Open-Ended Community-Sensing: Collecting and Processing City Data from Citizens

Sonia Soubam*, Naveen Prathapaneni, Vikrant Nandakumar, Kuldeep Yadav*, Vinayak Naik*, Nithya Rajamani, Dipanjan Chakraborty, L. Venkata Subramaniam, Sameep Mehta, Pradipta De

IBM Research, India

*Indraprastha Institute of Information Technology (IIIT), Delhi, India

ABSTRACT

With the growing number of cities and population, continuous monitoring of city's infrastructure and automated collection of day-to-day events (such as traffic jam) is essential. Continuous data analysis on collected events can be used to find patterns which may help in taking development decision for the city and also, eventually improve the life style of citizens. In this paper, we describe a crowdsensing effort for capturing events affecting citizens in cities in India. We have designed an end-to-end system which allows citizens to *sense* the city and report different events happening in their surroundings. The system currently allows submission of events under five broad categories: Civic complaints, traffic, neighbourhood issues, emergency and others and events can be submitted through different submission modes i.e. mobile application, web-based form and SMSes. Users can report events by using text, images or audio. Several of these reports are also tagged with location information that includes names of locations and/or latitude and longitude position and/or cell tower location.

We have deployed this system among university students to report events happening around them. We find and describe various challenges in analysing the multi-modal data and our efforts in pre-processing of data to make it usable. We also do a preliminary analysis of issues that the community is facing and to detect any trends that emerge from the data.

Categories and Subject Descriptors

I.2.7 [**Artificial Intelligence**]: Natural Language Processing—*text analysis*

Keywords

Crowdsensing, Information Extraction, Text Classification

AND '09, July 23-24, 2009, Barcelona, Spain

1. INTRODUCTION

Many developed countries have city-wide deployment of sensing infrastructure to collect data about day-to-day city problems (i.e. traffic congestion), analyze collected data and then takes prompt action based on that data. For instance, USA have deployed traffic sensors across major highways to monitor the health of roads and to detect timely events such as traffic congestion, data collected from these sensors is also useful to make broad city development decisions. Continuous monitoring of city's infrastructure using sensors is costly and does not exist in many countries due to lack of resources, cost, and bigger scale of deployment. With the advent of mobile phones, citizens can help in collecting this data with rich contextual information (such as location, pictures), we call it as *crowdsensing*. Crowdsensing has emerged as a viable way of collecting rich data about day-to-day city problems without investing in specialized sensing infrastructure specifically in resource constrained environments. There has been several efforts to use crowdsensing with mobile phones for variety of purpose such as to study pollution exposure [8], pothole detection [7] and traffic conditions [10]. However, all these efforts cater to only one application scenarios and does not scale well due to their specifics.

There has also been efforts on using social media to collect and disseminate information about city events. For instance, there are some initiatives started by government departments in India [1] and individuals [2] [3] to collect data using crowdsensing primarily using social networks. However, these initiatives focus on reporting of only one kind of events (reports) and also, many of such initiatives collect text reports which are hard to analyze without support of any contextual data.

Mobile phones enable easy sharing of information about unfolding events by the people witnessing the event. Also, mobile devices are equipped with various kinds of sensors such as GPS (Global Positioning System), accelerometer, microphone and camera. To enable crowdsensing, we have built an end to end system which constitutes of a mobile application and a central server. Our mobile application pro-

[1] https://www.facebook.com/pages/Delhi-Traffic-Police/117817371573308?fref=ts
[2] www.powercuts.in
[3] http://www.ichangemycity.com/

vides an intuitive and usable user interface to allow users to report events of a city falling under five broad categories: Civic complaints, traffic, neighbourhood issues, emergency and others. Our mobile application provides a choice to user to further, enrich the textual report with contextual data such as image and audio; it also captures user's accurate location using GPS if the user enables it or coarse grained using cellular tower information. For instance, if there is a traffic jam at some place B then user can submit a textual report "Traffic Jam at B", clicks a picture showing the length of traffic jam and mobile application automatically gets the accurate location using GPS and sends it to the server. Central server is used to automatically aggregate reports submitted by mobile application used by many participants and then, finding patterns from the data. Unlike existing initiatives, proposed system provides a unified interface to crowdsense a city's problems.

To build large participation in this effort, we have started a national university challenge where university students, mostly engineering college students, from India are invited to register and start contributing information about events happening around them. To make the data collection truly participatory, all the registered participants are allowed access to all the collected data from the challenge. The students are offered incentives in the form of awards, which are given to the students collecting the most amount of data and the students doing the most interesting analysis on the data. We believe that data collected using this crowdsensing effort can be used to determine the issues concerning the citizens. Our goal is to analyze the data to determine local and global issues and trends as perceived by the community. We also wish to demonstrate the effectiveness of crowdsensing in capturing events of interest to the community. It is expected that most participants will report events that they are experiencing in first hand and it is hoped that they will also report the event during the time they are witnessing it or immediately afterwards. The system is still in deployment phase and many users are actively submitting events, in this paper, we describe our efforts of preprocessing of the collected data and also, report the results of some initial analysis of the collected data.

The paper is organized as follows, Section 2 discusses the principles and aim of using humans with mobile phones for sensing tasks, it also distinguishes between crowdsensing and crowdsourcing. Section 3 presents the design of the system, implementation details and its deployment with real-world participants. Section 4 presents the amount of data collected, pre-processing of data and preliminary analysis on the collected data. Finally, Section 5 presents the discussion section with implications of proposed system and plans for future analysis on collected data.

2. THE PRINCIPLES OF CROWDSENSING

Crowdsensing is a relatively newer term than Crowdsourcing. It is sometime also referred as participatory sensing [4]. The fundamental difference between crowdsourcing and crowdsensing is in the nature of the task. Typically Crowdsourcing tasks are driven top-down, i.e. a task is decomposed into suitable pieces that can be autonomously executed. Thereafter, different solutions are used to reach out to the community and get the pieces executed. Crowdsensing on the other hand is driven bottom-up, i.e. the community here is organically involved in sensing a phenomenon of interest; results are aggregated and quality of data is critically guided by what the community percieves and how close it is to reality.

Both crowdsourcing and crowdsensing systems are investigating how to exploit mobile phones. Crowdsourcing typically utilizes the mobile as a communication modality. The human owner is typically responsible for executing the task (e.g. translating a text). In Crowdsensing, the mobile phone and its sensors plays the role of *sensors* as well [1], aiding the task owner in the process. For e.g, in our investigation, the primary purpose of the task collector was to *sense the city*, i.e. any public events that a human senses in his/her surrounding, we are interested in receiving some information about it. Such information could be user's interpretation via text/voice updates, accoustic context, video, location, social vicinity, activities around the event. The mobile phone's features are actively used to carry out such a task.

As such, crowdsensing driven applications have a few goals that should ideally be met: (1) The sensed data should provide progressively richer understanding about the phenomenon (2) It should be beneficial to execute the sensing task using the crowd as opposed to traditional sensors, where benefit can come either from infrastructure cost saving, or from quality of data being sensed, (3) Privacy and autonomy of the task executors must be respected. Our focus is on sensing events of public interest in the city (e.g. neighborhood issues, traffic, public safety etc). These events are typically ad-hoc as opposed to pervasive in nature (e.g. air pollution, sound pollution). We call this kind of task *adhoc crowdsensing*, where the community is doing at maximum a best-effort practise of sensing such events.

Figure 2 depicts the high-level system description of our crowdsensing and multi-modal data fusion ecosystem that we implemented in our pan-india trial.

3. SYSTEM DETAILS AND DEPLOYMENT

In our system, we have allowed three different modes to submit event, (1) Using an Mobile application created for this task, (2) Short messaging service (SMS) and (3) Web-based form. Participants are required to register online with their details before they can start submitting events [4].

3.1 Mobile Application

We built an application for only Android OS due to two different reasons; (1) It provides rich support of APIs to capture contextual data (2) Android based mobile devices are getting increasingly popular in developing countries such as India. The Android application is designed in such a way that it provides a user friendly UI for the participants to report events with minimum efforts. Whenever, a user wants to submit an event, she can launch application in her phone as shown in Figure 1a. On application launch, the user is directed to the event type screen on click of *Start Reporting* button as shown in Figure 1b. In this screen, the user chooses a category which broadly describes the event viz. civic complaints, traffic, neighbourhood issues, emergency or others. After choosing an appropriate category of the

[4]http://kalpa.haifa.il.ibm.com:9080/indiaChallenge/

(a) Home Screen of the Android app (b) Event Type Screen of the Android app(c) Event Input Screen of the Android app

Figure 1: Step by step working of our Android mobile application to submit events

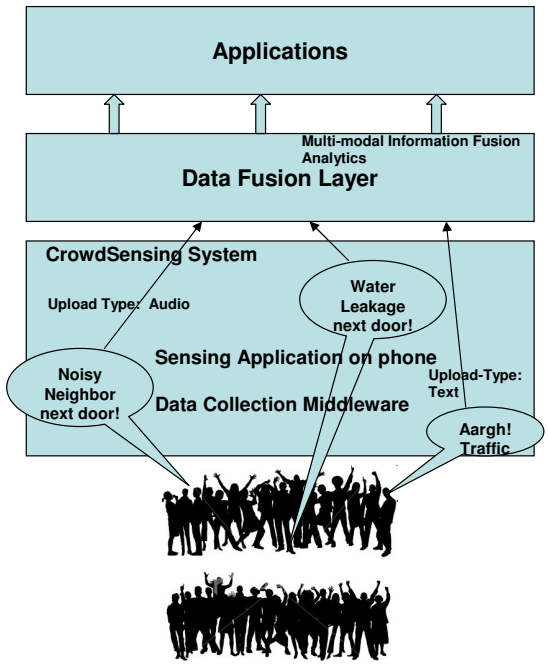

Figure 2: CrowdSensing and Multi-modal Data Fusion Ecosystem

event, the application leads the user to the main screen as shown in Figure 1c asking for details about the event. At this point, the GPS sensor on the phone (if available) is turned on and starts searching for a valid signal.

The user is asked to input a free form text describing the event, the location of the event, and some appropriate tags related to the event. To provide more contextual information which can further assist the event report, the participant can also click the button *Click Image* which starts the camera of the phone and captures an image. The *Record Audio* button records a short audio clipping of 10 second duration to capture the sound in the vicinity of the event. On pressing the *Submit* button all the data including the text input, image, audio clip along with the GPS coordinates is uploaded using HTTP Post request to a server. Based on HTTP post request response, user gets a notification on their phone either acknowledging successful upload or an error message incase of failure.

Since, most of crowdsensing participants are volunteers with little or no incentives, system has to take special consideration to minimize costs (both monetary and energy) at participant end. While using our android app, participant have to incur data connection cost while submitting an event. As shown in Table 1, submitting a single event transfers total data of less than 0.5 MB which is reasonable and does not put extra burden on participant. We are also working towards providing a cache functionality in mobile application where participant can record an event whenever it happens but upload it in bulk whenever there is free internet connection available i.e WiFi etc.

Event Fields	Size
Textual Description + Location Info	Few Bytes
Audio Recording (10 sec)	350 KB
Picture (4 MP Camera)	40 KB

Table 1: Data size of different event fields captured by Android Mobile Application

3.2 SMS-based Event Report Submission

In developing countries, there are many phones which have limited capability and may not have any programmable interface [5]. Also, there are significant number of phones which do not run Android OS. To extend the reach of our event report submission, we enabled participant to send report

[5]http://www.newscientist.com/article/mg21528844.600-gps-workaround-helps-make-dumb-phones-smart.html

via SMS messages also. This option is is suitable for non-programmable phones, non-supported smartphones, and users who do not prefer to use their data connection for sending reports.

When a user wants to submit a report, the user generates an SMS in the following format and sends it to our published cellphone number. < Event Description [Delimiter] Contextual Information > The user starts by putting an event description and appends contextual information if any, e.g., location. The following is a sample report: *Police asking 1000 rs bribe for approving passport for a friend, though all documents are perfect @ indore*

We have setup a SMS center which can receive and process SMS sent by a user. The SMS center constitutes a Linux-based server connected to a GSM modem, which has a SIM card in it. The server periodically polls the modem for received SMS messages. Once a SMS is received, the server retrieves the SMS content with other header information, such as timestamp, sender, etc.

Though, SMS based event report submission enables people with different phones to participate in crowdsensing efforts, however, this method can not take advantage of various sensors in the phone for adding contextual data to event report.

3.3 Web-based Event Report Submission

For non-Android smartphone or tablet users, we have also enabled web based event report submission. After one time registration and login, user can go to our website [6] and submit the even report similar to Android application. Our web-based form too allows rich data collection which consists of text, audio and video inputs.

3.4 System Deployment

To deploy the system in real-world, we have started a national university challenge (NUC) where university students, mostly engineering college students, from India are invited to register and start contributing information about events happening around them. We have hosted our Android application on Google Play Store [7] and put a link on our website. Our website had other details such as SMS center number and registration form to register into the NUC and submit events. The challenge is publicized by directly reaching out to engineering colleges though emails and posters. There are also facebook and twitter pages to continuously engage the students. The system is currently running in real-world where participants are submitting reports and will be open till March 2013.

Due to multiple event report submission methods, we captures different kind of information among which some of them need to be input by the user, while others get automatically captured and are sent when the event is submitted. Following is a listing of different event fields categorized by automatic and manual submission where submission methods that support each detail are listed inside brackets.

1. **Automatically Sent Details**
   - Event Time Stamp: Actual date and time when event occurred (SMS, web, mobile application)
   - Latitude: Latitude of event location (mobile application)
   - Longitude: Longitude of event location (mobile application)
   - Cell Tower Info: MCC (Mobile Country Code), MNC (Mobile Network Code), LAC (Location Area Code), Cell ID (a unique identifier given to each cell tower in an area)
2. **Manually Submitted Details**
   - Type of Event: Mapped to one of 5 event categories (web, mobile application)
   - Message: Actual message in text format (SMS, web, mobile application)
   - Tags: Tags related to message (web, mobile application)
   - Location: Location in Area , City (SMS, web, mobile application)
   - Image : Image recorded/submitted (web, mobile application)
   - audio: Audio recorded /submitted (web, mobile application)

3.5 Incentives for participation

Incentives are particularly important in crowdsensing, because they can mean the difference between success and failure. The principal dimension along which types of incentives can be split are monetary (e.g. rewards) and non-monetary (e.g. reputation). Our effort is to have a mix of both to encourage continued participation.

The rewards include t-shirts for the most active participants. Also the authors of the best papers will win mobile phones and music devices.

The non-monetary rewards include classfication of users as level-I, level-II and so on based on the number of events submitted by each user. Also all participants are given access to the data being collected. They are free to analyze the data and publish their work based on it. Further we a conference will be held at the end of the challenge where the participants are invited to submit their research papers. The papers will be reviewed and the top submissions selected, recognized and rewarded. We hope that these incentives will promote healthy competition among the participants both to share events and then to analyze the collected data. But we also hope to create an environment for collectively gathering sufficient data to sense events of importance to the community.

To publicize the event we distributed posters and sent email messages to engineering colleges throughout India. We also created Facebook and Twitter pages to constantly engage the participants and recognize active participants. Since this is a fairly long crowdsensing effort to maintain sufficiently high quality submissions, we plan to keep engaging the users to keep up their interest levels.

[6] http://kalpa.haifa.il.ibm.com:9080/indiaChallenge/
[7] http://goo.gl/dJK8Y

Participants can be classified based on their participation levels and incentives can be designed accordingly. The SCOUT model [2] segregates the crowd into: (S)uper contributors, (C)ontributors, and (OUT)liers.

Super Contributors are an elite group of participants who exceed the expectations, by submitting a large number of high quality event data. They are motivated by their own advancement or passion for the topic and can be counted on to continue to participate for the full duration of the challenge.

Contributors give reasonable effort and contribute just enough to make it to the set goals for getting a prize. They focus on the rewards (extrinsic motivation) and are always conscious of where they rank (at each point) relative to the reward and the end date of the task that is being accomplished. Essentially, they monitor the communication for the announced tasks, duration, and associated rewards, and then contribute accordingly.

Finally, OUTliers (crowd) are participants who may be considered as leisurely participants. They tend to have just enough motivation to participate, but often don't sufficiently engage to make any solid contributions that would qualify them to receive rewards or recognition.

The levels that we assign to the users map to the SCOUT model.

4. DATA COLLECTION AND ANALYSIS

Till now, we are able to collect a total of 838 events from three different submission modes as shown in Table 2. A total of 435 users have registered so far in the system. Interestingly, the total number of registered participants from Uttarakhand state is only 10 but several of them have been super contributors. But, Andhra Pradesh state with over 85 registered users has only 15 event submissions.

While participants can submit text, audio and images, predominantly they have submitted text details of the events. Overall, we got 838 text reports i.e. all the submitted events had text, 182 of events had images and 5 also had audio clip. Because, we got mostly text reports, we will focus on the text submissions only.

Input method	Number of Submissions
Android Application	140
Web-based	488
SMS-based	210
Total	838

Table 2: Break up of different event report submission methods

4.1 Pre-processing

Data collected using crowdsensing may contains noise and need some pre-processing before it can be used for analysis. Intentional corruptions are very common in data uploaded from mobile devices [3]. This is due to the limited data entry options (keypad constraints on mobile devices) and due to the pressure of reducing communication latency (or cost in case of SMS) by keeping messages short yet intelligible. Words are intentionally compressed by non-standard spellings, abbreviations and phonetic transliteration. Also markers such as full-stops and commas are either not used or then used in non traditional ways.

Apart from cleaning, one of the aim of pre-processing step to extract data which can help while analyzing an event report i.e. location, category tags etc. For instance, Table 3 presents a row entry for an event which has all the entries filled, however there are many reports submitted by participants does not have many of these fields which have to be extracted from raw text. Hereby, we list some of the techniques we have used for pre-processing of the collected data.

4.1.1 Cleansing

Participants submitted junk events to try out the submission modes which does not carry any useful data. The initial cleansing step comprises of the following tasks: Cleaning special characters in the messages, removing spam, removal of unwanted or junk events. Each of these tasks is accomplished by defining rules. For example one of the rules to remove advertisements and promotional SMSes is to mark messages where the sender id has markers such as AD, LM, TD, DM and other alpha-numeric identifiers used for commercial SMS. Sometime it is not possible to filter spam SMSes based on only sender information, we take help of techniques presented in [5] to filter based on content.

4.1.2 Classification

Events submitted using SMSes do not have any broad category classification during submission. We classify received event reports that do not have tags into broad event categories, i.e., civic complaints, traffic, neighbourhood issues, emergency or others. The classification of an event report is done using simple keyword matching and verified by manual tagging. As we get more reports in the system, we will use automated text classification methods which can be trained on category tagged reports and can automatically classify reports which does not carry any category information during submission.

4.1.3 Location Determination

Location of the event is one of the most important contextual information needed for analysis. Mobile application automatically captures GPS coordinates and Cell Info, GPS is high in accuracy (10-100 meter) and coordinates are directly usable. In absence of GPS, Cell Info is converted in location coordinates with the help of war-driving databases such as OpenCellID, Google's geocoding API etc [9]. Events submitted using SMS and web-based form only have textual location information embedded in the report which too varies in accuracy i.e. locality name (medium accuracy), to city name (low accuracy).

Since, automatic location capturing is not possible in some of submission modes, we have prescribed that users add the location in the end of the text separated by @ as the delimiter, many participants do not follow this convention. Often the location name is embedded into the text which needs to be extracted. If delimiter such as @ is not there in the

Event ID	Message	Tags	Latitude	Longitude	Cell Tower ID	Category	Service Provider	Location	Event Date
1	large voltage fluctuations hope appliances dont get affected	problem voltage	28.54398	77.2002	6621291	neighbour--hood issues	10-404-31031	iit Delhi	7/2/12 12:58 AM

Table 3: Row Showing Event Stored in Database

text, we parse the message and use popular location suffixes such as 'nagar', 'chowk', 'cross' etc to estimate the location name. We have used a location dictionary to automatically learn location suffixes which can help in location extraction. Extracted location name could be a locality name or a city name, we used Google's geocoding API to convert the location names to approximate geo-coordinates but there may exist some location names which can not be geocoded by Google's geocoding service i.e. some lesser known locality names which does not exists on Google Maps. For such names, we used a dictionary of 963 Indian cities and towns to translate the location names to their corresponding city names. Those city names were in turn fed to the Google's geocoding service to retrieve approximate geo-coordinates. These geo-coordinates will help us in populating the location column wherever possible as shown in Table 3.

4.2 Data Analysis

As part of analysis, we analyzed the distribution of different event categories across different states for finding major patterns in the submitted data. Figure 4 shows the break up of different event categories across whole country; most of events (43%) are about civil issues.

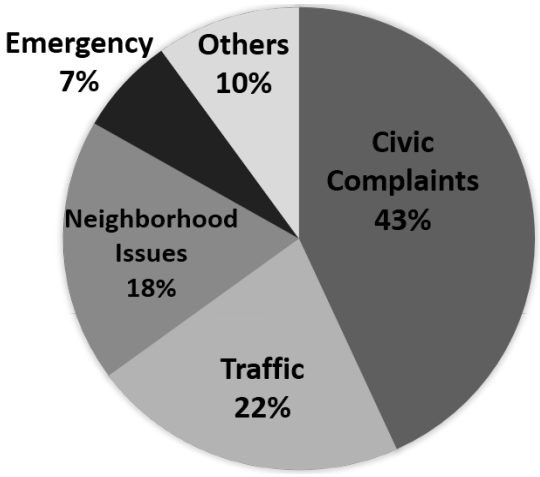

Figure 4: Category-wise distribution of events submitted from all over India

Figure 5 shows all the reported events on the map which depicts that we had representation from many different states in India.

The three states from which we have received the maximum data are Uttrakhand (105), Delhi (95) and Tamilnadu

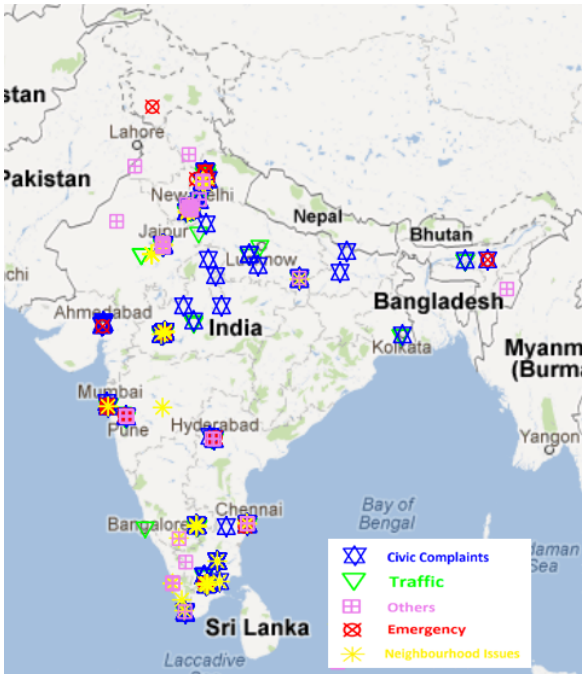

Figure 5: Distribution of the submitted events from across India

state (80). Figure 6 presents the break up of the event categories from Uttarakhand. In Uttarakhand, 54.2% of the events reported are about Civic Complaints, followed by Traffic and emergency accounting 18% and 10.4% respectively. Neighborhood Issues and other events account for 8.5% each. In Figure 7, all the reported events from Uttarakhand are mapped. We have seen a large variance in distribution of various events categories across different states, for instance, maximum reports (54%) from Uttrakhand contains civil complaints; in Delhi, traffic events (42%) and from Tamilnadu, both civic and neighborhood issues (80%) were maximum.

Further, we analyzed the text in specific event categories to find different pattern in the data. Figure 3a shows the tag cloud of textual reports submitted for traffic related events in Delhi. Most of the event deals in reporting of high traffic, jam, congestion, water logging etc. Similarly, Figure 3a shows the tag cloud of report submitted for neighborhood issues in Tamilnadu and most of them deals with water logging, waster, drainage, unauthorized parking etc.

(a) Traffic Issues in Delhi (b) Neighborhood Issues in Tamilnadu (c) Civic Complaints in India

Figure 3: Tag Clouds of some of major pattern found in data collection

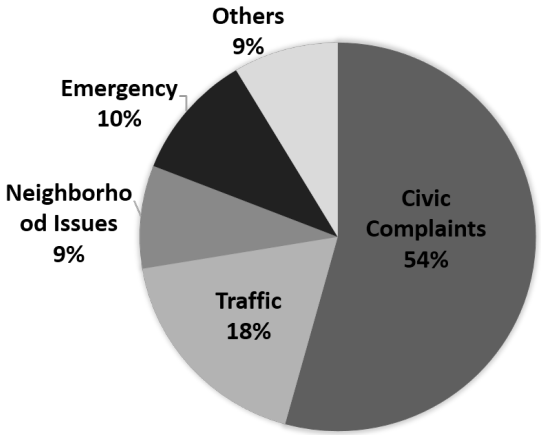

Figure 6: Category-wise distribution of events submitted from Uttrakhand (highest contributor among states)

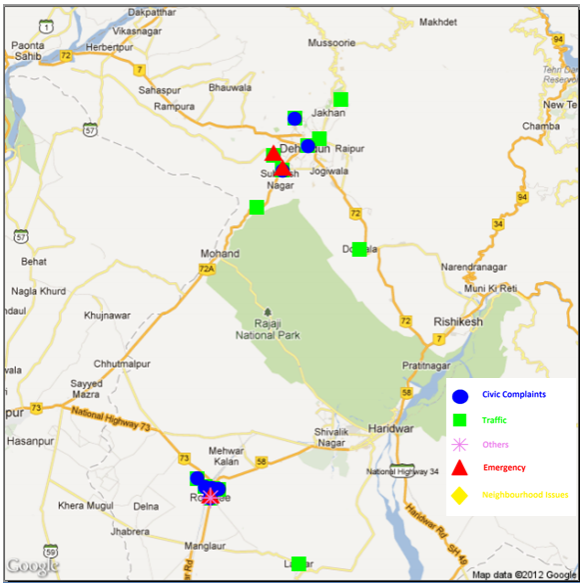

Figure 7: Distribution of the submitted events from across Uttarakhand

Civic Complaints are a major concern in all the three states as well as all over India and most of them are about heavy rain, water logging, bad roads etc as shown in Figure 3c.

A total of 435 users have registered so far in the NUC challenge but many of them are not frequent in reporting events. Obviously, several participants are not being sufficiently engaged. As data collection move forward, our efforts will be focussed on engaging the participants better. One way of engaging will be to share reports generated from the data. It is expected that the quality of the data will improve as the participants become more mature and receive more feedback from the reports generated from the data collected so far. These reports will be shared through the website and will be viewable by all the registered participants.

5. DISCUSSION

We present a system to sense various city's events or day-to-day problems using citizens's participation termed as *crowdsensing*. Crowdsensing can be extremely useful for countries like India where due to large scale, it is very hard for city authorities to keep track of different events and taking of prompt action. In the proposed system, we provide different modes to report events which are aggregated at a single server. We have deployed the system in real-world and received event reports from many different states and cities in India. The data collection exercise has been fairly successful so far, we have over 400 registered participants with several of them being very active in submitting events. Our main goal was to execute the crowdsensing experiment and show its benefits with respect to traditional sensors in terms of cost saving especially in resource constrained setting like India. Going forward, our aim will be to work on engaging with participants better. For now, we are only sharing reports with all participants, we also plan to share feedback on these reports so that noise in data could be minimized and data quality will improve.

The data analysis in this paper is preliminary, as we collect more data using crowdsensing, we hope to answer some of following questions (a) what kinds of events is the community interested in sensing? (b) Can we extract trends in space or time dimension? (c) What modalities (text, voice, audio, photographs) are used and their correlation with sensed events? (d) To what extent can we measure the

real phenomenon from the sensed data?

Some of the above questions are long-term and will requires a good amount of data which we hope to collect in due course of time with the help of participants.

6. ACKNOWLEDGMENTS

We would like to thank all the student participants who contributed in this crowdsensing effort by sending event reports.

7. REFERENCES

[1] R. K. Ganti, and H. Lei, Mobile Crowdsensing: Current State and Future Challenges, *Communications of ACM*, 2010.

[2] O. Stewart, J. M. Huerta, M. Sader: Designing crowdsourcing community for the enterprise. *KDD Workshop on Human Computation*, 2009: 50-53.

[3] L. V. Subramaniam, S. Roy, T. A. Faruquie, S. Negi, A survey of types of text noise and techniques to handle noisy text, *Workshop on Analytics for Noisy Unstructured Text Data*, 2009: 115-122.

[4] Goldman, Jeffrey, et al. "Participatory Sensing: A citizen-powered approach to illuminating the patterns that shape our world." Foresight & Governance Project, White Paper (2009).

[5] Yadav, K., Kumaraguru, P., Goyal, A., Gupta, A., and Naik, V. Smsassassin : Crowdsourcing driven mobile-based system for sms spam filtering. ACM HotMobile' 2011.

[6] Roy, Swaroop, et al. "Wireless across road: RF based road traffic congestion detection." Communication Systems and Networks (COMSNETS), 2011 Third International Conference on. IEEE, 2011.

[7] Eriksson, Jakob, et al. "The pothole patrol: using a mobile sensor network for road surface monitoring." ACM MobiSys. 2008.

[8] Min Mun, Sasank Reddy, Katie Shilton, Nathan Yau, Jeff Burke, Deborah Estrin, Mark Hansen, Eric Howard, Ruth West, and PÃl'ter Boda. 2009. PEIR, the personal environmental impact report, as a platform for participatory sensing systems research. In Proceedings of the 7th international conference on Mobile systems, applications, and services (MobiSys '09).

[9] Yadav, Kuldeep; Naik, Vinayak; Singh, Amarjeet; Singh, Pushpendra; Chandra, Umesh; , "Low Energy and Sufficiently Accurate Localization for Non-smartphones," Mobile Data Management (MDM), 2012 IEEE 13th International Conference on , vol., no., pp.212-221.

[10] Mohan, Prashanth, Venkata N. Padmanabhan, and Ramachandran Ramjee. "Nericell: rich monitoring of road and traffic conditions using mobile smartphones." Proceedings of the 6th ACM conference on Embedded network sensor systems. ACM, 2008.