A New Clustering Algorithm for Task Packaging Problem in Crowdsourcing Model

Content

A New Clustering Algorithm for Task Packaging Problem in Crowdsourcing Model

1. Abstract

In this project, we take "taking photos to make money", a self-service labor crowdsourcing task as an example. we attempt to use a clustering algorithm to unite tasks together for publishing. The key reason for task packaging publish is to improve the user's willingness to accept tasks, thus improving task completion. However, the K-means clustering algorithm is not very effective in this kind of problem, because the number of the task package is an important indicator to measure the packaging effect. If the K value is specified artificially, the final clustering effect cannot be evaluated. Also, the Greedy Algorithm is based on the local optimal idea, which cannot be used to achieve a global optimal packaging scheme for packaging problems. Therefore, we developed a New Clustering Algorithm for task packaging based on the optimal radius.

First, the original data of users and tasks is visualized and a significantly negative correlation is found between the number of users around a task (User Density) and the task price. Different task radius leads to different User Density and different User Density leads to different negative correlation coefficients between task-user densities and task price. Then using polynomial regression to get the User Density when the negative correlation coefficient is the largest, and selecting its radium as the optimal radius(threshold) and as the basis of clustering. According to the idea of hierarchical clustering, a new packing algorithm was designed, which can automatically be calculated and divided the clusters by giving the optimal radius, and obtained the number of subtasks in each package, the total price, and completion.

Finally, the packaging result of the New Clustering Algorithm is evaluated by the distance between two subtask points in each packaged task, the number of subtasks and the total number of packaged tasks. The experimental results show that the New Clustering algorithm performs better than the Greedy Algorithm and K-Means Algorithm in Task Packaging Problem.

2. Introduction

2.1 Background

Crowdsourcing is a sourcing model in which individuals or organizations obtain goods and services, including ideas and finances, from a large, relatively open and often rapidly-evolving group of internet users instead of employees. There are many advantages in the crowdsourcing model: problems can be explored and discussed with lower cost and less time; only pay when there are results; organizations can rely on a wider range of people; By listening to the user's voice, the organization can first-hand insight into customer needs. With the development of Internet and mobile communication, crowdsourcing mode is becoming more and more popular, and may even become the outsourcing terminator. In crowdsourcing mode, combining some tasks to a package can not only improve task completion but also reduce the total cost.

2.2 Problem Identification

The user downloads the app, registers as a member of the app and then receives the tasks that need to be completed from the app (for example, going to the supermarket to photograph the listing of a certain product), and earns the remuneration for the tasks. In reality, multiple tasks may be located in a centralized area, which leads to users Scramble for. Sometimes, we need to package some tasks together. It requires consideration of the latitude and longitude position of the task, the number of tasks, the membership density, and the price of each task package. The differences in task counts, task position, membership density, and the price of the task package lead to different packaging results, which affects the completion of the task.

2.3 Related Works

In the field of crowdsourcing task packaging, the algorithms used by people are the K-Means and Greedy Algorithms. Both of them have limitations in packaging tasks.

K-means clustering is the most famous clustering algorithm, which is widely used in all clustering algorithms due to its simplicity and efficiency. Given a set of data points and the number of clusters K, K-means clustering divides the data into K clusters according to a certain distance function. One of the drawbacks is that the K-means clustering algorithm requires artificially specifying the value of K. The value of K is often defined according to people's experience. For some specific problems, the K value is difficult to determine, resulting in poor clustering.

A Greedy Algorithm is an algorithm that follows the problem-solving heuristic of making the locally optimal choice at each stage[5] with the intent of finding a global optimum. For a set of data, Greedy Algorithm finds the optimal value and process it, and then find the optimal value and process it, which means taking the optimal choice in the current state in each step selection, so that it is expected to get the optimal result.

2.4 Novelty of New Clustering Algorithm

Compared with K-means, the New Clustering algorithm combine tasks and users data and calculates the key threshold "optimal radius" as the basis for clustering. As a result, different user distributions and different task prices will affect the final packaging result, which not need a specified K value as K-Means Algorithm. Additionally, the K-Means algorithm selects randomly different initial points which lead to different results for each clustering, and the packaging results of the New Clustering algorithm are the same in the same data.

The Greedy algorithm packs the optimal package in the current state in each step, and each package once packed would not be changed anymore. Compared Greedy Algorithm, the New Clustering algorithm selects tasks dynamically at each step which means new attachment points may be added in a previous package, not a one-time packaging. The packaging result of The new clustering algorithm shows more reasonable in the distribution of sub-tasks and a larger packing rate.

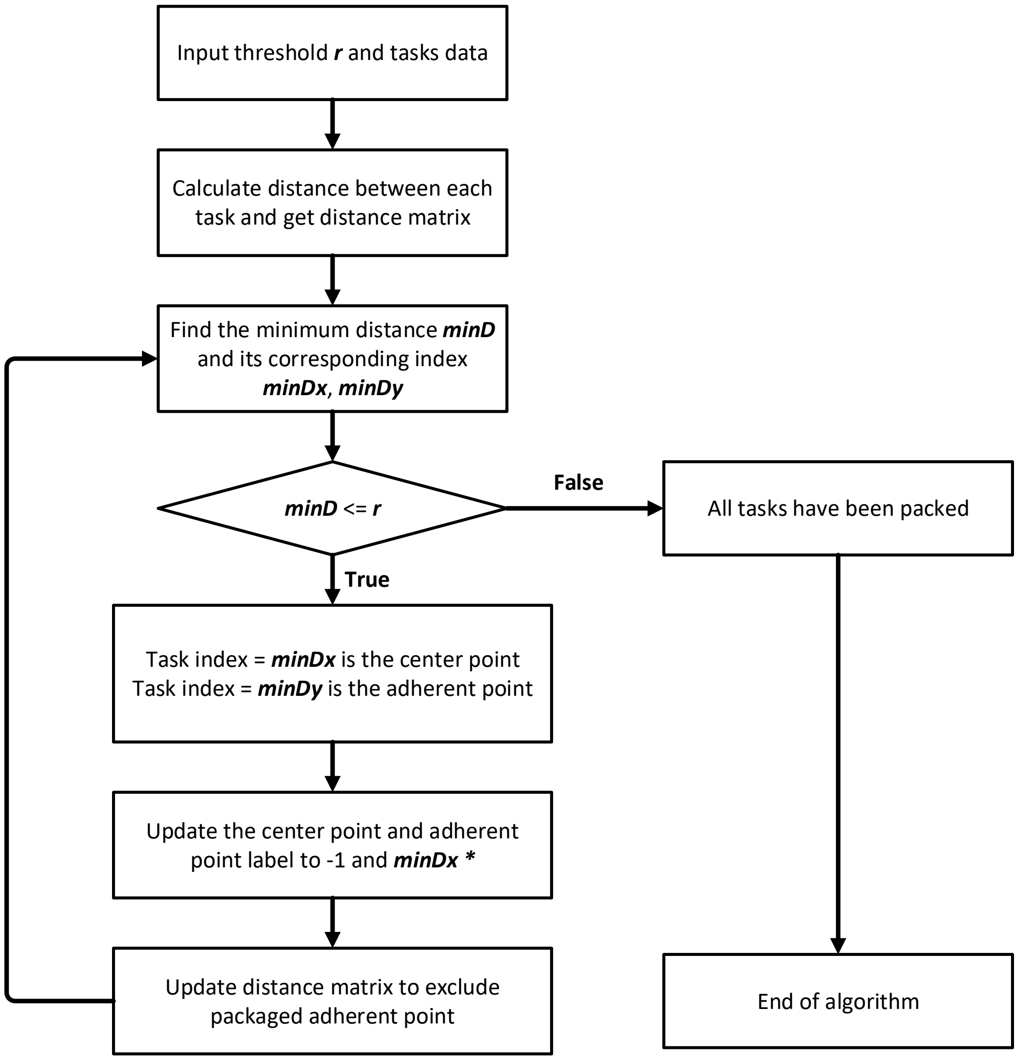
Therefore, the New Clustering greatly optimizes the original two algorithms.

3. Methodology

3.1 the New Clustering Algorithm Design

Referring to the hierarchical clustering method, the New Clustering Algorithm calculates the distance between all task points. Among these distance, select the minimum distance in each step (Figure 1). When the is minimum distance less than or equal to the threshold value, there are still task points that need to be packed. Package two points corresponding to the minimum distance and keep cycling this process. The packaging is done when the minimum distance greater than the threshold value which means the task points are far away from each other after clustering and not necessary to be packed.

Algorithm execution steps: Firstly, input the task data and the optimal radius ***r*** as thresholds; Secondly, calculate the distance between each task point and get the distance matrix; Thirdly, take the minimum value of distance matrix as **minD** and the corresponding position (**minDx**, **minDy**). When **minD** less than and equal to threshold r, set the task point corresponding to **minDx** as a center point and the task point corresponding to **minDy** as an attachment point of this center point. Then the task point packed as an attachment point complete packaging and the task point packed as a center point can continue to become a center point but not an attachment point of other unpacked task point. After that, set the label of the center point and the attachment point as -1 and **minDy** and update the distance matrix. After each packing, update the distance matrix, get a new minimum distance **minD**, and repeat this process. When **minD** is greater than threshold **r**, the task packaging has completed and output the result.



* Center point never become a adherent point

Figure 1 New Clustering Algorithm Flow Diagram

To explain the packaging process of the New Clustering Algorithm more clearly, take five task points as an example to execute the process of algorithm. The detail packaging process of the New Clustering Algorithm is shown in Figure 2.

Figure 2 Using New Clustering in a Simple Sample

3.2 Algorithm Evaluation

The advantages and disadvantages of the New Clustering Algorithm, K-Means Algorithm, and Greedy Algorithm are shown in table 1.

Table 1 the Advantages and Disadvantages among Three Algorithms

	advantages	disadvantages
New Clustering	● Not need given K ● Can find the optimal solution ● More clusters	● High time complexity
Greedy Algorithm	● Not need given K	● Nor complete, only local optimal solution ● High time complexity ● Less clusters
K-Means Algorithm	● Can find the optimal solution ● Low time complexity	● Need given K ● Different initial points lead to different clustering results

4. Experimental Study

4.1 Experimental procedure

4.1.1 Data Source

The data in the experiment is from open-source data on the Internet, including task data and user data (Table 2). The task data contains 835 task records and the attributes of the data set include Task ID, Latitude of Task, Longitude of Task, Task Pricing, and Task Completion status. The data contains 1877 pieces of user information including User ID, Latitude of User, Longitude of User, Task Quota, Start Time of Task, and Credit. These data are in the Data folder of the supporting file.

Table 2 Part of Data Source

Task ID	Latitude of Task	Longitude of Task	Task Pricing	Task Completion status	
A0001	22.56614225	113.9808368	66	0	
A0002	22.68620526	113.9405252	65.5	0	
A0003	22.57651183	113.957198	65.5	1	
A0004	22.56484081	114.2445711	75	0	
A0005	22.55888775	113.9507227	65.5	0	
A0006	22.55899906	114.2413174	75	0	
A0007	22.54900371	113.9722597	65.5	1	
A0008	22.56277351	113.9565735	65.5	0	
A0009	22.50001192	113.8956606	66	0	
A0010	22.5437861	113.9239778	66	1	
User ID	Latitude of User	Longitude of User	Task Quota	Start Time of Task	Credit
B0001	22.947097	113.679983	114	6:30:00	67997.3868
B0002	22.577792	113.966524	163	6:30:00	37926.5416
B0003	23.192458	113.347272	139	6:30:00	27953.0363
B0004	23.255965	113.31875	98	6:30:00	25085.6986
B0005	33.65205	116.97047	66	6:30:00	20919.0667
B0006	22.262784	112.79768	72	6:30:00	18237.6295
B0007	29.560903	106.239083	15	6:30:00	15729.3601
B0008	23.143373	113.376315	95	6:42:00	14868.4446
B0009	23.28528	113.651842	110	6:36:00	13556.1555
B0010	23.099259	113.488909	64	6:36:00	13327.9511

4.1.2 Data Preprocess

After statistical analysis of each indicator in the user data, it was found that the range of the user GPS latitude and user GPS longitude is too large which means there is a problem in data collection. The problem was caused by the inversion of latitude and longitude records (Table 3), so this outlier was corrected.

Table 3 Outlier in User Data

1174	B1173	22.95841	113.083468	7	8:00:00	19.9231
1175	B1174	22.870735	113.070645	8	6:51:00	19.9231
1176	B1175	113.131483	23.031824	1	6:36:00	19.9231
1177	B1176	23.199288	112.862711	7	6:39:00	19.9231
1178	B1177	23.044966	113.070853	8	8:00:00	19.9231

4.1.3 Data Visualization

To visualize the distribution of tasks and users, we used Tableau to draw a scatter plot on the map. The green points shown in Figure 3 indicate the position of the users. The other points are the positions of the tasks where the color reflects the price of the task. The task point closer red task has a higher task price. The figure shows an inverse proportional relationship between the User Density and the task price. The task price is lower in areas where users are concentrated.

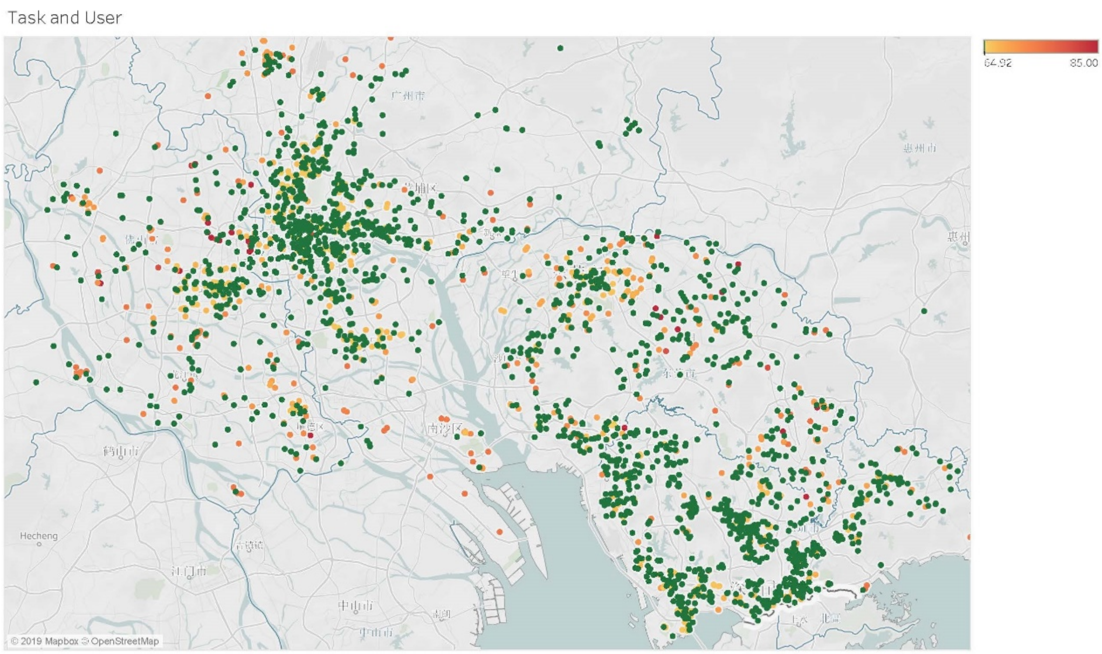


Figure 3 the Distribution of Tasks and Users

4.1.4 New Clustering Algorithm

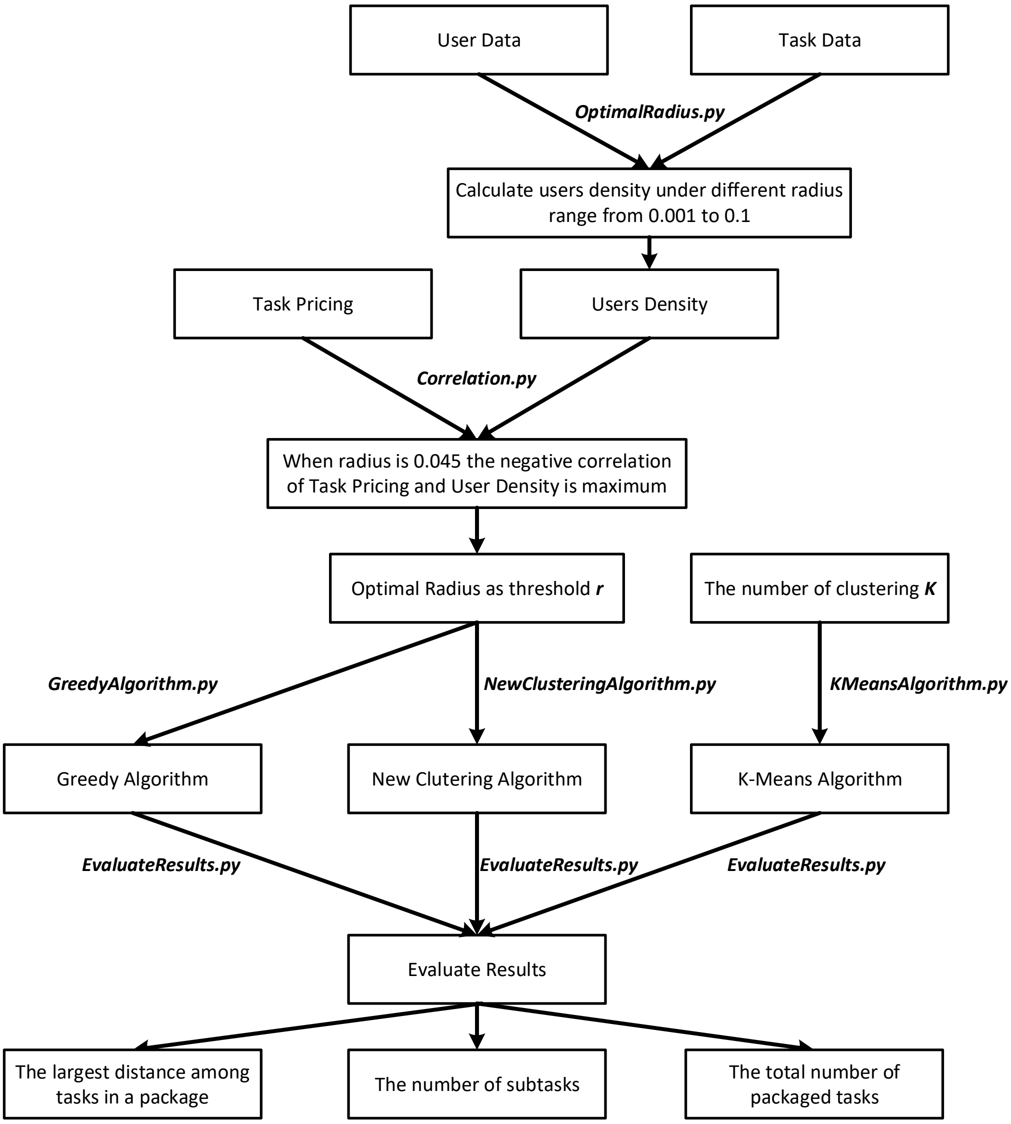


Figure 4 Experiment Flow Diagram

Figure 4 shows the whole process of the experiment in the project. First, input user data and task data, run the OptimalRadius.py file to calculate User Density under different radius range from 0.001 to 0.1. Then run the correlation.py file to carry out the polynomial expression with the User Density and task price, and get the result as Figure 5.

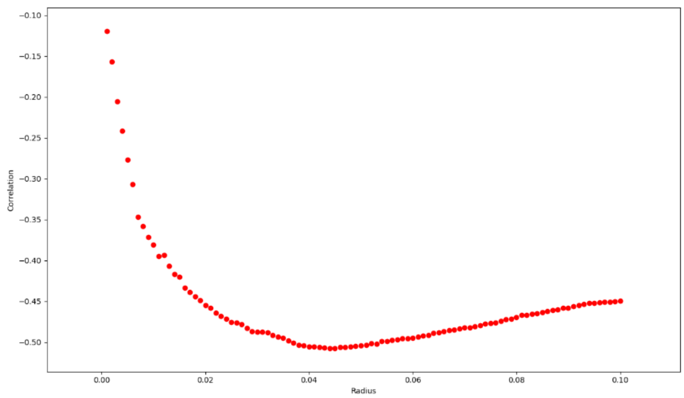


Figure 5 Polynomial Regression between Task Price and User Density

In the correlation coefficient table (Table 4) which was exported by the program, the negative correlation of Task Pricing and User Density is maximum when the radius is 0.045. So we chose 0.045 as the optimal radius, which is the threshold input in the New Clustering Algorithm and Greedy algorithm.

Table 4 Correlation

Radius	Correlation	Radius	Correlation	Radius	Correlation	Radius	Correlation	Radius	Correlation
0.045	-0.50766857	**0.056**	-0.497646588	**0.069**	-0.48386789	**0.083**	-0.465307576	**0.1**	-0.449417376
0.044	-0.507375397	**0.057**	-0.496691781	**0.028**	-0.482900095	**0.084**	-0.465150877	**0.019**	-0.448929234
0.043	-0.507186471	**0.059**	-0.495777529	**0.07**	-0.482331732	**0.022**	-0.464230553	**0.018**	-0.444301527
0.042	-0.50624099	**0.058**	-0.495438209	**0.071**	-0.482164653	**0.085**	-0.463712141	**0.017**	-0.439174133
0.046	-0.506074368	**0.06**	-0.495228595	**0.072**	-0.48079989	**0.086**	-0.462017713	**0.016**	-0.433290554
0.047	-0.50596119	**0.035**	-0.494925279	**0.073**	-0.479439894	**0.087**	-0.461163847	**0.015**	-0.420540038
0.048	-0.505664458	**0.061**	-0.493875966	**0.027**	-0.478266693	**0.088**	-0.460171421	**0.014**	-0.416865697
0.041	-0.505604482	**0.034**	-0.493553922	**0.074**	-0.477728587	**0.021**	-0.458533492	**0.013**	-0.407160938
0.04	-0.505457044	**0.062**	-0.492267696	**0.075**	-0.476901687	**0.089**	-0.45844358	**0.011**	-0.394712891
0.049	-0.5046624	**0.033**	-0.49177993	**0.076**	-0.476170059	**0.09**	-0.458209173	**0.012**	-0.393676651
0.05	-0.504214827	**0.063**	-0.491395649	**0.026**	-0.4760353	**0.091**	-0.456402789	**0.01**	-0.381014381
0.039	-0.504195927	**0.064**	-0.488842828	**0.025**	-0.475493854	**0.02**	-0.4550576	**0.009**	-0.371566997
0.038	-0.503578693	**0.032**	-0.488388754	**0.077**	-0.474316925	**0.092**	-0.454651912	**0.008**	-0.358228306
0.051	-0.503401137	**0.065**	-0.488319018	**0.078**	-0.472520498	**0.093**	-0.453381602	**0.007**	-0.346901291
0.053	-0.502253971	**0.031**	-0.487381487	**0.079**	-0.471442463	**0.094**	-0.452165739	**0.006**	-0.306803869
0.052	-0.501568403	**0.03**	-0.487315928	**0.024**	-0.471268177	**0.095**	-0.451957777	**0.005**	-0.276739826
0.037	-0.50065002	**0.066**	-0.486712812	**0.08**	-0.469903756	**0.096**	-0.451545596	**0.004**	-0.241344451
0.054	-0.499140915	**0.029**	-0.486685641	**0.023**	-0.468380548	**0.098**	-0.450952042	**0.003**	-0.20554444
0.055	-0.499138226	**0.067**	-0.485723585	**0.081**	-0.467158539	**0.097**	-0.450866483	**0.002**	-0.157242873
0.036	-0.497946012	**0.068**	-0.48474928	**0.082**	-0.466680536	**0.099**	-0.450282881	**0.001**	-0.119316103

Input the optimal radius of 0.045 to GreedyAlgorithm.py and NewClusteringAlgorithm.py and set K=300 in K-MeansAlgorithm.py. Execute the above three python program and then execute EvaluateResults.py outputting the packaging results. The part of the packing results of the three algorithms are shown in Table 5.

Table 5 Part of Packages after Clustering

K-Means Algorithm				New Clustering Algorithm				Greedy Algorithm			
id	count	price	complete	id	count	price	complete	id	count	price	complete
0	3	202.5	0	1	3	197.5	1	2	7	479.5	0
1	7	456.5	3	2	5	341	0	4	2	150	0
2	5	335.5	5	3	3	197.5	1	9	9	593	7
3	4	321.5	4	4	2	150	0	31	3	200.5	0
4	4	290	0	5	4	263	2	33	27	1780	13
5	4	276.5	1	9	2	132	0	34	12	811	1
6	2	143	2	10	3	198	3	38	2	150	0
7	14	1000	14	11	2	131	1	40	3	205	1
8	1	70	1	12	4	262	2	42	6	401	1
9	7	465.5	1	13	2	131	2	43	7	466	3
10	2	133	2	15	4	264	4	46	3	201.5	0

4.2 evaluation method

We evaluate the packaging results of the three algorithms by comparing the following four indicators:

(1) The task completion rate: The higher task completion rate, the better packaged result of algorithm is.

(2) The largest distance among tasks in a package: The smaller the maximum distance between the two task points in the package, the better packaged result of algorithm.

(3) The number of subtasks: The better the subtasks in all packages should be moderate because the excessive subtasks cannot guarantee completion rate, too few subtasks are not very attractive to members.

(4) The total number of packaged tasks: The more the total number of packages after packaging, the better the packaging rate is, the better packaged result of algorithm is.

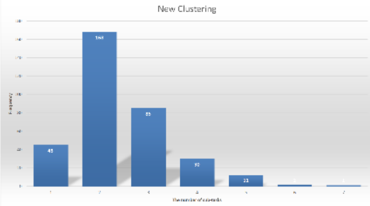
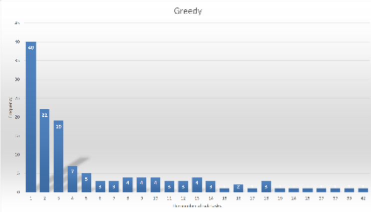
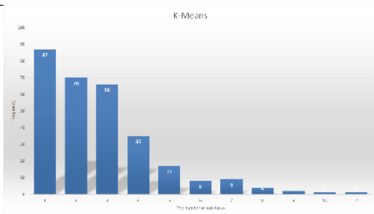
4.3 Results Analysis

Table 6 the Task Completion Rate

K-Means Algorithm					New Clustering Algorithm					Greedy Algorithm				
count	complete	complete/count	>=0.05 1	complete total	count	complete	complete/count	>=0.05 1	complete total	count	complete	complete/count	>=0.05 1	complete total
3	0	0	0	0	3	1	0.333333333	0	0	7	0	0	0	0
7	3	0.428571429	0	0	5	0	0	0	0	2	0	0	0	0
5	5	1	1	5	3	1	0.333333333	0	0	9	7	0.777777778	1	9
4	4	1	1	4	2	0	0	0	0	3	0	0	0	0
4	0	0	0	0	4	2	0.5	1	4	27	13	0.481481481	0	0
4	1	0.25	0	0	2	0	0	0	0	12	1	0.083333333	0	0
2	2	1	1	2	3	3	1	1	3	2	0	0	0	0
14	14	1	1	14	2	1	0.5	1	2	3	1	0.333333333	0	0
1	1	1	1	1	4	2	0.5	1	4	6	1	0.166666667	0	0
7	1	0.142857143	0	0	2	2	1	1	2	7	3	0.428571429	0	0
2	2	1	1	2	4	4	1	1	4	3	0	0	0	0

The calculation method is to use the total completion in a package divided by the total number of sub-tasks in this package. The ratio greater than or equal to 0.5 is counted as 1 for completion, and less than 0.5 is counted as 0 for incomplete. Then multiply the completion by the number of subtasks in a package, and sum the multiplying results in all packages and divide by the total number of tasks to get the task completion rate (Table 6).

Table 7 Comparison of Three Algorithms Results

	the largest distance among tasks in a package	the total number of packaged tasks	the number of subtasks	the task completion rate
New Clustering	r = 0.045 the largest distance under diameter = 0.09	343		(559/835)*100% = 67%
Greedy Algorithm	r = 0.045 the largest distance under diameter = 0.09	138		(517/835)*100% = 62%
K-means Algorithm	depends on different initial points	K		(545/835)*100% = 65%

According to the four indicators among three algorithms (Table 7), the result of New clustering algorithm is the best.

The result of New clustering algorithm: the largest distance among tasks in a package is the diameter(0.09), which is equal to 9.2223 kilometer according to the conversion formula between degrees and kilometers is roughly 102.47 * 0.09 = 9.2223 kilometer (Table 8 is the corresponding conversion method); The total number of packaged tasks is 343, and the number of subtasks is distributed in the interval [1,7], which is a reasonable interval acceptable to the user; Finally, the task completion rate(67%) is the highest of the three algorithms.

The result of Greedy algorithm: the largest distance among tasks in a package is the diameter(9.2223 kilometer) which is also a moderate distance for a user; the total number of packaged tasks is 138 and the number of subtasks is distributed in the interval [1,42] which means a task package has 42 subtask and is very unreasonable; The task completion rate(62%) is the lowest.

The result of K-Means Algorithm: the largest distance among tasks in a package will vary according to the value of K and the initial center points , and the packaging results are different each time; the total number of packaged tasks is the given K; Set K=3 and random initial center points, the number of subtasks is distributed in the interval [1,14] and is a little unacceptable for users; the task completion rate is 65%.

Table 8 Convert from Degree to Kilometer

Degree precision versus length

decimal places	decimal degrees	DMS	Object that can be *unambiguously* recognized at this scale	N/S or E/W at equator	E/W at 23N/S	E/W at 45N/S	E/W at 67N/S
0	1.0	1° 00′ 0″	country or large region	111.32 km	102.47 km	78.71 km	43.496 km

5. Conclusion

We developed a New Clustering Algorithm for task packaging based on the optimal radius. On the one hand, the radius can be used to measure and divide different clusters. On the other hand, radius can associate task data with user data. Then We use the New Clustering Algorithm, taking the optimal radius 0.045 as the judgment basis of clustering or not. Finally, through continuous iteration, different tasks can be packaged into one task and the number of subtasks in each task package can be counted. At the same time, we also implement the Greedy Algorithm and K-Means Algorithm.

The experimental result indicates the large distance along tasks in a package of the three algorithms is moderate; in the term of total number of packaged tasks, the New Clustering Algorithm packs the largest number of packages, and the packaging rate is the highest, so the effect is the best in this evaluation index; the number of subtasks of the New Clustering Algorithm in a reasonable range, It will not lead to the situation that users cannot complete due to too many subtasks; The task completion rate of the New Clustering Algorithm is also the highest among the three algorithms. Therefore, the packaged effect of the New Clustering Algorithm is the best among the three algorithms.

In summary, the New clustering algorithm can effectively solve task packaging problems in the crowdsourcing model. It optimizes the disadvantages of the Greedy and K-Means Algorithm, which has great practical significance.

6. Reference

[1] Data Source: China Undergraduate Mathematical Contest in Modeling. Available at http://www.mcm.edu.cn/.

[2] Crowdsourcing. Available at https://en.wikipedia.org/wiki/Crowdsourcing

[3] Decimal degrees. Available at https://en.wikipedia.org/wiki/Decimal_degrees

[4] (Introduction to Algorithms (Cormen, Leiserson, Rivest, and Stein) 2001, Chapter 16 "Greedy Algorithms".

7. Appendix

7.1 Notation Explanation

Notation	Explanation
r	Threshold, radium
minD	The minimum distance in distance matrix
minDx	The row index of **minD** in distance matrix
minDy	The column index of **minD** in distance matrix
K	The number of clustering

7.2 Attachment Explanation

Attachment	Explanation
Correlation.py	Conduct regression between different radius and corresponding coefficient of correlation
OptimalRaidus.py	Calculate the density of users
NewClusteringAlgorithm.py	Packing tasks by using New Clustering Algorithm
GreedyAlgorithm.py	Packing tasks by using the Greedy Algorithm
K-MeansAlgorithm.py	Packing tasks by K-means Algorithm
EvaluateResults.py	Evaluate packing results, and calculate the number of sub-task and the sum of completion and price in each package

7.3 Supporting Document

- Data
- Figures & Tables
- Result
- Correlation.py
- EvaluateResults.py
- GreedyAlgorithm.py
- KMeansAlgorithm.py
- NewClusteringAlgorithm.py
- OptimalRadius.py

In the Data folder, it's open data source on the website, also the file read by python; In Figures & Tables folder, it's the figures and tables used in the report; In the Result folder, it's the output results after executing python files, the file name is the same as python file name; Other Files are python code files mentioned in the report.