Vol.2017-IFAT-125 No.10
Vol.2017-DC-104 No.10
2017/3/10

An Experiment on Simple and Practical Methods of Cluster Labeling for Hierarchically Organized Document Subsets

Kazuaki KISHIDA[1,a)]

Abstract: This paper reports the results of an experiment on cluster labeling for a hierarchical tree of document subsets which were generated by the Hierarchical Multi-way Divisive Clustering (HMDC) algorithm. The data used was a set of 6,374 news articles extracted from the RCV-1 test collection (Reuters Corpus). The tree contained 2,252 clusters (nodes), to which labels were assigned by selecting terms from those included in documents of the target cluster. More specifically, the terms (noun phrases or their components) in the documents were ranked by three well-known weighting methods, and three top ranked terms were used as labels of the cluster. In the experiment, three sets of labels selected by the three methods, respectively, were comparatively assessed by a human reviewer for the set of 1,201 clusters, and the degrees of goodness or badness of these labels were measured by each cluster.

1. Introduction

Document clustering (DC) is often useful for detecting a topical structure inherent in a heterogeneous set of news articles, scientific papers, patents, web pages and so on. For example, a set of web pages returned by a search engine for a given query may often be concerned with several different topics partly because the search terms entered by the user are semantically ambiguous. So-called 'clustering search engines' display the result of applying a DC algorithm to the topically heterogeneous set (i.e., *search results clustering*), which allows the users to specify easily a subset of the pages relevant to their needs.

When automatically generated clusters are presented to the users in an application, it is desirable that appropriate labels or descriptions are attached to each cluster so that the clustering result can be browsed efficiently. This is usually called *cluster labeling*, for which effective and efficient methods have been explored so far by many researchers (see Section 2).

This paper reports on an experiment in which simple and practical labeling methods were applied to a hierarchical tree of clusters generated from a 'large-scale' set of documents. If the target document set is small, then it may be possible to apply a complicated algorithm for the clustering and labeling. However, in the case of large sets, computational complexity of the algorithm becomes a critical issue. Such situation occurs often in topic detection tasks when the target is a medium- or large-scale document set.

In the experiment, the Hierarchical Multi-way Divisive Clustering (HMDC) algorithm developed by Kishida(2014) [15] was applied to a set of news articles extracted from a well-known test collection RCV1 (Reuter Corpus) [21] for obtaining a hierarchal tree of clusters because the HMDC algorithm is suitable for large-scale document sets. Next, labels were assigned to each cluster in the tree by applying 'simple' term weighting methods. Note that this paper does not propose a new algorithm for cluster labeling. Rather, the main purpose is to show a practical way of clustering and labeling for a large-scale document set. Actually, in the experiment, only well-known algorithms or methods were used with some special heuristic rules for obtaining good labels. Since the HMDC algorithm and other hierarchical clustering methods have already been discussed in Kishida(2014) [15], this paper reviews only cluster labeling methods (Section 2). The experiment is reported in Section 3 .

2. Automatic Assignment of Cluster Labels

2.1 Cluster labeling problem

Let a set of N documents be denoted by

$$D = \{d_1, d_2, \ldots, d_N\}, \tag{1}$$

and assume that the set is divided into L clusters $C_1, \ldots, C_L$ by a clustering algorithm, namely $D = C_1 \cup C_2 \cup \cdots \cup C_L$. After obtaining such clusters, a few words or phrases representing contents of the clusters may have to be automatically identified as cluster labels in some applications as mentioned above.

Cluster labeling is similar to *feature selection* in text categorization or DC because both of them try to find subject representations that (1) reflect 'accurately' a topic of the target cluster (or class), and that (2) appear 'distinctively' in the target cluster and rarely appear in the other clusters. However, in addition to these, (3)'conciseness' and (4)'comprehensibility' (or 'transparency') should be considered as requirements of cluster labels

[1] School of Library and Information Science, Keio University, Minato-ku, Tokyo 108–8345, Japan
[a)] kz_kishida@keio.jp

(see Zhang et al., 2009 [51]). Namely, cluster labels must be understood easily and correctly by human beings. Therefore, a cluster label should be short enough (i.e., conciseness) and allow users to imagine sufficiently the actual contents of the cluster (i.e., comprehensibility), which makes the cluster labeling problem more difficult.

Some researchers have pointed out that phrases are more desirable as labels than single words in term of comprehensibility. For example, "machine learning" would be better than "learning, machine" in which two single words "learning" and "machine" are simply enumerated in alphabetical order. This example suggests that simple automatic indexing based on the bag-of-words architecture in the field of information retrieval (IR) is insufficient for cluster labeling.

2.2 Sources of cluster labels

There are two typical sources from which cluster labels are selected:

- Documents belonging to the target cluster, and
- External sources such as WordNet, Wikipedia and so on.

A simple and practical way of determining automatically cluster labels would be to detect appropriate terms from the text of documents in the target cluster. The basic procedure is as follows.

(1) Extract words or phrases from the text of each document.
(2) Assign a weight to each word or phrase.
(3) Sort words or phrases by the weights.
(4) Adopt top-ranked words or phrases as labels.

When the algorithm for clustering estimates explicitly the weight of each word or phrase in the process (e.g., spectral co-clustering [8], fuzzy co-clustering [16], NMF-based clustering [48], SKWIC algorithm [10] and so on), it may be possible to use it for the labeling. Unless such weight is available, a special weight for the labeling has to be computed from statistics on *term frequency* (tf), *inverse document frequency* (idf) and so on (see Section 2.3 for details). The weighting method is often independent of the clustering algorithm to be applied [*1].

In the case of selecting relevant labels from external sources such as Wikipedia, WordNet and so on, the degrees of relationship between terms in the external source and words of documents in each cluster are measured. For instance, Hotho & Stumme (2002) [14] specified 'synsets' of WordNet having a close relation with the centroid of each cluster, and adopted them as labels of the cluster. Additionally, hypernym relations defined in WordNet were used by Lau et al.(2010) [19] for selecting appropriate labels from candidate words. In Lau et al.(2011) [20], titles of Wikipedia article were employed for specifying cluster labels, and similarly, Nayak et al.(2014) [34] tried to select the labels from subject categories of Wikipedia.

2.3 Term weighting for selection of labels from documents

2.3.1 Use of weights in cluster vectors

The simplest way of weighting words or phrases in documents is to examine cluster vectors or profiles that are generated by clustering algorithms if they are available. In the Scatter/Gather system (Cutting et al., 1992 [6]; 1993 [7]), several words with high weights in the cluster's profile were automatically displayed as a cluster summary.

When executing the clustering algorithms, stopwords are removed and the other words are automatically stemmed. Sometimes, phrases are extracted from the target text by applying a technique such as *part-of-speech* (POS) tagging. The word stem or a sequence of the stems (i.e., phrase) is often called an *index term*, which corresponds to an element of cluster vectors. In this paper, the index terms are denoted by t_j ($j = 1, \ldots, M$) where M indicates the total number of different index terms in D.

After calculating a weight of term t_j in document d_i based on a formula, which is written by w_{ij} in this paper, the jth element of the kth cluster vector ($k = 1, \ldots, L$) is typically computed such that

$$\tilde{w}_{jk} = \frac{1}{\tilde{n}_k} \sum_{i:d_i \in C_k} w_{ij}, \quad i = 1, \ldots, N; j = 1, \ldots, M \tag{2}$$

where $\tilde{n}_k = |C_k|$ (i.e., the number of documents belonging to C_k). An M-dimensional vector $\tilde{\mathbf{w}}_k = [\tilde{w}_{1k}, \ldots, \tilde{w}_{Mk}]^T$ is usually called a *cluster centroid*. Terms with the highest values in the cluster centroid may become candidates of cluster labels.

The value of w_{ij} can be actually computed according to a tf-idf weighting scheme in IR theory, a basic version of which is

$$w_{ij} = x_{ij} \log(N/n_j) \tag{3}$$

where x_{ij} denotes the occurrence frequency of t_j in d_i ($i = 1, \ldots, N$; $j = 1, \ldots, M$), and n_j means the number of documents including t_j in D. Note that n_j should be considered as a 'global' document frequency (df) because a 'local' df can be defined as the number of documents including t_j within a particular cluster in the case of DC, which is written as $n_{j|k}$ in this paper ($j = 1, \ldots, M$; $k = 1, \ldots, L$).

2.3.2 Weighting based on tf-idf scheme

Independently of cluster vectors used for partitioning the set D, the tf-idf weighting scheme can be directly applied for selecting cluster labels from a set of index terms (see Tonella et al., 2003 [43]). For instance, if $\mathrm{f}_{kj} \equiv \sum_{i:d_i \in C_k} x_{ij}$ and

$$S(j|k) = \mathrm{f}_{kj} \log(N/n_j), \quad j = 1, \ldots, M; k = 1, \ldots, L \tag{4}$$

are computed similarly with Equation (3), then it is possible to rank the terms in each cluster by scores of $S(j|k)$ and to adopt the top-ranked terms as labels.

Otherwise, the score may be defined such that

$$S(j|k) = \mathrm{f}_{kj} \log(L/c_j), \quad j = 1, \ldots, M; k = 1, \ldots, L \tag{5}$$

where c_j denotes the number of clusters in which t_j appears (see Ayad & Kamel, 2002 [2] or Maqbool & Babri, 2005 [26]). The term $\log(L/c_j)$ is expected to increase the weights of specific terms occurring in only a few clusters, which may help to avoid selecting non-specific terms as labels like the standard idf factor in Equation (4).

Also, Popescul & Ungar (2000) [36] employed 'predictiveness' which was computed by $\mathrm{f}_{kj} / \sum_{k'=1}^{L} \mathrm{f}_{k'j}$, which can be interpreted as a kind of idf factor measuring the specificity of term t_j

[*1] In Matsumoto & Hung (2010) [27], a value of each term that was computed for the clustering process was combined with the tf value when determining cluster labels.

in cluster C_k [*2]. According to an idea by Lamirel(2013) [18], a label for cluster C_k may be selected by calculating the harmonic mean of $\mathrm{f}_{kj}/\sum_{k'=1}^{L}\mathrm{f}_{k'j}$ and $\mathrm{f}_{kj}/\sum_{j'=1}^{M}\mathrm{f}_{kj'}$.

2.3.3 Use of reference corpus

When trying to identify specific terms inherent in a given cluster, it is natural to examine the degree to which they appear in another document set, which is often called a *reference corpus*. In this paper, the reference corpus for cluster C_k is denoted by R_k, which is typically defined such that

$$R_k = D \backslash C_k. \tag{6}$$

For instance, if the term occurrence in R_k is incorporated into tf-idf weighting, then Equation (4) becomes

$$S(j|k) = \frac{\mathrm{f}_{kj}}{\sum_{i:d_i \in R_k} x_{ij}} \times \log \frac{N}{n_j} \tag{7}$$

based on a suggestion in Chuang et al.(2012) [5].

Another way of finding terms inherent in a particular cluster based on a reference corpus is to compile a distribution $f(x)$ of frequencies of each term ($x = 0, 1, 2, \ldots$) in C_k and R_k, respectively, and to compare the two distributions. More specifically, for a particular value of x (e.g., $x = 1$), $f(x)$ represents the relative frequency of documents in which the target term appears x times. For instance, the statistical χ^2 test may be used for evaluating the degree of difference between the two distributions as suggested by Popescul & Ungar (2000) [36]. Namely, if the null hypothesis that two distributions are generated independently is not rejected (e.g., at the significance level of 5%), then the term can be considered to be an inappropriate label. Also, *Jensen-Shannon Divergence* (JSD), which is a metric for measuring the distance between two distributions, can be applied to the labeling problem (see Carmel et al., 2009 [4]; Muhr et al.,2010 [32]; or Roitman et al., 2014 [37]).

Additionally, when using the reference corpus, feature selection techniques for text categorization may be applied by considering C_k and R_k as positive and negative cases, respectively. A typical measure for the selection is *information gain* (IG), which was modified for cluster labeling such that

$$IG_m = P(t, C) \log \frac{P(t, C)}{P(t)P(C)} + P(\bar{t}, \bar{C}) \log \frac{P(\bar{t}, \bar{C})}{P(\bar{t})P(C)} \tag{8}$$

in Geraci et al.(2006) [12] by removing factors of negative correlation. Note that $P(t, C)$ means the probability that the target cluster includes term t whereas $P(\bar{t}, \bar{C})$ indicates the probability that t does not appear in the reference corpus. When the negative correlation works well in standard IG, terms that appear rarely in C_k and occur frequently in R_k may have a high value, which does not indicate there appropriateness as cluster labels.

If $R_k = D \backslash C_k$, then $P(t, C)$ in Equation (8) can be operationally defined as $n_{j|k}/N$, which is the proportion of documents including t_j and belonging to C_k. By determining similarly the other probabilities, a score based on the modified IG becomes

$$g_{j|k} = \frac{n_{j|k}}{N} \log \frac{n_{j|k}N}{n_j \tilde{n}_k} + \frac{N - n_{j|k}}{N} \log \frac{(N - n_{j|k})N}{(N - n_j)(N - \tilde{n}_k)}. \tag{9}$$

[*2] Other term statistics for identifying important keywords from text were discussed in Chuang et al.(2012) [5].

Table 1 Contingency table

	Cluster: C_k	Reference: R_k	Total
t_j appears	a_{11}	a_{12}	$a_{1\cdot}$
t_j does not appear	a_{21}	a_{22}	$a_{2\cdot}$
Total	$a_{\cdot 1}$	$a_{\cdot 2}$	$a_{\cdot\cdot}$

The numbers included in Equation (9) are obtained as a result of compiling a contingency table shown in Table 1 (i.e., $n_{j|k} = a_{11}$, $n_j = a_{1\cdot}$, $\tilde{n}_k = a_{\cdot 1}$ and $N = a_{\cdot\cdot}$). This means that various correlation coefficients computed from the 2×2 contingency table can be used for selecting cluster labels. For instance, Tseng et al. (2006) [45] and Tseng (2010) [44] adopted a 'four-fold correlation coefficient' [*3],

$$r_{j|k} = \frac{a_{11}a_{22} - a_{12}a_{21}}{\sqrt{(a_{11} + a_{12})(a_{21} + a_{22})(a_{11} + a_{21})(a_{12} + a_{22})}}, \tag{10}$$

and determined the rank of t_j for C_k according to $S(j|k) = \mathrm{f}_{kj} \times r_{j|k}$, in which the correlation works as a kind of idf factor. Otherwise, it may be possible to exploit a $2 \times L$ contingency table whose inner cells are $a_{1k} = n_{j|k}$ and $a_{2k} = \tilde{n}_k - n_{j|k}$ ($k = 1, \ldots, L$) straightforwardly (see Moura et al., 2008 [29]).

The χ^2 statistic is computed by $N \times r_{j|k}^2$, which can be used to test statistically whether the relationship between the term occurrence and the partitioning to C_k and R_k is significant or not. However, because the χ^2 statistic, like the standard IG, does not discern negative correlation from positive correlation, it is better to depend on $r_{j|k}$ for label selection (see Tseng et al., 2006 [45] and Tseng, 2010 [44]). Actually, Moura & Rezende (2007) [30] and Moura et al. (2008) [29] have also adopted metrics measuring the dependency in a contingency table other than the χ^2 statistic (see Moura et al., 2008 [29] about the metrics).

2.3.4 Combining term characteristics

Treeratpituk & Callan(2006) [46] proposed 'DScore' (descriptive score) that was computed by a linear function $y = b_0 + b_1 x_1 + \cdots + b_{10} x_{10}$ where the set of independent variables consists of a tf-idf weight, a local idf factor (normalized document frequency within the cluster), rank by tf-idf weight, phrase (term) length and so on. The DScore was originally developed for predicting the appropriateness of a given term as a label of a cluster in a hierarchy (the labeling for a cluster hierarchy is discussed below).

Similarly, a technique explored by Zhang & Xu (2008) [52] and Zhang et al.(2009) [51] exploits a function having many variables to discriminate whether a term should be selected as a cluster description or not. Interestingly, information on the location where the term appeared was incorporated into some variables (e.g., "percentage of documents whose title contains the term in a cluster"). In practice, Zhang & Xu (2008) [52] and Zhang et al.(2009) [51] employed a statistical *machine learning* approach (e.g., SVM) for constructing empirically the discriminating functions. The machine learning approach was also explored by Lau et al.(2010; 2011) [19], [20].

2.4 Use of phrases as as cluster labels

A phrase consisting of multiple single words often becomes a more 'comprehensible' label of a cluster, and therefore, some researchers have tried to generate automatically phrase-based labels

[*3] The coefficient is defined as a Pearson product-moment correlation coefficient when the two variables are binary.

from the text of documents.

2.4.1 Identification of phrases as labels

Müller et al.(1999) [33] extracted pairs of words co-occurring frequently within a five-word window in the text of documents (e.g. "anti, virus"), and selected the five most frequent pairs as the cluster labels. Similarly, in Anaya-Sánchez et al.(2010) [1], pairs of words were treated as representations of document clusters, and appropriate word pairs were identified based on their occurrences in relevant and irrelevant documents that were determined by a complicated algorithm.

Mei et al.(2007) [28] compared 'chunking / shallow parsing' and 'n-gram testing' as approaches for extracting a set of phrases from text in the process of obtaining candidate labels. Similar methods were also used by Lau et al.(2011) [20]. The two studies aimed at labeling nodes that were generated by a topic model such as LDA (*latent Dirichlet allocation*) [*4].

Also, patterns of POS tags were applied by Li et al. (2015) [23] for generating 'readable' and 'informative' phrases as candidates of cluster labels. In the study, a graph showing the relationship within the set of terms and a candidate in the cluster was constructed, and final labels were selected based on the graph. Such kind of graph was also used by Scaiella et al.(2012) [39] for labeling in the case of search results clustering [*5].

2.4.2 Use of suffix tree

Zamir & Etzioni(1998) [49] used a suffix tree to create document clusters by treating the documents as strings of words. Because each node of the suffix tree corresponds to a phrase that is common to all documents belonging to the node, it is natural to adopt the phrase as a label of the document cluster.

2.4.3 Determining labels before clustering

Another approach is to determine cluster descriptions used as labels before clustering, and then, to allocate each document to a particular description, by which clusters of documents are generated as a result. For instance, 'Lingo algorithm' by Osiński & Weiss (2005) [35] exploits *latent semantic indexing* (LSI) for identifying beforehand phrases used as labels of snippet clusters. More specifically, the algorithm specifies phrases corresponding to some latent semantic components obtained by *singular value decomposition* (SVD) of a term-document matrix, components of which are tf-idf weights of individual words appearing frequently in the target set of snippets. After phrases are determined, each snippet is assigned to a phrase in the algorithm, which was called the 'description-comes-first' (DCF) approach in the paper [35].

Stefanowski & Weiss (2007) [40], [41] proposed another algorithm based on the DCF approach, in which a k-means algorithm works without LSI for a large-scale document collection. In the algorithm, a set of phrases is generated first as candidate labels by extracting them from the target document set and external resources. Concurrently, a k-means algorithm is executed for a sample extracted from the target document set, and cluster centroids in the result are recorded. After that, phrases having the highest similarity with each centroid are selected as final labels, respectively, and each document is allocated to one of the final labels. A modification of the DCF approach was also tried by Zhang (2009) [50].

An algorithm by Li et al.(2013) [22] generates first a graph representing 'sentence co-occurrences' of keywords which were extracted from text according to a result of POS tagging, and identifies groups of highly co-occurring keywords as 'communities'. In the next step, documents are allocated to relevant communities.

2.4.4 Use of itemsets

When data mining techniques are applied to DC, a cluster is usually constructed so as to include documents having a particular subset of words that were identified by an association rule. The subset is often called an *itemset*, which may contain two or more words. For instance, FTC (frequent term-based clustering) by Beil et al.(2002) [3] and FIHC (frequent itemset-based hierarchical clustering) by Fung et al.(2003) [11] are well-known as such kind of data mining techniques for DC. Since those studies were published, various DC techniques based on itemsets have been developed.

Naturally, itemsets can be adopted as cluster labels. In this case, the data mining technique is considered to determine cluster labels before grouping documents.

2.5 Labeling nodes in cluster hierarchy

2.5.1 Parent, child and sibling nodes

In hierarchical clustering, a tree consisting of nested nodes (i.e., *dendrogram*) is finally generated, and each node is usually interpreted as a cluster. Therefore, when assigning labels to each cluster, the relationships between a parent node, child nodes and 'sibling' nodes [*6] have to be considered. Glover et al. (2002) [13] defined three types of terms in a node of the tree as "self terms that describe the cluster as a whole, parent terms that describe more general concepts, and child terms that describe specializations of the cluster" (p.507 in [13]) [*7].

The structure of a cluster hierarchy is more complicated than the result of flat partitioning. For example, when $N = 4$, the entire set D can be divided into seven clusters such as

(1) Root node (D itself, i.e., $\{d_1, d_2, d_3, d_4\}$),
(2) Leaf nodes consisting of a single document (i.e., $\{d_1\}$, $\{d_2\}$, $\{d_3\}$ and $\{d_4\}$, which are called *singletons*), and
(3) Other 'interim' nodes (e.g., $\{d_1, d_2\}$ and $\{d_3, d_4\}$).

If $C_1 = \{d_1\}$, $C_2 = \{d_2\}$ and $C_5 = \{d_1, d_2\}$, then $C_5 = C_1 \cup C_2$ and $C_1 \cap C_2 \neq \emptyset$. Let the index number of a parent cluster of C_k be denoted by $p(k)$. In this example, $p(1) = p(2) = 5$.

2.5.2 Term selection in cluster hierarchy

The χ^2 test discussed above may be applied to label selection for a cluster hierarchy by checking the dependency of two distributions in children C_k and $C_{k'}$ of parent C_h where $p(k) = p(k') = h$ (see Popescul & Ungar, 2000 [36]). If the dependency of two distributions on a particular term t_j is detected, then t_j can be considered to represent a 'general' topic relevant to parent C_h and

[*4] Magatti et al.(2009) [25] used external sources (Google Directory and a thesaurus) for labeling clusters obtained from the topic model.

[*5] In Role & Nadif (2014) [38], a similar graph structure was also employed, but the graph itself was considered as a cluster representation.

[*6] The sibling nodes mean the other nodes having the same parent node.

[*7] Glover et al.(2002) [13] explored heuristic if-then rules for identifying self, parent and child terms based on occurrence frequencies. For instance, good self terms can be assumed to be contained commonly in documents of the relevant cluster, but to appear relatively rarely in the whole collection.

may become its label. Otherwise, t_j is a specific term of C_k or $C_{k'}$ and remains as a candidate for one of the child clusters. By starting this procedure at the root node and descending the hierarchy sequentially, it may be possible to obtain a set of cluster labels for a cluster hierarchy (see Popescul & Ungar, 2000 [36]).

Another strategy is to compute IG_m in Equation (9) or the correlation coefficient in Equation (10) for two sets C_k and $R_k = C_{p(k)} \setminus C_k$, and to select the top-ranked terms as labels of C_k ($k = 1, \ldots, L-1$) except for root node C_L (e.g., see Muhr et al., 2010 [32]). This technique can be applied to not only a binary tree but also a multi-branch tree because $C_{p(k)} \setminus C_k$ is a set of one or more siblings of C_k. For the multi-branch tree, the contingency table with two or more clusters described above may also be used (see Moura et al., 2008 [29]).

Otherwise, the term weighting methods discussed in Section 2.3.2 can be straightforwardly applied to label selection for a particular node in a cluster hierarchy. However, in the case of cluster hierarchy, it may be better to add a factor reflecting usage of the target term in the other sibling nodes. For instance, Muhr et al.(2010) [32] incorporated a 'local' idf factor such that

$$I_{GL}(j|k) = \log\left(\frac{N}{n_j} + 1\right) \times \log\left(\frac{\tilde{n}_{p(k)}}{n_{j|p(k)}} + 1\right) \tag{11}$$

where the second term in the right side is the local idf factor (the first is the global one). Then, a term weight was finally computed such that $S(j|k) = \mathrm{f}_{kj} \times I_{GL}(j|k)$ [*8]. When t_j appears frequently in sibling(s) of C_k, the local idf factor becomes smaller and the term weight tends to decrease.

Also, Muhr et al.(2010) [32] introduced another factor measuring an aspect of term t_j in C_k such that

$$I_C(j|k) = \log\left(\frac{c_{\cdot|p(k)}}{c_{j|p(k)}} + 1\right) \times \exp\left(\frac{n_{j|k}}{\tilde{n}_k}\right) \tag{12}$$

where $c_{\cdot|p(k)}$ denotes the number of child nodes belonging to the parent node of C_k and $c_{j|p(k)}$ indicates the number of those including t_j.

2.5.3 Incorporating path length factor

However, labels of a parent node may be assigned again to its children when using simply Equation (11). To overcome this problem, Muhr et al.(2010) [32] incorporated additionally a factor measuring the 'path length' between two clusters in the hierarchy into the formula for computing term weights (see Muhr et al., 2010 [32] for details).

Also, the path length was employed for calculating $S(j|k)$ by Mao et al.(2012) [24]. For instance, a weight of t_j for C_k for label selection was calculated such that

$$S(j|k) = m_{j|k} \times \sum_{h:C_h \subseteq C_k} \frac{\mathrm{f}_{hj} \times I_{GL}(j|h) \times I_C(j|h)}{l(h,k)} \tag{13}$$

where $m_{j|k}$ denotes the number of all clusters including t_j in a subtree having C_k as its root, and $l(h,k)$ indicates the number of links between C_h in the subtree and the root C_k (another weighting based on IG_m was also explored by Mao et al., 2012 [24]).

[*8] The DScore function by Treeratpituk & Callan (2006) [46] can be interpreted as another approach to estimation of $S(j|k)$ for a cluster hierarchy.

2.5.4 Other techniques for labeling a cluster hierarchy

Also, dos Santos et al.(2010) [9] attempted to construct association rules for labeling a cluster hierarchy under the assumption that each parent node influences its child nodes and the selection of labels for the parent should reflect this information. A hierarchical 'monothetic' clustering algorithm by Kummamuru et al. (2004) [17] tries to extract iteratively an important 'concept' from each document set according to a criterion, and consequently, a hierarchy of documents can be obtained by assigning each document to a relevant concept.

2.6 Other techniques for cluster labeling

Tholpadi et al.(2012) [42] applied a multilingual topic model (the 'Polylingual Topic Model') for labeling clusters that include documents written in different languages. Also, there are some attempts at cluster labeling for the self-organizing map (SOM) (e.g., see van Heerden & Engelbrecht, 2013 [47]).

Recently, Mu et al.(2016) [31] proposed a complicated method of 'descriptive document clustering' by which a set of clusters and their labels can be obtained simultaneously. This method uses three types of similarity matrix: (1) document-by-document, (2) phrase-by-phrase, and (3) document-by-phrase similarity matrices. Mu et al. (2016) [31] reviewed related works which are not descried in this paper.

3. Experiment of Cluster Labeling

3.1 Outline of the experiment

The basic procedure of the experiment was as follows.

(1) Clustering: A hierarchical tree of clusters was obtained by the HMDC algorithm.
(2) Labeling: Three sets of labels for each cluster (node) were obtained by using three different term weights, respectively.
(3) Evaluation: A human reviewer determined the best one among the three sets of labels for each cluster (node).

The main purpose of this experiment is to compare effectiveness between three term weighting schemes when selecting labels from index terms of documents.

The three weights were

- COR: correlation coefficient $r_{j|k}$ in Equation (10),
- TF-COR: $z_{j|k} \times r_{j|k}$ where $z_{j|k}$ is a tf factor, and
- MUHR: $z_{j|k} \times I_{GL}(j|k)$ (see Equation (11)),

where the tf factor was computed such that

$$z_{j|k} = \sum_{i:d_i \in C_k} \tilde{x}_{ij} \quad \text{and} \quad \tilde{x}_{ij} = \frac{x_{ij}}{\sqrt{\sum_{j=1}^{M} x_{ij}^2}}. \tag{14}$$

Note that $\tilde{x}_{ij}$ is a normalized element of the document vector consisting of only term frequency x_{ij}. The three weights can be computed without any external source, and therefore, the implementation is easier (the path length factor is not considered).

3.2 Identification of noun phrases

Only noun phrases and their components were chosen as index terms, which were used in the process of clustering and labeling. For example, if a phrase "electoral votes" was identified by a morphological analyzer, then three terms "electoral votes", "electoral" and "votes" were sent to the next step.

In this experiment, the Stanford Log-linear Part-Of-Speech Tagger [*9] was used for identifying noun phrases included in the text of documents according to the simple rule that a sequence consisting of nouns or adjectives was specified as a noun phrase except for those in which the component at the most right-hand side was an adjective.

Actually, when similarity between two documents was computed, Porter's stemming algorithm had to be applied to each component (i.e., single word) of noun phrases. For example, the stemming algorithm converted the noun phrase "electoral votes" to "elector vote" which was an actual index term forming document vectors. However, it is clear that "elector vote" is not appropriate as a cluster label.

Therefore, the stemmed form "elector vote" had to be restored to the original representation "electoral votes" at the stage of labeling, but the process was not easy because the other original representations (e.g., "electorate vote") were also converted to "elector vote". In this experiment, each stemmed index term was uniformly restored to the original representation appearing most frequently in the target document set, and that representation was always used as a label. In the above example, because "electoral votes" appeared more frequently than "electorate vote" in the entire document set of this experiment, "electoral votes" was adopted as a label for the index term "elector vote".

Also, upper case letters were processed similarly. For example, "Clinton" was converted to the index term "clinton", which was restored to "Clinton" in the step of labeling.

3.3 Heuristic rules for selecting labels

For selecting appropriate labels from the set of index terms in the documents, an attempt was made to use heuristic rules as follows.

- Index terms not appearing in over half of the documents within the cluster were removed from candidates because it was considered that they did not reflect the overall content of the cluster.
- Only the top 'three' index terms having the largest weights (COR, TF-COR and MUHR weights, respectively) were selected.
- If the selected term was a component of other index terms that were ranked in the top '20', the longest term was finally adopted as a label.

Suppose that a ranked list of index terms for a cluster was

1. weather condition, 2. condition, 3. storm, 4. threat, ...,

and that "emergency weather condition" was included in the top 20 index terms. If "emergency weather condition" is the longest term including "weather condition" and "condition", then "emergency weather condition", 'storm", and "threat" were finally selected as index terms from which labels of the cluster were derived unless "storm" and "threat" were included in the other index terms.

3.4 Evaluation of assigned labels

It was very difficult to measure directly the validity of labels for documents in the target cluster. Therefore, in this experiment, a human reviewer was asked to specify subjectively or intuitively the best sets of labels from the three ones obtained by using COR, TF-COR and MUHR weights, respectively. Because only index terms appearing in over the half of documents became candidates of labels and the maximum number of labels was three, clusters in the hierarchical tree could be classified into two categories: (A) clusters with 0, 1, 2 or 3 terms as labels and (B) clusters with over three terms as candidates of labels.

Obviously, it was not necessary to evaluate labels in clusters of category A because no selection by term weights was done in this experiment. The human reviewer tried to classify only clusters in category B into three classes: (1) a set of labels was better than the other two sets (e.g., labels by COR are better than those by TF-COR and MUHR), (2) two sets were better than another set (e.g., labels by TF-COR and MUHR are better than those by COR), and (3) there was no difference in quality between the three sets. This means that the experiment could show only a result of comparison between the three weighting methods under the condition that 'special' heuristic rules in Section 3.3 were applied.

An example of the comparison is as follows. If a result of labeling a cluster in which many documents describe pollution problems in city areas is such that

- Weighting Method 1: street, danger, rise
- Weighting Method 2: pollution, air quality, ozone
- Weighting Method 3: pollution, ozone, air quality

then it can be judged that Method 2 and 3 generate better labels than Method 1 because labels by Method 1 would not be able to tell us that the main topic of the cluster is pollution.

3.5 Dataset

The document set in this experiment was the same as that used in Kishida(2014) [15], which was constructed by extracting 6,374 records of English news articles from RCV-1 [21] under two conditions; (1) the news article was published during August in 1996 and (2) the news article was assigned only a single class code (the class codes were assigned by human experts for tests of supervised text categorization, which were not used for clustering in the experiment).

3.6 Results

The number of different index terms generated in the indexing process described in Section 3.2 was 320,920. After removing terms appearing in just one document, 105,865 terms remained at the step of clustering. The average length of documents was 185.96 terms.

Values of elements in document vectors were computed by Equation (3). After all document vectors were normalized (i.e., $\mathbf{d}_i/\|\mathbf{d}_i\|$), the HMDC algorithm (Kishida, 2014 [15]) was executed. The clusters with fewer than 11 documents were not divided further and were used as a final leaf node. As a result, the algorithm created a hierarchical tree consisting of 29 levels. The total number of clusters (nodes) in it was 2,252, which included 1,215 leaf nodes [*10].

[*9] http://nlp.stanford.edu/software/tagger.shtml

[*10] The nMI(max) score for the set of leaf nodes was 0.371 by using the class codes in RCV1 where the number of 'true' clusters was 68.

Table 2 Examples of labeling: a path in the hierarchy

1st level	(None)
2nd level	(None)
3rd level	peace, president
4th level	President Bill Clinton, convention, White House
5th level	President Bill Clinton, Bob Dole, convention
6th level	President Bill Clinton, Dick Morris, White House
7th level	Drug Administration recommendation, White House officials, tobacco

Table 3 Result 1: The numbers of clusters

No. of terms	No. of documents in a cluster <11	11-50	51-100	101-500	>500	Total	%
0 term	2	0	3	5	5	15	1.2
1 term	11	22	11	17	3	64	5.3
2 terms	23	36	12	10	0	81	6.7
3 terms	36	29	7	9	0	81	6.7
>3 terms	489	361	54	52	4	960	79.9
Total	561	448	87	93	12	1201	100.0
%	46.7	37.3	7.2	7.7	1.0	100.0	

An example of cluster labeling is shown as Table 2, in which labels of clusters belonging to a path in the hierarchy are displayed from the 1st to 7th levels (the leaf node is at the 7th level). In this path, whereas clusters at the 1st and 2nd levels have no labels because there was no term appearing in over half of the documents (note that the cluster at the 1st level is the root node), the two terms "peace" and "president" were included in over half of the documents at the 3rd level. From the 4th to 7th levels, because the clusters had four or more candidates, three labels were selected by MUHR weights for each level according to the heuristic rules described in Section 3.3.

In this experiment, labels assigned to only 1,201 clusters located at the 2nd to 10th levels were evaluated because they were considered to be a sample with enough size. Table 3 shows the distribution of clusters by the number of candidate terms for cluster labels. Among 1,201 clusters, 960 ones (79.9%) had four or more terms, and the selected three terms were evaluated for each cluster according to the procedure described in Section 3.4. In other words, because of the heuristic rules described in Section 3.3, three weighting methods (COR, TF-COR and MUHR) inevitably provided the same labeling results for the other set of 241 clusters (20.1 %) in this experiment.

The result of evaluating comparatively cluster labels is indicated in Table 4. For example, "C>TM" means that COR weight specified better labels than TF-COR and MUHR weights. Also, "C=T=M" indicated that there were no differences between the labels by three weighting methods, and 614 clusters (64.0 % of 960 clusters) belonged to this class. Note that 'Unknown' was the case that three sets of labels could not be evaluated due to the fact that meaningless terms were selected as labels by all three methods (just in seven clusters).

In Table 4, "TM>C" has the second largest set of clusters (128 clusters), which implies that the tf factor tends to work positively although 45 clusters were categorized as "C>TM". This tendency is more clearly indicated in Table 5 which was created by summing up the clusters for which the method selected labels that were not better than those selected by the other two methods. Whereas the number of clusters were 131 and 125 by the TF-COR and MUHR methods, the COR method did not provide better labels in 221 clusters than the other methods.

Table 4 Result of evaluation 1: The numbers of clusters

Evaluation	No. of documents in a cluster <11	11-50	51-100	101-500	>500	Total	%
C>TM	27	15	2	1	0	45	4.7
T>CM	8	9	1	1	0	19	2.0
M>CT	39	26	4	4	1	74	7.7
CT>M	29	27	4	1	0	61	6.4
CM>T	8	4	0	0	0	12	1.3
TM>C	77	45	4	2	0	128	13.3
C=T=M	295	234	39	43	3	614	64.0
Unknown	6	1	0	0	0	7	0.7
Total	489	361	54	52	4	960	100.0

Note: C - COR, T - TF-COR, and M - MUHR

Table 5 Summary of data in Table 4: The numbers of clusters

Evaluation	COR	TF-COR	MUHR
Not better than others	221	131	125
Better than or equal to others	732	822	828
Unknown	7	7	7
Total	960	960	960

4. Concluding remarks

As shown in Table 5, the experiment showed that

(1) The tf factor worked positively for selecting cluster labels, and

(2) There was no significant difference in effectiveness between the TF-COR and MUHR methods having the tf factor.

Note that the conclusions were obtained when special heuristic rules were applied to a single hierarchical tree. Therefore, the experimental result merely provides a suggestion for future research.

References

[1] Anaya-Sánchez, H., Pons-Porrata, A., and Berlanga-Llavori, R.: A document clustering algorithm for discovering and describing topics, *Pattern Recognition Letters*, Vol. 31, No. 6, pp. 502 – 510 (2010).

[2] Ayad, H. and Kamel, M.: Topic discovery from text using aggregation of different clustering methods, *Advances in Artificial Intelligence*, Vol. 2338, pp.161 – 175 (2002).

[3] Beil, F., Ester, M., and Xu, X.: Frequent term-based text clustering, *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 436 – 442 (2002).

[4] Carmel, D., Roitman, H., and Zwerdling, N.: Enhancing cluster labeling using Wikipedia, *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp.139 – 146 (2009).

[5] Chuang, J., Manning, C. D., and Heer, J.: "Without the clutter of unimportant words": Descriptive keyphrases for text visualization, *ACM Transactions on Computer-Human Interaction*, Vol. 19, No. 3, pp.1-29 (2012).

[6] Cutting, D. R., Karger, D.R., Pedersen, J.O., and Tukey, J.W.: Scatter/Gather: A cluster-based approach to browsing large document collections, *Proceedings of the 15th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 318 – 329 (1992).

[7] Cutting, D.R., Karger, D.R., and Pedersen, J.O.: Constant interaction-time Scatter/Gather browsing of very large document collections, *Proceedings of the 16th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 126–134 (1993).

[8] Dhillon, I. S.: Co-clustering documents and words using bipartite spectral graph partitioning, *Proceedings of the Seventh ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '01)*, pp. 269 – 274 (2001).

[9] dos Santos, F. F., de Carvalho, V. O., and Rezende, S. O.: Selecting candidate labels for hierarchical document clusters using association rules, *Advances in Soft Computing: 9th Mexican International Conference on Artificial Intelligence, MICAI 2010*, (Sidorov, G. et al. eds.), Springer-Verlag, pp. 163–176 (2010).

[10] Frigui, H. and Nasraoui, O.: Simultaneous clustering and dynamic keyword weighting for text documents, *Survey of Text Mining*, (Berry, M. W. ed.), Springer-Verlag, pp. 45–72 (2004).

[11] Fung, B. C. M., Wang, K., and Ester, M.: Hierarchical document clustering using frequent itemsets, *Proceedings of the 2003 SIAM International Conference on Data Mining*, pp. 59 – 70 (2003).

[12] Geraci, F., Pellegrini, M., Maggini, M., and Sebastiani, F.: Cluster generation and cluster labelling for Web snippets, *String Processing and Information Retrieval: 13th International Conference, SPIRE 2006*, (Crestani, F. et al. eds.), Springer-Verlag, pp. 25–36 (2006).

[13] Glover, E., Pennock, D. M., Lawrence, S., and Krovetz, R.: Inferring hierarchical descriptions, *Proceedings of the Eleventh International Conference on Information and Knowledge Management (CIKM '02)*, pp. 507–514 (2002).

[14] Hotho, A. and Stumme, G.: Conceptual clustering of text clusters, *Proceedings of FGML Workshop*, pp. 37–45 (2002).

[15] Kishida, K.: Algorithm for hierarchical multi-way divisive clustering of document collections, *IPSJ SIG Technical Report*, Vol.2014-IFAT-116, No.1, pp.1–8 (2014).

[16] Kummamuru, K., Dhawale, A., and Krishnapuram, R.: Fuzzy co-clustering of documents and keywords, *The 12th IEEE International Conference on Fuzzy Systems, 2003 (FUZZ '03)*, Vol.2, pp. 772 – 777 (2003).

[17] Kummamuru, K., Lotlikar, R., Roy, S., Singal, K., and Krishnapuram, R.: A hierarchical monothetic document clustering algorithm for summarization and browsing search results, *Proceedings of the 13th International Conference on World Wide Web (WWW '04)*, pp. 658–665 (2004).

[18] Lamirel, J.-C.: Combining neural clustering with intelligent labeling and unsupervised Bayesian reasoning in a multiview context for efficient diachronic analysis, *Advances in Self-Organizing Maps*, (Estévez, P. A. et al. eds.), Springer-Verlag, pp. 245 – 254 (2013).

[19] Lau, J. H., Newman, D., Karimi, S., and Baldwin, T.: Best topic word selection for topic labelling, *Proceedings of the 23rd International Conference on Computational Linguistics (COLING '10)*, pp. 605 – 613 (2010).

[20] Lau, J. H., Grieser, K., Newman, D., and Baldwin, T.: Automatic labelling of topic models, *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies (HLT'11)*, Vol. 1, pp. 1536 – 1545 (2011).

[21] Lewis, D. D., Yang, Y., Rose, T. G., and Li, F.: RCV1: A new benchmark collection for text categorization research, *Journal of Machine Learning Research*, Vol. 5, pp. 361–397 (2004).

[22] Li, X., Chen, J., and Zaiane, O.: Text document topical recursive clustering and automatic labeling of a hierarchy of document clusters, *Advances in Knowledge Discovery and Data Mining: 17th Pacific-Asia Conference, PAKDD 2013, Proceedings, Part II* (Pei, J. et al. eds.), Springer-Verlag, pp. 197 – 208 (2013).

[23] Li, Z., Li, J., Liao, Y., Wen, S., and Tang, J.: Labeling clusters from both linguistic and statistical perspectives: A hybrid approach, *Knowledge-Based Systems*, Vol. 76, pp. 219 – 227 (2015).

[24] Mao, X.-L., Ming, Z.-Y., Zha, Z.-J., Chua, T.-S., Yan, H., and Li, X.: Automatic labeling hierarchical topics, *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, pp. 2383–2386 (2012).

[25] Magatti, D., Calegari, S., Ciucci, D., and Stella, F.: Automatic labeling of topics, *Ninth International Conference on Intelligent Systems Design and Applications 2009 (ISDA '09)*, pp. 1227-1232 (2009).

[26] Maqbool, O. and Babri, H. A.: Interpreting clustering results through cluster labeling, *Proceedings of the IEEE Symposium on Emerging Technologies, 2005*, pp. 429–434 (2005).

[27] Matsumoto, T. and Hung, E.: Fuzzy clustering and relevance ranking of web search results with differentiating cluster label generation, *IEEE International Conference on Fuzzy Systems (FUZZ), 2010*, pp. 1 – 8 (2010).

[28] Mei, Q., Shen, X., and Zhai, C.-X.: Automatic labeling of multinomial topic models, *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD '07)*, pp. 490–499 (2007).

[29] Moura, M. F., Marcacini, R. M., and Rezende, S. O.: Easily labelling hierarchical document clusters, *IV Workshop em Algoritmos e Aplicações de Mineração de Dados*, pp. 37–45 (2008).

[30] Moura, M. F. and Rezende, S. O.: Choosing a hierarchical cluster labelling method for a specific domain document collection, *EPIA 2007 - 13 th Portuguese Conference in Artificial Intelligence*, pp. 812 – 823 (2007).

[31] Mu, T., Goulermas, J. Y., Korkontzelos, I., and Ananiadou, S.: Descriptive document clustering via discriminant learning in a co-embedded space of multilevel similarities, *Journal of the Association for Information Science and Technology*, Vol. 67, No. 1, pp. 106 – 133 (2016).

[32] Muhr, M., Kern, R., and Granitzer, M.: Analysis of structural relationships for hierarchical cluster labeling, *Proceedings of the 33rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 178 – 185 (2010).

[33] Müller, A., Dörre, J., Gerstl, P., and Seiffert, R.: The TaxGen framework: Automating the generation of a taxonomy for a large document collection, *Proceedings of the 32nd Hawaii International Conference on Systems Sciences (HICSS-32)*, pp. 1 – 9 (1999).

[34] Nayak, R., Mills, R., De-Vries, C., and Geva, S.: Clustering and labeling a web scale document collection using Wikipedia clusters, *Proceedings of the 5th International Workshop on Web-scale Knowledge Representation, Retrieval & Reasoning (Web-KR 2014)*, pp. 23 – 30 (2014).

[35] Osiński, S. and Weiss, D.: A concept-driven algorithm for clustering search results, *IEEE Intelligent Systems*, Vol. 20, No. 3, pp. 48–54 (2005).

[36] Popescul, A. and Ungar, L. H.: Automatic labeling of document clusters, Unpublished manuscript (2000).

[37] Roitman, H., Hummel, S., and Shmueli-Scheuer, M.: A fusion approach to cluster labeling, *Proceedings of the 37th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 883 – 886 (2014).

[38] Role, F. and Nadif, M.: Beyond cluster labeling: Semantic interpretation of clusters' contents using a graph representation, *Knowledge-Based Systems*, Vol.56, pp.141–155 (2014).

[39] Scaiella, U., Ferragina, P., Marino, A., and Ciaramita, M.: Topical clustering of search results, *Proceedings of the Fifth ACM International Conference on Web Search and Data Mining*, pp. 223 – 232 (2012).

[40] Stefanowski, J. and Weiss, D.: Extending k-means with the description comes first approach, *Control and Cybernetics*, Vol. 36, No. 4, pp. 1009 – 1035 (2007).

[41] Stefanowski, J. and Weiss, D.: Comprehensible and accurate cluster labels in text clustering, *RIAO '07 Large Scale Semantic Access to Content*, pp. 198–209 (2007).

[42] Tholpadi, G., Das, M. K., Bhattacharyya, C., and Shevade, S.: Cluster labeling for multilingual Scatter/Gather using comparable corpora, *Advances in Information Retrieval: 34th European Conference on IR Research, ECIR 2012*, (Baeza-Yates, R. et al. eds.), Springer-Verlag, pp. 388–399 (2012).

[43] Tonella, P., Ricca, F., Pianta, E., and Girardi, C.: Using keyword extraction for Web site clustering, *Proceedings of the Fifth IEEE International Workshop on Web Site Evolution (WSE '03)*, pp. 41–48 (2003).

[44] Tseng, Y.-H.: Generic title labeling for clustered documents, *Expert Systems with Applications*, Vol. 37, No. 3, pp. 2247–2254 (2010).

[45] Tseng, Y.-H., Lin, C.-J., Chen, H.-H., and Lin, Y.-I.: Toward generic title generation for clustered documents, *Information Retrieval Technology: Third Asia Information Retrieval Symposium, AIRS 2006*, (Ng, H. T. et al. eds.), Springer-Verlag, pp. 145–157 (2006).

[46] Treeratpituk, P. and Callan, J.: Automatically labeling hierarchical clusters, *Proceedings of the 7th Annual International Conference on Digital Government Research*, pp. 167 – 176 (2006).

[47] van Heerden, W. S. and Engelbrecht, A. P.: Unsupervised weight-based cluster labeling for self-organizing maps, *Advances in Self-Organizing Maps: 9th International Workshop, WSOM2012*, (Estévez, P. A. et al. eds.), Springer-Verlag, pp. 45 – 54 (2013).

[48] Xu, W., Liu, X. and Gong, Y.: Document clustering based on non-negative matrix factorization, *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '03)*, pp. 267 – 273 (2003).

[49] Zamir, O. and Etzioni, O.: Web document clustering: a feasibility demonstration, *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 46–54 (1998).

[50] Zhang, C.: Document clustering description based on combination strategy, *Fourth International Conference on Innovative Computing, Information and Control (ICICIC)*, pp. 1084–1088 (2009).

[51] Zhang, C., Wang, H., Liu, Y., and Xu, H.: Document clustering description extraction and its application, *Computer Processing of Oriental Languages. Language Technology for the Knowledge-based Economy: 22nd International Conference (ICCPOL 2009)*, (Li, W. and Mollá-Aliod, D. eds.), Springer-Verlag, pp. 370–377 (2009).

[52] Zhang, C. and Xu, H.: Clustering description extraction based on statistical machine learning, *Second International Symposium on Intelligent Information Technology Application (IITA '08)*, Vol. 2, pp. 22–26 (2008).