ACCD: An Adaptive Clustering-Based Collusion Detector in Crowdsourcing (Student Abstract)

Ruoyu Xu[1], Gaoxiang Li[1], Wei Jin[2], Austin Chen[3], Victor S. Sheng [1]

[1]Computer Science Departement, Texas Tech University, Lubbock, Texas, USA
[2] Computer Science and Engineering Departement, University of North Texas, Denton, Texas, USA
[3] Lubbock High School, 2004 19th St, Lubbock, Texas, USA
ruoyxu@ttu.edu, gaoli@ttu.edu, wei.jin@unt.edu, austinchen2005@gmail.com, victor.sheng@ttu.edu

Abstract

Crowdsourcing is a popular method for crowd workers to collaborate on tasks. However, workers coordinate and share answers during the crowdsourcing process. The term for this is "collusion". Copies from others and repeated submissions are detrimental to the quality of the assignments. The majority of the existing research on collusion detection is limited to ground truth problems (e.g., labeling tasks) and requires a predetermined threshold to be established in advance. In this paper, we aim to detect collusion behavior of workers in an adaptive way, and propose an Adaptive Clustering Based Collusion Detection approach (ACCD) for a broad range of task types and data types solved via crowdsourcing (e.g., continuous rating with or without distributions). Extensive experiments on both real-world and synthetic datasets show the superiority of ACCD over state-of-the-art approaches.

Introduction

Crowdsourcing is popular in academia and industry. It helps solve scientific problems that machines cannot, like image labeling, sentiment analysis, and handwriting recognition. Multiple studies have found that solution quality is linked to worker quality. The quality of workers refers to their knowledge, responsibility, and honesty. Research shows that normal collaboration in crowdsourcing improves the solution quality (Sheng, Provost, and Ipeirotis 2008). However, some participants may converse on social media and copy others' answers while doing tasks. It is known as "collusion". Obviously, collusion decreases crowdsourcing solution quality.

In order to reduce the impact of the collusion of participants and improve the quality of solutions from crowdsourcing, only a few cutting-edge collusion detection methods are available, such as FINDCOLLUDERS (FC) (KhudaBukhsh, Carbonell, and Jansen 2014), Collusion-Proof (CP) (Chen et al. 2018) and PROCAP (Song, Liu, and Zhang 2021). However, their applications are limited by the kinds of tasks and corresponding collaboration mechanisms. In this paper, we aim to propose a new method, an adaptive clustering-based collusion detection approach (ACCD), for a broad range of task types and data types solved via crowdsourcing.

Methods

Inspired by the Hierarchical Density-Based Spatial Clustering of Applications with Noise (HDBSCAN) (Campello, Moulavi, and Sander 2013), we develop the ACCD method based on the HDBSCAN to adaptively detect the collusion behavior of colluders for a variety of data distributions.

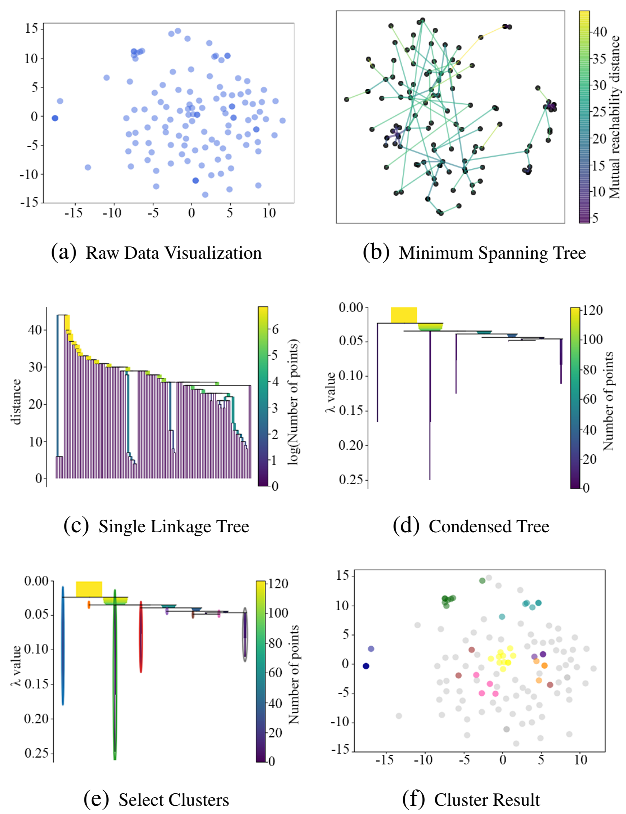
Unlike previous density-based clustering methods, the core idea of HDBSCAN is to calculate the distance differently, including the following key definitions: **core distance** and **mutual reachable distance**. The distance between the sample and the k_{th} nearest neighbor sample point is referred to as the core distance d_{core}. Mutual reachable distance is the maximum value of the core distance of two sample points and the distance between two sample points. The mutual reachable distance can be obtained with: $d_{mreach}(a, b) = max\{d_{core}(a), d_{core}(b), d(a, b)\}$, where $d(a, b)$ denotes the distance between points a and b. Sample distance in the dense region does not change, but sample distance in the sparse region grows, which makes it easier for the algorithm to deal with noise points and increases the robustness of the algorithm to noise points. The procedure of our ACCD algorithm is presented in Figure 1.

Experiments and Results

Real and Synthetic Datasets

The real dataset is from an e-commerce company's product rating problem and is the only published dataset where workers admit collusion (KhudaBukhsh, Carbonell, and Jansen 2014). It contains 20 rating tasks. There are 123 participants, and 36 of them are suspected of colluding.

Due to the limited availability of real data, we construct multiple synthetic datasets. To simulate a variety of crowdsourcing problems and test collusion detectors, synthetic datasets contain rating and ground truth problems. Rating problems are more subjective inquiries in which consumers give a product a personal subjective rating based on their own opinion and experience. While the ground truth problems are those that have actual answers, which are responses based on prior knowledge and common scientific senses. Also, we generate two types of responses: categorical and continuous. A categorical data type has a limited number of categories or groups to choose. A continuous data type (i.e., a numeric variable) has infinite continuous values.



(a) Raw Data Visualization
(b) Minimum Spanning Tree
(c) Single Linkage Tree
(d) Condensed Tree
(e) Select Clusters
(f) Cluster Result

Figure 1: The procedure of ACCD with HDBSCAN

Methods	Precision	Recall	F1 Score	Accuracy
FC	0.775	0.861	0.816	0.886
CP	0.583	0.389	0.467	0.740
PROCAP	0.804	0.736	0.768	0.772
ACCD	**0.829**	**0.944**	**0.883**	**0.927**

Table 1: Performance of different detection methods on the real-world dataset

For the synthetic datasets, we create a collection of simulated answers from non-colluding and colluding workers with the following four parameters. Number of tasks denotes the total number of these tasks. Number of workers denotes the total number of people involved in these tasks. Non-Collusion ratio denotes the percentage of total workers who not colluded, and number of collusion groups denotes the total number of collusion groups in these tasks.

Experimental Results

We first conduct the experiments to compare the performances of FC, CP, PROCAP and our ACCD on the real-world dataset. Our experimental results (see Table 1) show that our ACCD performs consistently better than the other three methods in terms of all measures.

On the synthetic datasets, we assume 50 equal-difficulty tasks. A total of 250 workers participate, and 30% of them are colluders. There are 4 collusion groups, and the number of members in each group is determined at random. We conduct experiments on four types of crowdsourcing problems. Our experimental results (see Table 2) show that our ACCD outperforms the other three approaches.

Problem Types	Methods	P	R	F1	Acc
Categorical Rating Problems	FC	1.000	0.027	0.052	0.708
	CP	0.900	0.120	0.212	0.732
	PROCAP	0.685	0.631	0.656	0.760
	ACCD	**1.000**	**0.933**	**0.966**	**0.980**
Continuous Rating Problems	FC	1.000	0.667	0.800	0.720
	CP	0.676	0.333	0.446	0.752
	ACCD	0.915	**1.000**	**0.955**	**0.972**
Categorical Ground Truth Problems	FC	1.000	0.107	0.193	0.732
	CP	0.554	0.671	0.607	0.737
	PROCAP	0.800	0.693	0.743	0.810
	ACCD	0.872	**1.000**	**0.932**	**0.956**
Continuous Ground Truth Problems	FC	1.000	0.093	0.171	0.728
	CP	0.000	0.000	0.000	0.700
	ACCD	0.935	**0.960**	**0.947**	**0.968**

Table 2: Performance of different detection methods for different type of problems (P, R, F1 and Acc denote precision, recall, F1 score and accuracy respectively)

In order to further test our ACCD's accuracy in detecting collusion, we conduct more experiments with various settings. We create 4000 simulated datasets in total and 1000 datasets for each type of problem. Various settings include the number of tasks, the number of workers, the non-collusion ratios, and the number of collusion groups. We keep three of the data generator's four variables constant while changing only one of them to generate datasets. According to accuracy, we can find that our ACCD not only performs better than the other three methods, but also keeps a consistently high performance for all types of problems.

References

Campello, R. J.; Moulavi, D.; and Sander, J. 2013. Density-based clustering based on hierarchical density estimates. In *Pacific-Asia conference on knowledge discovery and data mining*, 160–172. Springer.

Chen, P.-P.; Sun, H.-L.; Fang, Y.-L.; and Huai, J.-P. 2018. Collusion-proof result inference in crowdsourcing. *Journal of Computer Science and Technology*, 33(2): 351–365.

KhudaBukhsh, A. R.; Carbonell, J. G.; and Jansen, P. J. 2014. Detecting non-adversarial collusion in crowdsourcing. In *Second AAAI Conference on Human Computation and Crowdsourcing*.

Sheng, V. S.; Provost, F.; and Ipeirotis, P. G. 2008. Get another label? improving data quality and data mining using multiple, noisy labelers. In *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, 614–622.

Song, C.; Liu, K.; and Zhang, X. 2021. Collusion Detection and Ground Truth Inference in Crowdsourcing for Labeling Tasks. *Journal of Machine Learning Research*, 22(190): 1–45.