`

**A**
**Dissertation**
**On**

# "Crowdsourcing On Mobile Application"

**Submitted in Partial fulfillment of the requirement**
**For the award of Degree of**

**MASTER OF TECHNOLOGY**
**in**
**Computer Technology and Application**

**SUBMITTED BY**

**ADARSH PANDEY**
**University Roll No: 04/CTA/2010**

**Under the Guidance of:**

**Mr. R.K. Yadav**

**Assistant Professor**

**DEPARTMENT OF COMPUTER ENGINEERING**
**DELHI TECHNOLOGICAL UNIVERSITY,DELHI**
**2010-2012**

`

# CERTIFICATE

This is to certify that the work contained in this dissertation entitled "**CROWDSOURCING ON MOBILE APPLICATION**" submitted in the partial fulfillment, for the award for the degree of M.TECH in Computer Technology and Applications at **DELHI TECHNOLOGICAL UNIVERSITY** by **ADARSH PANDEY, Roll No. 04/CTA/10,** is carried out by him under my supervision. This matter embodied in this project work has not been submitted earlier for the award of any degree or diploma in any university/institution to the best of our knowledge and belief.

**(R.K. YADAV)**
 **Project Guide**
 **Assistant Professor**
 **Department of Computer Engineering**
 **Delhi Technological University**

`

# ACKNOWLEDGEMENT

First of all, let me thank the almighty god and my parents who are the most graceful and merciful for their blessing that contributed to the successful completion of this project.

I feel privileged to offer sincere thanks and deep sense of gratitude to **Mr. R.K. YADAV**, project guide for expressing his confidence in me by letting me work on a project of this magnitude and using the latest technologies and providing their support, help & encouragement in implementing this project.

I would like to take this opportunity to express the profound sense of gratitude and respect to all those who helped us throughout the duration of this project. DELHI TECHNOLOGICAL UNIVERSITY, in particular has been the source of inspiration, I acknowledge the effort of those who have contributed significantly to this project.

**ADARSH PANDEY**
**(University Roll No.: 04/CTA/2010)**

`

# ABSTRACT

As crowd-sourcing is becoming popular for problem solving and completing a task, it is now very important to use this concept in an advance manner. It can also be used as a distributed and vast source of information. This concept is now evolving in world of mobile systems. This will be a little different from that of computer systems., we have discussed some new technologies and challenges before us to implement these advancements in crowd-sourcing.

In this thesis, we have emphasized on cheat-detection techniques which can be implemented on various types of crowd-sourcing applications. Some applications requires accurate or close to accurate answers while other applications are seeking for just opinion of user. The cheat detection mechanisms in both the cases will be different. As in latter case, we don't have any set of correct solutions.

We have discussed the need of multimedia data management. As, we are supposed to transfer i.e. send and receive multimedia files, we requires a standard interface which is capable to allow data directly from websites(WAP in case of mobiles), resolve description of unknown resources and integrate the data sets accordingly. The problem of ambiguity is also considered to be an important part of challenges in crowd-sourcing mechanisms. In some applications, use of handwriting and sound recognition techniques may play an important role to improve the effectiveness of applications where handwriting or sound clips are going to fall under information category.

# TABLE OF CONTENTS

`

# List of Figures

`

# List of Tables

# Chapter 1: Introduction

The main topic of research of this master thesis is the evolution of crowd-sourcing from computer systems to mobile systems. It can be seen as the latest trend as the more active role of consumers or users in the innovation process and thus observing a wider range of diversity.

With the exponential growth of internet in recent years, the huge workspace and opportunities are developed. To meet this growth, a number of ways and techniques are acquired in last decade. Crowd-sourcing is one of the most famous and effective approach to meet the desired rate of internet growth.

Crowd-sourcing was introduced in 2006, by Jeff Howe. Basically, crowd-sourcing is a blend of "crowd" and "outsourcing". It can be explained philosophically as "borrowing brain from the crowd". The term can be defined as "the act of getting a job from a professional agent and passing it(outsource) on to a large group of people(crowd)". An important difference is that the entrepreneur doesn't know who has completed his task. The people from crowd don't form any group. They are anonymous members of a crowd. Another big difference is that in traditional outsourcing approach, the worker is assigned a task but in crowd-sourcing, worker gets to choose on which task he wants to work on.

Crowd-sourcing approach was successfully used commercially in Murk or Microworkers. By commercial use of crowd-sourcing, work can be done at a great pace by distributing it to a large group. But, the results may not be very reliable.

1

`

Now, with the high usage of advanced mobile systems, such as tablets and smart phones, a large set of information, it can be users information or the activities log etc are available. This information can be analyzed or measured by mobile device itself or can be passed to a dedicated mechanism depending upon the user preferences. Crowd-Sourcing has brought a big change in problem solving mechanism in world of computers and now it has started changing the use of mobile devices and invented a new dimension of analyzing the patterns of mobility, log of activities etc.

## 1.1 Motivation

Now companies have started outsourcing activities to volunteers via making an open call on the internet. In return, these volunteers or users get rewards or some kind of money. This money can be air-time as well.

But, the accessibility of desktop is not that easy for most of the peoples in all over the world. This makes a big part of crowd unable to participate in crowd-sourcing. In contrast to this, cell phone penetration is very high. In India alone, the percentage of mobile users is much higher than that of desktop users. Many of the mobiles are simple phones having a simple and common feature of accessing the web. Also, the cost of internet on mobile phone is well affordable. This makes the mobile internet a cost effective approach to perform crowd-sourcing upon.

With the evolution of this crowd-sourcing approach towards mobile systems, the implementation is needed to be reviewed. As the implementation mechanism of crowd-sourcing in computer systems is well organized and managed, it is now the need of time to do the same for mobile systems.

`

## 1.2 Research Objective

This thesis reports on our approach to implement crowd-sourcing on mobile systems. With respect to this, we have discussed the various implementation issues. We have focused on different cheat detection techniques for different types of crowd-sourcing approaches. The problem statement is: **"To propose effective approaches for detecting the cheats and frauds in various mobile based crowd-sourcing services while discussing the various challenges in implementation of crowd-sourcing for mobile systems".**

## 1.3 Related Work

In this section, we will discuss about the research work and projects which has been based on idea of crowd-sourcing. As the concept of crowd-sourcing is introduced in last decade only, there are very few successful and big projects which were successful.

Crowd-sourcing in the world of mobile is even more recent idea and is only implemented on very few ideas. We are going to discuss some of these, both crowd-sourcing on computer system and on mobile system, ideas and projects which were implemented and were successful.

*A, MTurk Platform*

Amazon Mechanical Turk was launched in 2005 end and is in beta phase in 2011 December[1]. Mturk differentiates between two kinds of roles in this mechanism. There are two accounts maintained. One is for the Requesters, who submits a task and other one is Workers, who works on those tasks. These tasks are called *Human Intelligence Tasks* (HIT). Workers choose an appropriate task and locks the particular

`

HIT for himself. Once a HIT is locked for a worker, it cannot be assigned or locked by any other worker.

Each HIT is paid in range of $0.01 to few dollars. This amount depends upon the difficulty level and time completion of that HIT.

*B. Microworkers Platform*

This platform was launched in 2009 and is similar to Mturk. There are some minute differences in this platform. One is that in Microworkers[1], all the users have only one account and with one account, the user can login as both *employer* and *worker*. No US based accounts are needed to get payments as the payment service is online and is done through various online payment mechanisms like *PayPal*. This gives an inspiration for international user to work on this platform. Similar to Mturk's HIT, Microworkers has *jobs* and *campaigns*. In contrast to Mturk, in Microworkers jobs are predefined into different categories with different pay scheme depending upon the time consumption and complexity. Jobs are paid in range of 0.10 to few dollars.

*C. Mobile Image Search*

A study survey was done in Japan based on the concept of crowd-sourcing[3]. They developed a mobile social search application. With the assistance of local people or crowd via social mediums, the application helps foreigners in Japan by replying their image based queries in a timely fashion. They ran a field practical for 45 days with over 50 participants and found that the mobile crowd-sourcing model result in a

4

`

reliable performance in terms of response speed and quantity. 50% of all the requests were answered within 10mins and 75% were answered in 30 mins and on an average every query had 4.2 replies.

This kind of application is very useful for tourists and travelers who are not able to understand of local language of a city. Ordinary digital translators may not be useful in this case  because the user is unable to input the alphabets/words of the language they see for e.g. Chinese or Japanese language.  Image based text translation systems were developed but haven't yet proved very useful for the purpose.  To resolve this problem, UbiAsk a social media based crowd-sourcing application built on networking infrastructure.  Instead of using machine algorithms, use of power of general crowd in cloud via social means to solve typical computational problems like image recognition and translation of text were implemented. To evaluate this system, a controlled experiment was conducted.  In this experiment, main focus was on quantity and response time.

### D. Txteagle

It is a system which enables ordinary people to earn a little income in terms of money by completing tasks on cell phones for a employer who pays them either mobile money or air time.

This system was launched in Kenya and Rwanda. Collaboration was made with the cell phone service providers. The tasks were translation, survey and transcription. This study was done by including high school students, local security guards and taxi drivers of Nairobi.

`

In contrast of other services like Mturk, which are passive, Txteagle[4] service is active in behavior. This system is built to actively select the fit task for the user. The task difficulty for a particular user is customized. The user's response to a task derives the system to learn the areas in which the user is interested and expert. This helps the user to assign the right task to right user.

## 1.4 Scope of the work

In this thesis, some approaches for cheat-detection and recognition of fraudulent cases are proposed. Also, other issues and challenges in implementation of crowd-sourcing in a mobile system are also discussed. We have proposed some new approaches which can be used for detecting the cheats and frauds performed by user in order of getting quick or easy money/reward.

Our proposed approaches are capable of identifying whether the user is submitting the solutions/answers sincerely or he is replying absurdly with the intention of receiving reward money. These approaches basically work on specific types of crowd-sourcing. These types are crowd-voting and crowd-journalism. In crowd-journalism, a trust factor is introduced which keeps record of past uploads/submission by a user and this factor indicated the probability of correctness of current solution/submission. In crowd-voting, a new approach is been introduced which determines the sincerity and intentions of user who is submitting the solutions/answers.

## 1.5 Organization of the thesis

In this chapter, we have explained one of the recent trends i.e. crowd-sourcing, both in computer systems and in mobile systems and also we have discussed the challenges of

6

`

implementing crowd-sourcing in mobile system which serves as the motivation for the work reported in this thesis. Furthermore we have also outlined the specific objective of our research and related research work that has occurred in the past.

Chapter 2 provides an overview of related works in the field of crowd-sourcing along with the fundamentals of crowd-sourcing and some already existing approaches which can be helpful in building a particular crowd-sourcing scheme. Finally, we present a study that shows the need of our methodologies.

Chapter 3 introduces our approach to detect the cheat and fraud cases along with trust factor maintenance which can be useful in verifying the correctness or truthfulness of solutions submitted by a user. It also includes the limitation of earlier approach towards cheat-detection phenomena.

Chapter 4 presents the performance study conducted on the proposed approaches. It also describes the environmental setup and technical description for implementing the proposed approaches. Analysis and outcome of our study is presented in this chapter.


 Finally, Chapter 5 concludes the thesis and gives some suggestions for future work.

# Chapter 2: Literature Review

## 2.1 Crowd-sourcing Basic Concepts

### 2.1.1 What is Crowd-sourcing?

Crowd-sourcing was introduced in 2006, by Jeff Howe. Basically, crowdsourcing is a blend of "crowd" and "outsourcing". It can be explained philosophically as "borrowing brain from the crowd". The term can be defined as "the act of getting a job from a professional agent and passing it(outsource) on to a large group of people(crowd)". An important difference is that the entrepreneur doesn't know who has completed his task. The people from crowd don't form any group. They are anonymous members of a crowd.
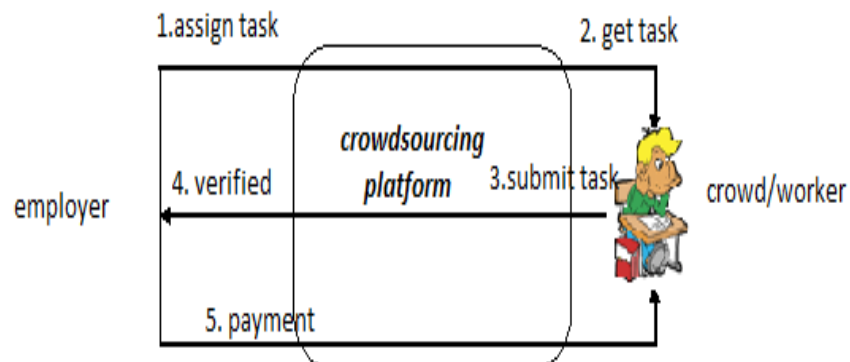
Figure 2.1 Crowd-sourcing system

`

### .2.1.2 Crowd-sourcing Background, Research and Evolution

It is very clear from the definition that crowd-sourcing is somewhat same as outsourcing yet there are some differences in both the terms. Our study is related to the crowd-sourcing which is carried out on mobile systems. There are some projects in past which have discussed the scope and importance of crowd-sourcing on mobile systems.

Smart phones have brought a huge change in the scope of participation of a mobile phone in field of crowd-sourcing. The project "Smart phones as sensors" present a method of rapid development of crowd sourcing applications . The framework includes features like writing scripts, which a user can write in accordance of his choice and requirements. Some important steps are also discussed in engineering of such kind of applications.

A survey and classification was performed and it shows an over view of taxonomy of crowd-sourcing and human computation tasks[5]. According to this research, there are many dimensions of social networking systems. This research shows the development in field of crowd-sourcing and influence of social computing on it. A classification is shown on crowd-sourcing systems, which explains the motivation to take part in crowd-sourcing activities, explaining what kinds of contracts requester, worker and a computer enters.

Web crowd-sourcing markets are very unlikely to accessible by workers who are considered as economically poor or those who are residing at lower level of economic pyramid. Also, those who are illiterate in field of computer education are unaware of even this crowd-sourcing term. Many big firms and industries have understood this

9

`

and now they are shifting their focus towards mobile implementation of crowd-sourcing

A mobile phone based crowd-sourcing platform was presented with the name of "MobileWorks"[6] which is intended to provide a way of employment for developing world users. MobileWorks has basically three main components. First one is a preprocessing work that takes scanned paper documents for the generation of many small OCR out of these papers. Second stage is to construct a interface enabling workers to perform given OCR tasks. Third stage is a post-processing stage which reassembles completed OCR and ensures the quality of submissions.

A similar study was conducted in Japan based on crowd-sourcing on mobile phones. In this system, a user can put requests via multiple channels that use same API. This request is pushed to a community of local experts as an open call via different social media platforms such as twitter, facebook etc. The crowd-sourced result data is returned to requesters. This results in an information pool which contributes in public good.

### 2.1.3 Introduction to Crowd-sourcing fundamentals

Mobile systems are very different from computers. Although they are considered as compact form of computer systems now a days, but still there are some major differences which cannot be ignored while talking about the implementation of crowd-sourcing on mobile systems.

The browsers of mobile phones are very much less capable if compared to that of a computer's browser. And also, the capabilities of these browsers vary from phone to phone.

10

`
Besides the limitation of a mobile phone, crowd-sourcing itself has some challenges which need to be deal with. We are discussing some major fundamental issues which are must to well implement the crowd-sourcing approach to a widespread level.

### 2.1.3.1 Cheat Detection

Crowd-sourcing is attractive because it is scalable, cheap and gives us easy access to information that would be very expensive or difficult to collect. And, as crowd-sourced systems are now becoming a important part of computer world, they will definitely become targets for malice and mischief.

Even if the user or worker doesn't intent to cheat, wrong solution or information can be submitted by mistake. These cases are also needed to be taken care of. So in general manner, we can say that some users submit wrong results to get more income and others make mistake unintentionally. Here, we will denote them both as cheaters. Sometimes small mistake in results of a worker or user can be tolerated as it may not cause big problem to our overall solution, but sometimes it is very important to get accurate response. Some general solutions have been discussed in past. Two easy to understand and simple methods are "The majority decision approach" [7] and "The control group approach" [7]. These methods are suitable for the cases in which many users are submitting the answer of same problem and the answer is accurate and does not vary people to people.

There are some other cases where solutions or answers may vary user to user. These cases are very difficult to monitor. It is not easy to say that the user is submitting answers responsibly or he is just answering vaguely to maximize his income. In these

11

`

cases, the work or questions may be repeated after a while. This will record the behavior of the user that he is submitting the same answer to that question each and every time or not. This will reveal the nature of response of the user.

**2.1.3.2 Multimedia Database Management**

It has been problematic technically to interoperate multimedia files, which includes the use of various formats for same data, of different data models and transaction processing methods. These all problems lead to what generally known as representational heterogeneity. Sometimes these issues are handled separately for different types of file systems. A uniform data system *extensible Meta system* M(DM) [8] were proposed. In this system, the definitions of database schemas, database models can be handled uniformly.

Another approach was proposed for audio files. The spectral properties of files of audio formats were analyzed and the acoustic features which were based on spectral properties were proposed and harmonic enhancement was used to classify audio. This was known as multi model HMM [9].

Another recent open source database for mobile devices was introduced which is known as WURFL [10]. It maps a cell phone to its capabilities. It makes easy to take http request and recognize whether it comes from the corresponding mobile browser without maintaining huge list of user agents of mobile browsers.

**2.1.3.3 Handling Ambiguity**

Recognition technologies like gesture, handwriting and speech recognition techniques have made a huge workspace in recent years. More support for natural forms of communication and recognition can make a computer more like human expert. It will

`

become more accessible. These natural interfaces are used generally where mouse and keyboard are not available. However, these recognizers can generate errors as these are error prone. The user's intention of input can be different from what is understood by recognizer. This can cause problems in terms of performances and confusion of user and also it will result in uncertain interactive statements. Architecture [11] was designed to handle such kind of ambiguities.

## 2.2 Cheat detection methodologies

### 2.2.1 Problem Definition

Crowd-sourcing is attractive because it is scalable and cheap and provides us access to information that would otherwise be very difficult to collect. The crowd when equipped with modern Smartphone's connectivity and sensing capabilities is a huge source of useful services. And hence, crowd-sourcing systems have become a much integral part of our computer world and therefore it will undoubtedly become the target for mischief and malice.

In order to earn easy money, some users submit incorrect results by completing as many tasks as possible in given span of time. From now on, we will assume all of them as cheaters for submitting the incorrect results or submitting their vote insincerely. Such attacks or cheats are particularly worrisome as they can be launched by random users and are very difficult to detect. Faced with such emerging threat, we should reconsider the question rose i.e. "how can we trust data contributed by a crowd of anonymous internet workers?"

13

`

**2.2.2 Existing approaches**

There are some pre-existing approaches. Some of the well defined approaches are being explained as follows:

1) *The Majority Decision Approach*
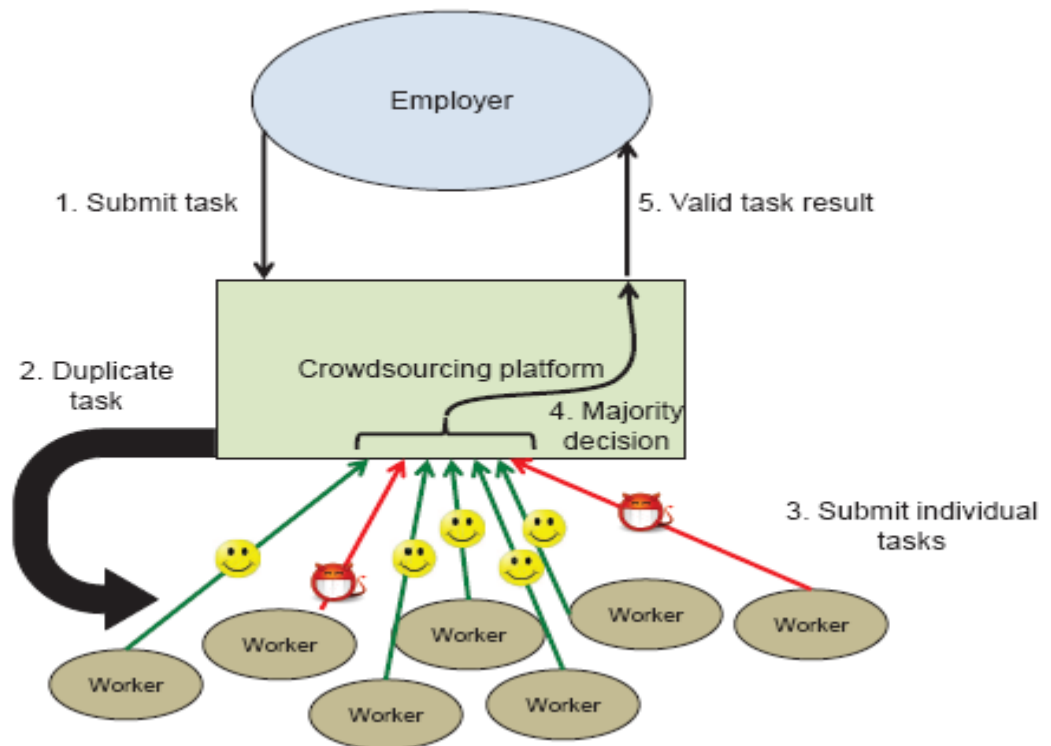


Figure 2.2

A simple approach to eliminate incorrect results is to use a majority decision (MD)[7]. This means the same task is given to i different workers and the results are compared. The result which most of the workers submitted is assumed to be correct. Figure 2 depicts the work flow of a campaign using the MD approach. Similar to a normal campaign, the employer submits his task to the crowd-sourcing platform (1). However, the task is not only done by one worker, but the  rowdsourcing platform

14

`

duplicates the task several times and offers the same task to a large number of workers (2). Each of these workers submits his individual task result (3), which might be correct or incorrect. The crowdsourcing platform performs a majority decision to validate the correct result (4) and returns it to the employer (5). In this approach each worker submitting a result is paid. As an example application of the MD approach, think of an relevance evaluation of an article, where 100 workers have to judge whether the text is related to a certain topic and 95 workers rate it off-topic. Even if there are some cheaters among the workers, the article is with a high probability off-topic.

2) *The Control Group Approach*



Figure2.3

15

`
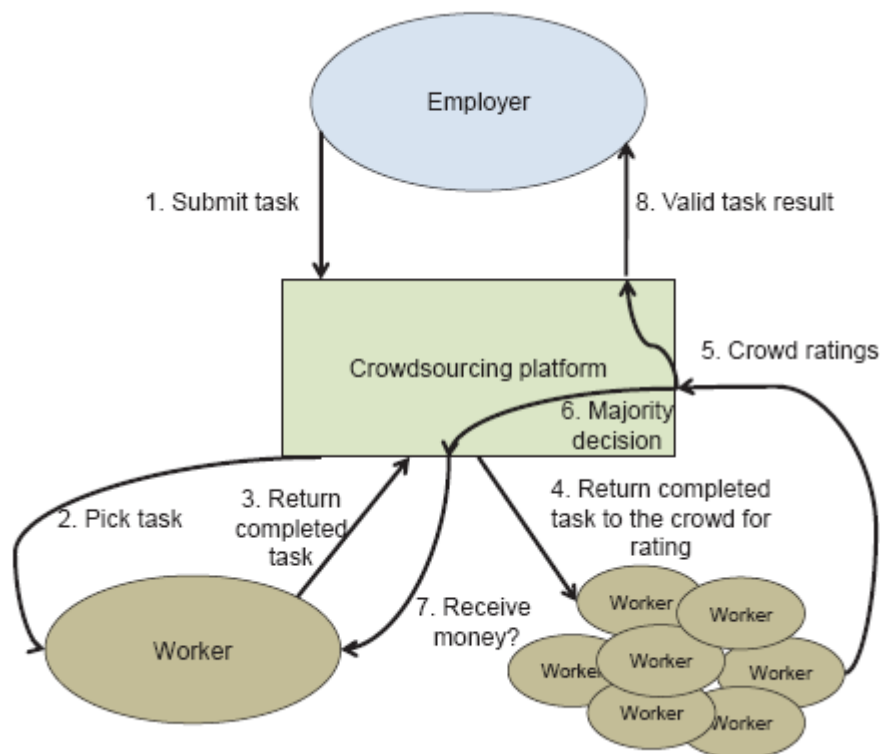
The  second approach is based on a control group (CG). A single worker works on a main task and a control group consisting of j other workers re-checks the result, whether it is valid or not. A task is considered to be valid, if the majority of the control group decides the task is correctly done. An important point of this approach is that the main task and the"re-check" task are assumed to have different costs. Usually the main task is expensive, like writing an article to given keywords, while the control task is cheaper. The work flow of a CG campaign is shown in Figure 3. The employer submits the main task to the crowdsourcing platform (1) and the task is chosen by a worker (2), who submits the required task result (3). This result is not directly returned to the employer, but the crowdsourcing platform generates a new campaign for validating this result. Therefore, the result of the main task is given to a group of workers, who rate it according to given criteria (4). The ratings of the different workers are returned to the crowdsourcing platform (5), which calculates the final rating of the main task using a majority decision (6). This is necessary, because the workers in the control group may be cheating and submit wrong ratings. If the main task is rated valid, the main worker is paid (7) and the result is returned to the employer (8). A possible application of this approach is a task where a worker has  to write a long article including some given keywords. This is a quite expensive task, as it is time intensive and the worker has to be creative. As the worker submits the completed article, a new campaign is created during which 100 workers have to judge whether the article matches the initial keywords or not. This task is simple and therefore, less paid. If enough workers submit that the article matches the given keywords, it can be assumed to be valid.

`

### 2.2.3 Motivation for new approach

The approaches discussed above can be useful in some crowd-sourcing systems where users or workers are assigned a task and they are supposed to submit a solution to that particular question. This solution is defined and will remain same for all users i.e. will not depend on user's state or his behavior. In this case, the solution can be verified by comparing the result submitted by a user with a standard solution. Both above explained approaches can be implemented to verify the correctness of the solution.

But, there are some other types of crowd-sourcing systems. These are generally termed as "crowd-voting" or "crowd-journalism". In these systems, the solution/answer will vary from user to user or the data provided by a user may not be compared with a standard set of solution.

To evaluate the sincerity and intention of user in these type of cases, we need new approaches. Requirement of these new approaches are the driven force of this thesis.

## 2.3 Summary

In this chapter, we have surveyed the broad area of crowd-sourcing with a specific focus on cheat detection approaches. We have also explained various type of approaches which exist for cheat detection. The problem definition of cheats detection and observation of such users are explored in detail. We found that all the earlier approaches are not suitable for all crowd-sourcing systems. So we have proposed a new effective approach for cheat detection in the next chapter.

`

# Chapter 3: Proposed Approaches for cheat-detection

The objective of this chapter is to introduce our approaches to detecting the cheat or frauds committed by a user. Also, the behavior of a user can be determined by these approaches. In particular, we will suggest our approach to give a cheat detection mechanism for cases which were unable to detect using previous approaches.

The chapter is organized as follows. Section 3.1 gives the detailed description of the weakness in previous approaches with. Section 3.2 provides the proposed approaches. Finally section 3.3 summarizes the chapter.


## 3.1 Traditional Approach Weakness

The approaches discussed above can be useful in some crowd-sourcing systems where users or workers are assigned a task and they are supposed to submit a solution to that particular question.

But, there are some other types of crowd-sourcing systems. These are generally termed as "crowd-voting" or "crowd-journalism". In these systems, the solution/answer will vary from user to user or the data provided by a user may not be compared with a standard set of solution. For example, if we are implementing a crowd-voting system, then no expert can state that user's vote is correct or not as it is the personal decision of every individual voter. And, that vote of a particular user cannot be compared with the majority decision for verifying the truthiness.

Now consider a data-upload system. In this type of crowd-sourcing mechanism, the data uploaded by a user for, let's say, news crowd-sourcing cannot be

`

compared with other uploads or cannot be stated as correct or incorrect by any user. So, in all these cases the existing approaches will not be able to work well.

## 3.2 Proposed Approaches

We have reviewed some approaches like "Majority Decision" approach and "The Control Group" approach. But they were not good enough for all type of crowd-sourcing systems. We have given two new approaches which can be useful for some crowd-sourcing systems which were not handled well by existing approaches.

### 3.2.1 VERITY Index

This approach is applicable for systems where user has to upload data. In this kind of crowd-sourcing system, user will act as a source of information. This can be understood as crowd-journalism. In crowd-journalism, a person from crowd acts as a journalist and provides news from anywhere and anytime. Now a day, news channels are very much interested in adopting crowd-journalism. Very soon, they are going to implement this kind of news collection mechanism.

We have designed a cheat detection system, by which an organization, a news channel in this case, will be able to take action based on the information given by a user or a crowd-journalist. In this system, we are going to maintain a VERITY index for each user. This index will indicate the probability of truthiness of information provided by that user.

Here, we have considered two types of users.

`

### 3.2.1.1 For Paid User

This category of users will include those who have registered to by paying some amount for being a part of crowd-journalism system. These are the people who will be considered as serious workers/users. They will provide information and in return, they will be rewarded. This reward can be money or airtime depending upon the organization. VERITY index of these users will represent the probability of correctness of news provided by these users. The formula of VERITY index for these type of users can be defined as

$$I_{new} = i_{old} + \alpha. S. V$$

$I_{new}$ = new VERITY index

$I_{old}$ = previous VERITY index

$\alpha$    = variation factor

$S$    = sensitivity factor

$V$    = value, 1 or -1

$I_{new}$ is the new VERITY index which is updated after each submission by the user. $I_{old}$ is the previously updated VERITY index for a user. This $I_{old}$ will be used as the input for calculating new VERITY index. "$\alpha$" is defined as variation factor. It is used to determine the variation of VERITY index after each time a user uploads data. This variation factor may be different for individual user depending upon his past record. It can also depend upon organization policies. Sensitivity factor indicates the

20

`

importance of that news for both user and the news channels. So, if the sensitive news is true, the variation in that user's VERITY index must be greater than that of some less sensitive news. So, the value of sensitivity factor will be directly proportional to the importance/sensitivity of the news. The value "V" will be assigned only +1 or -1. +1 for true news and -1 for false news.

Here, we can apply some constraints like the maximum value of VERITY index can be .9

Example: consider the following news crowd-sourcing system:

$I_{initial}$ = .7

$\alpha$ = .6

| S | .4 | .5 | .9 | .2 | .5 | .8 | .4 | .3 | .1 |
|---|----|----|----|----|----|----|----|----|----|
| V | -1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 |

By using these values, the value of $I_{new}$ we will get is as follows:

| $I_{new}$ | .46 | .16 | .70 | .58 | .88 | .4 | .64 | .46 | .52 |
|-----------|-----|-----|-----|-----|-----|----|-----|-----|-----|

`
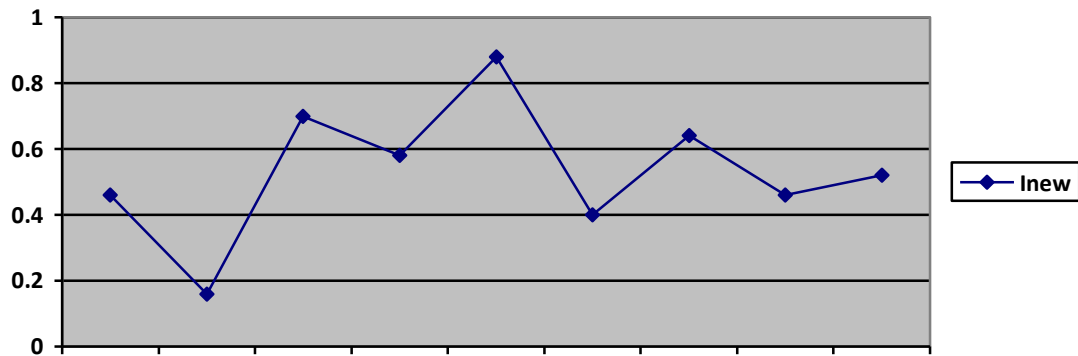


### 3.2.1.2. For Free User

In this class, the users who are opting journalism as their interest or for goodwill of society will be included. They will not pay any amount for the registration but they have to go through authentication process. Because they are free users, they will not get any reward for their news.

The difference in calculating their VERITY index will occur due to the variation factor used in the formula. Though it is obvious that the initial value of VERITY index will be small than that of paid user.

## 3.2.2 Random Repeat Approach

We have been talking about crowd-voting systems. In these systems, the vote of a mass is collected via a crowd-sourcing medium. Sometimes these systems are used to know the public opinion of a product or determining the preferences of users for similar category of items.

`

In these types of crowd-sourcing systems, none of the above discussed approaches or already existing approaches will work because it is the personal choice of a user to vote for something or to choose his preferences. For such kind of system, "Random Repeat Approach" will work. In this approach, one or more than one questions/query will be repeated randomly. If the user is voting or ranking insincerely, then there is a possibility that the ranking done by him for one question may differ the other time when the same question is encountered. This will reveal the insincere behavior of the user. On the other hand, if the user is reading all queries and questions sincerely, then he will input the same choice for a question, no matter how many times the question is encountered.

The probability of verifying a user's behavior correctly by using Random Repeat Approach can be calculated by using following formulas:

$$P = 1-(1/C)^R$$

P: probability of detecting cheat

C: no. of choices for a question

R: no. of repetition of a question

Let us take an example: suppose we have a set of 20 questions, of those 20 questions 2 questions are repeated i.e. 18 questions+ 2 repeated questions. Each of the questions has 3 choices as options. Then, the probability calculation can be done as follows:

`

Sol:

If the user has chosen different options for same question(s), it is clear that he is insincere. Now, if user has chosen same options for same question then probability of identifying the true behavior of user is as follows:

No. of total cases for each question= 3

If the user has chosen option 1 in first attempt, the probability of attempting same option unknowingly i.e.

Probability of false observation of behavior = 1/3

Now, if two questions are repeated then

Probability of false observation of behavior = 1/3 X 1/3

$$= 1/9$$

Probability of cheat detection = 1- 1/9

$$= 8/9$$

So, we can say that using Random Repeat Approach, the probability of detecting cheat is quite high and is accurate enough. It is supposed that not more than 10% of questions should be repeated otherwise user will get careful for this kind of check.

## 3.3 Summary

In this chapter, we have identified the drawbacks in the existing approaches used to detecting the cheat and fraud cases in crowd-sourcing. With respect to this, we have introduced our approach to cheat detection by proposing two new cheat detection approaches for various crowd-sourcing systems in detail with examples.

`

# Chapter 4: Implementation and Experimental Results

## 4.1 Environmental Setup

We have used the following configuration while finding the experimental results

### 4.1.1 Hardware Configuration

| | | |
|---|---|---|
| Processor | : | Intel I3 |
| Processor Speed | : | 2.26GHz |
| Main Storage | : | 4GB RAM |
| Hard Disk Capacity | : | 500GB |
| Monitor | : | HP 14'' Color |

### 4.1.2 Software Configuration

| | | |
|---|---|---|
| Operating System | : | Windows 7 |
| Front end | : | java eclipse and Android |
| Back end | : | Datasets (explained in 4.2) |

## 4.2 Technical Description

We have used java as front end. To save time and for the ease, we have used java eclipse for coding and implementation of our project. The work is done on eclipse version 3.7.2 SDK. It is a eclipse classic variant. The android SDK version is 4.0.3. The API level applied is 15. Java compiler compliance level must be 1.6 or latest level.

`

**4.2.1 Synthetic Datasets**

To do analysis, we have used a synthetic or self-produced datasets. Some questions are taken from online surveys. The purpose of these survey questionnaires is to collect some data samples only. The objective of our thesis is to find the behavior of user. Not his opinion about the products.

The data collected from survey is as follows:

There is a set of 20 questions. Each of the following table shows name of the person who took this survey and 4 out of those 20 questions asked.

**Survey spreadsheet part-1**

| your name | what do you think is the better mobile manufacturing company | what do you think is nokia best at? | the best sound quality is offered by: | which of the following has better touch display? |
|---|---|---|---|---|
| adarsh | nokia | battery backup | nokia | i-phone |
| Anwar Dilawar Shaikh | samsung | battery backup | samsung | nokia |
| Vipin | samsung | robustness | i-phone | i-phone |
| Rajiv Ratn Shah | apple | robustness | sony erricson | i-phone |
| Satyendar Yadav | apple | battery backup | sony erricson | i-phone |

`

| | | | | |
|---|---|---|---|---|
| Kshitiz | samsung | robustness | samsung | i-phone |
| Vikas Kumar | samsung | battery backup | sony erricson | i-phone |
| manvi | samsung | robustness | sony erricson | i-phone |
| ajay | apple | robustness | sony erricson | i-phone |
| Mukul Jain | samsung | robustness | i-phone | samsung |
| Teeja | samsung | robustness | samsung | i-phone |
| Mohd Imran | apple | robustness | i-phone | i-phone |
| Gaurav Chawla | samsung | other | i-phone | i-phone |
| tanu | apple | robustness | i-phone | i-phone |
| Garima Jindal | apple | battery backup | i-phone | i-phone |
| Abhinav Choudhry | other | robustness | sony erricson | samsung |
| Niharika | apple | robustness | sony erricson | i-phone |
| ashutosh | samsung | robustness | i-phone | i-phone |
| Mukul Rawat | apple | robustness | sony erricson | i-phone |
| upasna | apple | battery backup | i-phone | i-phone |
| Abhinav Kumar Dwivedi | samsung | battery backup | nokia | i-phone |
| Srideepa Banerjee | apple | robustness | i-phone | i-phone |

`

| | | | | |
|---|---|---|---|---|
| Jayakumar | apple | battery backup | sony erricson | i-phone |
| kewal | nokia | robustness | sony erricson | i-phone |
| BRAHMRAJ SINGH RAWAT | samsung | battery backup | sony erricson | i-phone |
| Jyoti Swarup | samsung | battery backup | i-phone | samsung |
| sundram guha | nokia | other | sony erricson | samsung |
| Mohit Khanna | apple | other | sony erricson | samsung |
| POOJA ARORA | samsung | robustness | i-phone | i-phone |
| piyush | apple | battery backup | sony erricson | i-phone |

Table 4.1

**Survey spreadsheet part-2**

| What size would you prefer your screen to be ? | What method of interaction do you prefer when using mobile phones ? | What do you consider important when choosing a mobile phone? | How much are you willing to spend on a mobile phone? | excluding voice calls, which feature do you use regularly in your smartphone? |
|---|---|---|---|---|
| Large (3.6 to 4.4) inches | both touchscreen and keypad | brand name | more than 25k | music |
| Medium (2.5 to 3.5) inches | physical button | functionality | 10-15k | game |
| Large (3.6 to 4.4) inches | touch screen | brand name | 15-25k | browsing |
| Large (3.6 to 4.4) inches | touch screen | application store | more than 25k | browsing |
| Medium (2.5 to 3.5) inches | touch screen | look and feel | 5-10k | music |

`

| | | | | |
|---|---|---|---|---|
| Medium (2.5 to 3.5) inches | physical button | functionality | 5-10k | browsing |
| Large (3.6 to 4.4) inches | both touchscreen and keypad | application store | more than 25k | browsing |
| Large (3.6 to 4.4) inches | both touchscreen and keypad | functionality | 15-25k | game |
| size is not a criterion | both touchscreen and keypad | functionality | 15-25k | browsing |
| Large (3.6 to 4.4) inches | touch screen | brand name | 15-25k | game |
| Large (3.6 to 4.4) inches | touch screen | application store | 10-15k | music |
| Large (3.6 to 4.4) inches | both touchscreen and keypad | functionality | 5-10k | browsing |
| Medium (2.5 to 3.5) inches | touch screen | brand name | 10-15k | browsing |
| Medium (2.5 to 3.5) inches | both touchscreen and keypad | functionality | 5-10k | music |
| Large (3.6 to 4.4) inches | both touchscreen and keypad | functionality | more than 25k | browsing |
| Medium (2.5 to 3.5) inches | physical button | functionality | 5-10k | music |
| Large (3.6 to 4.4) inches | touch screen | look and feel | more than 25k | texting |
| Medium (2.5 to 3.5) inches | touch screen | functionality | 10-15k | browsing |
| Large (3.6 to 4.4) inches | touch screen | functionality | 15-25k | browsing |
| Large (3.6 to 4.4) inches | touch screen | functionality | 15-25k | browsing |
| Large (3.6 to 4.4) inches | both touchscreen and keypad | brand name | 5-10k | browsing |
| Medium (2.5 to 3.5) inches | touch screen | brand name | 10-15k | texting |

29

`

| | | | | |
|---|---|---|---|---|
| Medium (2.5 to 3.5) inches | keypad | functionality | 15-25k | game |
| Large (3.6 to 4.4) inches | both touchscreen and keypad | functionality | 15-25k | browsing |
| Medium (2.5 to 3.5) inches | touch screen | functionality | 10-15k | browsing |
| Medium (2.5 to 3.5) inches | touch screen | functionality | 5-10k | texting |
| Large (3.6 to 4.4) inches | both touchscreen and keypad | functionality | 15-25k | music |
| Medium (2.5 to 3.5) inches | both touchscreen and keypad | brand name | 5-10k | music |
| Medium (2.5 to 3.5) inches | touch screen | functionality | 5-10k | music |
| Medium (2.5 to 3.5) inches | touch screen | functionality | 15-25k | music |

Table 4.2

**Survey spreadsheet part-3**

| Which provider do you currently have a contract or pay as you go with? | Which of these is your regular in use feature? | What is your primary purpose for using internet on your mobile phone? | What would you describe as the main benefits of using internet on your mobile phone? | What would you describe as the main limitation of using internet on your mobile phone? |
|---|---|---|---|---|
| vodafone | browsing | to social network | Its accessible everywhere | The websites do not look as good on mobile |
| tata | gaming | other | Its accessible everywhere | The websites do not look as good on mobile |
| other | browsing | to social network | It allows me to use online utilities when I am out | The websites do not look as good on mobile |
| other | browsing | to access e-mail | Its accessible everywhere | It is only good for limited use |
| reliance | texting | to social network | It is a way to pass my time (eg transport, queue,etc) | It is very slow |

`

| | | | | |
|---|---|---|---|---|
| reliance | texting | to read news | It is a way to pass my time (eg transport, queue,etc) | The websites do not look as good on mobile |
| vodafone | browsing | to access e-mail | I can search for information that I need urgently | It is only good for limited use |
| other | gaming | other | I can search for information that I need urgently | It is only good for limited use |
| other | texting | to access e-mail | It allows me to use online utilities when I am out | It is only good for limited use |
| other | gaming | other | I can search for information that I need urgently | It is very slow |
| vodafone | music | other | I can search for information that I need urgently | The websites do not look as good on mobile |
| vodafone | browsing | to access e-mail | I can search for information that I need urgently | The websites do not look as good on mobile |
| vodafone | music | to social network | Its accessible everywhere | It is only good for limited use |
| other | music | to access e-mail | Its accessible everywhere | It is only good for limited use |
| vodafone | music | to access e-mail | Its accessible everywhere | It is very slow |
| other | music | other | I can search for information that I need urgently | It is only good for limited use |
| other | texting | to social network | I can search for information that I need urgently | It is only good for limited use |
| vodafone | browsing | to read news | It allows me to use online utilities when I am out | It is only good for limited use |
| other | texting | to access e-mail | Its accessible everywhere | its slow |
| vodafone | texting | to access e-mail | I can search for information that I need urgently | The websites do not look as good on mobile |
| vodafone | browsing | to social network | Its accessible everywhere | The websites do not look as good on mobile |
| vodafone | texting | to access e-mail | I can search for information that I need urgently | It is very slow |

31

`

| vodafone | gaming | to access e-mail | It allows me to use online utilities when I am out | It is only good for limited use |
| --- | --- | --- | --- | --- |
| other | music | other | It is a way to pass my time (eg transport, queue,etc) | It is only good for limited use |
| reliance | browsing | to access e-mail | I can search for information that I need urgently | It is only good for limited use |
| vodafone | texting | to social network | Its accessible everywhere | It is very slow |
| reliance | music | to access e-mail | It allows me to use online utilities when I am out | its slow |
| vodafone | texting | to access e-mail | Its accessible everywhere | It is only good for limited use |
| other | music | to access e-mail | Its accessible everywhere | It is very slow |
| other | music | to access e-mail | It allows me to use online utilities when I am out | It is only good for limited use |

Table 4.3


**Survey spreadsheet part-4**

| Do you think Internet on mobile would get better in the future? | will you compromise with brand name for better camera quality | How much would you be ready to spend for an online application that you want to obtain? | How much time do you spend on your mobile phone on average in a day (calls only)? | What would you describe as the main benefits of using internet on your mobile phone? |
| --- | --- | --- | --- | --- |
| no | depends | more than 100 | more than 4 hr | I can search for information that I need urgently |
| yes | no | more than 100 | less than 60 mins | Its accessible everywhere |
| yes | no | more than 100 | less than 60 mins | It allows me to use online utilities when I am out |
| yes | no | will always go for free download | 2-4 hr | Its accessible everywhere |
| may be | depends | will always go for free download | 2-4 hr | I can search for information that I need urgently |

32

`

| | | | | |
|---|---|---|---|---|
| yes | no | will always go for free download | 2-4 hr | It is a way to pass my time (eg transport, queue,etc) |
| yes | depends | 50-100 rs | less than 60 mins | I can search for information that I need urgently |
| may be | depends | will always go for free download | 2-4 hr | I can search for information that I need urgently |
| yes | depends | will always go for free download | less than 60 mins | It allows me to use online utilities when I am out |
| yes | no | 50-100 rs | 2-4 hr | I can search for information that I need urgently |
| may be | no | will always go for free download | less than 60 mins | I can search for information that I need urgently |
| yes | yes | will always go for free download | 1-2 hr | It is a way to pass my time (eg transport, queue,etc) |
| yes | depends | will always go for free download | less than 60 mins | It allows me to use online utilities when I am out |
| yes | depends | will always go for free download | 1-2 hr | Its accessible everywhere |
| may be | yes | will always go for free download | 1-2 hr | Its accessible everywhere |
| yes | depends | will always go for free download | less than 60 mins | Its accessible everywhere |
| yes | no | will always go for free download | 2-4 hr | I can search for information that I need urgently |
| yes | no | more than 100 | 1-2 hr | It allows me to use online utilities when I am out |
| yes | yes | 50-100 rs | 2-4 hr | Its accessible everywhere |
| yes | no | will always go for free download | 2-4 hr | I can search for information that I need urgently |
| yes | no | will always go for free download | less than 60 mins | Its accessible everywhere |
| yes | depends | 50-100 rs | 1-2 hr | It allows me to use online utilities when I am out |

33

`

| | | | | It allows me to use online utilities when I am out |
|---|---|---|---|---|
| yes | no | 10-50 rs | less than 60 mins | |
| yes | depends | will always go for free download | less than 60 mins | It is a way to pass my time (eg transport, queue,etc) |
| yes | depends | will always go for free download | 1-2 hr | I can search for information that I need urgently |
| may be | yes | will always go for free download | 2-4 hr | Its accessible everywhere |
| yes | no | will always go for free download | less than 60 mins | It allows me to use online utilities when I am out |
| yes | no | 10-50 rs | 1-2 hr | Its accessible everywhere |
| yes | no | 50-100 rs | less than 60 mins | It allows me to use online utilities when I am out |
| yes | depends | will always go for free download | 1-2 hr | It allows me to use online utilities when I am out |

Table 4.4

## 4.3 Analysis and Results

As mentioned earlier, we have repeated some questions and then analyzed the behavior of user. The repeated questions and choices made by users are tabulated as follows:

| your name | excluding voice calls, which feature do you use regularly in your smartphone? | Which of these is your regular in use feature? | What would you describe as the main benefits of using internet on your mobile phone? | What would you describe as the main benefits of using internet on your mobile phone? |
|---|---|---|---|---|
| adarsh | music | browsing | Its accessible everywhere | I can search for information that I need urgently |
| Anwar Dilawar Shaikh | game | gaming | Its accessible everywhere | Its accessible everywhere |

34

`

| Name | | | | |
|---|---|---|---|---|
| Vipin | browsing | browsing | It allows me to use online utilities when I am out | It allows me to use online utilities when I am out |
| Rajiv Ratn Shah | browsing | browsing | Its accessible everywhere | Its accessible everywhere |
| Satyendar Yadav | music | texting | It is a way to pass my time (eg transport, queue,etc) | I can search for information that I need urgently |
| Kshitiz | browsing | texting | It is a way to pass my time (eg transport, queue,etc) | It is a way to pass my time (eg transport, queue,etc) |
| Vikas Kumar | browsing | browsing | I can search for information that I need urgently | I can search for information that I need urgently |
| manvi | game | gaming | I can search for information that I need urgently | I can search for information that I need urgently |
| LOL | browsing | texting | It allows me to use online utilities when I am out | It allows me to use online utilities when I am out |
| Mukul Jain | game | gaming | I can search for information that I need urgently | I can search for information that I need urgently |
| Teeja | music | music | I can search for information that I need urgently | I can search for information that I need urgently |
| Mohd Imran | browsing | browsing | I can search for information that I need urgently | It is a way to pass my time (eg transport, queue,etc) |
| Gaurav Chawla | browsing | music | Its accessible everywhere | It allows me to use online utilities when I am out |
| tanu | music | music | Its accessible everywhere | Its accessible everywhere |
| Garima Jindal | browsing | music | Its accessible everywhere | Its accessible everywhere |
| Abhinav Choudhry | music | music | I can search for information that I need urgently | Its accessible everywhere |
| Niharika | texting | texting | I can search for information that I need urgently | I can search for information that I need urgently |
| ashutosh | browsing | browsing | It allows me to use online utilities when I am out | It allows me to use online utilities when I am out |
| Mukul Rawat | browsing | texting | Its accessible everywhere | Its accessible everywhere |

`

| | | | I can search for information that I need urgently | I can search for information that I need urgently |
|---|---|---|---|---|
| upasna | browsing | texting | | |
| Abhinav Kumar Dwivedi | browsing | browsing | Its accessible everywhere | Its accessible everywhere |
| Srideepa Banerjee | texting | texting | I can search for information that I need urgently | It allows me to use online utilities when I am out |
| Jayakumar | game | gaming | It allows me to use online utilities when I am out | It allows me to use online utilities when I am out |
| kewal | browsing | music | It is a way to pass my time (eg transport, queue,etc) | It is a way to pass my time (eg transport, queue,etc) |
| BRAHMRAJ SINGH RAWAT | browsing | browsing | I can search for information that I need urgently | I can search for information that I need urgently |
| Jyoti Swarup | texting | texting | Its accessible everywhere | Its accessible everywhere |
| sundram guha | music | music | It allows me to use online utilities when I am out | It allows me to use online utilities when I am out |
| Mohit Khanna | music | texting | Its accessible everywhere | Its accessible everywhere |
| POOJA ARORA | music | music | Its accessible everywhere | It allows me to use online utilities when I am out |
| piyush | music | music | It allows me to use online utilities when I am out | It allows me to use online utilities when I am out |

Table 4.5

36

`

If we analyze the set of same questions and choices made by user, the results will be as follows:

| your name | choices for 1st set | choices for 2nd set |
|---|---|---|
| adarsh | Different | different |
| Anwar Dilawar Shaikh | Same | same |
| Vipin | Same | same |
| Rajiv Ratn Shah | Same | same |
| Satyendar Yadav | Different | different |
| Kshitiz | Different | same |
| Vikas Kumar | Same | same |
| manvi | Same | same |
| LOL | Different | same |
| Mukul Jain | Same | same |
| Teeja | Same | same |

`

| | | |
|---|---|---|
| Mohd Imran | Same | different |
| Gaurav Chawla | Different | different |
| tanu | Same | same |
| Garima Jindal | Different | same |
| Abhinav Choudhry | Same | different |
| Niharika | Same | same |
| ashutosh | Same | same |
| Mukul Rawat | Different | same |
| upasna | Different | same |
| Abhinav Kumar Dwivedi | Same | same |
| Srideepa Banerjee | Same | different |
| Jayakumar | Same | same |
| kewal | Different | same |
| BRAHMRAJ SINGH RAWAT | Same | same |
| Jyoti Swarup | Same | same |
| sundram guha | Same | same |
| Mohit Khanna | Different | same |
| POOJA ARORA | Same | different |

38

`

| | | |
|---|---|---|
| piyush | Same | same |

Table 4.6

We have found that 17 users have chosen same options. Hence, with a probability of 0.94, we can say that these users has taken this survey seriously and replied sincerely. On the other hand, 11 users have replied with same option for $1^{st}$ set and different options for $2^{nd}$ set. Their behavior can be defined as partial-sincere. They are the users which are needed to be monitored and some deduction in rewards can be done to compensate for the insincere behavior.

There are 3 users who have submitted different options for both the questions. They can be easily declared as the cheat or fraud cases who are replying insincerely with the purpose of making easy and quick money.

## 4.4 Summary

In this chapter I have introduced the environmental setup which I have used while making the experimental results. In addition to this I have also explain the types of dataset which I have used and explain them in detail. Finally I have mentioned all the analysis and experimental results that we have got and found that our proposed approach is capable of detecting cheat and fraud cases for the crowd-voting systems. It can also classify the user behavior into sincere, partially sincere and insincere category.

# Chapter 5: Conclusion & Future Scope

## 5.1 Conclusion

In this thesis, new approaches for detection of fraud and cheat cases are proposed . We have discussed many of the existing approaches and methodologies and found that how they are not capable of cheat detection for all type of crowd-sourcing systems. So we propose two new approaches for detecting cheat and fraud cases for these types of crowd-sourcing systems. While discussing the second approach, we designed synthetic data sets and collected the data by means of online survey forms very similar to what a ranking or voting type of crowd-source system does.  The best part of our proposed approach is that no heavy calculation is needed thus it can easily meet the constraints of mobile systems

## 5.2 Future Scope

a) Using our Verity Index approach, the trust factor is calculated, the parameters of the given formula can be further optimized for various applications by practically implementing these types of systems.

b) In Random Repeat Approach, the behaviors were classified as sincere, partially sincere and insincere. A cost optimizing mechanism is required for the pay scheme for each kind of users. Also, a set of rules for partially sincere and insincere users are needed to be defined.

`

    c) There are some crowd-sourcing systems which are still not implemented with any cheat-detection mechanism.

In general, we feel that crowd-sourcing systems are still developing in the world of web 2.0 and have a huge scope for improvement. There are many issues related to the crowd-sourcing system implementation which are yet not well included. So, crowd-sourcing can be seen as a

`

# REFERNCES

[1] Anatomy of a Crowdsourcing Platform - Using the Example of Microworkers.com ,Matthias Hirth, Tobias Hoßfeld, Phuoc Tran-Gia, 2011 Fifth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing, DOI 10.1109/IMIS.2011.89, 2011 IEEE.

[2] Crowdsourcing in Mobile- A Three Stage Context Based Process-Ahmad Hassan Afridi, 2011 Ninth IEEE International Conference on Dependable, Autonomic and Secure Computing, DOI 10.1109/DASC.2011.60, 2011 IEEE.

[3] Mobile Image Search via Local Crowd: a User Study: Yefeng Liu, odorka Alexandrova, Tatsuo Nakajima, 17th IEEE International Conference on Embedded and Real-Time Computing Systems and Applications, DOI 10.1109/RTCSA.2011.10, 2011 IEEE.

[4] txteagle: Mobile Crowdsourcing:Nathan Eagle, MIT Media Laboratory.

[5] Managing the Crowd: Towards a Taxonomy of Crowdsourcing Processes

[6] MobileWorks: A Mobile Crowdsourcing Platform for Workers at the Bottom of the Pyramid

[7] Cost-Optimal Validation Mechanisms and Cheat-Detection for Crowdsourcing Platforms, Matthias Hirth, Tobias Hoßfeld, Phuoc Tran-Gia, 2011 Fifth

`

International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing, DOI 10.1109/IMIS.2011.91, 2011 IEEE.

[8] M(DM): An Open Framework for Interoperation of Multimode1 Multidatabase Systems: Thierry Barsalou, Dipayan Gangopadhyay, 0-81862545-7192,1992 IEEE.

[9] BROADCAST NEWS SEGMENTATION BY AUDIO TYPE ANALYSIS: Tin LSay NEW and Haizhou LI, ICASSP 2005, 0-7803-8874-7/05/2005 IEEE.

[10]    https://www.facebook.com/note.php?note_id=441195668919

[11]    An Architecture and Interaction Techniques for Handling ambiguity in recognition based input, Jennifer Mankoff,  Georgia Institute of Technology, May 2001.

[12]    Truth in CrowdSourcing, Patrick McDaniel, Sean W. Smith,THE IEEE COMPUTER AND RELIABILITY SOCIETIES ,1540-7993/11/September/October 2011 IEEE.

`

# APPENDIX A: CODING

**AnswersActivity.java**

```java
/**
 *
 */
package com.tmm.android.chuck;

import android.app.Activity;
import android.os.Bundle;
import android.view.KeyEvent;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;
import android.widget.TextView;

import com.tmm.android.chuck.quiz.GamePlay;
import com.tmm.android.chuck.util.Utility;

/**
 * @author robert.hinds
 *
 */
public class AnswersActivity extends Activity implements
OnClickListener {

    @Override
    public void onCreate(Bundle savedInstanceState) {
            super.onCreate(savedInstanceState);
            setContentView(R.layout.answers);
            GamePlay currentGame =
((ChuckApplication)getApplication()).getCurrentGame();

            TextView results = (TextView)findViewById(R.id.answers);
            String answers =
Utility.getAnswers(currentGame.getQuestions());
            results.setText(answers);

            //handle button actions
            Button finishBtn = (Button) findViewById(R.id.finishBtn);
            finishBtn.setOnClickListener(this);

    }




    /* (non-Javadoc)
     * @see android.app.Activity#onKeyDown(int,
android.view.KeyEvent)
     *
     * This method is to override the back button on the phone
     * to prevent users from navigating back in to the quiz
     */
    @Override
```

44

```
`
       public boolean onKeyDown(int keyCode, KeyEvent event)
       {
              switch (keyCode)
              {
              case KeyEvent.KEYCODE_BACK :
                     return true;
              }

              return super.onKeyDown(keyCode, event);
       }


       @Override
       public void onClick(View v) {
              switch (v.getId()){
              case R.id.finishBtn :
                     finish();
              }
       }

}
```

## ChuckApplication.java

```
/**
 *
 */
package com.tmm.android.chuck;

import com.tmm.android.chuck.quiz.GamePlay;

import android.app.Application;

/**
 * @author rob
 *
 */
public class ChuckApplication extends Application{
       private GamePlay currentGame;

       /**
        * @param currentGame the currentGame to set
        */
       public void setCurrentGame(GamePlay currentGame) {
              this.currentGame = currentGame;
       }

       /**
        * @return the currentGame
        */
       public GamePlay getCurrentGame() {
              return currentGame;
       }
}
```

45

`

**EndGameActivity.java**
```java
/**
 *
 */
package com.tmm.android.chuck;

import android.app.Activity;
import android.content.Intent;
import android.content.SharedPreferences;
import android.os.Bundle;
import android.view.KeyEvent;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.TextView;

import com.tmm.android.chuck.quiz.Constants;
import com.tmm.android.chuck.quiz.GamePlay;
import com.tmm.android.chuck.quiz.Helper;

/**
 * @author robert.hinds
 *
 */
public class EndgameActivity extends Activity implements
OnClickListener {

    @Override
    public void onCreate(Bundle savedInstanceState) {
            super.onCreate(savedInstanceState);
            setContentView(R.layout.endgame);
            GamePlay currentGame =
((ChuckApplication)getApplication()).getCurrentGame();
            String result = "You Got " + currentGame.getRight() + "/"
+ currentGame.getNumRounds() + ".. ";
            String comment =
Helper.getResultComment(currentGame.getRight(),
currentGame.getNumRounds(), getDifficultySettings());


            TextView results =
(TextView)findViewById(R.id.endgameResult);
            results.setText(result + comment);

            int image = Helper.getResultImage(currentGame.getRight(),
currentGame.getNumRounds(), getDifficultySettings());
            ImageView resultImage =
(ImageView)findViewById(R.id.resultPage);
            resultImage.setImageResource(image);

            //handle button actions
            Button finishBtn = (Button) findViewById(R.id.finishBtn);
            finishBtn.setOnClickListener(this);
            Button answerBtn = (Button) findViewById(R.id.answerBtn);
            answerBtn.setOnClickListener(this);
```

46

```
        `

        }


        /**
         * Method to return the difficulty settings
         * @return
         */
        private int getDifficultySettings() {
                SharedPreferences settings =
getSharedPreferences(Constants.SETTINGS, 0);
                int diff = settings.getInt(Constants.DIFFICULTY, 2);
                return diff;
        }


        /* (non-Javadoc)
         * @see android.app.Activity#onKeyDown(int,
android.view.KeyEvent)
         *
         * This method is to override the back button on the phone
         * to prevent users from navigating back in to the quiz
         */
        @Override
        public boolean onKeyDown(int keyCode, KeyEvent event)
        {
                switch (keyCode)
                {
                case KeyEvent.KEYCODE_BACK :
                        return true;
                }

                return super.onKeyDown(keyCode, event);
        }


        @Override
        public void onClick(View v) {
                switch (v.getId()){
                case R.id.finishBtn :
                        finish();
                        break;

                case R.id.answerBtn :
                        Intent i = new Intent(this, AnswersActivity.class);
                        startActivityForResult(i, Constants.PLAYBUTTON);
                        break;
                }
        }

}
```

`

### SplashActivity.java

```java
package com.tmm.android.chuck;

import java.io.IOException;
import java.util.List;

import android.app.Activity;
import android.content.Intent;
import android.content.SharedPreferences;
import android.database.SQLException;
import android.os.Bundle;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;
import android.widget.Toast;

import com.tmm.android.chuck.db.DBHelper;
import com.tmm.android.chuck.quiz.Constants;
import com.tmm.android.chuck.quiz.GamePlay;
import com.tmm.android.chuck.quiz.Question;

public class SplashActivity extends Activity implements
OnClickListener{


    public static int iii=0;
    public static String [] myanswers=new String[20];

    @Override
    public void onCreate(Bundle savedInstanceState) {
            super.onCreate(savedInstanceState);
            setContentView(R.layout.welcome);


    ////////////////////////////////////////////////////////////
///////
            //////// GAME MENU
/////////////////////////////////////////////
            Button playBtn = (Button) findViewById(R.id.playBtn);
            playBtn.setOnClickListener(this);
            Button settingsBtn = (Button)
findViewById(R.id.settingsBtn);
            settingsBtn.setOnClickListener(this);
            Button rulesBtn = (Button) findViewById(R.id.rulesBtn);
            rulesBtn.setOnClickListener(this);
            Button exitBtn = (Button) findViewById(R.id.exitBtn);
            exitBtn.setOnClickListener(this);
    }


    /**
     * Listener for game menu
     */
    @Override
    public void onClick(View v) {
            Intent i;
```

48

```
        `
                    switch (v.getId()){
                    case R.id.playBtn :
                            //once logged in, load the main page
                            //Log.d("LOGIN", "User has started the game");

                            //Get Question set //
                            List<Question> questions = getQuestionSetFromDb();

                            //Initialise Game with retrieved question set ///
                            GamePlay c = new GamePlay();
                            c.setQuestions(questions);
                            c.setNumRounds(getNumQuestions());

            ((ChuckApplication)getApplication()).setCurrentGame(c);

                            //Start Game Now.. //
                            i = new Intent(this, QuestionActivity.class);
                            startActivityForResult(i, Constants.PLAYBUTTON);
                            break;

                    case R.id.rulesBtn :
                            i = new Intent(this, RulesActivity.class);
                            startActivityForResult(i, Constants.RULESBUTTON);
                            break;

                    case R.id.settingsBtn :
                            i = new Intent(this, SettingsActivity.class);
                            startActivityForResult(i,
        Constants.SETTINGSBUTTON);
                            break;

                    case R.id.exitBtn :
                            finish();
                            break;
                    }

            }


            @Override
        public void onBackPressed() {
            // TODO Auto-generated method stub
            Toast.makeText(this, "back pressed", 1000).show();
            super.onBackPressed();
        }


            /**
             * Method that retrieves a random set of questions from
             * the database for the given difficulty
             * @return
             * @throws Error
             */
            private List<Question> getQuestionSetFromDb() throws Error {
                    int diff = getDifficultySettings();
                    int numQuestions = getNumQuestions();

                            49
```

```
`
            DBHelper myDbHelper = new DBHelper(this);
            try {
                    myDbHelper.createDataBase();
            } catch (IOException ioe) {
                    throw new Error("Unable to create database");
            }
            try {
                    myDbHelper.openDataBase();
            }catch(SQLException sqle){
                    throw sqle;
            }
            List<Question> questions =
myDbHelper.getQuestionSet(diff, numQuestions);
            myDbHelper.close();
            return questions;
      }


      /**
       * Method to return the difficulty settings
       * @return
       */
      private int getDifficultySettings() {
            SharedPreferences settings =
getSharedPreferences(Constants.SETTINGS, 0);
            int diff = settings.getInt(Constants.DIFFICULTY,
Constants.MEDIUM);
            return diff;
      }

      /**
       * Method to return the number of questions for the game
       * @return
       */
      private int getNumQuestions() {
            SharedPreferences settings =
getSharedPreferences(Constants.SETTINGS, 0);
            int numRounds = settings.getInt(Constants.NUM_ROUNDS,
20);
            return numRounds;
      }

}
```

# APPENDIX B: SCREEN SHOTS