# Supplementary Document for Location-and-Preference Joint Prediction for Task Assignment in Spatial Crowdsourcing

Xiaohui Wei, *Member, IEEE,* Bingyi Sun, Jiaxu Cui, Meikang Qiu, *Senior Member, IEEE*

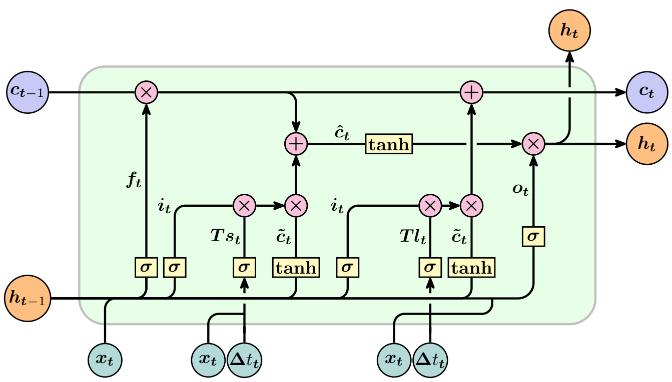



Fig. S1: The architecture of the Temporal Gated Network (TGN)

## I. Proof of Lemma 1

**Lemma 1.** ① $f(\emptyset) = 0$; ② $f(A)$ *is monotonically non-decreasing*; ③ $f(A)$ *is submodular.*

**Proof.** ① $A = \emptyset$ means that no pair of worker and task has been selected. Thus, $f(\emptyset) = 0$.

② Without loss of generality, we define $\beta_g$ as the selected pair of the worker $w^{(g)}$ and task $sp^{(g)}$ at step $g$, i.e., $\beta_g = (w^{(g)}, sp^{(g)})$. Thus, we have $A_g = \{\beta_1, \beta_2, \cdots, \beta_g\}$, and $A_{g+1} = \{\beta_1, \beta_2, \cdots, \beta_{g+1}\}$; that is, $A_g \subseteq A_{g+1}$. Now we derive the difference between $f(A_{g+1})$ and $f(A_{g+1})$ as

$$
\begin{aligned}
&f(A_{g+1}) - f(A_g) \\
&= \sum_{j=1}^{|SP|} 1 - \prod_{i=1}^{|W|} (1 - P(w_i, sp_j)\mathbb{1}((w_i, sp_j) \in A_{g+1})) \\
&- \sum_{j=1}^{|SP|} 1 - \prod_{i=1}^{|W|} (1 - P(w_i, sp_j)\mathbb{1}((w_i, sp_j) \in A_g)) \\
&= \sum_{j=1}^{|SP|} \prod_{i=1}^{|W|} (1 - P(w_i, sp_j)\mathbb{1}((w_i, sp_j) \in A_g)) \\
&\times (1 - (1 - P(w^{(g+1)}, sp^{(g+1)}))) \\
&= \sum_{j=1}^{|SP|} \prod_{i=1}^{|W|} (1 - P(w_i, sp_j)\mathbb{1}(w_i, sp_j) \in A_g) P(w^{(g+1)}, sp^{(g+1)}).
\end{aligned}
$$

Since $P(w_i, sp_j)$ represents the probability of $w_i$ performing $sp_j$, we know that $0 \le P(w_i, sp_j) \le 1$ always holds. We thus have $f(A_{g+1}) - f(A_g) \ge 0$. That is, $f(A)$ is monotonically non-decreasing.

③ Without loss of generality, we define $A_g \subseteq A_q \subseteq ALL$ and $(w', sp') \in (ALL \backslash A_q)$, where $ALL$ means the set including all pairs of worker and task. Similarly, we derive

$$
\begin{aligned}
&[f(A_g \cup \{(w', sp')\}) - f(A_g)] - [f(A_q \cup \{(w', sp')\}) - f(A_q)] \\
&= [\sum_{j=1}^{|SP|} 1 - \prod_{i=1}^{|W|} (1 - P(w_i, sp_j)\mathbb{1}((w_i, sp_j) \in A_g \cup \{(w', sp')\})) \\
&- \sum_{j=1}^{|SP|} 1 - \prod_{i=1}^{|W|} (1 - P(w_i, sp_j)\mathbb{1}((w_i, sp_j) \in A_g] \\
&- [\sum_{j=1}^{|SP|} 1 - \prod_{i=1}^{|W|} (1 - P(w_i, sp_j)\mathbb{1}((w_i, sp_j) \in A_q \cup \{(w', sp')\})) \\
&- \sum_{j=1}^{|SP|} 1 - \prod_{i=1}^{|W|} (1 - P(w_i, sp_j)\mathbb{1}((w_i, sp_j) \in A_q] \\
&= \sum_{j=1}^{|SP|} P(w', sp')[\prod_{i=1}^{|W|} (1 - P(w_i, sp_j)\mathbb{1}((w_i, sp_j) \in A_g)) \\
&- \prod_{i=1}^{|W|} (1 - P(w_i, sp_j)\mathbb{1}((w_i, sp_j) \in A_q))]
\end{aligned}
$$

This work was supported by the National Natural Science Foundation of China (NSFC) (Grants No.U19A2061, No.61772228), National key research and development program of China under Grant No.2017YFC1502306, China Postdoctoral Science Foundation funded project under Grant No.2021M701388. (*Corresponding author: Jiaxu Cui.*)

X. Wei, B. Sun and J. Cui are with the College of Computer Science and Technology, Jilin University, Changchun, Jilin 130012, China, and the Key Laboratory of Symbolic Computation and Knowledge Engineering of the Ministry of Education, Jilin University, Changchun, Jilin 130012, China (e-mail: weixh@jlu.edn.cn; bysun15@mails.jlu.edu.cn; cjx@jlu.edu.cn).

M. Qiu is with Department of Computer Science, Texas A&M University, Commerce, TX 75428. E-mail: qiumeikang@yahoo.com.

Here, we set $A_{q\backslash g} = A_q \backslash A_g$ and continue to derive as

$$[f(A_g \cup \{(w', sp')\}) - f(A_g)] - [f(A_q \cup \{(w', sp')\}) - f(A_q)]$$
$$= \sum_{j=1}^{|SP|} P(w', sp') \prod_{i=1}^{|W|} (1 - P(w_i, sp_j) \mathbb{1}((w_i, sp_j) \in A_g))$$
$$\times (1 - \prod_{i=1}^{|W|} (1 - P(w_i, sp_j)) \mathbb{1}((w_i, sp_j) \in A_{q\backslash g})).$$

Since $P(.,.)$ represents the probability, we can get that $0 \leq P(.,.) \leq 1$. Thus, we can obtain $[f(A_g \cup \{(w', sp')\}) - f(A_g)] - [f(A_q \cup \{(w', sp')\}) - f(A_q)] \geq 0$; that is, $f(A_g \cup \{(w', sp')\}) - f(A_g) \geq f(A_q \cup \{(w', sp')\}) - f(A_q)$, which means $f(A)$ is submodular. □

## II. Effect of Parameters in Location-and-preference Joint Prediction Model

To evaluate the effects of parameters on the prediction, we investigate four parameters, including hidden state size, batch size, $\alpha$, and L2 regularization coefficient. We show the results in Table S1, Table S2, Table S3, and Table S4.

In Table S1, with the changing of hidden state size, the accuracy of location prediction and preference prediction is relatively stable on the three datasets. The optimal value of hidden state size is 520, which is marked in bold. In most cases, the value of 520 brings the highest accuracy on both location and preference prediction. Next, we test the batch size on 16, 32, 64, and 128. The same as hidden state size, in Table S2, with the increase of batch size, the accuracy of location prediction and preference prediction is relatively stable on the three datasets. The best value is 32, and we select it as the default value of batch size. Then, we focus on selecting a suitable $\alpha$, which is used to balance the location and preference prediction in the loss function. In Table S3, both the accuracy of location prediction and preference prediction gently increases with the growth of $\alpha$ until $\alpha = 1$, then, declined slightly. The effect of $\alpha$ on the accuracy of prediction is also not obvious. In Table S4, we change the L2 regularization coefficient from $1e-3$ to $1e-6$. The first and second best results are boldfaced and underlined respectively. We find that the accuracy of location prediction and preference prediction is relatively stable on the three datasets. When L2 regularization coefficient is $1e-5$, we obtain the highest location and preference prediction accuracy on Geolife and Foursquare-TKY datasets, and the second highest location and preference prediction accuracy on Foursquare-NYC dataset.

Therefore, the effects of parameters on the JPM are inapparent, i.e., the JPM is not sensitive to the parameters.

## III. Compared Methods and Experimental Setup

For prediction, since JPM can perform location and preference prediction simultaneously, we compare our model with two types of baselines. The first type only predicts locations, and the second type only infers worker preference category. For task assignment, MAJA is compared with other approaches aiming at prediction-based task assignment.

**Location prediction only**

- LSTM [1]: This is a variant of the RNN model which is effective in handling sequential data.
- ST-RNN [2]: This is an RNN-based model that incorporates spatial and temporal context into spatial-temporal transition matrices.
- DeepMove [3]: This method learns periodicity from the historical trajectory with the attention mechanism and learns sequentiality from the current trajectory using an RNN model.
- STGN [4]: This models uses time and distance gates to capture temporal and spatial intervals.
- STGCN [4]: This is a variant of STGN to reduce computation.
- LSTPM [5]: This approach focuses on the influence of distance on location prediction from both history and current trajectories.
- JPM-w/o-Pref: This method is a variant of our JPM, which does not input preference and only predict location probability distribution.

**Preference inference only**

- TD [6]: This approach is a tensor decomposition approach that fills the missing entries by decomposing the tensor constructed by recent trajectories.
- HCTD [6]: This method is a tensor decomposition approach that fills the missing entries by decomposing the tensor constructed by all trajectories and other auxiliary information.

**Task assignment**

- g-MUS-D [7]: This approach employs a semi-Markov model to predict location probability distribution within the sensing period, and then assign tasks to workers.
- OPP-Greedy [8]: This method predicts the probability of each worker connecting to different task location at least once during the sensing period, and then allocates tasks to workers.
- LTA: This method is a variant of our MAJA, which only predicts locations but do not predicts preference, and then assigns tasks to workers.

Since the baselines do not consider the preference of workers, to compare with our MAJA, we add a preference category statement for every worker. We deem each worker's preference as the category of the last task completed by the worker.

All experiments are performed on the same hardware environment, equipped with an Intel(R) Xeon(R) CPU E5-2680 v4 @ 2.40GHz, 128 GB RAM, and GeForce RTX 3090 GPU.

## IV. Ablation Tests of Location-and-preference Joint Prediction Model

To verify the contribution of each part (network) in JPM to the results, we construct an ablation experiment. Specifically, we implement the following three simplified versions to compare their performance with our model.

- JPM w/o Long: This version removes the long-term prediction module of JPM and engages the TGN network and geo-dilated LSTM in the short-term prediction module.
- JPM w/o TGN: This version removes the TGN of the short-term prediction module in JPM and engages the

TABLE S1: The effect of hidden state size on different datasets.

| | acc_location@5 | | | | | acc_preference@5 | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Hidden state size | 220 | 320 | 420 | **520** | 620 | 220 | 320 | 420 | **520** | 620 |
| Geolife | 0.9121 | 0.9155 | 0.9174 | **0.9208** | 0.9171 | 0.9265 | 0.9249 | 0.9292 | **0.9339** | 0.9223 |
| Foursquare-NYC | 0.6273 | 0.6289 | **0.6540** | 0.6517 | 0.6459 | 0.7633 | 0.7776 | 0.7944 | **0.7998** | 0.7861 |
| Foursquare-TKY | 0.6095 | 0.6194 | 0.6169 | **0.6219** | 0.6191 | 0.8283 | 0.8343 | 0.8362 | **0.8397** | 0.8395 |

TABLE S2: The effect of batch size on different datasets.

| | acc_location@5 | | | | acc_preference@5 | | | |
|---|---|---|---|---|---|---|---|---|
| Batch size | 16 | **32** | 64 | 128 | 16 | **32** | 64 | 128 |
| Geolife | 0.9087 | **0.9208** | 0.9176 | 0.9145 | 0.9260 | **0.9339** | 0.9323 | 0.9218 |
| Foursquare-NYC | 0.6460 | **0.6517** | 0.6394 | 0.6387 | 0.7829 | **0.7998** | 0.7641 | 0.7795 |
| Foursquare-TKY | 0.6143 | **0.6219** | 0.6174 | 0.6199 | 0.8379 | **0.8397** | 0.8378 | 0.8323 |

TABLE S3: The effect of $\alpha$ on different datasets.

| | acc_location@5 | | | | acc_preference@5 | | | |
|---|---|---|---|---|---|---|---|---|
| $\alpha$ | 0.4 | 0.7 | **1.0** | 1.3 | 0.4 | 0.7 | **1.0** | 1.3 |
| Geolife | 0.9156 | 0.9166 | **0.9208** | 0.9140 | 0.9229 | 0.9224 | **0.9339** | 0.9292 |
| Foursquare-NYC | 0.6264 | 0.6497 | **0.6517** | 0.6177 | 0.7663 | 0.7997 | **0.7998** | 0.7811 |
| Foursquare-TKY | 0.6189 | 0.6194 | **0.6219** | 0.6191 | 0.8334 | 0.8383 | **0.8397** | 0.8304 |

TABLE S4: The effect of L2 regularization coefficient on different datasets.

| | acc_location@5 | | | | acc_preference@5 | | | |
|---|---|---|---|---|---|---|---|---|
| L2 | 1e-3 | 1e-4 | 1e-5 | 1e-6 | 1e-3 | 1e-4 | 1e-5 | 1e-6 |
| Geolife | 0.9123 | 0.9113 | **0.9208** | 0.9203 | 0.9328 | 0.9275 | **0.9339** | 0.9302 |
| Foursquare-NYC | **0.6528** | 0.6475 | <u>0.6517</u> | 0.6373 | 0.7954 | **0.8077** | <u>0.7998</u> | 0.7661 |
| Foursquare-TKY | 0.4798 | 0.6082 | **0.6219** | 0.6170 | 0.8293 | 0.8368 | **0.8397** | 0.8377 |

TABLE S5: Performance of different JPM variants on Foursquare-NYC dataset.

| | acc_location@k | | | acc_preference@k | | |
|---|---|---|---|---|---|---|
| | k=1 | k=5 | k=10 | k=1 | k=5 | k=10 |
| LSTPM | 0.3372 | 0.5924 | 0.6953 | - | - | - |
| HCTD | - | - | - | 0.3140 | 0.7348 | 0.8704 |
| JPM w/o Long | 0.3178 | 0.5325 | 0.6079 | 0.2929 | 0.7444 | 0.9796 |
| JPM w/o TGN | 0.3223 | 0.5867 | 0.6794 | 0.2987 | 0.7553 | 0.9801 |
| JPM w/o Geo-dilated LSTM | 0.3503 | 0.6382 | 0.7274 | 0.3059 | 0.7803 | 0.9833 |
| JPM | **0.3838** | **0.6517** | **0.7448** | **0.3263** | **0.7998** | **0.9919** |

long-term prediction module and geo-dilated LSTM in short-term prediction.

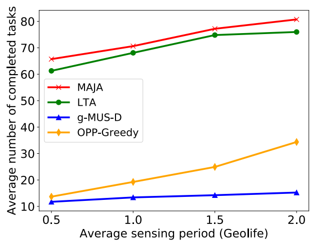
- JPM w/o geo-dilated LSTM: This version removes the geo-dilated LSTM of the short-term prediction module in JPM and engages the long-term prediction module and TGN in short-term prediction.

As network sensitivity experiments have obtained similar results on all datasets, we report the results on the Foursquare-NYC dataset in Table S5. For ease of comparison, we also put the results of the best baseline methods in the table. Through the ablation tests, we can observe that:

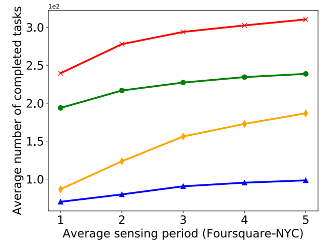
- Although JPM w/o Long is less competitive than other degraded versions of JPM, it can still get better prediction performance than many baselines, such as the LSTM, ST-RNN, STGN, and STGCN for location prediction, and TD for preference prediction (please see the Table IV and Table V in the paper). Hence the effectiveness of JPM w/o Long comes from our short-term prediction module, which consists of TGN and geo-dilated LSTM.
- The accuracy of the JPM w/o TGN is better than JPM w/o Long, mainly because the JPM w/o TGN captures the periodicity and distance influence of worker mobility by the long-term prediction module and the spatial relations by the geo-dilated LSTM in the short-term prediction module.
- JPM w/o geo-dilated LSTM outperforms other degraded versions of JPM, mainly because it can capture the long-term dependence of user mobility. Moreover, compared with the JPM w/o TGN, the added TGN part in the short-term prediction module could integrate the influence of the time interval of the current trajectory on joint prediction.
- The complete model JPM achieves the best performance, showing that the three parts have positive impacts on the location-and-preference joint prediction.

## References

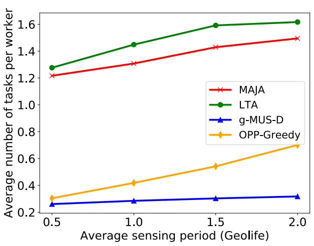
[1] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.



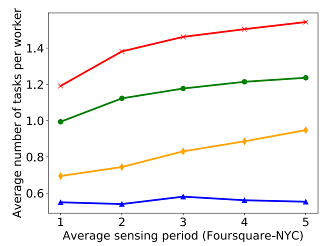

(a) Avg # completed tasks, Average task sensing period (Geolife)




(b) Avg # completed tasks, Average task sensing period (Foursquare-NYC)




(c) Avg # completed tasks per worker, Average task sensing period (Geolife)




(d) Avg # completed tasks per worker, Average task sensing period (Foursquare-NYC)

Fig. S2: Performance of task assignment: Effect of task sensing period. The two metrics increase as the sensing period becomes longer in the three datasets. The reason is that workers have more chances to be assigned to the tasks for the more relaxed sensing period.

[2] Q. Liu, S. Wu, L. Wang, and T. Tan, "Predicting the Next Location: A Recurrent Model with Spatial and Temporal Contexts," in *30th AAAI Conference on Artificial Intelligence*, 2016, pp. 194–200.

[3] J. Feng, Y. Li, C. Zhang, F. Sun, F. Meng, A. Guo, and D. Jin, "DeepMove: Predicting Human Mobility with Attentional Recurrent Networks," in *2018 World Wide Web Conference*, 2018, pp. 1459–1468.

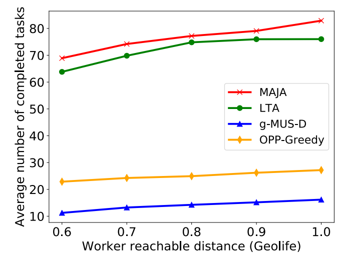
[4] P. Zhao, H. Zhu, Y. Liu, J. Xu, Z. Li, F. Zhuang, V. S. Sheng, and X. Zhou, "Where to Go Next: A Spatio-Temporal Gated Network for Next POI Recommendation," in *33rd AAAI Conference on Artificial Intelligence*, 2019, pp. 5877–5884.

[5] K. Sun, T. Qian, T. Chen, Y. Liang, Q. V. H. Nguyen, and H. Yin, "Where to Go Next: Modeling Long- and Short-Term User Preferences for Point-of-Interest Recommendation," in *34th AAAI Conference on Artificial Intelligence*, 2020, pp. 214–221.

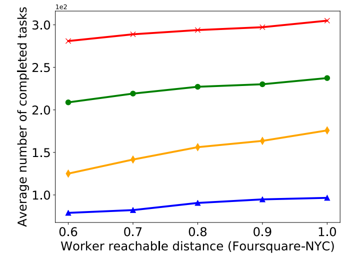
[6] Y. Zhao, J. Xia, G. Liu, H. Su, and K. Zheng, "Preference-Aware Task Assignment in Spatial Crowdsourcing," in *33rd AAAI Conference on Artificial Intelligence*, 2019, pp. 2629–2636.

[7] Y. Yang, W. Liu, E. Wang, and J. Wu, "A Prediction-Based User Selection Framework for Heterogeneous Mobile CrowdSensing," *IEEE Transactions on Mobile Computing*, vol. 18, no. 11, pp. 2460–2473, 2019.

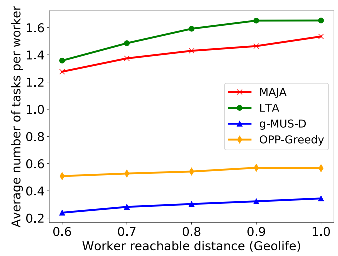
[8] J. Wang, F. Wang, Y. Wang, L. Wang, Z. Qiu, D. Zhang, B. Guo, and Q. Lv, "HyTasker: Hybrid Task Allocation in Mobile Crowd Sensing," *IEEE Transactions on Mobile Computing*, vol. 19, no. 3, pp. 598–611, 2020.



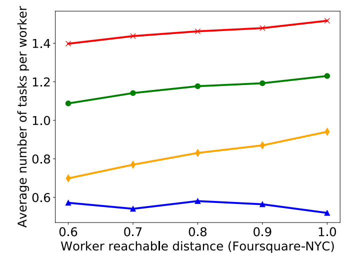

(a) Avg # completed tasks, Average worker reachable distance (Geolife)




(b) Avg # completed tasks, Average worker reachable distance (Foursquare-NYC)

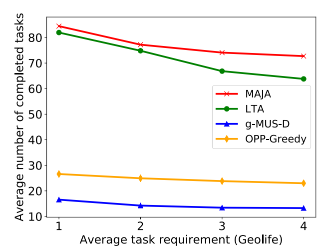



(c) Avg # completed tasks per worker, Average worker reachable distance (Geolife)

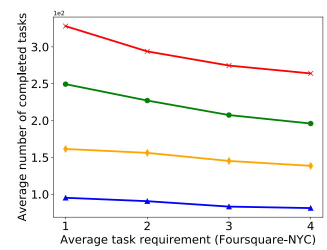



(d) Avg # completed tasks per worker, Average worker reachable distance (Foursquare-NYC)

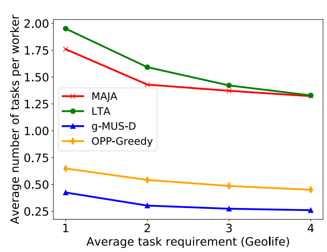
Fig. S3: Performance of task assignment: Effect of worker reachable distance. The two metrics have a growing tendency with the increasing worker reachable distance. The reason is similar to the effect of tasks sensing period, i.e., the greater the reachable distance of the worker, the more opportunities to assign tasks.



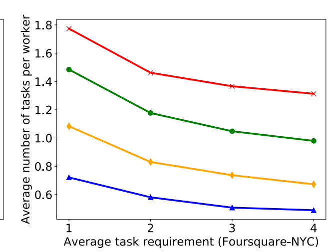

(a) Avg # completed tasks, Average task requirement (Geolife)




(b) Avg # completed tasks, Average task requirement (Foursquare-NYC)

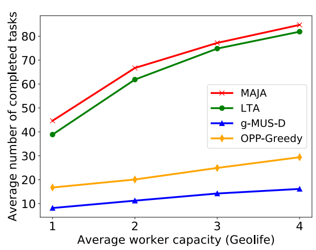



(c) Avg # completed tasks per worker, Average task requirement (Geolife)

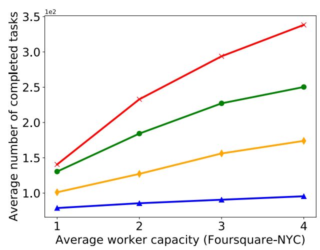



(d) Avg # completed tasks per worker, Average task requirement (Foursquare-NYC)

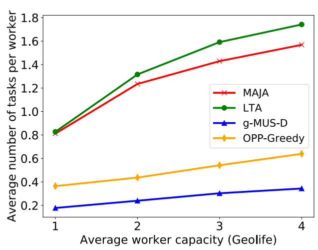
Fig. S4: Performance of task assignment: Effect of task requirement. The two metrics significantly reduce when the number of task requirements increases, especially on the Foursquare-NYC and Foursquare-TKY datasets. Because the number of workers who satisfy the spatial-temporal and preference category constraints is limited, a task requires more workers to perform it.



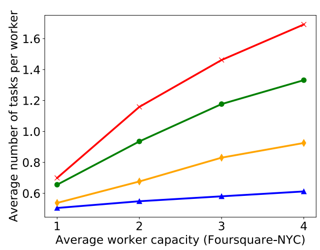

(a) Avg # completed tasks, Average worker capacity (Geolife)




(b) Avg # completed tasks, Average worker capacity (Foursquare-NYC)




(c) Avg # completed tasks per worker, Average worker capacity (Geolife)




(d) Avg # completed tasks per worker, Average worker capacity (Foursquare-NYC)

Fig. S5: Performance of task assignment: Effect of worker capacity. The two metrics dramatically rise with the number of worker capacity increasing. Since the more capable the worker is, the more chance that the SC server has to assign the workers more tasks.