Research Article

MCDM Approach for Assigning Task to the Workers by Selected Features Based on Multiple Criteria in Crowdsourcing

Zhao Huiqi ¹, Abdullah Khan,² Xu Qiang,³ Shah Nazir ², Yasir Ali,² and Farhad Ali ²

¹College of Intelligent Equipment, Shandong University of Science and Technology, Qingdao, Shandong, China

²Department of Computer Science, University of Swabi, Swabi, Pakistan

³College of Computer Science and Engineering, Shandong University of Science and Technology, Qingdao, China

Correspondence should be addressed to Zhao Huiqi; zhqskd@163.com and Shah Nazir; snshahnzr@gmail.com

Received 8 May 2021; Accepted 9 June 2021; Published 18 June 2021

Academic Editor: Zhu Xiao

Copyright © 2021 Zhao Huiqi et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Crowdsourcing in simple words is the outsourcing of a task to an online market to be performed by a diverse group of crowds in order to utilize human intelligence. Due to online labor markets and performing parallel tasks, the crowdsourcing activity is time- and cost-efficient. During crowdsourcing activity, selecting the proper labeled tasks and assigning them to an appropriate worker are a challenge for everyone. A mechanism has been proposed in the current study for assigning the task to the workers. The proposed mechanism is a multicriteria-based task assignment (MBTA) mechanism for assigning the task to the most suitable worker. This mechanism uses approaches for weighting the criteria and ranking the workers. These MCDM methods are Criteria Importance Through InterCriteria Correlation (CRITIC) and Technique for Order Preference by Similarity to Ideal Solution (TOPSIS). Criteria have been made for the workers based on the identified features in the literature. Weight has been assigned to these selected features/criteria with the help of the CRITIC method. The TOPSIS method has been used for the evaluation of workers, with the help of which the ranking of workers is performed in order to get the most suitable worker for the selected tasks to be performed. The proposed work is novel in several ways; for example, the existing methods are mostly based on single criterion or some specific criteria, while this work is based on multiple criteria including all the important features. Furthermore, it is also identified from the literature that none of the authors used MCDM methods for task assignment in crowdsourcing before this research.

1. Introduction

The term crowdsourcing refers to the outsourcing of different tasks to a huge amount of people known as crowd in order to utilize collective human intelligence. Crowdsourcing was first defined by J. Howe as the outsourcing of tasks or work to a network of undefined people by means of an open call format. The term crowdsourcing represents an act of an institution by taking a function performed by crowd and then outsourcing it to a network of undefined group of individuals [1]. The process of crowdsourcing contains three main actors, the requester or client who requests the work or task to be performed, the crowd who performs the requested task, and the platforms who act as a broker between the clients and crowd [2]. Crowdsourcing

platforms contain paid and unpaid platforms, where in paid platforms crowd performs the tasks as a result of monetary rewards, while in unpaid platforms the volunteer crowd performs the tasks [3]. The use of crowdsourcing is increasing day after day as it is time- and cost-efficient for software development and many other tasks. It has been applied in several domains such as design-based apps, text translation to different languages, and labelling datasets [2].

The term CSE (crowdsourced software engineering) is derived from crowdsourcing. By means of an open call, it recruits online workers globally to perform several software engineering tasks such as requirement elicitation, coding, designing, and testing. It reduces the time to market due to parallelism. CSE rapidly gained interest in both industry and academia [4]. Day after day, crowdsourcing is gaining the

attention of the communities. While crowdsourcing the task assignment is a challenging phase for everyone, selecting an appropriate labeled task from the client or the requestor and assigning it to an appropriate worker are a challenging issue in crowdsourcing. During crowdsourcing process, some of the crowds select irrelevant tasks in order to get the rewards but they do not have the potential to perform the task. As a result, the crowd then submit the low-quality result and thus decrease the client trust, which directly affects the crowdsourcing process. So it is shown that the task assignment is an important step to be taken during the crowdsourcing activity [5]. That is why a solution is required in order to fix the task assignment problem. The main contributions of the proposed study are as follows:

- (i) A mechanism has been presented for addressing the issue regarding the assignment of different tasks during crowdsourcing activity
- (ii) The proposed work identified different features of workers and selected the important ones for making the criteria for task assignment
- (iii) The existing task assignment methods are mostly based on single criterion, while this work is based on multiple criteria
- (iv) Two MCDM methods, CRITIC and TOPSIS, have been used for giving weight to the selected features and evaluation of the workers to rank them for assigning tasks to the appropriate workers

The remainder of the article is composed of four main sections. Section 2 describes the existing/related work, Section 3 describes research method, and the conclusion is given in Section 4.

2. Related Work

In the existing literature, the task assignment models and methods are recommended with the help of different techniques. The competitor's history of participation such as participation frequency and recency as well as winning frequency and recency along with tenure and last performance is derived in order to construct a model [6]. Literature work also proposed the framework for task recommendation for the task-based preference modelling as well as preference-based recommendation of task, the goal of which is to recommend task to crowd [7]. The recommendation algorithm for the personalized task is also advocated by the authors. The authors also suggest approaching the issue regarding the design of the mechanism of the personalized task recommendations. The three task selection methods, heuristic-based method, bound-based method, and active learning method, have also been identified by the literature [8]. The SmartCrowd framework is also proposed by the authors, which focuses on the optimization of crowd to the task assignment in a knowledge-intensive crowdsourcing environment. It focuses on the production of knowledge instead of simple tasks. It also integrates multiple factors of humans such as crowd expertise, the required wage, and the

ratio of acceptance in the assignment process [9]. The task offloading was considered as one of the significant areas of research [10].

A bandit formulation for the assignment of tasks in heterogeneous crowdsourcing is also proposed, which is known as bandit-based task assignment. The worker is represented by the arm of the bandit. It mainly focuses on the selection strategy of workers in the heterogeneous crowdsourcing. The goal is to select the worker that is suitable and good for the task [11]. A dynamic solution is also presented for the assignment problem of tasks in the crowdsourcing platforms. The crowd select approach offers an algorithm for assigning workers to the tasks in a cost-efficient way and also ensures the accuracy of the tasks. The two main components of the approach are worker selection and their error rate prediction [12]. A learning algorithm has been described by the authors, which groups the task from the history into cluster and then derives from every cluster the features of the worker which optimize the quality of the contribution. Then these features are used by the algorithm to select the appropriate worker for the task [13]. To measure the effect of personality on the selection of tasks, the experiment was conducted based on the task characteristics such as type, money, and time. The type of personality of the workers was measured based on the Myers-Briggs Type Indicator. Sixteen different personality types are categorized. Then an experiment was performed (four rounds) to determine which worker belongs to which category. The round 1 and round 3 personality types are interested in prize and money instead of complexity. The other two rounds of personality types want the prize and money rather than deadlines [14]. The Skills Improvement Aware Framework is also proposed by the authors to recommend crowd for task in the development process of software and crowdsourcing. A study of the developer performance on the TopCoder platform was also conducted [15].

The assignment problem of tasks is also explored in a budget constraint with different variety of skill levels and different required quality. An algorithm was also designed for the generation of outcomes for many-to-one matching problem with upper bound and lower bound and the skill level of the worker [16]. The sensitive task assignment is proposed by the authors. First of all, sensitive task is partitioned and then assigned to workers. For the avoidance of colluding participants with the help of which they can exchange the data, a three-step task assignment method is proposed, which is known as sensitive task assignment. The steps include collusion estimation, worker selection, and task partitioning [17]. A data-driven learning approach was also proposed by the authors. The supervised learning and reinforcement learning have been combined in the approach to enable the agents to imitate the task allocation strategies, which show good performance [18]. TOP-K-T and TOP-K-W are the two real-time recommendation algorithms proposed by Safran and Che. The first one computes the appropriate task for a worker and the second one computes the appropriate worker for the task [19]. The expertise prediction heuristics have also been proposed by the authors to identify the experts automatically and to filter a nonexpert

during crowdsourcing activity. Based upon the four expertise prediction heuristics, an experiment was performed. These include demographics of evaluator, reaction time of evaluation, and mechanical reasoning aptitude, and the last one is easy version of the evaluation tasks [20]. The Learning Automata Based Task Assignment (LEATask) is also proposed by the authors, which works upon the worker similarity in their performances. The algorithm has exploration stage and exploitation stage [21]. Different areas can be considered in the task assignment allocation such as energy cooperation and joint information for heterogeneous networks [22].

The batch allocation technique is also proposed for the crowdsourcing tasks with the overlapping skill requirements. The designed heuristics approaches include core-based batch allocation and layered batch allocation. The experiment is made on the upwork dataset [23]. Two online task assignment mechanisms are also developed by the authors, which will dynamically assign set of tasks to the incoming crowd (worker) with the help of which the worker will gain the maximum expected gain and maximum expected and potential gain. The task is divided into clusters by the authors and then they proposed a Latent Topic Model to describe the structure of topic and expertise of the workers [24]. Based on the deep learning, the existing literature also proposed Tag-Semantic Task Recommendation model. The similarity of the word vectors is computed and then the Semantic Tag Matrix Database has been established based upon the Word2vec deep learning. Then the recommendation model for task is established based upon the Semantic Tags to achieve the recommendation of the tasks in crowdsourcing. The task and worker relevancy is obtained by computing the tags similarity [25]. The Dynamic Utility Task Allocation (DUTA) algorithm was proposed in the paper. For the estimation of the worker's initial value, the attributes are used, which are given by the worker during the time of registration. The developmental capabilities of workers are also calculated by the history of completed task, complexity of tasks, and quality and efficiency of the result. Then the matching degree is calculated based upon the weight of crowd skills and posted task requirement [26].

The active time of the worker is also used to get a solution for Multitask Multiworker Allocation. The three factors that the authors consider are the ability of the worker, active time of the worker, and the complexity of task module. The individuals are divided into collaborative groups, and then, for the optimal selection of worker to perform a task, the Hungarian algorithm is used [27]. The automatic detection for improper task is also proposed in the process of crowdsourcing. With the help of analyzing the estimated classifier, a variety of effective measures for the detection of improper task are observed, including words that appeared in task information, reward or money which will be received by the workers after performing the task, and their qualification for performing the task [5]. Various research fields can benefit from using the proposed approach such as IoT underlying heterogeneity [28], investigation of data aggregation in mobile sensor networks for IIoT [29], sharing of resource in heterogeneous vehicular network [30], and many other fields.

3. Methodology

As the assignment of task during crowdsourcing activity is a challenge for everyone, in the current study, a mechanism is proposed for task assignment method, which is based on multiple criteria. The study proposed a multicriteria-based task assignment (MBTA) mechanism. Two methods have been used in the current study. The CRITIC method has been used for assigning weights to the selected features and the TOPSIS method is used for ranking the workers. The work done in these methods can also be performed manually, but selecting these methods for doing the proposed work can give authenticity and appropriateness to the research work. Performing work manually contains several chances of mistakes, but doing it with predefined and already experimented methods increases the quality of the work; therefore, these two methods have been selected for the proposed study. The details of the study are discussed in the following sections.

3.1. Criteria for Task Assignment. To define the criteria for task assignment, first of all a variety of features have been identified from the existing literature. 33 of the most famous and important features have been identified during literature study. These features are then analyzed and the most important ones are selected for the development of mechanism for task assignment. The weight has then been assigned to these identified features by the CRIRIC method, which is discussed in the next section. The list of identified features is shown in Table 1.

Table 1 shows all the identified features analyzed during systematic literature review. The important features are then selected from the list in order to develop a mechanism.

3.2. Case Study. To complete data collection for making criteria, a case study was performed. In this case study, the issues regarding the assignment of task have been highlighted. All the gaps have been discussed briefly. A comprehensive observation was carried out in order to select the features for criteria from the identified features during literature. The experts were asked different questions in order to select the most important features. These features were then scaled, ranging from 1 to 10, with the help of experts. A group of experts scaled these features so that important features get more weight among the other features, thus making it easy to rank the crowd who have good qualities at the top. After that, these features have been used for analyzing and making criteria as well as for evaluation of the workers to rank them for assigning a task. As all the features have been identified from the existing literature, for further analysis, experts were asked some questions based on these selected features in order to analyze the importance of these features. The questions the experts were asked are shown in Table 2.

A list of the selected features is shown in Table 3.

3.3. Weight of Selected Features. The features have been analyzed by experts in the relevant field. Scaling was given to each criterion/feature, ranging from 1 to 10, by these experts

TABLE 1: Identified features.

S. no.	Features	Citations
1	Profile management	[31]
2	Flexibility	[32]
3	Worker history	[7]
4	Worker performance history	[7]
5	Worker task searching history	[7]
6	Task completion ratio	[7]
7	Period of time for task	[7]
8	Participation frequency	[6]
9	Participation recency	[6]
10	Winning frequency	[6]
11	Winning recency	[6]
12	Tenure	[6]
13	Reliability	[33]
14	Worker qualification	[5, 34]
15	Quality of task	[5]
16	Knowledge	[35]
17	Skills/expertise	[9, 35]
18	Cheap/cost-effective/cost-efficient	[1, 9, 12]
19	Software worker behavior	[36]
20	Task similarity	[37]
21	Delivery time	[38]
22	Task acceptance ratio	[9]
23	Accuracy ratio	[12]
24	Response ratio/quality of response	[12]
25	Quality of task	[12]
26	Trustworthy/honesty	[12]
27	Relevant experience	[39]
28	Interest	[40]
29	Reaction time	[20]
30	Personality type	[14]
31	Skill level	[16]
32	Active time	[27]
33	Development efficiency	[26]

in order to get the most important criteria. Weights have been assigned to all these selected features with the help of the CRITIC method. The final weights have been obtained by applying equations (1)–(4), respectively. The final weights have been shown in Table 4 and Figure 1. Table 5 describes the scales of the selected features.

The weights for each criterion are shown in Figure 2.

3.4. Proposed Mechanism for Task Assignment. The MBTA mechanism has been proposed, which is based on multiple criteria. This mechanism has been developed based upon two methods. The CRITIC method is used to assign weights

TABLE 2: Questionnaire.

Q1	What is the importance of worker history while assigning a task?
Q2	What is the importance of trustworthiness in task assignment?
Q3	How much worker qualification matters during task assignment?
Q4	Is the reliability of the worker important for assigning a task?
Q5	What is the role of response ratio in assigning a task?
Q6	Does skill level matter for task assignment?
Q7	Is the quality of task important for clients?
Q8	What is the importance of delivery time in crowdsourcing?
Q9	What is the role of cost in assigning a task?

TABLE 3: Selected features.

S. no.	Features	Citations
1	Worker history	[7]
2	Trustworthiness/honesty	[12]
3	Worker qualification	[5, 34]
4	Reliability	[33]
5	Response ratio/quality of response	[12]
6	Skill level	[16]
7	Quality of task	[5]
8	Delivery time	[38]
9	Cheap/cost-effective/cost-efficient	[1, 9, 12]

to the selected features, and then the TOPSIS method is used for ranking the workers. The details are discussed in the following sections.

3.4.1. CRITIC Approach for Allocating Weights to Features. CRITIC is a type of correlation method which was first introduced in 1995. It is a multicriteria decision-making approach that is used for assigning weights to features or criteria during research work. During this method, the weight is assigned to the criteria objectively rather than by pairwise comparison or decision-makers judgments [41].

“ m ” is the number of possible alternatives such as A_i , when $i = 1, 2, 3, \dots, m$, and “ n ” is the number of evaluation criteria such as C_j for $j = 1, 2, 3, \dots, n$, in a problem. The following steps are followed in the approach.

Step-1. Building a Decision Matrix

A decision matrix “ X ” is created in the first step:

$$X = [X_{ij}] = \begin{bmatrix} X_{11} & X_{12} & \dots & X_{1n} \\ X_{21} & X_{22} & \dots & X_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ X_{m1} & X_{m2} & \dots & X_{mn} \end{bmatrix} \quad (\text{for } i = 1, 2, 3, \dots, m \text{ and } j = 1, 2, 3, \dots, n). \quad (1)$$

TABLE 4: Measuring standard deviation and its correlation with other criteria for criteria weights.

Sum	Standard deviation	C_j	W_j
8.888	0.332	2.953	0.114
8.879	0.313	2.778	0.107
8.286	0.389	3.226	0.125
8.690	0.314	2.731	0.105
8.768	0.359	3.149	0.122
9.070	0.281	2.549	0.098
8.830	0.359	3.172	0.122
7.810	0.334	2.611	0.101
8.087	0.337	2.728	0.105

FIGURE 1: Criteria weights.

TABLE 5: Scales of the selected features.

Features	Worker history	Trustworthiness/honesty	Qualification	Reliability	Response ratio/quality of response	Skill level	Quality of task	Delivery time	Cheap/cost-effective/cost-efficient
Worker 1	6	7	5	8	4	3	9	7	8
Worker 2	3	4	8	6	5	7	6	9	7
Worker 3	9	7	8	5	7	7	6	4	5
Worker 4	5	7	4	8	9	6	5	8	7
Worker 5	7	5	3	8	9	6	4	2	6
Worker 6	4	8	7	5	3	6	9	1	5
Worker 7	8	5	7	3	2	8	6	2	8
Worker 8	2	7	7	6	8	9	3	4	5
Worker 9	7	6	3	1	8	8	9	5	4
Worker 10	1	7	5	9	6	8	2	4	3

FIGURE 2: Steps of the CRITIC method.

In equation (1), X_{ij} shows the performance value of the i th alternative on the j th criterion.

Step-2. Decision Matrix Normalization

The process of normalization is done through the following equation:

$$X_{ij}^* = \frac{X_{ij} - \min(X_{ij})}{\max(X_{ij}) - \min(X_{ij})}, \quad i = 1, 2, \dots, m \text{ and } j = 1, 2, \dots, n. \quad (2)$$

X_{ij}^* is the normalized performance value of the i th alternative on the j th criterion.

Step-3. Calculating Standard Deviation and Its Correlation

In the third step, the weights of the j th criterion can be found with the following equation:

$$W_j = \frac{C_j}{\sum_{j=1}^n C_j}, \quad (3)$$

In equation (3), C_j is the amount of information contained in the j th criterion. C_j is calculated as follows:

$$C_j = \sigma_j \sum_{j'=1}^n (1 - r_{jj'}), \quad (4)$$

where σ_j is the standard deviation of the j th criterion and $r_{jj'}$ is the correlation coefficient between the two criteria [41].

3.4.2. Numerical Work of the CRITIC Method. Weights are assigned to the criteria using the CRITIC method. The determination of this study was to find the top worker based upon the features for the offered task. The workers that will perform the tasks have been used as alternatives such as A1, A2, A3, A4, A5, A6, A7, A8, A9, and A10; and the features have been used as criteria such as worker history (C1), trustworthiness/honesty (C2), worker qualification (C3), reliability (C4), response ratio/quality

of response (C5), skill level (C6), quality of task (C7), delivery time (C8), and cheap/cost-effective/cost-efficient (C9). Decision matrix has been established for these 10 workers (alternatives) with respect to defined features/criteria as shown in Table 6. The results given in Table 6 are obtained by normalizing the decision matrix while applying equation (2). Figure 2 shows the steps followed in this method.

Now calculations of the CRITIC method are followed step by step. Table 5 describes the CRITIC method decision matrix.

Table 7 shows the CRITIC method normalized decision matrix.

Measure of conflict has been calculated as shown in Table 8.

Standard deviation and its correlation with other criteria have been calculated for criteria weights as shown in Table 9.

For each worker, all the 9 features/criteria have been scaled, ranging from 1 to 10, as shown in Table 4.

3.4.3. TOPSIS Approach for Ranking of Workers. The TOPSIS approach deals with achieving ideal solutions. This approach has adopted simple computation procedures and thus it is reliable and well established. The selected alternatives in the TOPSIS method should have a minimum distance from positive ideal solution and maximum distance from negative ideal solution [41]. In this work, we will apply the TOPSIS method for ranking the alternatives. In this section, first of all, the TOPSIS method along with its steps and procedure to be followed will be discussed and then how this method has been used in this research will be discussed. The following are the steps used in the TOPSIS method in order to select and rank the best workers among different alternatives:

Step-1. Determining Weight and Building Decision Matrix

Decision matrix D is constructed in the first step by using multiple criteria and alternatives. For example, for “ n ” number of alternatives and criteria, the decision matrix can be found as follows:

$$D = \begin{matrix} & \begin{matrix} C_1 & \dots & C_n \end{matrix} \\ \begin{matrix} A_1 \\ \vdots \\ A_n \end{matrix} & \begin{bmatrix} X_{11} & \dots & X_{1n} \\ \vdots & \ddots & \vdots \\ X_{n1} & \dots & X_{nn} \end{bmatrix} \end{matrix}, \quad (5)$$

where $A_1, A_2, A_3, \dots, A_n$ are variable alternatives and $C_1, C_2, C_3, \dots, C_n$ are the criteria.

Step-2. Normalized Decision Matrix

As the input data of the decision matrix is originated from several different sources, it has to be converted into a dimensionless matrix by normalization.

The comparison between different criteria is done via this dimension matrix. By using formula (6), a normalized decision matrix has been built.

TABLE 6: The CRITIC method decision matrix.

Decision matrix									
	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8	C_9
A_1	6	7	5	8	4	3	9	7	8
A_2	3	4	8	6	5	7	6	9	7
A_3	9	7	8	5	7	7	6	4	5
A_4	5	7	4	8	9	6	5	8	7
A_5	7	5	3	8	9	6	4	2	6
A_6	4	8	7	5	3	6	9	1	5
A_7	8	5	7	3	2	8	6	2	8
A_8	2	7	7	6	8	9	3	4	5
A_9	7	6	3	1	8	8	9	5	4
A_{10}	1	7	5	9	6	8	2	4	3
Best	9	8	8	9	9	9	9	9	8
Worst	1	4	3	1	2	3	2	1	3

$$R_{ij} = \frac{X_{ij}}{\sqrt{\sum_{i=1}^m x_{ij}^2}}, \quad (6)$$

where $i = 1, \dots, m$ and $j = 1, \dots, n$.

Step-3. Weighted Normalized Decision Matrix

As it is not necessary that the importances of all attributes will be the same, by multiplying the elements of the normalized decision matrix with random weight number, a weighted normalized decision matrix can be obtained. The weight number for multiplication is given in the following formula:

$$V = V_{ij} = W_j \times R_{ij},$$

$$V = \begin{bmatrix} V_{11} & V_{12} & V_{1j} & V_{1n} \\ \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ V_{i1} & V_{i2} & V_{ij} & V_{in} \\ \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ V_{m1} & V_{m2} & V_{mi} & V_{mn} \end{bmatrix} = \begin{bmatrix} w_1 r_{11} & w_1 r_{12} & w_1 r_{1j} & w_1 r_{1n} \\ \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ w_i r_{i1} & w_i r_{i2} & w_i r_{ij} & w_i r_{in} \\ \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ w_m r_{m1} & w_m r_{m2} & w_m r_{mi} & w_m r_{mn} \end{bmatrix}. \quad (7)$$

Step-4. Finding Ideal Positive and Negative Solutions

In this step, A^+ denotes positive ideal solution and A^- denotes negative ideal solution. These are demonstrated through the weighted decision matrix.

$$A^+ = \{V_1^+, V_2^+, V_3^+, V_n^+\}, \quad \text{Where } V_j^+ = \{(\max_i (V_{ij}) \text{ if } j \in J); (\min_i V_{ij} \text{ if } j \in J')\}, \quad (8)$$

$$A^- = \{V_1^-, V_2^-, V_3^-, V_n^-\}, \quad \text{Where } V_j^- = \{(\min_i (V_{ij}) \text{ if } j \in J); (\max_i V_{ij} \text{ if } j \in J')\}, \quad (9)$$

where J denotes the beneficial attributes and J' denotes the nonbeneficial attributes.

Step-5. Separation Measures

By the following formulas, ideal and nonideal separations are calculated.

$$S^+ = \sqrt{\sum_{j=1}^n (V_{ij} - V_j^+)^2}, \quad \text{for } i = 1, \dots, m, \quad (10)$$

$$S^- = \sqrt{\sum_{j=1}^n (V_{ij} - V_j^-)^2}, \quad \text{for } i = 1, \dots, m. \quad (11)$$

Step-6. Finding Relative Closeness

It is determined with respect to the ideal solutions by using the following equation:

$$C_i = \frac{S_i^-}{(S_i^+ + S_i^-)}, \quad 0 \leq C_i \leq 1. \quad (12)$$

Step-7. Alternatives Ranking

By using C_i value, the process of ranking is prepared; high C_i value shows top rank order of the alternative, which can be labeled as superior in terms of efficiency.

The descending order can be adopted for the comparison of improved performance [41].

3.4.4. Numerical Work of the TOPSIS Method. In this section, evaluation of the workers and their ranking will be obtained based upon 9 identified features by using the TOPSIS method. The data has been collected from different questionnaires answered by several experts in relevant fields. The decision matrix is constructed by the data obtained from the panel of experts.

All the work is done step by step as shown in Figure 3.

By using equation (6), normalized decision is obtained. The results are listed in Table 10 along with criteria weights.

Decision matrix based on weighted normalization is obtained by using equation (7). Ideal positive and ideal negative solutions are calculated by using equations (8) and (9) and their values are given in Table 11.

For alternatives such as $W_1, W_2, W_3, W_4, W_5, W_6, W_7, W_8, W_9$, and W_{10} , the relative closeness (P_i) value of potential location is computed to solution through equation (12) which is shown in Table 12.

Positive ideal solution and negative ideal solution are used for finding ideal and nonideal separation measures. These separation measures are calculated by using equations (10) and (11). Ideal separation measures (S^+) for $W_1, W_2,$

TABLE 7: The CRITIC method normalized decision matrix.

Normalized decision matrix									
	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8	C_9
A_1	0.375	0.750	0.600	0.125	0.714	1.000	0.000	0.250	0.000
A_2	0.750	0.000	0.000	0.375	0.571	0.333	0.429	0.000	0.200
A_3	0.000	0.750	0.000	0.500	0.286	0.333	0.429	0.625	0.600
A_4	0.500	0.750	0.800	0.125	0.000	0.500	0.571	0.125	0.200
A_5	0.250	0.250	1.000	0.125	0.000	0.500	0.714	0.875	0.400
A_6	0.625	0.000	0.200	0.500	0.857	0.500	0.000	1.000	0.600
A_7	0.125	0.750	0.200	0.750	1.000	0.167	0.429	0.875	0.000
A_8	0.875	0.250	0.200	0.375	0.143	0.000	0.857	0.625	0.600
A_9	0.250	0.500	1.000	1.000	0.143	0.167	0.000	0.500	0.800
A_{10}	1.000	0.250	0.600	0.000	0.429	0.167	1.000	0.625	1.000
Standard deviation	0.332	0.313	0.389	0.314	0.359	0.281	0.359	0.334	0.337

TABLE 8: Correlation coefficients.

	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8	C_9
C_1	1.000	-0.655	-0.116	-0.462	-0.020	-0.238	0.452	-0.206	0.357
C_2	-0.655	1.000	0.178	0.117	-0.046	0.242	-0.166	-0.159	-0.389
C_3	-0.116	0.178	1.000	-0.143	-0.538	0.217	-0.007	-0.026	0.149
C_4	-0.237	0.117	-0.143	1.000	0.213	-0.424	-0.529	0.241	0.073
C_5	-0.020	-0.046	-0.543	0.213	1.000	0.189	-0.420	0.221	-0.362
C_6	0.000	0.242	0.217	-0.424	0.189	1.000	-0.503	-0.266	-0.523
C_7	0.452	-0.166	-0.007	-0.529	-0.420	-0.503	1.000	0.059	0.283
C_8	-0.206	-0.159	-0.026	0.241	0.221	-0.266	0.059	1.000	0.325
C_9	0.357	-0.389	0.149	0.073	-0.362	-0.523	0.283	0.325	1.000

TABLE 9: Measure of conflict calculations.

	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8	C_9
C_1	0.000	1.655	1.116	1.462	1.020	1.238	0.548	1.206	0.643
C_2	1.655	0.000	0.822	0.883	1.046	0.758	1.166	1.159	1.389
C_3	1.116	0.822	0.000	1.143	1.538	0.783	1.007	1.026	0.851
C_4	1.237	0.883	1.143	0.000	0.787	1.424	1.529	0.759	0.927
C_5	1.020	1.046	1.543	0.787	0.000	0.811	1.420	0.779	1.362
C_6	1.000	0.758	0.783	1.424	0.811	0.000	1.503	1.266	1.523
C_7	0.548	1.166	1.007	1.529	1.420	1.503	0.000	0.941	0.717
C_8	1.206	1.159	1.026	0.759	0.779	1.266	0.941	0.000	0.675
C_9	0.643	1.389	0.851	0.927	1.362	1.523	0.717	0.675	0.000

W_3 , W_4 , W_5 , W_6 , W_7 , W_8 , W_9 , and W_{10} are 0.050, 0.054, 0.045, 0.080, 0.067, 0.073, 0.066, 0.082, 0.054, and (0.078), respectively. Similarly, nonideal separation measures are calculated by equation (11) and values of S^- for W_1 , W_2 , W_3 , W_4 , W_5 , W_6 , W_7 , W_8 , W_9 , and W_{10} are 0.078, 0.074, 0.078, 0.080, 0.070, 0.059, 0.071, 0.047, 0.073, and 0.061, respectively.

Ranking is done upon the value of P_i , and the high value of P_i shows the top alternative. After relative closeness has been calculated, the ranking of workers is done based upon the value of P_i . In this research, alternative W_3 had higher P_i value among other alternatives and thus got the first ranking, while W_1 had second highest value and got rank 2, and so on. As W_3 had higher P_i value and was ranked as 1, it was more reliable among all the other workers and was the most appropriate for the selected task to be performed. The details are shown in Table 13.

FIGURE 3: Steps of the TOPSIS method.

From Table 13, ranking of the workers is clearly presented, and tasks will be assigned to the most suitable workers according to their ranking. Graphical representation of workers' ranking is shown in Figure 4.

As the figure shows, workers W_1 , W_2 , W_3 , W_4 , W_5 , W_6 , W_7 , W_8 , W_9 , and W_{10} are ranked as 2, 3, 1, 6, 7, 8, 5, 10, 4, and 9, respectively. These rankings are directly dependent

TABLE 10: Decision matrix.

Input data									
Attributes	Worker history	Trustworthiness/honesty	Qualification	Reliability	Response ratio/quality of response	Skill level	Quality of task	Delivery time	Cheap/cost-effective/cost-efficient
W1	6	7	5	8	4	3	9	7	8
W2	3	4	8	6	5	7	6	9	7
W3	9	7	8	5	7	7	6	4	5
W4	5	7	4	8	9	6	5	8	7
W5	7	5	3	8	9	6	4	2	6
W6	4	8	7	5	3	6	9	1	5
W7	8	5	7	3	2	8	6	2	8
W8	2	7	7	6	8	9	3	4	5
W9	7	6	3	1	8	8	9	5	4
W10	1	7	5	9	6	8	2	4	3

TABLE 11: Normalized decision matrix.

Attributes	Worker history	Trustworthiness/honesty	Qualification	Reliability	Response ratio/quality of response	Skill level	Quality of task	Delivery time	Cheap/cost-effective/cost-efficient
W1	0.33	0.35	0.26	0.40	0.19	0.14	0.45	0.42	0.42
W2	0.16	0.20	0.42	0.30	0.24	0.32	0.30	0.54	0.37
W3	0.49	0.35	0.42	0.25	0.34	0.32	0.30	0.24	0.26
W4	0.27	0.35	0.21	0.40	0.43	0.27	0.25	0.48	0.37
W5	0.38	0.25	0.16	0.40	0.43	0.27	0.20	0.12	0.32
W6	0.22	0.39	0.37	0.25	0.14	0.27	0.45	0.06	0.26
W7	0.44	0.39	0.37	0.25	0.14	0.27	0.45	0.06	0.26
W8	0.11	0.25	0.37	0.15	0.10	0.36	0.30	0.12	0.42
W9	0.38	0.35	0.37	0.30	0.39	0.41	0.15	0.24	0.26
W10	0.05	0.30	0.16	0.05	0.39	0.36	0.45	0.30	0.21
Weights	0.114	0.107	0.125	0.105	0.122	0.098	0.122	0.101	0.105

TABLE 12: Normalized decision matrix.

Weighted normalized data											
Attributes	Worker history	Trustworthiness/honesty	Qualification	Reliability	Response ratio/quality of response	Skill level	Quality of task	Delivery time	Cheap/cost-effective/cost-efficient	S^+	S^-
W1	0.037	0.037	0.033	0.042	0.023	0.013	0.055	0.042	0.044	0.050	0.078
W2	0.019	0.021	0.053	0.031	0.029	0.031	0.037	0.055	0.039	0.054	0.074
W3	0.056	0.037	0.053	0.026	0.041	0.031	0.037	0.024	0.028	0.045	0.078
W4	0.031	0.037	0.026	0.042	0.053	0.027	0.030	0.049	0.039	0.080	0.080
W5	0.044	0.026	0.020	0.042	0.053	0.027	0.024	0.012	0.033	0.067	0.070
W6	0.025	0.042	0.046	0.026	0.018	0.027	0.055	0.006	0.028	0.073	0.059
W7	0.050	0.042	0.046	0.026	0.018	0.027	0.055	0.006	0.028	0.066	0.071
W8	0.012	0.026	0.046	0.016	0.012	0.036	0.037	0.012	0.044	0.082	0.047
W9	0.044	0.037	0.046	0.031	0.047	0.040	0.018	0.024	0.028	0.054	0.073
W10	0.006	0.032	0.020	0.005	0.047	0.036	0.055	0.030	0.022	0.078	0.061
Positive ideal	0.056	0.042	0.053	0.042	0.053	0.040	0.055	0.055	0.044		
Negative ideal	0.006	0.021	0.020	0.005	0.012	0.013	0.018	0.006	0.022		

TABLE 13: Ranking of workers.

Workers	S^+	S^-	$S^+ + S^-$	P_i	Rank
W1	0.050	0.078	0.128	0.6087	2
W2	0.054	0.074	0.129	0.5776	3
W3	0.045	0.078	0.123	0.6353	1
W4	0.080	0.080	0.159	0.5000	6
W5	0.067	0.070	0.137	0.5085	7
W6	0.073	0.059	0.132	0.4481	8
W7	0.066	0.071	0.137	0.5180	5
W8	0.082	0.047	0.129	0.3650	10
W9	0.054	0.073	0.127	0.5758	4
W10	0.078	0.061	0.140	0.4389	9

FIGURE 4: Workers' ranking.

upon the P_i values of these workers, which have been calculated with the help of the TOPSIS method. Higher P_i value indicates higher rank, while lower P_i value indicates lower rank, as shown in Table 13.

4. Conclusion

Assigning the task to the most appropriate worker is very important in crowdsourcing because if the task is assigned to an inappropriate worker it affects crowdsourcing activity in several ways such as waste of time, money, and clients trusts. The proposed research presents a mechanism for assigning a task to the worker. This proposed mechanism is based on multiple criteria. Worker features such as worker history, trustworthiness/honesty, worker qualification, reliability, response ratio/quality of response, skill level, quality of task, delivery time, and cheap/cost-effective/cost-efficient are selected by the identified features. Two MCDM methods, CRITIC and TOPSIS, have been used. Weights have been assigned to these features by the CRITIC method and then evaluation and ranking of the workers have been analyzed by the TOPSIS method in order to assign the task to the most appropriate worker. As the existing task assignment is based on single criterion, the proposed work is novel in terms of assigning workers based on multiple criteria as well as using MCDM methods for current work in crowdsourcing.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that there are no conflicts of interest.

References

- [1] M. Hosseini, A. Shahri, K. Phalp, J. Taylor, and R. Ali, "Crowdsourcing: a taxonomy and systematic mapping study," *Computer Science Review*, vol. 17, pp. 43–69, 2015.
- [2] A. Sari, A. Tosun, and G. I. Alptekin, "A systematic literature review on crowdsourcing in software engineering," *Journal of Systems and Software*, vol. 153, pp. 200–219, 2019.
- [3] R. M. Borromeo, T. Laurent, and M. Toyama, "The influence of crowd type and task complexity on crowdsourced work quality," in *Proceedings of the 20th International Database Engineering & Applications Symposium*, Montreal, Canada, July 2016.
- [4] K. Mao, L. Capra, M. Harman, and Y. Jia, "A survey of the use of crowdsourcing in software engineering," *Journal of Systems and Software*, vol. 126, pp. 57–84, 2017.
- [5] Y. Baba, H. Kashima, K. Kinoshita, G. Yamaguchi, and Y. Akiyoshi, "Leveraging non-expert crowdsourcing workers for improper task detection in crowdsourcing marketplaces," *Expert Systems with Applications*, vol. 41, no. 6, pp. 2678–2687, 2014.
- [6] H. J. Khasraghi and A. Aghaie, "Crowdsourcing contests: understanding the effect of competitors," *Behaviour & Information Technology*, vol. 33, no. 12, pp. 1383–1395, 2014.
- [7] M. C. Yuen, I. King, and K. S. Leung, "Task recommendation in crowdsourcing systems," in *Proceedings of the First International Workshop on Crowdsourcing and Data Mining*, Beijing, China, August 2012.
- [8] G. Li, J. Wang, Y. Zheng, and M. Franklin, "Crowdsourced data management: a survey," in *Proceedings of the IEEE 33rd International Conference on Data Engineering (ICDE)*, San Diego, CA, USA, April 2017.
- [9] S. B. Roy, I. Lykourantzou, S. Thirumuruganathan, S. Amer-Yahia, and G. Das, "Task assignment optimization in knowledge-intensive crowdsourcing," *The VLDB Journal*, vol. 24, no. 4, pp. 467–491, 2015.

- [10] Z. Xiao, X. Dai, H. Jiang et al., "Vehicular task offloading via heat-aware MEC cooperation using game-theoretic method," *IEEE Internet of Things Journal*, vol. 7, pp. 2038–2052, 2019.
- [11] H. Zhang and M. Sugiyama, "Task selection for bandit-based task assignment in heterogeneous crowdsourcing," in *Proceedings of the 2015 Conference on Technologies and Applications of Artificial Intelligence (TAAI)*, pp. 164–171, Tainan, Taiwan, November 2015.
- [12] C. Qiu, A. C. Squicciarini, B. Carminati, J. Caverlee, and D. R. Khare, "CrowdSelect increasing accuracy of crowdsourcing tasks through behavior prediction and user selection," in *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management*, Indianapolis, IN, USA, October 2016.
- [13] T. Awwad, N. Bennani, K. Ziegler, V. Sonigo, L. Brunie, and H. Kosch, "Efficient worker selection through history-based learning in crowdsourcing," in *Proceedings of the IEEE 41st Annual Computer Software and Applications Conference (COMPSAC)*, pp. 923–928, Turin, Italy, July 2017.
- [14] M. Z. Tunio, H. Luo, W. Cong et al., "Impact of personality on task selection in crowdsourcing software development: a sorting approach," *IEEE Access*, vol. 5, pp. 18287–18294, 2017.
- [15] Z. Wang, H. Sun, Y. Fu, and L. Ye, "Recommending crowdsourced software developers in consideration of skill improvement," in *Proceedings of the 32nd IEEE/ACM International Conference on Automated Software Engineering*, Urbana, IL, USA, November 2017.
- [16] Y. Xiaoyan, C. Yanjiao, and L. Baochun, "Task assignment with guaranteed quality for crowdsourcing platforms," in *Proceedings of the IEEE/ACM 25th International Symposium on Quality of Service (IWQoS)*, pp. 1–10, Vilanova i la Geltrú, Spain, June 2017.
- [17] H. Sun, B. Dong, B. Zhang, W. H. Wang, and M. Kantarcioglu, "Sensitive task assignments in crowdsourcing markets with colluding workers," in *Proceedings of the IEEE 34th International Conference on Data Engineering (ICDE)*, pp. 377–388, Paris, France, April 2018.
- [18] L. Cui, X. Zhao, L. Liu, H. Yu, and Y. Miao, "Learning complex crowdsourcing task allocation strategies from humans," in *Proceedings of the 2nd International Conference on Crowd Science and Engineering*, Beijing, China, July 2017.
- [19] M. Safran and D. Che, "Real-time recommendation algorithms for crowdsourcing systems," *Applied Computing and Informatics*, vol. 13, no. 1, pp. 47–56, 2017.
- [20] A. Burnap, R. Gerth, R. Gonzalez, and P. Y. Papalambros, "Identifying experts in the crowd for evaluation of engineering designs," *Journal of Engineering Design*, vol. 28, no. 5, pp. 317–337, 2017.
- [21] A. Moayedikia, K. L. Ong, Y. L. Boo, and W. G. S. Yeoh, "Task assignment in microtask crowdsourcing platforms using learning automata," *Engineering Applications of Artificial Intelligence*, vol. 74, pp. 212–225, 2018.
- [22] Z. Xiao, F. Li, H. Jiang et al., "A joint information and energy cooperation framework for CR-enabled macro-femto heterogeneous networks," *IEEE Internet of Things Journal*, vol. 7, pp. 2828–2839, 2019.
- [23] J. Jiang, B. An, Y. Jiang, P. Shi, Z. Bu, and J. Cao, "Batch Allocation for tasks with overlapping skill requirements in crowdsourcing," *IEEE Transactions on Parallel and Distributed Systems*, vol. 30, no. 8, pp. 1722–1737, 2019.
- [24] Y. Du, Y. E. Sun, H. Huang, L. Huang, H. Xu, and X. Wu, "Quality-aware online task assignment mechanisms using latent topic model," *Theoretical Computer Science*, vol. 803, pp. 130–143, 2019.
- [25] Q. Pan, H. Dong, Y. Wang, Z. Cai, and L. Zhang, "Recommendation of crowdsourcing tasks based on Word2vec semantic tags," *Wireless Communications and Mobile Computing*, vol. 2019, Article ID 2121850, 10 pages, 2019.
- [26] D. Yu, Y. Wang, and Z. Zhou, "Software crowdsourcing task allocation algorithm based on dynamic utility," *IEEE Access*, vol. 7, pp. 33094–33106, 2019.
- [27] D. Yu, Z. Zhou, and Y. Wang, "Crowdsourcing software task assignment method for collaborative development," *IEEE Access*, vol. 7, pp. 35743–35754, 2019.
- [28] H. Jiang, Z. Xiao, Z. Li, J. Xu, F. Zeng, and D. Wang, "An energy-efficient framework for internet of things underlying heterogeneous small cell networks," *IEEE Transactions on Mobile Computing*, vol. 99, p. 1, 2020.
- [29] Z. Qin, D. Wu, Z. Xiao, B. Fu, and Z. Qin, "Modeling and analysis of data aggregation from convergecast in mobile sensor networks for industrial IoT," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 10, pp. 4457–4467, 2018.
- [30] Z. Xiao, X. Shen, F. Zeng et al., "Spectrum resource sharing in heterogeneous vehicular networks: a noncooperative game-theoretic approach with correlated equilibrium," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 10, pp. 9449–9458, 2018.
- [31] R. Khazankin, H. Psai, D. Schall, and S. Dustdar, "QoS-based task scheduling in crowdsourcing environments," in *Proceedings of the Service-Oriented Computing*, pp. 297–311, Berlin, Germany, December 2011.
- [32] D. Schall, B. Satzger, and H. Psai, "Crowdsourcing tasks to social networks in BPEL4People," *World Wide Web*, vol. 17, no. 1, pp. 1–32, 2014.
- [33] A. Tarasov, S. J. Delany, and B. M. Namee, "Dynamic estimation of worker reliability in crowdsourcing for regression tasks: making it work," *Expert Systems with Applications*, vol. 41, no. 14, pp. 6190–6210, 2014.
- [34] L. Machado, R. Prikladnicki, F. Meneguzzi, C. R. B. D. Souza, and E. Carmel, "Task allocation for crowdsourcing using AI planning," in *Proceedings of the 3rd International Workshop on CrowdSourcing in Software Engineering*, Austin, TX, USA, May 2016.
- [35] D. Geiger and M. Schader, "Personalized task recommendation in crowdsourcing information systems—current state of the art," *Decision Support Systems*, vol. 65, pp. 3–16, 2014.
- [36] R. L. Saremi and Y. Yang, "Dynamic simulation of software workers and task completion," in *Proceedings of the Second International Workshop on CrowdSourcing in Software Engineering*, Florence, Italy, May 2015.
- [37] W. Maalej and M. Ellmann, "On the similarity of task contexts," in *Proceedings of the Second International Workshop on Context for Software Development*, Florence, Italy, May 2015.
- [38] A. Ellero, P. Ferretti, and G. Furlanetto, "Realtime crowdsourcing with payment of idle workers in the retainer model," *Procedia Economics and Finance*, vol. 32, pp. 20–26, 2015.
- [39] A. Carvalho, S. Dimitrov, and K. Larson, "How many crowdsourced workers should a requester hire?" *Annals of Mathematics and Artificial Intelligence*, vol. 78, no. 1, pp. 45–72, 2016.

- [40] A. S. Fonteles, S. Bouveret, and J. Gensel, "Trajectory recommendation for task accomplishment in crowdsourcing—a model to favour different actors," *Journal of Location based Services*, vol. 10, no. 2, pp. 125–141, 2016.
- [41] L. Ning, Y. Ali, H. Ke, S. Nazir, and Z. Huanli, "A hybrid MCDM approach of selecting lightweight cryptographic cipher based on ISO and NIST lightweight cryptography security requirements for internet of health things," *IEEE Access*, vol. 8, Article ID 220187, 2020.