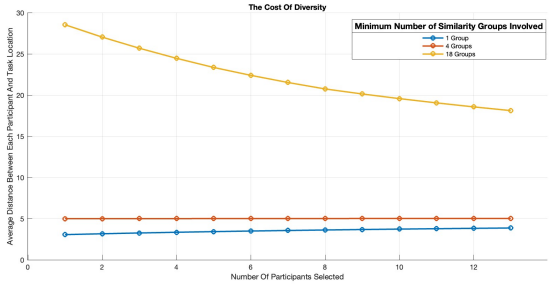

Fig. 5: Influence of Participant Selection Size on Time Efficiency Under Varying Diversity Requirements

distinct similarity groups, respectively, with 35, 4, 7, 2, and 2 participants for each group.

For privacy considerations, actual participant or restaurant locations are not shown; instead, a mock-up map is used for illustration, shown in 3. Fifteen arbitrary markers are distributed across this map, each symbolizing a restaurant. The 30 participants are likewise spread around these restaurants.

It's worth noting that the group distribution is uneven in practice with our data set of 951 participants. We have one large group with 936 participants, a second group with 4 participants, a third group with 7, etc. This disparity in group sizes is expected since minority groups, by definition, have fewer members and are often underrepresented in data collection processes.

D. The Cost of Selection Algorithm

Further analysis is conducted on the efficiency cost of diversity during selection. Efficiency is measured by calculating the selected participants' average distance with respect to task location. Further average distance represents higher time efficiency cost. Diversity is measured by the minimum number of similarity groups selected by the algorithm.

Fig 4 showcases the inverse relationship between diversity requirement and time efficiency. With a constant participant selection of 18, the y-axis represents the average distance of the participants from the task location, while the x-axis denotes the count of distinct similarity groups. The blue, upward-trending line indicates that as the number of distinct similarity groups increases, the average distance linearly increases, and thereby the time efficiency decreases.

Fig 5 portrays the influence of participant selection size on time efficiency under varying diversity requirements. The blue, red, and yellow trend lines correspond to the selection results when the count of similarity groups is set to 1, 4, and 18, respectively. As indicated by the orange trend line, higher diversity requirements see a reduction in the average distance as the number of participants increases. Conversely, under lower diversity requirements, represented by the blue and red trend lines, an increase in participant count results in a minor increase or stable average distance, indicating that larger group sizes do not significantly affect time efficiency.

IV. Conclusion and Future Direction

In conclusion, this research shed light on the potential of MCS systems for accomplishing opinion-based tasks. Initially, we developed a methodology to categorize participants' interests and the clustering process that groups similar participants, enhancing the diversity in participant selection for MCS tasks. Yet, notable disparities in the size of these groups highlight the gap in voice power between majority and minority groups. Using this model, more attention can be given to minority groups, whose needs are often neglected during data collection. Looking forward, our future research aims to extend this model to more intricate scenarios, including managing participant selection for multiple tasks under a single diversity requirement. Given the added complexity of such situations, more comprehensive solutions will need to be developed and refined.

References

[1] R. Hanni, M. M. Patil and P. M. Patil, "Summarization of customer reviews for a product on a website using natural language processing", International Conference on Advances in Computing Communications and Informatics (ICACCI), pp. 2280-2285, 2016.

[2] Blei, D. M., Ng, A. Y., Jordan, M. I. (2003). Latent Dirichlet Allocation. Journal of Machine Learning Research, 3, 993-1022.

[3] X. Hong, "Evaluating the Value of User Comments on the Platform based on K-means Clustering," 2021 International Conference on Computer Information Science and Artificial Intelligence (CISAI), Kunming, China, 2021, doi: 10.1109/CISAI54367.2021.00087.

[4] [1] X. Hu and Q. Zhong, "Analyzing Airbnb User Preferences Using Topic Modeling and Sentiment Analysis," in 2022 5th International Conference on Data Science and Information Technology (DSIT), Shanghai, China, 2022, doi: 10.1109/DSIT55514.2022.9943877.

[5] A. Wang, L. Zhang, L. Guo, M. Ren, P. Li and B. Yan, "A Task Assignment Approach with Maximizing User Type Diversity in Mobile Crowdsensing" . Journal of Combinatorial Optimization, Nov 2020.

[6] M. H. Cheung, F. Hou and J. Huang, "Make a difference: Diversity-driven social mobile crowdsensing," IEEE INFOCOM 2017 - IEEE Conference on Computer Communications, Atlanta, GA, USA, 2017, pp. 1-9, doi: 10.1109/INFOCOM.2017.8057035.

[7] S. Kim, S. Jeon, J. Kim, Y. -H. Park and H. Yu, "Finding Core Topics: Topic Extraction with Clustering on Tweet," 2012 Second International Conference on Cloud and Green Computing, Xiangtan, China, 2012, pp. 777-782, doi: 10.1109/CGC.2012.120.

[8] Yu, J., Xiao, M., Gao, G., Hu, C. "Minimum cost spatial-temporal task allocation in mobile crowdsensing," in: Yang, Q., Yu, W., Challal, Y. (eds.) WASA 2016. LNCS, vol. 9798, pp. 262–271. Springer, Cham (2016).

[9] Duan, Z., Li, W., Cai, Z. " Distributed auctions for task assignment and schedul- ing in mobile crowdsensing systems," in: IEEE 37th International Conference on Distributed Computing Systems (ICDCS), Atlanta, GA, USA, pp. 635–644. IEEE Computer Society (2017)