IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received March 17, 2019, accepted May 23, 2019, date of publication June 4, 2019, date of current version June 26, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2920667

Applying Cluster Refinement to Improve Crowd-Based Data Duplicate Detection Approach

CHARLES ROLAND HARUNA[1,2], MENGSHU HOU[1], RUI XI[1], MOSES JOJO EGHAN[2], MICHAEL Y. KPIEBAAREH[1], LAWRENCE TANDOH[1], BARBIE EGHAN-YARTEL[2], AND MAAME G. ASANTE-MENSAH[2]

[1] Department of Computer Science and Technology, University of Electronic Science and Technology of China, Chengdu 611731, China
[2] Department of Computer Science and Information Technology, University of Cape Coast, Cape Coast, Ghana

Corresponding author: Charles Roland Haruna (charuna@ucc.edu.gh)

This work was supported in part by the NSFC under Grant 61472067, in part by the Leadership Funding Support for Academic and Technical Leaders of Sichuan, under Grant W03002360100203, and in part by the National Key Research and Development Program "Research on Intelligent Disposal Technology of Multiple Complaint Letters and Visits" under Grant 2018YFC0831800.

ABSTRACT In this paper, we present an extension on a hybrid-based deduplication technique in entity reconciliation (ER), by proposing an algorithm that builds clusters upon receiving a pre-specified K number of clusters, and second developing a crowd-based procedure for refining the results of the clusters produced after the clustering generation phases. With the clusters refined, we aim to minimize the cost metric $\Lambda'(R)$ of the solitary and compound cluster generation algorithms, to achieve an improved and efficient deduplication method, to have an increase in accuracy in identifying duplicate records, and finally, further reduce the crowdsourcing overheads incurred. In this paper, in the experiments, we made use of three datasets commonly known to hybrid-based deduplication such as paper, product, and restaurant. The performance results and evaluations demonstrate clear superiority to the methods compared with our work offering low-crowdsourcing cost and high accuracy of deduplication, as well as better deduplication efficiency due to the clusters being refined.

INDEX TERMS Cluster refinement, minimization approach, triangular split and merger operations, entity reconciliation, crowdsourcing.

I. INTRODUCTION

In Entity Reconciliation (ER), records that may be identical are identified in database systems [2]–[4], [23]. ER is very important in cleaning records that belong to the same real world object from database systems. The duplicate records are put into clusters and if they don't have a common key or are noisy, then accurate deduplication becomes a challenge. Haruna et al. [1], presented a hybrid based data deduplication approach in ER. Where a machine-based system, the Cosine similarity function [11], [12], was first used on sets of data to calculate for the similarity scores between each pair of records using metrics with a set threshold. The pairs with scores greater than the threshold were pruned to form a candidate set S. An adaptive clustering algorithm, the Chromatic Correlation Clustering [12] under crowdsourcing was used on the pairs of records in S to either group them into single or compound clusters. Finally, the clusters were submitted to the crowdsourcing platform, for the humans to thoroughly examine the pairs of records, to confirm their equivalence and submit their answers. Based on the crowd's confidence [9] and triangular similarity scores [1], a permanent cluster is either formed, implying the records in it are almost equal, or otherwise not formed.

However, during the cluster generation stages in the previous work [1], some of the record pairs were not issued to the crowd for examination. They were either deleted when forming the clusters or were not chosen at all for the cluster formation. This is a huge problem because the aim of the deduplication mechanism is to examine all records and identify duplicates among them. Therefore, for some of the record pairs not be examined defaults the objective of the deduplication mechanism.

The associate editor coordinating the review of this manuscript and approving it for publication was Ashish Mahajan.

The extended work on the hybrid deduplication method proposed in [1] consists of:

Firstly, propose a procedure to fine-tune the clusters. Secondly, we present an adaptive alternating minimization approach to specify a desired number of K output clusters if needed.

Finally, with the clusters refined, we again conducted the same experiments as in [1] and compared the results to some existing models including the proposed human-machine based deduplication in [1]. Our results showed improvements on the results obtained in [1] in terms accuracy and efficiency but incurred a little more crowdsourcing cost. Also, we compared the efficiency of the proposed work, in terms of modification of parameters against some existing work.

The proposed methods in this work, seek to rectify the shortcomings of work [1], by posting all record to the crowd while using additional Human Intelligent Tasks (HITs) to examine the record pairs, An HIT is where the crowd are given easy duties to execute and are given some reward. In examining all the record pairs while generating the clusters as well, there is the possibility of having a higher accuracy and efficiency during the experiments, which shall be determined and explained later in the work. The crowd cost, which is dependent on the number of record pairs submitted to the crowd will be determined as well.

Many works have been done on data deduplication but to the best of our knowledge, only one work [9] further refined their clusters (which were generated using a single record pair as pivot). Therefore, no deduplication method has been proposed that makes use of cluster refinement and using a record pair as pivot.

Haruna *et al.* [1] proposed a hybrid data deduplication method, blending a machine based algorithm and a crowdsourcing platform which offered a high accuracy of deduplication, incurring lower crowd cost. Abualigah *et al.* [17] used an algorithm called the particle swarm optimization (PSO) (FSPSOTC) to answer some feature selection problems. The creation of a new subset of informative text features was proposed and this tend to improve the performance of the clustering technique of the text and also to reduce the time needed for computation. A weighting scheme, LFW - Length feature weight, proposed by Abualigah *et al.* [18], used information gained from the documents collection to enhance clustering algorithms in text documents. In the work of Abualigah et al. [19], to resolve the text document clustering problem, they proposed algorithm made of an objective functions with a hybrid KH. From K-mean clustering method, the preliminary results of the KH algorithm are gained. Based on two merged objective functions, decisions on clustering were made. The proposed method was termed MHKHA. Further, Abualigah et al. [20] presented a novel text clustering approach called MMKHA, which is an algorithm enhancement fusing the krill herd and a hybrid function. The proposed work was proved to be an efficient way of clustering to obtain accurate results. To upgrade the global search, Abualigah *et al.* [21] suggested H-KHA, a novel fusion of algorithms; KH and harmony search, to improve the exploring capabilities by a new probability factor called a Distance factor. Finally, Abualigah *et al.* [22] proposed an approach using the particle swarm technique of optimization. For the problem of feature selection the method also used genetic operators. Here, the effectiveness of the algorithm was conducted by using the K-means clustering.

When confronted with postclustering problems, clustering refinement is useful to tackle these issues. The objective of clustering refinement is to refine and improve the single data clusters. Vega-Pons *et al.* [28]did a survey on clustering refinement techniques to aid the education and industry in choosing the right method to resolve a problem encountered. Gionis *et al.* [24] proposed formal definitions and several algorithms for the problem of aggregating clusters. The algorithms used linkages between correlation clustering drawbacks [25] and clustering aggregation and provided high quality solutions. Also, a method to scale algorithms for datasets which are huge was introduced. Given bits of contradictory input information, and with the objective of obtaining a consistent universal solution, to minimize the degree of disagreement, Ailon *et al.* [26] addressed and used a simple algorithm to improve ranking aggregation, clustering from consensus and correlation and the problems incurred o tournament from the feedback arc set. Goder *et al.* [27] instigated various heuristics methods to tackle the problems associated with the consensus clustering by comparing their operations with respect to efficiency and efficacy using two datasets; simulated and real , and taking into account the data sizes individualistic. They discovered that the heuristics can be grouped into two discrete groups. The also proposed a solution for situations when clusterings are different.

To the best of our knowledge, only Wang *et al.* [9], proposed in their work, a mechanism to refine the clusters after they have been generated, under a crowdsourcing platform. They refined cluster using a only single record in their operations. In this work we propose a mechanism to refine the clusters after they have been generated, under a crowdsourcing platform using a pair of record in the operations.

II. CONTRIBUTIONS OF THE WORK

Given a record $R = (r_1, r_2, ..., r_n)$ and a function g under a crowdsourcing setting, Haruna et al. [1] presented a hybrid data deduplication technique, where pairs of records r_i, r_j in R were initially pruned using a machine-based technique. Then based on algorithm, clusters were formed and submitted to the crowd to determine the records that are of the same real world entity. In work [1], minimizing $\Lambda'(R)$ had a very poor performance because the clusters' generations were done randomly and also some record pairs in S in the iterations were not issued to the crowd to be processed because the related edges were removed from the graph as the temporary clusters are removed. For example generating a single cluster by executing Human-Edge-Pivot Algorithm in [1] forms triangular clusters $\langle ABC \rangle$ and $\langle MNO \rangle$ shown in Fig. 1. In the generation process, edges (B, N), (BO), (AN), (AO), (AM),

TABLE 1. Table of notations.

Notation	Description
$R = r_1, r_2, ..., r_n$	Records set in a database
S	Candidate set
$G = (V_r$ and $E_s)$	An undirected graph where V_r are the records in R and E_s are the edges corresponding to a record pair in the candidate set S
$f(r_1, r_2)$	Similarity score between two records
$f_c(r_1, r_2)$	Crowd's confidence score that two records are equal
$f_t(r_1, r_2, r_3)$	Crowd's confidence score that three records in a cluster are the same
metric cost, $\Lambda'(R)$	The goal to minimize the cost of our algorithm
$b(o)$	benefit of executing operation o
$b^*(o)$	an approximated benefit of o
$c(o)$	cost incurred by the crowd executing o

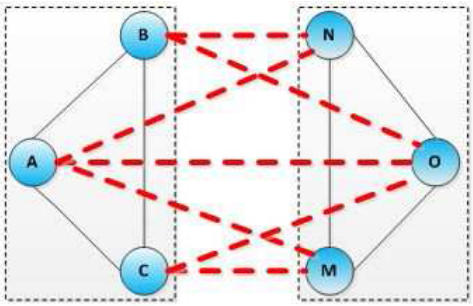


FIGURE 1. Formation of single clusters showing deleted edges.

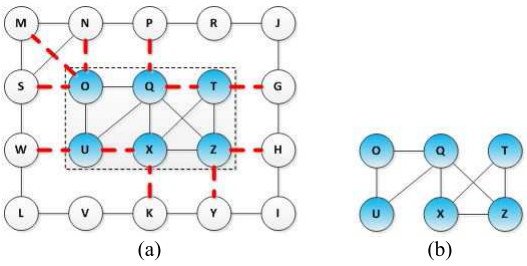


FIGURE 2. An example of compound cluster with eliminated edges. (a) A simple undirected graph. (b) Compound triangular clustering.

(CO) and (CM) marked in red broken lines are removed and not processed by the crowd, in which some may be of the same real world entity as the ones already processed. Likewise, generating a compound cluster comprising of a group of triangular clusters $\langle OUQ\rangle$, $\langle QXZ\rangle$ AND $\langle TXZ\rangle$, shown in Fig. 2, edges (MO), (SO), (NO), (PQ), (QT), (GT), (HZ), (UW), (KX) and (TZ) were deleted from the graph as well. To address these issues, the cluster refinement phase of our solution post-processes the results produced by the cluster generation phases, by using additional HITs to fine-tune the clusters, so as to reduce $\Lambda'(R)$. In doing so, any of vertices M, N and S after preprocessing may be added to the clusters.

The extended work on the hybrid deduplication method proposed in [1] consists of:

Firstly, we propose a mechanism to fine-tune the clusters after the clustering stages by taking them through a sequence of operations.

Secondly, we present an adaptive alternating minimization approach (AAM) to specify a desired number of K output clusters if needed.

Finally, with the clusters refined, to prove refining the clusters may have an enhanced effect on deduplication, we conducted the same experiments as in [1] and compared the results to some existing models including the proposed human-machine based deduplication [1]. Our results showed improvements on the results obtained in [1] in terms accuracy and efficiency but incurred a little more crowdsourcing cost. The cost could be as a result of posting additional pairs of records to the crowd. Also, we compared the efficiency of the proposed work, in terms of modification of parameters against some existing work.

A. REFINING GENERATED CLUSTERS ANALYSIS AND PROCEDURE

To the best of our knowledge, no work has been done on refining chromatic correlation clusters [13] with crowdsourcing, where in the generation of clusters, a record pair (an edge) is used as pivot to construct triangular clusters and posted to the crowd. Randomly choosing an edge as pivot sometimes produces in-accurate deduplication, thus to solve this problem, we use a cluster refining algorithm with supplementary HITs after the cluster generation phases to lessen the metric cost $\Delta'(R)$. A few machine-based methods for refining clusters after they have been generated exist [26], [27] and [9], with only work [9] performing cluster refinement under crowdsourcing platform. All these methods, take as input, the similarity scores of all the pairs of records, implying the methods incur additional crowdsourcing overheads. To solve the huge cost problems, we propose in this work, how to fine-tune clusters with regards to the other edges not examined by the crowd, that will in the long run reduce the cost of the crowdsourcing. We adopt similar operations; merger and split (in this work, we shall term triangular split and merger) from work [9] and use the approach on a pair of records (an edge), instead of using a single record like in ACD. Basically, a triangular split, takes a one triangular cluster from the graph with clusters, and with while triangular merger fuses two triangular clusters to form one cluster. The goal of this work is to propose an mechanism, use it at the post clustering generation phases, which will improve the value

of the clusters by refining the clusters under crowdsourcing while not incurring huge crowd overheads.

1) ANALYZING THE COST-BENEFIT OF THE OPERATIONS

$$\Lambda'(R) = \sum_{r_i, r_j \epsilon R. i<j} x_{i,j} \cdot \left(1 - f_c\left(r_i, r_j\right)\right) + \sum_{r_i, r_j \epsilon R. i<j} \left(1 - x_{i,j}\right) \cdot \left(1 - f_c\left(r_i, r_j\right)\right) \quad (1)$$

if x_i and x_j belong to the same cluster, $x_{i,j} = 1$, otherwise $x_{i,j} = 0$.

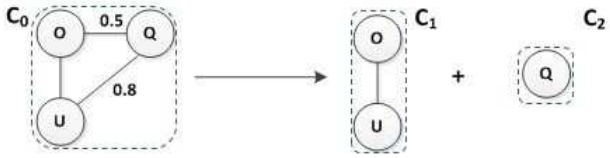


FIGURE 3. An example of a record split operation from a solitary cluster.

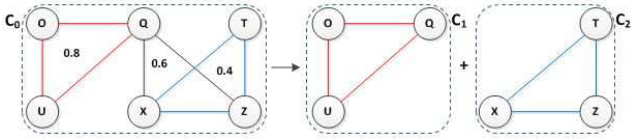


FIGURE 4. An example of triangular split operation.

In this section, the cost-benefit is analyzed using a pair of records as a pivot in the analysis. Suppose that we split a record r from a triangle cluster C_0 in fig. 3, the resulting clusters are not triangle, thus the split deviates from our chromatic correlation clustering purposes. In this work, we deal with a record pair or an edge, therefore we cannot split just a record. However, fig. 4 shows triangular split from a cluster C_0. Let $o_t s$ denote this triangular split operation, and $C' = C \setminus (r_i, r_j, r_k)$. According to equation 1 (Haruna et al. [1]), after the triangular split, $\Delta'(R)$ decreases by:

$$b(o_{ts}) = \sum_{tc' \epsilon C \setminus \{r_i, r_j, r_k\}} \left(1 - 2f_t\left(r_i, r_j, r_k\right)\right) \quad (2)$$

where $f_t(r_i, r_j, r_k)$ is the crowd triangular confidence score [1]. tc is the triangular cluster. We define $b(o_t s)$ as the benefit of $o_t s$. On the other hand, if we merge two clusters C_1 and C_2, then the triangular merger operation (denoted as $o_t m$) would reduce $\Delta'(R)$ by:

$$b(o_{tm}) = \sum_{tc'_1 \epsilon C_1, tc'_2 \epsilon C_2} \left(2f_t\left(r_i, r_j, r_k\right) - 1\right) \quad (3)$$

We refer to $b(o_t m)$ as the benefit of $o_t m$. $b(o_t s)$ and $b(o_t m)$ are illustrated with an example. We do not consider fig. 3 in this work. Fig. 4 demonstrates a triangular split operation $o_t s$ to split a triangular cluster $\langle t, x, z\rangle$ from a cluster $C_0 = \{o, u, q, x, t, z\}$. The scores indicates the triangular similarity score of each triangle in the cluster that would be obtained, if the pairs were issued to the crowd. Based on Equation 2, the benefit of this operation is 0.4. That is, by applying this triangular split operation, $\Delta'(R)$ can be reduced by 0.4.

After the triangular split operation, two clusters were obtained; $C_1 = \{o, u, q\}$ and $C_2 = \{t, x, z\}$. Fig.5 illustrates a triangular merger operation $o_t m$ to merge clusters $C_1 = \{m, n, o\}$ and $C_2 = \{p, q, r\}$. With Equation 3, it can be calculated that the benefit of this triangular merger operation is 0.8. After the triangular merger process, a new cluster $C_3 = \{m, n, o, p, q, r\}$ was generated.

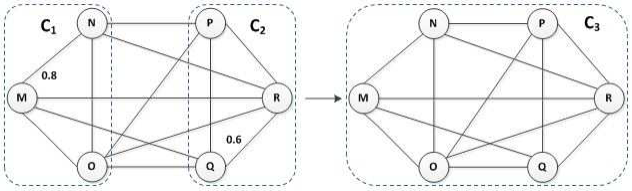


FIGURE 5. An example of a triangular merger operation.

Supposing a cluster set C is to be refined, the operation o with the largest positive benefit $b(o_t m)$ is identified and o applied to C. On the other hand, crowd confidence scores of the pairs of records, thus the triangular similarity scores as well are required to compute $b(o_t m)$, but may not readily be available because not all were posted to be examined. In this work, the crowd's confidence score as well as the triangular similarity score, $f_t(r_1, r_2, r_3)$ will both not be known. This may incur crowdsourcing cost, therefore depending on $b(o_t m)$ only will not work. We thus adopt an economical method as follows:

1. If the set of all possible (triangular split and merger) operations on a given set C of clusters and the set of operations in O whose benefits are known and are larger than zero, they are represented respectively by O and O^+. Then the operation in O^+ with the largest benefit is chosen and applied on C if and only if $O^+ \neq 0$.

2. Furthermore, if O^+, the benefit of each operation in O is estimated and an operation o is chosen. Afterwards, the exact benefit $b(o)$ of o (by calculating the triangular similarity scores after crowdsourcing the relevant record pairs with unknown similarity scores), to check if $b(o) > 0$, is computed. o is performed on C provided that $b(o) > O$, other o are disregarded.

In a triangular split operation analysis, assuming $b(o)$ is unidentified, we say that o is a triangular split process if from each cluster C, a triangle cluster $\langle r_i, r_j, r_k\rangle$ is removed. To calculate $b(o)$, equation 2 requires the $f_t(r_i, r_j, r_k)$ of all the triangle clusters in graph G with $tc' \epsilon C \setminus \{r_i, r_j, r_k\}$. We initiate a set D representing the unidentified edges that are needed by $b(o)$ to execute. To calculate an approximate figure for $b(o)$;

1. we assume for all deleted pairs in a triangle cluster, a crowd confidence score $f_c(r_1, r_2)$ and

2. with $f_c(r_1, r_2)$ of all the pairs assumed, we can calculate triangular crowd confidence scores [1], $f_t(r_i, r_j, r_k)$ of all triangular clusters that are possible to be formed, which are members of set D as $f_t(r_i, r_j, r_k) \epsilon D$.

Works [26], [29] proposed a solution whereby the crowd's confidence scores $f_c(r_1, r_2)$ were assumed to be equal to their respective machine-based similarity scores $f(r_1, r_2)$.

This machine-based method was proved to be incorrect [9] [30]. An enhanced and accurate result was presented where the pairs of records are sent to the crowdsourcing platform and their confidence scores f_c are used to generate a histogram. In the histogram, $f_c(r_1, r_2)$ maps onto $f(r_1, r_2)$, which is more accurate. In this extended work, we adopted the more accurate proposed solution and used it on the refining of the chromatic correlation clustering under the crowd, to present a better hybrid deduplication accuracy.

First and Foremost, from the solitary and compound clustering generation phases [1], the edges whose crowd confidence scores as well as the triangular similarity scores of the cluster generation phases that have already been computed are put in a set TC. We note that, that in work [1], it is a possible that some pairs of records would have a crowd confidence score but won't form a triangular cluster, because of the triangular similarity score threshold. For now, in this work, once a triangular cluster can be formed, irrespective of the triangular similarity scores threshold, we shall put the record pairs of the triangles in set TC. We build a histogram H having $m = 20$ buckets (same as [7], [9]). In building H we use the three records of machine-based technique's similarity scores that are relative to the pairs of records in a triangle cluster in set TC. Each bucket B will contain pairs of records that have the possibility of forming triangular clusters with each other. After H has been constructed, we calculate for the average f_t of the triangles in TC. In the cluster generation phases, not all pairs of records were posted to the crowdsourcing platform thus they do not have crowd confidence scores. So, when encountered with such record pairs, firstly, we ascertain the bucket B related to the record pair's machine-based score. Then with the record pair as an edge, we form a solitary cluster [1] by selecting other vertices to form a triangle. After that we approximate each of the $f_c(r_1, r_2)$ of the three pairs of records connected to the respective bucket. Any other pairs of records posted to the crowd is added to set TC and the histogram is updated. Now, given C and an estimated benefit of the operations, the goal to reduce $\lambda'(R)$ is obviously to opt for a triangular split or merger operation which will yield the greatest benefit. However, ACD [9] proved that it is not a wise choice when dealing with enormous number of pairs of records, because to calculate the precise value of $b(o)$ before using o on the clusters, may come with huge crowd overheads. Thus, we have to strike a balance to satisfy both the cost of the crowdsourcing and the approximated benefit of the triangular split or merger operations. Therefore, we select an o method that is likely to maximize $b^*(o)/c(o)$. The approximate value of $b(o)$ and the price of o are denoted by $b^*(o)$ and $c(o)$ respectively. When there is a triangular split from a cluster, o is defined as

$$c(o) = \left|\left\{tc' | tc' \epsilon C \setminus \{r_i, r_j, r_k\} \wedge f_t(r_i, r_j, r_k) \notin TC\right\}\right|, \quad (4)$$

where TC contains the pairs of records that forms triangular clusters and have crowd scores too. Implying that, to successfully calculate $b(o)$, $c(o)$ represents the edges needed to be sent to the crowdsourcing platform. Likewise, using the triangular merger operation o on two clusters, we have

$$c(o) = \left|\left\{(tc_1, tc_2) | tc_1 \epsilon C_1 \wedge tc_2 \epsilon C_2 \wedge f_t(r_i, r_j, r_k) \notin TC\right\}\right| \quad (5)$$

Naturally, choosing a triangular split or merger operation having the largest $b^*(o)/c(o)$, offers a stability between the crowd cost and trying to reduce $\lambda'(R)$. In this work we term $b^*(o)/c(o)$ as triangular benefit-cost ratio of o (benefit-cost ratio derived from ACD [9]).

A Graphical Example of Triangular Refinement Generation Steps

Generating an undirected graph G from the candidate set S yields the graph in Fig. 6a (Refer to [1] with regards to the generation of triangular clusters to obtain). The numbers represent the triangular similarity scores f_t of each triangular cluster and are as follows $\langle m, n, o\rangle = 0.8$, $\langle o, p, r\rangle = 0.6$, $\langle p, q, r\rangle = 0.4$ and $\langle s, t, u\rangle = 0.4$. These f_t are calculated after the crowd had examined each edge for their respective f_c. In Fig. 6a, $b^*(o) = b(o)$. From the crowd's confidence scores, the triangular similarity scores are thus generated and two clusters are formed, Fig. 6b. The two clusters are encompassed in boxes with dashed lines.

The cluster refinement procedure after inspecting the methods, then calculates their triangular benefit-ratio accordingly. The triangular split operation $o_t s$, removes triangular cluster $\langle p, q, r\rangle$ with the largest triangular benefit-cost ratio from the first cluster. Using equation 2 to calculate $b(o_t s)$ yields 0.2, thus the $o_t s$ ensues. Fig. 6c shows the triangular split operation. The refinement procedure again calculates the triangular benefit-cost ratio using equation 3 on the triangular merger operation $o_t m$ to fuse $\langle p, q, r\rangle$ with the second cluster. The triangular method goes on because after calculating for $o_t m$ the result is 0.4. Fig. 6d shows the final refinement clustering. The procedure then terminates.

B. AN ADAPTIVE ALTERNATING MINIMIZATION APPROACH

The human edge-pivot and compound clustering algorithms presented are parameter-free. That is they do not force any number of output clusters, instead the clustering is produced using information, local to the pivot edges. Like with most clustering algorithms, there could be the desire to possess a pre-specified K number of clusters. Based on an adaptive alternative minimization algorithm [16], we present an algorithm that builds clusters upon receiving a number of K output clusters as input. Algorithm **??** shows the proposed algorithm's pseudocode and is termed Adaptive Alternating Minimization (AAM). It produces a solution which assigns for every $x \epsilon V$ given the assignments of every other $y \epsilon V$ to the best cluster.

Definitions: A matrix notation was adopted to make the presentation well-defined. We denote matrices and vectors with capital boldface and small boldface letters respectively. Distinctively, $\mathbf{X}_{ij}$ and $\mathbf{x}(i)$ represent the coordinate (i, j) for matrix $\mathbf{X}$ and the i-th coordinate for the vector $\mathbf{x}$.

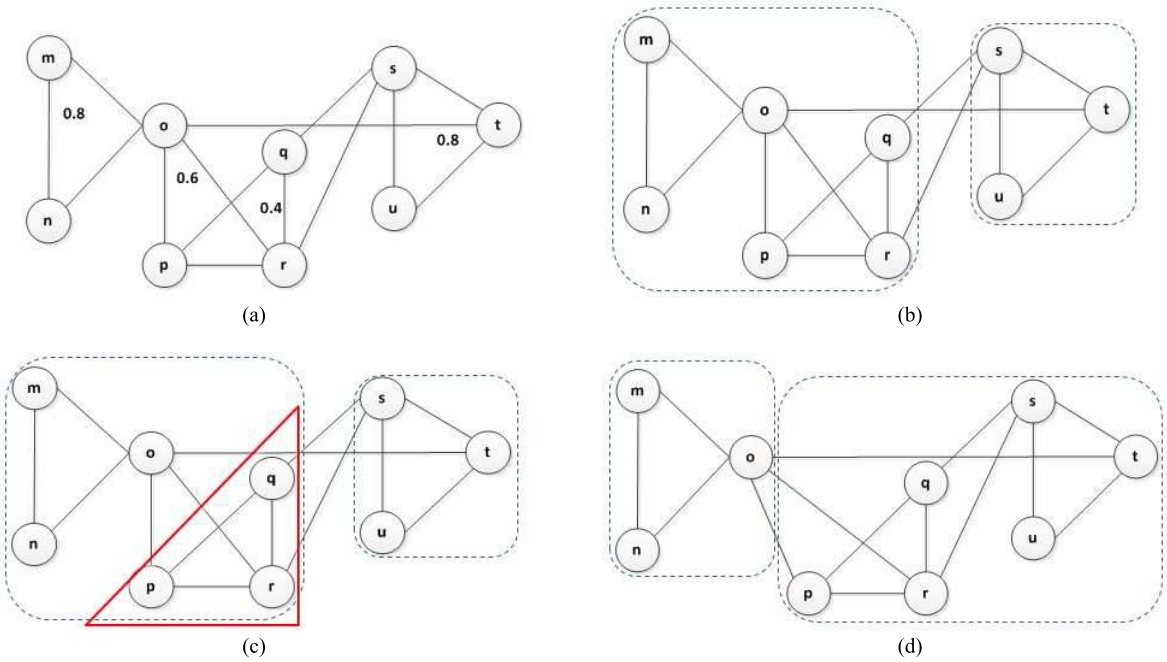


FIGURE 6. Triangular cluster refinement steps. (a) Undirected graph *G*. (b) Clustering generation phase. (c) Triangular split operation. (d) Triangular merger operation.

Algorithm 1

INPUT an undirected graph $G = (V_r \text{ and } E_s)$; K number of output clusters.
OUTPUT A set of clusters, $C : V \rightarrow \mathbb{N}$.
1: **Initialize** $A = [a_1, \ldots, a_N]$ at random
2: **Do**
3: according to Proposition 1, compute optimal a_x for all $x \epsilon V$
4: **Until A** has changed
end

For every object $x \epsilon V$, the problem's parameter space encountered, has a cluster assignment, given by a matrix $\mathbf{A}$ for every cluster $k \epsilon \{1, \ldots, K\}$. $\mathbf{A}_{kx} = 1$, when object x is put in cluster k, otherwise $\mathbf{A}_{kx} = 0$.Given that each object must be part of one and only one cluster, $\mathbf{A}$ is forced to contain of zeros with each column having a single 1. $\mathbf{a}_x$ denotes $\mathbf{A}$'s column relative to x.

For each $x \epsilon V$, a binary matrix set represents its input with matrix $\mathbf{Z}_x$. $\mathbf{Z}_x$ has a column denoted by $\mathbf{z}_{xy}$ that corresponds to the object $y \epsilon V$. Thus, every $\mathbf{Z}_x$ has zeros, with precisely one 1 on every column. In this problem formulation, the inputs are assumed to be represented by numerous bulky matrices. This idea, nonetheless, is noted to only be conceptual. The importance of this formulation permits to use linear-algebra methods to state the optimization process and also objective function. The matrices are not used in the implementation.

1) OPTIMAL CLUSTER ASSIGNMENT

If the fraction of items $y \epsilon V$ in cluster k is denoted by N_{xk}^{-} and N_{xk}^{+} then we have $N_{xk}^{-} = (\mathbf{A}\mathbf{Z}_x\mathbf{b})(k)$, if $\mathbf{z}_{xy}\mathbf{b} = 1$. Furthermore, we have $\mathbf{z}_{xy}\mathbf{C}\mathbf{a}_y = 1$ and can state that $N_{xk}^{+} = (\mathbf{A}\mathbf{w}_x)(k)$; $\mathbf{w}_x = \left[\mathbf{z}_{x1}^T\mathbf{C}\mathbf{a}_1 \ldots \mathbf{z}_{xn}^T\mathbf{C}\mathbf{a}_n\right]$ if $y \epsilon k$.

Proposition 1: For $x \epsilon V$ given $\mathbf{A}$, the optimal cluster assignment is $k^* = argmin_k\, N_{xk}^{-}$ - N_{xk}^{+}.

Proof:

$$\sum_{x,y} \mathbf{a}_x^T\mathbf{a}_y\left(1 - \mathbf{z}_{xy}^T\right) + \left(1 - \mathbf{a}_x^T\mathbf{a}_y\right)\left(1 - \mathbf{z}_{xy}^T\mathbf{b}\right)$$
$$= \sum_{x} \mathbf{a}_x^T\mathbf{A}\left(1 - \mathbf{w}_x\right) + \left(1^T - \mathbf{a}_x^T\mathbf{A}\right)\left(1 - \mathbf{Z}_x^T\mathbf{b}\right) \quad (6)$$

where $\mathbf{w}_x$ is defined as above, and for dimensional vector of all 1s, 1 is denoted by the $|V|$. Further, terms that correspond to a fixed $x \epsilon V$ are simplified to $\mathbf{a}_x^T\mathbf{A}^T\mathbf{Z}_a\mathbf{b} - \mathbf{a}_x^T\mathbf{A}\mathbf{w}_w + d_x$, where the "degree" of object x is represented by the constant $d_x = 1^T\,\mathbb{1}$ - $1^T\,\mathbf{Z}_x^T b$. Each x must be assigned to no more than one cluster. Therefore, minimizing the simplified expression can be done by assigning to cluster k, x, so that it is minimized to $(\mathbf{A}\mathbf{Z}_x^T b)(k)$ - $(\mathbf{A}\mathbf{w}_x)k = N_{xk}^{-}$ and N_{xk}^{+}.

2) COMPUTATIONAL COMPLEXITY

The computational complexity of Adaptive Alternating Minimization depends on the running time of the cluster step. The assignment of clusters require two sub-steps; for each vertex and cluster, evaluating $S_k - 2\,N_{xk}^{+}$ and N_{xk}^{0} can be achieved by traversing the input graph in a period of $O(m)$. Also, to select the bestfit cluster for a vertex, it takes a period of $O(Kn)$

TABLE 2. Characteristics of datasets and crowd answers.

Datasets	# of Records	# of Entities	# of Candidate Pairs S	Rate of Errors of Crowds (3w)	Rate of Errors of Crowds (5w)
Paper	997	191	29,581	23%	21%
Product	858	752	4,788	0.8%	0.2%
Restaurant	3,073	1,076	3,154	9%	5%

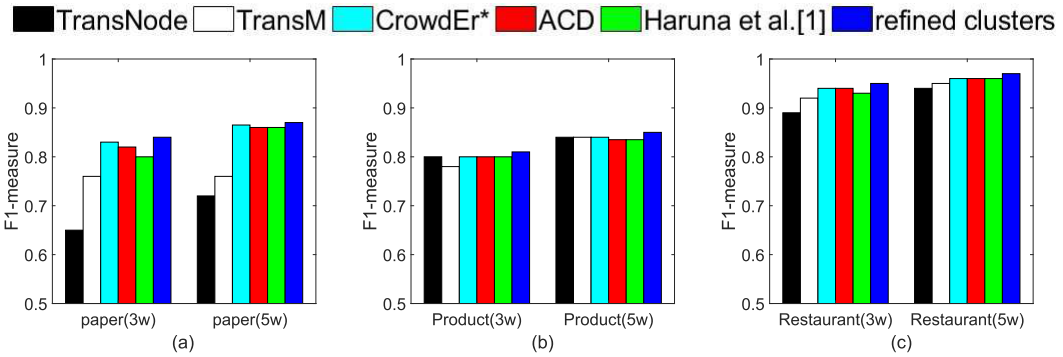


FIGURE 7. Experiments on deduplication accuracy. (a) Paper dataset. (b) Product dataset. (c) Restaurant dataset.

to examine all the clusters. In conclusion, the computational complexity of AAM for s number of iterations to consolidate, is $O(s(Kn + m))$.

III. EXPERIMENTS

In this section, we re-performed the same experiments from the previous work (Haruna *et al.* [1]), in terms of the accuracy of deduplication, the number of crowdsourced record pairs, and the total number of iterations it takes the proposed algorithm execute. The results were evaluated against Transnode [8], TransM [6], CrowdEr [5], ACD [9] and Haruna *et al.* [1].

A. SETUP OF EXPERIMENT

The same experiments setup as in Haruna *et al.* [1] was implemented, using the three benchmark datasets in deduplication literature; Paper [14], Product [10] and Restaurant [15]. Table 2 shows in each of the three datasets, the records and number of entities recorded. In this work CrowdEr is denoted as *CrowdEr** because a novel clustering algorithm [7], was used on CrowdEr as the authors did not specify the clustering algorithm used. The F-1 measure used in previous works like in Wang *et al.*0 [6] was also used in the experiments. It is worthy to note that:

1. To prove that refining clusters could have an effect in the deduplication method compared to Haruna *et al.* [1], we used the same threshold score $t = 0.4$ to generate the candidate pairs S, recorded in table 2. The Amazon Mechanical Turk (AMT) crowdsourcing platform was also maintained. Basically, the evaluation methods in CrowdEr [5], TransM [6] and ACD [9] was adopted in this work too.

2. We asked for and reused the experimental evaluations and answers of the results from the authors of CrowdEr [5] and ACD [9]. Each HIT consists of twenty record pairs. Each pair of record is made up of feedback from 3 humans denoted as $3w$. Also, like in other works, we used 5 humans ($5w$) on each pair to lessen the crowd's cost. Table 2 shows incorrect answers from the crowdsourcing platform given to us by the authors under both $3w$ and $5w$ settings. Under $5w$ setting, the results were accurate but at a relatively high overheads. Based on results form table 2, product and restaurant datasets have an error rate less than 10% with paper datasets slightly over 20%. It can be inferred that it is easier to deduplicate product and restaurant datasets compared to paper datasets.

B. EXPERIMENTAL EVALUATION WITH SOME HYBRID DEDUPLICATION ALGORITHMS

1) EXPERIMENTS ON DEDUPLICATION ACCURACY:

In this type of experiment, we performed tests on deduplication accuracy using the three datasets stated under both $3w$ and $5w$ settings. Already stated, we used the F-1 measure to calculate the number of record pairs posted to the crowdsourcing platform. Fig. 7 shows the results. In all the experiments and comparisons, our refined clusters approach, though slightly comparable to ACD, Haruna et al. and *CrowdEr**, provided the best deduplication accuracy. On paper dataset, TransM and TransNode offered the weakest accuracy. But on product and restaurant datasets, all the works compared had almost equal accuracy. Under all settings using all datasets Haruna et al. accuracy is lower than refined clusters. This proves that Haruna et al. provides lower-quality clustering results than refined clusters does, even though they both crowdsource almost equal number of record pairs. We can clearly see that, on Paper, *CrowdEr**, ACD, Haruna et al. and refined clusters have a higher accuracy under $5w$ settings but not TransNode and TransM. While on Product and Restaurant datasets, all the mechanisms offered a higher accuracy under $5w$ setting compared to $3w$.

2) EXPERIMENTS ON COST OF CROWDSOURCING:

Cost of Crowdsourcing is the price to be paid when the record pairs are posted to the crowd to examine for duplicates.

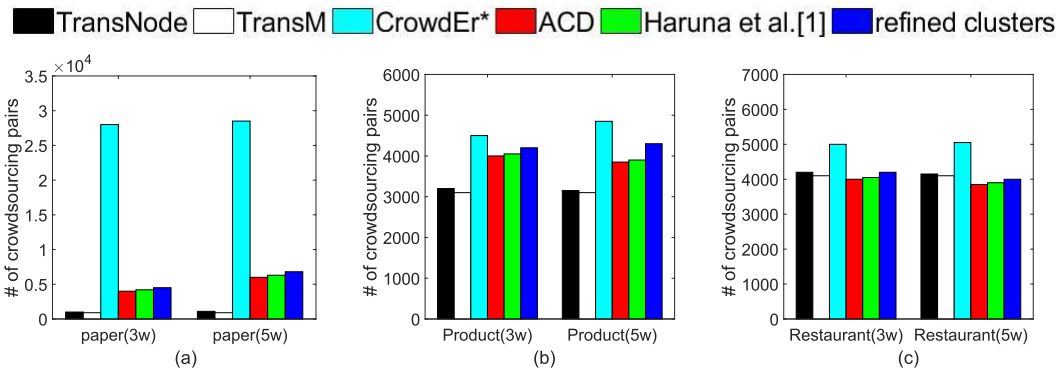


FIGURE 8. Experiments on crowdsourcing costs. (a) Paper dataset. (b) Product dataset. (c) Restaurant dataset.

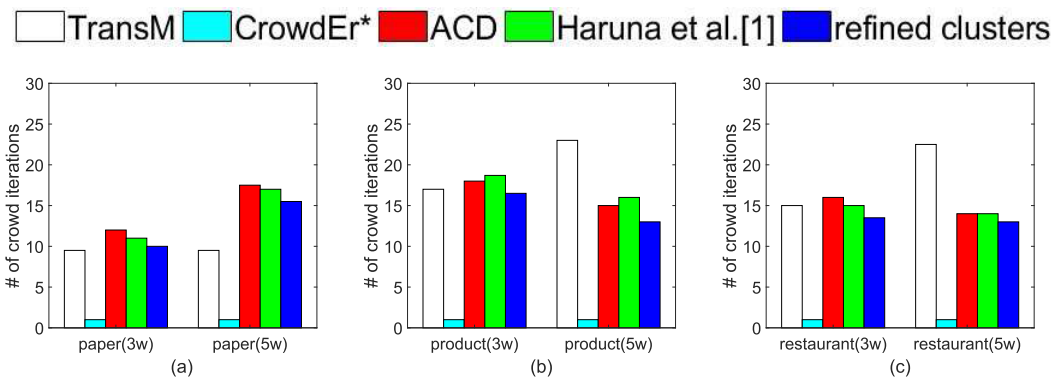


FIGURE 9. Experiments on the efficiency of crowdsourcing. (a) Paper dataset. (b) Product dataset. (c) Restaurant dataset.

It can be noted that, *CrowdEr**, under both settings using all three datasets, had the highest cost of crowdsourcing shown in fig. 8. This is because, in *CrowdEr** all pairs of records in the candidate set were issued to the corwdsourcing platform. The algorithm in this work, where the clusters were refined incurred little cost compared to Haruna *et al.* [1] and ACD. TransM and TransNode had the least cost on both product and paper, especially on the latter, in which they had minute cost. On restaurant, the two techniques had more cost than our refined work approach. The refined clusters mechanisms generated incurred more cost of crowdsourcing than all methods but *CrowdEr**. This is because, in this work, more HITs were used to examine additional record pairs to refine the clusters. While in other works, additional HITs were not needed for the deleted edges.

3) EXPERIMENTS ON THE EFFICIENCY OF CROWDSOURCING:

In fig 9, we present the efficiency of the crowd of all the algorithms being compared. Crowdsourcing efficiency is the number of iterations it takes the crowd to examine the pairs of records in S and for the algorithm to execute. In *CrowdEr**, all the pairs of records in S were issued one time to the crowdsourcing platform, thus it has an efficiency of only 1 under both $3w$ and 5 settings for all three datasets. TransM, ACD and Haruna *et al.* [1] method have some how a better efficiency than The cluster refinement algorithm proposed. This may be because after the refinement of the clusters, a lot of multiple clusters were generated and submitted to the crowd. The more multiple clusters generated, the fewer iterations required. The more singular clusters available, the more iterations needed. This work has a better efficiency *CrowdEr**. Under both settings, we the differences in efficiency are insignificant.

C. MODIFICATION OF PARAMETERS

In the modification of parameters experiments, the efficiency of crowd-pivot and PC-pivot algorithms in the work [9] as well as human-edge-pivot and compound clustering algorithms in this work were studied based on the effect of varying ε. Experiments showed similar results from the $3w$ and 5 settings, thus we only explain using $5w$ setting. The number of crowd iterations needed by crowd-pivot, PC-pivot algorithms, human-edge-pivot and compound clustering, when ε varies were compared and are shown in Figs. 10(a), (b) and (c). The compound clustering algorithm presented in this work incurs a much lesser number of crowd iterations than PC-pivot, crowd-pivot and human-edge-pivot do. On all datasets, when ε is set to 0.1, the reduction of crowd iterations is more noteworthy. However, from 0.1 to 1.0 the crowd iterations is

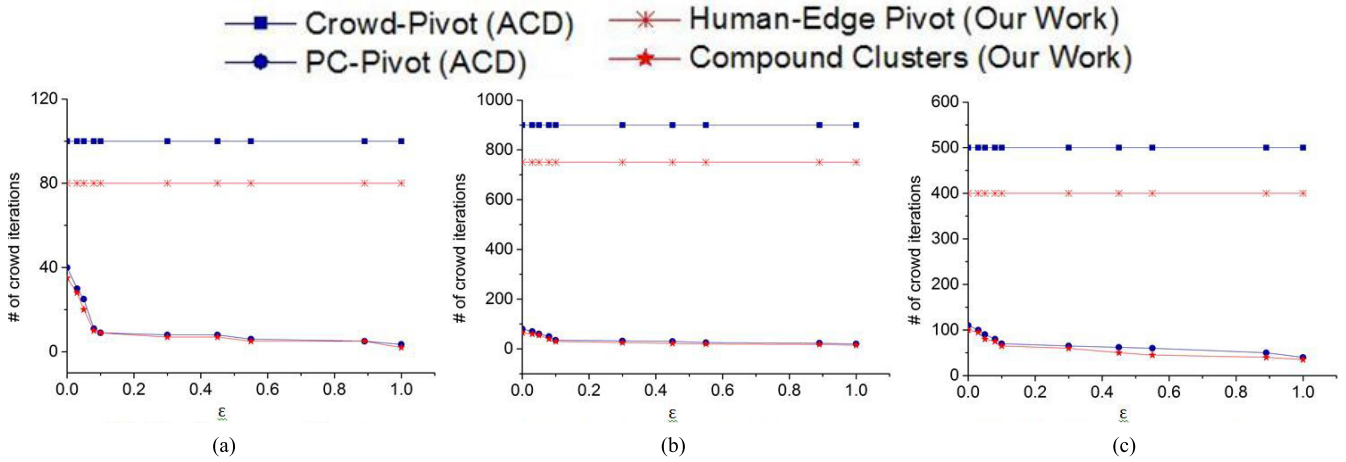


FIGURE 10. **Impacts of varying ε.**

more steady. In increasing ε, there is a decrease in number of crowd iteration. This is due to the fact that, with an increase in ε, compound clustering is able to form more triangle clusters from one single edge pivot, thus the clustering process is quicker.

IV. CONCLUSION

In this work, we extended the work on our previous hybrid data deduplication method, Haruna *et al.* [1]. In work [1], some of the record pairs, during the cluster generation phases were not examined by the crowdsourcing platform. Thus in this work, we proposed a refining of clusters technique during the clustering generation stages to fine-tune the clusters. Also, we proposed and algorithm that forces any number of output clusters. From the experimental results and evaluations, when clusters are refined, the data deduplication method, has a better accuracy, and higher efficiency and incurs low crowd cost when compared to other existing hybrid deduplication methods. In future, we can consider using 7 humans ($7w$) on each pair to further lessen the crowd's cost.

REFERENCES

[1] C. R. Haruna, M. Hou, M. J. Eghan, M. Y. Kpiebaareh, and L. Tandoh, "A hybrid data deduplication approach in entity resolution using chromatic correlation clustering," in *Proc. Int. Conf. Frontiers Cyber Secur.* Singapore: Springer, 2018, pp. 153–167.

[2] A. K. Elmagarmid, P. G. Ipeirotis, and V. S. Verykios, "Duplicate record detection: A survey," *IEEE Trans. Knowl. Data Eng.*, vol. 19, no. 1, pp. 1–16, Jan. 2007.

[3] A. K. Jain, M. N. Murty, and P. J. Flynn, "Data clustering: A review," *ACM Comput. Surv.*, vol. 31, no. 3, pp. 264–323, Sep. 1999.

[4] W. E. Winkler, "Overview of record linkage and current research directions," Bureau Census Stat. Res. Division, Washington, DC, USA, Res. Rep. 2006-2, 2006.

[5] J. Wang, T. Kraska, M. J. Franklin, and J. Feng Crowder, "Crowdsourcing entity resolution," *Proc. VLDB Endowment*, vol. 5, no. 11, pp. 1483–1494, Jul. 2012.

[6] J. Wang, G. Li, T. Kraska, M. J. Franklin, and J. Feng, "Leveraging transitive relations for crowdsourced joins," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, Jun. 2013, pp. 229–240.

[7] S. E. Whang, P. Lofgren, and H. Garcia-Molina, "Question selection for crowd entity resolution," *Proc. VLDB Endowment*, vol. 6, no. 6, pp. 349–360, Apr. 2013.

[8] N. Vesdapunt, K. Bellare, and N. Dalvi, "Crowdsourcing algorithms for entity resolution," *Proc. VLDB Endowment*, vol. 7, no. 12, pp. 1071–1082, Aug. 2014.

[9] S. Wang, X. Xiao, and C. H. Lee, "Crowd-based deduplication: An adaptive approach," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, May 2015, pp. 1263–1277.

[10] (Jun. 2018). *Product Data Set*. [Online]. Available: http://dbs.uni-leipzig.de/Abt-Buy.zip

[11] R. J. Bayardo, Y. Ma, and R. Srikant, "Scaling up all pairs similarity search," in *Proc. 16th Int. Conf. World Wide Web*, pp. 131–140, May 2007.

[12] S. Chaudhuri, V. Ganti, and R. Kaushik, "A primitive operator for similarity joins in data cleaning," in *Proc. ICDE*, Apr. 2006, p. 5.

[13] F. Bonchi, A. Gionis, F. Gullo, C. E. Tsourakakis, and A. Ukkonen, "Chromatic correlation clustering," *ACM Trans. Knowl. Discovery Data*, vol. 9, no. 4, p. 34, Jun. 2015.

[14] (Jun. 2018). *Paper Data Set*. [Online]. Available: http://www.cs.umass.edu/ mccallum/data/cora-refs.tar.gz

[15] (Jun. 2018). *Restaurant Data Set*. [Online]. Available: http://www.cs.utexas.edu/users/ml/riddle/data/restaurant.tar.gz

[16] U. Niesen, D. Shah, and G. Wornell, "Adaptive alternating minimization algorithms," in *Proc. IEEE Int. Symp. Inf. Theory*, Jun. 2007, pp. 1641–1645.

[17] L. M. Abualigah, A. T. Khader, and E. S. Hanandeh, "A new feature selection method to improve the document clustering using particle swarm optimization algorithm," *J. Comput. Sci.*, vol. 25, pp. 456–466, Mar. 2018.

[18] L. M. Abualigah, A. T. Khader, and E. S. Hanandeh, "A novel weighting scheme applied to improve the text document clustering techniques," in *Innovative Computing, Optimization and Its Applications*. Cham, Switzerland: Springer, 2018, pp. 305–320.

[19] L. M. Abualigah, A. T. Khader, and E. S. Hanandeh, "A combination of objective functions and hybrid Krill herd algorithm for text document clustering analysis," *Eng. Appl. Artif. Intell.*, vol. 73, pp. 111–125, Aug. 2018.

[20] L. M. Abualigah, A. T. Khader, and E. S. Hanandeh, "Hybrid clustering analysis using improved krill herd algorithm," *Appl. Intell.*, vol. 48, no. 11, pp. 4047–4071, Nov. 2018.

[21] L. M. Abualigah, A. T. Khader, E. S. Hanandeh, and A. H. Gandomi, "A novel hybridization strategy for krill herd algorithm applied to clustering techniques," *Appl. Soft Comput.*, vol. 60, pp. 423–435, Nov. 2017.

[22] L. M. Abualigah and A. T. Khader, "Unsupervised text feature selection technique based on hybrid particle swarm optimization algorithm with genetic operators for the text clustering," *The J. Supercomput.*, vol. 73, no. 11, pp. 4773–4795, Nov. 2017.

[23] P. Christen, "Data matching: Concepts and techniques for record linkage, entity resolution, and duplicate detection," in *Data-Centric Systems and Applications*. Berlin, Germany: Springer, 2012.

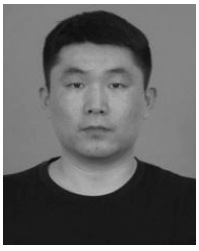
[24] A. Gionis, H. Mannila, and P. Tsaparas, "Clustering aggregation," *ACM Trans. Knowl. Discovery Data*, vol. 1, no. 1, p. 4, Mar. 2007.

[25] N. Bansal, A. Blum, and S. Chawla, "Correlation clustering," *Mach. Learn.*, vol. 56, nos. 1–3, pp. 89–113, 2004.

[26] N. Ailon, M. Charikar, and A. Newman, "Aggregating inconsistent information: Ranking and clustering," *J. ACM*, vol. 55, no. 5, p. 23, Oct. 2008.
[27] A. Goder, and V. Filkov, "Consensus clustering algorithms: Comparison and refinement," in *Proc. Meeting Algorithm Eng. Expermiments*, Jan. 2008, pp. 109–117.
[28] S. Vega-Pons and J. Ruiz-Shulcloper, "A survey of clustering ensemble algorithms," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 25, no. 3, pp. 337–372, 2011.
[29] Y. Amsterdamer, Y. Grossman, T. Milo, and P. Senellart, "Crowd mining," in *Proc. ACM SIGMOD Int. Conf. Manage. Data* , Jun. 2013, pp. 241–252.
[30] A. Arasu, M. Götz, and R. Kaushik, "On active learning of record matching packages," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, Jun. 2010, pp. 783–794.

CHARLES ROLAND HARUNA received the bachelor's degree in computer science from the Kwame Nkrumah University of Science and Technology, in 2009, and the master's degree in computer and science and engineering from the University of Electronic Science and Technology of China, where he is currently pursuing the Ph.D. degree in computer science and engineering. He is currently a Research Assistant with the Department of Computer Science and Information Technology, University of Cape Coast. His research interests include distributed data storage, data cleaning, and privacy preservation.

MENGSHU HOU received the Ph.D. degree in computer applications from the University of Electronic Science and Technology of China, Chengdu, in 2005, where he is currently a Professor with the School of Computer Science and Engineering. His research interests include wireless sensor networks, distributed computing, and trusted P2P computing.



RUI XI received the M.E. degree in computer science and engineering from the University of Electronic Science and Technology of China, Chengdu, China, where he is currently pursuing the Ph.D. degree. His current research interests include indoor localization, activity recognition, deep learning.

MOSES JOJO EGHAN received the M.Phil. and Ph.D. degrees in physics from Kwame Nkrumah University of Science and Technology, Ghana. He is currently a Professor and the Dean of the University of Cape Coast, where he was previously the Head of the Department of Computer Science and Information Technology. His research interests include data mining, digital image processing, laser spectroscopy, optical properties of materials, opticsplant, and fluorescence.

MICHAEL Y. KPIEBAAREH received the M.Sc. degree in computer science and engineering from the University of Electronic Science and Technology of China, Chengdu, China, where he is currently pursuing the Ph.D. degree in computer science. His current research interests include recommendation systems for configurable product design engineering and machine learning for mass customization.

LAWRENCE TANDOH received the bachelor's and M.Sc. degrees in computer science and engineering from the University of Development Studies, in 2016. He is currently pursuing the Ph.D. degree in computer science and engineering with the University of Electronic Science and Technology of Chengdu. He is currently a Senior Research Assistant with the University of Development Studies. His research interests include data compression techniques and network security.

BARBIE EGHAN-YARTEL received the B.Sc. and the Post Graduate Diploma degrees in architecture. She is currently pursuing the master's degree in information technology. She is currently an Assistant Architect with the Directorate of Physical Development and Estate Management Section, University of Cape Coast, Ghana. She has a keen interest in data mining.

MAAME G. ASANTE-MENSAH received the bachelor's degree in information technology from the University of Cape Coast, Ghana, in 2011, and the master's degree in computer science and engineering from the University of Electronic Science and Technology of China, Chengdu, Sichuan, China, in 2017. She is currently with the Skolkovo Institute of Science and Technology, Moscow, Russia. She is also a Research Assistant with the Department of Computer Science and Information Technology, University of Cape Coast. Her research interests include medical data analysis using deep learning technologies, machine learning, and tensor decomposition techniques.