Chapter 8
An Overview of Multi-View Methods for Text Clustering

Maha Fraj, Mohamed Aymen Ben HajKacem, and Nadia Essoussi

8.1 Introduction

The rapid development of information technology and the abundant amount of available data have considerably contributed to the growth of studies on multi-view clustering [8, 32] . Multi-view data is observed from varying points resulting in different representations (views) with distinct statistical properties. In text clustering, these views can be obtained through word frequencies, topic and context based representations, and many other embedding models capable of capturing either syntactic or semantic information or both [14]. The main task of multi-view text clustering is to achieve better clustering by combining information held by each view, such information is disregarded when only a single view is used. However, efficiently integrating different views while preserving their characteristics remains a challenge. A naive solution for multi-view clustering consists in concatenating features from all views then apply a single-view clustering algorithm. Nevertheless, such combination fails to exploit the specificity of each view. Hence, multiple approaches have been proposed to optimize multi-view clustering [16, 19, 35].

This chapter reviews multi-view methods for text clustering. In fact, textual data was examined early on in the context of multi-view, particularly in cross-lingual text categorization where the data is labeled in one view and not in another, the aim is to use the information in both views to label all data [1, 28, 30]. With the abundance of unlabeled data, this process was extent to multi-view text clustering [13].

M. Fraj (✉) · M. A. Ben HajKacem · N. Essoussi
Institut Supérieur de Gestion de Tunis, LARODEC, Université de Tunis, Le Bardo, Tunisia
e-mail: maha.fraj.m@gmail.com; nadia.essoussi@isg.rnu.tn

B. Alyoubi et al. (eds.), *Machine Learning and Data Analytics for Solving Business Problems*, Unsupervised and Semi-Supervised Learning,
https://doi.org/10.1007/978-3-031-18483-3_8

The reminder of this chapter is organized as follows: Sect. 8.2 presents an overview of exiting multi-view clustering methos, specifically for text clustering. Section 8.3 evaluates the performance on real-world textual data. Finally, Sect. 8.4 presents the conclusion and current challenges.

8.2 Overview of Multi-View Textual Data Clustering

The main challenge of multi-view clustering consists in integrating the different views while taking advantage of the characteristics of each view to improve the clustering results. An intuitive solution consists of concatenating all features from all views and apply a clustering algorithm afterwards, this, however, ignores the statistical properties of each view and can conceal valuable information [3]. To this end, according to the integration scheme, existing multi-view clustering methods can be presented under three main categories [22]. The first category called late integration derives clustering results from each view individually, then a fusion step is applied to reach a consensus clustering [7, 29]. The second category is based on co-training, which incorporate multi-view integration into the clustering process directly through jointly optimizing the objective function [2, 17]. The third category is based on space learning, such that views are mapped into a new space to reveal the latent data structure. We present in the following the characteristics of each category and detail a number of existing methods.

8.2.1 Late Integration Based Methods

The late integration approach, also known as late fusion, consists of applying a clustering algorithm on each view individually and subsequently combines the results into a consensus clustering. The idea examines the relations between the clusters derived from each view rather than the relations between data points. The combination of clustering results can be obtained using different methods, such as latent probabilistic models [7] or more recently ensemble methods [9, 13, 26]. Figure 8.1 presents the overall process of late integration based methods.

8.2.1.1 Ensemble Methods for Multi-View Text Clustering

Xie et al. [31] proposed a multi-view clustering ensembles, an combination of multi-view clustering algorithms and ensemble clustering. The method extends both multi-view kernel K-means [27] and multi- view spectral clustering [16] to ensemble clustering and compares the two methods. Given a data set **X**, different clustering results $\{\pi^1, \pi^2, \ldots, \pi^H\}$ are obtained through different runs of the clustering algorithm. These clustering are then combined based on plurality voting,

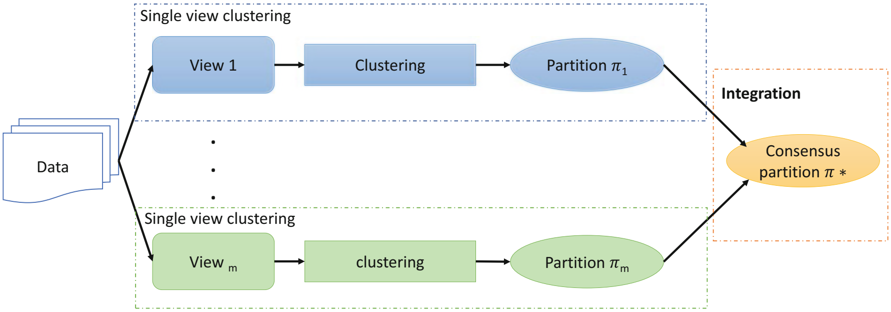

Fig. 8.1 General process of late integration based methods

Algorithm 1: Multi-view clustering ensembles

```
Input: data set X, number of clusters $k$, number of clustering ensembles $L$
Output: clustering ensembles
for each$\sigma_l \in [\sigma_{min}, \sigma_{max}]$ do
    for $v = 1$ to $m$ do
        Compute RBF $K^v = \exp \frac{-\|x_i^v - x_j^v\|^2}{2\sigma_l^2}$
    end for
    $\tilde{K}_l = [K_l^1, K_l^2, \ldots, K_l^m]$
    Run multi-view kernel K-means or multi-view spectral clustering with $\tilde{K}$ and $k$
end for
Combine the clusterings using selective voting
```

i.e., considering the majority cluster label for each data point to give the final clustering $\pi*$.

Hussain et al. [13] presented a late integration framework for multi-view document clustering based on ensemble method. The proposed method first converts views into term weighted matrices using two weighting schemes: TF-IDF and TF-ICF [24]. Hierarchical clustering is then applied on each view individually to obtain different partitions. In order to aggregate the clustering results, different ensemble techniques are adopted: the Cluster Based Similarity Partitioning (CBSP) [25], the Pairwise Dissimilarity (PD) [36], and the Affinity matrix based technique. Each ensemble technique provides a similarity matrix, which are then aggregated into an overall similarity matrix used for the final clustering. Similarly, Fraj et al. [9] proposed a multi-view ensemble methods for text clustering (MEMTC) based on multiple representations. The main idea consists of integrating different text representation models: TF-IDF, LDA, and skip-gram to generate, respectively, syntactic, topic, and semantic views. Lastly, ensemble techniques CBSP and Pairwise Dissimilarity are used to aggregate the different partitions yielded by each view. The main steps of MEMTC are presented in Algorithm 2

Algorithm 2: Ensemble methods for multi-view clustering

```
   Input: a collection of text documents X
   Output: final consensus clustering
1  $\mathbf{X}_v \leftarrow \mathrm{R}(\mathbf{X})$    $v \in$ {TF-IDF, LDA, Skip-gram }
                                              // R: document representation
2  Apply hierarchical clustering to obtain per-view partitions. Calculate the cluster based
   similarity partitioning matrix $\mathrm{S_H}$
3  Calculate the pairwise similarity matrix $\mathrm{S_{PDM}}$
4  Aggregate the similarity matrices into one matrix S
5  Apply the hierarchical clustering on S
```

8.2.1.2 Multi-View Clustering Based on Latent Models

Bruno et al. [7] proposed an integration framework based on latent models for document clustering. In this work, documents from each view are clustered into k^v clusters. The set of clusterings $\{c_1^v, \ldots, c_k^v\}$ are then concatenated into $K \times M$ matrix $\mathbf{C}$, such that $K = \sum_v k^v$ is the total number of clusters over all views. Based on $\mathbf{C}$, a joint probability $P(c_k, c_{k'})$ is derived to deduce the pairwise cluster agreement, which represents the number of documents belonging simultaneously to clusters c_k and $c_{k'}$. The joint cluster-cluster probability is defined as follows:

$$\begin{aligned} P(c_k, c_{k'}) &= \sum_n P(c_k|x_i)P(c_{k'}|x_i)P(x_i) \\ &= \sum_n \frac{P(c_k, x_i)P(c_{k'}, x_i)}{P(x_i)} \end{aligned} \tag{8.1}$$

where the joint-cluster document probability is obtained by:

$$P(c_k, x_i) = \frac{\mathbf{C}_{k,i}}{MN}, \quad \forall k \in [1, K], \forall i \in [1, N] \tag{8.2}$$

To obtain the final clustering for each document, the Probabilistic Latent Semantic Analysis (**PLSA**) [12] is adopted to derive latent variable z_j such that

$$P(c_k, c_{k'}) = P(c_{k'}) \sum_j^L P(c_k|z_j)P(z_j|c_{k'}) \quad , j = 1, \ldots, L \tag{8.3}$$

PLSA seeks to find the relationship between the clusters observations across different views and the latent variables z. The overall clustering is established by assigning to document x_i the discrete variable z_j that maximizes the following posterior probability:

Algorithm 3: A late fusion approach using latent models

```
Input: multi-view documents $\mathbf{X}^v$
Output: Final clustering assignment $z$
Run a clustering algorithm on each view individually
Concatenate clusterings $\{c_1^v, \ldots, c_k^v\}$ into matrix $\mathbf{C}$
Apply PLSA using Eqs. 8.1 and 8.2
for $i = 1$ to $N$ do
    Assign $z_j$ to $x_i$ by maximizing Eq. 8.4
end for
```

$$P(z_j|x_i) = \frac{\sum_k P(z_j|c_k)P(c_k, x_i)}{P(x_i)} \tag{8.4}$$

To estimate the latent variables, experiments were carried using the Expectation-Maximization (**EM**) algorithm and the Nonnegative Matrix Factorization (**NMF**) [11] where both methods have performed similarly. Algorithm 3 summarizes the main step of the approach.

8.2.2 Co-training Based Methods

Co-training based methods seek to find a consensus by maximizing the mutual agreement across all views. Co-training was originated by Blum et al. [4] in order to tackle semi-supervised problem. Given the abundance of unlabeled data, such data can be used to enrich the training set of the labeled data, such that given two views the learning algorithm is trained on the labeled data of both views in a bootstrapping manner. Finally, based on the consensus principle, the views should agree on all labeled data. Eventually, co-training was adopted in unsupervised learning [3] and has shown good performance despite the absence of labeled data. In general, co-training based methods are based on three main assumptions: *Sufficiency*: each view is sufficient to perform the clustering task, *Compatibility*: each pair of views predicts with high probability the same label for data points with co-occurring features, and *Conditional independence*: the views are conditionally independent given the class label [16]. Figure 8.2 presents the general process of co-training based methods.

8.2.2.1 Multi-View K-Means Based Methods

Multi-view clustering was advanced by Bickel et al. [3], where the empirical results show that the proposed multi-view spherical k-means improves the quality of document clustering in comparison to the single-view version of the algorithm. The presented co-training algorithm is based on the following assumptions: given two views v^1 and v^2, each view is sufficient to output clustering results by itself, and

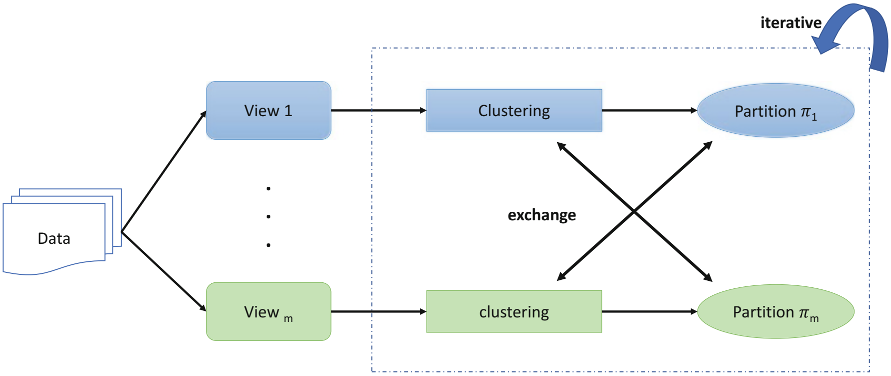

Fig. 8.2 General process of co-training

views are conditionally independent given the class label. The clustering process starts by randomly initializing the set of parameters Θ^v including the centers c_j^v, $j = 1, \ldots, k$, where k is the desired number clusters and $v = 1$ or $v = 2$. Documents are then assigned to clusters given the smallest computed distance to c_j^v. A two-step iterative process is applied afterwards taking turns between views. The first step consists of updating the clusters centers such that:

$$c_j^v = \frac{\sum\limits_{x^v \in \pi_j^v} x^v}{\| \sum\limits_{x^v \in \pi_j^v} x^v \|} \tag{8.5}$$

where π_j^v is the jth partition given the vth view. The assignment step consists of calculating the distance between documents and centers, and finding the new partitions. After each iteration, partitions are exchanged for an updating and assignment steps for the other view. For the final clustering, consensus centers are calculated by considering the documents that both views agree on such that:

$$cons\_c_j^v = \frac{\sum\limits_{x_i^1 \in \pi_j^1 \wedge x_i^2 \in \pi_j^2} x_i^v}{\| \sum\limits_{x_i^1 \in \pi_j^1 \wedge x_i^2 \in \pi_j^2} x_i^v \|} \tag{8.6}$$

The final partitioning is obtained by assigning documents to the closest consensus vector. Given that the algorithm is based on alternating partitions between views, convergence is not guaranteed. The main steps of multi-view spherical k-means are presented in Algorithm 4.

Bettoumi et al. [2] proposed a collaborative multi-view K-means CO-K-means that introduces an interconnection term to overcome the inter-view disagreement.

Algorithm 4: Multi-view spherical k-means

```
Input: multi-view documents $\mathbf{X} = \{\mathbf{X}^1, \mathbf{X}^2, \dots, \mathbf{X}^m\}$, number of clusters $k$
Output: Final clustering assignment $\pi$
Initialize randomly $\Theta^2$ and $c_j^2$,    $j = 1, \dots, k$
Assign each document to the partition corresponding to the closest center $c_j^2$
t=0
while $t < tmax$ do
    for $v = 1 : 2$ do
        $t = t + 1$
        Calculate the new centers using Eq. 8.5
        Compute the cosine distance between documents and centers
        Assign each document its closest center
    end for
    Compute Objective function by $J(\Theta_t) = \sum_{j=1}^{k} \sum_{x^v \in \pi_j^v} \langle x^v, c_j^v \rangle$
    if $(J(\Theta_t) < J(\Theta_{min}))$ then
        $t = 0$
    end
end
Calculate the consensus centers using Eq. 8.6
Find the final clustering assignment
```

Views are encouraged to reach an agreement by minimizing the contradiction across partitions. To solve this problem, the K-means objective function is altered such that:

$$\Omega = \sum_{v} \sum_{i} \sum_{k} \|\mathbf{x}_i^v - \mathbf{c}_k^v\|_2^2 + \mu\varphi \tag{8.7}$$

where μ is a modulation parameter and φ is the interconnection term denoted by:

$$\varphi = \frac{1}{|V| - 1} \sum_{v > v'} \sum_{i}^{n} \sum_{k} (\|\mathbf{x}_i^v - \mathbf{c}_k^v\|_2^2 - \|\mathbf{x}_i^{v'} - \mathbf{c}_k^{v'}\|_2^2) \tag{8.8}$$

Similarly to the classic K-means, the proposed algorithm starts by randomly initializing the clusters centers for each view, followed by an assignment step. Then, for each view, new centers are computed. The interconnection term φ aims to reduce the distance between the partitions yielded from each view. The main steps of Co-K-means are given in Algorithm 5.

8.2.2.2 Self-Organizing Map Multi-View Clustering

Fraj et al. [10] proposed a multi-view clustering method based on the Self-Organizing Map neural network [15]. Similarly to [9], each view corresponds to

Algorithm 5: Collaborative multi-view K-means

```
Input: multi-view data $\mathbf{X} = \{\mathbf{X}^1, \mathbf{X}^2, \ldots, \mathbf{X}^m\}$, number of clusters $k$
Output: Final clustering assignment $\pi$
For all views, initialize the clusters centers.
repeat
    Assign data points to clusters with the smallest distance
    $\pi = \text{argmin}(\sum_v \|\mathbf{x}_i^v - \mathbf{c}^v\|_2^2 + \frac{1}{|V|-1}(\sum_{v>v'} \|\mathbf{x}_i^v - \mathbf{c}_k^v\|_2^2 - \|\mathbf{x}_i^{v'} - \mathbf{c}_k^{v'}\|_2^2))$
    for $v = 1 : m$ do
        Update centers $\mathbf{c}_k^v$ by $\mathbf{c}_k^v = \underset{c^v}{\text{argmin}} \sum \|\mathbf{x}_i^v - \mathbf{c}^v\|_2^2$
    end for
until convergence of 8.7;
```

a text representation model, i.e., TF-IDf, LDA, and skip-gram. The views are presented as input layers, such that each document has three vector representations $\mathbf{x} = \{x^1, x^2, x^3\}$. Documents are then mapped onto the output layer, such that each document is assigned to a node on the map. Consequently, each node (neuron) of the output layer is defined by v prototypes $\mathbf{w}$ each of which is associated with a view. First, the learning process consists in generating random SOM prototypes, $\mathbf{W}^v$. Secondly, an overall distance is calculated for each document $\mathbf{x}_i^v$ in the view v and the node $\mathbf{w}$ such that

$$D = \sum_i D_v(x^v, w), \;\; v \in 1, 2, 3 \tag{8.9}$$

The node with the smallest distance is considered the Best Matching Unit BMU to which the document $\mathbf{x}_i$ is assigned. The number of nodes on the output map is set empirically to boost the performance of the SOM learning, the number, however, may not coincide with the desired number of clusters which is usually less important. Therefore, the nodes on the map are clustered using agglomerative hierarchical clustering and each document is assigned to the same cluster as its corresponding SOM node. The main steps of MVSOM are presented in Algorithm 6.

8.2.2.3 Multi-View Spectral Clustering

Kumar et al. [16] have presented a co-training based spectral clustering, where two views exchange the eigenvectors resulting from the graph Laplacian of each view. The algorithm ensures consistency across views such that if two points are assigned in same cluster in one view, it should be so in all the views. On the other hand, if two points belong to different clusters in one view, they should be clustered separately across all views. The proposed algorithm first builds an adjacency matrix $\mathbf{A^v}$ for each view, from which the graph Laplacian matrix $\mathbf{L}^v$ is obtained such that:

Algorithm 6: Self-organizing map for multi-view text clustering

```
Input: multi-view documents $\mathbf{X}^v$, number of SOM neurons $l$, learning rate $\alpha_0$, radius $\sigma_0$
Output: SOM prototypes of each view $\mathbf{W}^v$
$t \leftarrow 1$
repeat
    for $v = 1$ to $m$ do
        Initialize random SOM prototypes $\mathbf{W}^v$
        for $i = 1$ to $n$ do
            determine Best Matching Unit $BMU$ for document $\mathbf{x}_i$
            // Update SOM prototypes
            for $j = 1$ to $l$ do
                $\mathbf{w}_j^v \leftarrow \mathbf{w}_j^v + h \times \alpha \times \left(\mathbf{x}_i^v - \mathbf{w}_j^v\right)$
            end for
        end for
    end for
    // Update radius of the neighborhood
    $\sigma \leftarrow \sigma_0 \exp\left(\frac{t}{tmax}\right)$
    // Update the learning rate
    $\alpha \leftarrow \alpha_0 \exp\left(\frac{t}{tmax}\right)$
    $t \leftarrow t + 1$
until $t > tmax$
```

$$\mathbf{L}^v = \mathbf{D}^{v-1/2}\mathbf{A}^v\mathbf{D}^{v-1/2} \tag{8.10}$$

where $\mathbf{D}^v$ is the diagonal matrix such that $\mathbf{D}_{ii}^v = \sum_j \mathbf{A}_{ij}^v$. The k largest eigenvectors of $\mathbf{L}$ hold the discrimination information for clustering. Thus, the eigenvectors are exchanged across views to propagate the per-view clustering information, such that the largest k eigenvectors form the matrix $\mathbf{U}^v$. Precisely, the co-trained spectral clustering uses the eigenvectors of one view to modify the adjacency matrix of the other view and consequently the graph structure, such that each column $\mathbf{a}_i$ of $\mathbf{A}$ represents the similarity of the data point i with all point in the graph. The algorithm projects the column vectors of one view in the direction of the k eigenvectors of the other view, then back projects them to the original space to obtain the modified graph. To obtain the update adjacency matrix $\mathbf{S}^v$, a symmetrization step is performed such that:

$$\mathbf{S}^v = sym(\mathbf{U}^{\bar{v}}\mathbf{U}^{\bar{v}^T} A^v) \tag{8.11}$$

where $sym(\mathbf{S}) = (\mathbf{S} + \mathbf{S}^T)/2$. The new graph Laplacian $\mathbf{L}^v$ are obtained from $\mathbf{S}^v$, from which the k eigenvectors and $\mathbf{U}v$ are deduced. The algorithm performs these steps for a defined number of iteration. The final clustering is given by the k-means algorithm performed on matrix $\mathbf{V}$, the column-wise concatenation of $\mathbf{U}^v$. The main steps of co-training multi-view spectral clustering are given in Algorithm 7.

Algorithm 7: Co-training based multi-view spectral clustering

```
Input: multi-view documents $\mathbf{X} = \{\mathbf{X}^1, \mathbf{X}^2, \ldots, \mathbf{X}^m\}$, number of clusters $k$
Output: Final clustering assignment $\pi$
// Initialization
for $v = 1:2$ do
    Compute adjacency matrix $\mathbf{A^v}$
    Compute normalized Laplacian matrix using Eq. 8.10
    Initialize $\mathbf{U}^{v0}$ by $\mathbf{U}^{v0} = \text{argmax}\, tr(\mathbf{U}^{v^T}\mathbf{A}^v\mathbf{U}^v) \quad \text{s.t}\ \mathbf{U}^{v^T}\mathbf{U}^v = \mathbf{I}$
end for
for $t = 1$ to $tmax$ do
    Compute $\mathbf{S^1}$ and $\mathbf{S^2}$ using 8.11
    Compute the Laplacian matrices $\mathbf{L}^v$ from $\mathbf{S}^v$
    Build $\mathbf{U}^v$ from the $k$ largest eigenvectors of $\mathbf{L}^v$
end for
Normalize the rows of $\mathbf{U}^1$ and $\mathbf{U}^2$
Build $\mathbf{V}$ as the column-wise concatenation of $\mathbf{U}^1$ and $\mathbf{U}^2$
Run $k$-means on $\mathbf{V}$ to obtain the clustering assignments
```

Lin et al. [20] proposed Multi-view Proximity Learning for Clustering (MVPL), a method that learns the proximity matrix based on data representative and spectral clustering. Given a set of multi-view data $\mathbf{X} = \{\mathbf{X}^1, \mathbf{X}^2, \ldots, \mathbf{X}^m\} \in \mathbb{R}^{d^v \times n}$, a data representative matrix $\mathbf{U}^v \in \mathbb{R}^{d^v \times n}$ is associated with each view to exploit the relations between objects within the same view. The new data representative considers the proximity between each pair of data points. Therefore the learned similarity matrix is affected by these representatives, and inversely. On the other hand, MVPL considers the spectral embedding of data to integrate the different views and thus consider the inter-view relations into the similarity matrix. The goal of MVPL is to minimize the following objective function:

$$\begin{aligned}
&\min_{\{\mathbf{U}^v\},\{\mathbf{S}^v\},\mathbf{F}} \frac{1}{n}\sum_{v=1}^{m}\left(\sum_{i=1}^{n}\|\mathbf{x}_i^v-\mathbf{u}_i^v\|_2^2+\frac{\alpha}{n^2}\left(\sum_{i,j=1}^{n}\|\mathbf{u}_i^v-\mathbf{u}_j^v\|_2^2 s_{ij}+\beta\|\mathbf{S}\|_F^2\right)\right)\\
&+\gamma\frac{1}{2n^2}\sum_{v=1}^{m}\sum_{i,j=1}^{n}s_{ij}^v\|\mathbf{f}_i-\mathbf{f}_j\|_2^2 \qquad (8.12)\\
&s.t \sum_{j=1}^{n}s_{ij}=1,\ s_{ij}^v\geq 0, \forall i, j, \mathbf{F}\mathbf{F}^T=\mathbf{I}
\end{aligned}$$

where the first term considers the impact of the data representatives, while the second term models the relation between the spectral embedding matrix $\mathbf{F}$ and the similarity matrix $\mathbf{S}^v$, γ is a trade-off parameter that balances the two terms, α controls the distance between the original data features and their representatives, and β controls the sparsity of $\mathbf{S}$. Algorithm 8 describes the main steps of MVPL.

Algorithm 8: Multi-view proximity learning

```
Input: multi-view documents $\mathbf{X} = \{\mathbf{X}^1, \mathbf{X}^2, \ldots, \mathbf{X}^m\}$, number of clusters $k$, parameters $\alpha, \gamma$
Output: Proximity matrices $\{\mathbf{S}^1, \mathbf{S}^2, \ldots, \mathbf{S}^m\}$
Initialize representative matrices $\mathbf{U}^v$ as $\mathbf{X}^v$
Initialize proximity matrices $\mathbf{S}^v$ by
Determine sparsity parameter $\beta$
Initialize F by solving $\min_{\mathbf{F}\mathbf{F}^T=\mathbf{I}} Tr(\mathbf{F}\mathbf{L}_{\mathbf{SF}}^T)$
repeat
    for $v = 1$ to $m$ do
        Update $\mathbf{U}^v$ by solving $\mathbf{U}^v(\mathbf{I} + \frac{2\alpha}{n}\mathbf{L}_{\mathbf{S}}) = \mathbf{X}^v$
        Update $\mathbf{S}^v$ by solving $\min_{\mathbf{s}_i^v} \|\mathbf{s}_i^v + \frac{\mathbf{d}_i^v}{2\beta^v}\|_2^2$
    end for
    Update F
until converged or max iteration is reached;
```

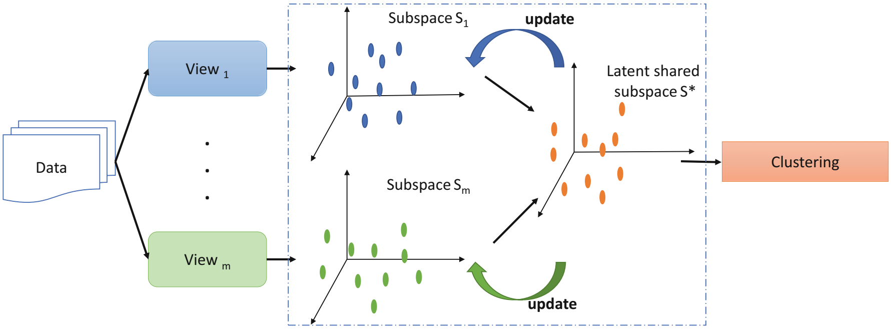

Fig. 8.3 General process of multi-view subspace clustering

8.2.3 Subspace Clustering Based Methods

The third category of multi-view clustering is based on subspace learning. Recently, more and more studies have exploited subspace clustering to extract distinct clustering features. Multi-view subspace clustering assumes that the data samples from different views share the same subspace [33]. Figure 8.3 illustrates the process of learning a shared subspace from multi-view data. The performance of subspace clustering relies on the latent representation matrix obtained from the different multi-view subspaces. Several methods have been proposed in order to identify the common subspace, we distinguish two main subcategories: NMF based methods and latent representation based methods.

8.2.3.1 Muti-View Subspace Clustering Based on Nonnegative Matrix Factorization

Liu et al. [22] proposed MultiNMF, a multi-view clustering via joint nonnegative matrix factorization. The algorithm enforces each view's indicator matrix towards a common consensus. Given multi-view data $\mathbf{X}^v \in \mathbb{R}_+^{d \times n}$, its matrix factorization is:

$$\mathbf{X}^v \approx \mathbf{U}^v \mathbf{V}^{vT} \tag{8.13}$$

where $\mathbf{V}^v \in \mathbb{R}_+^{n \times k}$ and $\mathbf{U}^v \in \mathbb{R}_+^{d \times k}$ represent the indicator matrix and the basis matrices of view v, respectively. MultiNMF adopts a normalization constraint so that all indicator matrices are comparable and significant for clustering. The problem can be defined as a joint minimization of the following objective function:

$$\begin{gathered}\sum_{v}^{m} \|\mathbf{X}^v - \mathbf{U}^v \mathbf{V}^{vT}\|_F^2 + \sum_{v}^{m} \lambda_v \|\mathbf{V}^v \mathbf{Q}^v - \mathbf{V}^*\|_F^2 \\ s.t\ v \in \{1, \ldots, m\}, \mathbf{U}^v \geqslant 0, \mathbf{V}^v \geqslant 0, \mathbf{V}^* \geqslant 0\end{gathered} \tag{8.14}$$

where $\mathbf{V}^*$ is the consensus matrix, and $\mathbf{Q}^v$ is a diagonal matrix such that:

$$\mathbf{Q}^v = Diag\left(\sum_{j=1}^{d} \mathbf{U}_{j1}^v, \sum_{j=1}^{d} \mathbf{U}_{j2}^v, \ldots, \sum_{j=1}^{d} \mathbf{U}_{jk}^v\right) \tag{8.15}$$

Finally, the clustering assignment of data point i is computed as $\operatorname{argmax}_k \mathbf{V}_{ik}^*$. The main steps of MultiNMF are given in Algorithm 9.

Zhang et al. [34] proposed a constrained NMF based clustering (CMVNMF) that uses an inter-view must-link (*ML*) and cannot-link (*CL*) constraints in order to minimize the disagreement between each pair of views. To accomplish the clustering task, the following objective function is minimized:

$$\|\mathbf{X}^v - \mathbf{U}^v \mathbf{V}^{vT}\| + \beta \sum_{v, v' \in [1, m]} \Delta_{v, v'} \quad s.t\ \mathbf{U}^v \geq 0, \mathbf{V}^v \geq 0 \tag{8.16}$$

where β is a regularization parameter, and Δ measures the disagreement between v and v' such that:

$$\Delta_{v, v'} = \sum_{(\mathbf{x}_i^v, \mathbf{x}_j^{v'}) \in ML^{v, v'}} (\mathbf{v}_i - \mathbf{v}_j') + 2 \sum_{(\mathbf{x}_i^v, \mathbf{x}_j^{v'}) \in CL^{v, v'}} \mathbf{v}_i \mathbf{v}_j' \tag{8.17}$$

The must-link and cannot-link constraints are defined by matrices $\mathbf{M}^{vv'}$ and $\mathbf{C}^{vv'}$, respectively, such that :

Algorithm 9: Multi-view NMF

```
Input: multi-view documents $\mathbf{X} = \{\mathbf{X}^1, \mathbf{X}^2, \ldots, \mathbf{X}^m\}$, number of clusters $k$, parameters
    $\{\lambda_1, \lambda_2, \ldots, \lambda_m\}$
Output: Basis matrices $\{\mathbf{U}^1, \mathbf{U}^2, \ldots, \mathbf{U}^m\}$, Consensus Matrix $\mathbf{V}^*$
Normalize each view $\mathbf{X}^v$ such that $\|\mathbf{X}^v\|_1 = 1$
Initialize $\mathbf{U}^v$ and $\mathbf{V}^*$
repeat
    for $v = 1$ to $m$ do
        repeat
            Fixing $\mathbf{V}^*$ and $\mathbf{V}^v$, update $\mathbf{U}^v$ by
            $\mathbf{U}^v = \mathbf{U}^v \odot \frac{(\mathbf{X}^v\mathbf{V}^v) + \lambda_v \sum^n \mathbf{V}^v\mathbf{V}^*}{(\mathbf{U}^v\mathbf{V}^{v^T}\mathbf{V}^v) + \lambda_v \sum^d \mathbf{U}^v \sum^n \mathbf{V}^{v2}}$
            // $\odot$ is the element-wise multiplication
            Normalize $\mathbf{U}^v$ by $\mathbf{U}^v = \mathbf{U}^v\mathbf{Q}^v{-1}$
            Normalize $\mathbf{V}^v$ by $\mathbf{V}^v = \mathbf{V}^v\mathbf{Q}^v$
            Fixing $\mathbf{V}^*$ and $\mathbf{U}^v$, update $\mathbf{V}^v$ by $\mathbf{V}^v = \mathbf{V}^v \odot \frac{(\mathbf{X}^{v^T}\mathbf{U}^v) + \lambda_v\mathbf{V}^*}{(\mathbf{V}^v\mathbf{U}^{v^T}\mathbf{U}^v) + \lambda_v\mathbf{V}^v}$
        until convergence of $\|\mathbf{X}^v - \mathbf{U}^v\mathbf{V}^{v^T}\|_F^2 + \lambda_v\|\mathbf{V}^v\mathbf{Q}^v - \mathbf{V}^*\|_F^2$;
    end for
    Fixing $\mathbf{U}^v$ and $\mathbf{V}^v$, update $\mathbf{V}^*$ by $\mathbf{V}^* = \frac{\sum_v^m \lambda_v\mathbf{V}^v\mathbf{Q}^v}{\sum_v^m \lambda_v}$
until convergence of 8.14;
```

$$\mathbf{M}_{ij}^{vv'} \begin{cases} 1, & (x_i^v, x_j^{v'}) \in ML^{v,v'} \\ 0, & \text{otherwise} \end{cases}$$

$$\mathbf{C}_{ij}^{vv'} \begin{cases} 1, & (x_i^v, x_j^{v'}) \in CL^{v,v'} \\ 0, & \text{otherwise} \end{cases}$$

The distance between a pair of data points in the same cluster from different views is minimized through the must-link constraints, while the cannot-link constraints aim to maximize the distance of data points belonging to different views and different clusters. The main steps of CMVNMF are given in Algorithm 10.

8.2.3.2 Multi-View Subspace Clustering Based on Shared Latent Representation

Zhang et al. [33] proposed Latent Multi-view Subspace Clustering (LMSC), which is based on the assumption that multi-view data share a latent subspace representation. LMSC learns a common representation from the different views based on subspace clustering. First, the original multi-view data $\mathbf{X}^v$ is reconstructed based on projection models $\mathbf{P}^v$ and achieve a common latent representation $\mathbf{H}$ such that:

```
Algorithm 10: Constrained multi-view NMF
Input: multi-view documents $\mathbf{X} = \{\mathbf{X}^1, \mathbf{X}^2, \ldots, \mathbf{X}^m\}$, the number of cluster $k$, the must-link
       constraints matrix $\mathbf{M}^{vv'}$, the cannot-link constraints matrix $\mathbf{C}^{vv'}$
Output: the clustering assignment $\pi$
Normalize $\mathbf{X}^v$
For each pair of views $(v, v')$, compute diagonal matrices $\mathbf{D}^{vv'}$ by $\mathbf{D}_{ii}^{vv'} = \sum_j^n \mathbf{M}_{ij}^{vv'}$ with
$i = 1, \ldots, n$
Initialize $\mathbf{U}^v$ and $\mathbf{V}^v$
repeat
    for $v = 1$ to $m$ do
        Fix $\mathbf{V}^v$, and update $\mathbf{U}^v$ by $\mathbf{U}^v = \mathbf{U}^v \odot \frac{(\mathbf{X}^v\mathbf{V}^v)}{(\mathbf{U}^v{\mathbf{V}^v}^T\mathbf{V}^v)}$
        Fix $\mathbf{U}^v$, and update $\mathbf{V}^v$ by
        $\mathbf{V}^v = \mathbf{V}^v \odot \frac{({\mathbf{X}^v}^T\mathbf{U}^v) + \beta\sum_{v=1,v\neq v'}^{m}(\mathbf{M}^{v,v'}\mathbf{V}^{v'}}{(\mathbf{V}^v{\mathbf{U}^v}^T\mathbf{U}^v) + \beta\sum_{v=1,v\neq v'}^{m}(\mathbf{D}^{vv'}\mathbf{V}^v + \mathbf{C}^{vv'}\mathbf{V}^{v'})}$
    end for
until convergence of 8.16;
```

$$\mathbf{x}_i^v = \mathbf{P}^v\mathbf{h}_i + \mathbf{e}_i^v \tag{8.18}$$

where $\mathbf{e}_i^v$ denotes the reconstruction error. Then, the latent representation is integrated into subspace clustering, such that the clustering problem is defined as:

$$\min_{\mathbf{Z}} L_r(\mathbf{H}, \mathbf{HZ}) + \alpha\Omega(\mathbf{Z}) \tag{8.19}$$

where $\mathbf{Z}$ is the subspace representation matrix, $L_r()$ is the loss function of the subspace reconstruction, $\Omega()$ corresponds to the regularization term, α balances the regularization. By introducing the parameters λ_1 and λ_2, the overall objective function of LMSC becomes as follows:

$$\begin{aligned} &\min_{\mathbf{P},\mathbf{H},\mathbf{Z},\mathbf{E}_h,\mathbf{E}_r} \|\mathbf{E}_h\|_{2,1} + \lambda_1\|\mathbf{E}_r\|_{2,1} + \lambda_2\|\mathbf{Z}\|_* \\ &s.t \quad \mathbf{X} = \mathbf{PH} + \mathbf{E}_h, \mathbf{H} = \mathbf{HZ} + \mathbf{E}_r, \text{and } \mathbf{PP}^T = 1 \end{aligned} \tag{8.20}$$

The $\ell_{2,1}$ norm ensures robustness in the presence of noise, while the nuclear norm ℓ_* captures the underlying clustering structure. To solve Eq. 8.20, the error matrices $\mathbf{E}_h$ and $\mathbf{E}_r$ are vertically concatenated, and the Augmented Lagrangian Multiplier with Alternating Direction Minimization (ALM-ADM) strategy proposed in [21] is adopted. The main steps of LMSC are given in Algorithm 11.

Brbic et al. [6] proposed a multi-view low-rank and sparse subspace clustering (MLRSSC), with two regularization scheme: pairwise and centroid based. The first establishes a pairwise agreement across views, whereas the second coerces the representations towards a common centroid, as first introduced by Kumar et al. [17]. Both methods are based on constructing a low-rank and sparse affinity matrix from

Algorithm 11: Latent multi-view subspace clustering

```
Input: multi-view documents $\mathbf{X} = \{\mathbf{X}^1, \mathbf{X}^2, \ldots, \mathbf{X}^m\}$, number of clusters $k$, parameter $\lambda$
Output: $\mathbf{Z}, \mathbf{H}, \mathbf{P}, \mathbf{E}$
Initialize $\mathbf{P} = 0, \mathbf{E} = 0, \mathbf{Z} = 0, \mathbf{Y_1} = 0, \mathbf{Y_2} = 0, \mathbf{Y_3} = 0, \mu = 10^{-6}, \rho = 1.1, \epsilon = 10^{-4}$
Initialize randomly $\mathbf{H}$
while not converged do
    Update $\mathbf{P}$ by $\mathbf{P} = \text{argmin}\ \frac{\mu}{2}\|(\mathbf{X} + \frac{1}{\mu}\mathbf{Y}_1 - \mathbf{E}_h) - \mathbf{H}^T\mathbf{P}^T\|$
    Update $\mathbf{H}$ by
        $$\mathbf{AH} + \mathbf{HB} = \mathbf{C}$$
        $$\text{with } \mathbf{A} = \mu\mathbf{P}^T\mathbf{P}\,, \mathbf{B} = \mu(\mathbf{ZZ^T} - \mathbf{Z} - \mathbf{Z^T} + \mathbf{I})$$
        $$\mathbf{C} = \mathbf{P}^T\mathbf{Y}_1 + \mathbf{Y}_2(\mathbf{Z}^T - \mathbf{I})$$
        $$+\mu(\mathbf{P}^T\mathbf{X} + \mathbf{E}_r^T - \mathbf{P}^T\mathbf{E}_h - \mathbf{E}_r\mathbf{Z}^T)$$
    Update $\mathbf{Z}$ by $\mathbf{Z} = (\mathbf{H}^T\mathbf{H} + \mathbf{I})^{-1}[(\mathbf{J} + \mathbf{H}^T\mathbf{H} - \mathbf{H}^T\mathbf{E}_r) + (\mathbf{Y}_3 + \mathbf{H}^T\mathbf{Y}_2)/\mu]$
    Update $\mathbf{E}$ by $\mathbf{E} = \text{argmin}_E\ \frac{1}{\mu}\|\mathbf{E}\|_{2,1} + \frac{1}{2}\|\mathbf{E} - \mathbf{G}\|_F^2$
    Update $\mathbf{J}$ by $\mathbf{J} = \frac{\lambda}{\mu}\|J\|_* + \frac{1}{2}\|\mathbf{J} - (\mathbf{Z} - \mathbf{Y}_3/\mu)\|_F^2$
    Update $\mathbf{Y}_1, \mathbf{Y}_2, \mathbf{Y}_3$ by $\begin{cases} \mathbf{Y}_1 = \mathbf{Y}_1 + \mu(\mathbf{X} - \mathbf{PH} - \mathbf{E}_h \\ \mathbf{Y}_2 = \mathbf{Y}_2 + \mu(\mathbf{H} - \mathbf{HZ} - \mathbf{E}_r \\ \mathbf{Y}_3 = \mathbf{Y}_3 + \mu(\mathbf{J} - \mathbf{Z}) \end{cases}$
    Update $\mu$ by $\mu = \min(\rho\mu; \max_\mu)$
    Check convergence criteria $\|\mathbf{X} - \mathbf{PH} - \mathbf{E}_h\|_\infty < \epsilon$, $\|\mathbf{H} - \mathbf{HZ} - \mathbf{E}_r\|_\infty < \epsilon$ and
    $\|\mathbf{J} - \mathbf{Z}\|_\infty < \epsilon$
end
```

multi-view data. Given a set of multi-view data $\mathbf{X}^v$, MLRSSC aims to find a joint representation matrix $\mathbf{C}$ that presents an agreement across views by minimizing the following objective function:

$$\begin{aligned} \min_{\mathbf{C}^{(1)},\mathbf{C}^{(2)},\ldots,\mathbf{C}^{(m)}} & \sum_{v=1}^{m}(\beta_1 \left\|\mathbf{C}^v\right\|_* + \beta_2 \left\|\mathbf{C}^v\right\|_1) + \sum_{1\leq v,w\leq m, v\neq w} \lambda^v \left\|\mathbf{C}^v - \mathbf{C}^w\right\|_F^2 \\ \text{s.t.} \quad & \mathbf{X}^v = \mathbf{X}^v\mathbf{C}^v, \quad diag(\mathbf{C}^v) = 0, \quad v = 1, \ldots, m, \end{aligned} \tag{8.21}$$

where $\mathbf{Z}^v$ is the representation matrix of view v, β_1 and β_2 are the balancing parameters of low-rank and sparsity constraint, λ^v is the consensus parameter. In case where all views are considered equally important, the same λ^v is used. The last term maximizes the pairwise similarity across views. To solve the problem in Eq. 8.21, the Alternating Direction Method of Multipliers (ADMM) strategy is used [5]. Algorithms 12 and 13 summarize the steps of pairwise MLRSSC and centroid-based MLRSSC, respectively.

Algorithm 12: Pairwise MLRSSC

```
Input: Multi-view documents $\mathbf{X} = \{\mathbf{X}^1, \mathbf{X}^2, \ldots, \mathbf{X}^m\}, k, \beta_1, \beta_2, \{\lambda^v\}_{v=1}^m, \{\mu_i\}_{v=1}^4, \mu^{max}, \rho$
Output: $k$ clusters assignments
Initialize: $\{\mathbf{C}_i^v = 0\}_{i=1}^3, \mathbf{A}^v = 0, \{\mathbf{\Lambda}_i^v = 0\}_{i=1}^4, i = 1, \ldots, m$
repeat
    for $v = 1$ to $m$ do
        Update $\mathbf{A}^v$ by solving
            $\mathbf{A}^v = [\mu_1 \mathbf{X}^{v^T}\mathbf{X}^v + (\mu_2 + \mu_3 + \mu_4)\mathbf{I}]^{-1} \times (\mu_1 \mathbf{X}^{v^T}\mathbf{X}^v + \mu_2 \mathbf{C}_2^v$
            $+ \mu_3 \mathbf{C}_1^v + \mu_4 \mathbf{C}_3^v + \mathbf{W}^{v^T}\mathbf{\Lambda}_1^v + \mathbf{\Lambda}_2^v + \mathbf{\Lambda}_3^v + \mathbf{\Lambda}_4^v)$
        Update $\mathbf{C}_1^v$ by solving $\min_{\mathbf{C}_1^v} \beta_1 \left\|\mathbf{C}_1^v\right\|_* + \frac{\mu_3}{2}\left\|\mathbf{A}^v - \mathbf{C}_1^v + \frac{\mathbf{\Lambda}_3^v}{\mu_3}\right\|_F^2$
        Update $\mathbf{C}_2^v$ by solving $\min_{\mathbf{C}_2^v} \beta_2 \left\|\mathbf{C}_2^v\right\|_1 + \frac{\mu_2}{2}\left\|\mathbf{A}^v - \mathbf{C}_2^v + \frac{\mathbf{\Lambda}_2^v}{\mu_2}\right\|_F^2$
        Update $\mathbf{C}_3^v$ by solving
        $\min_{\mathbf{C}_3^v} \lambda^v \sum_{1 \leq w \leq m, v \neq w} \left\|\mathbf{C}_3^v - \mathbf{C}^w\right\|_F^2 + \frac{\mu_4}{2}\left\|\mathbf{A}^v - \mathbf{C}_3^v\right\|_F^2 + tr\left[\mathbf{\Lambda}_4^{v^T}\left(\mathbf{A}^v - \mathbf{C}_3^v\right)\right]$
        Update $\mathbf{\Lambda}_1^v, \mathbf{\Lambda}_2^v, \mathbf{\Lambda}_3^v, \mathbf{\Lambda}_4^v$
    end for
    Update $\mu_i = \min(\rho\mu_i, \mu^{max}), i = 1, \ldots, 4$
    Check convergence conditions: $\|\mathbf{A}^v - \mathbf{C}_1^v\|_\infty \leq \epsilon$, $\|\mathbf{A}^v - \mathbf{C}_2^v\|_\infty \leq \epsilon$,
    $\|\mathbf{A}^v - \mathbf{C}_3^v\|_\infty \leq \epsilon$, and $\|\mathbf{A}_t^v - \mathbf{A}_{t-1}^v\| \leq \epsilon$
until Convergence or reaching the maximum number of iterations
Combine $\mathbf{C}_1^v, \mathbf{C}_2^v, \mathbf{C}_3^v$ by considering the element-wise average
Perform spectral clustering on the affinity matrix $\mathbf{S} = |\mathbf{C}_{avg}| + |\mathbf{C}_{avg}|^T$
```

Algorithm 13: Centroid-based MLRSSC

```
Input: Multi-view documents $\mathbf{X}^v, k, \beta_1, \beta_2, \{\lambda^v\}_{v=1}^m, \{\mu_i\}_{v=1}^4, \mu^{max}, \rho$
Output: k clusters assignments
Initialize: $\{\mathbf{C}_i^v = 0\}_{i=1}^3, \mathbf{C}^* = 0, \mathbf{A}^v = 0, \{\mathbf{\Lambda}_i^v = 0\}_{i=1}^4, i = 1, \ldots, m$
repeat
    for $v = 1$ to $m$ do
        Update $\mathbf{A}^v, \mathbf{C}_1^v, \mathbf{C}_2^v, \mathbf{C}_3^v$ as in Algorithm 12
        Update $\mathbf{\Lambda}_1^v, \mathbf{\Lambda}_2^v, \mathbf{\Lambda}_3^v, \mathbf{\Lambda}_4^v$
    end for
    Update $\mu_i = \min(\rho\mu_i, \mu^{max}), i = 1, \ldots, 4$
    Update $\mathbf{C}^* = \frac{\sum_v \lambda^v \mathbf{C^v}}{\sum_v \lambda^v}$
    Check convergence conditions: $\|\mathbf{A}^v - \mathbf{C}_1^v\|_\infty \leq \epsilon$, $\|\mathbf{A}^v - \mathbf{C}_2^v\|_\infty \leq \epsilon$,
    $\|\mathbf{A}^v - \mathbf{C}_3^v\|_\infty \leq \epsilon$, and $\|\mathbf{A}_t^v - \mathbf{A}_{t-1}^v\| \leq \epsilon$
until Convergence or reaching the maximum number of iterations
Perform spectral clustering on the affinity matrix $\mathbf{S} = |\mathbf{C}^*| + |\mathbf{C}^*|^T$
```

8.2.4 Summary of Multi-View Methods for Text Clustering

Compared to single-view data, multi-view data presents multiple advantages given its ability to describe objects from different aspects and thus give a more comprehensive representation of data. However, the manipulation and exploitation of multi-view data require further advanced algorithms in order to mine the complementarity between views and discover knowledge that is otherwise hidden in a single-view framework. Multi-view data is furthermore challenging in the case of unlabeled data given that no prior knowledge is available. The existing multi-view clustering algorithms, as the ones presented in this chapter, have shown good performance in dealing with different points of multi-view data such as finding a consensus across views, integrating the information provided by each view, discovering hidden patterns, etc.

Multiple methods for multi-view text clustering rely on a single representation model, usually the TF-IDF [16]. Although this model is capable of capture the syntactic properties of text, it is, however, unable to give an insight on semantic concepts or topically related features of text data. To this end, other methods exploited different representation models such as TF-ICF in [13] or topic models and word embeddings [9, 10]. Table 8.1 summarizes the characteristics of multi-view clustering methods.

8.3 Experiments

We evaluate in this section the performance of multi-view clustering methods on text data. We select methods from each category: MEMTC [9], MVEM [13], MVKM [3], MVSOM [10], LMSC [33], pairwise MLRSSC and centroid-based MLRSSC [6]. We also compare these methods to other baseline such as PCA and basic spectral clustering applied to concatenated views.

8.3.1 Data Sets Description

The experiments are carried on four commonly used data sets for multi-view text clustering. The *Reuters* data set is a collection of 2189 documents belonging to 8 classes. The *20 Newsgroups* consists of 2828 news articles distributed on 20 classes. The *WebKB* data set is a collection of 4168 web pages collected from computer science departments, belonging to 4 classes (student, faculty, project, course). The *BBC Sport* consists of 737 documents from the BBC Sport website corresponding to sports news articles belonging to 5 areas: football, rugby, tennis, athletics, and cricket. Before applying the clustering algorithms, a preprocessing step is performed on the data sets including stop words removal. Stop words removal consists in

Table 8.1 Summarization of multi-view clustering methods

Approach	Method	# views	Text representation	Pros	Cons
Late integration	MVEM [13]	>2	TF-IDF, TF-ICF	– Aims towards a consensus – Good empirical performance – Complementarity of views based on their clustering results	– Quality of clustering depends on the consensus technique – Does not explore the inter-relation across views – Computational cost
	MEMTC [9]		TF-IDF, LDA, Skip-gram		
	MVEC [26]		TF		
	LFALM [7]		–		
	MVCE [31]		–		
Co-training	MVKM [3]	= 2	TF-IDF	– Maximizes mutual agreement across views – Exchanges information, i.e., clustering assignment	– Sensitive to noise – Becomes challenging when number of views increases
	Co-trained spectral [16]	=2	TF-IDF		
	MVPL [20]	>2	TF		
	Co-Kmeans [2]		>2		
	MVSOM [10]		TF-IDF, LDA, Skip-gram		
Subspace based	MultiNMF [22]	>2	TF	– Explores the specificities of each view – Suitable for high-dimensional data	– Depends on the optimization of the latent subspace – Parameter tuning
	CMVNMF [34]				
	LMSC [33]		TF-IDF		
	MLRSSC [6]				

Table 8.2 Data sets description

Data set	Documents	Features	k
Reuters	2189	2577	8
BBC Sports	737	3853	5
20 newsgroup	2263	6943	20
webKB	2084	3857	4

eliminating common words that appear frequently and offer no additional semantic value. Table 8.2 summarizes the properties of all data sets.

8.3.2 Evaluation Measures

To measure the quality of the clustering and compare it with existing methods, three evaluation measures are utilized: the F-measure [18], the Normalized Mutual Information (NMI) [37], and Purity [23]. Given a set of clusters $C = \{c_1, c_2, \ldots, c_k\}$ and the gold standard classes $G = \{g_1, g_2, \ldots, g_j\}$:
F-measure is a trade-off between $Precision$ and $Recall$ such that:

$$F - measure(c_k, g_j) = 2 * \frac{Precision(c_k, g_j) \times Recall(c_k, g_j)}{Precision(c_k, g_j) + Recall(c_k, g_j)} \tag{8.22}$$

$$Precision(c_k, g_j) = \frac{|c_k \cap g_j|}{|c_k|} \tag{8.23}$$

$$Recall(c_k, g_j) = \frac{|c_k \cap g_j|}{|g_j|} \tag{8.24}$$

Normalized Mutual Information (NMI) measures the quality of clustering with regards to the number of clusters and their sizes. NMI is defined as:

$$NMI(C, G) = \frac{I(C, G)}{[E(C) + E(G]]/2} \tag{8.25}$$

where I is the mutual information and $E(C)$ is entropy.

$$I(C, G) = \sum_k \sum_j \frac{|c_k \cap g_j|}{N} \log \frac{N|c_k \cap g_j|}{|c_k||g_j|} \tag{8.26}$$

$$E(C) = -\sum_k \frac{|s_k|}{N} \log \frac{|s_k|}{N} \tag{8.27}$$

Purity: measures the number of correctly assigned documents, where each cluster is assigned to the dominant class in that cluster. The larger the number of clusters is, the higher the Purity is. Unlike NMI, Purity cannot trade-off the quality of the clustering against the number of clusters

$$Purity(C, G) = \frac{1}{N} \sum_{k} \max_{j} |c_k \cap g_j| \tag{8.28}$$

For all measures, the values range from 0 to 1, such that values closer to 0 represent poor quality

8.3.3 Experimental Results

Table 8.3 reports the performance of the different methods. Given the results, we can observe that most multi-view methods provided better clustering in comparison to concatenated views. This shows that concatenating views can result in losing the individual properties of views and affect the overall clustering. Another noticeable observation is that all methods have given their best results on the smallest data set, the BBC Sport, while the overall performance is affected on the largest data set, 20 newsgroup. We can conclude that the size and the dimension of the data set can jeopardize the performance; this may be due to noise and redundant information. Although all methods have yielded close results, we can notice that multi-view subspace clustering methods achieve relatively better results on almost all data sets, which can indicate that these methods are capable of learning a common latent representation from all views. On the other hand, both ensemble methods have performed similarly, however, MEMTC had better performance, which indicates that including other representation scheme can improve the final clustering.

Overall, late integration based method has shown good empirical performance given that the individual clustering provided by each view can compensate the clustering inaccuracy of another view. However, such methods can be computationally expensive since the clustering is performed of the number of views and the integration phase is independent from the clustering phase and can add on to the computational cost.

Co-training based on a simultaneous optimization of one unified objective function to achieve one clustering result from different views [2, 3]. However, having a unified objective function does not allow to learn from each view independently, which can result in losing the knowledge held in different views and can later be integrated to improve the overall clustering. Furthermore, co-training based method becomes intractable when the number of views is over three.

Another issue consists of integrating multiple views while maintaining their diversity. Precisely, in the clustering process reaching a consensus, or co-training based clustering can result in losing the specificity of each view. To this end subspace clustering based algorithm can present a solution [6]. However, the challenge

Table 8.3 Comparison of clustering results with multi-view methods

Data set	Method	F-score	NMI	Purity
Reuters	PCA	0.442	0.335	0.422
	Concat SC	0.476	0.227	0.436
	MVKM	0.648	0.428	0.743
	MEMTC	0.814	0.604	0.458
	MVEM	0.490	0.337	0.493
	LMSC	0.705	0.508	0.593
	Centroid MLRSSC	0.629	0.430	0.534
	Pairwise MLRSSC	0.539	0.339	0.443
	MVSOM	0.709	0.464	0.606
BBC Sport	PCA	0.613	0.388	0.606
	Concat SC	0.500	0.206	0.405
	MVKM	0.693	0.564	0.633
	MEMTC	0.797	0.730	0.771
	MVEM	0.819	0.717	0.753
	LMSC	0.804	0.711	0.767
	Centroid MLRSSC	0.838	0.708	0.833
	Pairwise MLRSSC	0.873	0.716	0.871
	MVSOM	0.821	0.728	0.744
20 newsgroup	PCA	0.356	0.302	0.290
	Concat SC	0.440	0.439	0.392
	MVKM	0.432	0.380	0.373
	MEMTC	0.511	0.534	0.458
	MVEM	0.380	0.305	0.300
	LMSC	0.539	0.470	0.525
	Centroid MLRSSC	0.540	0.531	0.494
	Pairwise MLRSSC	0.519	0.516	0.482
	MVSOM	0.445	0.446	0.382
webKB	PCA	0.578	0.304	0.558
	Concat SC	0.277	0.172	0.558
	MVKM	0.564	0.321	0.460
	MEMTC	0.596	0.406	0.465
	MVEM	0.542	0.268	0.448
	LMSC	0.394	0.160	0.294
	Centroid MLRSSC	0.622	0.418	0.561
	Pairwise MLRSSC	0.632	0.405	0.581
	MVSOM	0.618	0.255	0.597

remains in finding a shared subspace while incorporating the diversity aspect. To summarize, this experimental results help drawing the following conclusions:

- Large data set and high-dimensional data affects the performance of multi-view methods. Therefore, considering a dimensionality reduction methods can help avoid this issue.
- Taking advantage of different representation schemes can improve the clustering performance of multi-view methods.
- Subspace based methods have good performance, yet these methods include multiple parameters and the optimization scheme is not evident to achieve.

8.4 Conclusion

We have presented in this chapter a categorization of existing multi-view clustering methods based on the fusion style of multi-view data. Three main integration scheme can be distinguished: late integration, co-training based methods, and subspace based methods. For each category, we have detailed a number of multi-view clustering algorithms, and the means of managing text data. Lastly, we have discussed the advantages and the limits of these methods and raised the following issues: the representation of multi-view text data relies on terms frequencies only, the intra-view properties of each view can be further leveraged to improve the clustering results, incorporating the specificity of each view in the clustering process can provide a better understanding of data. Multiple recent research studies focus on incomplete views with missing values. Some other works rely on incorporating deep learning into multi-view clustering to further discover hidden patterns shared among views.

References

1. M. Amini, N. Usunier, C. Goutte, Learning from multiple partially observed views-an application to multilingual text categorization, in *Advances in Neural Information Processing Systems* (2009), pp. 28–36
2. S. Bettoumi, C. Jlassi, N. Arous, Collaborative multi-view k-means clustering. Soft Comput. **23**(3), 937–945 (2019)
3. S. Bickel, T. Scheffer, Multi-view clustering, in *ICDM*, vol. 4 (2004), pp. 19–26
4. A. Blum, T. Mitchell, Combining labeled and unlabeled data with co-training, in *Proceedings of the Eleventh Annual Conference on Computational Learning Theory* (ACM, New York, 1998), pp. 92–100
5. S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, et al., Distributed optimization and statistical learning via the alternating direction method of multipliers. Found. Trends® Mach. Learn. **3**(1), 1–122 (2011)
6. M. Brbić, I. Kopriva, Multi-view low-rank sparse subspace clustering. Pattern Recogn. **73**, 247–258 (2018)
7. E. Bruno, S. Marchand-Maillet, Multiview clustering: a late fusion approach using latent models, in *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval* (2009), pp. 736–737

8. G. Chao, S. Sun, J. Bi, A survey on multi-view clustering (2017). arXiv preprint arXiv:1712.06246
9. M. Fraj, M.A.B. Hajkacem, N. Essoussi, Ensemble method for multi-view text clustering, in *International Conference on Computational Collective Intelligence* (Springer, Berlin, 2019), pp. 219–231
10. M. Fraj, M.A.B. Hajkacem, N. Essoussi, Self-organizing map for multi-view text clustering, in *International Conference on Big Data Analytics and Knowledge Discovery* (Springer, Berlin, 2020), pp. 396–408
11. E. Gaussier, C. Goutte, Relation between PLSA and NMF and implications, in *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval* (2005), pp. 601–602
12. T. Hofmann, Unsupervised learning by probabilistic latent semantic analysis. Mach. Learn. **42**(1), 177–196 (2001)
13. S.F. Hussain, M. Mushtaq, Z. Halim, Multi-view document clustering via ensemble method. J. Intell. Inform. Syst. **43**(1), 81–99 (2014)
14. D. Kim, D. Seo, S. Cho, P. Kang, Multi-co-training for document classification using various document representations: tF–IDF, LDA, and Doc2Vec. Inform. Sci. **477**, 15–29 (2019)
15. T. Kohonen, The self-organizing map. Proc. IEEE **78**(9), 1464–1480 (1990)
16. A. Kumar, H. Daumé, A co-training approach for multi-view spectral clustering, in *Proceedings of the 28th International Conference on Machine Learning (ICML-11)* (2011), pp. 393–400
17. A. Kumar, P. Rai, H. Daume, Co-regularized multi-view spectral clustering, in *Advances in Neural Information Processing Systems*, vol. 24 (2011)
18. B. Larsen, C. Aone, Fast and effective text mining using linear-time document clustering, in *Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. (CiteSeer, 1999), pp. 16–22
19. Y. Liang, Y. Pan, H. Lai, J. Yin, Robust multi-view clustering via inter-and-intra-view low rank fusion. Neurocomputing **385**, 220–230 (2020)
20. K.Y. Lin, L. Huang, C.D. Wang, H.Y. Chao, Multi-view proximity learning for clustering, in *International Conference on Database Systems for Advanced Applications* (Springer, Berlin, 2018), pp. 407–423
21. Z. Lin, R. Liu, Z. Su, Linearized alternating direction method with adaptive penalty for low-rank representation, in *Advances in Neural Information Processing Systems*, vol. 24 (2011)
22. J. Liu, C. Wang, J. Gao, J. Han, Multi-view clustering via joint nonnegative matrix factorization, in *Proceedings of the 2013 SIAM International Conference on Data Mining* (SIAM, 2013), pp. 252–260
23. F. Nie, G. Cai, X. Li, Multi-view clustering and semi-supervised classification with adaptive neighbours, in *AAAI* (2017), pp. 2408–2414
24. J.W. Reed, Y. Jiao, T.E. Potok, B.A. Klump, M.T. Elmore, A.R. Hurson, TF–ICF: a new term weighting scheme for clustering dynamic data streams, in *2006 5th International Conference on Machine Learning and Applications (ICMLA'06)* (IEEE, Piscataway, 2006), pp. 258–263
25. A. Strehl, J. Ghosh, Cluster ensembles—a knowledge reuse framework for combining multiple partitions. J. Mach. Learn. Res. **3**(Dec), 583–617 (2002)
26. Z. Tao, H. Liu, S. Li, Z. Ding, Y. Fu, From ensemble clustering to multi-view clustering, in *IJCAI* (2017)
27. G. Tzortzis, A. Likas, Kernel-based weighted multi-view clustering, in *2012 IEEE 12th International Conference on Data Mining* (IEEE, Piscataway, 2012), pp. 675–684
28. X. Wan, Co-training for cross-lingual sentiment classification, in *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1-Volume 1* (Association for Computational Linguistics, 2009), pp. 235–243
29. Q. Wang, Y. Dou, X. Liu, Q. Lv, S. Li, Multi-view clustering with extreme learning machine. Neurocomputing **214**, 483–494 (2016)

30. B. Wei, C. Pal, Cross lingual adaptation: an experiment on sentiment classifications, in *Proceedings of the ACL 2010 Conference Short Papers* (Association for Computational Linguistics, 2010), pp. 258–262
31. X. Xie, S. Sun, Multi-view clustering ensembles, in *2013 International Conference on Machine Learning and Cybernetics (ICMLC)*, vol. 1 (IEEE, Piscataway, 2013), pp. 51–56
32. Y. Yang, H. Wang, Multi-view clustering: a survey. Big Data Mining Anal. **1**(2), 83–107 (2018)
33. C. Zhang, Q. Hu, H. Fu, P. Zhu, X. Cao, Latent multi-view subspace clustering, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (2017), pp. 4279–4287
34. X. Zhang, L. Zong, X. Liu, H. Yu, Constrained NMF-based multi-view clustering on unmapped data, in *AAAI* (2015), pp. 3174–3180
35. X. Zhao, N. Evans, J.L. Dugelay, A subspace co-training framework for multi-view clustering. Pattern Recogn. Lett. **41**, 73–82 (2014)
36. L. Zheng, T. Li, C. Ding, Hierarchical ensemble clustering, in *2010 IEEE International Conference on Data Mining* (IEEE, Piscataway, 2010), pp. 1199–1204
37. F. Zhuang, G. Karypis, X. Ning, Q. He, Z. Shi, Multi-view learning via probabilistic latent semantic analysis. Inform. Sci. **199**, 20–30 (2012)