Project Report: Crowdsourcing to Indoor Localization based on Wireless Fingerprint

Zhicheng Gu
Student ID: 5110309240
Department of Electronic Engineering
Shanghai Jiao Tong University
E-mail: zchenggu@gmail.com

June 12, 2014

Abstract

Indoor localization has been studied with different approaches. Indoor localization based on fingerprint works properly in places with obstacles. However, when it comes to large-scale implements, none of traditional ways can work efficiently, say they need either huge amounts of database or high technique devices for measuring. Crowdsourcing is suitable for solving large scale problems by allocating small pieces of tasks to workers, and it is just appropriate for this scenario. On the other hand, we must suffer the unavoidable unreliability of the crowdsourcing system. In this paper, we try to design a reliable spot indoor localization model based on wireless fingerprint. In this system, the database collecting process is accomplished by crowdsourcing approach and the difference of the kinds of measuring information has no affect, which means we can use RSSI, CSI or whatever measuring technique we want. Through the collecting and matching algorithm provided, we give out the error probability and other properties of this system and prove the results. It shows that this model performance well for the indoor localization, epically in the accuracy and efficiency aspects.

1 Introduction

1.1 Indoor Localization

One method to determine the location of a device is through manual configuration, which is often infeasible for large-scale deployments or mobile systems. As a popular system, Global Positioning System (GPS) is not suitable for indoor or underground environments and suffers from high hardware cost. Local Positioning Systems (LPS) rely on highdensity base stations being deployed, an expensive burden for most resource-constrained wireless ad-hoc networks. However, nowadays almost everyone has got a high technique equipment for wireless sensing, that is, our smart phone. By using the people as a crowd and this resource constrained is easy to solve.

Almost all existing localization algorithms consist of two stages: 1) measuring geographic information from the ground truth of network deployment; 2) computing node locations according to the measured data. In our system, we use the wifi fingerprint as the measurement, and the crowdsourcing as the measuring technique. And once we got the information of the area, computing locations is simply by matching.

There are mainly two errors of the indoor localization. The extrinsic error is attributed to the physical effects on the measurement channel, such as the presence of obstacles, multipath and shadowing effects, and the variability of the signal propagation speed due to environmental dynamics. On the other hand, the intrinsic error is caused by limitations of hardware and software. While the extrinsic one is more unpredictable and challenging during real deployments, the intrinsic one causes many complications when using multi-hop measurements to estimate node locations. Results from field experiments demonstrated that even relatively small ranging errors can significantly amplify the error of location estimates [1]. Thus, dealing with such errors is an essential issue for high-accuracy localization algorithms. Our system tend to estimate the influence of the intrinsic error and the obstacles and shadowing effects in extrinsic one. But since our system need the environment to be statics, it can not deal with the signal noise and environment dynamics of the extrinsic error.

We are interest in designing a spot localization model to identify the device in 1m x 1m square. Localizing the device outside the box will be useless, irrespective of whether the estimated location is close or far away from the box. This model can be spread used. For instance, the advertising industry is beginning to expect location accuracies at the granularity of an aisle in a grocery shop. Museums are expecting user locations at the granularity of paintings so users can automatically receive information about the paintings that they walk by [2]. In addition to such high accuracy demands, these applications are inherently intolerant to small errors. If a localization scheme incorrectly places a user in the adjacent aisle in the grocery store, or downloads information about the adjacent painting, the purpose of localization is entirely defeated. Localization schemes will need to meet strict standards, with incurring little costs of installation and maintenance.

1.2 crowdsourcing

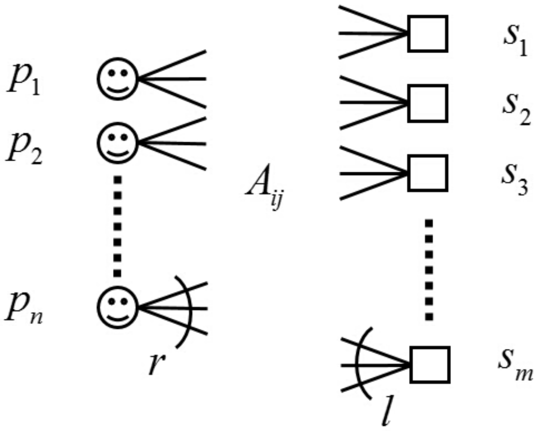

Figure 1: Task allocation in crowdsourcing system

Crowdsourcing is considered as an effective way for addressing problems by utilizing human computation power in domains like classification, scoring and etc. Crowdsourcing is an effective way for resolving tasks such as classification, identification, scoring by accumulating answers from human worker population. One typical crowdsourcing system is Amazon Mechanical Turk where people can upload tasks on the system platform and obtain answers from workers later. In this system, workers can finish tasks to get a small payment while the task providers could obtain a large number of answers submitted by workers to estimate question answer at a low cost. We can take the measuring process as the requirement and take the answers as feedback. There have already been many typical crowdsourcing systems worldwide[3] such as Amazon Mechanical Turk[4], Yelp[5] and Yahoo! Answers[6],

Due to the complex composition of workers, obtained answers will be a mix of noise which enhance the difficulty of estimating true answers. To get reliable answers from workers, we shall utilize redundancy, which indicates that we shall allocate each task to multiple workers. When obtaining multiple answers for each task, we also have to utilize efficient inference algorithm to filter noise answers from low-qualified workers and estimate answers more accurately.

Expectation maximization (EM) is another algorithm for obtaining the reliable answer from the crowd, moreover, it can also derive the reliability of the workers. It has been proved to be effective in the classification system adopting labeling model[7][8], where we want to choose one correct label from multiple labels for each tasks. However, the result of the EM algorithm is highly sensitive to the initial that is usually randomly guessed; therefore, it is difficult to guarantee the accuracy of the derived answer.

Now many researchers are focusing on estimating task answers by estimating the reliability of workers and many algorithms have been proved to be effective in the labeling* tasks. Karger et al. propose a task allocation method based on random regular bipartite graph and use low rank approximation to generate the estimated answer [9]. Moreover, they also develop an efficient crowdsourcing system in minimizing the task assignment redundancy while achieving a desirable reliability.

However, these research normally focused on the inference problem, but ignored the question of how to assign workers to tasks by assuming that the learner has no control over the assignment. There have been researches about the allocation method in labeling problems[10][9]. The authors use low-rank approximation of weighted adjacency matrix for a random regular bipartite graph and design crowdsourcing systems that are efficient in the sense of achieving reliability at the minimal cost of redundancy. When tasks are homogeneous, eigenvalue decomposition could also be used to estimate each workers quality[11]. Their bounds depend on a quantity which they refer to as the populations average competence. The labeling problems could also be generalized to heterogeneous classification tasks[12]. By employing online primal-dual techniques, the authors derive a provably near-optimal adaptive assignment algorithm, and show that assigning workers adaptively to tasks can lead to more accurate predictions at a lower cost when the available workers are diverse. The schemes proposed for labeling scheme as mentioned above, however, are unsuitable to be applied to scoring problems, because the answers in the latter could be corcorrelative with each other.

While the work mentioned above are focusing on how to infer the correct answer, efforts have been made to investigate how to appropriately assign tasks to workers.

Karger *et al.* propose a task allocation method based on random regular bipartite graph and use low rank approximation to generate the estimated answer[9][10]. Moreover, they also develop an efficient crowdsourcing system in minimizing the task assignment redundancy while achieving a desirable reliability. Ghosh *et al.* apply eigenvalue decomposition method to homogeneous-task model [11] and Ho *et al.* generalize the model to be heterogeneous[12].

In this paper, we propose a indoor localization system with crowdsourcing approach. And we design an algorithm for crowdsourcing tasks, which infers task answers based on not only the worker quality but also the answers' correlations with each other. Specifically, the main contributions are as follows.

- We build a indoor localization system by using crowdsourcing method and propose the allocation method, database calculating method and matching algorithm. In the system, the different measuring methods have no affect.
- We propose and iterative algorithm which assigns diverse weights to workers' submitted answers based on their quality.

2 System Model

2.1 Overview

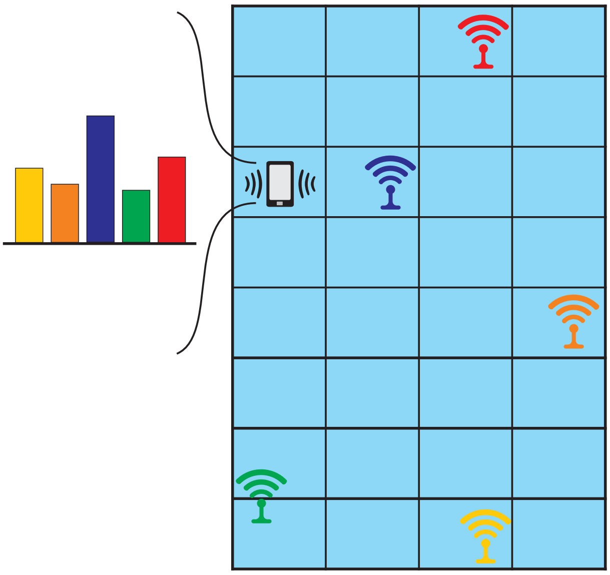

Figure 2: Measurement of indoor localization with wireless approach

We have a total indoor area of $l_1\ (m) \times l_2\ (m)$. The whole place could be divided into many $l_a\ (m) \times l_b\ (m)$ cells. So there are totally $M = \frac{l_1}{l_a} \times \frac{l_2}{l_b}$ cells; Many wireless transmitting routers* is placed in this indoor area. Suppose there are totally N wireless routers denoted by $\{R_j; j = 1, 2 \cdots, N\}$. So there is totally N signals as well. The objectives of indoor localization is tell the user which cell he is in. Since we use fingerprinting mapping method, we need to design a calibration method. We use crowdsourcing to collect signal features of all possible locations in the area to build a fingerprint database. Features at each location should differ from all the others to avoid ambiguity, and this is one critical factor we studied in our system. The whole system could be divided into two steps, which is database collecting step* and service step*.

2.2 database collecting step

For the database collecting step, each worker submit their answers

$$\vec{r_i} = (r_{i1}, r_{i2} \cdots, r_{iN}) \quad r_{ij} \in (0, r_j^{max}), \tag{2.1}$$

where r_i is the RSS(Received Signal Strength) of signal send by wireless router R_j and has its own maximum value s_i^{max}. We want to build a database of all the cells

$$D = \{\vec{d_1}, \vec{d_2} \cdots, \vec{d_M}\}, \tag{2.2}$$

where

$$\vec{d_i} = (d_{i1}, d_{i2} \cdots, d_{iN}) \quad d_{ij} \in (0, s_j^{max}). \tag{2.3}$$

After all of the information is collected, we build the database of 2.3 by doing

$$\vec{d_i} = \frac{1}{n} \sum \vec{r_i} \quad i \in \{1, 2 \cdots, n\} \tag{2.4}$$

There is another way that we can get the database is doing an iterative algorithm.

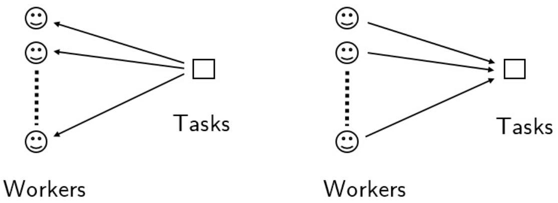

Figure 3: Task allocation in crowdsourcing system

For tasks with an continuous answer interval in $[a, b]$, we define a correlation function to describe the proximity (correlation) of two scores are with each other in as the equation (2.5) shows. For the simplicity of mathematical analysis, we use the square form $(\cdot)^2$ instead of the absolute value sign form $|\cdot|$ to profile the correlation.

$$R(x, y) \triangleq \begin{cases} 1 - c(x-y)^2 & if\ |x-y| < \frac{1}{\sqrt{c}}\ , \\ 0 \quad otherwise\ , \end{cases} \tag{2.5}$$

where c is the constant.

And in to clarify the analysis in the following sections, we also give definitions of some derivative parameters. We define $\tilde{q}_{j|y}$ as

$$\tilde{q}_{j|y} \triangleq \int_a^b f_j(x|y) R(x, y)\, dx, \tag{2.6}$$

The *biased correlation function* $\tilde{R}(x, y)$ is defined as

$$\tilde{R}(x, y) \triangleq R(x, y) - \frac{\int_a^b R(x, y)\, dy}{b - a}, \tag{2.7}$$

which satisfies $\int_a^b \tilde{R}(x, y)\, dx = 0$. Then the quality of worker j when the true answer of task is y is defined as

$$q_{j|y} \triangleq \int_a^b f_j(x|y) \tilde{R}(x, y)\, dx\ . \tag{2.8}$$

When quality of workers is divergent, MCE algorithm is error-prone since it gives identical weight to each worker's answer. In order to estimate task answers accurately, we can use workers' quality as their weight while to estimate workers' quality answers precisely, we shall use true answers of tasks. However, neither workers' quality nor task answer is prior known. To resolve such a challenge, we propose the WMCE algorithm which can estimate both the worker quality and task answers by iterations at the same time. At each iteration, the algorithm will update new estimated answer vector $\hat{t}^{(h)} = \left[\hat{t}_i^{(h)}\right]_{i\in[m]}$ and new estimated quality vector $\hat{q}^{(h)} = \left[\hat{q}_j^{(h)}\right]_{j\in[n]}$, where the superscript (h) denotes the value of variables in the h_{th} iteration.

Algorithm 1 Database Algorithm

```
Input:
    The answer matrix $A = [A_{ij}]^{m\times n}$
    The maximum iteration number $h_{max}$
    The biased correlation function $\tilde{R}(x,y)$
Output:
    The estimated answer vector $\hat{t}$
Initialize the estimated qualtiy vector $\hat{q}^{(1)} = \mathbf{1}^n$, where $\mathbf{1}^n$ denotes the all-ones vector in n-dimensional.
for $h = 1, 2, ..., h_{max}$ do
    for $i = 1, 2, ..., m$ do
        $\hat{t}_i^{(h)} = \arg \max\limits_{x\in[a,b]} \sum\limits_{j\in\partial i} \hat{q}_j^{(h)} \tilde{R}\left(x, A_{ij}\right)$;
    end for
    for $j = 1, 2, ..., n$ do
        $\hat{q}_j^{(h+1)} = \frac{1}{r} \sum\limits_{i\in\partial j} \tilde{R}(\hat{t}_i^{(h)}, A_{ij})$;
    end for
end for
The estimated answer vector $\hat{t} = \hat{t}^{(h_{max})}$
return $\hat{t}$;
```

2.3 service step

Then at service step, users $\{u_k | k \in 1, 2 \cdots, K\}$ submit there receive signal strength

$$\vec{r}_k = (r_{k1}, r_{k2} \cdots, r_{kN}) \quad r_{kj} \in (0, r_j^{max}). \tag{2.9}$$

We do the judgement by Algorithm 2

Intuitively, we can use the hamming distance to make a judgement, which can be expressed as $J(u_k) = \arg \left\|\vec{r}_k - \vec{d}_i\right\| = \arg \sum\limits_{j=1}^{N} |r_{kj} - d_{ij}|$. However, it is not suitable for our system according to the following reasons.

- The wireless strength can be collected through different method. Power, time and angle features are among conventional physical measurements. Our system allow the combination of different measurement. The information collected by different methods have different units and can not be calculated together
- Even we climate the difference in units and uniformizate before calculate different information together, different information tend to have different weight. The information have different density distribution as well as the maximum and minimum value, so it is hard to estimate the real weight of different information.
- Our algorithm have a better performance to climate the shape noisy influence* for the users. That is *** in

Metric:

$$P_e = \mathbb{P}(J(u_k) \neq i | u_k = i) \tag{2.10}$$

Algorithm 2 Matching Algorithm

```
Input:
    The wireless fingerprint database $D$
    The receive signal strength of user $\{\vec{r}_k\}$
Output:
    The localization results $\{J(u_k)\}$
for $k = 1, 2..., K$ do
  $position = 1$;
  for $i = 2, 3..., M$ do
    $judge = 0$;
    for $j = 1, 2..., N$ do
      if $|r_{kj} - d_{position,j}| > |r_{kj} - d_{ij}|$ then
        $judge++$;
      else
        $judge--$;
      end if
    end for
    if $judge > 0$ then
      $position = i$;
    end if
  end for
  $J(u_k) = position$;
end for
return $\{J(u_k)\}$;
```

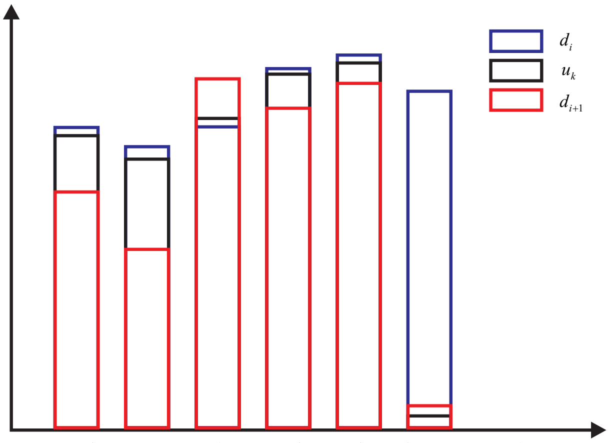

Figure 4: A sharp noisy* situation example

3 Analysis

3.1 Lower Bound

To get the lower bound of the error probability, we assume every worker can give the

Theorem 1. *Under the ideal situation, the probability of error is*

$$\lim_{M\to\infty} P_e \geq \sum_{k=N/2+1}^{N} C_N^k (P_e^1)^k (1-P_e^1)^{N-k} \tag{3.1}$$

Proof. To get the lower bound of the probability of error, we should use the following Lemma 1 □

Lemma 1. *Consider one wireless signal s_j transmitting in the direction which is vertical of the cell border. When the total cell in this line is M_j and maximum and minimum signal strength is r_j^{max} and r_j^{min}, respectively. Then the average probability of error between adjacent cells can be express as*

$$P_e^1 \geq \Phi(-\sqrt{3n}\frac{(r_j^{max}-r_j^{min})^2}{2\sqrt{2}M}) \tag{3.2}$$

where the .

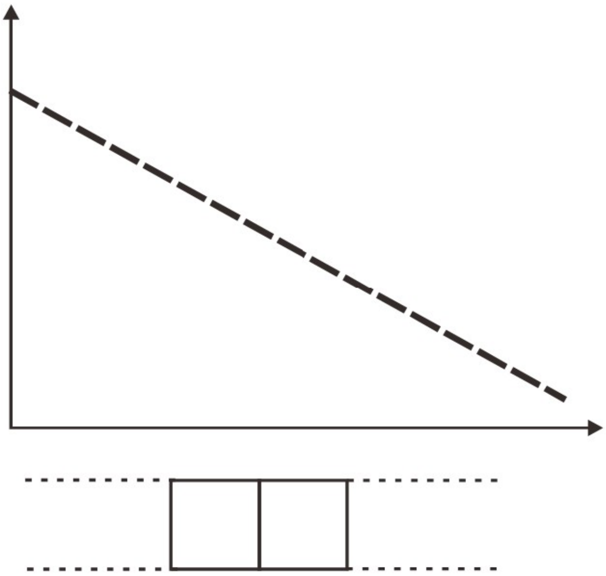

Figure 5: The situation between two adjacent cells

Proof. The probability of error is

$$P_e^l = \mathbb{P}(J(u_k)=i+1|u_k=i)+\mathbb{P}(J(u_k)=i|u_k=i+1) \tag{3.3}$$

□

Where the right part can be express as

$$2\mathbb{P}(J(u_k)=i+1|u_k=i). \tag{3.4}$$

By using the geometric relationships, we can get

$$\mathbb{P}(|r_{kj}-d_{ij}|>|r_{kj}-d_{(i+1)j}|) \tag{3.5}$$

and

$$\mathbb{P}(r_{kj}<\frac{d_{ij}+d_{(i+1)j}}{2}) \tag{3.6}$$

Since we have the definition that

$$\vec{d_i}=\frac{1}{n}\sum\vec{r_i} \quad i\in\{1,2\cdots,n\}. \tag{3.7}$$

and

$$d_{ij} = \frac{1}{n} \sum r_{ij} \quad i \in \{1, 2 \cdots, n\}, \tag{3.8}$$

which can be expressed as

$$\boldsymbol{d}_i = \mathbb{E}[\boldsymbol{r}_{ij}] \quad i \in \{1, 2 \cdots, n\} \tag{3.9}$$

where

$$\begin{aligned} \mathbf{r}_{ij} \sim U[r_j^{max} - i\tfrac{r_j^{max} - r_j^{min}}{M}, r_j^{max} - (i-1)\tfrac{r_j^{max} - r_j^{min}}{M}] \\ \triangleq U[a, b]. \end{aligned} \tag{3.10}$$

Using Lindeberg-Levy Theorem we can get that

$$\begin{aligned} &\sqrt{n}(\mathbf{d}_{ij} - (r_j^{max} - (i - \tfrac{1}{2})\tfrac{r_j^{max} - r_j^{min}}{M})) \\ &\xrightarrow{d} N(0, \tfrac{1}{12}(\tfrac{r_j^{max} - r_j^{min}}{M})^2) \end{aligned} \tag{3.11}$$

$$\begin{aligned} &\sqrt{n}(\mathbf{d}_{(i+1)j} - (r_j^{max} - (i - \tfrac{3}{2})\tfrac{r_j^{max} - r_j^{min}}{M})) \\ &\xrightarrow{d} N(0, \tfrac{1}{12}(\tfrac{r_j^{max} - r_j^{min}}{M})^2) \end{aligned} \tag{3.12}$$

And by using the definition of the normal distribution, we can get that

$$P_e^1 \geq \Phi(-\sqrt{3n}\frac{(r_j^{max} - r_j^{min})^2}{2\sqrt{2}M}) \tag{3.13}$$

Theorem 2. *For $\forall y \in [a, b]$ and $\forall j \in [n]$, if $f_j(x|y) = 0$ when $|x - y| > \frac{1}{\sqrt{c}}$, the mathematical expectation of correlation error E_c can be expressed as*[1]

$$\mathbb{E}[\mathbf{E}_c] = \frac{1}{m} \sum_{i \in [m]} \mathbb{E}[\mathbf{E}_{c,i}], \tag{3.14}$$

where $\mathbb{E}[\mathbf{E}_{c,i}]$ is the mathematical expectation of correlation error for i_{th} task. $\mathbb{E}[E_c]$ can be expressed as

$$\mathbb{E}[E_c] = X + Y\ , \tag{3.15}$$

where

$$\left\{ \begin{array}{l} X = c\Big(\sum\limits_{j \in \partial i} \tilde{w}_{l,j} b_{j|t_i}\Big)^2 \\ Y = c \sum\limits_{j \in \partial i} \tilde{w}_{l,j}^2 v_{j|t_i} \end{array} \right. . \tag{3.16}$$

The normalized worker weight satisfies $\sum\limits_{j \in \partial i} \tilde{w}_{l,j} = 1$, where $\tilde{w}_{l,j} = \frac{1}{l}$ for $\forall j \in \partial i$ in MCE algorithm while $\tilde{w}_{l,j} = \frac{q_j}{\sum\limits_{j \in \partial i} q_j}$ for $\forall j \in \partial i$ in WMCE algorithm.

3.2 Upper Bound

This part need to be done in the future.

4 Simulation

Use a 100*100 indoor region, 10000 workers, There are noise influence on workers answers

$$r'_{mj} = r_{mj} + \alpha U[-1, 1] \tag{4.1}$$

The result is shown as follows. Each blue points means that a work is allocated to get the signal strength of this point. And when the database is collected, we use 1000 users to test the performance of the system.

By using users to estimate the performance, we can get the probability of error for all of the users. The result is shown as follows figure.

[1] $\mathbb{E}[\cdot]$ is a notation of mathematical expectation. Throughout this paper, we use boldface characters to denote random variables.

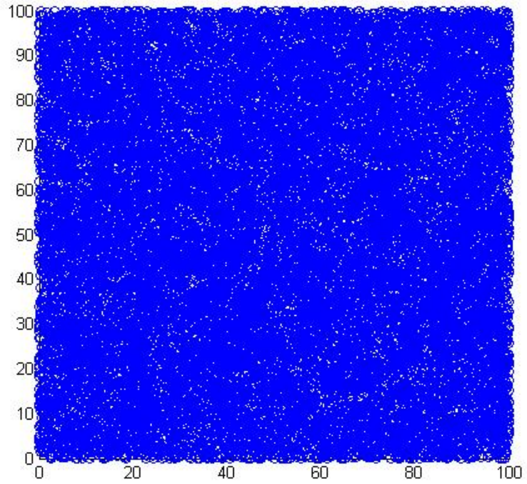

Figure 6: Task allocation

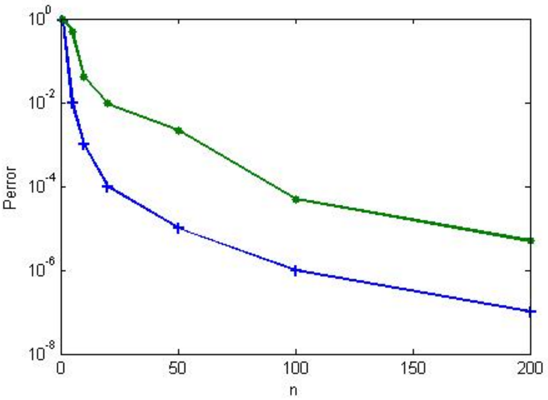

Figure 7: Perror

5 Future Work

This is only a junior work for using our crowdsourcing model into the indoor localization field. In the summer vacation we want to build a real system to test the algorithm.

References

[1] D. Moore, J. Leonard, D. Rus, and S. Teller, “Robust network localization with noisy range measurements,” in *Proceedings of the 2nd international conference on Embedded networked sensor systems*. ACM, 2004, pp. 50–61.

[2] E. Bruns, B. Brombach, T. Zeidler, and O. Bimber, “Enabling mobile phones to support large-scale museum guidance,” *IEEE multimedia*, vol. 14, no. 2, pp. 16–25, 2007.

[3] A. Doan, R. Ramakrishnan, and A. Y. Halevy, “Crowdsourcing systems on the world-wide web,” *Communications of the ACM*, vol. 54, no. 4, pp. 86–96, 2011.

[4] A. Kittur, E. H. Chi, and B. Suh, “Crowdsourcing user studies with mechanical turk,” in *Proc. 26th Annual SIGCHI conf. on human factors in computing systems*, 2008.

[5] J. Aasman, “Social network analysis and geotemporal reasoning in a web 3.0 world,” *Int’l Conf. on Computational Science and Engineering*, vol. 4, pp. 546–548, 2009.

[6] L. A. Adamic, J. Zhang, E. Bakshy, and M. S. Ackerman, “Knowledge sharing and yahoo answers: everyone knows something,” in *Proc. 17th int’l conf. on World Wide Web*, 2008.

[7] J. Whitehill, P. Ruvolo, T. Wu, J. Bergsma, and J. R. Movellan, “Whose vote should count more: Optimal integration of labels from labelers of unknown expertise.” in *NIPS*, vol. 22, 2009, pp. 2035–2043.

[8] P. Welinder, S. Branson, S. Belongie, and P. Perona, “The multidimensional wisdom of crowds.” in *NIPS*, vol. 10, 2010, pp. 2424–2432.

[9] D. R. Karger, S. Oh, and D. Shah, “Efficient crowdsourcing for multi-class labeling,” in *Proceedings of the ACM SIGMETRICS/international conference on Measurement and modeling of computer systems*. ACM, 2013, pp. 81–92.

[10] ——, “Budget-optimal crowdsourcing using low-rank matrix approximations,” in *Proc. of the allerton Conf. on Commun., Control and Computing*, 2011.

[11] A. Ghosh, S. Kale, and P. McAfee, “Who moderates the moderators?: crowdsourcing abuse detection in user-generated content,” in *Proc. ACM EC*, 2011, pp. 167–176.

[12] C.-J. Ho, S. Jabbari, and J. W. Vaughan, “Adaptive task assignment for crowdsourced classification,” in *Proc. ICML*, 2013, pp. 534–542.