Clustering Based Approach for Ground Truth Inference in Crowdsourced Data

*[1]Victor T. Odumuyiwa, [1]Anurika Umeanozie, [1]Oladipupo Sennaike, [2]Olubukola Adekola,
[1]Babatunde Sawyerr and [1]Ebun Fasina
[1]Department of Computer Sciences, University of Lagos, Akoka, Nigeria
[2]Department of Software Engineering, Babcock University, Ilisan Remo, Nigeria
{vodumuyiwa | osennaike | bsawyerr | efasina}@unilag.edu.ng | anurikaumeh14@gmail.com | adekolao@babcock.edu.ng

Received: 16-FEB-2020; Reviewed: 13-MAR-2020; Accepted: 26-APR-2020
https://doi.org/10.46792/fuoyejet.v7i2.800

ORIGINAL RESEARCH ARTICLE

Abstract- Crowdsourcing provides a means of gathering data from the public in order to infer what the ground truth label of an unfamiliar entity is. Such data are not used for decision making in their raw form until further processing is done to infer ground truth from the crowdsourced data. This paper presents a detailed comparative analysis of the ground truth inference ability of three clustering algorithms on crowd sourced datasets with different experimental scenarios (Initializing centroids and extracting class labels). The algorithms include, the self-organizing maps, the k-means and the expectation maximization clustering algorithm. The approach used entails generating a new dataset containing the probability distributions of the class predictions for each example in the noisy dataset, then clustering the data points using the generated probability features in order to infer their class labels. The three algorithms were implemented and compared with the Majority voting algorithm on the different datasets used in this research. The datasets used are Adult2, weather sentiments, emotion, valence5 and employee review dataset. Four possible experimental scenarios for inferring the ground truth label from the curated dataset were analysed. The first scenario makes use of the clustering algorithm alone relying on the inner workings of the algorithm to predict the ground truth, while the second scenario makes use of an extract class label mechanism where the ground truth label was inferred by performing a further analysis on the clusters provided by the algorithm. In the third scenario, the centroids of the clustering algorithm were pre-initialized by setting the maximum value in each class from the curated data as a centroid, where centroid might mean something different relative to the particular algorithm. The fourth experimental scenario is a combination of the second and third scenario. Experimental results show that the self-organizing map (SOM) performs best across all the datasets when the weights of the units in the SOM are pre-initialized. SOM had the best performance on the weather sentiments dataset recording 92.49% accuracy and ROC AUC score of 0.88. It also recorded the best overall average accuracy of 50.2% and ROC AUC score of 0.59365 across all the datasets.

Keywords—Clustering, Crowdsourcing, Expectation Maximization, K-means, ground truth inference, Self-Organizing Maps.

1 INTRODUCTION

Information gathering is an important component in data analysis. Data are facts and statistics collected for reference or analysis. There are different methods through which data can be generated. One of the methods is crowdsourcing. Crowdsourcing is a way of outsourcing different kinds of problems to the "crowd" and at the end, getting improved solutions to the problems (Chittilappilly, Chen, & Amer-Yahia, 2016; Demartini, Difallah, & Cudré-Mauroux, 2012; Xu, Jiang & Li, 2021; Song, Liu & Zhang, 2021).

The term "crowdsourcing" emanated from Jeff Howe and Mark Robinson (Howe, 2006). Though different researchers have explained crowdsourcing in several ways however, the main and acceptable definition of crowdsourcing is the one given by Jeff Howe who explained it as "the act of a company or institution taking a function once performed by employees and outsourcing it to an undefined (and generally large) network of people in the form of an open call" (Howe, 2006).

In recent times, various applications have been developed to help organizations and individuals to acquire data from the crowd. Examples of these applications are: Amazon Mechanical Turk, Clickworker and CrowdFlower. These applications have made crowdsourcing easier by bringing together the requesters and the workers. A crowdsourcing activity can only be declared successful if the requester is able to get the value required from the responses provided by the crowd and have enough participation while also considering various constraints like time, budget, and quality (Simperl, 2015).

Due to the presence of noisy labels in crowdsourced data, generally, such data are not used for decision making in their raw form until further processing is done to infer ground truth from them (Bi, Wang, Kwok, & Tu, 2014; Adeogun & Odumuyiwa, 2019; Xu, Jiang & Li, 2021; Cui et. al., 2021). This paper emphasizes the ground truth inference on crowdsourced data. Clustering based approaches for ground truth inference in crowdsourced data was used with different experimental scenarios (Initializing centroids and extracting class labels). The approach used entails generating a new dataset containing the probability distributions of the class predictions for each example in the noisy dataset, then clustering the data points using the generated probability features in order to infer their class labels.

Section 2 of this paper presents the literature review highlighting the approaches in classifying crowdsourced data as well as some related works. Section 3 discusses the methodology and techniques employed in this research,

*Corresponding Author

Section B- ELECTRICAL/ COMPUTER ENGINEERING & RELATED SCIENCES
Can be cited as:
Odumuyiwa V., Umeanozie A., Sennaike O., Adekola O., Sawyerr B. and Fasina E. (2022): Clustering Based Approach for Ground Truth Inference in Crowdsourced Data, *FUOYE Journal of Engineering and Technology* (FUOYEJET), 7(2), 141-147. http://doi.org/10.46792/fuoyejet.v7i2.800

the designed algorithms and the data sets used for experimentation. Results are discussed in section 4 and this is followed by the conclusion.

2 Literature Review

Over the years, various algorithms have been used for classifying crowdsourced data. According to Hung et. al. (2013), approaches in classifying crowdsourced data can generally be categorized as either iterative or non-iterative.

2.1 Non-Iterative Approach

This approach employs heuristics to evaluate a single aggregated value of each question separately. One of the common aggregating techniques in this approach is Majority Voting (MV), in which the label with major votes is selected as the true label. Other techniques in this approach are Honeypot (HP) and Expert Label Injected Crowd Estimation (ELICE).

Majority Voting is a straightforward method where a specific label is selected without pre-processing the data. In MV, the most frequent answer (label) is selected as the true label. The method however, does not take into consideration the expertise or knowledge level of every participant hence the possibility of having more noisy labels in the crowdsourced data. In addition, there is a high chance that this method may not provide a good result in a system infiltrated by spammers (Hung et.al., 2013; Hernández-González, Inza, & Lozano, 2018).

Honeypot method works the same way as MV but with a quality control technique. Some workers are filtered during the pre-processing step to minimize the number of less skilled workers. Quality control is achieved by including different trapping questions whose answers are already known into the main or original questions during the filtering process. A threshold value is set for which workers that get less than the threshold in answering the trapping questions are identified as spammers or less skilled and removed immediately (Hung et. al., 2013; Hernández-González, Inza, & Lozano, 2018). The possible labels are evaluated using MV method discussed above amongst the remaining workers. The major disadvantage faced in this approach is that, truthful or genuine workers might be categorized as spammers in cases where the trapping questions used are a bit complex.

Expert Label Injected Crowd Estimation (ELICE) is an extension of Honeypot. The knowledge level of each worker or participant (that is, whether the worker is skilled or unskilled) is determined by using the trapping questions embedded into the original questions. The expertise level of a worker is determined by computing the percentage of the correct answers provided by the worker out of the total number of questions. It also measures the difficulty of each question by calculating the percentage of workers that got the question correct. This approach measures both the level of expertise of the worker and the difficulty of the questions. ELICE also faces the same disadvantage discussed in the Honeypot model approach, where a genuine worker or participant might be classified as a spammer due to the complexity of the trapping questions used.

2.2 Iterative Approach

In this approach, series of iterations are performed on the data before classifying the data according to its labels. Algorithms that can be found under this approach include K-Means, Expectation Maximization, etc.

2.2.1. K-Means

The K-Means algorithm by design is a clustering algorithm, however it can be used for classification tasks. The K-means algorithm takes a full dataset consisting of binary or multiclass examples and then clusters the data into separate groups. K-means algorithm aims to partition a set of observations $\{x_1, x_2, x_3, \ldots, x_n\}$ (where each observation is a d-dimensional real vector) into k clusters $\{c_1, c_2, c_3, \ldots, c_k\}$.

2.2.2. Expectation Maximization (EM)

In statistical modelling, the expectation maximization (EM) algorithm is an iterative method for performing maximum-apriori (MAP) estimation in the presence of latent variables (Moon, 1996). There are two steps involved in EM algorithm; an expectation step (E) where the log-likelihood is evaluated using the most recent estimate for the parameters and a maximization step (M) where the computation of parameters that maximize the expected log-likelihood from the E-step is carried out. The estimated parameters are employed in in the next E-step to determine the distribution of the latent (hidden) variables.

2.2.3. Related Works

Besides the majority voting algorithm which is the most common approach used to remove noisy labels in crowdsourced data, Dawid and skene (Dawid & Skene, 1979) proposed a well-known approach for ground truth inference called the Dawid-Skene (DS) algorithm. DS is based on the Expectation-Maximization (EM) principle and it is used for estimating maximum likelihood of worker error rates. It computes a confusion matrix of each labeller and the class prior and uses an EM approach to infer the estimated labels for the examples. Many other algorithms from several researchers (Hung et. al, 2013; Cai, Nie, & Huang, 2013; Ipeirotis, Provost, & Wang, 2010; Li, Jiang, & Xu, 2019; Sinha, Sukrut, & Balasubramanian, 2018; Welinder et. al., 2010; Whitehill et. al., 2009) are derived from the DS methodology.

Zhang, Sheng, & Wu (2019) and Zhang et. al. (2015) made use of clustering approach to infer an estimated label for each example. A multi-class ground truth inference (GTIC) using K-means clustering algorithm for feature categorization was proposed by Zhang et. al. (2015). The conceptual features of each example were first generated, then K-means clustering algorithm was applied on the newly generated examples and each cluster was mapped to a specific class. The authors also furthered their research by proposing another approach called bilayer collaborative clustering (BLCC) (Zhang, Sheng, & Wu, 2019) which is an extension of the GTIC model. Using the BLCC approach, the conceptual-level features of the examples are first generated from the multiple noisy

http://doi.org/10.46792/fuoyejet.v7i2.800 http://journal.engineering.fuoye.edu.ng/

labels provided by the crowd and clustering performed on the conceptual level features. Then, the true labels estimations are formed by performing another clustering on the physical-level features.

Recent works reported by Song, Liu & Zhang (2021), in addition to inferring ground truth from crowdsourced data, tried to detect the collusive behaviours of workers in labelling tasks. Changyue, Kaibo & Xi (2021) also focussed on detecting collusive behaviour of workers. Resampling-based noise correction method proposed by Xu, Jiang & Li (2020) uses a filter to separate a clean set from a noisy set, and the two sets are continuously resampled severally based on a given ratio and a classifier is built on the data set at each resampling iteration. All the classifiers built during the resampling phase are then used to re-label each instance in the clean and noisy data set, and MV was applied on the output of all the classifiers on a given instance to determine the true label of that instance. In another work, Xu, Jiang & Li (2021) proposed a Cross-Entropy-based Noise Correction (CENC) method for noisy label removal. In order to filter noisy instances, they generated entropies of the label distribution from multiple noisy labels and they computed "cross-entropies between each possible true class probability distribution and each predicted class probability distribution" which in turn were used to correct the noisy instances. Another approach named co-training-based noise correction (CTNC) was used by Yu, Liangxiao and Chaoqun (2022) to correct label noise in crowdsourced data.

Most of the existing studies focused on binary ground truth inference problems. Zhang, et. al. (2015) and Zhang, Sheng, & Wu (2019) focused on multi-class inference, which has not been well studied. The literature reports several efforts made to improve the label quality gotten from crowdsourced data using different algorithms (Zhang, Sheng, & Wu, 2019) but recorded performances on different real-world datasets is still low regardless of the algorithms used for the ground truth inference. Our study provides experimental results of the superior ability of the self-organizing map over the expectation maximization and k-means algorithm used for ground truth inference problems on multi-class data.

3. Clustering Based Approach for Ground Truth Inference

3.1 Problem Definition

A crowdsourced dataset D is defined as one containing a set of examples $d_i \in \{d_1, d_2, \dots, d_n\}$; where D_N is the number of examples in D. Each example in the crowdsourced dataset is assigned a label or class $c_k \in \{c_1, c_2, \dots, c_k\}$; by L_N "labellers" where L_N is the number of labellers usually of unknown identities. The dataset G denotes a set of unique examples $g_i \in \{g_1, g_2, \dots, g_n\}$; where G_N is the total number of examples in the ground truth dataset ($G_N < D_N$ and $D_N <= G_N * L_N$). The dataset G is a ground truth dataset with the goal of determining the accuracy of the true labels provided for its features $g_i \in \{g_1, g_2, \dots, g_n\}$. It is safe to assume that most of the classes provided by the "labellers" will be incorrect, however studies have shown that at least 50% of labellers get the correct class (Li, Jiang, & Xu, 2019). The problem therefore is to infer the ground truth class c_i for each unique example by analyzing the classes assigned to it. The goal is to maximize the accuracy $\sum_{i=1}^{G_N}(c_i = \hat{c}_i)$ where $\hat{c}_i$ is the true label of example.

3.2 Generating Features from Crowdsourced Data

Some studies (Cai, Nie, & Huang, 2013; Xu, Jiang, & Li, 2020) have shown that clustering algorithms such as the k-means algorithm tend to perform poorly when tasked with clustering on single feature datasets. Many crowdsourced datasets contain single feature values representing an object such as a URL or the name of a place. The poor performance of clustering algorithms can be attributed to this property of the crowdsourced data. With this knowledge in mind, this paper follows in the footsteps of Zhang et. al. (2015) and extract a new set of features and thus create a new dataset that will be used for the clustering task. Continuing with the notations established in section 3.1, for each instance in the ground truth dataset g_i we denote the probability of g_i being a member of class c_k by θ_k. Therefore, the probabilities of g_i belonging to each class is:

$$Prob(g_i) = \{\theta_1, \theta_2, \dots . \theta_k\} \text{ where } \sum_1^k \theta_k = 1 \qquad (1)$$

In equation 1, θ_k is simply calculated as $\theta_k = M/L_N$, where the numerator M is the number of times class c_k was provided as label for instance g_i, the denominator is the sum of labellers that provided labels for instance g_i. Thus, the new dataset consists of the newly generated features $\{\theta_1, \theta_2, \dots . \theta_k\}$ for all samples g_i in G. This new dataset with the newly generated features is denoted as F.

3.3 Clustering Algorithms Employed

The simple k-means and expectation maximization as discussed in the iterative approach above (Section 2.2) were used in this work. In addition, self-organizing map (SOM) which can also be considered as iterative was included among the clustering algorithms employed.

The self-organizing maps (SOM) is a type of unsupervised artificial neural network (ANN) that makes use of competitive learning to learn 2-dimensional discretized representations of training data (Kohonen, 1990). The SOM is a 2D map that is defined beforehand and consists of components called nodes. These nodes are associated with a weight vector that have the same length as features in the training example. When a training example is presented to the SOM, the Euclidean distance to all weight vectors is computed. The node with the least distance to the training example is declared the best-matching unit (BMU). In essence, the SOM behaves like a clustering algorithm because similar training examples tend to cluster around the same BMU. Classification using the SOM is achieved by association, that is one can determine the class of a novel example by analysing the classes of all examples clustered around the BMU. This work explores the use of SOM for ground truth inference.

3.4 Ground Truth Inference Algorithm

Clustering based approach for ground truth inference provides a framework of using different clustering

http://doi.org/10.46792/fuoyejet.v7i2.800 http://journal.engineering.fuoye.edu.ng/

algorithms to infer the true label of an example. The clustering algorithms used in this experiment include the simple k-means, expectation maximization and self-organizing map (SOM) algorithm. Algorithm 1 shows the overall algorithm that was used in this experiment.

The algorithm starts by initializing an empty array to store the predictions from the clustering algorithm. Next, new features $\{\theta_1, \theta_2, \ldots . \theta_k\}$ were generated for all instances in D using the probability notation described in section 3.2 and they were stored in a new dataset F. The clustering algorithm is initialized next with its parameters. The two most important parts of the algorithm centre on the choice of providing the clustering algorithm with initializing the centroids' function or extracting the class label's function.

Clustering algorithms such as the k-means and expectation-maximization have the option of initializing the centroids of the clusters using the method of Zhang et. al. (2015). This has been shown to improve performance as demonstrated in their work (Zhang et. al., 2015). To initialize the centroids for each class, the features with the maximum value in each class position are selected. So, for example, the initial centroid for cluster 1 will be the set of features $\{\theta_1, \theta_2, \ldots . \theta_k\}$ where θ_1 is the largest value at index 1 for all features $\{\theta_1, \theta_2, \ldots . \theta_k\}$ in the new dataset.

The second experimental scenario is the option of using the Extract class labels technique. After training the clustering algorithm on the dataset F, the clustering algorithm will have the ability to predict ground truth labels for all the examples in D. In an experimental setup, these predictions can then be compared with the true labels to quantify performance. When the clustering algorithm assigns an example to a cluster, further analysis is carried out on all the examples in that cluster. As explained earlier, an example is presented as a set of features$\{\theta_1, \theta_2, \ldots . \theta_k\}$. Thus, the final prediction is the index at which the maximum value can be found for all features $\{\theta_1, \theta_2, \ldots . \theta_k\}$ in all examples in that cluster.

Algorithm 1: Ground Truth Inference Algorithm

Input: Crowdsourced dataset D, bool init_Centroid, bool extract_class_labels
Output: A sample set G, where each examples[g_i] has an estimated label

```
Initialize predictions = []
Generate new features[θ1, θ2, θ3, ..., θk] for all di in D using eq(1) in section 3.2
F = new array [ DN], where DN is the length of D
for i in D do
    F[i] = [θ1, θ2, θ3, ... θk]
end for
Run the clustering algorithms by using its default parameters on the new dataset F.
if (init_centroids == True) then
    Select the initial centroids using algorithm 2
    Run the clustering algorithms
    if (extract_class_labels == True) then
        Extract class labels using algorithm 3
    end if
end if
if (init_centroids == False) then
    Run the clustering algorithms
    if (extract_class_labels == True) then
        Extract class labels using algorithm 3
    end if
end if
return predictions
```

Algorithm 2 shows the initializing centroid algorithm. To initialize the centroids for each class, the features with the maximum value in each class position were selected. So, for example, the initial centroid for cluster 1 will be the set of features $\{\theta_1, \theta_2, \ldots . \theta_k\}$ where θ_1 is the largest value at index 1 for all features $\{\theta_1, \theta_2, \ldots . \theta_k\}$ in the new dataset. In this paper, initializing centroids in SOM implies that the weights of the nodes in the 2×2 dimensional map are pre-initialized using algorithm 2.

Algorithm 2: Initialize Centroids

Input: Sample set D where each sample d_i is a list of features $[\theta_1, \theta_2, \theta_3 \ldots \ldots \theta_n]$
Output: $n \times n$ array of centroids, where n is number of unique labels

```
centroids ← array[n × n]
for 1 to n do
    foreach di in D do
        centroids[i ] ← select di with largest value at index i
    end foreach
end for
return centroids
```

Algorithm 3 shows the option of using the Extract class labels technique. After training the clustering algorithm on the dataset D, the clustering algorithm would have the ability to predict ground truth labels for all the examples in D. When the clustering algorithm assigns an example to a cluster by predicting its label, an analysis of all the examples in that cluster is done. As stated in section 3.2, an example is presented as a set of features$\{\theta_1, \theta_2, \ldots . \theta_k\}$. Thus, the index at which the maximum value can be found is chosen as the class to be assigned to that cluster as the final prediction.

Algorithm 3: Extract Class Labels

Input: Sample set D where each sample d_i is a list of features $[\theta_1, \theta_2, \theta_3 \ldots \ldots \theta_n]$
Output: Sample set E where each sample e_i is a new estimated label for sample d_i

```
ei is a new estimated label for sample di
E ← emptyset[ ]
clusters ← clustering algorithms(D)
foreach cluster in clusters do
    ei ← get label with highest frequency in cluster
    E ← ei
end foreach
return E
```

3.5 DATASETS

The following datasets are used for the experiments; Table 1 provides a breakdown of the dataset.

http://doi.org/10.46792/fuoyejet.v7i2.800 http://journal.engineering.fuoye.edu.ng/

Table 1. Breakdown of the datasets used in the experiment

Datasets		Number of Classes	Number of Examples	Total number of Labels collected	Average number of Labelers per question
Adult2		4	309	3260	10
Weather Sentiments		5	25	500	20
Emotions	Anger	100	100	1000	10
	Sadness	100	100	1000	10
	Surprise	100	100	1000	10
	Joy	100	100	1000	10
	Disgust	100	100	1000	10
	Fear	100	100	1000	10
Valence		5	100	1000	10
Employee Review		5	500	5000	10

3.5.1 Adult2

Adult2 is made up of a crowd sourced dataset containing 3260 examples with noisy labels and ground truth dataset with 309 unique examples and their true labels. This dataset was gotten from various workers on Amazon MTurk by Ipeirotis, Provost, & Wang (2010). They requested that labellers should review the ratings of various websites under the category of G (General), PG (parental Guidance), R (Restricted for anyone under 17) and X (Adults only with explicit scenes).

3.5.2 Weather Sentiments

This dataset is made up of crowd sourced dataset containing 500 imbalanced examples with noisy labels and a ground truth dataset with 25 unique examples and their true labels. This dataset was generated by University of Southampton using Amazon Mechanical Turk (Venanzi et. al., 2015). The labels belong to five classes. The following categories are used to classify the sentiment judgments: negative (0), neutral (1), positive (2), tweet not related to weather (3) and can't tell (4). The goal for the labellers is to provide rating according to five classes for sample tweets about the weather.

3.5.3 Emotions Dataset

The emotions dataset tasks the labellers with providing a score between the range of 0 to 100 for news headline (Snow et. al., 2008). Where 0 represents no emotion whatsoever and 100 represents maximal emotion. The full emotions dataset consists of 6 unique datasets for different emotions including "happy", "sadness", "disgust", "joy", "fear", "surprise".

3.5.4 Valence5 Dataset

Snow et. al. (2008) selected a 100-headline sample from the SemEval-2007 Task 14 and collected labels for the dataset valence, where each example was labelled by 10 unique labellers. Using the valence data which is numeric, the valence value was divided into 5 classes (Strong, Negative, Neutral, Positive, and Strong Positive).

3.5.5 EmployeeReview Dataset

This data shows how employees rated various organizations. The data was gathered from Kaggle website. An extract of 500 unique glassdoor links for google employee review having 10 participants per link was used for this process. The rating was numeric within the range of (0 and 5) and this was divided into 5 classes (BAD (0,1); OKAY (2); AVERAGE(3); GOOD(4); BEST(5)).

4 Result and Discussion

4.1 Experimental Setup

The K-Means and expectation maximization algorithm are written in python using the Sci-kit learn machine learning library (Pedregosa et. al., 2011). The SOM is implemented in python using the open source MiniSOM implementation (Vettigli et. al., 2019). The majority voting algorithm is also written in python. All experiments are carried out on a Windows 10 operating system with 8GB of RAM.

4.2 Performance Metrics

The metrics used in this paper are Accuracy and Area Under the Receiver Operating Characteristic Curve (AUC ROC). Accuracy is the sum of true positives and true negatives, divided by the total number of examples in the sample. This ratio is multiplied by 100 to get a percentage integer. AUC ROC is a standard metric used to quantify the ability of a multiclass classifier to separate the sample data into their unique classes. In other words, it determines how good a model is by classifying or separating the data provided into their unique classes. ROC is a probability curve and AUC represents the measure or degree of separability. The value of this performance metric is between 0s and 1s. When AUC value = 1, then the model is able to separate between all the Positive and the Negative class points correctly. If, however, the AUC value had been 0, then the model would be predicting all Positives as Negatives and all Negatives as Positives.

4.3 Summary of Results

Table 2 summarizes all the experimental scenarios that was used across the crowdsourced datasets.

Table 2. Summary of all the experimental scenarios

Experimental Scenarios	Extract Class Labels (NO)	Extract Class Labels (YES)
Initialize Centroid(NO)	00	01
Initialize Centroid (YES)	10	11

Table 3 summarizes all the accuracy results showing the best performing models across the datasets. The best accuracy scores are shown for each model across all the datasets. The binary numbers in bracket signify the experimental scenario. Where 00 means "no initialize centroid and no extract class labels", 01 means "no initialize centroid and extract class labels", 10 means "initialize centroid and no extract class labels" and 11 means "initialize centroid and extract class labels". If no binary number is attached to the score that means the value is the same across all experimental scenarios.

http://doi.org/10.46792/fuoyejet.v7i2.800 http://journal.engineering.fuoye.edu.ng/

Table 3. Accuracy results for all models on all the datasets

Dataset	Majority voting	K-Means	Expectation Maximization	Self-Organizing Maps
Adult2	0.7540	0.7669 (10)	0.7411 (10)	0.7702 (10)
Weather Sentiments	0.6400	0.7212 (01)	0.6800 (01)	0.8800 (10)
Emotions (Anger)	0.41	0.41 (01)	0.41 (01)	0.41
Emotions (Disgust)	0.57	0.57 (01)	0.57 (01)	0.57
Emotions (Joy)	0.53	0.53 (01)	0.53 (01)	0.53
Emotions (Fear)	0.37	0.37 (01)	0.37 (01)	0.37
Emotions (Sadness)	0.36	0.36 (01)	0.36 (01)	0.36
Emotions (Surprise)	0.05	0.05 (01)	0.05 (01)	0.05
Valence	0.65	0.59 (01)	0.52(11)	0.64(00)
Employee Review	0.432	0.398 (01)	0.268(01)	0.44(00)
Average	0.4766	0.47661	0.44991	**0.50202**

The best performing model across all datasets and experimental scenarios is the self-organizing map implemented using the initialize centroid. The extract class labels mechanism provides an improvement for the k-means and expectation maximization algorithms but since the extract class labels mechanism does not show any effect on the SOM, it can be concluded that the SOM with initialized centroids is the best performing model on the ground truth inference task. Table 4 presents the ROC AUC summary for the models across the datasets. For each model, the best performance from the four experimental results is presented. The ROC AUC serves as a guide when deciding which experimental scenario to pick. From the results it is safer not to rely on the accuracy scores alone but to instead pick an experimental scenario where the accuracy scores and the ROC AUC scores tally with each other.

Table 4. ROC AUC results for all models on all the datasets

Dataset	Majority voting	K-Means	Expectation Maximization	Self-Organizing Maps
Adult2	0.7214	0.7581 (10)	0.7054 (10)	0.7475 (10)
Weather Sentiments	0.7755	0.8297 (01)	0.8803 (01)	0.9249 (10)
Emotions (Anger)	0.5008	0.5215 (10)	0.5165 (10)	0.5015 (11)
Emotions (Disgust)	0.5053	0.5060 (10)	0.5088 (10)	0.5056 (10)
Emotions (Joy)	0.5023	0.5023 (01)	0.5023 (01)	0.5023
Emotions (Fear)	0.5000	0.5080 (00)	0.5000 (01)	0.5000
Emotions (Sadness)	0.5026	0.5026 (01)	0.5026 (01)	0.5030 (01)
Emotions (Surprise)	0.5000	0.5039 (00)	0.5000 (01)	0.5000
Valence	0.6647	0.6451 (01)	0.6168 (11)	0.6607 (01)
Employee Review	0.5809	0.6125 (01)	0.5471 (10)	0.5910 (10)
Average	0.57535	0.58897	0.57798	**0.59365**

The results show that using the experimental scenario discussed above (that is, initializing centroid and extracting class labels) can improve the accuracy of the results. The self-organizing map performs far more accurately when used as compared to the k-means and expectation maximization algorithms. When the centroids of the k-means and expectation maximization algorithms were pre-initialized, a general increase in performance was observed, which is further enhanced by extracting classes directly from the clusters instead of relying on the algorithm's predictions. Once more, under these scenarios, the self-organizing map performs best achieving the highest accuracies and ROC AUC across all datasets. The self-organizing map's ability to fit the weights of the best matching unit to the samples in the dataset allows it to provide more accurate clusters. However, one does not need to extract class labels when using the self-organizing map because the same mechanism is employed already to make predictions using the algorithm. Finally, accuracy scores must be accompanied by another metric such as the ROC AUC score in order to establish an empirical evaluation of the algorithms ability to predict ground truth labels.

5 CONCLUSION

Crowdsourcing has been deployed severally to tackle tasks especially data gathering tasks and question answering tasks that would have required expensive experts and taken more time if they had been approached otherwise. Crowdsourcing allows such task to be completed on time and also at a lower cost. However, the inherent problem of noisy labels introduced by the "crowd" makes crowdsourced data unfit for decision making in their raw form until further processing is done to infer ground truth from them. This paper has presented some works done by different researchers in addressing noisy label problem in crowdsourced data. A comparative analysis of the ground truth inference ability of three clustering algorithms on crowd sourced datasets with different experimental scenarios (Initializing centroids and extracting class labels) was experimentally carried out and compared to the Majority Voting algorithm. The algorithms include, the self-organizing maps, the k-means and the expectation maximization clustering algorithm. In conclusion, this paper has demonstrated the superior ability of SOM for ground truth inference from crowdsourced data as compared to other clustering algorithms used. For the clustering algorithm, the computation of the centroids is the most important step and the algorithms gain a performance increase when the centroids are pre-initialized. Our experimental results support this notion as the

http://doi.org/10.46792/fuoyejet.v7i2.800 http://journal.engineering.fuoye.edu.ng/

algorithms achieve the highest performance when the centroids are initialized.

Classifying numerous data for various purposes remains a major task that needs to be accomplished and better improved upon. The classified data are important for machine learning and artificial intelligence, helping individuals or corporations to make the best decisions per time. In future, there is a need to study the optimal combination of machine learning algorithms to further reduce the noisy labels and to infer a better ground truth in crowdsourced data.

REFERENCES

Adeogun, Y., & Odumuyiwa, V. (August, 2019). A Comparative Analysis of Four Label Extraction Algorithms for Crowdsourced Data. 3rd international conference on Transition from Observation to Knowledge to Intelligence, (pp. 41-56). Lagos, Nigeria.

Bi, W., Wang, L., Kwok, J. T., & Tu, Z. (2014). Learning to predict from crowd sourced data. Uncertainty in Artificial Intelligence - Proceedings of the 30th Conference, UAI 2014, 82-91.

Cai, X., Nie, F., & Huang, H. (3–9 August 2013). Multi-View K-Means Clustering on Big Data. In F. Rossi (Ed.), Twenty-Third International Joint conference on artificial intelligence (pp. 2598-2604). Beijing, China: AAAI Press, Menlo Park, California.

Changyue, S., Kaibo, L., & Xi, Z. (2021). Collusion Detection and Ground Truth Inference in Crowdsourcing for Labeling Tasks. Journal of Machine Learning, 1-45.

Chittilappilly, A. I., Chen, L., & Amer-Yahia, S. (2016). A Survey of General-Purpose Crowdsourcing Techniques. IEEE Transactions on Knowledge and Data Engineering, 28(9), 2246-2266.

Cui, L., Chen, J., He, W., Li, H., Guo, W., & Su, Z. (2021). Achieving Approximate Global Optimization of Truth Inference for Crowdsourcing Microtasks. Data Science and Engineering, 294-309.

Dawid, A. D., & Skene, A. M. (1979). Maximum Likelihood Estimation of Observer Error-Rates Using the EM Algorithm. Royal Statistical Society, 28(1), 20-28.

Demartini, G., Difallah, D. E., & Cudré-Mauroux, P. (April 16 - 20, 2012). ZenCrowd: leveraging probabilistic reasoning and crowdsourcing techniques for large-scale entity linking. Proceedings of the 21st international conference on World Wide Web (pp. 469-478). Lyon France: WWW '12: Proceedings of the 21st international conference on World Wide Web.

Hernández-González, J., Inza, I., & Lozano, J. A. (2018). A Note on the Behavior of Majority Voting in Multi-Class Domains with Biased Annotators. IEEE Transactions on Knowledge and Data Engineering, 31(1), 195-200.

Howe, J. (2006). The Rise of Crowdsourcing. Wired, 14(6), 1-4.

Hung, N., Nguyen, T., Tran, L., & Aberer, K. (2013). An Evaluation of Aggregation Techniques in Crowdsourcing. International Conference on Web Information Systems Engineering (pp. 1-15). Springer, Berlin, Heidelberg.

Ipeirotis, P., Provost, F., & Wang, J. (25 July 2010). Quality Management on Amazon Mechanical Turk. Proceedings of the ACM SIGKDD workshop on human computation, (pp. 64-67). Washington DC: Association for Computing Machinery.

Li, C., Jiang, L., & Xu, W. (2019). Noise correction to improve data and model quality for crowdsourcing. Engineering Applications of Artificial Intelligence, 82, 184-191.

Moon, T. K. (1996). The expectation-maximization algorithm. Signal Processing Magazine, IEEE, 13(6), 47-60.

Simperl, E. (2015). How to Use Crowdsourcing Effectively: Guidelines and Examples. LIBER Quarterly, 25, 18-39.

Sinha, V. B., Sukrut, R., & Balasubramanian, V. N. (2018). Fast Dawid-Skene: A Fast Vote Aggregation Scheme for Sentiment Classification. Retrieved from https://arxiv.org/abs/1803.02781

Welinder, P., Branson, S., Belongie, S., & Perona, P. (2010). The Multidimensional Wisdom of Crowds. Advances in neural information processing systems, 23, 2424–2432.

Whitehill, J., Wu, T.-f., Bergsma, J., Movellan, J. R., & Ruvolo, P. L. (2009). Whose vote should count more: Optimal integration of labels from labelers of unknown expertise. Advances in neural information processing systems, 22, 2035-2043.

Yu, D., Liangxiao, J., & Chaoqun, L. (2022). Improving data and model quality in crowdsourcing using co-training-based noise correction. Information Sciences, 174-188.

Zhang, J., Sheng, V. S., & Wu, J. (2019). Crowdsourced Label Aggregation Using Bilayer Collaborative Clustering. IEEE Transactions on Neural Networks and Learning Systems, 30(10), 3172-3184.

Zhang, J., Sheng, V. S., Wu, J., & Wu, X. (2015). Multi-class ground truth inference in crowdsourcing with clustering. IEEE Transactions on Knowledge and Data Engineering, 28(4), 1080-1085.

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

Xu, W., Jiang, L., & Li, C. (2020). Resampling-Based Noise Correction for Crowdsourcing. *Journal of Experimental & Theoretical Artificial Intelligence, 33(6),* 985-999.

Xu, W., Jiang, L., & Li, C. (2021). Improving Data and Model Quality in Crowdsourcing using Cross-Entropy-Based Noise Correction. *Information Sciences, 546,* 803-814.

Jung, Y. G., Kang, M. S., & Heo, J. (2014). Clustering Performance Comparison using K-Means and Expectation Maximization algorithms. *Biotechnology & Biotechnological Equipment 28, S44-S48*.

Kohonen, T. (1990). The self-organizing map. *Proceedings of the IEEE, 78(9),* 1464 - 1480.

Venanzi, M., Teacy, W., Rogers, A., & Jennings, N. (2015). Weather Sentiment - Amazon Mechanical Turk dataset. *University of Southampton Institutional Repository*, doi:10.5258/SOTON/376543 [Online]. Available: https://eprints.soton.ac.uk/376543/. [Accessed November 2020].

Snow, R., O'Connor, B., Jurafsky, D., & Ng, A. (2008). Cheap and Fast – But is it Good? Evaluating Non-Expert Annotations for Natural Language Tasks. *Proceedings of the 2008 conference on empirical methods in natural language processing, Honolulu, Hawaii,* October 2008.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., & Cournapeau, D. (2011). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research, 12,* 2825-2830.

Vettigli, G. (2019). MiniSom: minimalistic and NumPy based implementation of the Self Organizing Map. 2019. [Online]. Available: https://github.com/JustGlowing/minisom. [Accessed 24 September 2020].