A Multi-Objective Clustering Ensemble Approach for Crowdsourced Clustering

Sujoy Chatterjee

Department of Informatics, University of Petroleum and Energy Studies (UPES), Dehradun, India

Abstract

Clustering ensemble approach aims to obtain a single clustering of the dataset by reaching a consensus between the base clustering solutions. The ultimate objective of the consensus solution is to produce a better clustering from a diverse set of clustering. In real-life, it can be observed that there are some hard image grouping tasks that cannot be easily achieved by a computer. However, if manual image annotations can be collected then these hard tasks can be accomplished very easily. Therefore, outsourcing the image clustering task to the crowd workers to cluster the multiple images depending upon similar features, can be an effective mechanism to complete the task in a time efficient manner. In this paper, we leverage the power of crowd to annotate the images and clustering solutions are obtained from them. Thereafter a multi-objective clustering ensemble method is introduced to make a consensus from multiple crowd clustering solutions. Moreover, this ensemble method is applied on the partial crowdsourced clustering solutions and it derives the actual number of clusters automatically from a set of diverse clustering solutions. The similarity between two clustering solutions is computed using Adjusted Rand Index and Jaccard Index. The performance of the proposed algorithm is demonstrated by comparing it with other well-known existing cluster ensemble algorithms over different crowdsourced clustering datasets.

Keywords

Clustering Ensemble, Adjusted Rand Index, Multi-objective Optimization

1. Introduction

Clustering groups a set of objects in such a way that the objects within a group are similar to one another and are dissimilar to objects of other groups [1, 2, 3]. The greater is the similarity within a group and the greater is the dissimilarity between groups, the better is the clustering. Over the years, it is already established that clustering plays an important role in the fields of pattern recognition, information retrieval, machine learning and data mining to solve various real-life problems.

Although numerous research has already been carried out in this domain, still, there are substantial limitations in the majority of clustering techniques. Most of the clustering methods are also very sensitive to the initial clustering settings. Another extremely important issue in cluster analysis is the validation of the clustering results, that is, how to impose importance about the significance of the clusters provided by a particular clustering technique. A number of cluster validity indices [4, 5, 6, 7, 8, 9, 2] which are used to measure the quality of clustering

VLDB 2021 Crowd Science Workshop: Trust, Ethics, and Excellence in Crowdsourced Data Management at Scale, August 20, 2021, Copenhagen, Denmark

sujoy.chatterjee@ddn.upes.ac.in (S. Chatterjee)

results. Sometimes the original dataset is not available, thus it becomes difficult to compute the internal cluster validity criteria. Still, effective use of the cluster validity indexes is not the definite solution. So, an ensemble of clustering solutions is needed in order to achieve a better clustering solution from these diverse clustering solutions.

During the last decade, a spectrum of clustering ensemble algorithms have been proposed to solve different ensemble problems. On the other hand, in real-life, there are numerous image clustering tasks that are impossible for a machine to perform perfectly in a time efficient manner. Basically, sometimes understanding the sentiment of the images is not very easy for a machine. Rather, if the problem can be outsourced to the crowd [10, 11, 12, 13] and opinions are collected from them then it can be resolved in a much more efficient manner. Furthermore, in the crowdsourcing domain the opinions are collected from the crowd workers independently, but there is a chance of bias via their consultation over social media. So instead of taking all the opinions of the crowd workers, if the partial opinions of the crowd workers are collected from their answers, then the biases can be removed partially. In traditional clustering solutions it is assumed that all the objects are clustered by all the clustering algorithms. On the other hand, in partial clustering solutions, it is assumed that all the objects are not visible to all the crowd workers. Therefore, only a partial set of objects are clustered by them. Now the objective is to find the consensus clustering solution from the partial crowdsourced clustering solutions. However, in this work, this information (i.e., which solution will be taken or not) was not disclosed to the crowd workers while collecting the solutions from them. Rather, we have designed a platform and posted some tricky images in such a fashion so that each crowd worker provides a clustering solution for all the objects. After that, some solutions are discarded randomly for each crowd worker in order to remove the biasness. Therefore, in this work all the opinions of the crowd workers are not taken as input. Rather some of the missing clustering solutions are predicted based on the similar set of crowd workers. So the crowd workers are totally unaware about the solutions that are to be discarded thus the biases can be removed efficiently. Here, an online platform is designed to collect the clustering solutions from the crowd workers over some ambiguous images and a multi-objective clustering ensemble is proposed [14, 15, 16] to find the consensus clustering from multiple input clustering solutions. Initially, as there are some missing values in the crowd solutions, therefore, the missing opinions regarding the clustering are predicted first. Then, the algorithm is applied on crowd based clustering solutions to achieve a more robust clustering solution. The performance and efficacy of the method proves that better utilization of enormous human power can easily solve the image clustering task. Moreover, it generates the number of clusters automatically from multiple diverse crowdsourced solutions which is a limitation in most state-of-the-art clustering ensemble approaches dealing with crowdsourced solutions.

This paper is organized as follows: Section 2 represents the state-of-the-art approaches in context of this topic. The problem formulation, proposed model, and proposed multi-objective approach are described in section 3, section 4 and section 5, respectively. Section 6 discusses the experimental design and results. Finally in section 7 the conclusion is drawn followed by some insights on future prospects of the proposed work.

2. Related Works

Over the years, researchers have investigated the techniques of combining the predictions of multiple classifiers to produce a single classifier. The resulting ensemble is generally more accurate than any of the individual classifiers making up the ensemble. One study, presented three clustering ensemble methods CSPA, MCLA and HGPA [17]. The consensus partition is defined as the partition that shares most information with all partitions in the dataset. In CSPA, a $n \times n$ co-association matrix is constructed. Then, the graph is partitioned using METIS [18] algorithm to obtain the consensus partition. HGPA partitions the hypergraph directly, by eliminating the minimal number of hyperedges. For partitioning the graph, the minimal cut algorithm HMETIS [19] is used in this context. In MCLA, first of all the similarity between two clusters is defined using the Jaccard index [20] in terms of the amount of objects grouped in both clusters. Then, a similarity matrix between clusters is formed. After that, this graph is partitioned using METIS [18] algorithm and the obtained clusters are called meta-clusters. To find the final partition, each object is assigned to the meta-cluster to which it is assigned more number of times. Another study uses cumulative voting method for relabeling and tackles the ensemble problem for variable number of clusters with linear computational complexity [21].

Over the last decade, some studies have been performed to obtain better clustering using plurality voting [22, 23]. In these methods, it is assumed that the number of clusters in each partition is fixed. The label correspondence problem is solved using the Hungarian method. Then, a plurality voting procedure is applied to obtain the winner cluster for each object.

Another adaptive voting based method proposed in [24] finds the consensus clustering where the votes are updated in order to maximize an overall quality measure. This method allows the combination of clustering from different locations, i.e. all the data does not have to be collected in one central workstation. The idea is to make different clustering from different portions of the original data in separate processing centers. Afterwards, the consensus clustering is obtained through a voting mechanism.

Another work [25] presented a mixed-membership model for learning cluster ensembles and is applicable to all the primary variants of the basic cluster ensemble problem. It can solve the basic cluster ensemble problem using a Bayesian approach, that is, by effectively maintaining a distribution over all possible consensus clustering. It treats all the input clustering results for each object as a feature vector with discrete feature values, and learns a mixed-membership model from such a feature representation. One most promising approach proposed in [26] that introduces a novel consensus function namely, weak evidence clustering accumulation (WECA). It is established that the four variants of the method outperform other well-known baseline approaches for different datasets. The first three variants are basically agglomerative clustering algorithms like average linkage (AL), single linkage (SL), and complete linkage (CL). The last variant is a graph partitioning based method namely, GP-MGLA. Although there are numerous clustering ensemble algorithms, still, there are limitations to finding the number of clusters automatically from the clustering solutions. Moreover, this problem becomes hard when the base clustering solutions are generated from the crowd and there is a possibility that all the objects are not clustered by all the crowd workers. Thus motivated by this we introduce a crowdsourcing based platform to collect clustering solutions from the crowd workers and generate a set of partial clustering solutions from that set of solutions. Finally, we use a

multi-objective optimization algorithm to find out the most promising consensus clustering solution.

3. Problem Formulation

Suppose, Z be the set of o data objects, i.e., $Z = \{z_1, z_2, \dots, z_o\}$ and there are p crowd workers. Here every crowd worker provides the individual clustering solutions. So, $E = \{e_1, e_2, \dots, e_p\}$ denotes a set of p input clustering solutions obtained from them. So, the objects are partitioned into n clusters denoted as $C = \{c_1, c_2, \dots, c_n\}$. The objective is to derive a consensus clustering τ from these multiple base clustering solutions $E = \{e_1, e_2, \dots, e_p\}$ such that closeness between the τ and all the base clustering solutions E produced by crowd is maximized. Note that, the label assigned by any clustering solution e_a on a particular object can be denoted as L_{e_a}, where $\forall e_a \in E$, $L_{e_a} \in C$.

4. Proposed Model

The overall ensemble framework consists of three phases. Initially, the crowd based ambiguous image clustering solutions are obtained from an open platform (as shown in Fig. 1). Then to alleviate the biases due to the social connectivity of the crowd workers we remove some of the grouping from each crowd workers. Thus it generates a set of partial clustering solutions. This is like the assumption that all the images are not clustered by all the crowd workers. Then these missing values of a particular crowd worker's solution are predicted based on his/her similar crowd workers. Finally, a multi-objective optimization method is applied to generate the consensus clustering solution. In this framework, there are three main phases, i.e., missing value prediction, label transformation and determination of good clusters in a clustering solution. Note that, label transformation (i.e., to make correspondence between the labels of clustering solutions) is the very first step of the proposed model. However, the motivation of label transformation is very much dependent on the missing value prediction of the crowdsourced clustering. Hence we discuss the missing value prediction first to make a better understanding about the requirement of label transformation. Then how the quality clustering solution is chosen is discussed subsequently. After that, a multi-objective optimization method is applied on the solutions. These are explained in the following subsections.

4.1. Missing Opinion Prediction

In this proposed model, although the opinions are collected from the crowd workers, all the opinions are not initially considered to derive the consensus. The reason is that in the crowdsourcing market there can be the scenario that some workers may manipulate opinions over some questions in which they are less confident. The social connectivity of the crowd workers can distort their opinions although these are taken independently. Therefore, while obtaining the opinions from them it is not disclosed which images are important to them. Rather, after obtaining all the opinions from them some opinions are randomly removed in order to reduce

the bias caused from the social connectivity. Thus this generates a set of partial clustering solutions from the crowd workers.

Now, to predict the full clustering solution from the partial clustering solutions traditional matrix factorization type of methods cannot be applied here. As the clustering problem basically deals with the labeling of the clustering solutions therefore the labels are basically the discrete values. So output of matrix factorization generates fraction values that cannot be treated as the labeling of a particular object. Therefore, the missing values of a particular crowd worker are predicted based on the similar crowd workers of his. In this process, the set of similar workers (neighbourhood) for the same worker can be different for different questions. To find the similar workers, the crowd workers having the same opinions are searched first and the absolute difference of his opinion from their opinions are computed. In this purpose, only the common opinions in which they respond can be taken into account to find the similar workers. But this may lead to losing some information as there is a need to find the similar crowd workers based on the missing opinion of a particular crowd worker. So for each crowd worker, all the crowd workers are considered depending on their similarity of opinions and they are sorted in ascending order. The reason is that the less difference means their similarity is higher. Finally, to predict any missing value the similar workers (except those who have not provided the opinion for the said object, we are interested to predict the missing value) are selected.

Here the crowd workers have no knowledge about which opinions will be taken as original and which opinions will be predicted based on their similar workers. Thus the prediction of missing values can be helpful in order to remove the biasness generated in the crowdsourcing market. Now, as in the clustering solutions there is no correspondence between the labeling of the objects, therefore, missing value prediction based on similar workers requires making the correspondence between them prior to applying it. Therefore in this context, to predict the missing value of the crowd workers' clustering solutions the label transformation is needed with an aim to make correspondence between them.

4.2. Label Correspondence

The objective of our cluster ensemble technique is to determine which cluster label should be associated with each object in the consensus partition. To do this we have analyzed how many times an object belongs to one cluster (recognized by the label associated with it) and the consensus is obtained through a voting process. Hence, it is imperative that the labels used by the input clustering solutions must be standardized.

The labeling of input clustering solutions may vary based on the different clustering algorithms, or different runs of the same algorithm. Cluster labels are very symbolic, i.e., two clustering solutions of a given dataset that have the same partition but different labels might appear different. Similarly, in the crowdsourcing domain also, the labeling schemes adopted by each crowd worker are different. Using input clustering solutions in our algorithm without addressing the label correspondence problem can produce incorrect results as there is a need to make missing value predictions here. In order to solve this problem we need to standardize all the input clustering labels according to a particular labeling standard. Here we choose the standard label to be that of the reference partition (a partition which is most similar to the rest of the partitions) by using Adjusted Rand Index (ARI) [2, 27] as a similarity measure. After

that, according to the reference partitions all the other clustering solutions are re-labeled. To explain in more depth, consider an example of two clustering solutions e_a and e_b (over 9 objects) such that $L(e_a) = \{1, 1, 1, 2, 2, 2, 3, 3, 3\}$ and $L(e_b) = \{2, 2, 2, 3, 3, 3, 1, 1, 1\}$. Here we can see that the objects 1, 2, 3 of clustering solution e_a belong to one cluster, objects 4, 5, 6 belong to a second cluster and objects 7, 8, 9 belong to a third cluster. e_b also represents the same clustering, but because of different labeling it appears different.

To carry out the relabeling process, we explain this with an example. Suppose {2, 2, 2, 1, 1, 3} and {3, 2, 3, 4, 4, 1} are two clustering solutions. Now we need to transform the label of the second clustering solution based on the first clustering solution. Therefore, first it is checked how many objects of clustering solution 2 with labels '1', '2', '3' and '4' are transformed into labels '1', '2' and '3' of clustering solution 1. At the end, the majority voting is applied among these values and the maximum value is treated as the final label of clustering solution 2. So applying this method, the labeling of the second clustering becomes {2, 2, 2, 1, 1, 3}. In case of ties, randomly any label is chosen as the final label. Thus in this similar way all the labels of the clustering solutions are transformed.

4.3. Quality of clusters

In clustering ensemble solutions, the quality of the clustering denotes how similar the clustering solutions are with respect to the base clustering solutions. Now in this context, normally, there can be two types of measures i.e., based on the internal cluster validity indices and external cluster validity indices. However, often there is a need to obtain the original dataset to measure the internal cluster validity indices. But in many situations the original datasets are not used; rather only the base clustering solutions are used in order to measure the goodness of the clustering solutions.

In the majority of clustering ensemble methods, the quality of the clustering solutions is measured using the ARI that computes the pair-wise similarity of the clustering solutions. In this measure if the two clustering solutions have high ARI values then both of them are treated as similar solutions. Basically, in this situation, the pairwise object being in the same cluster is considered but the quality of the clusters of a particular solution are not taken into account. Moreover, all the clusters in a particular clustering solution are treated as independent and as equally good although that is not true. Therefore, we should also rely on the fact that a cluster is said to be good if the internal bonding of these objects of a particular cluster remains the same in most clustering solutions. This means if the objects in a cluster of a particular clustering solution are the members of several other clusters in other clustering solutions then this cluster cannot be considered as a quality cluster.

In this context, to measure the homogeneous property between two clustering solutions in terms of quality of constituent clusters, Jaccard distance measure [20] is used. In this respect, let there are two clustering solutions i.e., clustering solution 1 = {1,1,1,2,2,2,3,3,3} and clustering solution 2 = {1,1,1,1,2,2,3,3,3}. Now there are three clusters in both of the solutions. So the Jaccard distance between the pair-wise clusters are computed first and finally the sum of the distances of a particular clustering solution are calculated here. To elaborate it, as there are three clusters in both of the solutions, the first cluster of clustering solution 1 is compared with each of the clusters of the clustering solution 2. Similarly, for the rest of the other clusters of

clustering solution 1, are compared with all the clusters of clustering solution 2. In this way, the summation value of the Jaccard distance is computed for the clustering solution 1. Thus if the individual clusters become stable (i.e., becomes homogeneous with minimum Jaccard distance) in most of the clustering solutions then it can be considered as a good cluster. Thus if any particular clustering solution has good quality constituent clusters then it can be treated as a good clustering solution. In this way, the quality of the individual constituent clusters are also taken into account to quantify the goodness of the whole clustering solution. This cluster-wise Jaccard similarity measure is integrated with ARI as objective functions in order to achieve a better consensus solution. To compute the Jaccard similarity between the clusters of any two clustering solutions, the clustering solutions are transformed as binary format as shown in [17].

4.4. Objective functions

In this model, the objective functions used are similarity of ARI values with respect to the clustering solutions. Here the higher value means the better clustering solutions. To find the goodness of the different clusters in a particular clustering solution Jaccard similarity index is used. The average values based on these two factors are considered as the first objective function. Again, we have computed the standard deviation of the obtained similarity values in order to remove biases towards any particular clustering solution. Therefore, the first objective function is maximization of average value of Jaccard similarity and ARI. On the other hand, the second objective function is minimization of standard deviation of these similarity values obtained by Adjusted Rand Index.

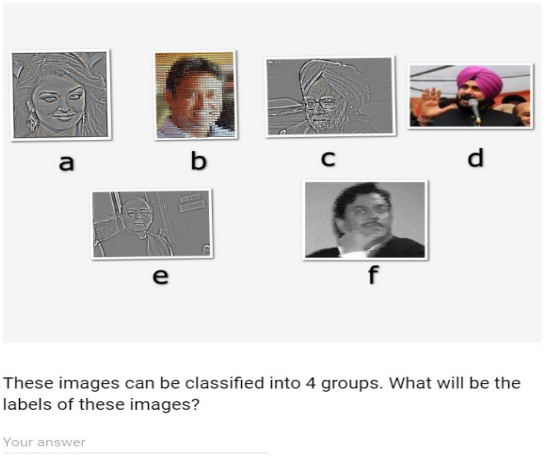

Figure 1: Snapshot of the second question posted to crowd workers.

5. Proposed Multi-objective Optimization Algorithm

In this section, we describe the proposed NSGA-II Multi-objective optimization algorithm [28] with an aim to produce non-dominated Pareto optimal ensemble solutions.

- Encoding Scheme: Chromosomes are represented by integers and it is denoted as $\{r_1, r_2, \ldots, r_n\}$. Here the same labeling means the objects are in same clusters. So if the labeling of two chromosomes are $\{1, 1, 1, 2, 2, 3, 3, 3\}$ and $\{2, 2, 2, 1, 1, 3, 3, 3\}$ that means the first three objects are in one cluster, 4^{th} and 5^{th} objects are in the second cluster and rest three are in other cluster.
- Initial Population: In the initial population, the clustering solutions after filling up the missing values obtained from the crowd workers are used. In addition to that, some random solutions are also generated keeping the number of clusters within the same range of clusters like the crowd workers solutions.
- Selection: The selection of chromosomes happens based on the survival of fittest concept. Here, the selection is done depending upon the crowded binary tournament selection strategy.
- Crossover: This is a probabilistic process that is used to exchange the information between two parent chromosomes. Here, we use the crossover based on the method described in [29]. In this context, single point crossover may distort the original solutions so to avoid it the above mentioned method is applied. To explain it, suppose there are two chromosomes $\{1, 1, 1, 2, 2, 3, 3, 3\}$ and $\{2, 2, 2, 1, 1, 3, 3, 3\}$. These two chromosomes represent the same clustering solution. In this example, if we perform single point crossover at point 4, then these solutions become $\{2, 2, 2, 1, 2, 3, 3, 3\}$ and $\{1, 1, 1, 2, 1, 3, 3, 3\}$. So after the crossover operation the original solutions become distorted. Therefore, the crossover operation is applied as described in [29].
- Mutation: In this method, each chromosome undergoes a mutation process with a small probability μp. A small fraction value is added or subtracted with each bit of the chromosome. As the label of each object (image) is represented as integer, therefore, the float value generated after mutation is transformed into the nearest integer.

6. Experimental Design and Results

In this section, we first describe the datasets that we used in our experiments. To find the efficacy of the proposed method we compare it with state-of-the-art methods like WECA-SL, WECA-AL, WECA-CL [26] along with the traditional methods like CSPA, HGPA and MCLA [17]. It is observed that different variants of WECA method [26] outperforms a majority of the existing methods and hence these are chosen as the baselines. The adopted performance metrics are ARI [2, 27].

In the experimental design, an online platform is created and we posted some ambiguous images (that is not possible for a computer to realize the sentiment). Then we solicited crowd opinions to cluster those images depending on similar characteristics. Initially, we treat it as fixed-sized clustering problem so we posted the most probable number of clusters in that platform. As various crowd workers might group the images from different view points therefore diverse clustering solutions can be obtained from them. In this way, the crowdsourced dataset is created. Fig. 1 shows the snapshot of a question for image clustering task posted to the crowd workers. The question contains 6 images (a, b, c, d, e, f) that are outsourced to be clustered into 4 groups based on some similarities in features. It is possible that a crowd worker may partition

the objects as [(a, d), (b), (c), (e, f)] as images a and d belong to entertainment industry, image b is from sports, images c and (e, f) are from political party although their political parties are different. In this case, one possible answer given by a particular crowd worker would be {1, 2, 3, 1, 4, 4}. While another crowd worker may perceive it differently and may partition the objects as [(a), (b, d), (c), (e, f)] as image d is a more popular as sportsman although recently he joined as an entertainer. So there can multiple ways in which a crowd worker can make the clusters based on his/her intuition. It can be seen that the images have been designed in such a way so that grouping formation becomes ambiguous and remains dependent on the perception of the crowd workers. As mentioned earlier, after obtaining the crowdbased clustering solutions we remove some opinions of each crowd worker. This produces a set of partial clustering solutions similar like that all the objects are not clustered by all the crowd workers. Then we predict the missing values of each clustering solution (crowd worker) based on the similar set of workers of a particular crowd worker. After refilling the missing value we then apply the proposed multi-objective optimization algorithm to produce consensus clustering. Finally, the performance of the proposed approach and power of crowd is studied. Note that, before filling up the missing values the label transformation is done to make the correspondence between the labeling. Experiments are performed in MATLAB 2013a and the running environment is an Intel (R) Core(TM)i3 CPU 2.53 GHz machine with 4 GB of RAM running Windows 7 Home Premium.

As described earlier, to collect the opinions from the crowd workers few questions (each containing few images) were posted online and opinions were solicited from them. For example, for the question posted as shown in Fig. 1, some crowd workers may cluster the images according to their profession and industry as [(a), (b), (c, e), (d, f)]. Here image a is of an actor, b is from sports, c and e are politicians, and lastly images d and f are personalities who have some dual characteristics. The dual characteristics of d and f means that both of the personalities have established themselves either as cricketer cum politician or actor turned politician. Again these same images can be grouped into four clusters in another different way such that images (a) and (d) remain in one cluster, image (b) is in another cluster, images (e), (f) are in some other cluster and lastly image (c) lies in another separate cluster. The reason is that image (a), (d) are from entertainment industry, image (b) is from sports, image (c) is a personality from a different political party whereas images (e), (f) belong to the same cluster as they both are from same political party. So these types of tricky images can generate a large amount of dilemma to the crowd workers and hence multiple diverse clustering solutions can be obtained from them. Therefore we cannot say that one way of thinking is proper and the other way is wrong. Different people perceive a set of pictures in different ways and hence resulting in different clusterings. Therefore the objective is to reach maximum agreement from the set of clustering solutions.

6.1. Descriptions of Datasets

Five crowdsourced datasets have been generated for this purpose by means of crowdsourcing. To generate the datasets an online platform was designed to collect the clustering solutions from the crowd workers over some tricky images. By means of this 26 clustering solutions were obtained for ‘Question 1’, 24 clustering solutions for ‘Question 2’, 26 clustering solutions

for ‘Question 3’, 26 clustering solutions for ‘Question 4’ and 26 solutions for ‘Question 5’ were collected. Each of the artificial datasets comprises of one such question producing in total five datasets for five questions. A short description of the crowdsourced datasets is provided in Table 1.

Table 1
Description of crowdsourced datasets.

Datasets	Number of Classes
Question 1	4
Question 2	4
Question 3	3
Question 4	4
Question 5	4

6.2. Study on the Datasets

In Tables 2 - 6, the performance metric values obtained by different clustering ensemble algorithms for the 5 crowdsourced datasets are reported. Due to the non-availability of a single ground truth solution in these crowdsourced datasets, to measure the accuracy, the ensemble solutions are compared with all the base clustering solutions and average values are shown. Here the clustering solutions provided by each of the crowd workers (for a specific question) after prediction of missing values is compared with the consensus solution produced by the proposed method along with others and the average ARI is reported. It is evident from the tables that in all the cases, the proposed algorithm provides equally good performance in terms of ARI. More interestingly, the proposed algorithm provides the number of clusters automatically which is the drawback of the other state-of-the-art approaches. As the number of clusters are different in different base clustering solutions, therefore, there is a need to change the number of clusters each time while executing the algorithms. This makes the situation more complex when there are a large number of clusters in the base crowdsourced clustering solutions. The experimental result demonstrates that the final consensus predicts the accurate number of clusters as predicted by other ensemble methods. It can be seen in some cases the ARI value is zero as the consensus clustering generates only a single cluster. The non-dominated Pareto optimal solutions obtained after applying the proposed algorithm on the clustering solutions of Question 4 is demonstrated in Fig. 2. Thus the effectiveness of the proposed method for making a final decision regarding selecting the number of clusters can be observed easily.

Interestingly, it is difficult to measure correctness of any such cluster solution, on the basis of the solution of the question setter who of course is not an expert in the job of clustering. Rather the question setter had only one such point of view. Therefore instead of using any particular ground truth solution, all the base clustering solutions (after missing value predictions) are compared with the consensus solutions derived by the various methods. Then based on the average ARI similarity the closeness of consensus solution with respect to the crowd workers solutions is estimated. The area of interest of a crowd-worker, the society to which he/she belongs to, his/her denomination plays a vital role in determining the accuracy of such image

Table 2
Performance values for Question 1 (26 samples) in terms of Adjusted Rand Index value.

Algorithm	K=2	K=3	K=4	K=5
CSPA	0.3178	0.5945	0.5945	0.5945
MCLA	0.2379	0.5945	0.7220	0.7239
HGPA	0.3178	0.5945	0.7220	0
WECA-SL	0.1197	-0.2767	0.1234	0
WECA-AL	0.2229	-0.0865	-0.1380	0
WECA-CL	0.2229	-0.0865	-0.1380	0
GP-MGLA	0.0586	0.4416	0.4416	0.7366
Proposed	–	–	0.7200	–

Table 3
Performance values for Question 2 (24 samples) in terms of Adjusted Rand Index value.

Algorithm	K=2	K=3	K=4	K=5
CSPA	0.2484	0.2319	0.4481	0.2799
MCLA	0.2484	0.0909	0.4481	0.2817
HGPA	0.2484	0.3065	0.4481	0
WECA-SL	0.1220	0.3075	0.4481	0.3373
WECA-AL	0.2484	0.3527	0.4481	0.3373
WECA-CL	0.1707	0.3075	0.4481	0.3373
GP-MGLA	0.0400	-0.0704	0	0
Proposed	–	–	0.4481	–

Table 4
Performance values for Question 3 (26 samples) in terms of Adjusted Rand Index value.

Algorithm	K=2	K=3	K=4	K=5
CSPA	0.4105	0.7222	0.7222	0.7222
MCLA	0.4572	0.7222	0.7222	0.7222
HGPA	0.3220	0.7222	0.6018	0.6018
WECA-SL	0.4572	0.7222	0.6018	0.3541
WECA-AL	0.4572	0.7222	0.6018	0.3541
WECA-CL	0.4572	0.7222	0.6018	0.3541
GP-MGLA	0.4572	0.7222	0.6018	0.3541
Proposed	–	0.7222	–	–

clustering jobs. However, obtaining multiple opinions from them for ambiguous image clustering tasks in an effective way and making consensus from these multiple opinions can further lead to generate a robust clustering solution from a set of diverse clustering solutions.

7. Conclusion

In this paper, we introduce a crowdsourcing model to solve ambiguous image clustering tasks. This model can provide us with the final consensus clustering solution after taking input from multiple crowd workers. The motivation behind using the crowd powered model is due to the inefficiency of machines to solve large image clustering tasks in a time efficient way. Therefore,

Table 5
Performance values for Question 4 (26 samples) in terms of Adjusted Rand Index value.

Algorithm	K=2	K=3	K=4	K=5
CSPA	0.3065	0.4037	0.5639	0.5639
MCLA	0.2111	0.4037	0.5639	0.5639
HGPA	0.3065	0.4037	0.5639	0
WECA-SL	0.3251	0.4034	0.5639	0.3465
WECA-AL	0.3251	0.4037	0.5639	0.3465
WECA-CL	0.3251	0.4037	0.5639	0.3465
GP-MGLA	0	0	0	0
Proposed	–	–	0.5639	–

Table 6
Performance values for Question 5 (26 samples) in terms of Adjusted Rand Index value.

Algorithm	K=2	K=3	K=4	K=5
CSPA	0.1807	0.3198	0.3198	0.4685
MCLA	0.2330	0.4069	0.5525	0.5525
HGPA	0.1716	0.3491	0	0.5525
WECA-SL	0.1716	0.3778	0.5525	0.5071
WECA-AL	0.1460	0.3060	0.5525	0.5071
WECA-CL	0.1276	0.3778	0.5525	0.5071
GP-MGLA	0	0	0.3189	0.5071
Proposed	–	–	0.5525	–

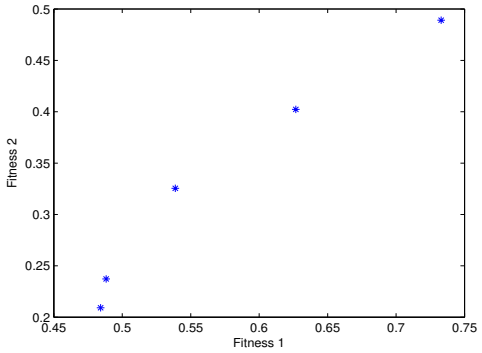

Figure 2: Sample Pareto optimal front for non-dominated solutions produced by the proposed method for a particular question set. Here fitness 1 is the average value of ARI and Jaccard similarity and fitness 2 is the standard deviation of the similarity values obtained by ARI.

crowdsourcing based clustering solutions are employed as an effective mechanism to generate a good consensus from multiple solutions. The consensus solutions obtained from the proposed technique are compared with that of other state-of-the-art approaches over 5 datasets. It is seen that in most of the datasets the ensemble solution provided by the proposed approach maintains a consistent good performance. More importantly, like the state-of-the-art methods the method does not require the number of desired clusters each time while applying the algorithm. It is also shown that the proposed framework can tackle the partial crowdsourced

clustering solutions by predicting the missing values first. Finally, it produces the accurate consensus clustering along with the most probable number of clusters automatically. In future, the proposed algorithm can be modified to work with input clustering solutions having different numbers of clusters instead of fixed number of clusters. Furthermore, as few spammers can try to manipulate the overall process, some additional filtering criteria can be imposed on the crowd workers at the time of collecting the opinions from them. Finally, selection of a perfect crowd worker depending upon the hardness of the question, might be another direction of research in the field of crowdsourced clustering. In addition, the Bayesian Posterior probability can be incorporated to debias the annotations and it can be effective to increase the quality of the consensus. Another direction can be the order of the questions/images can be kept random so that different crowd workers cannot see the same ordering, hence, further research can be performed to study the effectiveness of the random ordering of the questions to increase the quality.

References

[1] S. V. Pons, J. R. Shulcloper, A survey of clustering ensemble algorithm, International Journal of Pattern Recognition and Artifcial Intelligence 25 (2011) 337–372.

[2] K. Yeung, W. Ruzzo, An empirical study on principal component analysis for clustering gene expression data., Bioinformatics 17 (2001) 763 – 774.

[3] A. K. Jain, R. C. Dubes, Data clustering: A review, ACM Computing Surveys 31 (1999).

[4] S. Bandyopadhyay, U. Maulik, A. Mukhopadhyay., Multiobjective genetic clustering for pixel classification in remote sensing imagery, IEEE Transactions on Geoscicence and Remote Sensing (2007).

[5] A. Mukhopadhyay, U. Maulik, S. Bandyopadhyay., A survey of multiobjective evolutionary clustering, ACM Computing Surveys 47 (2015) 61:1–61:46.

[6] D. L. Davies, D. Bouldin, A cluster separation measure, IEEE Transactions on Pattern Analysis and Machine Intelligence 1 (1979) 224–227.

[7] W. M. Rand, Objective criteria for the evaluation of clustering methods, Journal of the American Statistical Association 66 (1971) 846–850.

[8] J. Dunn, Well separated clusters and optimal fuzzy partitions., J Cyberns 4 (1974).

[9] L. Hubert, P. Arabie, Finding Natural Clusters Using Multi-Clusterer Combiner Based on Shared Nearest Neighbors, Journal of Classification 2 (1985) 193–218.

[10] G. Demartini, D. E. Difallah, C. Mauroax, Zencrowd: leveraging probabilistic reasoning and crowdsourcing techniques for large scale entity linking., in: Proceedings of the 21st International Conference on World Wide Web, Lyon, France, 2012, pp. 469–478.

[11] D. C. Brabham, Detecting stable clusters using principal component analysis, Methods Mol. Biol. 224 (2013).

[12] D. Hovy, T. B. Kirkpatrick, A. Vaswani, E. Hovy, Learning whom to trust with mace, in: Proceedings of the North American Chapter of the Association for Computational Linguistics – Human Language Technologies (NAACL HLT), Atlanta, Georgia, 2013, pp. 1120–1130.

[13] M. Lease, On quality control and machine learning in crowdsourcing., in: Proceedings of 3rd Human Computation Workshop (HCOMP) at AAAI, France, 2011, pp. 97–102.
[14] S. Chatterjee, E. Kundu, A. Mukhopadhyay, A markov chain based ensemble method for crowdsourced clustering, in: WiP track of Fourth AAAI HCOMP, Austin, USA, 2016, (arXiv ID: 1609.01484).
[15] J. Whitehill, P. Ruvolo, T. Wu, J. Bergsma, J. Movellan, Whose vote should be count more: Optimal integration of labels from labelers of unknown expertise., in: Proceedings of Advances in Neural Information Processing Systems, Vancouver, Canada, 2009, pp. 2035–2043.
[16] R. Snow, B. O'Connor, B. Jurafsky, A. Ng, Cheap and fast-but is it good? evaluating non-expert annotation for natural language tasks, in: Proceedings of the Empherical Methods in Natural Language Processing (EMNLP), Hawali, USA, 2008, pp. 254–263.
[17] A. Strehl, J. Ghosh, Cluster ensembles - a knowledge reuse framework for combining partitionings, 11th National Conference of Artificial intelligence (2002).
[18] G. Karypis, V. Kumar, A fast and high quality multilevel scheme for partitioning irregular graphs, SIAM Journal on Scientific Computing 20 (1999) 359 – 392.
[19] G. Karypis, R. Aggarwal, V. Kumar, S. Shekhar, Multilevel hypergraph partitioning: Applications in vlsi design, ACM/IEEE Design Automation Conference (1997) 526–529.
[20] A. Ben-Hur, A. Elisseeff, I. Guyon, A stability based method for discovering structure in clustered data, Pac Symp Biocomputing (2002) 6–17.
[21] H. G. Ayad, M. S. Kamel, Cumulative voting consensus method for partitions with a variable number of clusters, IEEE Transactions on pattern analysis and machine intelligence 30 (2008).
[22] S. Dudoit, J. Fridlyand, Bagging to improve the accuracy of a clustering procedure, Bioinformatics 19 (2003) 1090–1099.
[23] B. Fischer, J. Buhmann, Bagging for path-based clustering, IEEE Trans. Patt. Anal. Mach. Intell. 25 (2003) 1411–1415.
[24] K. Tumer, A. Agogino, Ensemble clustering with voting active clusters, Patt. Recogn. Lett. 29 (2008) 1947–1953.
[25] H. Wang, H. Shan, A. Banerjee, Bayesian cluster ensembles, Wiley Online Library (2010).
[26] D. Huang, J. H. Lai, C. D. Wang, Combining multiple clusterings via crowd agreement estimation and multi-granularity link analysis, Neurocomput. 170 (2015).
[27] L. Hubert, P. Arbie, Comparing partitions, Journal of Classification 2 (1985) 193–218.
[28] K. Deb, A. Pratap, S. Agrawal, T. Meyarivan, A fast and elitist multiobjective genetic algorithm: Nsga-ii, IEEE Transactions on Evolutionary Computation 6 (2002) 182–197.
[29] S. Chatterjee, A. Mukhopadhyay, Clustering ensemble: A multiobjective genetic algorithm based approach, International Conference on Computational Intelligence: Modeling, Techniques and Applications (CIMTA) (2013).