Knowledge-Based Systems 272 (2023) 110551

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

Unsupervised classification for uncertain varying responses: The wisdom-in-the-crowd (WICRO) algorithm

Nir Ratner [a], Eugene Kagan [b], Parteek Kumar [c,*], Irad Ben-Gal [a]

[a] Department of Industrial Engineering, Tel-Aviv University, Ramat-Aviv, Tel-Aviv 69978, Israel
[b] Department of Industrial Engineering and Management, Faculty of Engineering, Ariel University, Ariel 40700, Israel
[c] Department of Computer Science and Engineering, Thapar Institute of Engineering and Technology, Patiala, 147004, India

ARTICLE INFO

Article history:
Received 26 March 2022
Received in revised form 22 November 2022
Accepted 4 April 2023
Available online 10 April 2023

Keywords:
Wisdom-of-the-crowd
Expert voting
Unsupervised classification
Data analysis
Online decision-making

ABSTRACT

This paper addresses the problem classification of instances/questions based on the opinions (classes) provided by anonymous agents. The solution aggregates the agents' classifications, aiming to obtain as close as possible to an unknown correct classification. However, the agents' fields or domains of competence and their levels of expertise are unknown and can vary extensively. Many popular classification algorithms address such a problem by following a "wisdom-of-the-crowd" approach while using different voting methods and expectation–maximization techniques. These algorithms lead to correct classifications when the majority of the agents are experts, thus classifying the instances correctly. However, they often result in erroneous classification when only a small subset of the agents are indeed correct. Moreover, these algorithms often assume a fixed set of classes for all instances. This study presents a fast (one-pass) classification algorithm that can estimate the unknown agents' expertise level and aggregates their classifications accordingly, even when these are obtained from different questionnaires; thus, when the instances are not necessarily classified to a fixed set of classes. The proposed algorithm finds the experts and the nonexpert agents for each question by analyzing the distance between them. The algorithm identifies the expert agents for each instance and then classifies them accordingly. The suggested algorithm is validated and compared against known methods by using both simulated datasets and real-world datasets collected from various sources. The obtained results clearly demonstrate the effectiveness and advantages of the proposed method.

© 2023 Published by Elsevier B.V.

1. Introduction

Classification of entities, variables, instances, questions or objects is a basic process that precedes many data processing algorithms. It can be performed by certain classification criteria or by labeling similar entities with the same tag. A primary challenge is finding the classification rules, i.e., the clustering criteria or the similarity measures, which lead to correct classifications.

In cases when a subset of correct classifications is known, classification criteria can be defined directly by observing the classes obtained by supervised learning [1]. In contrast, in cases where the correct classification is unknown, the definition of the classification criteria is problematic, and some type of 'unsupervised classification' (as we call it here) process must be developed by relying on other principles.

The unsupervised classification process can follow several approaches. In one approach, classification begins by clustering subsets of entities that are similar to each other in some sense, extracting classification rules based on these clusters, and then iteratively using the obtained rules to classify new entities and refine the rules until convergence to a final classification [1]. In a second approach, which is more related to this study, classification is defined based on the opinions of a group of agents or by following the classification criteria that they provide. This technique, when following majority voting, is also known as *the wisdom-of-the-crowd* - thus, labeling is obtained by aggregation of the labels provided by a "crowd" of agents. This approach is supported by several platforms, such as Amazon Mechanical Turk or CrowdFlower [2], and has been investigated in various studies [3,4].

In this paper, we implement the second approach and consider the methods of aggregating the agents' classifications. A simple and popular opinion aggregation method is based on majority voting, as indicated above. Following this approach, an entity is labeled or tagged to the class, which was chosen by the majority of the agents in a direct classification process. It is well known that majority voting can result in erroneous classification if the majority of the experts are wrong. For example, in a medical diagnosis case, the aggregated answers of several nonphysician

[*] Corresponding author.
E-mail address: parteek.bhatia@thapar.edu (P. Kumar).

https://doi.org/10.1016/j.knosys.2023.110551
0950-7051/© 2023 Published by Elsevier B.V.

agents are often less valuable than the opinion of a single physician, who is an expert in that domain (we use the terms *field* and *domain* interchangeably); therefore, under these conditions, majority voting is inadequate [5]. In addition, majority voting is misleading if there are biased voters [6], and it has been observed that confidence judgments in the responses to two-alternative forced-choice instances are correlated with the consensus in the responses rather than with their accuracy [7].

To address these challenges or at least to minimize the influence of the gap between expert and nonexpert agents, agent expertise plays an important role in deriving the appropriate judgment in crowd opinion aggregation models [8]. In recent decades, an approach implementing two-stage classification procedures has been intensively studied [9]. In such procedures, agents are categorized with respect to their levels of expertise, and then the classification is conducted [10]. However, applying these algorithms to real-world datasets met an additional problem of lack of data, which led to incompletely filled questionnaires or questionnaires with different options for each question. In such situations, the activity of most algorithms is reduced to basic majority voting and often leads to erroneous classifications.

In this paper, we introduce an unsupervised classification algorithm that effectively processes datasets with a lack of records and works well in datasets with different classification options per instance. The proposed algorithm aggregates the agents' opinions and identifies the expert and the nonexpert agents when classifying each instance. In contrast to the known *wisdom-of-the-crowd* approach, it follows a scheme that we call 'wisdom in the crowd' (WICRO), thus identifying those expert agents and classifying the instances according to their responses. The proposed WICRO algorithm is applicable to datasets in which the possible classes or labels per instance are not necessarily fixed. In addition, the proposed algorithm enables expert identification and classification in crowdsourcing environments with varying domains of agent expertise.

The WICRO algorithm assumes that in a group of agents, some agents are experts in specific fields or domains but not necessarily in all of them. Moreover, in some domains, there may be no experts at all among the agents, which may be the case when a questionnaire is executed over online crowdsourced platforms, such as Amazon Mechanical Turk and CrowdFlower. These platforms provide an interface for agents (workers) to answer questionnaires or label datasets for payment. However, these agents may not possess the necessary domain knowledge to answer the questions correctly or provide the correct labels for all instances.

In addition, the WICRO algorithm further assumes that experts will often have similar or even identical answers (or classifications) in instances that are associated with their domain of expertise; however, their answers may differ substantially in other domains. Accordingly, the WICRO classification uses the distance between the various agents' answers to identify whether they may be experts or nonexperts in the considered domain. Then, the labels provided by the agents, which are identified as experts in the relevant domain, are used to obtain a correct classification in each related instance, as seen in the next sections.

The rest of the paper is organized as follows. Section 2 reviews the relevant classification methods that inspired this work. Section 3 includes a formal description of the problem. Section 4 discusses the limitation of existing approaches for datasets having different options per instance. Section 5 presents the proposed algorithm and a running example. In Section 6, the algorithm is extended for the use of knowledge domain information. Section 7 presents the verification results, and Section 8 concludes this work.

2. Related works

Theoretical and applied research in classification can be traced back to the arrangement methods proposed by Dirichlet. In the case of uncertain classes or classification criteria, combinatorial techniques are often accompanied by voting methods, where the majority of the agents support aggregated opinion classifications. Often, opinion aggregation is conducted by majority voting, which provides reasonable results for tasks considered by a group of expert agents. If, in contrast, the group of agents includes both experts and nonexperts, majority voting can fail and result in erroneous classifications.

Several approaches have been suggested to address these problems that implement different statistical techniques. The most popular and successful approach implements the model suggested by Dawid and Skene [11]. The Dawid–Skene (DS) model is based on the expectation–maximization (EM) algorithm, which provides maximum likelihood estimates of individual error rates. Several classification algorithms were suggested in later developments of this approach. In particular, Whitehill et al. [9] proposed a probabilistic algorithm for binary classification that infers image labeling by considering the level of expertise of the agents and the difficulty level of each image. This model was tested on both simulated and real data and demonstrated its robustness to noisy and adversarial labelers.

Duan et al. [12] improved the DS model by utilizing label dependency and suggested three methods for estimating multiple true labels per instance in the datasets. The first method, denoted as D-DS, incorporates dependency relationships among all labels; the second method, denoted as P-DS, groups labels into pairs to prevent interference from uncorrelated labels; and the third method, denoted as ND-DS, is a Bayesian network label-dependent DS model, which compactly represents label dependency using conditional independence properties to overcome the sparse data problem.

Researchers have also proposed various approaches for crowdsourced datasets. Montejo et al. [13] proposed crowd-explicit sentiment analysis (CESA) as an approach for sentiment analysis in social media environments. They evaluated the proposed polarity classification system using English and Spanish datasets. Zhang et al. [14] proposed multiclass ground truth inference in crowdsourcing with clustering. The authors proposed ground truth inference using clustering (GTIC) to improve the integrated label quality for multiclass labeling. Noise filtering was proposed by Li et al. [15] to improve data and model quality for crowdsourcing. In this study, an attempt was made to employ noise filters to delete the noise in integrated labels, enhancing the training data and model quality. The authors empirically investigated the performance of noise filters in terms of improving crowdsourcing learning. Prelec et al. [16] suggested the 'surprisingly popular' (SP) algorithm. This model asks respondents to classify instances and to predict the distribution of other people's answers to the question. The model selects the answer that gains more support than predicted. A weighted rank aggregation approach to crowd opinion analysis was proposed by Chatterjee et al. [8]. They used rank-based features to exploit annotator quality instead of using only annotators' accuracy and bias as important features. Hagerer et al. [17] used end-to-end annotator bias approximation on crowdsourced single-label sentiment analysis. The proposed approach suggests improvement for precise neural end-to-end bias modeling and ground truth estimation, which reduces an undesired mismatch.

Further development of the DS model introduced additional measures and algorithm improvements. Shah et al. [18] proposed a permutation-based DS model with a novel error metric to compare different estimators. Sinha et al. [19] also developed a method called the fast Dawid–Skene algorithm (FDS), which converges to the estimated labels at a linear rate. Ibrahim et al. [20] proposed a framework using pairwise co-occurrences of the annotator responses by using an algebraic algorithm convex geometry-based structured matrix factorization to efficiently solve the model identification problem. They showed that the approach can identify the Dawid–Skene model under realistic conditions.

Schmidt and Zdeborová [21] analyzed a noisy dense limit of the Dawid–Skene model and showed that it belongs to a larger class of low-rank matrix estimation problems for which it is possible to express the Bayes-optimal performance for large system sizes in a simple closed form.

Finally, in 2021, Ghanaiem et al. [10], based on the DS approach, developed an efficient collaborative classification algorithm (DSC^2 algorithm) that provides effective solutions for large and small datasets considered by agents with various levels of expertise. First, the algorithm categorizes the agents with respect to their level of expertise and then partitions the given set of entities with respect to the agents' expertise levels so that the opinions of nonexpert agents are used with lower weights or even ignored.

Recently, Eshkevari et al. [22] proposed an end-to-end ranking method for integrating mechanisms such as text processing, sentiment analysis, and multicriteria decision-making. The proposed ranking method relies on integrating three methods: the aspect-based sentiment analysis (ABSA) method, the Dawid–Skene algorithm, and the best-worst method (BWM).

Current research continues the methods suggested by Shah et al. [18], Sinha et al. [19], Ghanaiem et al. [10], and Eshkevari et al. [22] that follow the line of the DS approach. The existing work in this direction is limited to finding the experts in the dataset, where an agent has to choose an appropriate response from the given set of fixed classes for each instance. For example, consider a dataset, where for each instance, the agent has to choose one option from a given set of fixed classes, such as (a) pneumonia, (b) bronchitis, (c) cyst tumors, (d) asthma, and (e) fractures, by looking at the input X-ray image. In this case, the classes are the same for each instance, and the agent has to choose the most suitable option for the given instance. Note, however, that we may have a different situation where the classes for each instance differ, depending on a relevant domain. For example, in the case of academic multiple-choice questions/sports quizzes/medical tests of varied types, the classes/options for each instance are different, and the agent has to choose the most appropriate response from the given set of classes. The novelty of the proposed algorithm is its capability to identify the experts for each instance in this setting, although the classes are different for each instance. In fact, for the case of a dataset having different classes for each instance, most existing methods are reduced to majority voting because they are designed to find the similarity distance between different options. However, in contrast to these methods, the proposed WICRO algorithm does not assume equivalence of the lists of possible labels for different instances. Therefore, it applies to general classification problems with varying entity characteristics.

3. The problem setup

The considered problem is known as an unsupervised collective classification under uncertainty problems. Given a set of instances (such as entities, records, and questions) and a set of possible classes (or tags), a set of agents is required to classify the instances (thus to partition the set), such that it is as close as possible to the unknown correct classification. The classification process implies that each agent chooses one of the relevant classes for each instance, and the final classification is obtained by aggregating the agents' classifications.

In a conventional classification process, all the instances are classified to fix a set of classes. However, in more general scenarios, the instances might be associated with different knowledge domains; therefore, they might have different sets of classes rather than a fixed set. Moreover, each agent may classify only a subset of instances (e.g., respond to some of the questions), while other instances are ignored. In these scenarios, most existing algorithms, including new algorithms that rely on the DS approach, are reduced to majority voting or simply fail. The purpose of the proposed algorithm is to address these general classification scenarios.

Let $X = \{x_1, x_2, \ldots, x_n\}$ be a set of n instances associated with certain entities or objects that should be labeled and distributed over $l \leq n$ classes $C_j \subset X$ such that the resulting partition $\alpha = \{C_1, C_2, \ldots, C_l\}$ represents available information about the characteristics of the entities from set X. For example, assume that instance x represents paintings, and the required task is to define the painter of certain paintings. Then, the C classes are associated with the artists, and the classification problem requires relating each instance $x_i \in X$, where $i = 1, 2, \ldots, n$, to the class $C_j \subset X$, where $j = 1, 2, \ldots, l$. The resulting partition $\alpha = \{C_1, C_2, \ldots, C_l\}$ completely defines the painters of the paintings. As a more complicated example, consider the entities x as symptoms and the classes C as diseases. Then, the diagnostic result is also a set α of classes, but since one symptom can appear for several diseases, different classes from α are not disjointed, and the set α is a cover of the set X. A more general case is defined below.

Assume that the classification α is generated by a group of m agents $A = \{a_1, a_2, \ldots, a_m\}$. In the first example, the agents can be people (e.g., general experts, specialists, or nonexperts) who relate the painting to the artist. In another example, the agents are persons (physicians who are experts in certain diseases, physicians who are nonexperts in certain diseases, or nonprofessional people) who diagnose the disease.

To formalize the classification process, we assume that each instance x_i is related to a multichoice questionnaire Q_i such that only one choice (class) is the correct choice; if l is the number of possible classes, then each questionnaire Q_i includes $l_i \leq l$ options, while each option is the index of class C in the list of l possible classes. Considering the questionnaire Q_i, where $i = 1, 2, \ldots, n$, agent a_k, where $k = 1, 2, \ldots, m$, chooses an index of the class from Q_i and saves it as a value r_{ik}.

Accordingly, the problem is formulated as follows: given the set $X = \{x_1, x_2, \ldots, x_n\}$ of instances, the set $A = \{a_1, a_2, \ldots, a_m\}$ of agents and the matrix $R = \|r_{ik}\|_{n\times m}$ of the agent's choices, find a classification $\alpha = \{C_1, C_2, \ldots, C_l\}$ (i.e., a partition or cover of the set X), which is as close as possible to the correct (yet unknown) classification $\alpha^* = \left\{C_1^*, C_2^*, \ldots, C_l^*\right\}$.

An immediate solution to the formulated problem can be obtained by direct implementation of the majority-voting technique, in which the classes are created as follows. Instance x_i is classified or mapped into class C_j if its index j appears a maximum number of times in the ith row of matrix R (ties are broken randomly). As described above, this method follows the principle of "one agent – one voice" and results in effective solutions if most of the agents are indeed experts in the domain associated with that row. However, incorrect classification can occur for groups of agents with varying or biased levels of expertise.

Nonetheless, the problem of interest can be addressed by methods that are fundamentally different than majority voting. One such approach that we follow in this study is by preprocessing the data and recognizing the agents who are experts in the classification domain per instance and then further considering their opinions, unlike the nonexpert agents. As said, the proposed WICRO algorithm follows the second approach, which is fundamentally different from 'the wisdom-in-the-crowd' principle.

Table 1
The dataset for $n = 3$ instances distributed into $l = 9$ classes by $m = 4$ agents.

	a_1	a_2	a_3	a_4
x_1	1	2	2	3
x_2	4	5	6	6
x_3	7	8	7	9

4. Limitations of existing approaches to datasets with different sets of classes for each instance

To clarify the considered classification problem, let us denote unequal questionnaires by $Q_i \neq Q_j$ for $i \neq j$, where $i = 1, 2, \ldots, n$, and apply to it the recently developed DS-based algorithm [10] for collaborative classification (DSC^2 algorithm) as well as the well-known fast Dawid–Skene algorithm [19].

Following the DSC^2 algorithm, agents a_k and $k = 1, 2, \ldots, m$ are first categorized according to their levels of expertise. Then, the agents' classifications α_k are executed with respect to these levels, and the classifications of agents with higher levels of expertise obtain a greater influence on the final classification α than the classifications provided by agents with lower levels of expertise. Specification of the expertized agents is based on weighted Hamming distances $d_{norHam}(\alpha_u, \alpha_v | j)$ between classifications α_u and α_v, where $u, v = 1, 2, \ldots, m$, with respect to the classes C_j, and $j = 1, 2, \ldots, l$. The weighted Hamming distance is defined as follows [10]:

$$d_{norHam}(\alpha_u, \alpha_v | j) = \frac{n(\alpha_u, \alpha_v | j)}{(n(\alpha_u | j) + n(\alpha_v | j))},$$

where $n(\alpha_u | j)$ is the number of times agent α_u selects class j, and $n(\alpha_u, \alpha_v | j) = \#\left(\left(C_j^u \cup C_j^v\right) \setminus \left(C_j^u \cap C_j^v\right)\right)$ is the cardinality of the *symmetric difference* between classes C_j^u and C_j^v, which represents the disagreement among agents about the jth class.

Following this approach, agents are categorized by the questions in the questionnaires rather than by their real levels and domains of expertise when questions with different options are utilized. As a result, the construction of the final classification α is reduced to majority voting. At the final stage, the DSC^2 algorithm acts as a majority-voting process among the agents that answer the specific question, while the other agents are ignored. A similar output occurs in the other classification algorithms that follow the DS approach, with obvious differences in similarity measures.

For example, assume that the dataset consists of $n = 3$ instances considered by $m = 4$ agents, and there are $l = 9$ possible classes among which the agents can distribute the instances. In other words, each agent a_j, where $j = 1, 2, 3, 4$, specifies a number $r_{ij} \in \{0, 1, 2, \ldots, 9\}$ to each instance x_i, where $i = 1, 2, 3$ The agent classifications are shown in Table 1.

In terms of the agents' partitions α_j of the set $X = \{x_1, x_2, x_3\}$ of instances, this dataset is represented as follows. Each agent's partition includes three nonempty classes (with the corresponding indices)

$$\alpha_1 = \{C_1 = \{x_1\}, C_4 = \{x_2\}, C_7 = \{x_3\}\},$$

$$\alpha_2 = \{C_2 = \{x_1\}, C_5 = \{x_2\}, C_8 = \{x_3\}\},$$

$$\alpha_3 = \{C_2 = \{x_1\}, C_6 = \{x_2\}, C_7 = \{x_3\}\},$$

$$\alpha_4 = \{C_3 = \{x_1\}, C_6 = \{x_2\}, C_9 = \{x_3\}\},$$

while the other six classes in each partition are empty.

For illustration, the weighted Hamming distances between the agents' classifications with respect to classes C_1, C_2, C_3 and C_4 are shown in Table 2. For example, the disagreement of the first and the second agents about class C_1 is $n(\alpha_1, \alpha_2 | 1) = 1$. In fact, partition α_1 includes class C_1, which includes a single instance x_1, while in partition α_2 class C_1 is empty. Thus,

$$\begin{aligned} n(\alpha_1, \alpha_2 | 1) &= \#\left(\left(C_1^1 \cup C_1^2\right) \setminus \left(C_1^1 \cap C_1^2\right)\right) \\ &= \#\left(\left(C_1^1 \cup \varnothing\right) \setminus \left(C_1^1 \cap \varnothing\right)\right) = \#\left(C_1^1 \setminus \varnothing\right) = \# C_1^1 = 1, \end{aligned}$$

$$n(\alpha_1 | 1) = 1,\ n(\alpha_2 | 1) = 0, \text{ and } d_{norHam}(\alpha_1, \alpha_2 | 1) = \frac{1}{(1 + 0)} = 1.$$

In contrast, the disagreement of the second and third agents about class C_2 is $n(\alpha_2, \alpha_3 | 2) = 0$ since both partition α_2 includes class C_2 and partition α_3 includes class C_2, and in both partitions, this class includes the same single instance x_1. Thus,

$$\begin{aligned} n(\alpha_2, \alpha_3 | 2) &= \#\left(\left(C_2^2 \cup C_2^3\right) \setminus \left(C_2^2 \cap C_2^3\right)\right) \\ &= \#\left(\left(\{x_1\} \cup \{x_1\}\right) \setminus \left(\{x_1\} \cap \{x_1\}\right)\right) \\ &= \#\left(\{x_1\} \setminus \{x_1\}\right) = \#\varnothing = 0, \end{aligned}$$

$$n(\alpha_2 | 2) = 1\, n(\alpha_3 | 3) = 1 \text{ and } d_{norHam}(\alpha_2, \alpha_3 | 2) = \frac{0}{(1 + 1)} = 0.$$

The dashes in the table denote the cases in which the considered class is empty in the partitions of both agents.

For each class, the group of experts includes the agents that considered this class and have chosen it for at least one instance. However, if the possibility of choosing this class is not included in the questionnaire, the agents are automatically considered nonexperts, and their opinions are not counted.

Accordingly, at the final stage, the DSC^2 algorithm acts as a majority-voting process among the agents that answer the specific question, while the other agents are ignored. A similar problem occurs in other classification algorithms that follow the DS approach, with obvious differences in the similarity measures.

In the fast Dawid–Skene (FDS) algorithm, for example, the first "proposed true choices" are generated by using a majority vote. The following steps proposed in the original algorithm cannot be completed in an environment where the dataset has different classes for each instance (there will not be any progress on the convergence step for this kind of dataset). As a result, the FDS also performs like a majority vote for datasets with varied classes per instance.

5. The suggested algorithm

The suggested classification algorithm follows the assumption that experts in a certain domain will often classify instances in that domain similarly, e.g., give similar answers to questions related to the domain, while nonexperts will tend to answer differently.

5.1. Outline of the WICRO algorithm

The algorithm consists of two main procedures. The first procedure identifies the expertized agents, and the second procedure classifies the instances using the opinions of recognized experts.

For each instance, the algorithm creates clusters of agents who are in agreement regarding certain instances and defines the group of experts for each instance as the cluster of agents with the highest agreement ratio. The opinions of the recognized experts are used for categorizing the instances by majority voting. The algorithm is outlined as follows.

The main part of the algorithm identifies expert agents. Classification is considered a ready-to-use procedure and can be conducted by any voting algorithm; in the scenarios presented in this paper, we implement the simplest majority-voting techniques, which maintain a fair comparison with respect to simple and common 'wisdom-in-the-crowd' methods.

Table 2
The distance between the agents' classifications with respect to classes.

c	C_1				C_2				C_3				C_4			
	a_1	a_2	a_3	a_4	a_1	a_2	a_3	a_4	a_1	a_2	a_3	a_4	a_1	a_2	a_3	a_4
a_1	0	1	1	1	0	1	1	–	0	–	–	1	0	1	1	1
a_2	1	0	–	–	1	0	0	1	–	0	–	1	1	0	–	–
a_3	1	–	0	–	1	0	0	1	–	–	0	1	1	–	0	–
a_4	1	–	–	0	–	1	1	0	1	1	1	0	1	–	–	0
Σ	3	–	–	–	–	2	2	–	–	–	–	3	3	–	–	–

For the remaining classes $C_5, \ldots, C_9$ the distances are defined in the same manner.

Table 3
Sample dataset for $\boldsymbol{n} = \mathbf{9}$ instances considered by $\boldsymbol{m} = \mathbf{8}$ agents.

	a$_1$	a$_2$	a$_3$	a$_4$	a$_5$	a$_6$	a$_7$	a$_8$	**Domain of knowledge**
x_1	0	0	1	3	2	4	3	2	Sports
x_2	1	1	2	3	2	3	1	4	Sports
x_3	2	2	1	0	3	1	4	1	Sports
x_4	4	0	3	3	2	2	1	0	Politics
x_5	4	1	4	4	0	3	3	2	Politics
x_6	4	1	1	1	2	0	0	3	Politics
x_7	3	2	0	4	1	1	3	3	Movies
x_8	1	0	4	1	2	2	0	1	Movies
x_9	4	0	1	3	3	3	2	1	Movies

The algorithm requires a single parameter y, which is the normalization factor. This parameter controls the influence of the number of agents in a cluster on the cluster's score. For a higher y value, the cluster's score increases, and there is a higher probability that the algorithm will result in a classification similar to the classification selected by a simple majority voting without identifying expert agents. A lower y value, in contrast, leads to results that are less influenced by the number of agents in each class and, as a result, will differ from the classifications provided by majority voting. [In such cases, the classification of the recognized experts will more strongly influence the final clustering outcome of the algorithm]

5.2. Running example

A simple example is used to further illustrate and clarify the algorithm. Let us set the normalization factor $y = 0.05$.

Assume that set X includes $n = 9$ instances, which are considered by group A of $m = 8$ agents. The questionnaire Q_i, where $i = 1, 2, \ldots, 9$, for each instance includes 5 options enumerated by integers 0–4. Each instance x_i is associated with a certain domain of knowledge from the set {*sports*, *politics*, *movies*}.

The agents were questioned regarding the instances, and each agent a_j, where $j = 1, 2, \ldots, 8$, specified a number $r_{ij} \in \{0, 1, 2, 3, 4\}$ to each instance x_i. The resulting matrix R is presented in Table 3.

The correct classification is generated by a random distortion such that

$$\alpha^* = \{\{x_1, x_6\}, \{x_2, x_7\}, \{x_3, x_8\}, \{x_4, x_9\}, \{x_5\}\}.$$

Note that as above, the classification is an ordered set. Thus, "0" is the correct classification for $\{x_1, x_6\}$, "1" is the correct classification for $\{x_2, x_7\}$, etc. However, that majority-voting results in the following classification (ties are broken randomly)

$$\alpha_{maj} = \{\{x_1\}, \{x_2, x_3, x_6, x_8\}, \{x_4\}, \{x_7, x_9\}, \{x_5\}\}.$$

where for the instances x_1 and x_4, classes C_0, C_2 and C_3 obtain an equal number of votes. Therefore, this classification by majority voting is not a single solution.

Let us now illustrate the WICRO algorithm. Consider the first instance x_1. Agents a_1 and a_2 indicated this instance by $r_{11} = r_{12} = 0$, which implies that they associate this instance with class C_0. Agents a_4 and a_7 classify this instance by $r_{14} = r_{17} = 3$, which implies that they associate this instance with class C_3. Finally, agents a_5 and a_8 classify this instance by $r_{15} = r_{18} = 2$, which means that they associate this instance with class C_2.

Now, the pair of agents a_1 and a_2 is considered, and the number of their agreements is computed. These agents identically classify instances x_1, x_2 and x_3 and disagree on the other instances' classification. Thus, the number of their agreements is 3, which means that in the first cluster, the agents agree about $\rho_1 = 3$ among $n = 9$ instances.

Recall (see line 14 in the WICRO algorithm) that the score of the cluster is defined as a sum of the normalized number of subclusters and the rate $\frac{\rho_1}{n}$ of agreement. In the considered case, the number of subclusters is given by the number $m_1 = 2$ of agents in the cluster. Using the indicated values, the score s_1 of cluster $\zeta_1 = \{a_1, a_2\}$ is

$$s_1 = \frac{\rho_1}{n} + m_1 y = \frac{3}{9} + 2 \times 0.05 = 0.43.$$

Calculating the similarity scores for clusters $\zeta_2 = \{a_4, a_7\}$ and $\zeta_3 = \{a_5, a_8\}$ (both having identical clustering in instance x_1) results in the same score value (0.21).

Let us illustrate the calculation of the similarity score for a cluster with a larger number of agents. Consider the second instance x_2. Agents a_1, a_2 and a_7 classify this instance into class C_1; thus, they form a cluster $\zeta_4 = \{a_1, a_2, a_7\}$.

Following the proposed algorithm, consider the following binary subsets B of cluster ζ_4: $B_{41} = \{a_1, a_2\}$, $B_{42} = \{a_1, a_7\}$ and $B_{43} = \{a_2, a_7\}$. For cluster B_{41}, the number of agents' agreements, $\rho_{41} = 3$; for cluster B_{42}, $\rho_{42} = 2$; and for cluster B_{43}, $\rho_{43} = 2$.

Since the number of agents in cluster ζ_4 is $m_4 = 3$, the average score s_4 of cluster $\zeta_4 = \{a_1, a_2, a_7\}$ is calculated as follows:

$$s_4 = \frac{\left(\frac{\rho_{41}}{n} + \frac{\rho_{42}}{n} + \frac{\rho_{43}}{n}\right)}{m_4} + m_4 y = \frac{\left(\frac{3}{9} + \frac{2}{9} + \frac{2}{9}\right)}{3} + 3 \times 0.05 = 0.409.$$

After calculating the scores for all the clusters per instance, the classification of the agents from the cluster with the maximum score is considered the true classification, and the instance is associated with the class chosen by these agents.

The calculation of the scores and the chosen classes are summarized in Table 4.

From the presented calculations according to the suggested algorithm, it follows that the resulting classification is:

$$\alpha = \{\{x_1\}, \{x_2, x_6, x_7\}, \{x_3, x_4, x_8\}, \{x_9\}, \{x_5\}\}.$$

The comparison of the obtained classification with the correct classification α^* demonstrates that it is more accurate than the classification α_{maj} obtained by majority voting: among $n = 9$ instances, the algorithm correctly classified 7 instances, while majority voting correctly classified only 4.

This example illustrates the properties of the proposed algorithm. The algorithm obtains the data representing the agent's responses and requires a single free parameter y. As an output,

Algorithm 1: Classification by "the wisdom in the crowd"

```
Input:  set $X = \{x_1, x_2, \dots, x_n\}$ of instances,
        set $\eta = \{Q_1, Q_2, .., Q_n\}$ of questionnaires (with respect to
        instances),
        set $A = \{a_1, a_2, \dots, a_m\}$ of agents,
        number $l$ of classes,
        normalization factor $y$.
Output: classification $\alpha = \{C_1, C_2, \dots, C_l\}$.

For each instance $x_i \in X$, $i = 1,2,\dots,n$, do
    For each agent $a_k \in A$, $k = 1,2,\dots,m$, do
        From the questionnaire $Q_i$ (with respect to the instance
        $x_i$), choose an appropriate class $C_j$ and save its index $r_{ik} =$
        $j$.
    End
End
For each instance $x_i \in X$, $i = 1,2,\dots,n$, do
    Consider row $r_{i\dots}$ (with respect to instance $x_i$) and create a
    cluster $\zeta \subset A$ of agents with the same values $r_{ik}$, $k = 1,2,\dots,m$.
End
For each cluster $\zeta$ do
    Create all possible subclusters $B$ including two agents.
    For each subcluster $B$ do
        Calculate the ratio $\rho$ of instances $x_i \in X$, where $i = 1,2,\dots,n$,
        for which agents $a_k$ and $a_{k'}$ from subcluster, $B$ agree (have
        equivalent $r_{ik}$ and $r_{ik'}$ values).
    End
    Calculate the score $s$ of cluster $\zeta$:
                 $s = \frac{sum\ of\ the\ ratios\ \rho}{\#subclusters\ in\ A_i} + ym,$
    where $m = \#\zeta$ is the number of agents in the cluster, and $y$
    is the normalization factor.
End
Among the clusters, $\zeta$ chooses the cluster $\zeta'$ with the highest
score $s'$ (the cluster of experts).
Create classification $\gamma = \{C_1, C_2, \dots, C_l\}$ of the set $X = \{x_1, x_2, \dots, x_n\}$ of
instances by the opinions of the agents from the cluster $\zeta'$.
```

the algorithm returns an array of integer labels of the classes that define the required classification of the instances.

In addition, the correspondence between clusters and domains of knowledge demonstrates that agents a_1 and a_2 are experts in sports, agents a_3 and a_4 are experts in politics and agents a_5 and a_6 are experts in movies.

The remaining agents a_7 and a_8 can be considered nonexperts in these domains. The next algorithm, which encompasses Algorithm 1, utilizes this information.

6. The use of knowledge domains

In WICRO Algorithm 1, the agents were categorized by their influence on the group's opinion without considering the fields of their expertise and the domains of knowledge of the considered instances. However, in practical tasks, in some instances, the domain of knowledge is either known ahead or can be easily recognized. In such a case, the suggested algorithm can be supplemented by additional operations that improve the accuracy of the

Table 4
Scores of the clusters and chosen classes.

	Cluster	Choice	Score	Experts' cluster	Choice
x_1	$\{a_1, a_2\}$	0	0.43	$\{a_1, a_2\}$	0
	$\{a_4, a_7\}$	3	0.21		
	$\{a_5, a_8\}$	2	0.21		
x_2	$\{a_1, a_2, a_7\}$	1	0.41	$\{a_1, a_2, a_7\}$	1
	$\{a_3, a_5\}$	2	0.21		
	$\{a_4, a_6\}$	3	0.32		
x_3	$\{a_1, a_2\}$	2	0.43	$\{a_1, a_2\}$	2
	$\{a_3, a_6, a_8\}$	1	0.30		
x_4	$\{a_2, a_8\}$	0	0.21	$\{a_5, a_6\}$	2
	$\{a_3, a_4\}$	3	0.43		
	$\{a_5, a_6\}$	2	0.54		
x_5	$\{a_1, a_3, a_4\}$	4	0.41	$\{a_1, a_3, a_4\}$	4
	$\{a_6, a_7\}$	3	0.32		
x_6	$\{a_2, a_3, a_4\}$	1	0.37	$\{a_2, a_3, a_4\}$	1
	$\{a_6, a_7\}$	0	0.32		
x_7	$\{a_1, a_7, a_8\}$	3	0.33	$\{a_5, a_6\}$	1
	$\{a_5, a_6\}$	1	0.54		
x_8	$\{a_1, a_4, a_8\}$	1	0.30	$\{a_5, a_6\}$	2
	$\{a_2, a_7\}$	0	0.32		
	$\{a_5, a_6\}$	2	0.54		
x_9	$\{a_3, a_8\}$	1	0.32	$\{a_4, a_5, a_6\}$	3
	$\{a_4, a_5, a_6\}$	3	0.44		

resulting classification as well as a novel input about the expertise level of the agents in each domain.

6.1. The outline of the extended algorithm

In the extended version of this algorithm, the set of instances is categorized with respect to the knowledge domain of the instances, while for each category of instances, the group of agents is clustered by using K-mode clustering. Then, Algorithm 1 is applied to each cluster, and the resulting classification is obtained by applying majority voting over the identified experts. The revised WICRO Algorithm 2 is outlined as follows.

In Algorithm 2, the formula for calculating the number of clusters K is heuristic and is obtained by numerical experiments with different datasets; in this formula, $[\cdot]$ denotes the round value of the number. This value of K results in small clusters of agents that allow the exclusion of the clusters with single agents and those with agents whose opinions strongly differ from the opinions of the other agents.

The call for Algorithm 1 assumes that the algorithm obtains all required data appearing in the input for Algorithm 2.

6.2. Running example

To illustrate the implementation of Algorithm 2, let us continue with the example considered in Section 5.2. The dataset for the example is presented in Table 3.

The instances are divided into the set η of $n' = 3$ domains of knowledge: sports $X_1 = \{x_1, x_2, x_3\}$, politics $X_2 = \{x_4, x_5, x_6\}$ and movies $X_3 = \{x_7, x_8, x_9\}$.

The number of clusters is defined by the parameter $K = \left[\frac{2m}{3}\right] = \left[2 \times \frac{8}{3}\right] = 5$.

Consider the sports domain X_1. For instances x_1, x_2 and x_3 from domain X_1 after applying the K-mode algorithm, the agent clusters for the sports domain X_1 are

$$\zeta_1 = \{\{a_1, a_2, a_7\}, \{a_3, a_5\}, \{a_4\}, \{a_6\}, \{a_8\}\}.$$

In fact, agents a_1 and a_2 equivalently classify instances x_1, x_2 and x_3 from the sports domain, and the classification provided by agent a_7 is the closest to the classifications provided by agents a_1 and a_2 (in classification a_7 instance x_2 in the same class C_1 as in classifications a_1 and a_2). The classifications provided by the agents a_3 and a_5 include the instances in the classes that differ from the classes in the first three classifications but equivalently include the instance x_2 into the class C_2.

Table 5
Scores of the clusters and chosen classes.

	Domain of knowledge	Cluster	Score	Choice
x_1	Sports	$\{a_1, a_2, a_7\}$	0.71	0
x_2		$\{a_3, a_5\}$	0.43	1
x_3				2
x_4	Politics	$\{a_2, a_8\}$	0.43	3
x_5		$\{a_3, a_4\}$	1.10	4
x_6		$\{a_6, a_7\}$	0.43	0
x_7	Movies	$\{a_1, a_3, a_8\}$	0.48	1
x_8		$\{a_5, a_6\}$	1.10	2
x_9				3

By the same reasoning, for the political domain X_2, the clusters of the agents are

$$\zeta_2 = \{\{a_2, a_8\}, \{a_3, a_4\}, \{a_6, a_7\}, \{a_1\}, \{a_5\}\},$$

and for the movie domain X_3, the clusters of the agents are

$$\zeta_3 = \{\{a_1, a_3, a_8\}, \{a_5, a_6\}, \{a_2\}, \{a_4\}, \{a_7\}\}.$$

After excluding the singleton clusters, one obtains the following subsets:

$$\zeta_1 = \{\{a_1, a_2, a_7\}, \{a_3, a_5\}\}, \quad \zeta_2 = \{\{a_2, a_8\}, \{a_3, a_4\}, \{a_6, a_7\}\}$$
$$\zeta_3 = \{\{a_1, a_3, a_8\}, \{a_5, a_6\}\},$$

and these subsets of agents are processed by WICRO Algorithm 1. The execution and output of the algorithm are summarized in Table 5.

Following the selections of the agents that are included in the clusters, the resulting classification is

$$\alpha = \{\{x_1\}, \{x_2, x_7\}, \{x_3, x_8\}, \{x_4, x_9\}, \{x_5\}\},$$

which is equivalent to the correct classification α^*, thus with 100% accuracy.

Algorithm 2 (WICRO) obtains the same data as Algorithm 1 with an additional partition of the set of instances into the domains of knowledge with a single additional heuristic parameter used in calculating the number of clusters.

7. Numerical simulations and verification of the algorithm

The proposed algorithms have been tested rigorously on synthetic as well as real-world datasets to show their validity. In the first approach, the proposed algorithms were tested on synthetic datasets that were generated on the basis of known datasets.

These artificial datasets were generated in two different manners. In the first approach, for standard datasets, such as Iris and Glass [23], the selection of the agents and their clusters were generated randomly. In the second approach, for a given set of $n = 100$ instances, the sets of agents and their opinions were generated randomly. In the case of synthetic data, we varied the number of options in the questionnaires.

Real data were obtained using two quizzes, i.e., a *country quiz* and an *academic quiz*. The details of real dataset preparations are discussed in Section 7.2.

Algorithm 2 (the use of the domains of knowledge)

```
Input:  set $X = \{x_1, x_2, \dots, x_n\}$ of instances,
        set $\eta = \{Q_1, Q_2, .., Q_n\}$ of questionnaires (with respect to
        instances),
        set $A = \{a_1, a_2, \dots, a_m\}$ of agents,
        number $l$ of classes,
        normalization factor $y$.
Output: classification $\alpha = \{C_1, C_2, \dots, C_l\}$.

Create partition $\eta = \{X_1, X_2, \dots, X_{n'}\}$ of $X$ to the domains of
knowledge.
For each subset $X_u \in \eta$,  $u = 1,2, \dots, n'$,  do
    Cluster the agents into $K = \left[\frac{2m}{3}\right]$ clusters $\xi$ by $k$ -mode
    clustering.
    Exclude the singleton clusters.
    Identify experts by Algorithm 1.
    For each instance $x_i \in X$,  $i = 1,2, \dots, n$,  do
        Label the instance $x_i$ by majority voting.
    End
End
Create classification $\alpha = \{C_1, C_2, \dots, C_l\}$ of the set $X = \{x_1, x_2, \dots, x_n\}$ of
instances with respect to the obtained labels.
```

Table 6
Classification results of artificial data with constant parameters.

Dataset	Number of agents	Number of options	Classification accuracy			
			Algorithm 1 $y = 0.25$	Algorithm 1 $y = 0.05$	Algorithm 2	Majority voting
Glass	17	6	0.645	0.678	**0.813**	0.664
Iris	21	6	0.847	0.873	**0.933**	0.800
Abalone	14	3	0.662	0.652	**0.708**	0.619
Students	21	4	0.683	0.662	**0.753**	0.650
Wine	19	6	0.746	0.732	**0.782**	0.689
Robots	17	5	0.760	0.781	**0.932**	0.768

7.1. Artificial datasets with constant parameters and fixed set of classes

In the first series of verifications, six standard datasets within the Kaggle collection are used: Abalone, Glass, Iris, Students, Wine, and Robots [23]. Table 6 shows the classification results using WICRO Algorithm 1, WICRO Algorithm 2, and majority voting, which also represent the DS- and FDS-based algorithms.

From the results presented in the table, one can see that WICRO Algorithm 1 provides more accurate classifications than the majority voting- and DS-based methods, and, as expected, the most accurate classifications are obtained by WICRO Algorithm 2.

In addition, the results enable a better choice of the parameter y. If the accuracy of majority voting is between 0.6 and 0.7, then it is better to use a lower value of y, while if the accuracy of majority voting is greater than 0.7, then it is better to use a higher value of y.

7.2. Artificial dataset with varying parameters and varying sets of classes

The second series of experiments was conducted over 12 synthetically generated datasets, each of which included 100 instances. The number of experts per domain of knowledge was specified to 2, and since an expert cannot be 100% accurate, the accuracy was specified to a reasonable value of approximately 90%.

In the datasets, the number of domains of knowledge varied between 1 and 4, the number of agents varied between 8 and 10, and the number of optional responses for each instance varied between 4 and 6. It should be noted that if the number of optional responses per instance is less than 4, then majority-voting techniques provide the best results. Table 7 compares the classification results using WICRO Algorithm 1, WICRO Algorithm 2, and majority voting that, as said, represent the DS-based algorithms. For illustration purposes, we also applied the recently proposed

Table 7
Classification results of artificial data with varying parameters.

Dataset	Number of agents	Number of options	Number of domains of knowledge	Classification accuracy			
				Alg 1 $y = 0.25$	Alg 1 $y = 0.05$	Alg 2	Majority voting/ DSC^2/FDS
S1	10	4	4	0.69	0.73	**0.89**	0.61
S2	8	4	4	0.70	0.65	**0.91**	0.66
S3	10	5	4	0.64	0.67	**0.89**	0.57
S4	8	5	4	0.67	0.73	**0.89**	0.61
S5	10	6	4	0.65	0.64	**0.92**	0.51
S6	8	6	4	0.72	0.72	**0.91**	0.61
S7	10	4	1	**0.90**	**0.9**	**0.9**	0.67
S8	8	4	1	**0.92**	0.90	**0.92**	0.77
S9	10	5	1	0.90	**0.91**	**0.91**	0.69
S10	8	5	1	0.90	**0.91**	**0.91**	0.69
S11	10	6	1	**0.90**	**0.90**	0.88	0.66
S12	8	6	1	**0.90**	**0.90**	**0.90**	0.74

DSC^2 algorithm and FDS on the synthetic datasets with different options for each question, and it acts as a majority approach, as shown in Table 7.

From these results, the proposed WICRO Algorithms 1 & 2 obtain more accurate classifications than the majority voting/DSC^2 algorithm/FDS algorithm.

For the datasets with 4 domains of knowledge (datasets S1–S6), Algorithm 2 utilizes information about the knowledge domains and outperforms WICRO Algorithm 1. For the datasets with a single domain of knowledge (S7–S12), the information about the domain of knowledge is meaningless, and both algorithms result in similar classifications with similar accuracy.

In the datasets that include 10 agents, an additional two agents (that is, an increase of 25%) are nonexperts; consequently, the classification accuracy in all methods decreases. However, while the accuracy of majority voting and WICRO Algorithm 1 decreases significantly, the decrease in the WICRO Algorithm 2 accuracy is minimal.

Finally, WICRO Algorithm 1 provides better results on the datasets with one domain of knowledge than on the datasets with four knowledge domains.

7.3. Verification on real-world datasets

To further test the suggested algorithms over real-world datasets, we created a proprietary dataset with known domains of knowledge and acceptable groups of expertized and nonexpert agents. In particular, we assumed that natural-born citizens of a given country are, on average, better acquainted with the *celebrities* and *landmarks* of their country than natural-born citizens of other countries. Thus, we defined two knowledge domains, *celebrities* and *landmarks,* and selected pictures of the two in six different countries. Therefore, six options were provided for each question. Examples of the images used in the tests are presented in Fig. 1.

Amazon Mechanical Turk was then used to collect data via an online quiz that contained 398 questions and was distributed over the 28 respondents in six different countries: Brazil, France, India, Israel, Italy, and the USA [24]. The dataset and additional details on this extensive study are given in Appendix.

First, we checked the assumption that natural-born citizens of the countries are indeed better acquainted with the celebrities and the landmark images of their respective countries. The response results are summarized in Table 8.

On average, the respondents provided more accurate results in questions related to their native countries and responded less accurately to the questions about foreign countries.

Table 8
The average accuracy of the responses from native and foreign countries.

	Brazil	France	India	Israel	Italy	USA
Accuracy of responses to the questions about native countries	0.74	0.76	0.8	0.8	0.74	0.69
Accuracy of questions to the questions about foreign countries	0.56	0.48	0.31	0.43	0.43	0.47

Table 9
Accuracy of the algorithms applied to real-world data.

Algorithm 1 $y = 0.25$	Algorithm 1 $y = 0.05$	Algorithm 2	Majority voting
0.804	0.802	0.867	0.781

Finally, Table 9 presents the accuracies of the algorithms applied to this collected real-world dataset.

As expected, the proposed WICRO algorithms outperform the majority-voting method, and the best results were provided by WICRO Algorithm 2, which results in nearly 9% higher accuracy in classification than majority voting and nearly 7% higher accuracy than WICRO Algorithm 1.

Academic dataset

The proposed WICRO algorithm was also tested on another real-world dataset from an academic setting. In particular, we created an academic quiz containing 30 questions, including 10 questions from each of four engineering domains, namely, computer engineering, electrical engineering, civil engineering, and biomedical engineering. The quiz contained multiple-choice questions, and each question had six options, such that they were different for each question. The data were collected from 26 agents (students) in a dataset containing their responses. We have identified the experts for each question by applying the proposed WICRO algorithm. The experimental results presented in Table 10 further support our hypothesis. As expected, students (agents) in any engineering domain performed better on questions related to their domain than on questions based on other domains. Each agent from a specific engineering domain has similar responses to the questions related to their domain. Additionally, the agents' answers diverged more in other domains. Thus, the underlying hypothesis of this study – that experts will agree more on topics in their field but may differ in their opinions regarding topics in other fields – was supported. Thus, the results of the experiment presented in Table 10 justify this hypothesis.

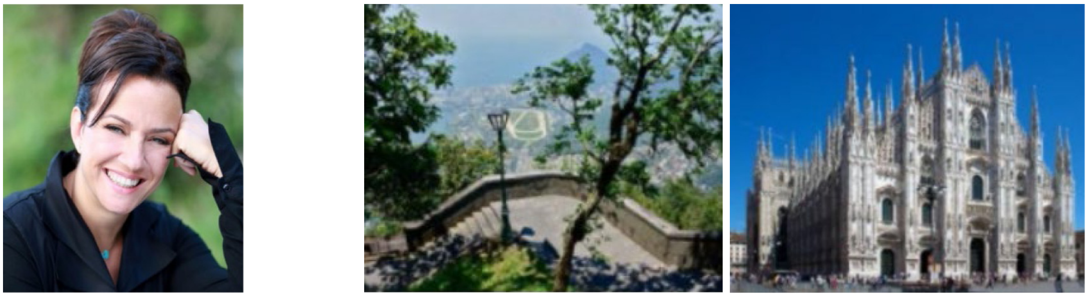

Fig. 1. Examples of the images used in the classification tests: (a) Celebrity – Tzufit Grant of Israel, and (b) Landmarks – Rio de Janeiro, Brazil (left) and Milano, Italy (right). Each participant was asked to classify each image into one out of six countries.

Table 10
Accuracy of the algorithms on the academic dataset.

Algorithm 1 $y = 0.25$	Algorithm 1 $y = 0.05$	Algorithm 2	Majority voting/ DSC^2/FDS
0.666	0.666	0.7	0.566

We also tested the FDS/DSC^2 algorithms on this academic dataset and obtain a response that is similar to the majority-voting approach. Thus, one can conclude from the results that the performance of the proposed WICRO algorithm is better than that of the majority approach, DSC^2 algorithm and FDS algorithm. Moreover, the proposed approach was used to identify the experts and nonexperts in each domain. Since domain knowledge is available in this dataset, this knowledge can be used to improve the accuracy of identifying experts in each question and domain by applying the WICRO Algorithm 2, which was mentioned in Section 6. As shown in Table 10, this algorithm has further improved the accuracy of expert identification to 0.7.

8. Conclusion and future scope

This paper presents a novel '*wisdom-in-the-crowd*' (WICRO) algorithm that, unlike '*wisdom-of-the-crowd*', does not assume that the answers of agents in the crowd are symmetrically distributed around the right unknown answer. Thus, WICRO does not assume that the majority of the agents' answers will converge to the correct answer. Instead, WICRO aims to identify those agents that are experts in their domain and then use such information to identify the correct answer. The algorithm is used for unsupervised (one-pass) classification and can also be used in datasets with different classes per instance over various domains. The proposed algorithm can use domain knowledge to further improve the accuracy of identifying experts and the correct answers. The WICRO algorithm is based on aggregating the agents' opinions and on agents divided into groups of experts and nonexperts.

In contrast to the existing methods, the proposed algorithms can be applied to datasets having different classes per instance in various domains, even if these domains are not identified ahead. The WICRO algorithm's performance was tested and compared to the popular majority voting and other well-known approaches suggested by the DSC^2 and FDS algorithms. It has been noted that the DSC^2 and FDS algorithms reduce to the majority-voting approach for datasets with different classes per instance, as they are designed for datasets with a fixed set of classes. It was shown that the proposed algorithms outperform these existing approaches and result in more accurate classifications in both synthetic and real-world dataset examples. In the proposed WICRO algorithm, it is assumed that the group of agents contains experts in specific domains that are not necessarily experts in all domains of interest. This is a reasonable assumption supported by many studies over various domains of expertise.

The proposed approach can be used for many real-life applications, such as health care, natural language understanding, image tagging, taxonomy creation, and learning management systems. In particular, the WICRO approach can be implemented in questionnaires and other ad hoc knowledge-seeking processes on the internet when the agents' domain and level of expertise are unknown.

In the future, the same principles can be applied to find the best artificial AI agents in various domains and classification problems. Furthermore, explainable AI (XAI) methods can use the WICRO output to attempt to explain why some of the AI agents perform better than others, as well as identifying relevant features that can be used when classifying specific instances or groups of instances.

CRediT authorship contribution statement

Nir Ratner: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Software, Visualization, Validation, Writing – original draft. **Eugene Kagan:** Conceptualization, Formal analysis, Methodology, Validation, Resources, Supervision, Writing – review & editing. **Parteek Kumar:** Conceptualization, Formal analysis, Methodology, Validation, Resources, Supervision, Writing – review & editing. **Irad Ben-Gal:** Conceptualization, Funding acquisition, Project administration, Supervision, Writing – review & editing.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Irad Ben-Gal reports financial support was provided by The Koret Foundation gift for Smart Cities and Digital Living 2030.

Data availability

Data will be made available on request.

Appendix. Countries quiz

Results: https://bit.ly/3CpZuUf
Countries quiz questionnaire
Part 1: PART 1 (gclocked.com)
Part 2: PART 2 (gclocked.com)
Part 3: PART 3 (gclocked.com)
Part 4: PART 4 (gclocked.com)

References

[1] O. Reyes, C. Morell, S. Ventura, Effective active learning strategy for multi-label learning, Neurocomputing 273 (2018) 494–508.

[2] V.B. Sinha, S. Rao, V.N. Balasubramanian, Fast Dawid-Skene: A fast vote aggregation scheme for sentiment classification, in: Proc. KDD Workshop on Issues of Sentiment Discovery and Opinion Mining, KDD WISDOM, 2018.

[3] C. Li, V.S. Sheng, L. Jiang, H. Li, Noise filtering to improve data and model quality for crowdsourcing, Knowl.-Based Syst. 107 (2016) 96–103.

[4] X. Qian, Y. Yan Tang, Z. Yan, K. Hang, ISABoost: A weak classifier inner structure adjusting based AdaBoost algorithm—ISABoost based application in scene categorization, Neurocomputing (Amsterdam) 103 (2013) 104–113.

[5] B. Latane, S. Wolf, The social impact of majorities and minorities, Psychol. Rev. 88 (1981) 438–453.

[6] R. Morton, M. Piovesan, J.-R. Tyran, The dark side of the vote: Biased voters, social information, and information aggregation through majority voting, Games Econ. Behav. 113 (2019) 461–481.

[7] A. Koriat, When two heads are better than one and when they can be worse: The amplification hypothesis, J. Exp. Psychol. 144 (5) (2015) 934–950.

[8] S. Chatterjee, A. Mukhopadhyay, M. Bhattacharyya, A weighted rank aggregation approach towards crowd opinion analysis, Knowl.-Based Syst. 149 (2018) 47–60.

[9] J. Whitehill, P. Ruvolo, T. Wu, J. Bergsma, J. Movellan, Whose vote should count more: Optimal integration of labels from labelers of unknown expertise, Adv. Neural Inf. Process. Syst. 22 (2009) 2035–2043.

[10] A. Ghanaiem, I. Ben-Gal, T. Raviv, E. Kagan, The Dawid-Skene Type Algorithm for Unsupervised Collaborative Classification under Uncertainty, Tel-Aviv University, LAMBDA, 2021, Unpublished results.

[11] A.P. Dawid, A.M. Skene, Maximum likelihood estimation of observer error-rates using the EM algorithm, J. Royal Stat. Soc. Ser. C 28 (1) (1979) 20–28.

[12] L. Duan, S. Oyama, H. Sato, M. Kurihara, Separate or joint? Estimation of multiple labels from crowd-sourced annotations, Expert Syst. Appl. 41 (13) (2014) 5723–5732.

[13] A. Montejo-Raez, M.C. Díaz-Galiano, F. Martinez-Santiago, L.A. Ureña-López, Crowd explicit sentiment analysis, Knowl.-Based Syst. 69 (2014) 134–139.

[14] J. Zhang, V.S. Sheng, J. Wu, X. Wu, Multi-class ground truth inference in crowd-sourcing with clustering, IEEE Trans. Knowl. Data Eng. 28 (4) (2015) 1080–1085.

[15] C. Li, V.S. Sheng, L. Jiang, H. Li, Noise filtering to improve data and model quality for crowd-sourcing, Knowl.-Based Syst. 107 (2016) 96–103.

[16] D. Prelec, H.S. Seung, J. McCoy, A solution to the single-question crowd wisdom problem, Nature 541 (7638) (2017) 532–535.

[17] G. Hagerer, D. Szabo, A. Koch, M.L.R. Dominguez, C. Widmer, M. Wich, et al., End-to-end annotator bias approximation on crowdsourced single-label sentiment analysis, 2021, arXiv preprint arXiv:2111.02326.

[18] N.B. Shah, S. Balakrishnan, M.J. Wainwright, A permutation-based model for crowd labeling: Optimal estimation and robustness, 2021, arXiv:1606.09632v3. (Accessed 27 September 2021).

[19] V.B. Sinha, S. Rao, V. N. Balasubramanian, Fast Dawid-Skene: A fast vote aggregation scheme for sentiment classification, in: Proc. KDD Workshop on Issues of Sentiment Discovery and Opinion Mining, KDD WISDOM, 2018.

[20] S. Ibrahim, X. Fu, N. Kargas, K. Huang, Crowd-sourcing via pairwise co-occurrences: Identifiability and algorithms, Adv. Neural Inf. Process. Syst. 32 (2019).

[21] C. Schmidt, L. Zdeborová, Dense limit of the Dawid–Skene model for crowd-sourcing and regions of sub-optimality of message passing algorithms, J. Phys. A 53 (12) (2020) 124001.

[22] M. Eshkevari, M.J. Rezaee, M. Saberi, O.K. Hussain, An end-to-end ranking system based on customers reviews: Integrating semantic mining and MCDM techniques, Expert Syst. Appl. 209 (2022) 118294.

[23] Kaggle datasets, 2019, www.kaggle.com, Kaggle Inc. (Accessed 27 September 2021).

[24] Amazon Mechanical Turk www.mturk.com. (Accessed 27 September 2021).