5-2019

Socially-enriched multimedia data co-clustering

Ah-hwee TAN
Singapore Management University, ahtan@smu.edu.sg

Published in Adaptive resonance theory in social media data clustering: Roles, methodologies, and applications. Springer. 2019. pp. 111-135. DOI: 10.1007/978-3-030-02985-2_5

Socially-Enriched Multimedia Data Co-clustering

Abstract Heterogeneous data co-clustering is a commonly used technique for tapping the rich meta-information of multimedia web documents, including category, annotation, and description, for associative discovery. However, most co-clustering methods proposed for heterogeneous data do not consider the representation problem of short and noisy text and their performance is limited by the empirical weighting of the multimodal features. This chapter explains how to use the Generalized Heterogeneous Fusion Adaptive Resonance Theory (GHF-ART) for clustering large-scale web multimedia documents. Specifically, GHF-ART is designed to handle multimedia data with an arbitrarily rich level of meta-information. For handling short and noisy text, GHF-ART employs the representation and learning methods of PF-ART as described in Sect. 3.5, which identify key tags for cluster prototype modeling by learning the probabilistic distribution of tag occurrences of clusters. More importantly, GHF-ART incorporates an adaptive method for effective fusion of the multimodal features, which weights the features of multiple data sources by incrementally measuring the importance of feature modalities through the intra-cluster scatters. Extensive experiments on two web image datasets and one text document set have shown that GHF-ART achieves significantly better clustering performance and is much faster than many existing state-of-the-art algorithms. The content of this chapter is summarized and extended from [12] (©2014 IEEE. Reprinted, with permission, from [12]), and the Python codes of GHF-ART are available at https://github.com/Lei-Meng/GHF-ART.

5.1 Introduction

The increasing popularity of social networking websites, such as Flickr and Facebook, have led to the explosive growth of sharing multimedia web documents online. In order to provide easy access for users to browse and manage large-scale repositories, effective organization of those documents with common subjects is desired. Clustering techniques, designed to identify groupings of data in multi-dimensional feature space based on measured similarity, are often applied to this task. As web multimedia resources are often attached with rich meta-information, for example,

category, annotation, description, images and surrounding text, finding a way to utilize the additional information that enhances the clustering performance poses a challenge to traditional clustering techniques.

In recent years, the heterogeneous data co-clustering approach, which is an advancement from the clustering of one data type to the co-clustering of multiple data types, has drawn much attention and been applied to the image and text domains [1, 3, 5, 11, 13]. However, the algorithms follow the similar idea of linearly combining the objective functions of each feature modality and subsequently minimizing the global cost. For the co-clustering of multimedia data, existing algorithms face three challenges elaborated as follows. First, like the short text clustering problem [7], meta-information is usually very short and therefore the extracted tags cannot be effectively weighted by traditional data mining techniques such as term frequency-inverse document frequency (tf-idf). Second, the weights of features in the objective function still rely on empirical settings, which usually leads to a sub-optimal result. Finally, this approach requires an iterative process to ensure the convergency, which leads to high computational complexity. Thus, existing methods are only applicable to small datasets consisting of up to several thousands documents, and they are very slow and even not scalable for big data.

In view of the above issues, this chapter describes using the Generalized Heterogeneous Fusion Adaptive Resonance Theory (GHF-ART) as a solution for fast and robust clustering of heterogeneous social media data. In this scenario, GHF-ART is favored for its low time complexity and the effective fusion of multimodal feature modalities. As introduced in Sect. 3.6, GHF-ART has multiple channels for the input features; each of which may receive different types of data patterns and have different similarity measures and weight learning functions. The similarities measured by different feature channel are integrated using the adaptive feature weighting algorithm, called *robustness measure*, which computes the weights of the feature modalities when computing their weighted average for the overall similarity evaluation between input data objects and cluster weights.

By analyzing the representations of social media data, GHF-ART adopts Fuzzy ART (Sect. 3.1) for the channels encoding features of text articles and image content and uses Probabilistic ART (Sect. 3.5) to process the meta-information. Additionally, the method for incorporating user preferences (Sect. 3.4) is extended to the multi-channel version, in order to take in prior knowledge by initializing the network with pre-defined clusters, indicating regions of interests to users. Contrary to traditional semi-supervised clustering techniques, such as [3], in which the user-provided knowledge is rarely reflected by the resulting clusters, GHF-ART incrementally generalizes and preserves the learned knowledge by identifying and learning from relevant input patterns, so it can present the resulting clusters, reflecting user preferences, directly to the users.

The performance of GHF-ART, in terms of clustering performance, effectiveness of *robustness measure*, incremental clustering property, noise immunity and time cost, has been evaluated on two public web image datasets, namely the NUS-WIDE [4] and Corel datasets, and a public text document set, known as the 20 Newsgroups dataset [8]. Comparing the empirical results with state-of-the-art heterogeneous data

co-clustering algorithms shows that GHF-ART consistently achieves better cluster quality and is much faster.

5.2 Problem Statement and Formulation

Considering a set of documents $\mathscr{D} = \{doc_n|_{n=1}^{N}\}$ with the associated meta-information, which may be tags, category information, or surrounding text, each document doc_n can be represented by a multi-channel input pattern $\mathbf{I}_n = \{\mathbf{x}^k|_{k=1}^{K}\}$, where $\mathbf{x}^k$ is a feature vector extracted from the document or one type of meta-information.

The goal of the heterogeneous data co-clustering task, as defined in this study, is to partition the set of N documents into a set of clusters $\mathscr{C} = \{c_j|_{j=1}^{J}\}$. It is achieved by evaluating the similarity between the input feature patterns of the documents doc_n according to their corresponding feature vectors $\mathbf{I}_n$, and grouping the documents into clusters such that the data objects in the same cluster should be more similar to each other than to the documents of the other clusters. For example, in the image domain, this task may be to identify similar images according to both the visual content and the surrounding text. In each cluster, the images therein are similar in their content and the high-level semantics reflected from the image content. Similarly, in the text domain, the task may be to consider both the features of the text document and the meta-information, such as category information and authors.

As reviewed in the previous section, the heterogeneous data co-clustering task presents several issues and challenges, especially for multimedia datasets. The key challenges in three aspects are discussed below:

1. **Representation of document content**: The representation issue of text documents has been well studied in literature. Typically, text documents are represented by the keywords appearing in the document collection, each of which is weighted based on its frequency in and across the documents, known as tf-idf. However, visual representation of images is still a challenge. Current techniques for visual feature extraction are usually based on color histograms, edge detection, texture orientation, scale-invariant points, and deep learning features, so the visual features are inadequate to represent the images at the semantic level, a problem known as semantic gap. It leads to difficulties when grouping images with very different appearances (Fig. 5.1c) or distinguishing those with similar backgrounds (Fig. 5.1a, b).
2. **Representation of meta-information**: The meta-information of documents provides additional knowledge which indicates the relations between documents from another perspective. However, in both image and text domains, the problem of noisy tags exists. Specifically, although the extracted tags from the meta-information of documents usually contain key tags that are helpful for identifying the correct groupings of documents, a large number of noisy tags exist which contribute nothing or even indicate incorrect relations between documents.

Fig. 5.1 Examples of web images sharing high-level semantics that look different in visual content, such as background scenes and the objects therein. © 2014 IEEE. Reprinted, with permission, from [12]

Distinguishing key tags from noisy text is also an open problem in tag ranking [9, 10].

3. **Integrating multiple types of features**: It is the key challenge which is related to heterogeneous data utilization for clustering. Existing works, as described in the Introduction, typically rely on some global optimization methods for the partitioning of each feature modality. However, they do not address the problem of weighting the feature modalities in their objective functions. Instead, either a uniform weighting or some empirical settings are used, which may not yield the desirable results.

5.3 GHF-ART for Multimodal Data Fusion and Analysis

As described in Sect. 3.6, GHF-ART is a multi-channel Fuzzy ART variant, which can employ different representation and learning methods for processing heterogeneous feature modalities, respectively. And the incorporated *robustness measure*

Algorithm 5.1 Clustering algorithm of GHF-ART

Input: Documents $\mathscr{D} = \{doc_n|_{n=1}^N\}$, α, β, ρ_0.

1: Generate pre-defined clusters for the initial network based on user preferences. If no prior knowledge is received, create an uncommitted cluster with all weight vectors containing 1's.
2: For each document, present its corresponding input pattern $\mathbf{I} = \{\mathbf{x}^1, \ldots, \mathbf{x}^K\}$ into the input field F_1.
3: For each cluster c_j in the category field F_2, calculate the choice function $T(c_j, \mathbf{I})$ defined in Eq. (5.2).
4: Identify the winner c_{j^*} with the largest value of the choice function such that $j^* = \arg\max_{j:c_j \in F_2} T(c_j, \mathbf{I})$.
5: Calculate the match function $M(c_{j^*}, \mathbf{x}^k)$ $(k = 1, \ldots, K)$ defined in Eq. (5.3).
6: If $\exists k$ such that $M(c_{j^*}, \mathbf{x}^k) < \rho^k$, set $T(c_{j^*}, \mathbf{I}) = 0$, update ρ^k $(k = 1, \ldots, K)$ according to Eq. (5.6), go to 4; else, go to 7.
7: If the selected c_{j^*} is uncommitted, set each cluster prototype to the corresponding feature vector of the input pattern such that $\mathbf{w}_{j^*}^k = \mathbf{x}^k$ $(k = 1, \ldots, K)$, update γ according to Eq. (5.10), and create a new uncommitted node, go to 9; else, go to 8.
8: Update $\mathbf{w}_{j^*}^k$ $(k = 1, \ldots, K)$ according to Eqs. (5.4) and (5.5) respectively, and update γ according to Eqs. (5.7)–(5.9).
9: If no input pattern exists, algorithm stops. Otherwise, go to 2.

Output: Cluster Assignment Array $\{A_n|_{n=1}^N\}$.

for weighting the importance of feature channels is a natural way for discovering the association between the feature modalities and data objects. Besides, GHF-ART can incorporate user preferences as described in Sect. 3.4. Beyond that, GHF-ART further incorporates the match tracking rule for the self-adaptation of the vigilance parameter. The above characteristics of GHF-ART make it a suitable approach for clustering the composite multimedia data objects and performing associative mining.

This section illustrates how to use the architecture of GHF-ART to represent, cluster and learn from the socially-enriched social media data. The following sub-sections offer the technical details in terms of multimedia feature construction, cross-modality similarity measure, heterogeneous cluster weights learning, adaptive parameter tuning, and algorithm time complexity. The pseudo code of GHF-ART is presented in Algorithm 5.1.

5.3.1 Feature Extraction

5.3.1.1 Feature Extraction for Document Content

In this study, a document refers to either an image or an article. For an image, the feature vector is the concatenation of multiple types of visual features. For an article, the term frequency-inverse document frequency (tf-idf) features are extracted. Since the ART-based algorithm requires the input values to be in the interval of [0,1], $min - maxnormalization$ is applied to the feature values.

5.3.1.2 Feature Extraction for Meta-Information

As the meta-information (e.g. the surrounding text for a web image or the author information for an article) is usually short and noisy, traditional text mining techniques cannot effectively weight the tags. For example, the tf-idf features usually lead to feature vectors with a flat distribution of low values [7]. Therefore, the data representation method is used in Probabilistic ART (Sect. 3.5), which models the textual features as the presence of tags, so that in the learning stage the probabilistic distribution of tag occurrences in the given clusters can be learned to be the cluster weights.

The textual feature vector for the meta-information is first constructed based on a word lexicon $\mathscr{G} = \{g_1, \ldots, g_M\}$ which consists of all the distinct tags g_m in the whole dataset. Subsequently, the textual feature vector for the nth document doc_n is denoted as $\mathbf{t}_n = [t_{n,1}, \ldots, t_{n,M}]$, where $t_{n,m}$ indicates the presence of the m-th tag $g_m \in \mathscr{G}$ in the document doc_n, defined as

$$t_{n,m} = \begin{cases} 1, & if\ g_m \in doc_n \\ 0, & otherwise \end{cases}. \tag{5.1}$$

The feature vector indicates a point in the textual feature space of M dimensions constructed by all tags. Therefore, more common tags in the two given data objects lead to a shorter distance in the feature space of GHF-ART.

5.3.2 *Similarity Measure*

GHF-ART consistently uses the choice and match functions for measuring the similarities of all feature channels. As detailed in Sect. 3.6, given a data object doc with k types of features $\mathbf{I} = \{\mathbf{x}^k|_{k=1}^K\}$ and a cluster $c_j \in \mathscr{C}$ with weight vectors $\{\mathbf{w}_j^k|_{k=1}^K\}$, the choice and match functions are defined as

$$T(c_j, \mathbf{I}) = \sum_{k=1}^{K} \gamma^k \frac{|\mathbf{x}^k \wedge \mathbf{w}_j^k|}{\alpha + |\mathbf{w}_j^k|}, \tag{5.2}$$

$$M(c_{j^*}, \mathbf{x}^k) = \frac{|\mathbf{x}^k \wedge \mathbf{w}_{j^*}^k|}{|\mathbf{x}^k|}. \tag{5.3}$$

As observed, the choice function computes a weighted sum of the similarities between $\mathbf{I}$ and c_j in terms of the K feature channels. The winning cluster $c_{j^*} = \{c_{j^*}|T(c_{j^*}, \mathbf{I}) \geq T(c_j, \mathbf{I}), c_j \in \mathscr{C}\}$ further undergoes the match function to ensure

the input data doc satisfies the intra-cluster similarity threshold in all the K channels, i.e. $M(c_{j^*}, \mathbf{x}^k) \geq \rho^k$ $(k = 1, \ldots, K)$.

This similarity measure guarantees the purity of data clusters by considering both the overall and the individual similarity between data objects and cluster weight vectors across all feature modalities. Specifically, the choice function evaluates the overall similarity with higher weights given to the robust feature channels, controlled by the contribution parameter γ, while the match function puts threshold to the similarity in terms of individual feature channels, so that a dissimilarity in even just one feature channel will incur a reset.

5.3.3 Learning Strategies for Multimodal Features

5.3.3.1 Learning Key Features of Document Content

The learning function of Fuzzy ART is used to learn the cluster prototype for the document content. Given an input document with its multi-channel input pattern $\mathbf{I} = [\mathbf{x}^1, \ldots, \mathbf{x}^K]$ and the winning cluster c_{j^*}, if $\mathbf{x}^k$ is the feature vector for document content, then the learning function for the corresponding weight vector $\mathbf{w}_{j^*}^k$ is defined by

$$\hat{\mathbf{w}}_{j^*}^k = \beta(\mathbf{x}^k \wedge \mathbf{w}_{j^*}^k) + (1 - \beta)\mathbf{w}_{j^*}^k, \tag{5.4}$$

As discussed in Sect. 3.1.2, this learning function does not increase feature values in order to depress the features having unstable values while retaining those that are stable in values.

5.3.3.2 Learning Key Features of Meta-Information

Using traditional statistical term weighting techniques, such as tf-idf, to learn the distribution of key tags of clusters is usually biased by the limited tag lexicon and the insufficient statistical information of word occurrence in the meta-information. Based on the above consideration, Probabilistic ART (Sect. 3.5) is used to model the cluster weights of this channel by the probabilistic distribution of tag occurrences. In this way, the weights of noisy tags are depressed while the key and sub-key tags are preserved.

Given the winner c_{j^*} containing L data objects, the weight vector $\mathbf{w}_{j^*}^k = [w_{j^*,1}^k, \ldots, w_{j^*,M}^k]$ of one of the meta-information channels, and the assigned input data object doc with the corresponding feature vector $\mathbf{t}_{L+1} = [t_{L+1,1}, \ldots, t_{L+1,M}]$, the learning function is defined by

$$\hat{w}^k_{j^*,m} = \begin{cases} \eta w^k_{j^*,m}, & if\ t^m_{L+1} = 0 \\ \eta(w^k_{j^*,m} + \frac{1}{L}), & otherwise \end{cases}. \tag{5.5}$$

where $\eta = \frac{L}{L+1}$.

5.3.4 Self-Adaptive Parameter Tuning

The settings of vigilance parameter ρ and contribution parameter γ affect the clustering results of GHF-ART greatly. Using some fixed values will certainly limit the robustness of GHF-ART for a diverse range of datasets. Therefore, self-adaptive tuning of the two parameters is desirable.

The following two sub-sections will describe the *match tracking rule* as a solution to relax the cases when the vigilance parameter ρ^k is set higher than the one required, and to brief the *robustness measure* (detailed in Sect. 3.6.3) for the tuning of contribution parameter γ^k.

5.3.4.1 Match Tracking Rule

The original match tracking rule was first used in ARTMAP [2] to maximize generalization with a minimum number of cluster nodes. GHF-ART utilizes a generalized form of the match tracking rule, wherein the vigilance value of each feature channel can be self-adapted.

At the beginning of each input pattern presentation, the vigilance parameters of all feature channels $\{\rho^1, \ldots, \rho^K\}$ are set to a baseline ρ_0. A change in the vigilance values is triggered when the template matching process causes a reset. The process is formalized as:

$$\hat{\rho}^k = M(c_{j^*}, \mathbf{x}^k) + \varepsilon. \quad (k = 1, \ldots, K) \tag{5.6}$$

where $\varepsilon > 0$ is a very small value and $M(c_{j^*}, \mathbf{x}^k)$ is the value of the match function, as defined in Eq. (5.3).

This extended *match tracking rule* is only triggered during the selection of the winning cluster c_{j^*}. That is, ρ^k always starts with ρ_0 at the presentation of each input data object. In this way, it does not really reshape the vigilance regions (VRs) of the clusters (See Sect. 3.2.2 for the theory of VR), but relaxes the threshold ρ^k to just above the match value of the winning cluster $M(c_{j^*}, \mathbf{x}^k)$ in order to avoid a vigilance value that is too high.

5.3.4.2 Robustness Measure of Features

Weighting the similarities of different feature modalities is an important step, since higher weights for more representative features result in better clustering performance. However, manual settings of such weights are increasingly difficult along with the increase in the number of feature modalities. As such, GHF-ART incorporates a *robustness measure* to make it possible to self-adapt the weight values, i.e. the contribution γ^k, by learning from the representativeness of different feature channels for clusters.

As illustrated in Sect. 3.6.3, the *robustness measure* updates γ^k after the assignment of each input data object. It considers two cases:

- **Resonance in existing cluster**: Given an existing cluster c_j with L data objects, when a new data object $\mathbf{I}_{L+1}$ is assigned to this cluster, the intra-cluster scatter, called *Difference*, is first updated by

$$\hat{D}_j^k = \frac{\eta}{|\hat{\mathbf{w}}_j^k|}(|\mathbf{w}_j^k| D_j^k + |\mathbf{w}_j^k - \hat{\mathbf{w}}_j^k| + \frac{1}{L}|\hat{\mathbf{w}}_j^k - \mathbf{x}_{L+1}^k|). \tag{5.7}$$

  where $\eta = \frac{L}{L+1}$.

  Subsequently, the *Robustness* of channel k is updated by

$$R^k = \exp(-\frac{1}{J}\sum_j D_j^k). \tag{5.8}$$

  where J is the number of clusters.

  Finally, the contribution value γ^k is updated by

$$\gamma^k = \frac{R^k}{\sum_{k=1}^K R^k}. \tag{5.9}$$

- **Generation of new cluster**: When generating a new cluster, the *Difference* of the other clusters remains unchanged. Therefore, the addition of a new cluster just introduces a proportion change to the *Robustness*, which is defined as

$$\hat{\gamma}^k = \frac{\hat{R}^k}{\sum_{k=1}^K \hat{R}^k} = \frac{(R^k)^{\frac{J}{J+1}}}{\sum_{k=1}^K (R^k)^{\frac{J}{J+1}}}, \tag{5.10}$$

5.3.5 *Time Complexity Comparison*

As demonstrated in Sect. 3.6.4, the time complexity of GHF-ART is $O(n_i n_c n_f)$, where n_i is the number of input patterns, n_c is the number of clusters, and n_f is

the total number of features. In comparison, the time complexity of the CIHC co-clustering algorithm is $O(QR\{n_i n_f\} + (n_i + n_f)\log(n_i + n_f))$, where QR$\{.\}$ is the time for QR matrix decomposition. The time complexity of NMF is $O(tn_c n_i n_f)$, SRC is $O(t(max(n_i^3, n_f^3) + n_c n_i n_f))$ and Comrafs is $O(t(max(n_i^3, n_f^3))$, where t is the number of iterations in the algorithm. GHF-ART has the lowest time cost and maintains a linear increase in running time with respect to the increase in the size of the dataset.

5.4 Experiments

This section presents an evaluation of the performance of GHF-ART on three real-world datasets: the NUS-WIDE dataset, the Corel5k dataset, and the 20 Newsgroups dataset, where the first two datasets are image sets associated with surrounding text and annotations and the last one is a set of text news articles. The clustering performance of GHF-ART is compared with state-of-the-art heterogeneous data co-clustering algorithms, and the properties of GHF-ART are analyzed, including the effectiveness of the *robustness measure*, the robustness to noisy data and the influence of incremental clustering manner.

5.4.1 NUS-WIDE Dataset

5.4.1.1 Data Description

The NUS-WIDE dataset [4] is the largest well-annotated web image set with filtered surrounding text, which consists of 269,648 images and their ground-truth annotations from 81 concepts. The images are downloaded from the famous photo sharing website *Flickr.com*. To effectively evaluate the clustering performance of the algorithms discussed in this chapter, a total of 23,284 images belonging to the nine biggest classes of NUS-WIDE dataset were collected, including dog, bear, cat, bird, flower, lake, sky, sunset and wedding, each of which contains nearly 3,000 images, except bear (1,271 images) and wedding (1,353 images).

The visual content and surrounding text of the images was used for clustering. For the visual features, a concatenation of Grid Color Moment (225 features), Edge Direction Histogram (73 features) and Wavelet Texture (128 features). The above three types of global features were used because they can be efficiently extracted and have been shown to be effective for image content representation [4]. Finally, each image was represented as a vector of 426 features. The textual feature vector was created by considering all distinctive and high frequency tags in the surrounding text of images. After filtering the infrequency tags, there was a total of 1,142 textual features, and each image was associated with seven tags on average.

5.4.1.2 Performance of Robustness Measure

In the experiments, the choice parameter was set to $\alpha = 0.01$, the learning parameter $\beta = 0.6$ and the baseline vigilance parameter $\rho_0 = 0.1$. A small choice parameter of $\alpha = 0.01$ is commonly used as it has been shown that the clustering performance is generally robust to this parameter [14]. $\beta = 0.6$ was empirically used to tune the cluster weight toward the geometric center of the cluster. In these experiments, the performance of GHF-ART remains roughly the same when the learning parameter changes from 0.3 to 0.8. Since the vigilance parameter has a direct effect on the number of generated clusters, $\rho_0 = 0.1$ was used which produces a small number of small clusters containing less than 1% of the data patterns. It was discovered that the performance of GHF-ART improved significantly when ρ_0 increases to 0.1. Beyond that, the performance improvement is rather small, but the number of clusters increased almost linearly. Therefore, $\rho_0 = 0.1$ was used consistently in all the experiments. Other vigilance values may still work, but a higher vigilance value may lead to a better performance in precision. However, It may create many more clusters resulting in poorer generalization.

The performance of the *robustness measure* is evaluated by comparing the clustering performance of GHF-ART using the self-adapted contribution parameter γ_{SA} with that of the fixed values γ. Since only two channels were utilized for visual and textual features respectively, the contribution parameter of the textual features γ was varied, and that of the visual features is calculated by $1 - \gamma$. The result of average precision weighted by cluster sizes is shown in Fig. 5.2a. It was observed that, without prior knowledge, the self-adaptive tuning method always has a comparable performance with the best settings and even slightly improves the results in several classes. The weighted average precision across all classes shows that the overall performance of the *robustness measure* is slightly better than the best results of the fixed settings of the contribution parameter. The time cost of GHF-ART with fixed settings is 9.610 s, and it is 9.832 s with the *robustness measure*. Therefore, this method is effective and efficient for solving the tuning problem of the contribution parameter and it is also scalable for big data.

To understand how the *robustness measure* works, the value tracking of γ_{SA} of the textual feature channel is shown in Fig. 5.2b. Despite the initial fluctuation, the value of γ_{SA} climbs from 0.5 to 0.8 and then stabilizes in the interval of [0.7, 0.8]. The initial fluctuation should be due to the order of input pattern presentation. As the *robustness measure* adjusts the contribution parameters along with learning from the input patterns, a large number of images with similar image content or tags may result in such a change in values. However, by learning from massive input patterns, the value of γ_{SA} becomes stable. It demonstrates the convergency of the *robustness measure*.

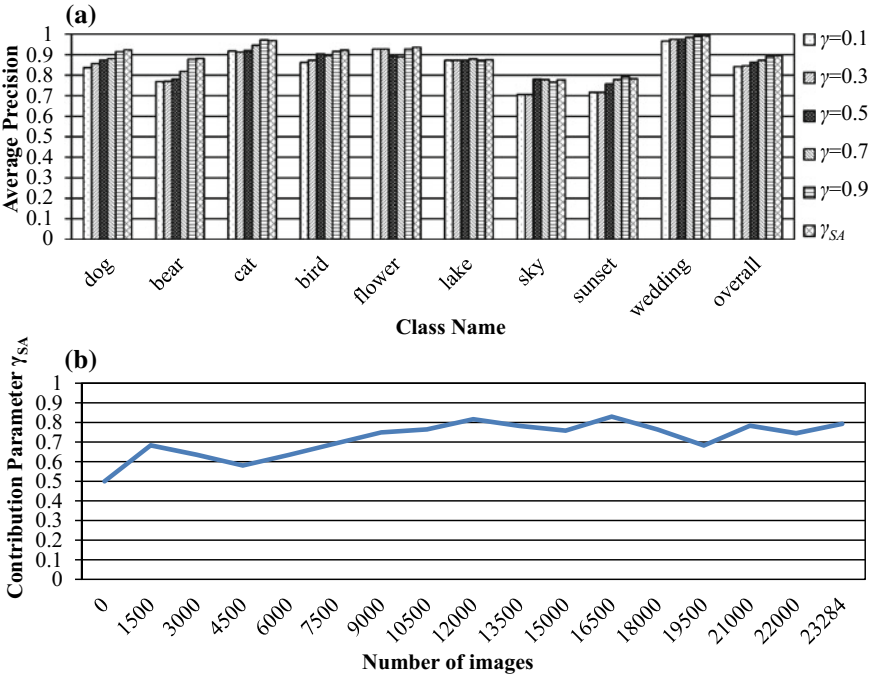

Fig. 5.2 **a** Clustering performance using fixed contribution parameters (γ) and self-adapted contribution parameter (γ_{SA}); **b** Tracking of γ_{SA} of textual feature channel on NUS-WIDE dataset. © 2014 IEEE. Reprinted, with permission, from [12]

5.4.1.3 Clustering Performance Comparison

The performance of GHF-ART is compared with Fusion ART (the base model of GHF-ART), the baseline algorithm K-means and existing heterogeneous data co-clustering algorithms, namely, CIHC, SRC, Comrafs, NMF and SS-NMF.

To make a fair comparison, since the ART-based algorithms need normalized features, experiments have been conducted to evaluate if the normalized features will benefit the other algorithms. The results show that the performance when using the normalized features is similar to those using the original features. Thus, the original features are used for the other algorithms. For K-means, the visual and textual features are concatenated and use the Euclidean distance as the distance measure. For K-means, SRC and NMF, which require a fixed number of clusters and iterations, their performance is averaged with different numbers of clusters ranging from 9 to 15, and the number of iterations is set to 50. The parameter settings of Fusion ART are the same as those of GHF-ART. A weight of 0.7, the best setting in the empirical study, is used for Fusion ART and SRC which need to set the weights for multimodal features. For the semi-supervised algorithms SS-NMF and GHF-ART(SS), three images of each class are used as user preferences. As CIHC applies

Table 5.1 Clustering performance on NUS-WIDE dataset using visual and textual features in terms of nine classes. © 2014 IEEE. Reprinted, with permission, from [12]

Average precision	Dog	Bear	Cat	Bird	Flower	Lake	Sky	Sunset	Wedding	Overall
K-means	0.8065	0.7691	0.8964	0.6956	0.7765	0.4873	0.5278	0.5836	0.9148	0.7175
CIHC	0.8524	0.8343	0.9167	0.8942	0.8756	0.6544	0.7466	0.6384	0.9127	0.8139
SRC	0.8184	0.7831	0.8193	0.8302	0.8713	0.6852	0.7132	0.5684	0.8723	0.7735
Comrafs	0.8292	0.6884	0.9236	0.8541	0.8667	0.6719	0.7240	0.6562	0.9065	0.7959
NMF	0.8677	0.8133	0.8623	0.7845	0.8259	0.7848	0.7134	0.6956	0.8648	0.8014
SS-NMF	0.8913	0.8272	0.9149	0.8366	0.8723	0.8213	0.7274	0.7346	0.9174	0.8381
Fusion ART	0.8139	0.7914	0.8500	0.9131	0.8368	0.7448	0.7039	0.6829	0.9653	0.8111
GHF-ART	0.9339	0.8814	0.9685	0.9231	0.9368	0.8755	0.7782	0.7829	0.9932	0.8971
GHF-ART(SS)	**0.9681**	**0.9023**	**0.9719**	**0.9655**	**0.9593**	**0.8864**	**0.8132**	**0.8482**	**0.9961**	**0.9234**

ratio cut, which only divides the dataset into two clusters, the precision of each class was measured by clustering with each of the other classes and calculating the average. Since two-class clustering is easier than this study's nine-class one, the effectiveness of GHF-ART can still be demonstrated if their performances are comparable.

Table 5.1 shows the clustering performance in weighted average precision for each class using the visual content of images and the corresponding surrounding text. GHF-ART outperforms the others in all cases. K-means usually achieves the worst result especially for the classes "bird", "lake" and "sky". The reason should be that the sample mean in the concatenated feature space cannot represent the common characteristics of features for some classes very well. CIHC, Comrafs and NMF usually achieve comparable performance and outperform SRC. For the semi-supervised algorithms, SS-NMF and GHF-ART(SS) achieve better performance than their unsupervised versions. Besides, GHF-ART outperforms Fusion ART in all classes, which shows the effectiveness of the proposed methods in addressing the limitations of Fusion ART.

To further evaluate the performance of GHF-ART under more complex problems, experiments were conducted with more classes and noisier data. To this end, nine new classes were chosen, including beach, boat, bridge, car, cloud, coral, fish, garden and tree, each of which contains 1500 images. Three classes "car", "cloud" and "tree" are deemed as noisy classes since all the algorithms achieve lower performance. In addition to weighted average precision, cluster and class entropies [6], purity [16] and rand index [15] are utilized as performance measures. For the algorithms that need a pre-defined number of clusters, the number was set from 18 to 30, and the average performance was calculated. For K-means, Fusion ART, GHF-ART and GHF-ART(SS), which are sensitive to initialization, the experiments were repeated for a total of ten times, and the means and standard deviations were computed. The default settings were kept for the other algorithms that are not sensitive to the initialization, and their performances were reported on a single run.

Table 5.2 Clustering results on NUS-WIDE dataset with (a) 9 and (b) 18 classes in terms of weighted average precision (AP), cluster entropy ($H_{cluster}$), class entropy (H_{class}), purity and rand index (RI). © 2014 IEEE. Reprinted, with permission, from [12]

(a)

	K-means	CIHC	SRC	Comrafs	NMF	SS-NMF	Fusion ART	GHF-ART	GHF-ART(SS)
AP	0.6582 ± 0.036	0.8139	0.7735	0.7959	0.8014	0.8381	0.8047 ± 0.031	0.8663 ± 0.022	**0.9035± 0.016**
$H_{cluster}$	0.4792 ± 0.037	0.3924	0.4169	0.4386	0.3779	0.3761	0.3744 ± 0.016	0.3583 ± 0.019	**0.3428 ± 0.013**
H_{class}	0.5317 ± 0.034	0.4105	0.4462	0.4367	0.4189	0.3922	0.4124 ± 0.024	0.3692 ± 0.018	**0.3547 ± 0.019**
Purity	0.7118 ± 0.029	0.8307	0.7891	0.8036	0.8167	0.8498	0.8352 ± 0.027	0.8863 ± 0.018	**0.9085 ± 0.021**
RI	0.6291 ± 0.031	0.7806	0.7485	0.7340	0.7615	0.7759	0.7467 ± 0.018	0.7961 ± 0.023	**0.8216 ± 0.013**

(b)

	K-means	CIHC	SRC	Comrafs	NMF	SS-NMF	Fusion ART	GHF-ART	GHF-ART(SS)
AP	0.4528 ± 0.042	0.7739	0.6812	0.6583	0.7209	0.7637	0.7379 ± 0.024	0.7933 ± 0.023	**0.8366± 0.024**
$H_{cluster}$	0.3892 ± 0.029	0.4161	0.4497	0.4667	0.4018	0.3894	0.4125 ± 0.021	0.3849 ± 0.016	**0.3624 ± 0.018**
H_{class}	0.6355 ± 0.024	0.4203	0.4726	0.4639	0.4491	0.4215	0.4378 ± 0.024	0.4109 ± 0.018	**0.3921 ± 0.019**
Purity	0.4682 ± 0.033	0.7795	0.6944	0.6727	0.7279	0.7346	0.7193 ± 0.018	0.8054 ± 0.022	**0.8433 ± 0.023**
RI	0.4677 ± 0.028	0.7049	0.6728	0.6496	0.7105	0.7488	0.7245 ± 0.022	0.7523 ± 0.012	**0.7681 ± 0.014**

Table 5.2 shows the results of the original dataset with nine classes and the new dataset with 18 classes. In Table 5.2a, it is observed that GHF-ART(SS) achieves the best results in all the evaluation measures in terms of the means. Without supervision, GHF-ART still obtains a better performance than all other algorithms. Comparing Table 5.2b with Table 5.2a show that all algorithms perform worse when the number of classes increases. This is expected as the increase in the number of classes makes it more difficult to partition the feature spaces. However, GHF-ART still obtains the best results.

To evaluate the statistical significance of performance difference, a t-test was conducted among Fusion ART, GHF-ART and GHF-ART(SS). The results show that the performance levels of Fusion ART and GHF-ART are significantly different at 0.05 level of significance in all the evaluation measures except cluster entropy, of which the difference is at 0.1 level. For GHF-ART and GHF-ART(SS), the difference between their performance in weighted average precision, purity and rand index is

Table 5.3 Clustering performance of weighted average prevision (AP) and the number of clusters generated (#clusters) on the NUS-WIDE dataset using the whole set and the subsets. © 2014 IEEE. Reprinted, with permission, from [12]

		Dog	Bear	Cat	Bird	Flower	Lake	Sky	Sunset	Wedding
Whole	AP	0.9339	0.8814	0.9685	0.9231	0.9368	0.8755	0.7782	0.7829	0.9932
	# clusters	3	2	3	4	2	3	3	1	1
Subsets	AP	0.9273	0.9036	0.9512	0.9039	0.9368	0.8622	0.7694	0.8315	0.9967
	# clusters	2	2	3	3	2	2	3	2	1

significant at 0.05 level of significance. For cluster entropy and class entropy, the performance difference is at 0.1 level.

5.4.1.4 Evaluation on Incremental Property

To evaluate the incremental property of GHF-ART, the original dataset with nine classes was divided into four smaller subsets, and GHF-ART was applied to them sequentially. Then, the clustering performance of GHF-ART was compared with that of the whole dataset. To make it a fair comparison, the sequence of the input patterns was randomized in all the subsets.

As shown in Table 5.3, all the classes, the number of clusters and the performance on weighted average precision are similar for clustering the whole dataset and the subsets. This shows that, given several sequential datasets with random pattern sequences, the cluster structures obtained by clustering the whole dataset and the subsets are similar. This demonstrates that GHF-ART is able to cluster the new patterns of the updated dataset by incrementally adapting the cluster structure learned from the original dataset.

5.4.1.5 Case Study Analysis of Performance

A case study is now presened to analyze why GHF-ART outperforms other algorithms. Since one major difference between GHF-ART and the other algorithms is the adaptive weighting method of GHF-ART, the performance was evaluated when all the algorithms employed equal weights for the visual and textual features. The results are summarized in Table 5.4. The performance of GHF-ART with adaptive weights (GHF-ART$_{aw}$) is also listed below for comparison.

Comparing GHF-ART$_{aw}$ with the performance of GHF-ART with equal weights (GHF-ART$_{ew}$) shows an obvious decrease in most classes, especially for the class

Table 5.4 Clustering performance on NUS-WIDE dataset in terms of weighted average precision (AP) using equal weights for visual and textual features in all the algorithms. GHF-ART$_{ew}$ indicates GHF-ART using equal weights and GHF-ART$_{aw}$ indicates GHF-ART using adaptive weights. © 2014 IEEE. Reprinted, with permission, from [12]

AP	Dog	Bear	Cat	Bird	Flower	Lake	Sky	Sunset	Wedding	Overall
K-means	0.8065	0.7691	0.8964	0.6956	0.7765	0.4873	0.5278	0.5836	0.9148	0.7175
CIHC	0.8524	**0.8343**	0.9167	0.8942	0.8756	0.6544	0.7466	0.6384	0.9127	0.8139
SRC	0.7629	0.7781	0.7667	0.8352	0.8274	0.6903	0.7095	0.5971	0.8566	0.7326
Comrafs	0.8292	0.6884	**0.9236**	0.8541	0.8667	0.6719	0.7240	0.6562	0.9065	0.7959
NMF	0.8677	0.8133	0.8623	0.7845	0.8259	0.7848	0.7134	0.6956	0.8648	0.8014
Fusion ART	0.7960	0.7835	0.8376	0.8891	0.8267	0.7614	0.6850	0.7035	0.9661	0.8037
GHF-ART$_{ew}$	**0.8746**	0.7812	0.9211	**0.9046**	**0.8952**	**0.8748**	**0.7814**	**0.7585**	**0.9746**	**0.8629**
GHF-ART$_{aw}$	0.9339	0.8814	0.9685	0.9231	0.9368	0.8755	0.7782	0.7829	0.9932	0.8971

"bear". Similarly, the performance of Fusion ART and SRC also have a decrease when using the equal weights. This demonstrates the importance of weighting the feature modalities in clustering. However, GHF-ART$_{ew}$ still obtains the best results in seven out of nine classes.

In addition, suppose the learning function of Fuzzy ART was used instead of the proposed learning method for the meta-information. In that case, GHF-ART degenerates to the original Fusion ART. It is observed that Fusion ART achieves a comparable performance with NMF and is a little bit lower than CIHC in the overall performance. For specific classes, Fusion ART obtains the best result in "wedding," and usually achieves a comparable performance for the other classes. However, with the proposed meta-information learning method, GHF-ART$_{ew}$ outperforms Fusion ART in most classes and shows a relatively big improvement in "lake", "sky" and "cat". This also demonstrates that the proposed learning method of meta-information enables GHF-ART to be robust in handling noisy text.

In comparison, all the other algorithms were found to achieve a low level of performance on these noisy classes. This is likely due to the differences between various methods in handling the patterns. For example, K-means generates hyperspherical clusters in the feature space which are sensitive to noise. Therefore, K-means performs poorly in the noisy classes but obtains comparable performance in classes such as "wedding". CIHC and SRC, which employ spectral clustering, derive eigenvectors from the graph affinity matrices. As such, the noisy features may lead to spurious correlations between patterns. This is why CIHC obtains reasonable performance in all the classes except the three noisy classes. Since SRC employs K-means to get the final clusters, it also suffers from the drawbacks of K-means. NMF derives the cluster indicator matrix from the relational matrices which maps the data into a non-negative latent semantic space. Like spectral clustering, noisy features should also be the main reason for the poor performance in the noisy classes. Comrafs performs clustering by finding a cluster structure of patterns that maximizes the Most Probable Explanation based on mutual information. Therefore, noisy features affect the calculation of mutual information and lead to an incorrect categorization of patterns.

Based on the above analysis, it can be concluded that GHF-ART outperforms the other algorithms when the surrounding text is noisy and when the desired weights for different feature modalities are not equal.

5.4.1.6 Time Cost Analysis

To evaluate the scalability of GHF-ART for big data, the time cost of each algorithm was studied with the increase in the number of input patterns. All the algorithms were performed on the computer with 2.66GHz Intel Core2 Duo CPUs and 3.25GB RAM. Since the user preferences for GHF-ART are given before the clustering, the time cost of GHF-ART(SS) is almost the same as that of GHF-ART. As shown in Fig. 5.3, along with the increase in the number of patterns, Comrafs has the highest time cost among all the algorithms. CIHC and NMF have a similar time cost and are slower than K-means. Fusion ART and GHF-ART incur a very small increase in

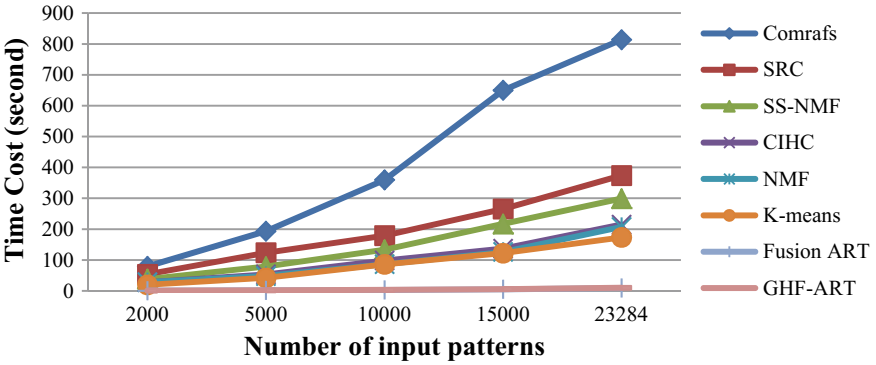

Fig. 5.3 Time cost of eight algorithms on NUS-WIDE dataset along with the increase in the number of input patterns. © 2014 IEEE. Reprinted, with permission, from [12]

time cost, while those of the other algorithms increase greatly. Although GHF-ART employs the *robustness measure*, their time costs are similar. For over 20,000 images, GHF-ART needs less than 10 s to complete the clustering process.

5.4.2 Corel Dataset

5.4.2.1 Data Description

Corel dataset is a subset of Corel CDs dataset and consists of 5,000 images from 50 Corel Stock Photo CDs, each of which contains 100 images on the same topic. Each image is annotated by an average of three to five keywords from a dictionary of 374 words. The images of six classes including "sunset", "plane", "birds", "bear", "beach" and "hills" were utilized. Similar to the NUS-WIDE dataset, 426 visual features were extracted, and the textual features were built using 374 words.

5.4.2.2 Performance of Robustness Measure

As with the NUS-WIDE dataset, the performance of GHF-ART was tested with different settings of contribution parameters on textual features in the Corel dataset. In Fig. 5.4a, it is observed that *robustness measure* achieves the best results for most classes except "sunset" and "birds," and the best overall performance is achieved by $\gamma = 0.7$. However, it still outperforms the other settings and achieves a performance that is very close to the best setting. The value tracking of γ is shown in Fig. 5.4b. In contrast to that of NUS-WIDE, this result shows a relatively smooth change in the contribution parameter value. The reason should be that the Corel dataset contains less noisy tags. As shown, the value gradually increases and stabilizes at $\gamma = 0.7$.

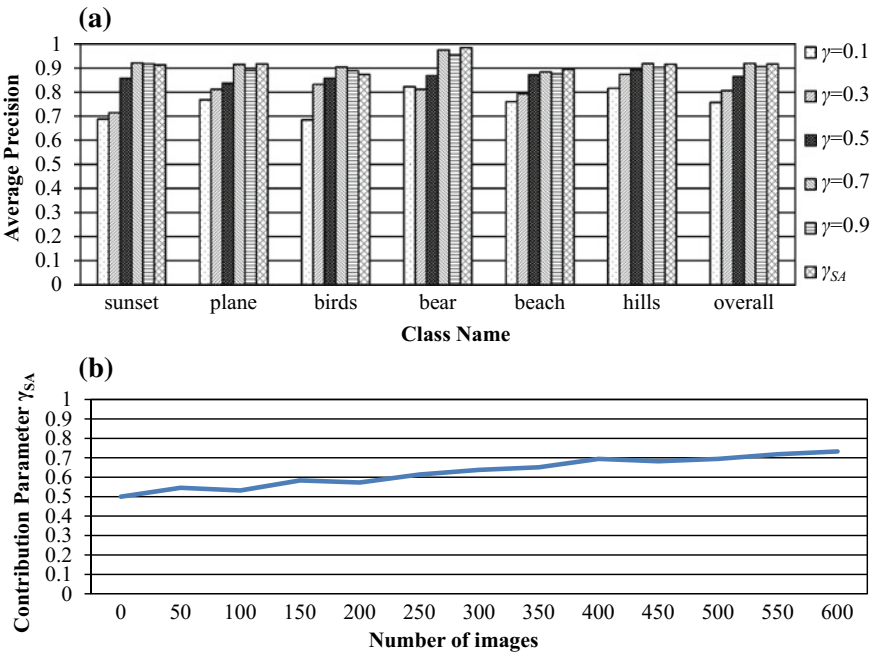

Fig. 5.4 **a** Clustering performance using fixed contribution parameters (γ) and self-adapted contribution parameter (γ_{SA}); **b** Tracking of γ_{SA} on Corel dataset. © 2014 IEEE. Reprinted, with permission, from [12]

It demonstrates that the *robustness measure* can effectively adjust the contribution parameter to the best setting.

5.4.2.3 Clustering Performance Comparison

Like the NUS-WIDE dataset, the performance of GHF-ART was evaluated in terms of weighted average precision, cluster and class entropies, purity and rand index. The number of clusters was set to range from 6 to 15 for those algorithms that need a pre-defined number of clusters.

As shown in Table 5.5, all algorithms achieve better clustering performance than they did for the NUS-WIDE dataset. One possible reason is that the visual content of the images belonging to the same category is more similar, and the tags of the Corel dataset are relatively cleaner. It can also be seen that GHF-ART and GHF-ART(SS) outperform the other algorithms in all the performance measures. Particularly, GHF-ART obtains a mean result close to CIHC and SS-NMF in weighted average precision, cluster entropy, purity and rand index but a much better performance in class entropy. With supervisory information, GHF-ART(SS) is an improvement over GHF-ART. Additionally, GHF-ART is a big improvement over Fusion ART, which demonstrates

Table 5.5 Clustering results on Corel dataset using visual content and surrounding text. © 2014 IEEE. Reprinted, with permission, from [12]

	K-means	CIHC	SRC	Comrafs	NMF	SS-NMF	Fusion ART	GHF-ART	GHF-ART(SS)
AP	0.7245 ± 0.023	0.8940	0.8697	0.8115	0.8794	0.8960	0.8525 ± 0.027	0.8944 ± 0.018	**0.9168± 0.019**
$H_{cluster}$	0.3538 ± 0.025	0.2566	0.2714	0.2972	0.2703	0.2667	0.2793 ± 0.022	0.2521 ± 0.018	**0.2366 ± 0.015**
H_{class}	0.3816 ± 0.024	0.2614	0.2803	0.3316	0.2771	0.2592	0.2409 ± 0.019	0.2184 ± 0.016	**0.1960 ± 0.014**
Purity	0.7263 ± 0.026	0.9031	0.8725	0.8304	0.8862	0.8997	0.8628 ± 0.023	0.8975 ± 0.021	**0.9176 ± 0.015**
RI	0.6635 ± 0.024	0.8347	0.8051	0.7734	0.8172	0.8416	0.8116 ± 0.015	0.8342 ± 0.018	**0.8533 ± 0.014**

the effectiveness of the proposed adaptive feature weighting and meta-information learning methods as discussed in this chapter in improving the performance and robustness of Fusion ART.

As with the NUS-WIDE dataset, a t-test was conducted between the performance of Fusion ART, GHF-ART and GHF-ART(SS). As reported in Table 5.5, the performance differences between Fusion ART, GHF-ART and GHF-ART(SS) are significant at 0.05 level of significance across all evaluation measures.

5.4.2.4 Clustering Performance Comparison with Category Information

The experiments conducted in this section consider incorporating the category information for clustering. The category information is used in the same way as surrounding text. Since the category information for each image is exactly one word, it can also be used as the noiseless tag for the image.

Generally speaking, the category information cannot be obtained for all the images under the clustering setting. It was used here as an additional tag feature to evaluate all the methods in an ideal case, and show that Fusion ART, GHF-ART and GHF-ART(SS) achieve perfect results in terms of weighted average precision, cluster entropy and purity, while the other algorithms cannot obtain such excellent results (See Table 5.6). It is because the ART-based algorithms not only evaluate the overall similarity across all the feature channels but also have constraints for each of them. Therefore, with category label, the ART-based algorithms can effectively identify the classes of images. An improvement of GHF-ART(SS) over Fusion ART and GHF-ART in class entropy and rand index, which considers how the patterns with the same label are grouped together, was also observed.

Comparing the results with those in Table 5.5, it is clear that Fusion ART, GHF-ART and GHF-ART(SS) obtain a big improvement in terms of class entropy and rand

Table 5.6 Clustering results on Corel dataset using visual content, surrounding text and category information. © 2014 IEEE. Reprinted, with permission, from [12]

	K-means	CIHC	SRC	Comrafs	NMF	SS-NMF	Fusion ART	GHF-ART	GHF-ART(SS)
AP	0.7254 ± 0.020	0.9014	0.8782	0.8279	0.8865	0.9047	**1**	**1**	**1**
$H_{cluster}$	0.3251 ± 0.026	0.2467	0.2682	0.2543	0.2489	0.2466	**0**	**0**	**0**
H_{class}	0.3688 ± 0.022	0.2544	0.2758	0.3263	0.2709	0.2537	0.1727 ± 0.023	0.1496 ± 0.016	**0.1362 ± 0.014**
Purity	0.7284 ± 0.020	0.9106	0.8721	0.8463	0.8917	0.9044	**1**	**1**	**1**
RI	0.6775 ± 0.021	0.8428	0.8147	0.8045	0.8276	0.8315	0.9061 ± 0.019	0.9297 ± 0.021	**0.9485 ± 0.016**

index, while the other algorithms have a relatively small improvement. The reason should be that the global optimization considers the overall similarity across all the feature channels, so the noisy features still contribute to incorrect categorization. It demonstrates the importance of taking in the fitness of patterns in terms of the overall similarity as well as the similarity in individual modality.

5.4.3 20 Newsgroups Dataset

5.4.3.1 Data Description

The 20 Newsgroups dataset [8] is a popular public dataset which comprises nearly 20,000 newsgroup documents across 20 different newsgroups and is widely used for experiments on text clustering techniques. Ten classes from the processed matlab version of the 20news-bydate dataset[1] were collected directly, and each of them contained nearly 1,000 documents. For the ease of discussion, the ten categories are referred to by the following abbreviations: comp.graphics (graphics), comp.windows.x (windows), rec.sport.baseball (baseball), rec.sport.hockey (hockey), sci.med (med), sci.space (space), misc.forsale (forsale), talk.politics.guns (guns), talk.politics.misc (misc) and alt.atheism (atheism). The traditional text mining algorithm tf-idf was used to extract the features of the documents, and the words in the category information were used to construct the category features.

[1] http://qwone.com/~jason/20Newsgroups/.

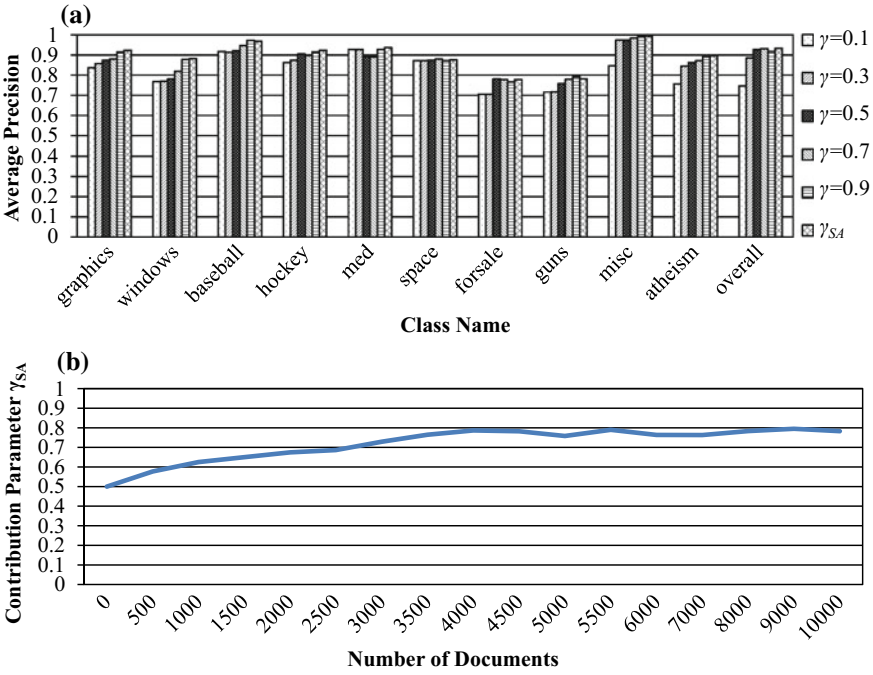

Fig. 5.5 **a** Clustering performance using fixed contribution parameters (γ) and self-adapted contribution parameter (γ_{SA}); **b** Tracking of γ_{SA} on 20 newsgroups dataset. © 2014 IEEE. Reprinted, with permission, from [12]

5.4.3.2 Performance of Robustness Measure

Figure 5.5 shows the clustering results with the contribution parameter at different settings for the category features. Figure 5.5a shows that the *robustness measure* works well for all classes and usually produces the best results. From Fig. 5.5b, it can be observed that the contribution parameter of category features gradually increases from 0.5 to over 0.6 after 1,500 input patterns. Despite the small fluctuation, the value stabilizes at around 0.8, which indicates that the category information is more robust during the clustering process.

5.4.3.3 Clustering Performance Comparison

Like the NUS-WIDE dataset, the clustering performance of GHF-ART was evaluated using weighted average precision, cluster and class entropies, purity and rand index. Since the number of classes in the 20 Newsgroups dataset is ten, the number of clusters was set to range from 10 to 15.

Table 5.7 Clustering results on 20 Newsgroups dataset using document content and category information. © 2014 IEEE. Reprinted, with permission, from [12]

	K-means	CIHC	SRC	Comrafs	NMF	SS-NMF	Fusion ART	GHF-ART	GHF-ART(SS)
AP	0.6386 ± 0.027	0.7583	0.7246	0.6547	0.7357	0.7869	0.7566 ± 0.021	0.8071 ± 0.023	**0.8452 ± 0.018**
$H_{cluster}$	0.4833 ± 0.025	0.4246	0.4432	0.4679	0.4267	0.3938	0.4016 ± 0.016	0.3822 ± 0.018	**0.3642 ± 0.018**
H_{class}	0.5284 ± 0.031	0.4573	0.4630	0.5162	0.4487	0.4296	0.4469 ± 0.015	0.4131 ± 0.017	**0.3824 ± 0.019**
Purity	0.6826 ± 0.027	0.7711	0.7348	0.6950	0.7503	0.7836	0.7538 ± 0.021	0.7994 ± 0.018	**0.8435 ± 0.021**
RI	0.6670 ± 0.025	0.7284	0.6867	0.6136	0.7019	0.7458	0.7268 ± 0.017	0.7759 ± 0.022	**0.8013 ± 0.019**

Table 5.7 shows that GHF-ART and GHF-ART(SS) outperform the other algorithms in all the performance measures. Moreover, both achieve higher than 80 percent in weighted average precision and purity, while the other algorithms typically obtain less than 75% (except CIHC and SS-NMF). Similarly, a gain of more than 3% over the best performance by the other algorithms is achieved in rand index. The t-test results further show that the performance of Fusion ART, GHF-ART and GHF-ART(SS) are significantly different at 0.05 level of significance in all evaluation measures. In fact, GHF-ART is a big improvement over Fusion ART. This demonstrates that the proposed feature weighting algorithm and meta-information learning method can help improve the performance of Fusion ART in the heterogeneous data co-clustering task.

5.5 Discussion

This chapter illustrates using Generalized Heterogeneous Fusion Adaptive Resonance Theory (GHF-ART) for clustering composite multimedia data objects, where the semantics of a single data object can be revealed by each type of data. This task is a fundamental problem to numerous real-world social media mining tasks, such as understanding the quality of online products from the behaviors and comments of users and searching for disease symptoms of images and descriptions on web forums.

Compared with the existing algorithms [1, 3, 11, 13], GHF-ART has the advantages in four aspects:

1. **Low computational complexity**: The computationally efficient clustering mechanism of GHF-ART makes it have a linear time complexity, which enables GHF-ART to be much more scalable for big data, as shown in Fig. 5.3.
2. **Adaptive channel weighting**: GHF-ART has a well-defined weighting algorithm for multi-modal feature channels. Contrary to the modality selection method in

SS-NMF [3], which only learns the weights from the prior knowledge in the distance learning step, GHF-ART evaluates the weights of feature modalities by incrementally learning from the intra-cluster scatters so that the importance of feature modalities in clustering can be incrementally evaluated. This increases the robustness of GHF-ART in fusing feature modalities for measuring pattern similarity.

3. **Strong noise immunity**: GHF-ART models the textual features of meta-information by the probability distribution of tag occurrences so that the key tags of clusters can be incrementally identified while the noisy tags are depressed. This helps maintain the robustness of GHF-ART when the quality of text is low.
4. **Incremental clustering manner**: Web multimedia data is usually big and requires frequent updates. Existing methods typically make use of a global objective function, which is then solved by an iterative optimization approach. When new data is available, these methods will have to be re-run on the entire dataset. In contrast, GHF-ART can re-cluster the new input data by adapting the original cluster structure incrementally, without referring to the old data.

To summarize, GHF-ART offers a base model for heterogeneous data co-clustering, which fits the requirement for tackling the big data challenges, i.e. volume, variety, veracity and velocity. Moving forward, GHF-ART is flexible to incorporate cutting-edge techniques for further improvement. For example, word2vector or tag ranking methods can be employed in the textual feature construction stage to filter noisy tags or give higher weights to the key tags, to further depress the effect of noisy tags. Novel mechanisms for learning cluster weights and self-adaptive tuning of parameters can also be further explored. To name a few, since the learning function for meta-information is designed to track the probabilistic distribution of the dataset in an incremental manner, there is no guarantee of convergence in response to the changing data characteristics. Since the current method for tuning vigilance parameters still cannot fully solve the problem of category proliferation, effective criteria, such VA-ARTs as described in Sect. 3.3, are required for learning the desired vigilance parameter values.

References

1. Bekkerman R, Jeon J (2007) Multi-modal clustering for multimedia collections. In: CVPR, pp 1–8
2. Carpenter GA, Grossberg S, Reynolds J (1991) ARTMAP: supervised real-time learning and classification of nonstationary data by a self-organizing neural network. Neural Netw. 4(5):565–588
3. Chen Y, Wang L, Dong M (2010) Non-negative matrix factorization for semisupervised heterogeneous data coclustering. TKDE 22(10):1459–1474
4. Chua T, Tang J, Hong R, Li H, Luo Z, Zheng Y (2009) NUS-WIDE: a real-world web image database from national university of singapore. In: CIVR, pp 1–9
5. Gao B, Liu TY, Zheng X, Cheng QS, Ma WY (2005) Consistent bipartite graph co-partitioning for star-structured high-order heterogeneous data co-clustering. In: Proceedings of international conference on knowledge discovery and data mining, pp 41–50

6. He J, Tan AH, Tan CL, Sung SY (2003) On quantitative evaluation of clustering systems. Clustering and information retrieval. Kluwer Academic Publishers, Netherland, pp 105–133
7. Hu X, Sun N, Zhang C, Chua TS (2009) Exploiting internal and external semantics for the clustering of short texts using world knowledge. In: Proceedings of ACM conference on information and knowledge management, pp 919–928
8. Lang K (2005) Newsweeder: Learning to filter netnews. In: Proceedings international conference machine learning, pp 331–339
9. Li X, Snoek CGM, Worring M (2008) Learning tag relevance by neighbor voting for social image retrieval. Proceedings of ACM multimedia, pp 180–187
10. Liu D, Hua X, Yang L, Wang M, Zhang, H (2009) Tag ranking. In: Proceedings of international conference on World Wide Web, pp 351–360
11. Long B, Wu X, Zhang Z, Yu PS (2006) Spectral clustering for multi-type relational data. In: ICML, pp 585–592
12. Meng L, Tan AH, Xu D (2014) Semi-supervised heterogeneous fusion for multimedia data co-clustering. IEEE Trans Knowl Data Eng 26(9):2293–2306
13. Rege M, Dong M, Hua J (2008) Graph theoretical framework for simultaneously integrating visual and textual features for efficient web image clustering. In: Proceedings of international conference on World Wide Web, pp 317–326
14. Tan AH (1995) Adaptive resonance associative map. Neural Netw. 8(3):437–446
15. Xu R, II DCW (2011) BARTMAP: A viable structure for biclustering. Neural Netw. 709–716
16. Zhao Y, Karypis G (2001) Criterion functions for document clustering: experiments and analysis. Technical report, Department of computer science, University of Minnesota