A Service Clustering Method based on Wisdom of Crowds

Hui Gao[*], Karolina K. Dluzniak[†], Hong Xia[*‡] Wei Jie[†], Yanping Chen[*‡], Wei Xing[‡], Xin Wang[*], Zhongmin Wang[*‡]

[*]School of Computer Science Xian University of Posts and Telecommunications, Xian, China
Email: hgao199523@163.com

[†]School of Computing and Engineering
University of West London, United Kingdom
Email: kainka.github@gmail.com

[*]School of Computer Science Xian University of Posts and Telecommunications, Xian, China
[‡]Shaanxi Key Laboratory of Network Data Analysis and Intelligent Processing
Email: xiahong@xupt.edu.cn

[†]School of Computing and Engineering
University of West London, United Kingdom
Email: wei.jie@uwl.ac.uk

[*]School of Computer Science Xian University of Posts and Telecommunications, Xian, China
[‡]Shaanxi Key Laboratory of Network Data Analysis and Intelligent Processi
Email: chenyp@xupt.edu.cn

[‡]Shaanxi Key Laboratory of Network Data Analysis and Intelligent Processing
Email: xingwei@xupt.edu.cn

[*]School of Computer Science Xian University of Posts and Telecommunications, Xian, China
Email: wang xin9426@163.com

[*]School of Computer Science Xian University of Posts and Telecommunications, Xian, China
[‡]Shaanxi Key Laboratory of Network Data Analysis and Intelligent Processi
Email: zm wang@xupt.edu.cn

***Abstract*—As the number and variety of services increase, it is becoming difficult and time-consuming to locate services that satisfy users' need. Service clustering is efficacious method to prune the query space, in order to narrow the search space, and improve the accuracy of locating services that satisfied users' needs. At present, clustering method of web services adopted single or traditional clustering algorithms. However, accuracy and stability of single or traditional clustering algorithms is poor. In this paper, we proposed SWOC a service clustering method based on wisdom of crowd. Firstly, by using SWOC we calculated document similarity. Secondly, we implemented a mapping algorithm that reduces the correlation of web services and improve accuracy of method. And then, we applyed different number of clusters using different individual clustering methods that increase the number of partitions in order to enhance the robustness of SWOC. Lastly, the diversity algorithm evaluates and selects the partitions to extract interesting information for the final aggregation with the weight of each individual result. Experiments were conducted on the real web service dataset crawled from ProgrammableWeb which demonstrate the accuracy, recall, F-value and stability of proposed method.**

***Index Terms*—cluster; service clustering; wisdom of crowds; clustering analysis; clustering ensemble**

I. Introduction

With the development of service oriented architecture technology and Software as a service (Saas), services on the Internet are showing a trend of rapid growth, and a large number of Internet applications have been created. By 2017, the number of published web services in website ProgrammableWeb has reached more than reached more than 12000. In order to use and integrate existing services, users need to find and match services that meet their needs from a large number of services on the Services Registration Platform. However, with the increasing number of services and service functions on the Internet, it is becoming more and more difficult and time-consuming to locate services accurately that satisfy users' specific business needs from a large number of service sets, which are difficult to define with different functional attributes. Therefore, how to discover services accurately and efficiently that meet users' needs has become a difficult problem in the field of Service Oriented Computing (SOC). Clustering services with similar functions can effectively perform service discovery [1] [2].

Service clustering is an effective method of assistanting service management and composition. Its main objective is to classify services into different types according to their functions, i.e. to divide all services into several functional independent categories. It makes the functions of services in the same category have high similarity, but the service functions among different categories have great differences, which can narrow the scope of service search, speed up the search and improve the accuracy of search. However, services are developed by different organizations, so it is difficult to use a unified method to extract useful information, and the information extracted lacks a unified standard. So that different clustering results can be obtained under different clustering algorithms for the same service. At present, many domestic and foreign researchers mainly divide the research of service clustering into two categories according to their focus: Function-based service clustering method and Non-Functional-based service clustering method. Among them, the function-based service clustering method can be summarized from two aspects: one is the data information used in service clustering (such as text and service network topology). The other is represented by Khalid et al. extract features from WSDL documents of services, such as content, service name, host name, and cluster web services [3]. They regard the clustering process as the pre-processing stage of discovery, hoping to help build a search engine to crawl and cluster non-semantic web services. In this paper, tags and description information in service are used to extract information. The other is the method used in clustering (such as keyword matching and topic model), based on the domain classification of services. Zheng et al. [4] proposed a service clustering model, domain service clustering model (DSCM), which was based on probability and fusion of domain characteristics. Based on this model, a topic oriented service clustering method was proposed. The non-functional service clustering method first clusters services according to their functional attributes, and then clusters services according to different quality of service attributes (such as price, availability, response time, reliability and reputation) in each corresponding functional category. Algorithms for non-functional attributes usually have relatively small execution complexity [5], but the non-functional attributes of services are usually difficult to obtain and dynamically change. These algorithms usually do not have good scalability. However, there are many deficiencies in the existing service clustering methods, such as:

a) Most of the existing service clustering methods have certain requirements for the types of service documents (such as OWL-S documents, WSDL documents and other single types of service requirements documents), and most of them use traditional clustering methods (such as K-Means clustering) to get clustering results.

b) Different parameters and initialization of clustering algorithm will have a great impact on clustering results. Most clustering algorithms are difficult to get the number of real clusters in data sets.

c) Different clustering algorithms may produce different clustering results for a unified data set, resulting in poor stability of clustering results. Compared with the existing work, the contributions of this paper are as follows.

We proposed a service clustering method based on the clustering aggregation framework of wisdom crowd.

a) Firstly, we gathered tags information into tag vocabulary and calculated tag similarity matrix, after that, we gathered description into description vocabulary and calculated description similarity matrix. Then, we obtained final similarity matrix base on aggregated tag similarity matrix and description similarity matrix.

b) After that, we employed mapping function to reduce the correlation among data features, in order to satisfy the independence criterion of wisdom crowd.

c) Next, we used different number of clusters in different individual clustering method. it generates high quality clustering results, in order to satisfy the decentralization criterion of wisdom crowd.

d) Then, we adopted diversity evaluated algorithm to calculate diversity of each partition, in order to satisfy the diversity criterion of wisdom crowd.

e) After that, we obtained individual clustering results combining the above steps, and which aggregated different clustering results with the weight of each individual results, and generated aggregation matrix. We clustered the aggregation matrix by Average-Linkage algorithm.

f) Finally, we conducted experiment on web services data that crawled from ProgrammableWeb, and web service is described the WSDL document that is XML-based file, then compared the performance against other individual clustering algorithm and other well-known ensemble clustering method. Experimental results show that the accuracy of SWOC not only surpassed those clustering method, but also at the service clustering recall and F-value in an acceptable runtime.

The rest of this paper is structured as follows. First, in section II we review briefly the related work. Next, in section III, we provide details of the clustering ensemble in the wisdom crowd. In section IV, evaluation standard and experimental results are presented. Finally, conclusions and directions for future works are presented in section V.

II. RELATED CONCEPTS

A. Service Clustering

At present, many domestic and foreign researchers have studied and implemented a variety of clustering algorithms. Ram et al. [6] completed the clustering of services based on the service description of Web Services Description Language(WSDL), so as to improve the efficiency of service discovery. Liu and Wong [7] used feature selection engineering in service description documents, key features reflecting service functions were selected, key features were quantified, and service similarity matrix was calculated to get results. Liu and Yang [8] extracted four features from service description documents, including namely content, context, hostname and service name, and adopted tree traversal algorithm to cluster services, which measured the similarity between content and context by normalizing Google distance. Wang et al. [9] counted the number of occurrences of each word in each service text, which constructed a matrix of words and documents, thus clustering. Rong et al. [10] described web services by using Ontology Web Language for Service (OWL-S), it extracted semantic information in services, which preprocessed semantic web services by fuzzy clustering based on domain ontology.

Huang et al. [11] firstly annotated the name, function and object of service, and used the improved Fuzzy c-means algorithm to cluster service labels. Dorn and Dustdar [12] proposed a K-means algorithm by using tag recommendation strategy to describe Mashup services, which is based on similarity of Mashup services. Shi et al. [13] extracted useful information from service requirements through natural language processing technology, which finally realized service clustering through K-means algorithm. Surowiecki [14] proposed probability and domain characteristics on the domain classification of services. Alizadeh et al. [15] proposed a method of WFCM clustering service using weighted fuzzy C-means (FCM) method. Singh et al. [16] proposed an enhanced LDA model (WE-LDA), which used high-quality word vectors, finally performance of web services was improved the by k-means++ clustering. Min et al. [1] considered multiple web service relationships and adopted improved MR-LDA to cluster services, which improved effectively the accuracy of web service clustering.

The above research only extracted subject words from a single aspect of services (service documents or domain characteristics of services, etc.) for clustering, without considering clustering results from different aspects. In order to solve the above problems, we considered services clustering results from multiple aspects. This paper applied service integration based on wisdom of crowd.

B. Clustering Ensemble

In 2003, Strehl et al. [17] proposed the concept of cluster ensembles which referred to a method of combining multiple partitioning of an object set into a unified clustering result. In 2005, Gionis et al. [18] also gave a description of the problem: given a set of clustering results, until now, the goal of clustering ensemble is to improve the accuracy and robustness of a given classification regression task, and spectacular improvements have been obtained for a wide variety of data sets [19].

The clustering aggregation process is defined as follows:

a) Assume that data set $X = \{X_1, X_2, \cdots, X_n\}$ has n data objects.

b) First use N clustering algorithm for data set X.

c) Get L cluster results $\Pi = \{\pi_1, \pi_2, \cdots, \pi_L\}$, each cluster results is $\pi = \{c_i^1, c_i^2, \cdots, c_i^k\}$, furthermore, $U_{j=1}^{k_i} C_j^i = X$, k_i represents the number of the c_i.

d) Then use the consistency function Γ ensembles the clustering results in Π to get a new data partition $\Pi^{'}$, which is used as the final clustering result.

C. Clustering Ensemble Based on Wisdom Crowd

Surowiecki [14] introduced the concept of wisdom of crowds. WOC illustrates how the prediction performance of a crowd is better than that individual members. As proposed by Alizadeh et al. [15], the concept of wisdom crowd was first applied to the clustering problem, and a clustering model based on wisdom crowd framework was presented. Yousefnezhad et al. [20] presented a framework for unsupervised and semi-supervised cluster ensemble adopted the wisdom of crowd, they employed four conditions in the wisdom of crowd theory, i.e. independence, decentralization, diversity, aggregation, to guide constructing of individual clustering results and final combination for clustering ensemble.

III. Service Clustering Method Based on Wisdom of Crowd

A. Overview of Methodology Framework

In order to improve accuracy of service clustering, we have designed a service clustering framework based on wisdom of crowd. The overview of the framework is shown in Fig.1. The framework composed of two parts: web documents similarity calculation part, and wisdom of crowd.

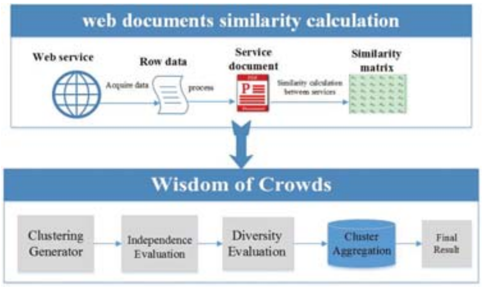

Fig. 1. SWOC the framework

The part of similarity calculated web document, firstly, we crawled the relevant service data from Programmable web with R language tools, next, we gathered some service description and tag information, and computed a service description matrix and tag information matrix, finally, service description similarity and tag information similarity were aggregated.

In this part to satisfy the independence criterion in the wisdom of crowd, we adopted a mapping function that mapped the data to different dimensions. This mapping function can reduce the relevance of data features, which can improve the performance of SWOC method. Next, to satisfy the decentralization criterion in the wisdom of crowd, we employed different kinds of clustering algorithms and different number of clusters. Then to satisfy the diversity in the wisdom of crowd, we adopted diversity algorithm criterion, which can compute the diversity of each partition. Finally, to satisfy the aggregation criterion in the wisdom of crowd, we aggregated different clustering results with the weight of each individual result, and generated aggregation matrix. We clustered the aggregation matrix by Euclidean distance.

The process of the method worked at over real and dynamically changing web services is as follow. Firstly, results of describes services were become different clusters with different labels. Secondly, the service provider published a service, and the services register monitors the published services, then the services register matches the published services with the existed clusters. If the match is successful, the publishing services are belong to the cluster. If the match is fail, builds a new cluster for the published services. Finally, the service customer sends a request to service register by WSDL, the service register obtain the interested information from WSDL and respond the service customer.

B. Independence of Wisdom Crowd

Following the independence condition of the wisdom of crowd theory, the characteristics of the data must satisfy the minimum relevance. However, there is complex topology network relationship between services and service. The accuracy of service clustering will be interfered by the network relationship, so these requires reducing the network relationship. There are many ways to reduce the correlation among data and obtain relatively independent data sets, such as principle component analysis, or linear discriminant analysis. This paper mapped data to different dimensions by utilizing dimension reduction methods, to obtain smaller correlation among service data features.

Given a data set $X = \{X_1, X_2, \cdots, X_n\}$, average the data set $\overline{X}$:

$$\overline{X} = \frac{1}{n}\sum_{i=1}^{n} X_i \qquad (1)$$

Where n represents the number of data in set X; and X_i represents the i^{th} data in service data set X. At this point, you can find $X^{'}$:

$$X^{'} = X - \overline{X} = \{(X_1 - \overline{X}), \cdots, (X_n - \overline{X})\} \qquad (2)$$

Definition Q, $X^{'} \in R^{m\times n} \rightarrow Y \in R^{m\times n}$, where m, n represent the number of service features and data points, respectively, R is an arbitrary real number matrix. The goal of this mapping is minimize the correlation among features, so this problem can be converted as, $Z = Q^T X^{'}$. For R, use the following method to calculate:

$$R = E\{X^{'} X^{'T}\} = \frac{1}{n}\sum_{i=1}^{n} X_i^{'} X_i^{'T} \qquad (3)$$

The data preprocessing algorithm used in this paper is as shown in Algorithm 1:

Algorithm 1: Data independence

```
Input: Data Set $X = \{X_1, X_2, \cdots, X_n\}$;
Output: Data Set to be used $Z$;
a) Use (1) to calculate the average $\overline{X}$;
b) Calculate $X^{'}$ using (2);
c) Generate $R$ using (3);
d) Calculate the eigenvalue $\Lambda$ and eigenvector $Q$ of $R$,
   and sort the eigenvectors in descending order based on
   eigenvalues;
e) Generate $Z$.
```

C. Decentralization of Wisdom Crowd

Following the decentralization of the wisdom of crowd theory condition, the different number of clusters were employed in the different individual clustering method, the different clustering algorithm or different number of clusters represented different person, they clustered services data set and generated different partition about services, their collective solution is likely to be better than any solution single person come up with [18]. There are many methods to generate clustering results, such as:

a) Using different subsets of data.

b) Assigning different parameters to the algorithm.

c) Using different clustering algorithms, as presented in this paper.

A variety of different clustering algorithms were used to generate individual results in tab.1, the individual clustering algorithms were used to satisfy the criterion of decentralization wisdom crowd method. The method improved accuracy of final result, based on decentralization, stability of final result will be increase.

TABLE I
LIST OF INDIVIDUAL ALGORITHMS USED IN SWOC

NO.	Algorihm name
1	K-menas
2	Fuzzy C-means
3	Median K-flats
4	Gaussian mixture
5	Subtract Clustering
6	Single-linkage Euclidean
7	Single-linkage cosine
8	Single-linkage hamming
9	Complete-linkage Euclidean
10	Complete-linkage cosine
11	Complete-linkage hamming
12	Ward-linkage Euclidean
13	Ward-linkage cosine
14	Ward-linkage hamming
15	Average-linkage Euclidean
16	Average-linkage cosine
17	Average-linkage hamming
18	Spectral using a sparse similarity matrix
19	Spectral using Nystrom method with orthogonalization
20	Spectral using Nystrom method without orthogonalization

D. Diversity of Wisdom Crowd

Following the diversity of the wisdom of crowd theory condition, each clustering algorithm has separate clustering result, even if it differs from the facts. We adopted diversity algorithm criterion, which can compute the diversity of each partition. Different clustering algorithms clustered the service data sets and generated different partition. The diversity of wisdom crowd evaluated and selected partition generated by service data sets. In this paper, we considered uniformity of each partition to compute the diversity. The generated individual clustering result was expressed as a reference set, $E = \{P_1, P_2, \cdots, P_T\}$, where T represents the number of individual clustering results, and P_i represents the i^{th} partition (i^{th} clustering result) in the generated result.

This paper finds maximum stability for each partition by considering the number of cluster versus the number of all partitions as follows.

$$\eta_1(P) = \max_{c_i \in P}(n_i \ln(\frac{n_i}{n})) \qquad (4)$$

Where P is a partition from the reference set, c_i is the i^{th} cluster of partition P, n_i and n is the number of cluster and partition. Furthermore, in this paper found maximum stability of each cluster by considering the number of all instances in the partition versus the number of instances in each cluster as follows.

$$\eta_2(P) = \max_{c_i \in P}(n_i \ln(\frac{n}{n_i})) \tag{5}$$

Where the parameters c_i, n_i, and n defined same as the (4). This paper considered the maximum stability of between partitions as follows.

$$\Theta(P, E) = \max_{p_i \in E}(max_{c_j \in P_i}(n_i^j \log(\frac{n_i^j}{n}))) \tag{6}$$

Where P_i represents the i^{th} partitions from the set, c_j represents the j^{th} clusters, n_i^j represents the number of cluster, n denotes the number of partitions. Furthermore, base on the (4), (5) and (6), the finally uniformity value of partition will be computed by (7).

$$Uniformity(P, E) = 1 - \frac{2\eta_2(P)}{3\eta_1(P) + \Theta(P, E)} \tag{7}$$

The weights of clustering algorithms will be computed by the single uniformity versus whole uniformity as (8).

$$\rho_i = \frac{Uniformity(P_i, E)}{\sum_{i=1}^{T} Uniformity(P_i, E)} \tag{8}$$

The diversity algorithm used in this paper is as shown in Algorithm 2:

Algorithm 2: Diversity algorithm

```
Input: generated individual clustering result $E$;
Output: uniformity value of each partition;
a) Use (4) to calculate maximum stability for each
   partition;
b) Use (5) to calculate maximum stability of each cluster;
c) Use (6) to calculate the maximum stability of between
   partitions;
d) Use (7) to calculate the finally uniformity value.
```

E. Aggregation of Wisdom Crowd

Following the aggregation condition of the wisdom of crowd theory: the transformation of their respective clustering results into a mechanism for aggregation results. One of the core issues of clustering integration is how to construct a similarity matrix between data points based on these clustering results obtained by cluster members.

The similarity between data points X_i,X_j is:

$$S_m(X_i, X_j) = \begin{cases} 1 & C(X_i) = C(X_j) \\ 0 & C(X_i) \neq C(X_j) \end{cases} \tag{9}$$

Where $C(X_i) = C(X_j)$ represents X_i,X_j belong to same service, $C(X_i) \neq C(X_j)$ represents X_i,X_j belong to different service.

The weighting similarity matrix was calculated as follows:

$$S_m(X_i, X_j) = \begin{cases} \frac{1}{M}\sum N_{ij} * \rho_{ij} & C(X_i) = C(X_j) \\ 0 & C(X_i) \neq C(X_j) \end{cases} \tag{10}$$

Where M is the number of individual clustering results, and N indicates that the sample i and sample j belong to the same cluster in the M partitions and the value is 1, ρ_i represents the clustering algorithms as weights. When both clustering algorithms have high uniformity, effective results are generated. At the same time, when the two clustering algorithms have smaller values in the uniformity measure, the effect of the generated results is close to zero. Therefore, this paper employed the method, which will ignore the effects of low-quality individuals, instead of selecting generated results through generating thresholds.

In summary, the SWOC algorithm is specifically shown in Algorithm3.

Algorithm 3: SWOC Algorithm

```
Input: Service Feature Data Set $Z$;
Output: The final service clustering result $T$;
a) Different data sets $Z$ are clustered using different
   clustering algorithms. The results of clustering are put
   into a reference set $E$;
b) Use (8) to calculate weights;
c) Use (9), (10) to calculate the weighted similarity
   matrix;
d) The weighted similarity matrix clustered using the
   Average-Linkage algorithm to obtain a service
   clustering result $T$.
```

IV. EXPERIMENTS

A. Experiments Data Sets

In this paper, the relevant web service data was crawled from Programmable web with R language tools, which included service name, service category, service text and tag information, there were 4800 API web services, we selected 317 email API, 289 video news and 263 reviews from the data, Tab.2 shows some of the API service data. Because of the description is too long, it will not be described in the text.

TABLE II
SERVICE DATA

name	tags	category
Yahoo Mail	email	Email
New York Times Movie Reviews	video news reviews	Video
ComplexityIntelligence Named Entity Recognition	tools semantic deadpool	Tools

B. Service Document Similarity Calculation

1) Service document with description similarity calculation: Similarity calculation will be performed on service description. Firstly, the sentences were divided into word, and acquiring feature words. Next, some meaningless words or symbols will be removed, such as '+', '-', 'and', 'but' and so on. Finally, feature words were selected from document by employing tf-idf algorithm,word frequency was calculated on the (11).

$$tf_{ij} = \frac{n_{ij}}{\sum n_{ij}} \tag{11}$$

Where, in the (11), n_{ij} is the j^{th} word of the i^{th} service document with description, $\sum n_{ij}$ is the number of overall words in the service document with description. idf_i represents important measurement of n_{ij}.

$$idf_i = \log(\frac{N}{n_i}) \tag{12}$$

In the (12), N is the number of document description, n_i represents the number of n_{ij} in the description. $w_{ij} = tf_{ij} \times idf_i$, w_{ij} was the product by tf_{ij} and idf_i.

Through the above steps, the weight vector of each description can be calculated. Weight vectors were assumed as $\vec{V_i} = \{w_1, w_2, \cdots, w_{n_i}\}$, $\vec{V_j} = \{w_1, w_2, \cdots, w_{n_j}\}$, similarity between service descriptions were calculated by the cosine angle of two weight vectors, which can be computed as (13).

$$sim_D = \cos\theta = \frac{\vec{V_i} \bullet \vec{V_j}}{|V_i| \bullet |V_j|} = \frac{\sum_{k=1}^{n} \omega_{ki}\omega_{kj}}{\sqrt{\sum_{k=1}^{n} \omega_{ki}^2}\sqrt{\sum_{k=1}^{n} \omega_{kj}^2}} \tag{13}$$

Where, sim_D presents the similarity between weight vectors $\vec{V_i}$, $\vec{V_j}$, θ is between weight vectors $\vec{V_i}$, $\vec{V_j}$ angle.

2) Service document with tag similarity calculation:

Tag information belongs to the service description which can effectively improve service clustering accuracy and query efficiency. Similarity model will be designed by Jaccard coefficient, which is described in (14).

$$sim_{tag}(s_i, s_j) = \frac{|T_i \cap T_j|}{|T_i \cup T_j|} \tag{14}$$

Where, $sim_{tag}(s_i, s_j)$ represents the similarity between vector s_i,s_j, the $|T_i \cap T_j|$ is a intersection of tag information set by s_i and s_j, the $|T_i \cup T_j|$ is a union of tag information set by s_i and s_j.

3) Similarity aggregation:

Base on the (13) and (14), the finally similarity value of service will be computed by (15).

$$sim(s_i, s_j) = sim_D(s_i, s_j) + sim_{tag}(s_i, s_j) \tag{15}$$

C. Evaluation Standard

In this paper, the accuracy index [21], entropy index [22], recall index, F-measure value (F value) [23] and stability were utilized to evaluate the performance of clustering.

Let D be the data set, C is the set of clustering result, $c_k \in C$ denotes the k^{th} cluster in a clustering result, and T is the standard data set, $t_k \in T$ refers to the k^{th} cluster in the standard clustering result, in the cluster, the entropy of the whole service clustering result is:

$$CP(c_k) = \frac{1}{|c_k|} max(|c_k^t|) \tag{16}$$

$$CP(C) = \sum_{k \in C} \frac{|c_k|}{|D|} CP(c_k) \tag{17}$$

Where, $|c_k|$ is the number of c_k cluster, $|c_k^t|$ denotes the number of intersections between the standard cluster and the services, $|D|$ is the number of data set. The entropy of the clustering results can reflect the performance of clustering, the higher the entropy, the better performance of clustering.

The accuracy rate is an important evaluation index. If accuracy of clustering algorithm is higher, the performance of the clustering algorithm is better. Accuracy of clustering algorithm can be calculated by the (18), where the parameters c_k, c_k^t defined as reference.

$$P = \sum_{k \in C} \frac{|c_k^t|}{|c_k|} \tag{18}$$

Recall of each cluster the (19), P_{ij} represents probability that member i is belongs to cluster j, it is calculated by the $P_{ij} = \frac{|c_k^t|}{|c_k|}$.

$$r = -\sum_{k=1}^{L} P_{ij} \log_2 P_{ij} \tag{19}$$

Where the parameters c_k, c_k^t defined as reference, where the parameters L defined the number of total clusters, furthermore, the F-value can be calculated.

$$F = \frac{2(P * r)}{P + r} \tag{20}$$

D. Experimental Result

We used MATLAB R2016a on a PC which generated our experimental results. Each algorithm is run 20 times, the average results are shown in Tab.3 The final clustering results are evaluated by accuracy, entropy, recall and F-value. The SWOC results are compared with some individual clustering algorithms such as K-means, ALE (Average-linkageEuclidean), WLE (Ward-linkage Euclidean), WLC(Ward-linkage cosine), and some well-known ensemble clustering algorithms, such as EAC [24], WPCA [25], GKPC [26], Tab.3, shows the compared results.

In Tab.3 the best results achieved for each clustering algorithms are highlighted. As we can see from the Tab.3, accuracy, entropy, recall and F-value of individual clustering algorithms all lower than result of SWOC. The reason is individual clustering algorithms cannot recognize true patterns in all of data set. And individual clustering algorithms just consider a specification of a dataset for solving the clustering problem [24].

TABLE III
ACCURACY, ENTROPY AND PURITY OF EACH ALGORITHM

Algorithm name	Accuracy(%)	Entropy(%)	recall(%)	F-value
k-means	60.58	60.14	64.73	62.586
ALE	59.52	57.91	62.27	68.714
WLE	61.04	76.45	61.08	61.059
WLC	57.96	51.57	63.56	60.631
EAC	75.32	69.34	62.54	68.733
WPCA	79.45	59.87	64.75	68.283
GKPC	81.01	**89.43**	61.67	70.092
SWOC	**87.90**	84.74	**66.23**	**75.540**

Furthermore, the SWOC outperformed in the accuracy, entropy of SWOC is lower than GKPC ensemble algorithms, entropy of SWOC is higher than other clustering algorithms. In the F-value, SWOC outperformed in individual clustering algorithms and ensemble method. Great majority of results proved superior accuracy for SWOC method. Fig.2 shows the average of accuracy for each clustering method.

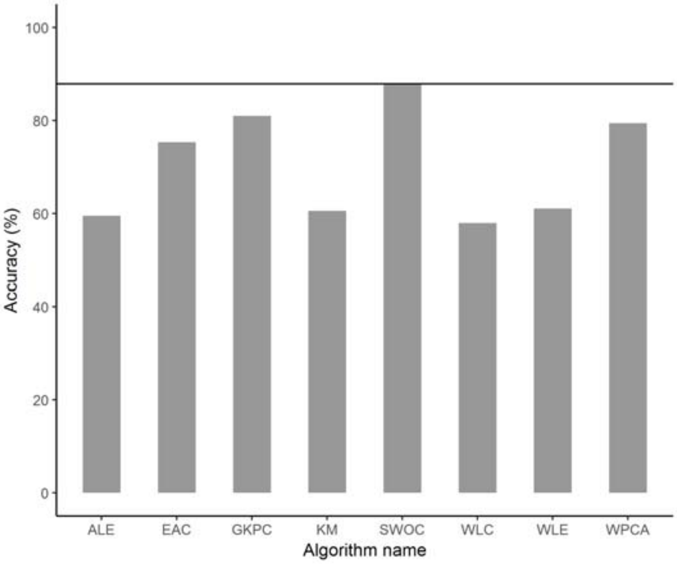

Fig. 2. Average of accuracy for each clustering algorithm

As we can see from the Fig.2, WLC and ALE generated poor results in comparison with other cluster method. They belong to hierarchical clustering, which can identify clusters of complex shapes and solve clustering in non-elliptical datasets, however, they are sensitive to outliers and noise in complex datasets [26]. For classic ensemble method, it did not have the evaluation and selection mechanism, so it cannot filter outliers and find correct information in the process of recognizing patterns. In the Tab.3 and Fig.2 show that accuracy of EAC is affected by evaluation and selection. It vivid proved the importance of four criterions in the SWOC for improving accuracy.

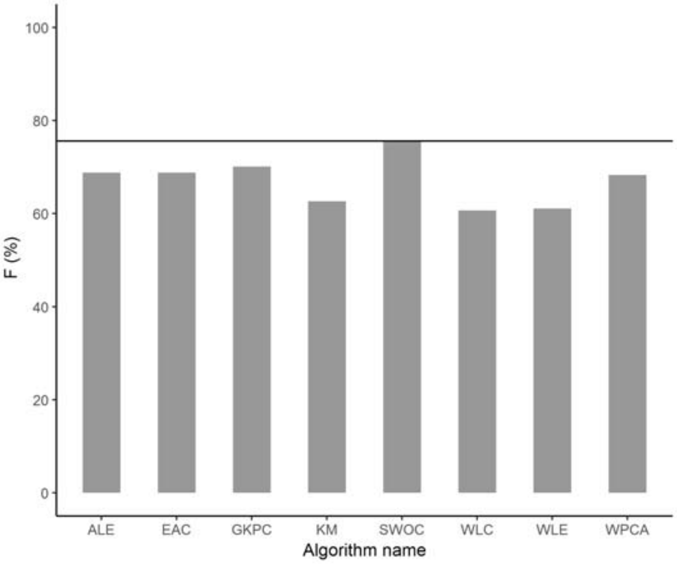

Fig. 3. Average of F-value for each clustering algorithm

As we can see from Fig.3, the F-value of SWOC is outperformed in the services data sets, the F-value of ensemble clustering method is higher than the individual clustering algorithms, It demonstrates that the ensemble clustering method is more stable than individual clustering algorithms. The reason why individual clustering method generate clustering results with global or a local optimizing function is that they did not consider natural relations among data points [15] [27] [28] [29] while diversity criterion of SWOC considers the relations among data points, so SWOC is stable in services data sets.

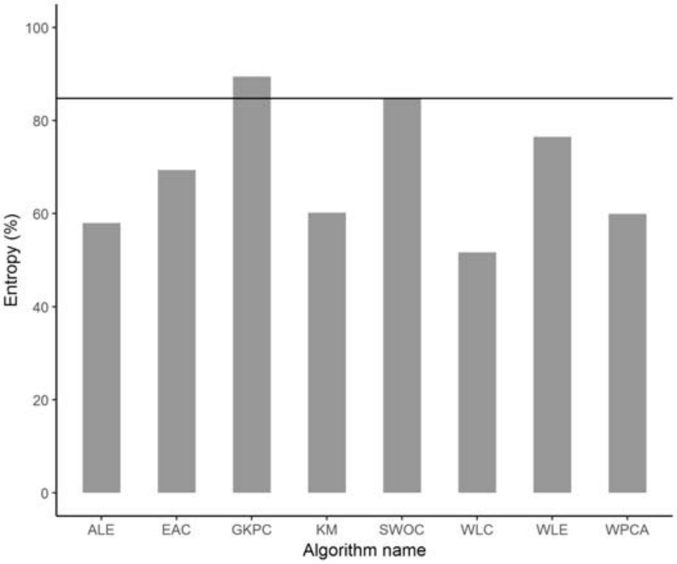

Fig. 4. Average of entropy for each clustering algorithm

As we can see from Fig.4, the entropy of SWOC is higher than other ensemble clustering algorithms. However, the entropy of GKPC is higher than SWOC. This can be explained by GKPC adopting co-association matrix as a similarity measure between objects, in the sense that it integrates information from the original data of object representations [30].

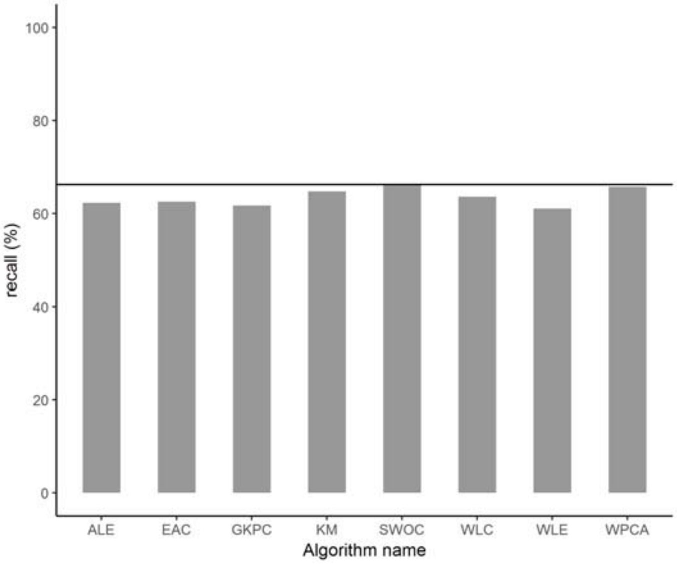

Fig. 5. Average of recall for each clustering algorithm

As we can see from Fig.5, it is difficult to distinguish the results of WPCA and SWOC in the recall, however, the average 20 times performance in service that SOWC outperformed WPCA by over 1.57%. Recall of SWOC is higher than other clustering algorithms, furthermore, the height of each column is very consistent, because of semantic analysis is common in tag similarity calculation and description similarity calculation.

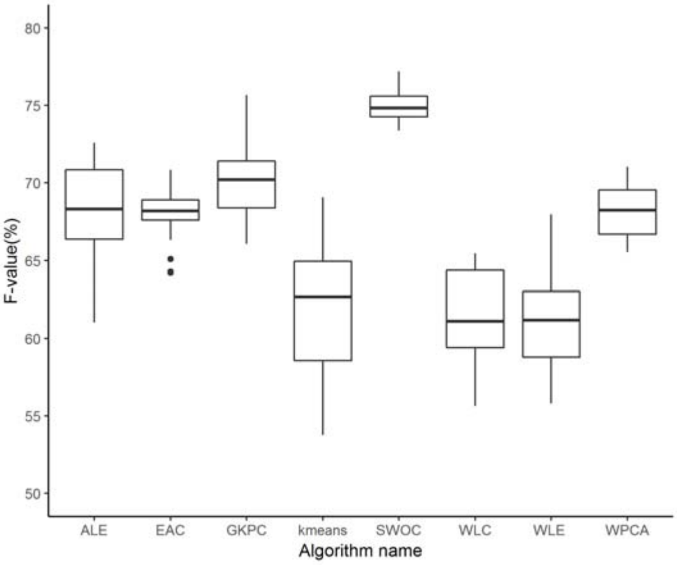

Fig. 6. Average of stability for each clustering algorithm

As we can see from Fig.6, the boxplot shows the individual clustering algorithm and ensemble clustering method. The area of each algorithm represents the stability of each algorithm. The smaller the area, the algorithm is more stable. In the individual algorithms, the area of k-means is larger than other algorithms which can be explained by the initial cluster centers randomness and sensitivity to noise and outliers [22]. ALE, WLC and WLE are relatively stable. In the ensemble clustering method, the area of SWOC is the smallest which shows the stability of SWOC algorithms. For EAC ensemble method, there are two outliers points, it bases on the voting machine [25], it does not consider evaluation and selection of base clustering results. The area of GKPC and WPCA are larger than the EAC. WPCA is weight of principal components analysis, which is a method that simplifies data sets [19], it is a linear transformation and it only considered the linear relationship between the data while the connection between the data points was not considered enough. The independence, the dispersion, the diversity and aggregation criterion are fully considered for data relevance, evaluation and selection of individual clustering algorithm.

E. Time Complexity Analysis

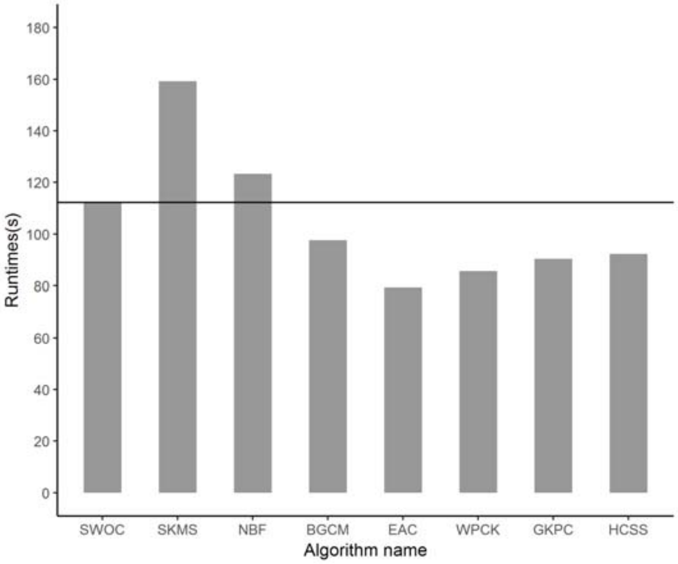

Fig. 7. Runtimes analysis

In this section, runtime of SWOC compared with semi-supervised and unsupervised method. As we can see from the Fig.7, the runtime of semi-supervised algorithm(the first four bars) is more than the runtimes of unsupervised algorithm (after four bars), since the semi-supervised algorithms need apply the semi-supervised information to guide clustering [31]. In this paper, SWOC is belong to semi-supervised algorithm, so the runtimes of SWOC is more than runtimes of unsupervised algorithm. In semi-supervised algorithms, the runtimes of SWOC is more than the runtimes of BGCM, and the runtimes of SWOC is less than the runtimes of SKMS and NBF.

On the one hand, since SWOC adopted the algorithm1 to increase the data independence by calculated the eigenvalue/vector, which can reduce the time complexity of the mapping function in algorithm1, on the other hand, the proposed method used weighting similarity matrix (10) to add the semi-supervised information, and limited the size of pairwise constraints. The size of weighting similarity matrix is small in compare with the size of instances; e.g., the size of data set from programmable is 869×869, the size of sampled pairwise constraints is 634×634.

Notably, SKMS employed pairwise constraints as semi-supervised information to guided clustering procedure, and the points are mapped to kernel space that is a high dimensional space [32], NBF consider active learning in an iterative manner, the method of active learning considers that extends the neighborhoods by selecting informative points and inquiring their relationship around the neighborhoods [33]. So the performance of SWOC is well in an acceptable runtime.

V. CONCLUSION

This paper proposed a clustering method based on the wisdom of crowds. We adopted ensemble clusters method about core ideal of wisdom crowd which as a collective solution is likely to be better than single solution. Web services were clustered by this method, the accuracy and stability of cluster method worked on web services are improved, solved the problem that accuracy and stability of single cluster algorithm or traditional cluster algorithm worked on web services is poor.

In the future, It is not full that the SWOC explores the relationships between data points, and we will further explore the potential information between data points and improve the accuracy of service clustering results. We will consider to adopt parallelization or distributed computing in larger data set.

ACKNOWLEDGMENT

This work is supported by the International Science and Technology Cooperation Program of the Science and Technology Department of Shaanxi Province, China (Grant No.2018KW-049), the Special Scientific Research Program of the Education Department of Shaanxi Province,China(Grant No.17JK0711), the National Natural Science Foundation of China(61702414) and Natural Science Basic Research Program of Shaanxi (Grant No.2018JQ6078).

REFERENCES

[1] S. Min, L. Jianxun, Z. Dong, C. Buqing, and W. Yiy, "Web service clustering method based on multiple relational topic model," *Chinese Journal of Computers*, pp. 1–16, 2018.

[2] J. Bo, Y. Lingyao, P. Weifeng, and W. Jialei, "Service clustering method based on demand function semantics," *Chinese Journal of Computers*, vol. 41, no. 6, pp. 1035–1046, 2018.

[3] K. Elgazzar, A. E. Hassan, and P. Martin, "Clustering wsdl documents to bootstrap the discovery of web services," in *IEEE International Conference on Web Services*, 2010, pp. 147–154.

[4] L. Zheng, H. Ke-Qing, W. Jian, and Z. Neng, "An on-demand services discovery approach based on topic clustering," *Journal of Internet Technology*, vol. 15, no. 4, pp. 543–555, 2014.

[5] C. Platzer, F. Rosenberg, and S. Dustdar, *Web Service Clustering using Multidimensional Angles as Proximity Measures*. ACM, 2010.

[6] S. Ram, Y. Hwang, and H. Zhao, "A clustering based approach for facilitating semantic web service discovery," *Social Science Electronic Publishing*, 2005.

[7] W. Liu and W. Wong, "Web service clustering using text mining techniques," *International Journal of Agent-Oriented Software Engineering*, vol. 3, no. 1, pp. 6–26, 2009.

[8] Y. S. Liu and Y. C. Yang, "Semantic web service discovery based on text clustering and similarity of concepts," *Computer Science*, vol. 40, no. 11, pp. 211–214, 2013.

[9] Y. M. Wang, Y. J. Zhang, B. H. Xie, L. H. Pan, and L. C. Chen, "Semantic web service discovery based on fuzzy clustering optimization," *Computer Engineering*, vol. 39, no. 7, pp. 219–223, 2013.

[10] Y. Y. Li Rong, Ye Junmin, "A tag based hierarchical web service clustering method," *Computer Era*, no. 11, pp. 30–34, 2017.

[11] H. Yuan, L. Bing, and e. a. He Peng, "Clustering of mashup services based on label recommendations," *Journal of Computer Science*, vol. 40, no. 2, pp. 167–171, 2013.

[12] C. Dorn, "Weighted fuzzy clustering for capability-driven service aggregation," *Service Oriented Computing & Applications*, vol. 6, no. 2, pp. 83–98, 2012.

[13] M. Shi, J. Liu, D. Zhou, M. Tang, B. Cao, M. Shi, J. Liu, D. Zhou, M. Tang, and B. Cao, "We-lda: A word embeddings augmented lda model for web services clustering," in *IEEE International Conference on Web Services*, 2017, pp. 9–16.

[14] E. A. Mennis, "The wisdom of crowds: Why the many are smarter than the few and how collective wisdom shapes business, economies, societies, and nations," *Personnel Psychology*, vol. 59, no. 4, pp. 982–985, 2010.

[15] H. Alizadeh, M. Yousefnezhad, and B. M. Bidgoli, "Wisdom of crowds cluster ensemble," *Intelligent Data Analysis*, vol. 19, no. 3, 2015.

[16] V. K. Singh, R. Jalan, S. K. Chaturvedi, and A. K. Gupta, "Collective intelligence based computational approach to web intelligence," in *International Conference on Web Information Systems and Mining*, 2009, pp. 27–31.

[17] A. Strehl and J. Ghosh, "Cluster ensembles – a knowledge reuse framework for combining multiple partitions," *Journal of Machine Learning Research*, vol. 3, no. 3, pp. 583–617, 2002.

[18] A. Gionis, H. Mannila, and P. Tsaparas, "Clustering aggregation," in *International Conference on Data Engineering, 2005. ICDE 2005. Proceedings*, 2005, pp. 341–352.

[19] A. L. N. Fred, Jain, and A. K, "Combining multiple clusterings using evidence accumulation," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 27, no. 6, pp. 835–850, 2005.

[20] M. Yousefnezhad, S. J. Huang, and D. Zhang, "Woce: A framework for clustering ensemble by exploiting the wisdom of crowds theory," *IEEE Transactions on Cybernetics*, vol. PP, no. 99, pp. 1–14, 2017.

[21] K. V. Tan P N, Steinbach M, *Introduction to Data Mining*. Addison-Wesley Longman Publishing Co., Inc., 2005.

[22] Z. Y, *Criterion functions for document clustering experiments and analysis*. University of Minnesota, 2005.

[23] M. J, "Improved k-means algorithm in text semantic clustering," *The Open Cybernetics & Systemics Journal*, vol. 8, no. 1, pp. 530–534, 2014.

[24] H. Alizadeh, H. Parvin, M. Moshki, and B. Minaei, *A New Clustering Ensemble Framework*. Springer Berlin Heidelberg, 2011.

[25] X. Bai and R. Qiuqi, "An improved wpca plus lda," in *International Conference on Signal Processing*, 2007, pp. 482–489.

[26] S. Vega-Pons, J. Ruiz-Shulcloper, and A. Guerra-Gandn, "Weighted association based methods for the combination of heterogeneous partitions," *Pattern Recognition Letters*, vol. 32, no. 16, pp. 2163–2170, 2011.

[27] A. Fred and A. Loureno, "Cluster ensemble methods: from single clusterings to combined solutions," vol. 126, pp. 3–30, 2008.

[28] A. K. Jain, A. Topchy, M. H. C. Law, and J. M. Buhmann, "Landscape of clustering algorithms," in *International Conference on Pattern Recognition*, 2004, pp. 260–263.

[29] E. A. Mennis, "The wisdom of crowds: Why the many are smarter than the few and how collective wisdom shapes business, economics, societies and nations," *Personnel Psychology*, vol. 59, no. 4, pp. 982–985, 2010.

[30] A. Topchy, A. K. Jain, and W. Punch, "Combining multiple weak clusterings," in *IEEE International Conference on Data Mining*, 2003, pp. 331–338.

[31] J. Gao, F. Liang, F. Wei, Y. Sun, and J. Han, "A graph-based consensus maximization approach for combining multiple supervised and unsupervised models," *IEEE Transactions on Knowledge & Data Engineering*, vol. 25, no. 1, pp. 15–28, 2013.

[32] S. Anand, S. Mittal, O. Tuzel, and P. Meer, “Semi-supervised kernel mean shift clustering,” *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 36, no. 6, pp. 1201–15, 2013.

[33] S. Xiong, J. Azimi, and X. Z. Fern, “Active learning of constraints for semi-supervised clustering,” *IEEE Transactions on Knowledge & Data Engineering*, vol. 26, no. 1, pp. 43–54, 2014.