Absolute and Relative Clustering

Toshihiro Kamishima and Shotaro Akaho
National Institute of Advanced Industrial Science and Technology (AIST)
AIST Tsukuba Central 2, Umezono 1-1-1, Tsukuba, Ibaraki, 305-8568 Japan
mail@kamishima.net, s.akaho@aist.go.jp

ABSTRACT

Research into (semi-)supervised clustering has been increasing. Supervised clustering aims to group similar data that are partially guided by the user's supervision. In this supervised clustering, there are many choices for formalization. For example, as a type of supervision, one can adopt labels of data points, must/cannot links, and so on. Given a real clustering task, such as grouping documents or image segmentation, users must confront the question "How should we mathematically formalize our task?" To help answer this question, we propose the classification of real clusterings into absolute and relative clusterings, which are defined based on the relationship between the resultant partition and the data set to be clustered. This categorization can be exploited to choose a type of task formalization.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Applications—*Data mining*; I.5.3 [**Pattern Recognition**]: Clustering—*Algorithms*

General Terms

Algorithms and Experimentation

1. INTRODUCTION

The goal of *(semi-)supervised clustering* is to partition a data set under the user's supervision [17, 9, 18, 2]. Various types of problem formalizations as well as algorithms have been proposed as methods of supervised clustering. For example, users represent their preferences for grouping structures by labels [20]. Data points that are assigned the same label are grouped into the same cluster, and data with different labels are separated into different clusters. Another way of implementing the user's supervision is to use must/cannot links [17]. A must link indicates that a pair of linked data should be clustered together, while cannot-linked data pairs should be separated into different clusters. When a user tries to formalize a given task in the real world as a mathematical problem, he/she has many choices in several components of supervised clustering, such as, the formats of input examples, the types of supervision, information provided by features, and the goal of learning.

To help user's choose among from various alternatives, we classify real clustering tasks into two categories: absolute and relative clustering. For each category, some choices of formalization are axiomatically constrained. For example, a supervision task using labels is not allowed when formalizing a relative clustering task. Therefore, considering the categorization is helpful for a user to select appropriate formalization candidates.

Intuitively speaking, in the determination of whether two objects are assigned to the same cluster or not, the clustering task is absolute if the determination is independent of other objects; if not, the task is relative. We give examples of each type. A reference matching task is an example of absolute clustering tasks [5]. The goal of this task is to group reference strings into clusters of multiple real references to persons or documents consisting of the same entity. Citation strings can differ greatly, even when referring to the same paper. For example, the phrase "Knowledge Discovery and Data Mining" can be written as "KDD." Two strings, "A. Turing" and "Alan Turing," refer to the same person. Further, the order of fields may change, e.g., Author$\rightarrow \cdots \rightarrow$Year or Author$\rightarrow$Year$\rightarrow \cdots$. To carry out this reference matching task, the sameness of an entity must be detected even if the citation strings are different. Note that the task of identifying the sameness of entities in the real world is also called "record linkage" or "identity uncertainty." In this clustering task, if any two strings truly refer to the same entity, they must be clustered together, independently from information in any other strings in the data set. Consequently, reference matching is an absolute clustering task.

A noun coreference task is an example of relative clustering tasks. The aim of this task is to group noun phrases in a document into clusters of phrases corresponding to the same entity or concept [8]. For example, in a news article, if one determines that the phrases, "Mr. Abe," "the prime minister of Japan," and "he" represent the same person, these phrases are clustered together. To show that this task is a kind of relative clustering, we represent the following example with three sentences:

- [A] There is $^{(1)}$a parent turtle.
- [B] On $^{(2)}$this turtle, there is $^{(3)}$a child turtle.

MultiClust '13, August 11, 2013, Chicago, Illinois, USA.

- [C] On [4]this turtle, there is [5]a grandchild turtle.

For a document consisting of sentences [A], [B], and [C], five phrases are partitioned into three clusters: {(1), (2)}, {(3), (4)}, and {(5)}. These three clusters represent "a parent turtle," "a child turtle," and "a grandchild turtle," respectively. In this case, the phrases (1) and (4) are assigned to different clusters. We next eliminate sentence [B] and consider the document consisting of [A] and [C]. The resultant clusters are changed to {(1), (4)} and {(5)}, and the phrases (1) and (4) are now assigned to the same cluster. That is to say, the determination of whether phrases (1) and (4) are assigned to the same cluster depends on the existence of phrases (2) and (3) in the document. Consequently, this noun coreference is a relative clustering task.

Depending on whether the real clustering task is absolute or relative, the following three points must be considered in order to formalize the task as a mathematical problem: the formats of the input examples, the types of supervision, and information provided by the features. Based on the input format and the goal of the algorithm, we classify supervised clustering problems into three types: transductive, semi-supervised, and fully supervised. We show that absolute and relative clustering tasks should be formalized as semi-supervised and fully supervised clustering problems, respectively. We then explain why supervision using labels is not appropriate for the formalization of a relative clustering task. Further, we discuss what kind of information features should represent to perform an absolute or a relative clustering task. By determining whether the targeted real task is absolute or relative, a user can appropriately formalize it.

In section 2, we formally state the notion of absolute and relative clustering tasks, and show examples of them. In section 3, we propose the categorization of supervised clustering problems. In section 4, we discuss how the distinction between an absolute and a relative task affects input formats, types of supervision, and design approaches of attributes. Section 5 summarizes our discussion.

2. ABSOLUTE CLUSTERING AND RELATIVE CLUSTERING

We have shown intuitive definitions of absolute and relative clustering tasks. While the assignments of objects in an absolute clustering task are unconditional, relative clustering depends on the composition of the object set. We here give a more formal definition of absolute and relative clustering tasks, which we intuitively described above. A universal object set is a population of all possible objects and is denoted by $\mathcal{X}$. An object set is $X = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\} \subset \mathcal{X}$, where N is the number of objects and $\mathbf{x}_1, \ldots, \mathbf{x}_N$ are objects that are generally represented by features. A partition of X is composed of exhaustive and disjoint subsets of X, and is denoted by $C_X = \{c_1, \ldots, c_K\}$, such that $X = c_1 \cup \cdots \cup c_K$ and $c_i \cap c_j = \emptyset, i \neq j$, where $c_1, \ldots, c_K$ are clusters and K is the number of clusters. The function $\delta(\{\mathbf{x}_i, \mathbf{x}_j\}, C_X)$ takes 1 if the objects $\mathbf{x}_i$ and $\mathbf{x}_j$, $i \neq j$, are in the same cluster in the partition C_X; otherwise, it takes 0.

A clustering function, $\pi(X)$, deterministically maps a given object set, X, into a partition, C_X. We use the term "task" to represent tasks in the real world, such as reference matching or noun coreference in the introduction, and the term "problem" to represent mathematically formalized problems. For a specific clustering task, X corresponds to the formal representation of entities to be clustered in the task and C_X corresponds to the appropriate partition that fits for the goal of the task; then a clustering function that maps the X into the C_X is called a true clustering function, which is denoted by π^*. A (supervised) clustering problem is to find an estimated clustering function, $\hat{\pi}(X)$, that best approximates a true function among candidate clustering functions. To carry out a clustering task in the real world, we have to formalize it as a corresponding mathematical problem.

Given these definitions, notions of absolute and relative clustering tasks are defined as follows:

DEFINITION 1. *The true clustering function, $\pi^*(X)$, corresponds to the target task in the real world. If the function satisfies the condition of equation* (1)*, the task that corresponds to this function is called* absolute clustering*; otherwise, it is called* relative clustering*.*

$$\delta(\{\mathbf{x}_i, \mathbf{x}_j\}, \pi^*(X)) = \delta(\{\mathbf{x}_i, \mathbf{x}_j\}, \pi^*(X')), \quad {}^{\forall}\mathbf{x}_i, \mathbf{x}_j \in X \cap X',\ \mathbf{x}_i \neq \mathbf{x}_j,\ {}^{\forall}X, X' \subseteq \mathcal{X}. \quad (1)$$

In other words, for any determination of whether the objects $\mathbf{x}_i$ and $\mathbf{x}_j$ are assigned to the same cluster or not, the determination is not dependent on the object set to be clustered, but is dependent only on the objects $\mathbf{x}_i$ and $\mathbf{x}_j$. In this case, the clustering function corresponds to the absolute clustering task.

An absolute clustering task is also defined by using the following notion of an *absolute partition*.

DEFINITION 2. *A partition of a universal object set $\mathcal{X}$ is an absolute partition, $\mathcal{C}$, if it satisfies the condition:*

$$\delta(\{\mathbf{x}_i, \mathbf{x}_j\}, \pi^*(X)) = \delta(\{\mathbf{x}_i, \mathbf{x}_j\}, \mathcal{C}), \quad {}^{\forall}\mathbf{x}_i, \mathbf{x}_j \in X,\ \mathbf{x}_i \neq \mathbf{x}_j,\ {}^{\forall}X \subseteq \mathcal{X}. \quad (2)$$

A true clustering function corresponds to an absolute clustering task if and only if there exists an absolute partition. A simple proof is as follows:
If the condition of equation (1) is satisfied, $\pi^*(X')$ is an absolute partition under $X' = \mathcal{X}$. Inversely, if equation (2) is satisfied for any X and X', equation (1) is also satisfied, because

$$\delta(\{\mathbf{x}_i, \mathbf{x}_j\}, \pi^*(X)) = \delta(\{\mathbf{x}_i, \mathbf{x}_j\}, \mathcal{C}) = \delta(\{\mathbf{x}_i, \mathbf{x}_j\}, \pi^*(X')).$$

An absolute clustering task has the following property: If the true clustering function $\pi^*(X)$ corresponds to an absolute clustering task, the must link satisfies transitivity for any $X \in \mathcal{X}$. The must link between $\mathbf{x}_i$ and $\mathbf{x}_j$ implies $\delta(\{\mathbf{x}_i, \mathbf{x}_j\}, \pi^*(X)) = 1$. For any two object sets, X_1 and X_2, that have an object $\mathbf{x}_1$ in common, two objects, $\mathbf{x}_1, \mathbf{x}_2 \in X_1$, are connected by a must link, and $\mathbf{x}_1, \mathbf{x}_3 \in X_2$ are must-linked too, i.e., $\delta(\{\mathbf{x}_1, \mathbf{x}_2\}, \pi^*(X_1)) = 1$ and $\delta(\{\mathbf{x}_1, \mathbf{x}_3\}, \pi^*(X_2)) = 1$. Together with equation (1), these links imply $\delta(\{\mathbf{x}_1, \mathbf{x}_2\}, \pi^*(X_1 \cup X_2)) = 1$ and $\delta(\{\mathbf{x}_1, \mathbf{x}_3\}, \pi^*(X_1 \cup X_2)) = 1$. Therefore, $\delta(\{\mathbf{x}_2, \mathbf{x}_3\}, \pi^*(X_1 \cup X_2)) = 1$ holds due to the definition of a must link. However, if $\pi^*(X)$ corresponds to a relative clustering task, this function does not have this property, because equation (1) is violated.

2.1 Further Examples of Absolute and Relative Clustering Tasks

In the introduction, we showed that reference matching problems are absolute and that noun coreference problems are relative. Below we show additional examples.

The goal of an image segmentation [9] is to divide a set of image primitives (e.g., pixels, line segments) into clusters that correspond to the same real entity. For example, for a portrait image, pixels that represent a specific person are clustered together. Whether two pixels belong to the same region or not depends on the other pixels around these two. Therefore, image segmentation is considered a relative clustering task.

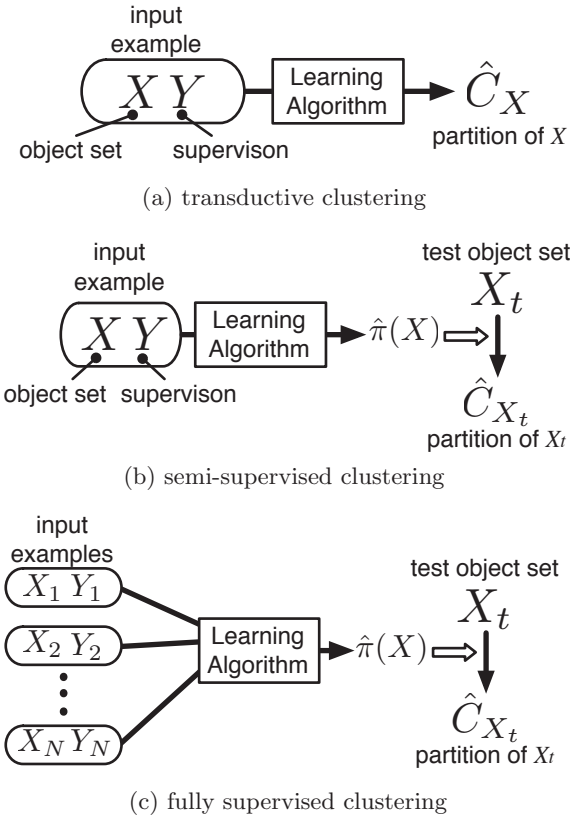
Document clustering can be classified into two cases. In the first case, the categorization of topics is given in advance and fixed. If clusters consist of documents on the same topic, each topic in the fixed categorization corresponds to one cluster in an absolute partition. This type of document clustering is the absolute type [20]. In the second case, there are multiple kinds of topic categorizations, and documents are clustered based on one of the categorizations that would best appropriately summarize the document set. Consider the documents of biology. If a topic of half of a number of documents is mammalian and that of the other half is reptilian, it would be appropriate to cluster the documents into mammalian and reptilian topics. However, a topic of all documents is mammalian, it would not be appropriate to gather all documents into a mammalian topic. In this case, it would be more appropriate to use mammalian sub-categories, such as primates or rodents. In this case, because the assignment of a pair of articles depends on the other components of the document set, this clustering task is relative clustering.

Word sense disambiguation is the task of dividing words having the same spelling into clusters of uses of words having the same semantics. For example, the word "bank" can be placed in the cluster of financial meaning or in that of geometrical meaning. If each target word is represented by a vector that contains all the information about the context of the word, the word can be categorized, without depending on the other words, into appropriate semantic categories. Hence, this word sense disambiguation is absolute clustering.

The prediction of DNA sequence splicing can be considered a kind of clustering. The problem is to segment a DNA sequence into exons (coding regions) and introns (non-coding regions). To determine whether two nucleotides are assigned to the same region or not, one must consider the other nucleotides in the DNA sequence. Therefore, this DNA sequence splicing task is relative clustering.

3. TYPES OF SUPERVISED CLUSTERING

To give a systematic point of view, we try to categorize supervised clustering problems. As described above, this categorization is closely related to notions of absolute and relative clustering tasks. First, we classify clustering tasks employing supervision as either constrained clustering or supervised clustering. In this paper, if supervision information is generalized, the clustering is supervised; otherwise, it is constrained. We take an example to show this notion more clearly. Wagstaff et al.'s COP-KMEANS [17] adopts must/cannot links as the format of supervision. Must-(resp. cannot-)linked pairs of objects should be assigned to the same cluster (resp. different clusters). In this method, link information is not generalized. That is, no information propagates to any objects that lie in the neighbor regions of



(a) transductive clustering

(b) semi-supervised clustering

(c) fully supervised clustering

Figure 1: Three types of supervised clustering problems

the linked objects. Therefore, we treat this method as constrained clustering. There are several types of constraints, such as the connectivity of objects [7] or the existence of obstacles between objects [6, 16].

A supervised clustering is defined as a clustering problem guided by the user's supervision, and the supervision is further generalized in the process of clustering. Based on the format of input examples and the goal of the algorithm, we further classify this supervised clustering into three types: (a) transductive clustering, (b) semi-supervised clustering, and (c) fully supervised clustering. These are summarized in Figure 1.

The input format of *transductive clustering* (Figure 1(a)) is (X, Y), where $X \subset \mathcal{X}$ is an object set and Y is a set of supervisions for X. An example of Y is a set of object pairs that are connected by must links. These supervisions are generally assigned not to all the objects but to some of the objects in X. The goal of this transductive clustering is to divide the given X into an appropriate partition, $\hat{C}_X$, guided by the given supervision Y. We should notice that any new objects that are not included in X cannot be clustered in this transductive case. This is similar to transductive learning [4] for a classification task.

The input format of *semi-supervised clustering* (Figure 1(b)) is the same as that of a transductive clustering, but the goal is different. The goal of semi-supervised clustering is to estimate a clustering function, $\hat{\pi}(\cdot)$, that outputs the appropriate partition, $\hat{C}_{X_t}$, for any test object set, $X_t \subseteq \mathcal{X}$, without direct reference to the supervision, Y. This function

is learned from the given training set, (X, Y).

The goal of *fully supervised clustering* (Figure 1(c)) is the same as that of a semi-supervised one, but the input format differs. The input of a fully supervised clustering is a set, $\{(X_1, Y_1), (X_2, Y_2), \ldots, (X_N, Y_N)\}$, where $X_1, \ldots, X_N \in \mathcal{X}$ are object sets and Y_i is a set of supervisions for the X_i. We here want to insist that the supervision information, Y_i, is valid only for the object set X_i. For example, assume that Y_1 contains a must link that connects two objects, $\mathbf{x}_1, \mathbf{x}_2 \in X_1$. Even if both $\mathbf{x}_1$ and $\mathbf{x}_2$ are members of X_2, this must link in Y_1 might not be valid for these objects in X_2.

Note that in semi-supervised and fully supervised clustering, it is desirable for the estimated function $\hat{\pi}(X)$ to satisfy equation (1), if the true clustering function $\pi^*(X)$ is absolute. However, this is not a necessary condition. This is similar to the case of unbiasedness in a regression task. The unbiasedness property is desirable, but in some cases, an estimator with a lower error can be acquired by ignoring this property. Therefore, the user should determine whether the estimated clustering function satisfies equation (1) depending on his/her choice.

We summarize the differences between the three types of supervised clusterings as follows. First, the goal of transductive clustering is to partition a given object set, while the goals of the other two types are to acquire a clustering function that can partition any new set of objects. Further, semi-supervised and fully supervised clusterings have different input examples formats. In the former, inputs are a single object set, X, and a set of supervisions, Y. In the latter, the algorithm is given as a set of tuples, $\{(X_1, Y_1), (X_2, Y_2), \ldots, (X_N, Y_N)\}$.

3.1 Examples of Three Types of Supervised Clusterings

We next show examples of the above three types of supervised clusterings. An example of a transductive clustering is MPCK-Means [2]. The objective function of the MPCK-Means method is

$$\begin{aligned}\mathcal{J}_{\mathrm{mpckm}} = &\sum_{\mathbf{x}\in X}\Big(\|\mathbf{x}_i - \boldsymbol{\mu}_{l_i}\|^2_{\mathbf{A}_{l_i}} - \log(\det(\mathbf{A}))\Big)\\ &+ \sum_{(\mathbf{x}_i,\mathbf{x}_j)\in Y_{\mathcal{M}}} w_{ij} f_M(\mathbf{x}_i, \mathbf{x}_j)\mathbb{I}[l_i \neq l_j]\\ &+ \sum_{(\mathbf{x}_i,\mathbf{x}_j)\in Y_{\mathcal{C}}} \bar{w}_{ij} f_C(\mathbf{x}_i, \mathbf{x}_j)\mathbb{I}[l_i = l_j]. \qquad (3)\end{aligned}$$

This method is outlined as follows. The first term of equation (3) represents the fitness to data. The second and third terms are penalties for the violation of must links and cannot links, respectively. $Y_{\mathcal{M}}$ and $Y_{\mathcal{C}}$ are sets of object pairs that are connected by must-links and cannot-links, respectively. The input of the MPCK-Means method is X together with supervisions Y, consisting of these $Y_{\mathcal{M}}$ and $Y_{\mathcal{C}}$. The goal of this method is to find an appropriate partition of X so as to optimize the objective function. Therefore, this method is considered a kind of transductive clustering. Further examples of this type are [19, 12, 11].

There is a supervised clustering approach by which a distance function is first learned, and an object set is clustered based on this learned distance function. Such methods are considered semi-supervised clustering. As an example, we summarize Xing et al.'s method [18]. The distance function $d(\mathbf{x}_i, \mathbf{x}_j)$ between two objects, $\mathbf{x}_i$ and $\mathbf{x}_j$, is defined as

$$d(\mathbf{x}_i, \mathbf{x}_j) = \|\mathbf{x}_i - \mathbf{x}_j\|_{\mathbf{A}} = \sqrt{(\mathbf{x}_i - \mathbf{x}_j)^\top A(\mathbf{x}_i - \mathbf{x}_j)},$$

where $\mathbf{A}$ is a semidefinite matrix. The supervision information, Y, consists of must-linked and cannot-linked object pairs, $Y_{\mathcal{M}}$ and $Y_{\mathcal{C}}$, as in the above MPCK-Means method. Matrix $\mathbf{A}$ is learned so as to optimize the following objective function:

$$\begin{aligned}&\min_{\mathbf{A}} \textstyle\sum_{(\mathbf{x}_i,\mathbf{x}_j)\in Y_{\mathcal{M}}} \|\mathbf{x}_i - \mathbf{x}_j\|_{\mathbf{A}}{}^2 \qquad (4)\\ &\text{s.t.} \sum_{(\mathbf{x}_i,\mathbf{x}_j)\in Y_{\mathcal{C}}} \|\mathbf{x}_i - \mathbf{x}_j\|_{\mathbf{A}} \geq 1, \text{ and } \mathbf{A} \text{ is semidefinite.}\end{aligned}$$

Once the distance function is learned, for any pair of objects in any object set, $X_t \in \mathcal{X}$, the distances between the objects can be calculated. Based on these distances, an appropriate partition for X_t can be derived by using unsupervised clustering algorithms. That is to say, the clustering function is composed of the acquired distance function and an unsupervised clustering algorithm. Further examples of this type are [14, 15, 10, 13, 1].

Let us stress again that transductive and semi-supervised clustering differ in their goals. The goal of the former is partitioning the given X, while that of the latter is acquiring a clustering function. Therefore, even after X is partitioned, a transductive clustering method, such as MPCK-Means, must be re-executed to partition another set X', and supervision information for X' has to be provided. On the other hand, in the semi-supervised case, such as [18], any unseen object sets can be clustered without referring to any supervision information. Accordingly, a semi-supervised clustering method must be used to partition any new object set. On the other hand, for the aim of partitioning a given object set, the transductive clustering method is preferred, because generally speaking, one should not solve an overly difficult problem. On the other hand, in order to partition a given object set, the transductive clustering method is preferred because, generally speaking, one should not solve an over-difficult problem.

Finally, as an example of fully supervised clustering methods, we briefly show Kamishima et al.'s method [9]. Each object set is represented by four types of attribute vectors.

- the object attribute, $\mathbf{x}_i$, represents features of objects, such as the location of points.
- the pair attribute, $\mathbf{p}_{ij}$, represents features between two objects, $\mathbf{x}_i$ and $\mathbf{x}_j$, such as the similarity between these points.
- the cluster attribute, $\mathbf{c}_k$, represents features of a specific cluster, such as the number of objects in the cluster.
- the partition attribute, $\mathbf{C}$, represents the features of the entire partition, such as the number of clusters.

In the learning stage, the following joint distribution is acquired:

$$\Pr[C_X{=}C_X^*, \{\mathbf{x}_i\}, \{\mathbf{p}_{ij}\}, \{\mathbf{c}_k\}, \mathbf{C}], \qquad (5)$$

where $C_X{=}C_X^*$ denotes an event where the partition C_X is equivalent to the appropriate partition C_X^*. This distribution is approximated by the product of three types of probabilistic mass/density distributions: $\Pr[C_X{=}C_X^*|\{\mathbf{x}_i\},$

$\{\mathbf{p}_{ij}\}]$, $\Pr[\{\mathbf{c}_k\}|C_X{=}C_X^*]$, and $\Pr[\mathbf{C}|C_X{=}C_X^*]$. Once this distribution function is learned, for any object set, X_t, its appropriate partition can be found so as to maximize this distribution. Further examples of this type are [5, 8].

4. EFFECTS OF ABSOLUTE AND RELATIVE CLUSTERING

As described in the introduction, depending on whether the real clustering task is absolute or relative, three points must be considered in the formalization of the task: the representations of objects, types of supervisions, and information provided by the features. We then describe each of these points.

4.1 Three Types of Supervised Clusterings

We here summarize the relationships types of real clustering tasks in section 2 and those of supervised clustering problems in section 3. If the goal of a real task is to partition the given object set, the task should be formulated as transductive clustering, regardless of whether the task is relative or absolute. Otherwise, the goal is to acquire a clustering function. In this case, absolute and relative clustering tasks must be formulated as semi-supervised and fully supervised clusterings, respectively. Below, we describe the reason for this.

First, we consider a transductive case. The distinction between absolute and relative clustering becomes apparent when the contents of an object set change. In the example of noun coreference (see the introduction), the assignment of phrases was changed when sentence [B] was eliminated. Because the contents of an object set are fixed in this transductive case, there is no need to differentiate between absolute and relative clustering.

Next, to learn a clustering function for a relative clustering task, the task must be formulated as a fully supervised clustering problem. In this case, because equation (1) is not satisfied, the supervision information, Y_i, is valid only for the object set, X_i. That is to say, even if two objects, $\mathbf{x}_1$ and $\mathbf{x}_2$, are contained in both X_1 and X_2, the must link $(\mathbf{x}_1, \mathbf{x}_2) \in Y_1$ might not be valid for X_2. Therefore, a relative clustering task must be solved by a fully supervised approach.

Finally, to learn a clustering function for an absolute clustering task, the task should be formulated as a semi-supervised clustering problem. If the format of the input examples is the fully supervised type, a set of paris of object sets and supervisions should be converted by transforming $X = X_1 \cup \cdots \cup X_N$ and $Y = Y_1 \cup \cdots \cup Y_N$. In this case, due to the property of equation (1), must and cannot links for $X_1, \ldots, X_N$ are guaranteed to be valid in X. Therefore, by using the transitivity of must and cannot links described in section 2, more supervision information can be obtained. Therefore, an absolute clustering task should be solved by a semi-supervised approach. Note that it is possible to formalize a relative clustering task as a fully supervised problem, but it is inefficient. This is because given supervisions can be augmented by merging object sets as above, and algorithms for fully supervised problems are generally slower than those for semi-supervised ones.

Two objects may be connected by both of must and cannot links after the merger. Even in the semi-supervised case, noise in the supervision might cause such an inconvenience. In this case, it would be a good idea to add weights to these links. For example, when $\mathbf{x}_1$ and $\mathbf{x}_2$ are connected by two must links and one cannot link, these two objects are connected by a must link with 2/3 weight and a cannot link with 1/3 weight. On the other hand, for a relative clustering task, such a weighted approach should not be adopted. This is because such disagreements in must and cannot links are caused not by noise but by an intrinsic property of the task.

4.2 The Type of Supervision

Labels are use occasionally to represent a supervision by assigning the same label to objects in the same cluster. If all possible labels are known *a priori*, the task is considered a semi-supervised learning/classification [3]. However, if some of the labels are unknown, the task can be considered a kind of supervised clustering [20].

We next claim that such a label style cannot be used for a relative clustering task. In label-style supervision, all the objects having the same label should be assigned to the same cluster independently of the other objects. That is to say, such objects should be assigned to a specific absolute cluster (see definition 2). If such an absolute cluster exists, the task must be absolute clustering.

4.3 Information Provided by Features

To cluster a set of objects, the set must be represented by features. We next discuss what kind of information should be represented by features.

In the case of absolute clustering, to obtain an appropriate partition, all that we have to do is to find the relationship between each object and an unknown absolute cluster. Because such relationships should be findable by considering the features of each object, it is sufficient to represent the objects by their features. In our example of a reference matching problem (see the introduction), given one reference string, i.e, the features of the object, the string should be related to some paper that corresponds to a hidden absolute cluster, without the need to check any other citation strings.

In the case of relative clustering, to cluster an object, we must consider its relationships to the other objects. To perform a noun coreference example (see the introduction), anaphora must be resolved. To determine whether phrase (1) is an antecedent of phrase (4), phrases (2) and (3) must be taken into account. Therefore, to perform a relative clustering task, at least the features between pairs of objects have to be provided. In this example, features assigned between the two phrases should convey information, e.g., the number of phrases between these two phrases or an order of them.

5. CONCLUSIONS

In this paper, we proposed a distinction between absolute and relative clustering tasks, which are determined by the condition of equation (1). Next, we systematically classified supervised clustering problems as transductive, semi-supervised, and fully supervised. If the goal of a task is to partition a given object set, the task should be formulated as transductive clustering. If the goal is to learn a clustering function, the task should be formulated as a semi-supervised problem for absolute clustering, and as a fully supervised problem for relative clustering. Additionally, when formulating a relative clustering task, label-type supervision can-

not be used, and features to represent relationships among objects must be provided.

6. ACKNOWLEDGMENTS

This work is supported by MEXT/JSPS KAKENHI Grant Number 16700157, 21500154, 23240043, 24500194, and 25540094.

7. REFERENCES

[1] A. Bar-Hillel, T. Hertz, N. Shental, and D. Weinshall. Learning a mahalanobis metric from equivalence constraints. *Journal of Machine Learning Research*, 6:937–965, 2005.

[2] M. Bilenko, S. Basu, and R. J. Mooney. Integrating constraints and metric learning in semi-supervised clustering. In *Proc. of the 21st Int'l Conf. on Machine Learning*, pages 81–88, 2004.

[3] O. Chapelle, B. Schölkopf, and A. Zien, editors. *Semi-supervised Learning*. MIT Press, 2006.

[4] O. Chapelle, V. Vapnik, and J. Weston. Transductive inference for estimating values of functions. In *Advances in Neural Information Processing Systems 12*, pages 421–427, 2000.

[5] H. Daumé, III and D. Marcu. A bayesian model for supervised clustering with the dirichlet process prior. *Journal of Machine Learning Research*, 6:1551–1577, 2005.

[6] V. Estivill-Castro and I. Lee. Autoclust+: Automatic clustering of point-data sets in the presence of obstacles. In *Proc. of the 1st Int'l Workshop on Temporal, Spatial, and Spatio-Temporal Data Mining*, pages 133–146, 2001. [LNCS 2007].

[7] A. Ferligoj and V. Batagelj. Clustering with relational constraint. *Psychometrika*, 47(4):413–426, 1982.

[8] T. Finley and T. Joachims. Supervised clustering with support vector machines. In *Proc. of the 22nd Int'l Conf. on Machine Learning*, pages 217–224, 2005.

[9] T. Kamishima and F. Motoyoshi. Learning from cluster examples. *Machine Learning*, 53:199–233, 2003.

[10] D. Klein, S. D. Kamvar, and C. D. Manning. From instance-level constraints to space-level constraints: Making the most of prior knowledge in data clustering. In *Proc. of the 19th Int'l Conf. on Machine Learning*, pages 307–314, 2002.

[11] N. Kumar, K. Kummamuru, and D. Paranjpe. Semi-supervised clustering with metric learning using relative comparisons. In *Proc. of the 5th IEEE Int'l Conf. on Data Mining*, pages 693–696, 2005.

[12] A. McCallum and B. Wellner. Conditional models of identity uncertainty with application to noun coreference. In *Advances in Neural Information Processing Systems 17*, pages 905–912, 2005.

[13] D. Mochihashi, G. Kikui, and K. Kita. Learning nonstructural distance metric by minimum cluster distortions. In *Proc. of the Empirical Methods in Natural Language Processing (EMNLP)*, pages 341–348, 2004.

[14] S. Oyama and K. Tanaka. Learning a distance metric for object identification without human supervision. In *Proc. of the 10th European Conf. on Principles of Data Mining and Knowledge Discovery*, pages 609–616, 2006. [LNCS 4213].

[15] M. Schultz and T. Joachims. Learning a distance metric from relative comparisons. In *Advances in Neural Information Processing Systems 16*, pages 41–48, 2004.

[16] A. K. H. Tung, J. Hou, and J. Han. Spatial clustering in the presence of obstacles. In *Proc. of the 17th Int'l Conf. on Data Engineering*, pages 359–367, 2001.

[17] K. Wagstaff, C. Cardie, S. Rogers, and S. Schroedl. Constrained k-means clustering with background knowledge. In *Proc. of the 18th Int'l Conf. on Machine Learning*, pages 577–584, 2001.

[18] E. P. Xing, A. Y. Ng, M. I. Jordan, and S. Russell. Distance metric learning, with application to clustering with side-information. In *Advances in Neural Information Processing Systems 15*, pages 521–528, 2003.

[19] S. X. Yu and J. Shi. Segmentation given partial grouping constraints. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 26(2):173–183, 2004.

[20] S. Zhong. Semi-supervised model-based document clustering: A comparative study. *Machine Learning*, 65:3–29, 2006.