Fragment-based Clustering Ensembles

Ou Wu, Mingliang Zhu, Weiming Hu
National Laboratory of Pattern Recognition, Institute of Automation
Chinese Academy of Sciences
95 Zhongguancun East, Beijing
{wuou, mlzhu, wmhu}@nlpr.ia.ac.cn

ABSTRACT

Clustering ensembles combine different clustering solutions into a single robust and stable one. Most of existing methods become highly time-consuming when the data size turns to large. In this paper, we study the properties of the defined 'clustering fragment' and put forward a useful proposition. Solid proofs are presented with two widely used goodness measures for clustering ensembles. Finally, a new ensemble framework termed as fragment-based clustering ensembles is proposed. Theoretically, most of existing methods can be improved by adopting this framework. To evaluate the proposed framework, three new methods are introduced by bring three popular clustering ensemble methods into our framework. The experimental results on several public data sets show that the three introduced methods are greatly improved in computational complexity and also achieved better or similar accurate results than the original methods.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Applications – *Data mining*; 1.5.3 [**Pattern Recognition**]: *Clustering – Algorithms*

General Terms

Algorithms, Experimentation, Theory

Keywords

Clustering Ensembles, Fragment, Mutual Information

1. INTRODUCTION

Clustering ensembles, also known as consensus clustering or clustering aggregation, have emerged as a powerful method to combine multiple inconsistent clustering solutions. Many applications arisen in various settings and from different disciplines can be transferred into the problem of clustering ensembles. Some typical applications are: categorical data clustering, heterogeneous data clustering, outlier detection, distributed clustering, knowledge reuse and aggregating clusterings of different methods [1]. There have been a great number of clustering aggregation methods in the literature, which can be roughly classified into: voting based methods [2], graph-based methods [3], mixture model based methods [4] and searching based methods [1].

We note that the computational complexity of clustering ensembles usually depends on the data size. Many existing clustering ensemble methods are quadratic of the data size. With the data size increasing, these methods become highly time consuming and are unable to be applied in real applications. This reminds us that if the data size is decreased before employing the ensemble algorithm, the time consumption can be reduced.

Each clustering can be viewed as a partition of the original data. We find that the whole data set consists of data subsets (named as 'clustering fragment') in which data points keep together in all of the input clustering. We call such subsets as 'clustering fragments (or fragments for simplification)'. In many clustering ensemble problems, the number of produced fragments is far smaller that the original data size. Naturally, an interesting question is arising: can we solve the clustering ensemble problem merely based on the fragments? If the answer is 'YES', the computational complexity of aggregation on fragments should be much lower than that of existing methods. We propose a useful proposition to answer the question. The proposition is proved within two widely used goodness measures. Then a fragment-based ensemble framework is introduced. Experimental results demonstrate that the new methods which are based on the proposed framework outperform the original methods significantly in time complexity and achieve better (or similar) results than the original ones.

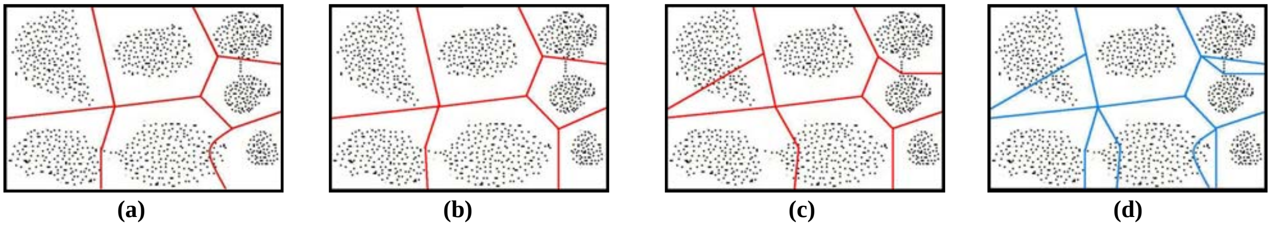
The remainder of this paper is organized as follows. Section 2 states the clustering fragment extraction algorithm. Section 3 provides our main theory and corresponding proof. Section 4 introduces the fragment based clustering ensemble framework and proposes three new methods by modifying existing ones. Section 5 gives our experimental evaluations on several public data sets and some discussions. Conclusions are given in Section 6.

2. EXTRACTION ALGORITHM

This section introduces the 'clustering fragments' extraction algorithm. Some symbols used in the paper are defined as follows.

Let $X = \{x_1, x_2, \dots, x_n\}$ denote a set of data points. $\Pi = \{\pi_1, \pi_2, \dots, \pi_H\}$ be a set of H input partitions of X. Each partition indicates a clustering and $\pi_i (x_j)$ denotes the label assigned to x_j by the i-th partition. Let $|\pi_i |$ be the number of clusters given by the i-th partition. We denote the cluster set as C_i with respect to the i-th partition π_i. Then C_{ij} means the j-th cluster obtained by the i-th partition. Let F represent the set of fragments.

CIKM'09, November 2–6, 2009, Hong Kong, China.
Copyright 2009 ACM 978-1-60558-512-3/09/11…$11.00.



(a) (b) (c) (d)

Figure 1. (a), (b) and (c) are three different partitions and (d) is the union of all the three partitions.

For each data point, its labels given by the partitions Π form a label sequence, for example, '$\pi_1(x_j)\ \pi_2(x_j)\ldots\pi_H(x_j)$'. We are able to get each point's label sequence by accessing each partition once. If suitable data structure is used, the time complexity is $O(n*H)$, where N represents the number of produced fragments. Table 1 shows the pseudo code of the extraction algorithm.

In Fig. 1, there are approximately 1000 data points. Three different partitions achieved by different clustering algorithms are given in Fig.1 (a), (b) and (c). Fig.1 (d) shows the produced fragments which are enclosed by blue lines and borders. The number of fragments is 11 which are far less than the data size 1000. It can be easily observed that the points in the same fragment keep together in each partition in Fig.1 (a), (b) and (c).

3. THE MAIN THEORY AND PROOF

The previous section has introduced the clustering fragments' extraction algorithm. This section presents our theory about whether the clustering aggregation can be achieved directly on fragments. We give proof for our theory under two widely used goodness measures in clustering ensembles. The first is the distance criterion proposed by Ginois [1]. The distance is used to measure the disagreement between two clustering. With this criterion, clustering ensemble becomes an optimization problem to find a new partition with the minimized total distance. Studies in [5, 7] are based on the distance criterion. The second measure is the mutual information criterion. This criterion applies mutual information to measure the agreement between two clustering. Studies in [2, 6] apply this criterion.

Table 1. Pseudo code of fragment extraction

```
Input: $X = \{x_1, x_2, \ldots, x_n\}$, $\Pi = \{\pi_1, \pi_2, \ldots, \pi_H\}$
Output: Fragments
Steps:
Initialization: string $Ls(j) = \phi$, $i = 1 : n$;
map<string, list<integer>> $F$;
for each partition $\pi_i$ do
    for each data point $x_j$ do
        $Ls(j) = Ls(j) \cup \pi_i(x_j)$
    end for
end for
for $j = 1 : n$
    if ($l = F$.find($Ls(j)$)) == null
        generate a new integral list and insert $j$ into the new list;
          then insert the $Ls(j)$ and the new list into $F$.
    else
        insert $j$ into list $l$.
end for
return $F$.
```

3.1 The Main Theory

Our theory can be summarized as the following proposition.

Proposition: *Let π^* be the optimal partition via clustering aggregation. All the data points located in the same clustering fragment definitely share the same label in π^*, or saying they are definitely in the same cluster of π^*.*

Intuitively, the above proposition is reasonably true. However, without a universal goodness measures for any candidate partition, it is difficult to give a direct proof. We note that most existing methods aim to optimize the distance criterion or mutual information criterion. If the proposition is proved to be true under these two measures, our new framework is adaptable to most of existing methods. We then introduce two theorems.

Theorem 1. *Let π^* be the optimal partition under the distance criterion, the proposition is true.*

Theorem 2. *Let π^* be the optimal partition under the mutual information criterion, the proposition is true.*

The following subsection will give the proof.

3.2 The Proof

Due to lack of space, we only give the proof sketch (Detailed steps are given in the full version of this paper).

3.2.1 The Proof of Theorem 1

Assuming that, on the contrary, we get a candidate partition π^a and that there exists at least one fragment whose points scatter in different clusters of π^a. We denote one of the scattered fragments as F_s and the point subset of F_s in the m-th cluster of π^a as $X_s^{(m)}$.

Now we focus on two arbitrary clusters $m1$ and $m2$ given by π^a. We introduce the following Lemma using the above assumptions.

Lemma 3. *The goodness of π^a can be increased through one of the two operations: 1) transfer data in $X_s^{(m1)}$ into cluster m2 or 2) transfer data in $X_s^{(m2)}$ into cluster m1.*

Proof. Please refer to the full version of this paper.

Lemma 4. *When using distance criterion to measure the goodness of a candidate partition, there never exists an optimal partition in which points from the same fragment are scattered in different clusters.*

Proof. Assuming there is an optimal partition π^* which places points of a fragment into different clusters. According to Lemma

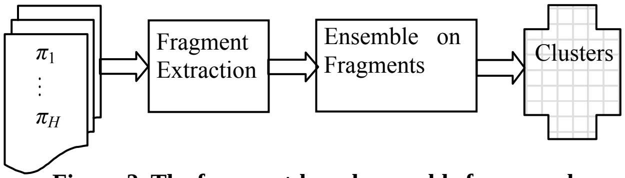


Figure 2. The fragment-based ensemble framework

3, the goodness of π^* can be increased, which indicates that π^* is not an optimal partition. □

As a result, Theorem 1 is true according to Lemma 4.

3.2.2 The Proof of Theorem 2

We still assume that there is a candidate partition (π^a) as the same as the one in previous subsection. It has two variations: π' and π''. One (π') is produced by transferring data in $X_s^{(m1)}$ into m2; the other (π'') is produced by transferring data in $X_s^{(m2)}$ into m1.

Lemma 5. *Without loss of generality, we assume $\Phi(\pi'') \geq \Phi(\pi')$. In this case, $\Phi(\pi'') > \Phi(\pi^a)$.*

Proof. Please refer to the full version of this paper.

Lemma 6. *When using mutual information criterion, there never exists an optimal partition in which points from the same fragment are scattered in different clusters.*

Proof. Assuming there is an optimal partition π^* which places points of a fragment into different clusters. According to Lemma 5, the goodness of π^* can be increased, which indicates that π^* is not an optimal partition. □

As a result, Theorem 2 is true according to Lemma 6.

4. FRAGMENT-BASED CLUSTERING ENSEMBLES

Intuitively, Proposition 1 will hold under any goodness measures. However, Theorem 1 and Theorem 2 are still very useful due to that most current methods are based on these two measures. Since the optimal partition is a combination of the fragments, the optimal partition can be deduced directly by managing points of a fragment as a whole. This approach can be summarized as the fragment-based clustering ensemble framework shown in Figure 2.

We present three new methods by modifying three existing typical ones to evaluate the proposed framework in the paper: Agglomerative, Furthest and Local Search (Their details can refer to [1]). To differ from the original methods, the terms of the new ones are added by the prefix 'F-'. We only take F-Agglomerative as an example to illustrate the implementation details shown in Table 2 due to limited space. The other two methods (F-Furthest and F-Local Search) have the similar procedures and can be found in the full version of the paper. The complexity of Agglomerative is $O(n^2H) + O(n^2\log n)$. Thus the complexity of F-Agglomerative is $O(nH)$ for fragment extraction and $O(N^2H) + O(N^2\log N)$ for running the aggregation method. It is obvious that the complexity can be greatly reduced if $N << n$. This conclusion is still available for F-Furthest and F-Local Search.

Table 2. Steps of F-Agglomerative

Input : $X = \{x_1, x_2, \ldots, x_n\}$, $\Pi = \{\pi_1, \pi_2, \ldots, \pi_H\}$
Output: clusters
Steps:
(1) Extract fragments using the algorithm in Table 1;
(2) Place each fragment in a single cluster;
(3) Calculate the average distance between each pair of clusters and choose the smallest average distance and the corresponding pairs of clusters;
(4) Merge the corresponding clusters and go to (3) if the smallest distance is below 0.5. Otherwise, go to next step.
(5) Output the current obtained clusters.

5. EXPERIMENTS

This section reports the results of the proposed fragment-based ensemble methods as well as the original ones on six public data sets from UCI Repository [8]: Hayes-Roth (#1), Glass (#2), Breast (#3), Yeast (#4), Wave (#5) and Magic (#6).

We employ running time as well as both the classification error E_c and disagreement error E_d defined in [1] to evaluate results of each method. E_c is defined as:

$$E_c = \frac{\sum_{i=1}^{|\pi|}(|C_{\pi,i}| - m_i)}{n}$$

where m_i denotes the size of the majority class in cluster $C_{\pi,i}$.

Because Hayes-Roth is a category data set, the input partitions are generated according to each of its attribute. For other sets, the input partitions are obtained by using K-means algorithm with different initialized number of centers.

Figure 3-5 show the running time of the three pairs of algorithms over the six data sets. It can be observed when the size is 160, the running time of original methods nearly equals to that of fragment-based ones. The main reason is that fragment-based methods need to extract fragments and the time-consumption of fragment extraction can not be ignored compared with the following fragment ensemble when the size is 160. However, with the data size increasing; the time-consumption of fragment extraction occupies little proportion. As a consequence, the running time of original methods increase sharply while that of the fragment-based methods maintains very small.

Figure 6-8 show the performance comparison between the three pairs of algorithms respectively in terms of E_c. It can be observed that the fragment-based methods achieve comparable or better results than the original ones through the six data sets except the Wave set (#5). For the Wave set, both F-Agglomerative and F-Furthest are inferior to their original methods. The discussion part will give an analysis.

Table 3 shows the performance comparing between the three pairs of algorithms respectively in terms of E_d. We can observe that in most cases, the fragment-based methods yields lower error. Three exceptional cases are bold and italic-represented.

The experimental results over the six data sets show the satisfactory performance of the fragment-based ensemble methods, which is consist with our theoretical analysis in time complexity of fragment-based framework. We observe that the running time of fragment based methods mainly depends on the number of fragments and appear to be insensitive to the size of point set.

Table 3. E_d of different methods over the six data sets.

Method	**#1**($\times10^3$)	**#2**($\times10^3$)	**#3**($\times10^5$)	**#4**($\times10^5$)	**#5**($\times10^6$)	**#6**($\times10^8$)
Agglomerative	***9.04***	5.35	1.39	6.28	8.23	4.130
F- Agglomerative	***9.41***	5.27	1.35	6.20	8.20	4.130
Furthest	12.94	6.26	1.38	22.16	***8.50***	4.496
F-Furthest	9.45	5.27	1.35	8.17	***12.21***	4.247
LocalSearch	8.92	5.17	1.42	***6.05***	8.40	4.135
F- LocalSearch	8.92	5.17	1.35	***6.07***	8.16	4.127

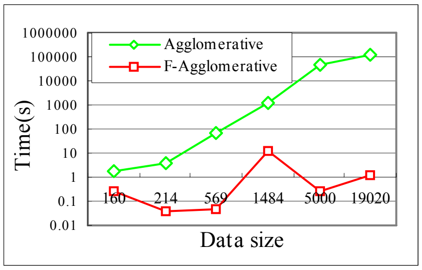


Figure 3. Running time

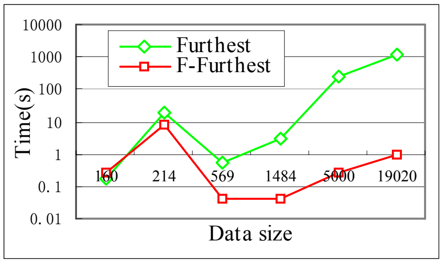


Figure 4. Running time

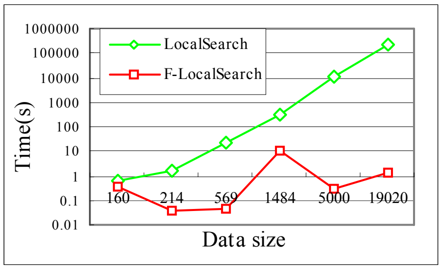


Figure 5. Running time

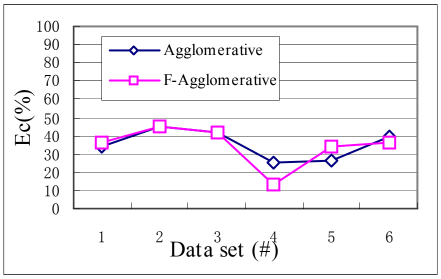


Figure 6. E_c

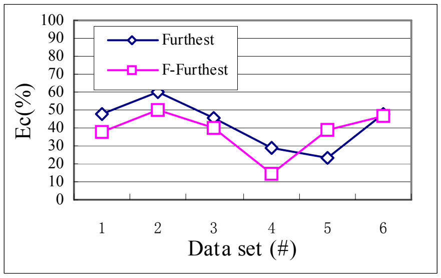


Figure 7. E_c

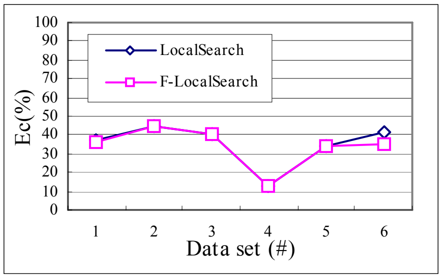


Figure 8. E_c

6. CONCLUSIONS

This paper defines the 'clustering fragment' and studies its useful properties. Based on the clustering fragments, we propose a new clustering ensembles framework: fragment-based clustering ensembles. This framework is based on the proposition that an optimal partition should ensure the data points of a fragment locate in the same cluster. We have proved this proposition under two widely used goodness measures: distance measure and mutual information measure. Because the size of fragment set is usually far smaller than the data size, existing methods can be improved with respect to the time complexity. Theoretically, most of existing methods can be improved by bring into this framework. To utilize the efficiency of the proposed framework, three new ensemble methods are presented, i.e. F-Agglomerative, F-Furthest and F-LocalSearch. We conducted experiments on six public data sets. The results show that the three new methods significantly outperform their original methods in terms of running time, which demonstrates the efficiency of our framework.

7. ACKNOWLEDGMENTS

We are very appreciative to Dr. Hanzi Wang for his revision in writing. We would like to thank the anonymous reviewers for their very useful comments and suggestions. This work is partly supported y NSFC (Grant No.60672040 and 60672040).

8. REFERENCES

[1]Gionis, A., Mannila, H. and Tsaparas, P. 2007. Clustering aggregation. ACM TKDD, Vol. 1, No. 1, pp. 1–30.

[2]Ayad, H.G. and Kamel, M.S. 2008. Cumulative Voting Consensus Method for Partitions with Variable Number of Clusters, IEEE TPAMI, Vol. 30, No. 1, pp. 160-173.

[3]Fern, X.Z. and Brodley, C.E., 2004. Solving cluster ensemble problems by bipartite graph partitioning, In Proc. of ICML, pp.36, July, Alberta, Canada.

[4]Lange, T. and Buhmann, J., 2005. Combining partitions by probabilistic label aggregation. In Proc. of ACM KDD, pp.147-156.

[5]Li, T. and Ding, C., 2008. Weighted consensus clustering. In Proc. of SDM, pp.798-808.

[6] Zhou, Z.H. and W. Tang., 2006. Clusterer ensemble. Knowledge-Based Systems, 2006,19(1), pp.77-83.

[7]Bahloul, S.N., Rouba, B. and Amghar, Y. 2008. Minimization of the Disagreements in Clustering Aggregation. In Proc. of ICIC, pp. 517–524.

[8] http://archive.ics.uci.edu.