Simulation of Consensus Based Approaches to Mitigate the Challenges in Crowdsourcing

Kumar Abhinav
IIIT Delhi, New Delhi
kumar1365@iiitd.ac.in

Shrikanth N.C
Accenture Technology Labs, Bangalore
shrikanth.n.c@accenture.com

Anurag Dwarakanath
Accenture Technology Labs, Bangalore
anurag.dwarakanath@accenture.com

***Abstract*—Crowdsourcing is an emerging area and has evolved as a powerful practice to leverage the collective intelligence of the crowd. It has been applied in various domains ranging from creative resolution of a problem to improving the business process using several platforms such as CrowdFlower, Freelancer and Amazon Mechanical Turk. Crowd is a creative workforce that has niche abilities to solve complex business challenges across various domains. It can be seen as an alternate workforce by participating in all phases of software development life cycle. However the common problem seen in crowdsourcing is the quality of the work performed by the crowd mostly due to the anonymity of the crowd member. In this work, we evaluated consensus based approach to assess the quality of the work done by the crowd through a simulation of crowd behavior. We also investigated the performance of these techniques for evaluating crowd members.**

I. Introduction

Crowdsourcing Software development is a promising and emerging field. It acts as a platform where the crowd can perform the entire software development tasks given by crowdsourcer or requester. In a software engineering context, client may not have knowledge of the crowd who develops the software and is unaware of the processes followed. It is likely that the developer decides on such a course of action that satisfies the minimum requirements to submit the task but such actions could bring liabilities to the enterprise [1]. Hence, owing to the anonymity of the crowd, evaluating the quality of the work of the crowd becomes a major challenge in crowdsourcing software development. There are different computational approaches in related literature to evaluate the submissions made by the crowd. In this paper we will discuss the following approaches which are commonly used in crowdsourcing

1) **Reputation based Approach**: In this approach, historical data of quality of work submitted by the crowd and interaction done with crowdsourcer, is used to generate a reputation score for each worker. The past performance of the workers assesses the quality of the workers.
2) **Gold Standard Approach**: In this approach a set of questions is put in the task for which answers are already known to the crowdsourcer. Based on the discrepancy between response submitted by the crowd and correct answer for predefined set of questions, workers' quality can be assessed.
3) **Consensus based Approach**: This is the most common approach to determine the true response and in turn to assess the credibility of the crowd. In this approach consensus is built by the crowd. Each response is considered as a vote and is based on the belief that eventually the most accurate solution will get maximum votes. This approach relies on redundancy i.e. ask multiple workers to complete the same task.

In this paper, we adopted consensus based approach. The submissions are given to the crowd members to evaluate quality pertaining to the three issues- Trojan code, Non-adherence to best practices and Non-compliant licensed software. (Figure 1). However, the challenge with this approach is to aggregate the response from the crowd and find out the best solution. There are various ways to aggregate the crowd's response and predict the true value [2]. In this paper, we will discuss two of these approaches.

1) **Majority Voting (MV)**: This is the most common and simple consensus based method. In majority voting, the label agreed with majority is treated as correct or true label. It assumes majority of workers in the crowd are quality workers who work independently and ultimately the majority of crowd workers' vote will agree on ground truth.
2) **Expectation Maximization (EM) Algorithm**: This is an algorithm for finding the probabilities of latent variables, which can be used to estimate the true labels and the workers' accuracy [3].

Due to lack of availability of real world datasets on which we can test performance of Majority voting and Expectation Maximization algorithm, we generated synthetic datasets based on the simulation of workers' behavior and prior probabilities for each category. We simulated the behaviour of the crowd as a probabilistic system while considering different types of crowd worker [4]. Each crowd worker is assumed to follow a Bernoulli distribution to give a binary answer to a question. Every question has an answer following the Bernoulli Distribution, but with a skewed prior probabilities e.g. the chance of having Trojan code is very low with 0.2 probability, Non-adherence to best practices is very high with 0.7 and Non-compliant licensed software with 0.5 probability

There can be different types of crowd workers in crowdsourcing. Mathematically, we have defined different types of

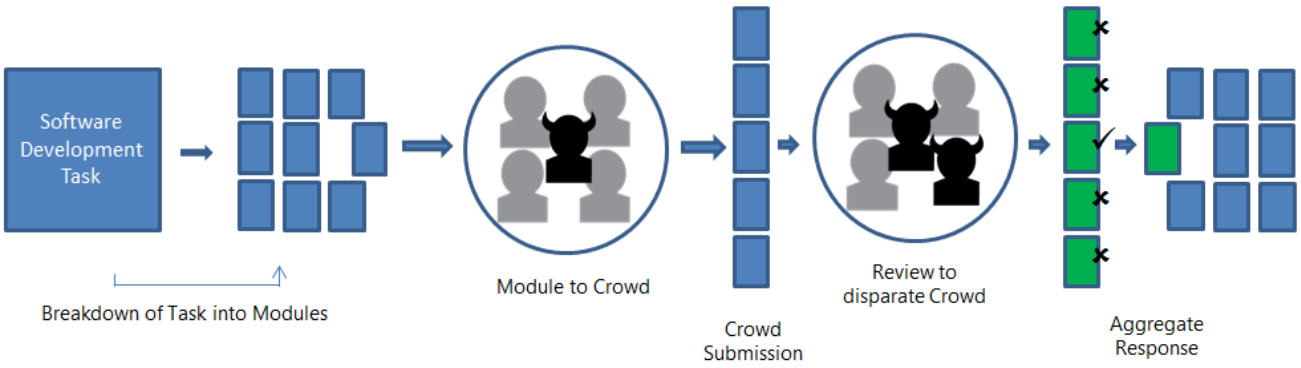

Fig. 1: Our Approach

workers' responses in a probabilistic manner. For example, probability of giving the correct answer for an expert worker is 0.65, for a biased worker it is 0.2.

1) **Expert worker**: expertise in the area with profound domain knowledge and the questions answered correct with a high probability.
   $p\,(EW = TRUE \mid Actual = TRUE)$=0.65
2) **Biased worker**: intentionally gives incorrect answers.
   $p\,(BW = FALSE \mid Actual = TRUE)$=0.8
3) **Random Spammer**: gives random answers for any question.
   $p\,(RS = TRUE \mid Actual = TRUE\,or\,FALSE)$=0.5
4) **Uniform Spammer**: with a specific motive give same answers for all the questions.
   $p\,(US = FALSE \mid Actual = TRUE\,or\,FALSE)$=0.9
5) **Adversarial Colluded worker**: give wrong answers by colluding with other workers having malicious intention. Adversarial Colluded Leader (ACL) with malicious intention marks all answers as wrong. Adversarial Colluded Followers (ACF) follow their leader and mark all answers same (with high probability) as leader.
   $p\,(ACL = TRUE \mid Actual = TRUE)$=0.2
   $p\,(ACF = TRUE \mid ACL = TRUE)$=0.9
6) **Non-Adversarial Colluded worker**: give correct answers by colluding with other workers for the sake of monetary benefits. Non-Adversarial Colluded (NACL) Leader marks all right answers and Non-Adversarial Colluded Followers (NACF) follow their leader and copy the answer marked by leader.
   $p\,(NACL = TRUE \mid Actual = TRUE)$=0.8
   $p\,(NACF = TRUE \mid NACL = TRUE)$=0.9

We conducted various experiments to observe how the accuracy of MV and EM algorithm varies with different types of workers. For all the experiments, we kept number of tasks as fixed to be 100, because this gives a high base of accuracy (based on our experiment) and varied the number of workers. We computed the average of each algorithm's accuracy over 100 runs to obtain the results.

1) **Effect of Expert workers**: The experiment was conducted with number of experts varying from 1 to 40. The accuracy of both the algorithms increases with the increase in number of expert workers. Here both algorithms have similar performance (shown in Figure 2).

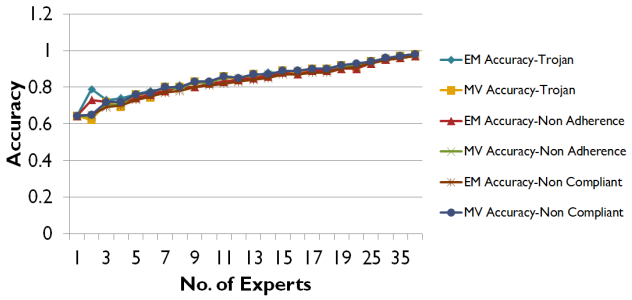

Fig. 2: Accuracy vs Number of Experts

Based on this result, we fixed the number of workers to 10 in all the subsequent experiments as this gives a high starting point of accuracy. As we increased the number of workers, accuracy increased but that came at a cost. Hence, there is a trade-off between cost and accuracy here.

2) **Effect of Biased Workers**: In this experiment, we varied the percent of biased workers from 10 to 40. We observed that for biased workers, EM performs better than MV as EM models the workers' behavior by confusion matrix. Figure 3 depicts the result.
3) **Effect of Spammers**: In this experiment, we increased the percent of spammers to study how it affects accuracy of both the algorithm. Figures 4 and 5 demonstrate the effects of Random Spammers and Uniform Spammers respectively. In general, both EM and MV are equally affected by presence of spammers. The accuracy of both approaches decreases as the number of spammers increases.

In case of uniform spammers, it is clearly evident from the graph that skewed prior probabilities affect accuracy of the algorithm. The probability of having Trojan code is very low (0.2) and the uniform spammer marks it as "Not Trojan" with 0.9 probability. This response from spammer acts like response from an expert worker, which in turn increases accuracy.

4) **Effect of Adversarial colluded workers**: The effect of adversarial colluded workers is depicted in Figure 6. As we increase the percent of adversarial colluded workers, accuracy of both EM and MV decreases. Based on our observation we concluded that for Adversarial colluded workers, EM is more affected than MV.
5) **Effect of Non-Adversarial colluded workers**: The effect of non-adversarial colluded workers is depicted in Figure 7. As we increase the percent of non-adversarial colluded workers, accuracy of both EM and MV increases upto certain point and then it becomes constant. Here, both algorithms have similar performance.

Based on our experiment, we concluded accuracy of both Majority voting and Expectation Maximization Algorithm are affected by different types of workers in crowdsourcing and skewed prior probabilities.

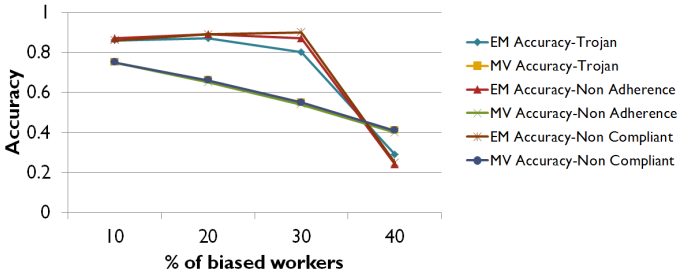

Fig. 3: Accuracy vs % of Biased workers

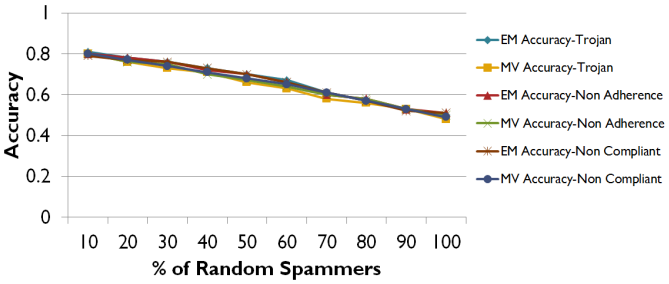

Fig. 4: Accuracy vs % of Random Spammers

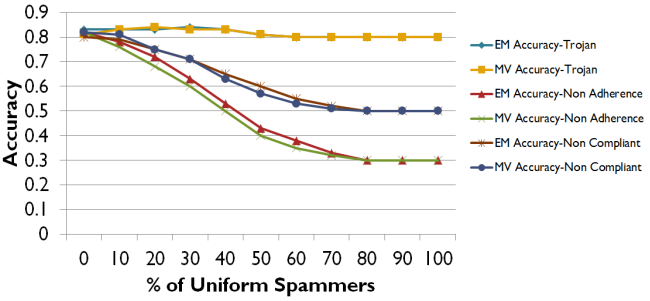

Fig. 5: Accuracy vs % of Uniform Spammers

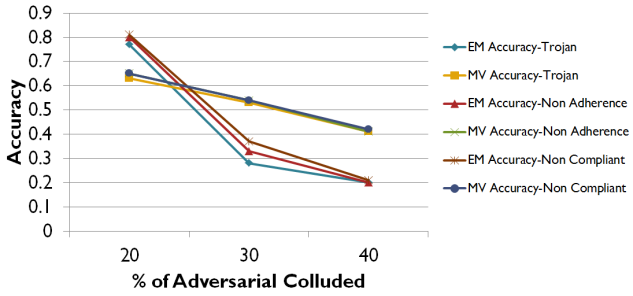

Fig. 6: Accuracy vs % of Adversarial Colluded workers

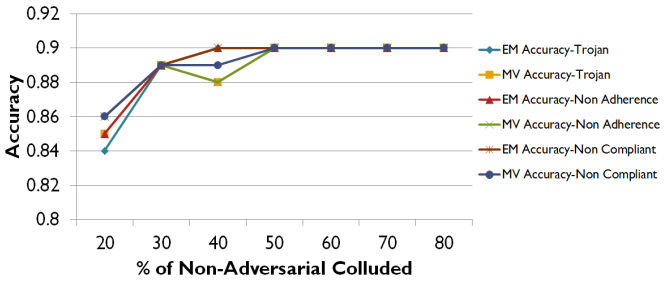

Fig. 7: Accuracy vs % of Non-Adversarial Colluded workers

REFERENCES

[1] A. Dwarakanath, U. Chintala, N. C. Shrikanth, G. Virdi, A. Kass, A. Chandran, and S. Paul, "Crowdbuild: A methodology for enterprise software development using crowdsourcing."

[2] A. Sheshadri and M. Lease, "Square: A benchmark for research on computing crowd consensus," in *First AAAI Conference on Human Computation and Crowdsourcing*, 2013.

[3] A. P. Dawid and A. M. Skene, "Maximum likelihood estimation of observer error-rates using the em algorithm," *Applied statistics*, pp. 20–28, 1979.

[4] G. Kazai, J. Kamps, and N. Milic-Frayling, "Worker types and personality traits in crowdsourcing relevance labels," in *Proceedings of the 20th ACM international conference on Information and knowledge management*. ACM, 2011, pp. 1941–1944.