INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, e-ISSN 1841-9844, 14(2), 170-182, April 2019.

Heterogeneous Data Clustering Considering Multiple User-provided Constraints

Yue Huang*
School of Information Science
Beijing Language and Culture University, Beijing, China
*Corresponding author: huang.yuet@blcu.edu.cn

Abstract: Clustering on heterogeneous networks which consist of multi-typed objects and links has proved to be a useful technique in many scenarios. Although numerous clustering methods have achieved remarkable success, current clustering methods for heterogeneous networks tend to consider only internal information of the dataset. In order to utilize background domain knowledge, we propose a general framework for clustering heterogeneous data considering multiple user-provided constrains. Specifically, we summarize that three types of manual constraints on the object can be used to guide the clustering process. Then we propose the User-HeteClus algorithm to solve the key issues in the case of star-structure heterogeneous data, which incorporating the user constraint into similarity measurement between central objects. Experiments on a real-world dataset show the effectiveness of the proposed algorithm.
Keywords: clustering, heterogeneous networks, relational data, multi-typed objects, user constraints.

1 Introduction

With the advent of "big data", data mining has become a widely accepted tool for data analysis and various data mining-related research appeared [4,5,25,26]. Among all the techniques of data mining, clustering presents an effective way of exploring data, especially scenarios with no available labelled data. Compared with homogeneous networks, heterogeneous information networks [20] which consist of different types of objects and links can be found in many actual scenarios. Clustering on heterogeneous data has become a key emerging challenge during the recent twenty years [21, 22, 24].

In earlier study of clustering, we only have to deal with objects of the same type and numerous methods have been proposed [10]. Later, clustering on data with two different types of objects emerged, such as two-way clustering [9], co-clustering [1, 3, 6, 7], and bi-clustering [15]. Recently, more attention has been paid to multi-way clustering [2], also called high-order co-clustering [8]. Generally, in the datasets with real heterogeneity [13], the object type that contains less number of distinct values among heterogeneous data are called central type or target type, while the other types are called attribute types [19]. Whereas, most existing research focus on the star-structure of heterogeneous networks [11, 14, 18, 20], where links only exist between objects of the central type and objects of the attribute types, representing many scenarios in the real world. Besides, several methods have also been put forward for other types of heterogeneous networks [12] or arbitrary heterogeneous networks [16, 17].

In some scenarios, it is impossible to perform effective cluster analysis without taking advantage of the external information of the dataset, such as domain knowledge provided by users. Taking document analysis as an example, for a heterogeneous dataset containing tens of thousands of documents, thousands of words, and several document clusters, if the information of cluster assignment of one hundred pairs of documents is provided, then the clustering efficiency

on the documents can be improved. To solve the issue of clustering heterogeneous data considering user-provided constraints, we firstly analyzed that there are three types of constraint information provided by the user to help the clustering process of the objects. Then we propose a complete general analysis framework, based on which we propose corresponding solutions to solve the key issues in the case of star-structure heterogeneous data.

The rest of this paper is organized as follows. Section 2 defines the research scope of the problem and analyzes the hypotheses. Section 3 first proposes a general framework for solving the heterogeneous data clustering analysis that takes into account user-provided constraints, then it analyzes and resolves the key issues with the proposed algorithm UserHeteClus for heterogeneous data clustering considering multiple user-provided constraints. Section 4 gives the experimental results and analysis. Conclusion is given in Section 5.

2 Problem definition and hypothesis

Similar to previous study on semi-supervised clustering [23], it is easy to find that in heterogeneous data, user-provided constraints can be of three types: (1) the user labels which central objects should belong to (or do not belong to) the same cluster; (2) the user indicates which central objects with which attribute values must belong to (or not belong to) the same cluster; (3) the user indicates which attribute values actually correspond to the same (or different) meanings.

When the user can determine the cluster attribution of the central object by providing the values of the central object for one or some attribute objects, it indicates that the user provides a decisive attribute object. In this case, assigning the object cluster under such a constraint condition is similar to classifying central objects, requiring the user to provide very precise knowledge, so the condition is clearly too high to meet. According to previous studies and applications of semi-supervised clustering, what the user generally provides is information about the relationships between objects, but in practical applications, the type of constraint that the user may provide is not limited to this, so we have the following hypotheses:

Hypothesis 1: In the problem of heterogeneous data clustering considering user-provided constraints, the constraints may be on the central object or on the attribute object.

Hypothesis 2: In the problem of heterogeneous data clustering considering user-provided constraints, it is assumed that the heterogeneous dataset has a star structure.

3 Key issues and steps of the UserHeteClus

In this section, we propose the UserHeteClus algorithm for solving the key issues in the procedure of clustering star-structured heterogeneous data.

3.1 Framework for clustering heterogeneous data considering user constraints

The information available in the analysis of heterogeneous data with a star structure considering user-provided constraints includes internal information (object type information, object attribute information, and object relation information) and external information.

In the clustering of heterogeneous data with a star structure considering user-provided constraints, on the one hand, the composition of the dataset itself is very complex; on the other hand, it is necessary to integrate external information. Therefore, comprehensively, in the semi-supervised clustering of heterogeneous data considering user-provided constraints, we still need to follow the principle of "paying more attention to relations and less to attributes" [12], and it is necessary to separate the attribute similarity computation from object clustering.

In addition, relative to unsupervised clustering, semi-supervised clustering still has a small amount of useful sample information. Therefore, how to effectively utilize the domain knowledge contained in the labeled samples to guide the clustering process in the clustering algorithm is a key issue that distinguishes the semi-supervised clustering algorithm from the unsupervised clustering algorithm. In the traditional general framework for attribute-based clustering studies, the measurement of object similarity or dissimilarity (or distance) is the primary issue to be addressed, but in practice, the constraint information provided by the user is mostly at the object level rather than the attribute level, so it is very difficult to integrate user-provided constraints into the measurement process of similarity or dissimilarity. Therefore, except for the case in which the values of the user-provided attributes are identical, other user-provided constraint information should be used in the clustering process of the central object. Therefore, regarding this problem, the measurement of central object similarity and the process of central object clustering are two secondary key issues that need to be solved.

Based on the above discussions, we conclude that a complete procedure of heterogeneous data clustering considering user-provided constraints include four steps: (1)The presentation of user-provided constraints; (2) The measurement of the central object similarity; (3) The semi-supervised clustering of central objects considering user-provided constraints; (4) The clustering of attribute objects.

3.2 Presentation of user-provided constraints

At present, it is agreed in semi-supervised clustering studies that it is necessary to impose certain constraints on clustering results, i.e., the descriptions of the relation constraints between data objects can be categorized into two categories of must-link constraints and cannot-link constraints, with some respective properties. However, according to the above analyses, user-provided constraints are not limited to the above mentioned user constraints on object relations and may also include decisive attribute constraints and attribute value equality constraints.

Definition 1. User constraint on object relation: For the given heterogeneous data of $D=(C,A)$, assume that the user-provided constraint is on the central objects of $C = \{C_i | i = 1, 2, ..., n\}$ and that the central objects cluster as the set of *Clus* that contains N central object clusters, described as $C.Clus=\{Clus_1, Clus_2, ..., Clus_N\}$. Suppose that there are two central objects, i.e., C_s and C_t $(s \neq t)$. The must-link constraint and the cannot-link constraint of the user-provided constraint on object relation can be specifically described as follows:

- If C_s and C_t should be in the same cluster, then $Must-link(C_s, C_t) = True$, which can be specifically described as $C_s \in Clus_i$, $C_t \in Clus_j$, $i = j$;
- If C_s and C_t should not be in the same cluster, then $Cannot-link(C_s, C_t) = True$, which can be specifically described as $C_s \in Clus_i$, $C_t \in Clus_j$, $i \neq j$.

Therefore, the must-link constraint and the cannot-link constraint are both Boolean functions and have the following properties:

Remark 1. The must-link constraint and the cannot-link constraint of the user's two types of constraints have symmetry, and for $C_s, C_t \in C$:

- *Must-link*$(C_s, C_t) \Leftrightarrow$ *Must-link*(C_t, C_s);
- *Cannot-link*$(C_s, C_t) \Leftrightarrow$ *Cannot-link*(C_t, C_s).

Remark 2. The must-link constraint and the cannot-link constraint of the user's two types of constraints have limited transitivity, and for $C_r, C_s, C_t \in C$:

- *Must-link*(C_r,C_s)&&*Must-link*(C_s,C_t)⇒*Must-link*(C_r,C_t);
- *Must-link*(C_r,C_s)&&*Cannot-link*(C_s,C_t)⇒*Cannot-link*(C_r,C_t).

Definition 2. User Constraint on Decisive Attribute: For the given heterogeneous data of D=(C,A), assume that the user-provided constraint is on the central objects of $C = \{C_i|i = 1, 2, ..., n\}$ and that the central objects eventually cluster as the set of *Clus* that contains N central object clusters, described as $C.Clus$=$\{Clus_1, Clus_2, ..., Clus_N\}$. Suppose that there are two central objects, i.e., C_s and C_t ($s \neq t$), and that the cluster assignments of C_s and C_t are represented by $C_s.Clus$ and $C_t.Clus$, respectively. Then, a user-provided constraint on decisive attributes can be described as $C_s.\ A_{kp}$=$C_t.A_{kp} \Leftrightarrow C_s.Clus$=$C_t.Clus$, in which A_k is denoted as the decisive attribute.

Definition 3. User Constraint on Attribute Value Equality: For the given heterogeneous data of D=(C,A), assume that the user-provided constraint is on the central objects of $C = \{C_i|i = 1, 2, ..., n\}$ and that there are two central objects, i.e., C_s and C_t (s≠t), the user-provided constraint on attribute value equality can be described as $A_{kp} = A_{kq}$ ($p \neq q$).

3.3 Measurement of central object similarity

The measurement of central object similarity in heterogeneous data with a star structure adopts the following ideas: first, if the user provides a constraint on the attribute value equality, then it is used to get the heterogeneous data with unique identifiers; otherwise, this step is skipped. In a practical problem, the user may not be able to provide all the above-described three types of constraints. Then, when the central object similarity is measured pairwise, the similarity between the two central objects is represented using linear combinations of the two objects on each of attribute objects, in which the coefficient of each attribute object in the linear combination constitutes the contribution coefficient vector, which can be adjusted according to the actual situation. Therefore, the following basic concepts are defined.

Definition 4. Star-structured Data with Unique Identifiers (IDs): Star-structured heterogeneous data with unique IDs containing n central objects and r ($r \geqslant 1$) attribute objects are described as D'=(C,A,ID,R) (D'=(C,A) for short). The specific meanings are as follows:

- C represents the collection of central objects, i.e., $C = \{C_i\}_{i=1}^{n}$, where C_i represents the ith central object.
- A represents the collection of attribute objects, i.e., $A = \{A_k\}_{k=1}^{r}$ (k∈{1,2,...,r}), where $A_k = \{A_{kp}\}_{p=1}^{n_{A_k}}$ represents the object collection of the kth attribute ($p \in \{1, 2, ..., n_k\}$), A_{kp} represents the pth attribute object of the kth attribute object ($p \in \{1, 2, ..., n_k\}$), and n_{A_k} represents the number of attribute objects included in A_k.
- ID represents the collection of all objects, i.e., $ID = C.ID \bigcup A.ID$. $C.ID$ represents the collection of central objects with unique IDs, and $C.ID = \{C_i.ID\}_{i=1}^{n}$, where $C_i.ID$ represents the unique ID of the ith central object (i∈{1,2,...,n}). $A.ID$ represents the collection of attribute objects, and $A.ID = \{A_k.ID\}_{k=1}^{r}$ (k∈{1,2,...,r}), in which $A_k.ID = \{A_{kp}.ID\}_{p=1}^{n_k}$ represents the collection of the objects with the kth attribute ($p \in \{1, 2, ..., n_k\}$), where A_{kp} represents the pth attribute object of the kth attribute object ($p \in \{1, 2, ..., n_k\}$) and n_k represents the number of attribute objects included in A_k.
- R represents the undirected relationship collection present in the dataset of D, i.e., $R = \{r_l\}_{l=1}^{n_R}$ ($l \in \{1, 2, ..., n_R\}$), and for any relationship that is $r_l =< r_l.one, r_l.theother >$,

it satisfies the following condition: $r_l.one \in C$ and $r_l.theother \in A$ or $r_l.one \in A$ and $r_l.theother \in C$.

Given that no relationship between central objects is available in the star-structured heterogeneous data, the measurement of central object similarity can only rely on attribute objects, so the similarity of central object in terms of the remaining attribute objects is defined as follows:

Definition 5. Similarity between Central Objects in terms of the kth Type of Attribute Object: For the given heterogeneous dataset of $D'=(C,A)$, the calculation formula of the similarity, $S_k(C_i, C_j)$, between two central objects C_i and C_j in terms of the kth type of attribute object is as follows:

$$S_k(C_i, C_j) = \frac{2 \times |C_i.A_k.ID \cap C_j.A_k.ID|}{|C_i.A_k.ID| + |C_j.A_k.ID|} \tag{1}$$

where $C_i.A_k.ID$ represents the collection of IDs of the kth type of attribute object correlated to C_i and $C_j.A_k.ID$ represents the collection of IDs of the kth type of attribute object correlated to C_j.

Obviously, the value range of $S_k(C_i, C_j)$ is [0, 1], and it is easy to prove that it meets the properties of similarity measurement.

The linear combinations of the similarities of the central object on each of various attributes are considered to be used to measure the similarity between the central objects, but the roles of different attribute objects also differ, so we provide the following definition:

Definition 6. Similarity between Central Objects: For the given heterogeneous dataset of $D'=(C,A)$, the similarity, $S(C_i, C_j)$, between two central objects C_i and C_j is the linear combinations of similarities between the two in terms of the kth type of attribute object, with the following calculation formula:

$$S(C_i, C_j) = \sum_{k=1}^{r} w_k \cdot S_k(C_i, C_j) \tag{2}$$

where w_k is called the contribution coefficient, representing the contribution of the kth type of attribute object to the judgment of whether C_i and C_j are similar and satisfying: (1) $\sum_{k=1}^{r} w_k = 1$ and (2) $w_k \geqslant 0, w_k \in R$, which jointly constitute the contributing coefficient vector $w = (w_1, w_2, ..., w_k, ..., w_r)$.

Obviously, the value range of $S(C_i, C_j)$ is [0, 1], and it is easy to prove that it meets the properties of similarity measurement. w_k is evaluated according to the actual situation.

Central object similarity measurement of the UserHeteClus

Input: Star-structured heterogeneous data with unique ID ($D'=(C,A)$) and contribution coefficient vector ($w = (w_1, w_2, ..., w_k, ..., w_r)$), in addition to the constraint of attribute value equality.

Output: The similarity matrix $SimMatrix(C)$ of the central objects of $C = \{C_i\}_{i=1}^{n}$.

Procedures:

Step 1: The central objects of $C = \{C_i\}_{i=1}^{n}$ are represented using their attribute objects.

Step 2: According to the user-provided attribute value equality constraint, the value IDs of attributes with equal value are represented by one of the IDs; if the user does not provide the attribute value equality constraint, then skip this step.

Step 3: The similarity between any two central objects on each of various attribute objects $S_k(C_i, C_j)$ is calculated according to Definition 5.

Step 4: The similarity between any two central objects $S(C_i, C_j)$ is calculated according to Definition 6 to obtain the central object similarity matrix $SimMatrix(C)$.

3.4 Semi-supervised clustering of central objects considering user-provided constraints

Definition 7. Similarity between Central Object Clusters: For the given heterogeneous dataset of $D'=(C,A)$ and several central object clusters $Clus$, the similarity between central object clusters $Clus_s$ and $Clus_t$, $S(Clus_s, Clus_t)$, the average of the similarities between central objects contained in one cluster and those contained in another cluster, has the following formula:

$$S(Clus_s, Clus_t) = \frac{\sum_{i=1}^{|Clus_s|} \sum_{j=1}^{|Clus_t|} S(C_i, C_j)}{|Clus_s| \times |Clus_t|} \tag{3}$$

where $S(C_i, C_j)$ represents the similarity between C_i and C_j.

Obviously, the value range of $S(Clus_s, Clus_t)$ is [0, 1], and it is easy to prove that it meets the three properties of similarity measurement.

For the above mentioned three forms of user-provided constraint, the constraint of attribute value equality is related to the ID of the object, so it needs to be addressed first; the user constraint of the object attribute can be converted into the user constraint of the object relation, which is related to the clustering process, and can be incorporated into the specific clustering process. Therefore, the rationale for the semi-supervised clustering of central objects considering multiple user-provided constraints is that first, if the user provides a constraint on decisive attributes, then it is converted into a user constraint on the object relation, and then, the pairwise central objects similarities are sorted in descending order. After completing the above preparatory steps, the actual clustering process is executed. First, each object is treated as a separate cluster, and the objects in the must-link set are first linked. Then, the two most similar central objects are judged in terms of whether they meet the condition for the linking and sequentially repeated; if the two are in the cannot-link collection, then proceed to the next pair of central objects. Otherwise, the two are judged in terms of whether they have already in the same cluster, and if they are, then proceed to judging the next pair of central objects. Otherwise, the similarity between the cluster in which two objects reside is calculated, and if the similarity threshold is met, then the two objects are linked; if it is not, then the two are not linked. The process is repeated until the objects are linked as one cluster or the number of clusters is reached.

Semi-supervised clustering of central objects considering user-provided constraints of the UserHeteClus

Input: The collection of central objects $C = \{C_i | i = 1, 2, ..., n\}$, the central object similarity matrix $SimMatrix(C)$, the must-link set, the cannot-link set, the constraint of decisive object, the constraint of attribute value equality, and central object similarity threshold λ.

Output: Central object clustering result $C.Clus$.

Procedures:

Step 1: If the user provides a constraint on a decisive attribute, then convert it into a user constraint on object relation.

Step 2: The central object similarities are sorted in descending order.

Step 3: Each central object is treated as a separate cluster.

Step 4: Search the must-link collection to link the clusters in which the objects with the must-link constraint reside.

Step 5: Sequentially, for the two central objects with the highest similarity:

Step 5.1: Determine whether they are in the cannot-link set; if they are, then they are not linked, and continue to determine the next pair of central objects.

Step 5.2: Determine whether the two belong to the same cluster: if they do, then continue to determine the next pair of central objects.

Step 5.3: Determine whether the central object pair formed by the central objects composed of the two clusters that the two are from is present in the cannot-link set; if it is, then they are not linked, and continue to determine the next pair of central objects.

Step 5.4: Calculate the similarity between the two clusters that the two central objects are from. If the similarity is greater than or equal to λ, then link the two; otherwise, do not link, and continue to determine the next pair of central objects.

Step 5.5: Repeat Steps 5.1-5.4 until the objects are linked as one cluster or reach the required number of clusters.

3.5 Clustering of attribute objects

In the process of heterogeneous data clustering considering user-provided constraints, central objects are the focus of heterogeneous data clustering; because it is insufficient for the original attribute object information to support the clustering of the central objects, the user-provided partially labeled data are introduced and integrated into the process of clustering the central objects.

Therefore, in this study, we believe that the clustering of attribute objects should be based on the clustering result of the central objects and adopt the approach of "voting" using the nearest cluster of the attribute object to classify the attribute object to the central object cluster with the most votes and then separate it from the type of attribute object cluster, which is used in our previous study [12].

4 Experimental analysis of the UserHeteClus

4.1 Experimental data preparation

To test the clustering effectiveness of the UserHeteClus on actual data, in this section, we used the China National Knowledge Infrastructure (CNKI) literature data source to extract nonredundant records (2467 entries, searched on March 19, 2014) from the CNKI dataset of the Donlinks School of Economics and Management (DSEM) of University of Science and Technology, Beijing, using the four textual segments of paper title, author, source, and keyword. For convenience of describing the clustering results of the algorithm, records (in a total of 168 entries) that have no reference and the Chinese Library Classification (CLC) containing T were used as the testing dataset of the UserHeteClus (part of which is presented in Table 8) to analyze the research fields of the papers that are related to computer applications by authors from DSEM.

The constraints for this dataset include the following:

(1) The constraints of attribute value equality

A. The keywords "steelmaking - concasting", "concasting", and "steelmaking and concasting", and they were involved in 9 records.

B. The keywords of "clustering", "cluster analysis", and "clustering algorithm", and they were involved in 17 records.

(2) The constraints on user relations

A. Must-link constraint. The papers titled "Chinese keyword extraction algorithm based on high-dimensional clustering techniques", "Pattern aggregation theory-based text feature dimensionality reduction method and its application in text classification", and "Text classification based on granular network generation rules" belonged to the same cluster and were involved in three records.

Table 1: CNKI experimental dataset for the UserHeteClus

Id	Title	Author	Source	Keywords
59	Overview and analysis of manufacturing execution system models	Li Tieke	Metallurgical automation	Manufacturing execution system, information system, system model, enterprise model
62	Research on cryptographic algorithm in data transmission of Internet of Things of RFID system	Wang Xiaoni, Wei Guiying	Journal of Beijing Information Science and Technology University (Natural Science Edition)	Internet of Things, radio frequency identification, cryptographic algorithm
137	Telecom customer segmentation methods and applications	Chen Fengjie	Technology and industry	Data mining, clustering, Clementine
223	Global neighborhood algorithm for Job Shop scheduling problem	Cui Jianshuang, Li Tieke	Computer integrated manufacturing system	Neighborhood structure, critical path, job shop scheduling, neighborhood switching, scheduling algorithm
259	Research on CURE algorithm of hierarchical clustering method	Wei Guiying, Zheng Xuanxuan	Technology and industry	CURE algorithm, hierarchical clustering, clustering
268	Hybrid vehicle routing problem based on improved fuzzy genetic algorithm	Zhang Qun, Yan Rui	Chinese Management Science	Vehicle routing problem, fuzzy genetic algorithm, multidistribution center
...	...	...	...	...
2456	Research on the problem of determining the number of rolling units	Chen Xiong, Pan Yongquan	Control and decision	Rolling unit, simulated annealing algorithm, random variable variance, scheduling
2473	Diagnostic theory of two fuzzy control charts	Chen Zhiqiang, Zhang Gongxu, Yan Zhilin	Journal of Beijing University of Science and Technology	Shewhart control chart, selection control chart, total quality, subquality, fuzzy judgment

B. Cannot-link constraint. The papers titled "Basic framework and method for China's overseas mining investment decision process" and "High temperature compressive strength of coke for blast furnace" did not belong to the same cluster and were involved in two records.

In this experiment, two constraints were provided, and in practice, the three types of user-provided constraints may all be provided, or only one type may be provided.

4.2 Analysis of experimental results

(1) Analysis of object clustering accuracy

The UserHeteClus ($w_{author} = 0.3, w_{venue} = 0.2, w_{term} = 0.5, \lambda = 0.01$) generated 22 clusters

on 168 paper objects of the experimental dataset, which included two major clusters and 13 clusters of isolated points (Table 2). The meaning of each cluster was explained by looking at the title and keywords of the paper, and the cluster of isolated points was explained based on the title of the paper.

Table 2: Clustering result on paper objects of the experimental dataset with UserHeteClus

No.	Cluster size	Objects in the cluster	Interpretation
1	72	"1191", “1005", “1109", “945", “982", ...	Manufacturing execution system, production scheduling algorithm, and other papers
2	63	“383", “284", “947", “1865", “436", ...	Data mining papers
3	7	“687", “1067", “1284", “468",...	Software project management research
4	3	“2284", “2294", “2253"	Calculate accounting indicators with Excel
5	2	“2437", “2436"	Management information system development
6	2	“2346", “1383"	Supply chain and ERP
7	2	“1902", “1045"	System design and implementation using ASP.NET, etc.
8	2	“1796", “1551"	Temperature of nanoporous vacuum insulation panel
9	2	“955", “325"	Lean improvement research
10	1	“881"	The basic framework and method of China’s overseas mining investment decision-making process
11	1	“2330"	EU textile environmental label and its comprehensive evaluation
12	1	“1178"	Control technology of pipeline steel inclusions
13	1	“2271"	Application of value engineering in the development of building material products
14	1	“1921"	Research on factors and methods of reservoir management post evaluation
15	1	“393"	Recovery of the thorium resources of the mine in Baotou and its research status for nuclear fuel
16	1	“395"	Elliptic curve encryption algorithm and case analysis
17	1	“917"	Combination forecast of technical maturity of patented products of industrial pulverized coal boiler based on TRIZ theory
18	1	“298"	Investigation on the construction of evaluation index system for energy efficiency reform of existing buildings
19	1	“618"	Internet and e-commerce and logistics
20	1	“1680"	Enterprise system semantic complexity based on isomorphic ontology structure
21	1	“1535"	Current situation and development strategy of UPS
22	1	“274"	High temperature compressive strength of coke for blast furnace

The analysis of the central object clustering results indicates the following:

A. The classification on the macroclusters via the UserHeteClus is clear, with high inter-cluster discriminability, and has identified two major directions of computer-related studies, i.e., "manufacturing execution system" and "data mining", in DSEM, which are in line with the actual meaning of the cluster, with ideal clustering effectiveness.

B. For each specific cluster, clear meanings of the cluster are present, indicating that the intracluster similarity between the objects generated by the UserHeteClus is high.

C. The clusters of isolated points identified by the UserHeteClus are different from the major clusters, and the difference between the isolated points is also large.

(2) Analysis of the effect of main parameters of the algorithm

The UserHeteClus has two sets of parameters: w and λ. Specifically, the parameter set of w is used to control the weight of each type of attribute object when calculating the similarity

between the central objects, and λ is the central object similarity threshold, which affects and controls the clustering process of central objects. In this study, we also examined the effect of λ on the clustering result.

Since different values of λ lead to different numbers of central object clusters, the relationship between different values of λ and the number of central object clusters was tested on the UserHeteClus ($w_{author} = 0.3, w_{venue} = 0.2, w_{term} = 0.5$) using the CNKI experimental dataset (Fig. 1). It can be observed that a monotonically increasing relationship was present between λ and the number of central object clusters (paper cluster); specifically, when $\lambda = 0$, the number of paper clusters was 1; as the value of λ gradually increased, the number of paper clusters also gradually increased; when $\lambda = 1.0$, the number of paper clusters reached 168, i.e., the number of paper objects. Understanding the relationship between λ and the number of central object clusters helps to determine the optimal value of λ in practical applications.

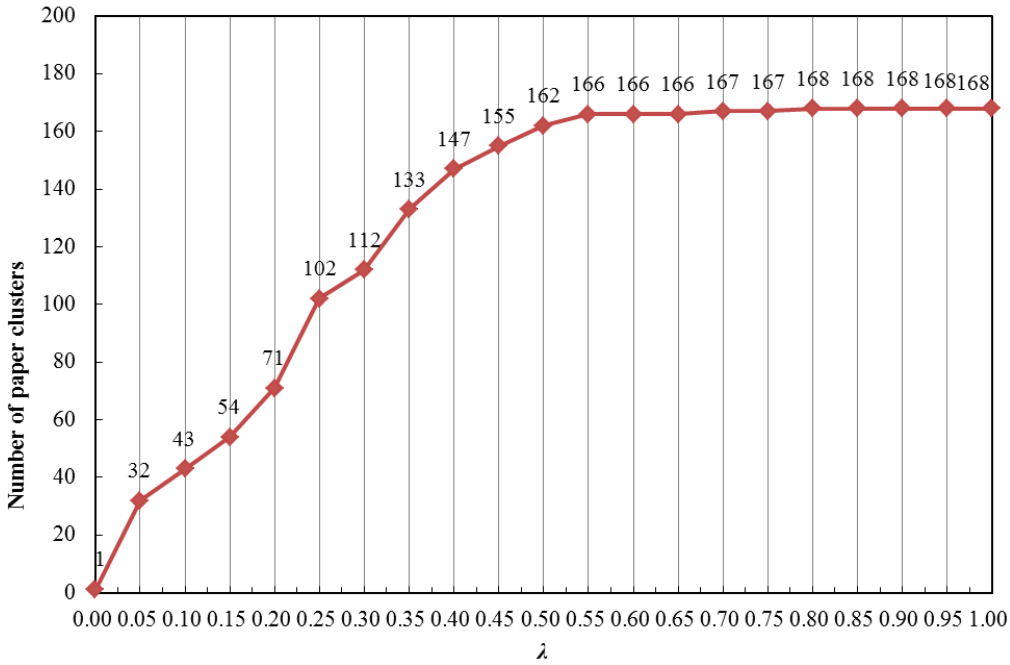

Figure 1: Relationship between the UserHeteClus parameter λ and the number of paper clusters

5 Conclusion

In this study, we investigated the issue of clustering heterogeneous data considering multiple user-provided constraints, in which the auxiliary external information about the object provided by the user is considered in addition to the information of the object itself. Firstly, we analyzed and we presented three types of constraint information that are provided by the user to help the clustering process of the objects, including constraints on user relations, constraints on user decisive attributes, and constraints on attribute value equality, which provide bases for future classification processes involving user-provided constraints in clustering algorithms. Secondly, we proposed a complete general analysis framework for clustering heterogeneous data considering user-provided constraints and then summarized the key issues for the case of star-structured heterogeneous data, including (1) the representation of user-provided constraints, (2) measurement of central object similarity, (3) semi-supervised clustering of central objects and (4) attribute

objects considering user-provided constraints. Lastly, we proposed using linear combinations of similarities of two central objects with all attribute objects to measure the similarity between the two. In addition, we defined the contribution coefficient to quantify the weight of various attribute objects on the central object similarity, based on which the similarities were sorted, enabling fast clustering of central objects under different types of user-provided constraints. For future work, we intend to investigate the issue of clustering heterogeneous data considering user constraints for arbitrary structure.

Funding

This work was partially supported by the Humanity and Social Science Youth Foundation of Ministry of Education of China (17YJCZH069), Science Foundation of Beijing Language and Culture University (supported by "The Fundamental Research Funds for the Central Universities") (19YJ040001) and BLCU Youth Talent Development Program.

Bibliography

[1] Banerjee, A.; Dhillon, I.S.; Ghosh, J. Merugu S.; Modha, D.S. (2004). A generalized maximum entropy approach to Bregman co-clustering and matrix approximation, *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 509–514, 2004.

[2] Bekkerman, R.; El-Yaniv, R.; McCallum, A. (2005). Multi-way distributional clustering via pairwise interactions, *Proceedings of the 22nd International Conference on Machine Learning*, 41–48, 2005.

[3] Chen, Y.; Wang, L.; Dong, M. (2010); Non-negative matrix factorization for semisupervised heterogeneous data coclustering, *IEEE Transactions on Knowledge and Data Engineering* 22(10), 1459–1474, 2010.

[4] Dai, Y.; Wu, W.; Zhou, H.; Zhang, J; Ma, F. (2018). Numerical simulation and optimization of oil jet lubrication for rotorcraft meshing gears, *International Journal of Simulation Modelling*, 17(2), 318–326, 2018.

[5] Dai, Y.; Zhu, X.; Zhou, H.; Mao, Z.; Wu, W. (2018). Trajectory tracking control for seafloor tracked vehicle by adaptive neural-fuzzy inference system algorithm, *International Journal of Computers Communications & Control*, 13(4), 465–476, 2018.

[6] Dhillon, I.S. (2001). Co-clustering documents and words using bipartite spectral graph partitioning, *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 269–274, 2001.

[7] Dhillon, I.S.; Mallela, S.; Modha, D.S. (2003). Information-theoretic co-clustering, *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 89–98, 2003.

[8] Gao, B.; Liu, T.; Ma, W. (2006). Star-structured high-order heterogeneous data co-clustering based on consistent information theory, *Proceedings of the Sixth IEEE International Conference on Data Mining*, 880–884, 2006.

[9] Getz, G.; Levine, E.; Domany, E. (2000). Coupled two-way clustering analysis of gene microarray data, *Proceedings of the National Academy of Sciences*, 97(22), 12079-12084, 2000.

[10] Han, J.; Kamber, M.; Pei, J. (2012). *Data Mining: Concepts and Techniques (Third Edition)*, Morgan Kaufmann Publishers, 2012.

[11] Huang, Y. (2016). A three-phase algorithm for clustering multi-typed objects in star-structured heterogeneous data, *International Journal of Database Theory and Application*, 9(8), 107–118, 2016.

[12] Huang, Y. (2017). Clustering multi-typed objects in extended star-structured heterogeneous data, *Intelligent Data Analysis*, 21(2), 225–241, 2017.

[13] Huang, Y.; Gao, X. (2014). Clustering on heterogeneous networks, *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 4(3), 213–233, 2014.

[14] Ienco, D.; Robardet, C.; Pensa, R.G.; Meo, R. (2013). Parameter-less co-clustering for star-structured heterogeneous data, *Data Mining and Knowledge Discovery*, 26(2), 217–254, 2013.

[15] Long, B.; Zhang, Z.; Wu, X.; Yu, P.S. (2006). Spectral clustering for multi-type relational data, *Proceedings of the 23rd International Conference on Machine Learning*, 585–592, 2006.

[16] Mei, J.; Chen, L. (2012). A fuzzy approach for multitype relational data clustering, *IEEE Transactions on Fuzzy Systems*, 20(2), 358–371, 2012.

[17] Pio, G.; Serafino, F.; Malerba, D.; Ceci, M. (2018). Multi-type clustering and classification from heterogeneous networks, *Information Sciences*, 425, 107–126, 2018.

[18] Rege, M.; Yu, Q. (2008). Efficient mining of heterogeneous star-structured data, *International Journal of Software and Informatics*, 2(2), 141–161, 2008.

[19] Sun, Y.; Han, J.; Zhao, P.; Yin, Z.; Cheng, H.; Wu, T. (2009). RankClus: integrating clustering with ranking for heterogeneous information network analysis, *Proceedings of the 12nd International Conference on Extending Database Technology: Advances in Database Technology*, 565–576, 2009.

[20] Sun, Y.; Yu, Y.; Han, J. (2009). Ranking-based clustering of heterogeneous information networks with star network schema, *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 797–806, 2009.

[21] Tang, L.; Liu, H. (2009). Uncovering cross-dimension group structures in multi-dimensional networks, *Proceedings of SDM Workshop on Analysis of Dynamic Networks*, 677–685, 2009.

[22] Tang, L.; Liu, H.; Zhang, J. (2012). Identifying evolving groups in dynamic multimode networks, *IEEE Transactions on Knowledge and Data Engineering*, 24(1), 72–85, 2012.

[23] Wagstaff, K.; Cardie, C. (2000). Clustering with instance-level constraints, *Proceedings of the 17th International Conference on Machine Learning*, 1103–1110, 2000.

[24] Yin, X.; Han, J.; Yu, P.S. (2006). LinkClus: efficient clustering via heterogeneous semantic links, *Proceedings of the 32nd International Conference on Very Large Data Bases*, 427–438, 2006.

[25] Zhang, W.; Zhang, Z.; Chao, H.; Tseng, F. (2018). Kernel mixture model for probability density estimation in Bayesian classifiers, *Data Mining and Knowledge Discovery*, 32(3), 675–707, 2018.

[26] Zhang, W.; Zhang, Z.; Qi, D.; Liu, Y. (2014). Automatic crack detection and classification method for subway tunnel safety monitoring, *Sensors*, 14(10), 19307–19328, 2014.