The AAAI 2017 Spring Symposium on
Artificial Intelligence for the Social Good
Technical Report SS-17-01

Smart City Planning with Constrained Crowd Judgment Analysis

Sujoy Chatterjee,[1] Anirban Mukhopadhyay,[1] and Malay Bhattacharyya[2]
[1]Department of CSE, University of Kalyani, Nadia – 741235, India
E-mail: {sujoy, anirban}@klyuniv.ac.in
[2]Department of IT, IIEST, Shibpur, Howrah – 711103, India
E-mail: malaybhattacharyya@it.iiests.ac.in

Abstract

Collecting opinions from multiple crowd workers has been proved to be very effective to reach into a prompt and robust decision. There are many real-life applications (like smart city planning and urban development) where public (skilled or unskilled) opinions play a better role in comparison with single expert opinion. A spectrum of algorithms has already been proposed for obtaining robust consensus judgment from multiple crowd opinions. In most of the problems, for a particular question we receive a single opinion with multiple options. But in this article, we have proposed a new research problem termed as *constrained judgment analysis* problem that consists of the questions having multiple opinions. Moreover, some constraints among the options are required to be satisfied while giving opinions. In this constrained judgment analysis, the traditional way of decision making, like majority voting, weighted voting or probablistic model, cannot be applied directly due to the purpose of constraint satisfaction. In this work, we have also proposed a probabilistic method for obtaining final solution for constrained judgment analysis problem and demonstrated its efficacy over a synthetic dataset, thereby proving its utility in resource allocation for smart city planning.

Introduction

Over the last decade, it has been seen that effective utilization of human power (Howe 2006) can easily resolve various real-life problems in very efficient way. The annotation of a large-scale dataset is very tedious task for obtaining judgment in a very limited time. Assigning proper annotator for a particular task and getting feedback from them promptly, is a very tiresome job. Therefore, the concept of outsourcing a huge task to crowd is getting so popular to accomplish the job in a very efficient way (Welinder et al. 2010; Whitehill et al. 2009; Ross et al. 2010; Ahn and Dabbish 2004; Chatterjee and Bhattacharyya 2015; 2017). Moreover, in some of the real-life applications like smart city planning, facility location problem, etc., it is necessary to realize the exact public demand in different locations of the city. On the other hand, there should be an in-depth-knowledge about the demographic information of the city. For example, in different smart city planning applications, it is required to allocate resources based on the practical demand in that city. But gathering of proper information within a narrow time with additional constraints is a major challenge. Therefore, use of human knowledge can be very effective to locate the exact distribution of demands in the city. This can have an immense role to make a robust final decision promptly and it can be used to facilitate various issues relating to the urban development and smart city planning.

Engaging citizens for various city planning has already drawn much research interest over the years (Le Dantec et al. 2015; Evans-Cowley 2010). In the same vein, few researches have been carried out by engaging civic and inviting solutions from them in a public forum. Now as the opinions from the public (crowd) cannot be treated as reliable and spam annotators could be involved there, the major challenge is how to aggregate the multiple noisy opinions. Various methods have already been proposed in deriving a better consensus from multiple opinions to make a robust model (Raykar and Yu 2011; Hovy et al. 2013). Although the state-of-the-art approaches attempt to find consensus judgment from multiple solutions. But in most of the problems the questions that are posted online, can have binary option or multiple options and there is only one component of the question. For example, in crowd based tweet sentiment analysis task (Venanzi et al. 2013), the question is to infer the actual sentiment of a particular tweet and there are 5 possible options. In those questions, there are no subpart (component) of the question. But in different real-life applications, it is seen that the question (that is posted online) comprises some subparts or components. Moreover, there is a relation between the options of various components of the question. So while attempting a question, the crowd workers should be aware of these relations for all the components of the question. Additionally, a few crowd workers may try to get extra payment by attempting a large number of questions without satisfying this relation. Hence satisfying this relation becomes a constraint to the crowd workers and aggregating these types of opinions to make a better consensus turns out to be more challenging. Furthermore, it unfolds a new type of judgment analysis problem and hereafter it is termed as constrained judgment analysis problem. This problem introduces a new research direction of judgment analysis domain and this is illustrated with a simple example hereafter.

To explain this in more detail, let us consider a scenario in which an organization having limited knowledge about a city, is willing to install a number of ATM counters (e.g., 3 ATM counters) in the city. But to locate appropriate three places in a limited time is not an easy task. Again professional planning demands that there should be some specific distance between any two ATM counters. Thus to obtain the knowledge about the real public demand for the ATM counters, this problem can be outsourced to the people in an open forum for collecting their opinions. Now for this case, the question has three components i.e., the X and Y coordinate values for three locations. Thus the crowd opinions are of triplet of doublets i.e., 2D coordinate values of three locations and there exist some relations between these coordinate values. The constraint here is that the distance between any two ATM counters should be greater than a particular threshold value. So each of the crowd opinions can be thought of as a triplet where there is a relation between any pair of the coordinate values. Now to find aggregated decision majority voting cannot be applied here because if component-wise majority voting is taken, it does not guarantee the preservation of the constraint. Moreover, there is a little chance to obtain the same opinions twice over all of the six coordinate values. Thus the motivation comes from the fact that new opinion aggregation technique should be devised satisfying all the constraints that majority voting (MV) or other traditional opinion aggregation techniques cannot perfectly produce.

Commonly, we use majority voting that finds aggregated solution from multiple solutions. However, majority voting treats all the crowd workers as equally reliable having uniform expertise. But in practice, as there some spamming strategies may be involved, therefore various sophisticated methods have been proposed over the years (Bhattacharyya 2013; Ipeirotis 2010; Liu, Peng, and Ihler 2013). Although in these problems the question can have only single answer and so there does not exist any relationship between the options. But in practice, there are several real-life problems where the question (posted online) comprises k-lets (e.g., k locations) opinions where k can be anything greater than 1. Moreover, as there are some relationships between the k opinions, therefore it is very hard to build an opinion aggregation model similar like the traditional opinion aggregation model. Furthermore, it is really challenging if the range of options of each opinion becomes very diverse as no other traditional method is applicable to find solutions from it. Minimal studies are available in literature in this area and thus it motivates us to derive a suitable opinion aggregation model and find effective solution for strategic city planning.

In this article, we have explored this new research direction of crowd judgment analysis termed as constrained crowd judgment analysis. We have shown that this model can be helpful for the strategic smart city planning. We have also proposed a model that can tackle the complex opinions preserving the constraints.

The main contributions of the paper are:

- We introduce a novel crowd judgment analysis problem where each opinion is basically k-let opinions and each belongs to multi-class types. Moreover, there are some relationship between the opinions.
- We propose a novel opinion aggregation technique that can tackle these type of opinions and find a consensus solution satisfying the constraints.
- The efficacy of the proposed method is shown by applying it on a synthetic dataset and it ensures how smart city planning can be accomplished efficiently using it.

Related Works

Crowdsourcing has become a very popular tool over the years for annotating large scale datasets. Soliciting the opinions from crowd workers and producing aggregated opinion from it can solve a particular problem in very limited time as well as very within a limited budget. Amazon Mechanical Turk (AMT) (Sorokin and Forsyth 2008; Ipeirotis 2010) is a very well known crowdsourcing platform for these online annotations. Now there are various challenges that are needed to resolve to derive the final aggregated judgment from multiple crowd opinions.

Among the several popular opinion aggregation models, (Welinder et al. 2010; Whitehill et al. 2009; Hovy et al. 2013; Ross et al. 2010) uses probabilistic models to seggregate annotators based on their accuracies and biasness. In those methods Expectation Maximization (EM) is used to simulataneously estimate annotator bias as well as accuracy. Furthermore, in those models the latent features of annotators are taken into account to predict better consensus from multiple crowd solutions. Again most of the crowd opinion aggregation models can be categorized into supervised model as well as unsupervised model. But in these problems the opinions of crowd workers are not complex types and there exist no relationship between different options.

Again in recent years few researches have been carried out to derive consensus decision identifying spamming strategies (Raykar and Yu 2011; Jung and Lease 2012) from multiple crowd opinions. In those methods similarity of crowd workers, correlation among them are taken into account. In a few tasks like protein structure prediction (Peng et al. 2013) the relationship among various options are taken into consideration. But this problem differs from the current problem as it assumes that most of the workers are reliable and similarity matrices between the annotators can be formed very easily here. Moreover, as the protein structure prediction requires expert domain knowledge therefore formation of similarity matrices is meaningful here. Again in those problems majority voting can be applied here and basically kernelized majority voting is used. But in this problem majority voting cannot ensure the satisfiability of the specified constraint. Therefore, in this work we propose a probabilistic method that combine multiple diverse solutions into an aggregated solution with satisfying those constraints.

Problem Formulation

Let us formalize the constrained judgment analysis problem for a crowdsourced environment. We consider a set of questions $Q = \{q_1, q_2, \ldots, q_t\}$ that are the annotation tasks and a set of annotators $A = \{a_1, a_2, \ldots, a_n\}$ who are the crowd workers. The set of opinion vectors is $O =$

	Question 1	Question 2
Annotator 1	{(10, 20), (22, 33), (42, 30)}	{(10, 20), (20, 30), (40, 30)}
Annotator 2	{(10, 21), (20, 30), (44, 35)}	{(40, 30), (20, 30), (10, 20)}
Annotator 3	{(10, 12), (21, 27), (27, 23)}	{(11, 20), (2, 30), (43, 33)}
Annotator 4	{(11, 22), (20, 30), (29, 50)}	{(12, 22), (20, 30), (30, 30)}
Annotator 5	{(11, 23), (20, 30), (50, 30)}	{(10, 10), (20, 30), (40, 30)}

Figure 1: Response matrix of a constrained crowd judgment analysis problem that takes opinions in the form of multiple coordinates.

$\{\{(i_{1j}^{11}, i_{1j}^{12}, \ldots, i_{1j}^{1m}), (i_{1j}^{21}, i_{1j}^{22}, \ldots, i_{1j}^{2m}), \ldots, (i_{1j}^{k1}, i_{1j}^{k2}, \ldots, i_{1j}^{km})\}, \{(i_{2j}^{11}, i_{2j}^{12}, \ldots, i_{2j}^{1m}), (i_{2j}^{21}, i_{2j}^{22}, \ldots, i_{2j}^{2m}), \ldots, (i_{2j}^{k1}, i_{2j}^{k2}, \ldots, i_{2j}^{km})\}, \ldots, \{(i_{nj}^{11}, i_{nj}^{12}, \ldots, i_{nj}^{1m}), (i_{nj}^{21}, i_{nj}^{22}, \ldots, i_{nj}^{2m}), \ldots, (i_{nj}^{k1}, i_{nj}^{k2}, \ldots, i_{nj}^{km})\}\}$, for any particular question j, where i_{nj}^{km} denotes the opinion given by the n^{th} annotator for the k^{th} dimension of m^{th} component of j^{th} question. Here, there should be a relation between any pair of components.

An annotation process is a 4-tuple (Q, A, O, τ) consisting of (i) a set of questions Q, (ii) a set of annotators A, (iii) a set of opinions O, and (iv) a mapping function $\tau : (Q \times A) \to O$. The objective is to obtain the final judgment of all the questions in Q. Here, the cardinalities of the sets Q and A (t, n, respectively) are not necessarily the same. We define a response matrix $\mathcal{R}$ (as shown in Fig. 1) as a matrix of dimension $n \times m$ whose elements $\mathcal{R}_{ij}$ denote the opinion provided by the i^{th} annotator for the j^{th} question such that $\mathcal{R}_{ij} \in O$ for all i, j. For each of the question described in Fig. 1, there are 3 components having 2 coordinate values. The corresponding components of the question and the relation among them is illustrated in Fig. 2. Now for each of the component there are multiple opinions obtained from the crowd workers. So by discretizing these opinions for a specific component (i.e., for X and Y coordinate) we can obtain a judgment matrix $\mathcal{J}$ as a matrix (as shown in Fig. 3) of dimension $r \times s$ whose elements $\mathcal{J}_{ij}^{c}$ denote a weight of the (i, j) option for the c^{th} component of a question. Here r and s are the number of possible options (along X and Y) after discretizing the responses collected from the crowd workers for a specific component of the said question.

Proposed Model

In this section, we introduce the key challenges that may generate hindrance at the time of aggregation of all the scores given by the various annotators. Thereafter, we discuss the proposed model for this problem.

Removal of Inconsistencies of Different Solutions

In this current problem the crowd workers are asked to choose k number of best possible location according to their perception for allocation of any kind of resources. Now as there are no restriction about ordering of the locations therefore two same solutions provided by two different crowd workers may seemed to be different. The reason is that one worker can label a location as first whereas the another worker may label the same location as second. For example, for annotator 1 and 2 in question 2 of Fig. 1, it is seen that the first location provided by crowd worker 1 and third location provided by crowd worker 2 are basically same coordinate values. But it seems from their solutions that these are different solutions. Therefore the labeling between the solutions should be standardized before proceeding further.

In this model, to remove this discrepancy we need to first relabel all of these solutions based on a reference solution. Now to choose the reference solution we have selected the medoid of those solutions. So for standardization of any solution, the first component of the reference solutions is compared with all of these components of the other solutions. Then the component with minimum Euclidean distance is selected as the first component of the other solution. This component of the solution that has already been relabeled is not considered again for further processing. After this step, this second component of reference solution is compared with all of components of the other solution excluding the first component. In this way, all of the components of a particular solution is relabeled and thus the standardization of all the solutions are done.

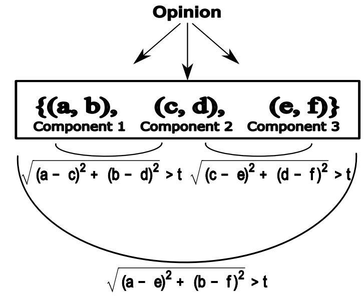

Figure 2: Various component of a sample question. The options of each component should have a particular relations among them. Here t denotes an arbitrarily chosen threshold value.

Y Coordinate / X Coordinate	20	30	40	50
10	0.03	0	0.02	0.04
20	0.05	0.2	0.3	0
30	0.02	0.00	0	0.02
40	0	0	0.06	0.03
50	0.2	0.02	0.01	0

Figure 3: Judgment matrix for a particular component of a question.

Removal of Constraints Violating Workers

In this step, as some of the annotators try to pretend themselves as an expert they can provide some wrong annotations violating the constraints. In order to keep the solution space noise free, the annotators violating the constraints are removed. Thus the response matrix in this step is reduced version of original response matrix and the each of the solution in this matrix can be a potential solution. So this response matrix is ultimately taken as an input for next processing.

Discretization of Coordinate Values

To choose a location based on human viewpoints, there is no fixed possible option set. Therefore, annotators are free to choose any coordinate values based on their own understanding. From the response matrix (after reduction) we can compute the maximum and minimum coordinate value of any particular component. Thus it can be easily realized that the final solution should lie within this range. But to derive the judgment matrix we need to define the possible option set. Basically the option set for which the weight is maximum should be chosen as final answer for a particular component. But specifying the range cannot derive the possible options of the judgment matrix and without knowing the possible option set the judgment matrix cannot be formed.

To illustrate this step we need to begin with the complicacy of not specifying the option set to derive final aggregated judgment. If the possible option set is not defined then the weight of the possible options cannot be calculated and thus it becomes hard to compute the final judgment. So as the problem does not define any possible option set therefore possible option set should be explicitly calculated here. We cannot define any particular threshold value arbitrarily to set the posible option set. To mitigate this problem, we have made an histogram based on the coordinate values. For example, for a particular component X coordinate values of all the annotators lie between 0-100. So as the values are continuous therefore we cannot imply anything about the possible options for the X coordinate. Furthermore, we can take decision about some hard boundaries but this might cause information loss of the annotators. But we can compute histogram of it i.e., the number of annotators who have selected a particular X coordinate value. Now on this histogram Bayesian binning (Knuth 2006) is used to capture the features of the underlying data reducing the high range into small range of options.

Bayesian binning deals with the problem of finding optimal number of bins in uniform bin-width histogram. It uses Bayesian probability to find posterior probability of number of bins for a dataset. In this framework, as the underlying density is unknown, so likelihood and prior are considered to calculate the posterior probability. Let d_n be the datum and probability density of finding it in the k^{th} bin, within the range $[x, x + dx]$, is calculated by using a likelihood function.

If I is the prior knowledge about the range of data, then for N independently sampled data points, the joint likelihood is computed as follows.

$$p(\underline{d}|\underline{\pi}, M, I) = \left(\frac{M}{V}\right)^N \pi_1^{n_1}\pi_2^{n_2}\dots\pi_{M-1}^{n_{M-1}}\pi_M^{n_M}. \quad (1)$$

Where $\underline{d} = \{d_1, d_2, \dots, d_N\}$, $\underline{\pi} = \{\pi_1, \pi_2, \dots, \pi_{M-1}\}$ and n_i be the number of datapoints in the i^{th} bin. M be the number of bins and V be the total width of the dataset. Each π_1, π_2, ..., π_{M-1} denotes the probability that sample is drawn from each of the M bins.

By using Bayes' Theorem, the posterior distribution based on the histogram is computed by using prior and joint likelihood as follows.

$$p(\underline{\pi}, M|\underline{d}, I) \propto p(\underline{\pi}|M, I)p(M|I)p(\underline{d}|\underline{\pi}, M, I). \quad (2)$$

In this problem, to find the number of bins for any particular coordinate values, the mean and standard deviation of the opinions of all the annotator are calculated. Then assuming the data is in Gaussian distribution, a dataset with same number of options is generated using the same mean and standard deviation. This dataset is basically the representative of the original dataset and from it the optimal number of bins can be calculated that captures the major features of entire dataset. After obtaining the optimal number of bins, the starting range of each bin in terms of coordinate value is measured. Finally the boundary values of all bins are treated as the possible options.

Probabilistic Graphical Model based Method

In this proposed methodology, a probabilistic model is used to compute the final posterior distribution of each option component-wise. Here the annotators opinion is dependent on the accuracy and question difficulty. If the difficulty of question becomes less, then the chance of getting correct answer becomes high, whereas if the accuracy of annotator becomes high then the probability of getting correct answer gets high.

Consider a response matrix with m annotators giving their opinions from the options set $\{1, 2, \dots, z\}$ over n component of a question. We have to predict the original label z_j of each component j for every question taking opinion from m annotators. Note that, in this problem the opinions are basically a 2D coordinate value. Therefore, the final option for each component will be a 2D coordinate value. As earlier mentioned, the observed opinion is dependent on two factors: (i) the accuracy of annotator, (ii) the difficulty of question. The expertise of an annotator i is denoted by a parameter α_i. Here accuracy is defined by measuring its deviation from the median solution in respective of all the components. We have also considered the coverage area enclosed by the points to compute accuracy. The area in the triangle enclosed by these three points is normalized by the deviation from the mean of each solution for initial accuracy. Here, α_i = 0 means the annotator has lowest accuracy whereas α_i = 1 means the annotator have given all the opinions correctly. The difficulty of question j is denoted by $\beta_j \in (0, \infty)$. How the predicted opinion is dependent on question difficulty and annotator accuracy is shown in Fig. 4.

Suppose, λ_{ij} denotes the opinion given by a particular annotator i for a given component j of a question. Then the probability that the given opinion matches with the true label is given by following equation (3). Here, z is the true option that needs to be estimated. Note that, z comprises two coordinate values (i.e., X and Y) for a particular component.

$$P(\lambda_{ij} = z|\alpha_i, \beta_j) = \frac{1}{1 + \exp(\frac{-\alpha_i}{\beta_j})}. \tag{3}$$

The rationale behind Eqn. (3) is that more the expertise of an annotator (annotator having high accuracy) means it has a higher chance to give the correct opinion. Again, more difficult the question the probability that it will be annotated correctly is lesser.

The graphical structure of the proposed model is shown in Fig. 4. Here, δ_j is the neighborhood set of annotators who gave their opinions for the j^{th} component of a question, and σ_i is the set of questions that have been annotated by the i^{th} annotator. The predicted label of component j and annotator accuracy are denoted by z_j and α_i respectively. Question difficulty β_j is calculated from the standard deviation of different options obtained from all the annotators. Given a set of observed labels $\lambda = \lambda_{ij}$, we adopt a Expectation-Maximization approach (EM) to estimate the posterior distribution of the true option. The observed opinion for the j^{th} component given by the i^{th} annotator is λ_{ij}. The posterior distribution of j^{th} component in case of true option z is denoted by $\mathcal{J}_j(z)$. Our goal is to find the most probable value of $\mathcal{J}_j(R_{ij})\ \forall i \in \delta_j$ by the maximum likelihood estimation method.

As the annotators give their opinions independently, therefore, the likelihood function will be the multiple of their probability distribution values and it has been defined in Eqn. (4). Here, θ denotes the set of parameter values that we need to estimate, i.e., annotator accuracy, and x_1, x_2 are the individual annotations. In this context, $f(x_i, \theta)$ denotes the probability distribution function depending upon θ.

$$\lambda_{ij}(\theta|x_1, x_2, \ldots, x_n) = \prod f(x_i, \theta) \tag{4}$$

E Step: We assume that one who answers a particular question should provide all the coordinate values for all the components. Let the set of opinions given by all the annotators who attempted the component j is δ_j. Initially, the different opinions (obtained from different annotators) are counted for a particular component. We need to estimate the true value of this particular component. Now, we compute the posterior distribution of each option and in this context binomial distribution is used. As the annotators give their opinions independently (one opinion is not affected by other) and for a single question multiple opinions are solicited, therefore binomial distribution is the best fit. So, the posterior distribution of each option for any particular component is defined as follows.

$$\mathcal{J}_j(z) = \prod_{i \in \delta_j} P(z)^{I(R_{ij}=z)}(1 - P(z))^{I(R_{ij} \neq z)} \tag{5}$$

Here, $\mathcal{R}_{ij}$ is the response of i^{th} annotator for the j^{th} component and $\mathcal{J}_j(z)$ is the posterior probability of the j^{th} component when the option is z. The indicator function I returns 1 if the annotator's response $\mathcal{R}_{ij}$ matches with the corresponding true label z for which the posterior distribution is calculated.

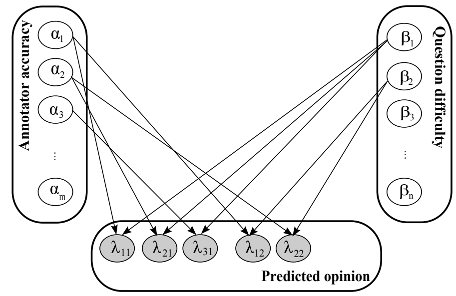

Figure 4: Graphical model for combining question difficulties, annotator accuracy and the predicted label.

M Step: Annotator accuracy (i.e., α_i values are updated in the following M step.

$$\alpha_i = \frac{1}{|\sigma_i|} \sum_{i \in \sigma_i} \mathcal{J}(R_{ij}) \times \mathcal{C}_i \tag{6}$$

Here $\mathcal{C}_i$ denotes the coverage of the particular solution i. In this problem, better the coverage means the points are well distributed.

In order to find the aggregated opinions of the crowd workers, we apply the graphical model on the response matrix (obtained after Bayesian binning) and generate a set of solutions as final solutions. In this dataset for each of the component of the question, the posterior distribution of each option is computed. Now in this respect each option for a particular component is basically a 2D coordinate value. The maximum posterior distribution value of a particular option denotes the most probable option for the particular component. Therefore by sorting the posterior distribution values in descending order we obtain the rank of the possible solutions for a particular component. In the similar way we can generate solutions and compute rank of the solutions for other 2 components (as there are 3 components in total). It is termed as component-solutions hereafter. Now these solutions are combined satisfying the constraint conditions. In this combination step, we compute the overall ranking of a solution by considering all of the component-solutions. In this regard we compute the summation of the individual ranking of the component-wise solutions. For example, the best possible solution will be taking rank-1 solutions from each of the component if it satisfies the constraint. So the overall ranking score is 3 for this choice. Similarly, if we can combine the rank-2 solution of first component with the rank-1 solutions of other two components then the overall ranking of the solution becomes 4. Thus in this respect, for a particular overall rank multiple solutions can exist and ultimately the coverage area can be used as an effective metric to determine the best possible solution over all the same rank solutions. In this way, final aggregated judgment can be calculated preserving the constrained conditions.

Experimental Design and Results

In order to measure and analyze the performance of the proposed approach, we have prepared one synthetic dataset. As

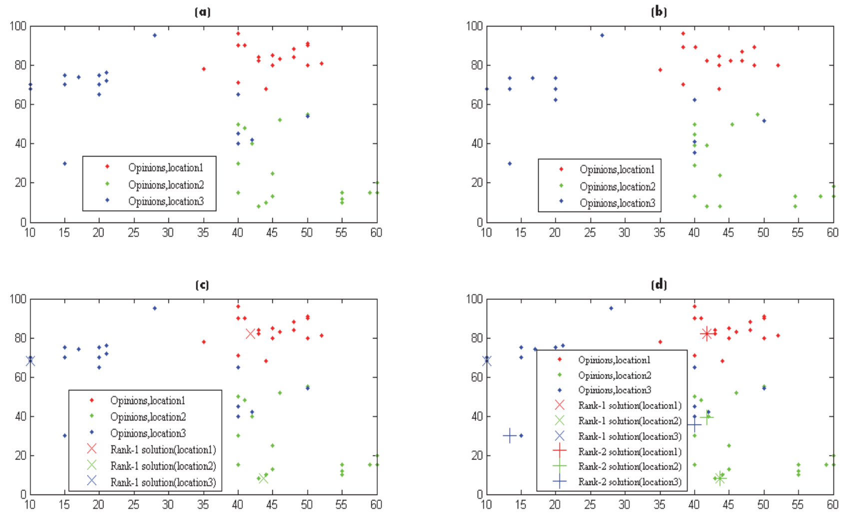

Figure 5: (a) Visualization of various crowd opinions (b) Visualization of crowd opinions after binning (c) Rank-1 solution for all of the locations (d) Rank-2 solution for all of the locations.

this problem is new one and no other approaches is available as per our knowledge therefore we are unable to compare this with other state-of-the-art opinion ensemble methods. Experiments have been performed with MATLAB 2008a and the running environment is an Intel (R) CPU 2.4 GHz machine with 4GB RAM running Windows 7 Ultimate. The description of the dataset is given below.

Dataset Preparation

We have designed an online platform and a grid map (specifying the coordinate value) of a particular state of India were posted. In this platform, we have a supplied an information that a top-tier US university wishes to introduce three extension centres in that state and there is a constraint that any pair of centres should be apart by a distance of 30 units. Now the opinions about the best possible answers are solicited from them. This kind of opinions basically fulfill to locate the exact demand of extension centre in that particular state. Therefore, every crowd worker may opt any three locations they find it appropriate. For example, few crowd workers may choose locations nearest to their home town. Another one can choose the most developed and well connected regions for the same. Thus in this way multiple diverse solutions can be obtained. In this current dataset, there are 20 opinions received from the crowd workers and 18 out of them provided the opinions satisfying the constraint.

Study on the Dataset

As different crowd workers registered their opinions regarding the best three locations for a particular case, the opinions are expected to be diverse. Therefore, to group their opinions into smaller number of specified options sets, we apply Bayesian binning on each of the set of coordinate values. Here, the problem is to calculate the optimal number of bins to capture the whole distribution of dataset. Bayesian binning is an effective approach to resolve this. Here, each annotator provides two coordinate values (i.e., X and Y coordinates) for three locations resulting into six coordinate values in total. For each of these six set of coordinate values for all the crowd workers, we apply Bayesian binning and the optimal number of bins are obtained as 10, 12, 11, 9, 12, 12, respectively (through Gaussian estimation). Thereafter, we apply the proposed model on this dataset to combine the multiple opinions in order to derive the aggregated opinions. The opinions of crowd workers is shown in Fig. 5 (a). The opinions of the same number of workers after Bayesian binning is shown in Fig. 5 (b). It is seen that the diversity of the opinions has been reduced due to grouping the coordinate values into smaller number of coordinate values. Finally, the rank-1 and rank-2 solutions are shown in Fig. 5 (c) and (d). It is seen from Fig. 5 (c), for location 2 and location 3 the final solution is in middle of the obtained opinions. But for location 1 the final solution is aligned in left (along the vertical axis). The reason is that although majority disagree with it but as accuracy of few annotators responding this solution are better so it provides higher posterior distribution.

Conclusions

In this paper, our prime focus is to utilize the power of crowd knowledge to accomplish a proper and prompt smart city planning. As expert smart city planning always requires an adequate knowledge about the city and it can be obtained very easily if the crowd can be involved logically in this planning process. We propose a probabilistic method that combines multiple diverse solutions. This has an enormous scope as other novel quality metrics for segregating crowd workers can be introduced to make the model more robust.

Acknowledgments

The work of Malay Bhattacharyya is supported by the Visvesvaraya Young Faculty Research Fellowship 2015-16 of DeitY, Government of India. Anirban Mukhopadhyay acknowledges the support received from DST-PURSE grant of University of Kalyani. All the authors would like to thank the crowd contributors involved in this work.

References

Ahn, L. V., and Dabbish, L. 2004. labeling images with a computer game. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 319–326.

Bhattacharyya, M. 2013. Opinion ensembling: Learning from dependent judgements of the crowd. In *Proceedings of the Crowdscale Shared Task Challenge*.

Chatterjee, S., and Bhattacharyya, M. 2015. A biclustering approach for crowd judgment analysis. In *Proceedings of the Second ACM IKDD Conference on Data Sciences*, 118–119.

Chatterjee, S., and Bhattacharyya, M. 2017. Judgment analysis of crowdsourced opinions using biclustering. *Information Sciences* 375:138–154.

Evans-Cowley, J. 2010. Planning in the real-time city: The future of mobile technology. *Journal of Planning Literature* 25(2):136–149.

Hovy, D.; Kirkpatrick, T. B.; Vaswani, A.; and Hovy, E. 2013. Learning whom to trust with mace. In *Proceedings of the North American Chapter of the Association for Computational Linguistics – Human Language Technologies (NAACL HLT)*, 1120–1130.

Howe, J. 2006. The rise of crowdsourcing. *Wired magazine*.

Ipeirotis, P. 2010. Analyzing the amazon mechanical turk marketplace. *ACM XRDS* 17(2):16–21.

Jung, J. H., and Lease, M. 2012. Improving quality of crowdsourced labels via probabilistic matrix factorization. In *4th Human Computation Workshop (HCOMP) at AAAI*.

Knuth, K. 2006. Optimal Data-Based Binning for Histograms. *arXiv:physics/0605197*.

Le Dantec, C. A.; Asad, M.; Misra, A.; and Watkins, K. E. 2015. Planning with crowdsourced data: Rhetoric and representation in transportation planning. In *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work*, CSCW '15, 1717–1727.

Liu, Q.; Peng, J.; and Ihler, A. 2013. Report of crowdscale shared task challenge 2013. In *Procedings of the Crowdscale Shared Task Challenge*.

Peng, J.; Mit, C.; Liu, Q.; Uci, I.; Ihler, A.; Uci, I.; Berger, B.; and Mit, C. 2013. Crowdsourcing for structured labeling with applications to protein folding. ICML Workshop: Machine Learning Meets Crowdsourcing, Atlanta, Georgia, USA.

Raykar, V., and Yu, S. 2011. Eliminating spammers and ranking annotators for crowdsourced labeling tasks. *Journal of Machine Learning Research* 13:491–518.

Ross, J.; Irani, L.; Silberman, M.; Zaldivar, A.; and Tomilson, B. 2010. Who are the crowdworkers? shifting demographics in mechanical turk. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, 2863–2872.

Sorokin, A., and Forsyth, D. 2008. Utility data annotation with amazon mechanical turk. In *Proceedings of the Computer Vision and Pattern Recognnition Workshops (CVPRW)*, 1–8.

Venanzi, M.; Guiver, J.; Kazai, G.; and Kohli, P. 2013. Bayesian combination of crowd-based tweet sentiment analysis jugdments. In *Proceedings of the Crowdscale Shared Task Challenge*.

Welinder, P.; Branson, S.; Belongie, S.; and Perona, P. 2010. The multidimensional wisdom of crowds. In *Proceedings of the Neural Information Processing System*, 2424–2432.

Whitehill, J.; Ruvolo, P.; Wu, T.; Bergsma, J.; and Movellan, J. 2009. Whose vote should be count more: Optimal integration of labels from labelers of unknown expertise. In *Proceedings of Advances in Neural Information Processing Systems*, 2035–2043.