Neural Computing and Applications (2022) 34:6097–6117
https://doi.org/10.1007/s00521-021-06766-1

ORIGINAL ARTICLE

# Multi-view multi-label-based online method with threefold correlations and dynamic updating multi-region

Changming Zhu[1] · Shuaiping Guo[1] · Dujuan Cao[1] · YiTing Zhou[1] · Duoqian Miao[2] · Witold Pedrycz[3]

Received: 12 April 2021 / Accepted: 15 November 2021 / Published online: 9 January 2022


**Abstract**
Semi-supervised real-time generation multi-view multi-label data sets are widely encountered in practical applications. A key issue is how to process the data whose information including labels or features may be lost due to some unforeknowable factors. In our work, we develop a multi-view multi-label-based online method with threefold correlations and dynamic updating multi-region ($M^2CR$) to solve this issue. First, we adopt three kinds of correlations between features and labels to recover the missing information. Second, we process new arriving instances with dynamic updating multi-region. Experiments on classical multi-view multi-label data sets validate the effectiveness of $M^2CR$ in terms of classification, time performance, convergence, etc.

**Keywords** Multi-view and multi-label · Feature and label correlation · Dynamic updating multi-region

## 1 Introduction

### 1.1 Background

Four kinds of data exist in real-world applications, namely, single-view single-label data, multi-view single-label data, single-view multi-label data, and multi-view multi-label data. Among them, the multi-view multi-label data [1] are ubiquitous and have more application scenarios. For a multi-view multi-label data set, each instance exhibits multiple views and in each view, an instance can be labelled by multiple classes. Suppose in Fig. 1, there is a publicity website about the Imperial Palace and people can appreciate it through multiple ways (views) including text introduction, image introduction, and video introduction. In different views, the content of the website can be labelled differently. With textual information, this publicity website can be treated as an introduction to history. With image introduction, the website can be treated as an oil painting which describes landscape and history, while with video introduction, we can treat the website as a stereoscopic introduction about landscape and history rather than an oil painting. Obviously, in different views, this publicity website is labelled with different classes (history, landscape, oil painting, etc.) and this website is treated as a multi-view multi-label instance. Indeed, single-view single-label data, multi-view single-label data, single-view multi-label data can be treated as the special cases of multi-view multi-label data.

What's more, in practical applications, forms of the above-mentioned data sets are more complicated and three forms are general.

One is incomplete form. With the electromagnetic interference to sample equipments, obsolescence of storage devices, some label or feature information about the collected data may be lost. This makes the collected data be incomplete. Some classical solutions are developed to handle this case, including matrix completion, graph-regularized matrix factorization, visual assessment, imputation schemes [2–6], etc.

Another is real-time generation form. With the advent of big data, the collected data in practical applications exhibit real-time attribute. For example, when we collect data in

✉ Changming Zhu
cmzhu@shmtu.edu.cn

[1] College of Information Engineering, Shanghai Maritime University, Shanghai, China

[2] Department of Computer Science & Technology, Tongji University, Shanghai, China

[3] Department of Electrical & Computer Engineering, University of Alberta, Alberta, Canada


**Fig. 1** Example of a multi-view multi-label data set

YouTube, the data stored are constantly changing since people upload videos whenever and wherever possible. Feasible solutions to process this form are online methods [7–9].

The third is semi-supervised form. With the lack of manpower, only a small amount of the collected data can be labelled and provide useful prior knowledge. Such a data set is semi-supervised. In order to cope with those data sets, scholars develop some semi-supervised-related methods [10–14].

## 1.2 Motivation and proposal

**Motivation:** Although those above-related solutions are feasible for corresponding cases, some of them still have several shortcomings (details can be found in Sect. 2). For example, solutions for incomplete data have complicated frameworks and their computational complexities are high, especially in case of matrix factorization and matrix completion-related solutions. Even though Nyström technology [15–17] can reduce the computational complexity at some extends, we still have some room to reduce the level of complexity further. Then, solutions for real-time generation data and semi-supervised data have no ability to process incomplete data. Moreover, some solutions maybe pay more attention to data statistics and model mechanisms; then, they omit the valuable correlations between information or seldom use multiple kinds of correlations among information simultaneously. Here, these correlations reflect the relationship between information and maybe bring better performances for methods (definitions of correlations can be found in Sect. 3.1). Furthermore, among those solutions, only a few ones are developed for multi-view multi-label data sets and if a multi-view multi-label data set exhibits these forms simultaneously, how to solve it is an open problem. Thus we expect to come up with a method which can process a semi-supervised real-time generation multi-view multi-label data set with incomplete information and overcome the above-mentioned shortcomings.

**Proposal** Different from the above solutions to incomplete form, we argue that the missing information can be recovered by the correlations among features and labels. Simple speaking, the collected data are $\boldsymbol{D} = (\boldsymbol{X}, \boldsymbol{Y})$ where $\boldsymbol{X}$ represents the features and $\boldsymbol{Y}$ represents the labels. If $\boldsymbol{V}$, $\boldsymbol{S}$, $\boldsymbol{W}$ denote the feature-feature, label-label, feature-label correlations, respectively, they are collectively called threefold correlations (TC). Then we can use $\boldsymbol{VX}$, $\boldsymbol{SY}$ to recover the missing features and labels and adopt $\min_{w.r.t\boldsymbol{V},\boldsymbol{S},\boldsymbol{W}} ||\boldsymbol{VXW} - \boldsymbol{SY}||_2^2$ to adjust the recovered results with some fine tuning.

In terms of the real-time generation and semi-supervised aspects, we refer to the way given in [18]. In generally speaking, we update the parameters of the model momentarily once an instance arrives. But in our work, a few tricks are used. Different from what [18] adopts, when a new labelled instance arrives, we use correlations between features and ones between labels to recover the missing information and update the correlations simultaneously; when a new unlabelled instance arrives, we recover the missing features firstly and then adopt the notion of dynamic updating multi-region (DUMR) which updates current optimistic and pessimistic multiple regions of data to predict the labels so as to update the correlations finally.

With the combination of the above operations, we develop a multi-view multi-label-based online method with TC and DUMR ($M^2CR$).

## 1.3 Contribution

In order to solve the shortcomings of the above-mentioned solutions for incomplete data, real-time generation data, and semi-supervised data, we develop $M^2CR$ which consists of two main parts. One is recovering missing information with TC (this part corresponds to incomplete form), and the other is processing new arriving instances with DUMR, recovering methods, and correlations among information (this part corresponds to real-time generation form and semi-supervised form).

Different from the past solutions for incomplete data processing problems, TC has the following advantages. (1) With the usage of TC, when we recover missing information, we can use matrix multiplication rather than other matrix-based operations including matrix factorization and matrix completion which maybe bring a high computational complexity; (2) compared with imputation schemes and methods using correlations between features or labels,

recovering with TC can not only reduce the dependence on data statistics and model mechanisms but also consider more valuable correlations among information; (3) recovering with TC need not a very complicated framework.

Different from the past solutions for real-time generation data and semi-supervised data processing problems, DUMR has the following advantages. (1) Together with recovering missing information and updating correlations among information, DUMR can help to process semi-supervised real-time generation data sets with incomplete information better. This cannot be achieved by past solutions; (2) DUMR can label an new arriving unlabelled instance with a higher accuracy since it labels an instance with updating the current optimistic and pessimistic multiple regions dynamically. Indeed, updating those regions dynamically can make the classification interface be more consistent with the current data distribution.

What's more, besides the introduction of TC and DUMR, $M^2$CR has two another advantages. One is that $M^2$CR can process multi-view multi-label data sets, while other last solutions are developed for single-view single-label data, multi-view single-label data, or single-view multi-label data. The other is that for the semi-supervised real-time generation data, different from traditional methods, $M^2$CR processes new arriving instances with DUMR, recovering missing information, and updating correlations among information.

According to the above statements, these advantages can reflect the novelties of $M^2$CR and its differences and superiorities compared with the past solutions for incomplete data, real-time generation data, and semi-supervised data. Then we can summarize the contribution of our work, namely, using more feasible, simpler, informative methods to process semi-supervised real-time generation multi-view multi-label data sets with incomplete information.

## 2 Related work

In order to state the effectiveness of the proposed $M^2$CR clearly, we review the related work about the above-mentioned three cases.

### 2.1 Solutions for incomplete data

According to the literatures, most state-of-the-art methods to solve incomplete data can be generally divided into three categories.

Firstly, according to the retrospective statement of [19], matrix factorization and matrix completion-related methods can be used to estimate the missing information and a series of solutions have been proposed. For example, Peng et al. [20] present a hierarchical block-based incomplete data recovery method by using adaptive nonnegative matrix factorization (NMF) and this method does not need any user assistance and the training priors. Then, Tan et al. [2] propose a multi-view weak-label learning based on matrix completion to solve incomplete labels and noisy features. After that, Zhang et al. [21] propose an isomorphic linear correlation analysis method to linearly map multi-view data to a feature-isomorphic subspace and on the base of learned features, the data matrix of missing information can be modelled as a low-rank component plus a sparse contribution, and its matrix completion can be accomplished by an identical distribution pursuit completion model so that the missing information can be recovered. In this year, Niu et al. [22] use low-rank matrix factorization to learn a consensus representation matrix and they apply their method in multi-view clustering tasks; then, better clustering performances are obtained.

Secondly, as [6] stated, imputation schemes can also be chosen to perform on missing information. These schemes aim to fill in missing values with plausible values that are estimated based on observed values [23]. Traditional imputation schemes include statistical-based and machine learning-based methods. For the statistical-based imputation methods, people use the statistical information of observed data to fill in missing ones. Four most widely used statistical techniques are expectation management, linear regression, least squares, and mean/mode, and they have been applied in different tasks [24–26, 28]. For the machine learning-based ones, people focus on the designing of models and top four used techniques include clustering, decision tree, K-nearest neighbour, and random forest[29–32]. At present, neural network and granular computing which are active branches of machine learning have been widely applied in imputations of missing information. For example, Yoon et al. [33] present a generative adversarial imputation network (GAIN) for missing information imputation, where the generator outputs a completed vector conditioned on what is actually observed, and the discriminator attempts to determine which entries in the completed data were observed and which were imputed. GAIN has been shown to outperform many state-of-the-art imputation models. On the base of GAIN, Wang et al. [34] propose an unsupervised missing data imputation method named PC-GAIN, which utilizes potential category information to further enhance the imputation power. Meanwhile, Zhang et al. [35] introduce an encoder network into the standard generative adversarial network architecture and propose an end-to-end model to impute the missing information in a multivariate time series. Then, Hu et al. [4] develop an information granule-based classifier to abstract and refine the clusters centres in multi-class subspaces, and then the key structural relationship of the classes of data distributions can be captured. After that, the

incomplete data can be imputed as hybrid numeric and granular data and then be classified accurately.

Thirdly, in recent several years, many scholars focus on the correlations between features or labels to process the missing information. For example, Zhu et al. [1] exploit both global and local label correlations to construct a model and recover the missing label information through learning a latent label representation and optimizing the label manifolds. Sun et al. [36] develop a weakly supervised multi-label learning framework called WML-LSC. WML-LSC captures the desired feature information with low rank and sparse constrain scheme, and it recovers the missing label assignments and reconstructs the label assignment matrix with a linear self-recovery model which is constructed by a linear aggregation coefficient matrix reflecting the correlation of labels. Jiang et al. [37] propose a weakly supervised multi-label feature selection method called FSLCLC for feature selection with incomplete label information. During the procedure of FSLCLC, it recovers the missing labels of partially labelled training instances by label compressing and local feature correlations. Then Li et al. [38] propose a probabilistic principal component analysis to determine both long-term correlation information and short-term correlation information for features of structural health monitoring data and estimate missing feature information.

While through depth analysis, these above methods have corresponding disadvantages. For the matrix factorization and matrix completion-related methods, the high computational complexities about matrix factorization or matrix computation make against to the improvement of performances about methods. For those imputation schemes, they pay more attention to impute missing data with statistical information of observed data or mechanisms of the models themselves and omit the valuable correlations between information. For the methods using correlations between features or labels, although they can reduce computational complexities, they seldom use correlations between features, correlations between labels, correlations between features and labels simultaneously which maybe bring better results for incomplete case.

## 2.2 Solutions for real-time generation data and semi-supervised data

In real-world applications, limited by insufficient manpower and real-time data generation, real-time generation data and semi-supervised data always exist simultaneously. Thus, for convenience, we review the work about these two forms of data in this subsection in together.

Firstly, because semi-supervised data are ubiquitous in all trades and professions, thus there are many related methods which can be applied in various tasks including image classification, clustering, expressive representation learning, proposed for this form of data. For example, Nie et al. [11] propose a structural regularized semi-supervised model called AMUSE for multi-view data to solve the image classification problem. Different from traditional conventional graph-based multi-view learning models which learn a linear combination of views while assuming a priori weights distribution, AMUSE learns weights from a priori graph structure with the proposed structural regularization term which can lead to a more suitable structured graph for semi-supervised learning. Bai et al. [12] focus on semi-supervised clustering which uses pre-given knowledge as constraints to improve the clustering performance and then analyze the relations among multi-source constraints and propose an uniform representation for them. Then on the base of the relations and uniform representation, they propose a semi-supervised clustering algorithm called SC-MPI to find out a clustering that has a good cluster structure and a high consensus of all the sources of constraints. Jia et al. [13] develop a semi-supervised multi-view deep discriminant representation learning (SMDDRL) approach to learn an expressive representation from multi-view data. Different from existing joint or alignment multi-view representation learning methods, SMDDRL comprehensively exploits the consensus and complementary properties of multi-view data and reduces the redundancy of learned representations by employing shared and specific multi-view deep representation learning network as well as designing orthogonality and adversarial similarity constraints for it. Then by designing the deep metric learning and density clustering-based semi-supervised learning framework, SMDDRL effectively exploits the unlabelled data to enhance its representation learning performance.

Secondly, as a widely used solution for real-time generation data, many scholars develop some online methods from different tasks. For example, Zhang et al. [7] focus on online feature transformation learning in the context of multi-class object category recognition and develop an online linear feature transformation method with the consideration about the problem of online learning a feature transformation matrix expressed in the original feature space. Then these original features are mapped to kernel space, and online nonlinear feature transformation method is developed which can further improve the performance of online feature transformation learning in large-scale application. Li et al. [9] state that real industrial control systems require real-time response and uninterrupted operations and then they propose an adaptive regularized cost-sensitive multi-class online learning to process data stream in the field of industrial control and enhance the effectiveness of detecting cyberattacks in industrial control systems. Baisa [39] develops an online multi-object visual tracker using hypothesized and independent stochastic population (HISP) filter, and this tracker overcomes the problem of two or more objects

having similar identity. Li et al. [40] focus on the challenge of traditional graph neural network (GNN) frameworks, namely, difficult to handle the real-time changing network structures as well as scale to big graph data, and then they develop an attention-based heterogeneous multi-view graph neural network (aHMGNN) to address this issue. With the usage of variable graph data storage method and dynamic node neighbourhood sampling strategy, online system implementation of the aHMGNN can be implemented and the above challenge of GNN can be solved.

Thirdly, different from the above methods which are developed for semi-supervised data or real-time generation data only, some other methods are developed for semi-supervised real-time generation data sets. For example, Chen et al. [41] concentrate on graph-based multi-view learning and develop a multi-view semi-supervised learning for classification on dynamic networks (MSCD). MSCD can obtain a sparse and smooth combination of the views with the aid of total variation regularization for time-varying networks and have a better classification result when the processing data are time-varying and semi-supervised. Nie et al. [42] focus on fast and accurate classification of polarimetric synthetic aperture radar (PolSAR) data in dynamically changing environments, and they propose an online semi-supervised active learning framework for multi-view PolSAR data classification, called OSAM. On the base of relationships among multiple views and a randomized rule, OSAM can only query the labels of some informative incoming instances. Then on the base of co-regularized multi-view learning and graph regularization, OSAM can utilize both the incoming labelled and unlabelled instances to update the classifiers and classify the PolSAR data quickly and accurately. Besides these methods, our previous work [18] is also a method for semi-supervised real-time generation data sets. In [18], we develop an approach to generate additional unlabelled instances which possess useful discriminant information firstly and then we update the model continuously with an arriving labelled or unlabelled instance so that the performance of the model can be feasible for time-varying data.

Similarly, according to analysis, these above methods have corresponding disadvantages as well. Some of the above methods can only process semi-supervised data or real-time generation data, and some of them have no ability to process incomplete data or omit the useful correlations among information.

## 3 Preliminaries

In order to solve the disadvantages of the above-mentioned methods, we develop M$^2$CR. For stating the framework of M$^2$CR clearly, we show its two preliminaries firstly. One is a way to recover missing information with TC and the other is DUMR. What's more, we emphasize the motivations for the usage of TC and DUMR.

### 3.1 Recover missing information with threefold correlations and its motivation of the proposal

#### 3.1.1 What is threefold correlations

In practical applications, there are three kinds of correlations among the features and labels.

First, if two features have a strong correlation, their values depend on each other strongly. A simple example is age and appearance. In normal circumstances, an older age always implies an older appearance. If we use a matrix $\boldsymbol{X}$ to store the information of age and appearance, it is easy to find that their correlations $\boldsymbol{V}$ can be derived from $\boldsymbol{X}$. In other words, $\boldsymbol{V}$ should be a function of $\boldsymbol{X}$, namely, $\boldsymbol{V} = f_{\boldsymbol{V}}(\boldsymbol{X})$.

Second, in real-world, labels of some data have correlations. For example, among the logistics data, if a container loading steels belong to building-materials-container-class, people will classify this container as heavy-container-class with a high probability. In other words, if an instance belongs to class $a$ and it will belong to class $b$ simultaneously with a high probability, then label $a$ and label $b$ exist a strong correlation. If we use a matrix $\boldsymbol{Y}$ to store the label information of data, we can see that correlations between labels $\boldsymbol{S}$ are related with $\boldsymbol{Y}$ and it should be a function of $\boldsymbol{Y}$, namely, $\boldsymbol{S} = f_{\boldsymbol{S}}(\boldsymbol{Y})$.

Third, there are some correlations between features and labels. As we know, features are always used to decide labels of data [1]. But sometimes, not all features are relevant. For example, it is the grades in each subject that determine the total score, not gender and age. So when we label the instances, only part features play important roles. In other words, strong correlations just exist in some features and labels. Then, if we still adopt $\boldsymbol{X}$ to store the features and $\boldsymbol{Y}$ to represent the labels, their feature-label correlations $\boldsymbol{W}$ should be a function of $\boldsymbol{X}$ and $\boldsymbol{Y}$, namely, $\boldsymbol{W} = f_{\boldsymbol{W}}(\boldsymbol{X}, \boldsymbol{Y})$.

#### 3.1.2 Motivation of the proposal for TC

Once we determine correlations, since $\boldsymbol{V}$, $\boldsymbol{S}$, $\boldsymbol{W}$ are derived from $\boldsymbol{X}$ and $\boldsymbol{Y}$, thus we can recover the missing information by these correlations with a reverse operation. For example, $\boldsymbol{X} = f_{\boldsymbol{V}}^{-1}(\boldsymbol{V})$ and $\boldsymbol{Y} = f_{\boldsymbol{S}}^{-1}(\boldsymbol{S})$ can be treated as the recovered version of the original $\boldsymbol{X}$ and $\boldsymbol{Y}$, respectively. Moreover, the recovered $\boldsymbol{X}$ and $\boldsymbol{Y}$ can be tuned by $\boldsymbol{X} = f_{\boldsymbol{W}}^{-1}(\boldsymbol{W}, \boldsymbol{Y})$ and $\boldsymbol{Y} = f_{\boldsymbol{W}}^{-1}(\boldsymbol{W}, \boldsymbol{X})$, respectively. So we say the

motivation of the proposal for TC is recovering the missing information about features and labels with the usage of these three kinds of valuable correlations among information. Moreover, using TC rather than matrix-based operations including matrix factorization and matrix completion to recover missing information can reduce the computational complexity and bring a simpler framework.

## 3.2 Dynamic updating multi-region and its motivation of the proposal

### 3.2.1 Optimistic and pessimistic multiple regions

Suppose (1) $\boldsymbol{U} = \{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_i, \ldots, \boldsymbol{x}_{n-1}\}$ is a data pool which consists of $n-1$ instances ($i = 1, 2, \ldots, n-1$) and each instance of $\boldsymbol{U}$ can be represented by $d$ features and labelled by $c$ classes; (2) according to information of $\boldsymbol{U}$, we have possessed the feature-label correlations $\boldsymbol{W}$ and from $\boldsymbol{W}$, in terms of each label, we choose the features which correlations to this label are larger than $\eta$ and gather them to be a subset of features $A_h$ ($h = 1, 2, \ldots, c$).

First, $\boldsymbol{X} \subseteq \boldsymbol{U}$ be a feature matrix. Then for each $\boldsymbol{x}_i$, we use Eq. (1) to define a neighbourhood granularity $\phi(\boldsymbol{x}_i)$ w.r.t. $A_h$ where $\Gamma_{A_h}(\boldsymbol{x}_i, \boldsymbol{x}_p)$ can be treated as a metric function w.r.t. $A_h$ and it represents the similarities between instances. The computation method of $\Gamma_{A_h}(\boldsymbol{x}_i, \boldsymbol{x}_p)$ is given in Eq. (2) where $x_{ik}$ represents the value of $k$th feature of $\boldsymbol{x}_i$.

$$\phi_{A_h}(\boldsymbol{x}_i) = \{\boldsymbol{x}_p \mid \boldsymbol{x}_p \in \boldsymbol{U}, \Gamma_{A_h}(\boldsymbol{x}_i, \boldsymbol{x}_p) \leq \gamma, p = 1, 2, \ldots, n-1\} \tag{1}$$

$$\Gamma_{A_h}(\boldsymbol{x}_i, \boldsymbol{x}_p) = \left( \sum_{k=1}^{d} \left| x_{ik} - x_{pk} \right|^2 \right)^{\frac{1}{2}} \tag{2}$$

Second, for the $\boldsymbol{X}$, we give its lower and upper approximations w.r.t. $A_h$ which are denoted as $\underline{N_{A_h}}(\boldsymbol{X})$ and $\overline{N_{A_h}}(\boldsymbol{X})$ (see Eq. (3)). Indeed, these approximations are used to compute the cut regions of $\boldsymbol{U}$.

$$\begin{cases} \underline{N_{A_h}}(\boldsymbol{X}) = \{\boldsymbol{x}_i \mid \phi_{A_h}(\boldsymbol{x}_i) \subseteq \boldsymbol{X}, \boldsymbol{x}_i \in \boldsymbol{U}\} \\ \overline{N_{A_h}}(\boldsymbol{X}) = \{\boldsymbol{x}_i \mid \phi_{A_h}(\boldsymbol{x}_i) \cap \boldsymbol{X} \neq \varnothing, \boldsymbol{x}_i \in \boldsymbol{U}\} \end{cases} \tag{3}$$

Third, according to the lower and upper approximations of $\boldsymbol{X}$ w.r.t. $A_h$ ($h = 1, 2, \ldots, c$), we can partition the whole $\boldsymbol{U}$ into three regions, namely, positive region, negative region, and boundary region and denote them as $POS_{A_h}(\boldsymbol{X})$, $NEG_{A_h}(\boldsymbol{X})$, $BND_{A_h}(\boldsymbol{X})$ where

$$\begin{cases} POS_{A_h}(\boldsymbol{X}) = \underline{N_{A_h}}(\boldsymbol{X}) \\ NEG_{A_h}(\boldsymbol{X}) = \boldsymbol{U} - \overline{N_{A_h}}(\boldsymbol{X}) \\ BND_{A_h}(\boldsymbol{X}) = \overline{N_{A_h}}(\boldsymbol{X}) - \underline{N_{A_h}}(\boldsymbol{X}) \end{cases} \tag{4}$$

These regions indicate that in terms of the current subset of features $A_h$, which instances should be (should not be/may be) covered in $\boldsymbol{X}$.

But since different $A_h$s correspond to different three regions, thus in terms of $\boldsymbol{X}$, we should get its multi-granulation lower and upper approximations firstly and then get the corresponding three optimistic regions and three pessimistic regions. Optimistic lower and upper approximations of $\boldsymbol{X}$ are

$$\begin{cases} \underline{\sum_{h=1}^{c} N_{A_h}^{o}(\boldsymbol{X})} = \{\boldsymbol{x}_i \mid \bigvee_{h=1}^{c} \left(\phi_{A_h}(\boldsymbol{x}_i) \subseteq \boldsymbol{X}\right), \boldsymbol{x}_i \in \boldsymbol{U}\} \\ \overline{\sum_{h=1}^{c} N_{A_h}^{o}(\boldsymbol{X})} = \{\boldsymbol{x}_i \mid \bigwedge_{h=1}^{c} \left(\phi_{A_h}(\boldsymbol{x}_i) \cap \boldsymbol{X} \neq \varnothing\right), \boldsymbol{x}_i \in \boldsymbol{U}\} \end{cases} \tag{5}$$

and the corresponding optimistic positive region, negative region, and boundary region are given as below where '$\bigvee$' and '$\bigwedge$' denote the disjunction 'OR' and conjunction 'AND' operations, respectively.

$$\begin{cases} POS^{o}(\boldsymbol{X}) = \underline{\sum_{h=1}^{c} N_{A_h}^{o}(\boldsymbol{X})} \\ NEG^{o}(\boldsymbol{X}) = \boldsymbol{U} - \overline{\sum_{h=1}^{c} N_{A_h}^{o}(\boldsymbol{X})} \\ BND^{o}(\boldsymbol{X}) = \overline{\sum_{h=1}^{c} N_{A_h}^{o}(\boldsymbol{X})} - \underline{\sum_{h=1}^{c} N_{A_h}^{o}(\boldsymbol{X})} \end{cases} \tag{6}$$

In the same way, the pessimistic lower and upper approximations of $\boldsymbol{X}$ are

$$\begin{cases} \underline{\sum_{h=1}^{c} N_{A_h}^{p}(\boldsymbol{X})} = \{\boldsymbol{x}_i \mid \bigwedge_{h=1}^{c} \left(\phi_{A_h}(\boldsymbol{x}_i) \subseteq \boldsymbol{X}\right), \boldsymbol{x}_i \in \boldsymbol{U}\} \\ \overline{\sum_{h=1}^{c} N_{A_h}^{p}(\boldsymbol{X})} = \{\boldsymbol{x}_i \mid \bigvee_{h=1}^{c} \left(\phi_{A_h}(\boldsymbol{x}_i) \cap \boldsymbol{X} \neq \varnothing\right), \boldsymbol{x}_i \in \boldsymbol{U}\} \end{cases} \tag{7}$$

and the corresponding pessimistic positive region, negative region, and boundary region are given as below.

$$\begin{cases} POS^{p}(\boldsymbol{X}) = \underline{\sum_{h=1}^{c} N_{A_h}^{p}(\boldsymbol{X})} \\ NEG^{p}(\boldsymbol{X}) = \boldsymbol{U} - \overline{\sum_{h=1}^{c} N_{A_h}^{p}(\boldsymbol{X})} \\ BND^{p}(\boldsymbol{X}) = \overline{\sum_{h=1}^{c} N_{A_h}^{p}(\boldsymbol{X})} - \underline{\sum_{h=1}^{c} N_{A_h}^{o}(\boldsymbol{X})} \end{cases} \tag{8}$$

### 3.2.2 Dynamic updating optimistic and pessimistic multiple regions

When there is a new instance $\boldsymbol{x}_s$ which also can be represented by $d$ features and labelled by $c$ classes arrives dynamically, we should update the current optimistic and

pessimistic multiple regions of $\boldsymbol{X}$ and the method is given as below.

First, for each $A_h$, we let $R_{A_h}^{\boldsymbol{X},\boldsymbol{x}_s} = [r_{is}^{A_h}]_{(n-1)\times 1}$ be an incremental relation where $i = 1, 2, \ldots, n-1$ and $s$ corresponds to $\boldsymbol{x}_s$. Then $r_{is}^{A_h}$ is computed as follows.

$$r_{is}^{A_h} = \begin{cases} 1, & \text{if } \Gamma_{A_h}(x_i, x_s) \le \gamma \; i = 1, 2, \ldots, n-1 \\ 0, & \text{otherwise} \end{cases} \tag{9}$$

Second, $D_{A_h} = [d_{ib}^{A_h}]_{(n-1)\times(n-1)}$ is the neighbourhood relation matrix of $\boldsymbol{U}$ and $d_{ib}^{A_h}$ is computed with Eq. (10).

$$d_{ib}^{A_h} = \begin{cases} 1, & \text{if } \Gamma_{A_h}(x_i, x_b) \le \gamma i, b = 1, 2, \ldots, n-1 \\ 0, & \text{otherwise} \end{cases} \tag{10}$$

Then

$$D_{A_h}^{'} = \begin{pmatrix} D_{A_h} & R_{A_h}^{\boldsymbol{X},\boldsymbol{x}_s} \\ R_{A_h}^{\boldsymbol{X},\boldsymbol{x}_s T} & 1 \end{pmatrix} \tag{11}$$

is an augmented neighbourhood relation matrix and its dimension is $n \times n$.

Third, we set a matrix $B(\boldsymbol{X}) = [b_i]_{(n-1)\times 1}$ to denote whether instances belong to the $\boldsymbol{X}$. Here,

$$b_i = \begin{cases} 1, & if \quad \boldsymbol{x}_i \in \boldsymbol{X} \quad i = 1, 2, \ldots, n-1 \\ 0, & otherwise \end{cases} \tag{12}$$

.

Fourth, since $\boldsymbol{x}_s$ is a new arriving instance, thus the $\boldsymbol{U}$ is updated as $\boldsymbol{U}^{'} = \boldsymbol{U} \cup \boldsymbol{x}_s$ and $\boldsymbol{X}$ is also updated as $\boldsymbol{X}^{'} = \boldsymbol{X} \cup \boldsymbol{x}_s$. As a result, we can get a new matrix to denote whether instances belong to the $\boldsymbol{X}^{'}$, namely, $B(\boldsymbol{X})$ is updated as $B(\boldsymbol{X}^{'}) = \begin{pmatrix} B(\boldsymbol{X}) \\ b_s \end{pmatrix}$ where

$$b_s = \begin{cases} 1, & \text{if } x_s \text{ is unlabelled} \\ 0, & \text{otherwise} \end{cases} \tag{13}$$

Moreover, we can also set a column vector $\boldsymbol{C}^{'}$ which elements are all 1 and its dimension is $n \times 1$. Then we let

$$Q_{A_h}^{\Uparrow}(\boldsymbol{X}^{'}) = D_{A_h}^{'} \times B(\boldsymbol{X}^{'}) \quad \& \quad Q_{A_h}^{\Downarrow}(\boldsymbol{X}^{'}) = D_{A_h}^{'} \times \boldsymbol{C}^{'} \tag{14}$$

be two intermediate matrices corresponding to $A_h$ and they are used to update the multiple regions for $\boldsymbol{X}^{'}$.

Fifth, we let $Q_{A_h}(\boldsymbol{X}^{'}) = Q_{A_h}^{\Uparrow}(\boldsymbol{X}^{'})/.Q_{A_h}^{\Downarrow}(\boldsymbol{X}^{'})$ where '/.' denotes the matrix dot divide. Each element of $Q_{A_h}(\boldsymbol{X}^{'})$ is recorded as $q_{A_h}^{i}$. Then, according to $Q_{A_h}(\boldsymbol{X}^{'})$, we get the matrices corresponding to the positive, boundary, and negative regions of $\boldsymbol{X}^{'}$ with Eq. (15) where $q_{A_h}^{i-POS}$, $q_{A_h}^{i-NEG}$, and $q_{A_h}^{i-BND}$ can be computed by Eqs. (16) $\sim$ (18) where $i = 1, 2, \ldots, n$ and $h = 1, 2, \ldots, c$.

$$\begin{cases} Q_{A_h}^{POS}(\boldsymbol{X}^{'}) = [q_{A_h}^{i-POS}]_{n\times 1} \\ Q_{A_h}^{NEG}(\boldsymbol{X}^{'}) = [q_{A_h}^{i-NEG}]_{n\times 1} \\ Q_{A_h}^{BND}(\boldsymbol{X}^{'}) = [q_{A_h}^{i-BND}]_{n\times 1} \end{cases} \tag{15}$$

$$q_{A_h}^{i-POS} = \begin{cases} 1, & if \quad q_{A_h}^{i} = 1 \\ 0, & otherwise \end{cases} \tag{16}$$

$$q_{A_h}^{i-NEG} = \begin{cases} 1, & if \quad q_{A_h}^{i} = 0 \\ 0, & otherwise \end{cases} \tag{17}$$

$$q_{A_h}^{i-BND} = \begin{cases} 1, & if \quad 0 < q_{A_h}^{i} < 1 \\ 0, & otherwise \end{cases} \tag{18}$$

Sixth, consider the $c$ $A_h$s, we compute the characteristic functions of the three optimistic regions, i.e. $H(POS^o(\boldsymbol{X}^{'}))$, $H(NEG^o(\boldsymbol{X}^{'}))$, $H(BND^o(\boldsymbol{X}^{'}))$, and three pessimistic regions, i.e. $H(POS^p(\boldsymbol{X}^{'}))$, $H(NEG^p(\boldsymbol{X}^{'}))$, $H(BND^p(\boldsymbol{X}^{'}))$. Please see the below equation.

$$\begin{cases} H(POS^o(\boldsymbol{X}^{'})) = \max\limits_{h=1}^{c}(Q_{A_h}^{POS}(\boldsymbol{X}^{'})) \\ H(NEG^o(\boldsymbol{X}^{'})) = \max\limits_{h=1}^{c}(Q_{A_h}^{NEG}(\boldsymbol{X}^{'})) \\ H(BND^o(\boldsymbol{X}^{'})) = \min\limits_{h=1}^{c}(Q_{A_h}^{BND}(\boldsymbol{X}^{'})) \\ H(POS^p(\boldsymbol{X}^{'})) = \min\limits_{h=1}^{c}(Q_{A_h}^{POS}(\boldsymbol{X}^{'})) \\ H(NEG^p(\boldsymbol{X}^{'})) = \min\limits_{h=1}^{c}(Q_{A_h}^{NEG}(\boldsymbol{X}^{'})) \\ H(BND^p(\boldsymbol{X}^{'})) = \max\limits_{h=1}^{c}(Q_{A_h}^{BND}(\boldsymbol{X}^{'})) \end{cases} \tag{19}$$

Here, *min* and *max* represent the following operations and $\boldsymbol{A}, \boldsymbol{B}$ are two $n \times n$ matrices, while $a_{ij}$, $b_{ij}$ represent their $i$th row $j$th column elements, respectively.

$$\max(\boldsymbol{A}, \boldsymbol{B}) = [\max(a_{ij}, b_{ij})]_{n\times n} \tag{20}$$

$$\min(\boldsymbol{A}, \boldsymbol{B}) = [\min(a_{ij}, b_{ij})]_{n\times n} \tag{21}$$

Finally, according to Eq. (19), for the $\boldsymbol{X}^{'}$, we can get its three optimistic regions and three pessimistic regions, namely, $POS^o(\boldsymbol{X}^{'})$, $NEG^o(\boldsymbol{X}^{'})$, $BND^o(\boldsymbol{X}^{'})$, $POS^p(\boldsymbol{X}^{'})$, $NEG^p(\boldsymbol{X}^{'})$, $BND^p(\boldsymbol{X}^{'})$. The construction method is straight forward, namely, if the $i$th element of $H(POS^o(\boldsymbol{X}^{'}))$ is 1, then $\boldsymbol{x}_i \in POS^o(\boldsymbol{X}^{'})$. For others, the construction method is same.

#### 3.2.3 Motivation of the proposal for DUMR

With the usage of DUMR, we can process a semi-supervised real-time generation data set better. For a data set, we can update the current optimistic and pessimistic multiple regions when a new instance arrives and get which instance belongs to a label definitely, which is not, and which is not

sure. In other words, we can label an instance with a more higher accuracy and this is also the motivation of the proposal for DUMR.

## 4 Multi-view multi-label-based online method with threefold correlations and dynamic updating multi-region ($M^2CR$)

### 4.1 Data preparation

Figure 2 shows the description of a multi-view multi-label data set. Suppose $\boldsymbol{X} \in \mathbb{R}^{d\times(n-1)}$ is a multi-view multi-label data set and it consists of $n-1$ instances. Each instance can be described by $m$ views, $d$ features, and $c$ classes. If we suppose $x^j_{ik_j}$ is the $k_j$th feature of $j$th view for $i$th instance, then $\boldsymbol{x}^j_i = (x^j_{i1},\ldots,x^j_{ik_j},\ldots,x^j_{id_j})^T \in \mathbb{R}^{d_j\times 1}$ represents the $j$th view for $i$th instance and $\boldsymbol{x}_i = (\boldsymbol{x}^{1^T}_i,\ldots,\boldsymbol{x}^{j^T}_i,\ldots,\boldsymbol{x}^{m^T}_i)^T \in \mathbb{R}^{d\times 1}$ denotes the $i$th instance. Here, $d_j$ is number of features for $j$th view and $d = \sum_{j=1}^{m} d_j$, $i \in [1, n-1]$, $j \in [1, m]$, $k_j \in [1, d_j]$. Then, we let $\boldsymbol{X}^j = (\boldsymbol{x}^j_1;\ldots;\boldsymbol{x}^j_i;\ldots;\boldsymbol{x}^j_{(n-1)}) \in \mathbb{R}^{d_j\times(n-1)}$ be the $j$th view for $\boldsymbol{X}$ and its $k_j$th feature vector is $\overline{\boldsymbol{x}^j_{k_j}} = (x^j_{1k_j},\ldots,x^j_{ik_j},\ldots,x^j_{(n-1)k_j}) \in \mathbb{R}^{1\times(n-1)}$.

In terms of the labels of $\boldsymbol{X}$, namely, $\boldsymbol{Y} \in \mathbb{R}^{c\times(n-1)}$, its presentation and definition are similar and $c$ is the total number of labels. Namely, $y^j_{ih_j}$ is the $h_j$th label of $j$th view for $i$th instance and $\boldsymbol{y}^j_i = (y^j_{i1},\ldots,y^j_{ih_j},\ldots,y^j_{ic_j})^T \in \mathbb{R}^{c_j\times 1}$ represents the labels of $j$th view for $i$th instance and $c_j$ is the number of labels in $j$th view. Then, $\boldsymbol{Y}^j = (\boldsymbol{y}^j_1;\ldots;\boldsymbol{y}^j_i;\ldots;\boldsymbol{y}^j_{(n-1)}) \in \mathbb{R}^{c_j\times(n-1)}$ represents the labels of $j$th view for $\boldsymbol{X}$ and its $h_j$th label vector is $\overline{\boldsymbol{y}^j_{h_j}} = (y^j_{1h_j},\ldots,y^j_{ih_j},\ldots,y^j_{(n-1)h_j}) \in \mathbb{R}^{1\times(n-1)}$. Then, $\boldsymbol{y}_i = (\boldsymbol{y}^{1^T}_i,\ldots,\boldsymbol{y}^{j^T}_i,\ldots,\boldsymbol{y}^{m^T}_i)^T \in \mathbb{R}^{c\times 1}$ denotes the labels of $i$th instance. Here, $c = \sum_{j=1}^{m} c_j$, $h_j \in [1, c_j]$.

Then for the initial collected data, we recover them with any feasible method including matrix completion [2] firstly to lay the foundation for better recovery further.

### 4.2 Initialize correlations between features and labels

On the base of the present $\boldsymbol{X}$ and $\boldsymbol{Y}$ which have been recovered initially, we can compute the corresponding feature-feature, label-label, feature-label correlations. Here, we take the $j$th view for example.

First, for $\boldsymbol{X}^j$, its $k_j$th feature vector is $\overline{\boldsymbol{x}^j_{k_j}}$. Then, if two features (for example, the $k_j$th feature and the $p_j$th feature) have a strong correlation, then values or distributions of instances $\boldsymbol{x}$ in $k_j$th feature are similar with the ones in $p_j$th feature with a high probability. Further, the feature vectors $\overline{\boldsymbol{x}^j_{k_j}}$ and $\overline{\boldsymbol{x}^j_{p_j}}$ should be similar. Thus, the feature-feature correlations $\boldsymbol{V}^j = \{[\boldsymbol{V}^j]_{k_jp_j}\} \in \mathbb{R}^{d_j\times d_j}$ can be computed by feature vectors. Refer to the function of a cosine measure, in $\boldsymbol{V}^j$, its $k_j$th row and $p_j$th column element $[\boldsymbol{V}^j]_{k_jp_j} = \frac{\overline{\boldsymbol{x}^j_{k_j}}\cdot\overline{\boldsymbol{x}^j_{p_j}}^T}{\left|\left|\overline{\boldsymbol{x}^j_{k_j}}\right|\right|\left|\left|\overline{\boldsymbol{x}^j_{p_j}}\right|\right|}$ represents the correlation between the $k_j$th feature and the $p_j$th feature.

Second, for $\boldsymbol{Y}^j$, its $h_j$th label vector is $\overline{\boldsymbol{y}^j_{h_j}}$. Then, if two labels (for example, the $h_j$th label and the $q_j$th label) have a strong correlation, then instances belonging to the $h_j$th label will belong to the $q_j$th label simultaneously with a high probability. Further, the label vectors $\overline{\boldsymbol{y}^j_{h_j}}$ and $\overline{\boldsymbol{y}^j_{q_j}}$ should be similar. Thus, the label-label correlations $\boldsymbol{S}^j = \{[\boldsymbol{S}^j]_{h_jq_j}\} \in \mathbb{R}^{c_j\times c_j}$ can be computed by label vectors. In $\boldsymbol{S}^j$, its $h_j$th row and $q_j$th column element $[\boldsymbol{S}^j]_{h_jq_j} = \frac{\overline{\boldsymbol{y}^j_{h_j}}\cdot\overline{\boldsymbol{y}^j_{q_j}}^T}{\left|\left|\overline{\boldsymbol{y}^j_{h_j}}\right|\right|\left|\left|\overline{\boldsymbol{y}^j_{q_j}}\right|\right|}$ represents the correlation between the $h_j$th label and the $q_j$th label.

Third, if the correlation between $k_j$th feature and $h_j$th label is strong, then $k_j$th feature plays an important role on whether the instance belongs to $h_j$th label. Thus, the feature-label correlations $\boldsymbol{W}^j = \{[\boldsymbol{W}^j]_{k_jh_j}\} \in \mathbb{R}^{c_j\times d_j}$ can be computed by feature vectors and label vectors. Refer to the traditional pattern recognition knowledge, $\boldsymbol{X}$ can be mapped into $\boldsymbol{Y}$ with a mapping matrix $\boldsymbol{W}$ which can be treated as the weight matrix, for example, $\boldsymbol{Y} = \boldsymbol{W}\boldsymbol{X}$. Thus, with the help of this knowledge, $\overline{\boldsymbol{x}^j_{k_j}}$ can be mapped into $\overline{\boldsymbol{y}^j_{h_j}}$ with a weight and we treat this weight as the corresponding feature-label correlation. In other words, $\boldsymbol{W}^j = \boldsymbol{Y}^j\boldsymbol{X}^{j-1}$ is a feasible method to compute the feature-label correlations in $j$th view and its $k_j$th row and $h_j$th column element $[\boldsymbol{W}^j]_{k_jh_j}$ represents the correlation between the $k_j$th feature and the $h_j$th label.

### 4.3 DUMR on $\boldsymbol{X}^j$

Suppose according to $\boldsymbol{Y}^j$, instances in $\boldsymbol{X}^j$ can be divided into $c_j$ clusters. Each cluster $\boldsymbol{X}^j_{h_j} \in \mathbb{R}^{d_j\times n'}$ corresponds to a

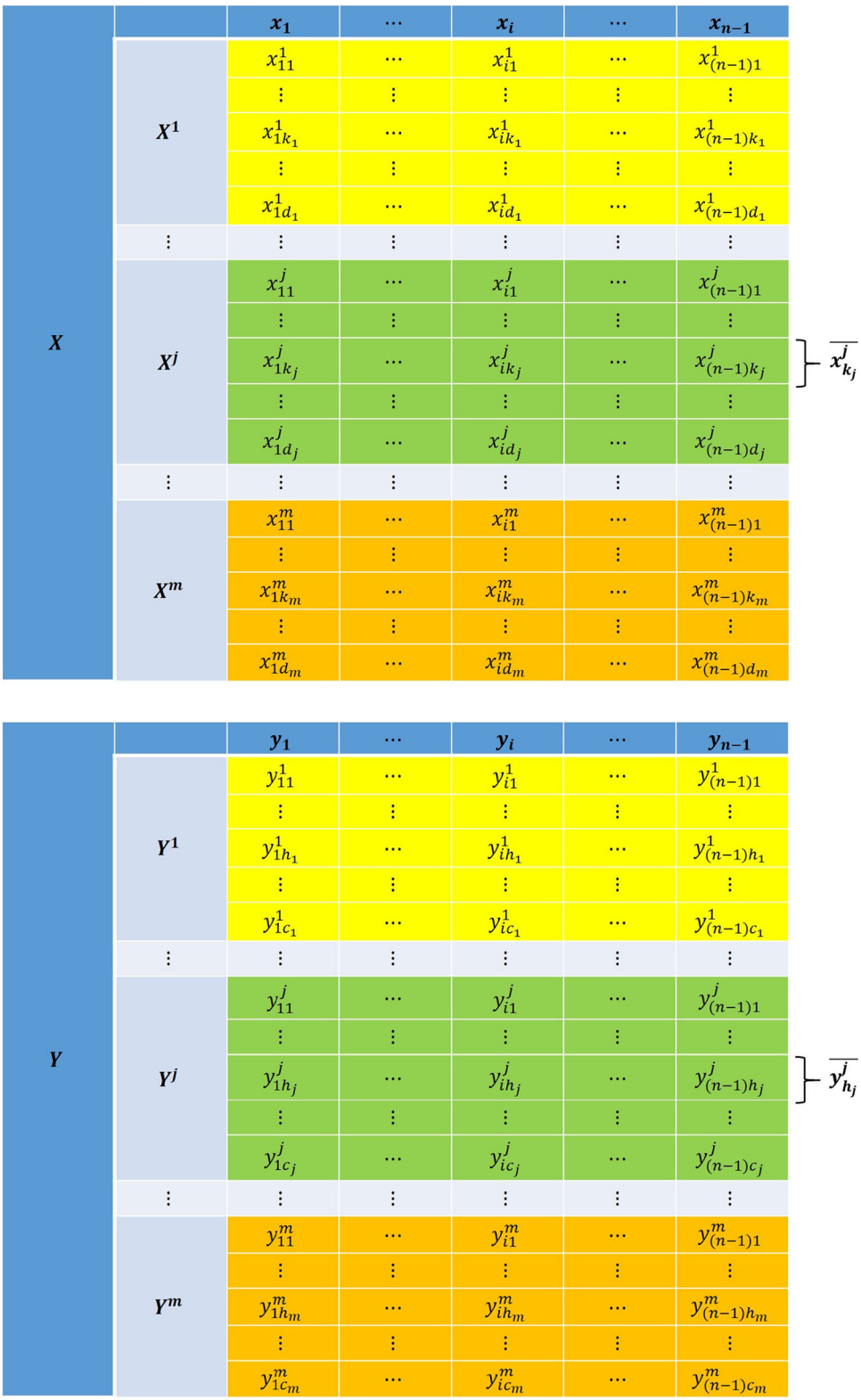


**Fig. 2** Description of a multi-view multi-label data set (top: features; bottom: labels)

label, and it covers $n'$ instances $\boldsymbol{x}_i^j$s which belong to the $h_j$th label. By this operation, instance $\boldsymbol{x}_i^j$ may be covered by multiple clusters because it belongs to multiple labels simultaneously.

Then according to the procedure of DUMR, the optimistic and pessimistic multiple regions of $\boldsymbol{X}_{h_j}^j$ are

$POS^o(\boldsymbol{X}^j_{h_j})$, $NEG^o(\boldsymbol{X}^j_{h_j})$, $BND^o(\boldsymbol{X}^j_{h_j})$, $POS^p(\boldsymbol{X}^j_{h_j})$, $NEG^p(\boldsymbol{X}^j_{h_j})$, and $BND^p(\boldsymbol{X}^j_{h_j})$, respectively.

### 4.4 Case 1: a new labelled instance arrives

Suppose $\boldsymbol{x}_n$ is a new arriving instance and its label is $\boldsymbol{y}_n$. The description of it is same as other instances. Then for $\boldsymbol{x}^j_n \in \mathbb{R}^{d_j \times 1}$ and $\boldsymbol{y}^j_n \in \mathbb{R}^{c_j \times 1}$, we recover them firstly with the solution of optimization problem (see Eq. (22)). The solution of this optimization problem can adopt the simple gradient descent optimization method.

$$\min_{w.r.t.\boldsymbol{V}^j,\boldsymbol{S}^j,\boldsymbol{W}^j} \left|\left|\boldsymbol{W}^j\boldsymbol{V}^j\boldsymbol{x}^j_n - \boldsymbol{S}^j\boldsymbol{y}^j_n\right|\right|^2_2 \tag{22}$$

After solving this optimization problem, we can get the updated corresponding correlations and get $\boldsymbol{W}^{j'}$, $\boldsymbol{V}^{j'}$, $\boldsymbol{S}^{j'}$. Then we can use $\boldsymbol{x}^{j'}_n = \boldsymbol{V}^j \cdot \boldsymbol{x}^j_n \in \mathbb{R}^{d_j \times 1}$ and $\boldsymbol{y}^{j'}_n = \boldsymbol{S}^j \cdot \boldsymbol{y}^j_n \in \mathbb{R}^{c_j \times 1}$ to recover the $\boldsymbol{x}_n$ and update the corresponding $\boldsymbol{X}^j$, $\boldsymbol{Y}^j$, $\boldsymbol{X}^j_{h_j}$ to get $\boldsymbol{X}^{j'}$, $\boldsymbol{Y}^{j'}$, $\boldsymbol{X}^{j'}_{h_j}$.

### 4.5 Case 2: a new unlabelled instance arrives

Suppose the new arriving instance is $\boldsymbol{x}_n$ and its description is same as other instances, then for each view of it, $\boldsymbol{x}^j_n \in \mathbb{R}^{d_j \times 1}$, we recover it firstly and the recovered result is $\boldsymbol{x}^{j'}_n = \boldsymbol{V}^j \cdot \boldsymbol{x}^j_n \in \mathbb{R}^{d_j \times 1}$.

Then, we let $\boldsymbol{X}^{j'} = \boldsymbol{X}^j \cup \{\boldsymbol{x}^{j'}_n\} \in \mathbb{R}^{d_j \times n}$ and get the optimistic and pessimistic multiple regions of $\boldsymbol{X}^{j'}_{h_j}$, namely, $POS^o(\boldsymbol{X}^{j'}_{h_j})$, $NEG^o(\boldsymbol{X}^{j'}_{h_j})$, $BND^o(\boldsymbol{X}^{j'}_{h_j})$, $POS^p(\boldsymbol{X}^{j'}_{h_j})$, $NEG^p(\boldsymbol{X}^{j'}_{h_j})$, and $BND^p(\boldsymbol{X}^{j'}_{h_j})$.

After that, according to the gotten optimistic and pessimistic multiple regions, we distribute the $\boldsymbol{x}^{j'}_n$ into multiple feasible clusters and give the corresponding $\boldsymbol{y}^j_n$.

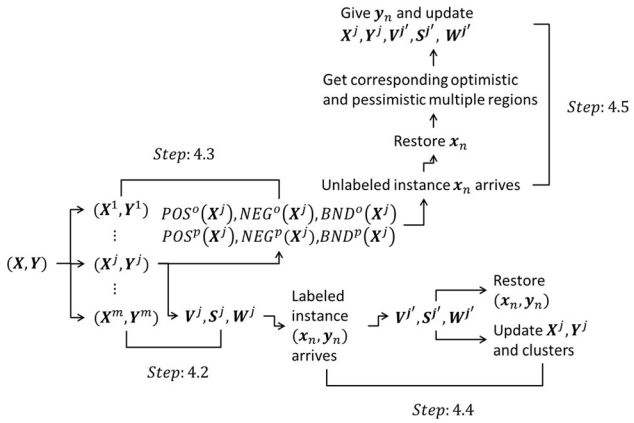


**Fig. 3** Description of M$^2$CR

Finally, we update the $\boldsymbol{Y}^{j'} = \boldsymbol{Y}^j \cup \{\boldsymbol{y}^j_n\} \in \mathbb{R}^{c_j \times n}$ and the corresponding correlations, $\boldsymbol{W}^{j'}$, $\boldsymbol{V}^{j'}$, $\boldsymbol{S}^{j'}$ according to the updated $\boldsymbol{X}^{j'}$ and $\boldsymbol{Y}^{j'}$.

### 4.6 Test the performance

Once we update the correlations and recover the missing information, the present model can be treated as the best form in terms of the present collected data. Now if some test instances arrive (test instances are the ones which are used to test the performance of the model rather than training the model), we can refer to the traditional pattern recognition knowledge and predict the labels of these test instances by $\boldsymbol{Y}^{jt} = \boldsymbol{W}^{j'}\boldsymbol{X}^{jt}$ where $\boldsymbol{X}^{jt}$ represents the test instances and their labels are $\boldsymbol{Y}^{jt}$.

Here, we use Fig. 3 to show the framework of the proposed M$^2$CR.

## 5 Experiments

### 5.1 Experimental setting

#### 5.1.1 Data set

We adopt some multi-view multi-label data sets including NUS-WIDE [44, 45], CoNLL-2003 [46], Corel5k [47], Mirflickr [48], Iaprtc12 [49], Espgame [50], EURLex-4K [51] for experiments (see Table 1).

#### 5.1.2 Compared method

We select the below four types of multi-view multi-label learning methods for comparison.

(1) Methods considering correlations: MMP[1] [52], MVCRF[2] [46], MVLE[3] [53], TG-CMTF[4] [54]. MVLE pays attention to feature-label correlations and others consider label-label correlations.

(2) Methods considering online learning: MVECF[5] [55], MLFSNRS[6] [56]. MVECF is an automated model to identify mentions of product defects from social media, such as online discussion forums; MLFSNRS proposes a neighbourhood relation to effectively

[1] multi-view based multi-label propagation for image annotation

[2] multi-view conditional random fields

[3] multi-view label embedding model

[4] trigraph regularized collective matrix tri-factorization framework

[5] multi-view ensemble learning with contextual features

[6] multi-label streaming feature selection based on neighbourhood rough set

**Table 1** Detailed information of multi-view multi-label data sets

| Order | Data | Instance | Label | View |
|---|---|---|---|---|
| 1 | NUS-WIDE | 810 | 81 | 6 |
| 2 | CoNLL-2003 | 20744 | 4 | 2 |
| 3 | Corel5k | 4999 | 260 | 3 |
| 4 | Mirflickr | 17500 | 457 | 3 |
| 5 | Iaprtc12 | 6952 | 291 | 3 |
| 6 | Espgame | 7081 | 268 | 3 |
| 7 | EURLex-4K | 19348 | 3993 | 3 |

solve the problem of granularity selection in neighbourhood rough set [57, 58] and solves online streaming feature selection.

(3) Methods considering semi-supervised cases: MVOCNMF[7] [59], MVMC[8] [60], MV$^3$MR[9] [61]. MVOCNMF learns the low-dimensional representations of data by the constrained NMF technique and puts forward an orthonormality constraint term to obtain the desirable representations for each view; MVMC applies a cross-validation strategy on the labelled set to learn the view combination weights effectively; MV$^3$MR exploits the complementary property of different features and discovers the intrinsic local geometry of the compact support shared by different features under the theme of manifold regularization.

(4) Methods without the consideration about the correlations, online learning, and semi-supervised cases: ICM2L[10] [62], BRSMVML[11] [63]. ICM2L can explore the individuality and commonality information of multi-label multi-view data in a unified model explicitly; BRSMVML can process multi-view image classification, via embedding a block-row regularizer into the multi-view multi-label framework.

### 5.1.3 Parameter setting and selection of best parameters

Parameters of the compared methods refer to respective references and then for M$^2$CR, its parameter setting is given below.

(A) We select $\{10\%, 20\%, \ldots, 90\%\}$ instances of each data set for training and validation and then the left are used for test. For the training and validation set, we adopt 10-fold cross-validation for experiments. Namely, we divide this set into 10 partitions and each round, and we select 1 partition as the validation set and the rest 9 partitions as the training set. On the base of the training set, we can train a learning machine and then we can use the validation set to validate its performance. After 10 rounds, we can get the average performance. In other words, for each training and validation set, 90% instances are used for training and the left 10% instances are used for validation. Then in the semi-supervised case, for each partition, we further select $\{10\%, 20\%, \ldots, 90\%\}$ instances as the labelled ones and the rest is the unlabelled part. Here, during the experiments, if a partition is used for validation, we suppose all instances are labelled temporarily. Moreover, for each instance, we remove $\{10\%, 20\%, \ldots, 90\%\}$ elements (features or labels or them both) for the incomplete experiments. All the above selections are random.

(B) There are another two adjustable parameters in M$^2$CR. One is $\eta$ which is used to choose features according to feature-label correlations and the other is $\gamma$ which is used to compose the neighbourhood granularity. Here, $\eta$ is selected from the set $\{0.1, 0.2, \ldots, 0.8, 0.9\}$ and $\gamma$ can be selected from the set $\{0.05, 0.1, \ldots, 0.95, 1, 2, \ldots, 9, 10\}$.

Then, in order to select the best parameters for a data set when a multi-view multi-label learning method is used, we adopt the similar way given in reference [43] so as to optimize the hyper-parameters. In simple terms, (1) for each data set, we split and process it firstly according to the setting of Sect. 5.1.3(A). (2) Then we fix the partitions and processing of the data set in the following experiments and set the adjustable parameters according to the setting of Sect. 5.1.3(B). (3) For the 10 partitions, we select 1 partition as the validation set and the rest 9 partitions as the training set. Then, on the base of training set and validation set, we carry out a multi-view multi-label learning method to get the classification performances. We repeat the experiments for ten rounds according to these 10 partitions and get the average classification performances. Here, as what we said before, for the validation partition, we suppose all instances are labelled temporarily. (4) We try all parameter combinations according to Sects. 5.1.3(A) and 5.1.3(B) until we get the best average classification performances of the learning method. Then, the parameters corresponding to best average classification performances

[7] semi-supervised multi-view clustering based on orthonormality-constrained NMF

[8] multi-view matrix completion

[9] multi-view vector-valued manifold regularization

[10] individuality- and commonality-based multi-view multi-label learning

[11] block-row sparse multi-view multi-label learning

are best parameters when we adopt a learning machine to process a data set.

#### 5.1.4 Experimental environment

All computations are performed on a node of compute cluster with 32 CPUs (Intel Core Due 3.0GHz) running Red Hat Linux Enterprise 5 with 48GB main memory. The coding environment is MATLAB 2014a.

### 5.2 Experimental results

#### 5.2.1 Classification and time performances

Figures 4 and 5 show the classification performances[12] and corresponding running time of different methods on the used data sets. The ranks of classification performances[13] can be also found in Fig. 9. According to the results, it is found that (1) M$^2$CR brings a better classification performances on most cases and the training time would not add all the time (for example, the results given by the comparison between ours and MVLE); (2) compared with the methods considering online learning and the methods without the consideration about the correlations, online learning, and semi-supervised cases, our method performs best on each data set, while this leads to a longer training time in average.

#### 5.2.2 Convergence

During the procedure of M$^2$CR, we should optimize Eq. (22) and whether the optimization procedure can be converge is discussed here. Refer to [1], the convergence results are given in Fig. 6 and in this figure, the objective function corresponds to Eq. (22). According to results, it is found that when a new labelled instance arrives, our method can converge within 15 iterations in general.

#### 5.2.3 Influence of $\eta$ and $\gamma$

According to the experimental setting, we know that there are many adjustable parameters and the performances of compared learning methods used here will be affected by these parameters. Thus, in the following experiments, we discuss the influence of different parameters and here, we discuss the influence of $\eta$ and $\gamma$ firstly. For the convenience of presentation, we give the changes of the classification performance w.r.t. the change of $\eta$ and $\gamma$. In simple speaking, we choose a value for $\eta$ from $\{0.1, 0.2, \ldots, 0.8, 0.9\}$ and for $\gamma$ from $\{0.05, 0.1, \ldots, 0.95, 1, 2, \ldots, 9, 10\}$. Then, we adjust other parameters and get the best corresponding performances. In the following experiments in Sects. 5.2.4 and 5.2.7, the presentation ways are same.

Then, Fig. 7 shows the influence of $\eta$ and $\gamma$ on the classification performances of M$^2$CR. According to this figure, it is found that (1) when $\eta \geq 0.3$, if the value of $\eta$ is larger, the features we chosen have more stronger correlations with the labels. Meanwhile, the number of those features is fewer. This leads the useful information which can be considered to design the learning method be less and then the performance of a learning method is reduced. (2) The classification performances of M$^2$CR when $\eta = 0.1$, $\eta = 0.2$, and $\eta = 0.3$ are similar. This indicates that the features whose correlations with the labels are fill in the range [0.1, 0.3] have similar effects on improving classification performances. (3) When $\gamma \geq 0.15$, if the value of $\gamma$ is larger, for any instance, its neighbourhood granularity will be composed by more instances whose similarities are much more smaller. This leads to a worse classification performance. (4) The classification performances of M$^2$CR when $\gamma = 0.05$, $\gamma = 0.1$, and $\gamma = 0.15$ are similar. This indicates that choosing instances whose similarities are less than 0.15 to compose a neighbourhood granularity have similar effects on improving classification performances.

Thus, according to the above statements, we can see that setting $\eta = 0.3$ and $\gamma = 0.15$ can get a relative better classification performance.

#### 5.2.4 Influence of the percentage of training and validation instances and labelled instances

Figure 8 shows the influence of the percentage of training and validation instances and labelled instances on the classification performances of M$^2$CR[14]. In addition, since we adopt 10-fold cross-validation for experiments and 90% instances in the training and validation set are used for training, thus the influence of the percentage of training and validation instances is equal to the influence of the percentage of training instances. According to the results, it is found that (1) for most data sets (except for the Corel5k), with the increasing of training instances, the accuracy appears a first up and then down trend. This indicates that when the size of a data set is kept, more training instances maybe lead to over-fitting, while few training instances maybe lead to under-fitting. So selecting a feasible

[12] indeed, we get the classification performances about accuracy, acc$^+$-true positive rate, acc$^-$-true negative rate, PPV-positive predictive value, F-measure, and G-mean [64], but with the limitation of the length for paper, only performances about accuracy are shown. While this would not disturb our conclusions.

[13] the ranks are given on the base of the six classification indexes

[14] with the limitation of the length for paper, we only show the results in terms of the accuracy under the semi-supervised case. Indeed, for other cases, the conclusions are same

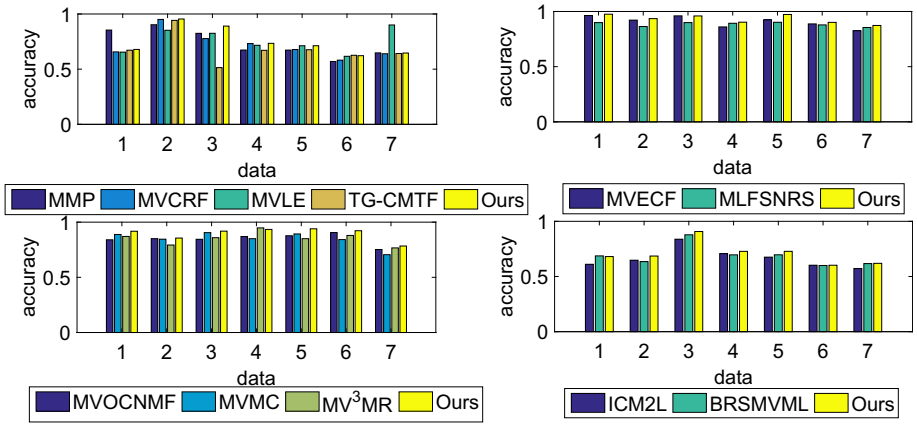


**Fig. 4** Accuracy comparisons for the used methods in different cases

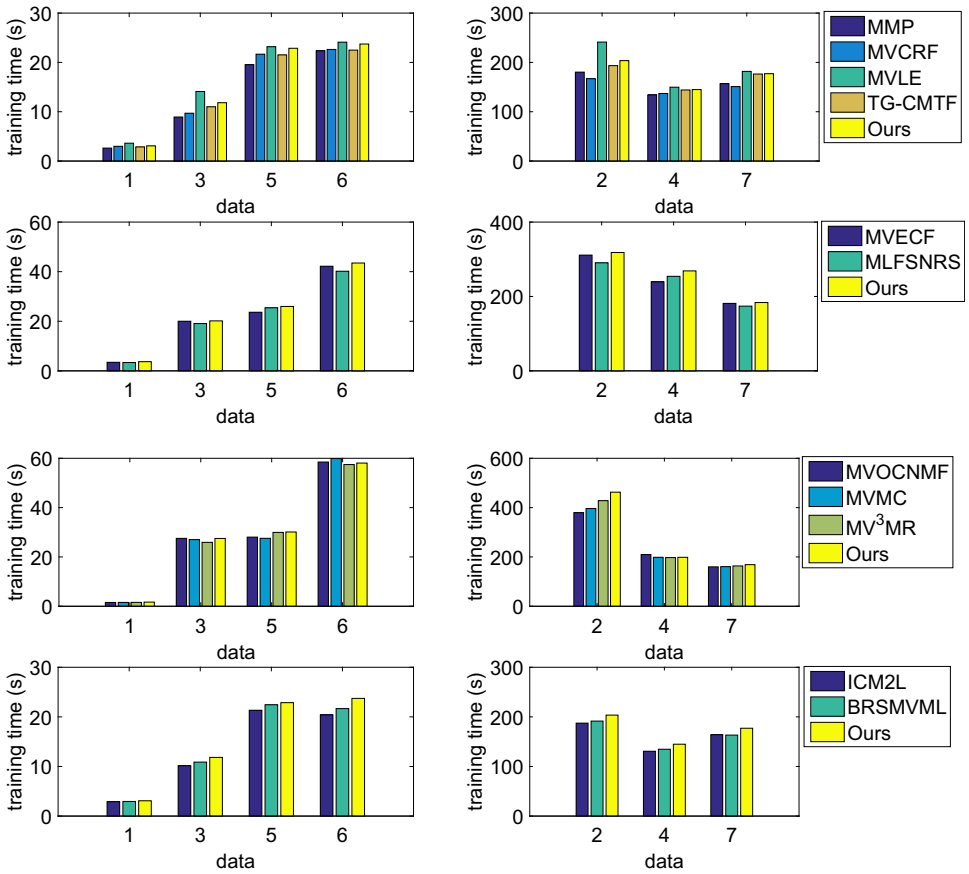


**Fig. 5** Experimental results on data sets about training time with different cases

percentage of training instances is important[15]; (2) for Corel5k, the accuracy rises continuously with the increasing of the percentage of training instances. After analysis, the reason is that the size of Corel5k is not enough to lead to the over-fitting; (3) more labelled instances brings a better accuracy.

[15] indeed, in our experiments expect for this subsection, we show the results with a feasible percentage of training instances set, namely, the percentage accords to best parameters

### 5.2.5 Statistical analysis

In order to check if the differences between $M^2CR$ and other compared methods are significant, we adopt Friedman-Nemenyi statistical test [65] and use Fig. 9 to show the statistical results. In this figure, 'rank' represents the ranks of classification performances in terms of the compared methods on the used data sets and the average rank of the methods is given in the titles of these sub-figures. At the left side of each sub-figure, the corresponding statistical

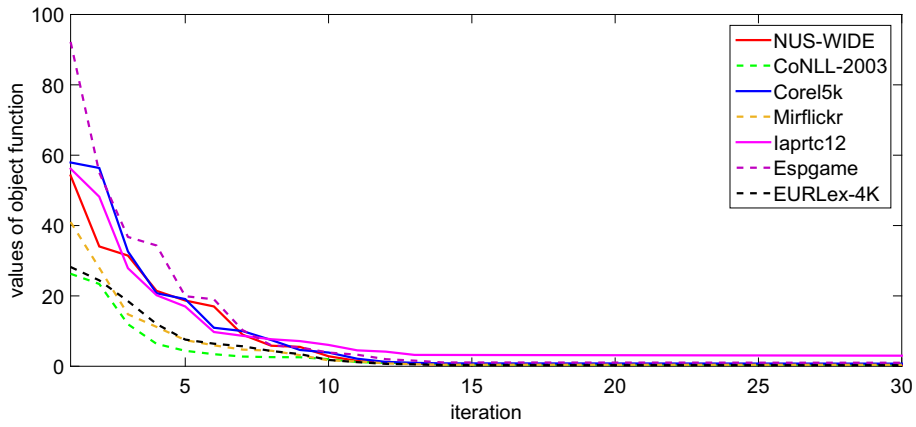


**Fig. 6** Convergence of M$^2$CR on different data sets

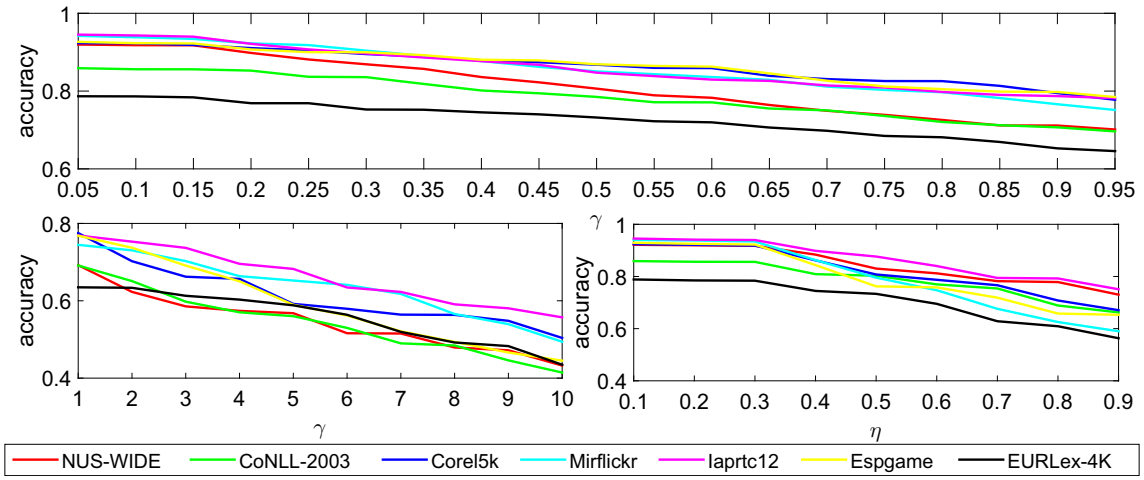


**Fig. 7** Influence of $\eta$ and $\gamma$ on the classification performances of M$^2$CR

results are shown simultaneously. The definitions about these statistical items can be found in [65], and we just analyze the experimental results as below.

(1) In terms of the statistical results about Friedman test, we give the ones of the top left sub-figure firstly. Since we adopt 7 data sets and 5 methods for experiments about the correlations case, thus the Friedman statistic $\chi_F^2 = 11.086$ and $F_F = 3.932$, further, $F_{0.05}(4, 24) = 2.776$ and $F_{0.10}(4, 24) = 2.195$. Since $F_F > F_{0.05}(4, 24)$ and $F_F > F_{0.10}(4, 24)$, so we can reject the null-hypothesis and say the differences between all compared methods on multiple data sets are significant. For other cases, we draw a same conclusion.

(2) Then we carry out Nemenyi test for pairwise comparisons and show the statistical results. (2.1) For the top left sub-figure, namely, correlations case, the corresponding critical differences (CD) are $CD_{0.05} = 2.306$ and $CD_{0.10} = 2.078$. Since under the case of $CD_{0.10}$, only rank difference between M$^2$CR and MVLE is smaller than $CD_{0.10}$, so we can say on this case, the performance of M$^2$CR is significantly better than MMP, MVCRF, and TG-CMTF, but not significantly better than MVLE. Under the case of $CD_{0.05}$, since only the rank difference between M$^2$CR and TG-CMTF is larger than $CD_{0.05}$, thus we say that on this case, the performance of M$^2$CR is significantly better than TG-CMTF, but not significantly better than MMP, MVCRF, and MVLE. (2.2) For the top right sub-figure, namely, online learning case, since the rank differences between M$^2$CR and MVECF (MLFSNRS) are larger than both $CD_{0.10}$ and $CD_{0.05}$, thus the performance of M$^2$CR is significantly better MVECF (MLFSNRS). (2.3) For the bottom left sub-figure, namely, semi-supervised case, the average rank of ours is 1.143 and the corresponding CDs are $CD_{0.05} = 1.581$ and $CD_{0.10} = 1.773$. Since $1.143 + 1.773 = 2.916$ and $1.143 + 1.581 = 2.724$, thus under the case of $CD_{0.05}$, the performance of M$^2$CR is significantly better than MVOCNMF and MV$^2$MR, but not significantly better than MVMC, while under the case of $CD_{0.10}$, the performance of M$^2$CR is significantly better than all compared semi-supervised methods. (2.4) For the

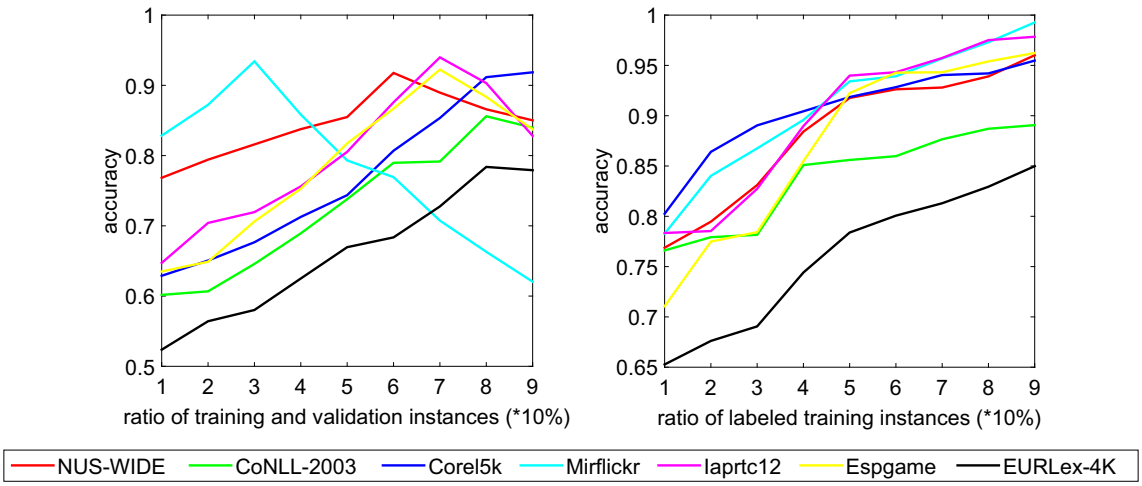


**Fig. 8** Influence of the percentage of training and validation instances and labelled instances on the classification performances of M$^2$CR

bottom right sub-figure, namely, without the consideration about the correlations, online learning, and semi-supervised cases, similar with the online learning case, because the rank differences between M$^2$CR and ICM2L (BRSMVML) are larger than both $CD_{0.10}$ and $CD_{0.05}$, thus the performance of M$^2$CR is significantly better ICM2L (BRSMVML).

In generally, we can validate the effectiveness of M$^2$CR from an average view. Specially, for online learning case and without the consideration about the correlations, online learning, and semi-supervised cases, our M$^2$CR performs significant best in statistics.

#### 5.2.6 Online experiments

Since M$^2$CR performs significant best in statistics for the online learning case, thus we want to show the online experiments through figures.

For convenience, we only select NUS-WIDE for description and in this data set, 3 labels ('bear', 'road', 'fish') and 17 instances per label are selected in random. 13 out of 17 instances per label are used as the original stored labelled training instances (see Fig. 10), and the other 2 out of the 17 instances are used in case 1 (see red boxes in the top sub-figure of Fig. 11), and the rest is treated as the unlabelled ones and used in case 2 (see blue boxes in the bottom sub-figure of Fig. 11). Then, the optimistic POS (NEG, BND), the pessimistic POS (NEG, BND) of the original stored ones, ones after case 1, and ones after case 2 are shown in the these figures. According to these figures, it is found that (1) with the new labelled instances arrive, the optimistic POS covers more instances and the pessimistic POS would not cover more instances. Then for the optimistic NEG and pessimistic NEG, they cover fewer instances. This indicates that new labelled instances arriving makes the positive regions be more clearly; (2) with the new unlabelled instances arrive, although for label 'road', the optimistic POS covers fewer instances, the cover areas of optimistic NEG and pessimistic NEG are further reduced. This indicates that new unlabelled instances arriving can make the negative regions be more smaller. Combining the conclusion (1) and conclusion (2), we argue that with new instances arrive, we can label the instances accurately with a higher probability.

#### 5.2.7 Incomplete experiments

We give the changes of the classification performance w.r.t. the change of amount of missing information. For convenience, we only show the changes in terms of the accuracy, NUS-WIDE and without the consideration about the correlations, online learning, and semi-supervised cases. Indeed, if we consider other cases, we can still draw the similar conclusions. Then Table 2 shows the results and 0% represents no information of features or labels is lost. According to this table, it is found that even though we lose more information initially, since we make full use of the threefold correlations of features and labels, recover the data, and update correlations continuously once a new instance arrives, thus the accuracy of M$^2$CR does not reduce too much.

#### 5.2.8 Ablation study

As we said before, M$^2$CR can process semi-supervised real-time generation multi-view multi-label data sets with incomplete information and it consists of two main parts. One is recovering missing information with threefold correlations (TC), and the other is processing new arriving instances with dynamic updating multi-region (DUMR),

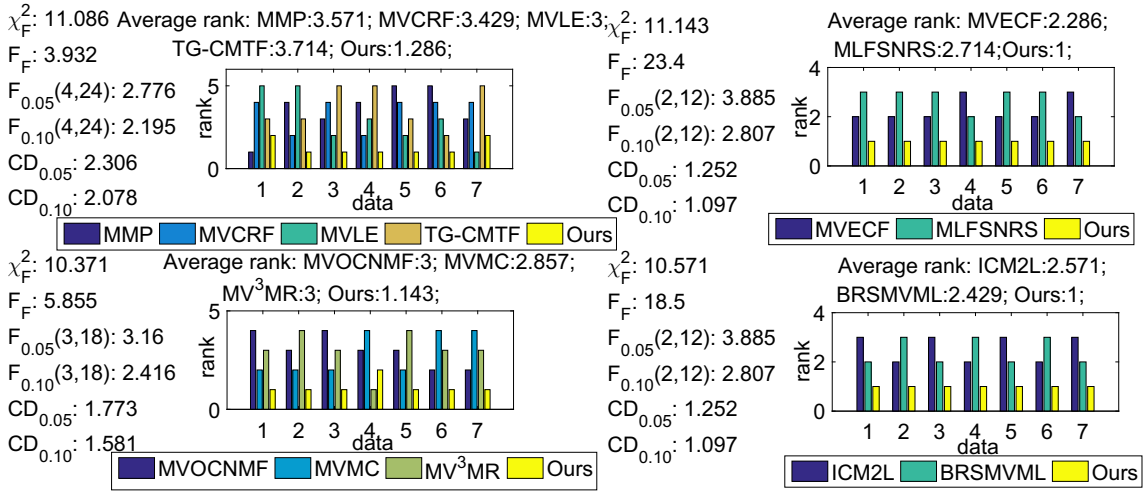


**Fig. 9** Friedman-Nemenyi statistical test for M$^2$CR on different data sets

recovering methods, and correlations among information. Among these two parts, TC and DUMR possess important positions and we want to discuss their effects on improving the classification accuracies.

In order to discuss the effects, we adopt ablation study. In simple speaking, among the framework of M$^2$CR (see Sect. 4 and Fig. 3), the operations related with TC or DUMR can be removed or modified. For example, in order to measure the effects of TC, we can remove step Sect. 4.2 and use other methods rather than TC to recover missing information. Similar, in order to measure the effects of DUMR, we can remove step Sect. 4.3 and use other methods rather than DUMR to process new arriving instances, namely, label new unlabelled arriving instances. Thus, the procedure of ablation study includes the below three parts.

For the first part, we assess that to what extent TC helps to improve the classification accuracies. So for the framework of M$^2$CR, we remove step Sect. 4.2 and use some recovering methods to replace TC to recover missing information. The alternative classical recovering methods are AWVRF[16] [32], McWL[17] [2], PC-GAIN[18] [34], and we define the new modified M$^2$CRs as AR, McR, PR, respectively.

For the second part, we assess the effects of DUMR and then we remove step Sect. 4.3 in the framework of M$^2$CR and use some classical methods including MSCD[19] [41], OSAM[20] [42], SSOPMV[21] [18] to replace DUMR to label new unlabelled arriving instances. The new modified M$^2$CRs are defined as CM, CO, CS, respectively.

For the third part, we want to see if we don't adopt both TC and DUMR, how about the classification accuracies change. So we combine the previous operations about two parts and define the new modified M$^2$CRs as $XY$ where $X \in \{A, Mc, P\}$ and $Y \in \{M, O, S\}$.

Now we show the classification performances of the above new methods on the used data sets in Fig. 12. These data sets are semi-supervised real-time generation ones, and information of some instances is incomplete. According to the experimental results, it is found that (1) compared with methods without the usage of DUMR and TC, methods with DUMR only and methods with TC only can both improve the classification performances; (2) ours M$^2$CR performs best here which indicates that combining TC with DUMR can bring better accuracies; (3) compared with TC, DUMR has a greater effect on improving the classification accuracies since the classification accuracies of AR, McR, PR are better than ones of CM, CO, CS in average.

## 6 Conclusions and future work

### 6.1 Purpose and proposal

In order to solve the semi-supervised real-time generation multi-view multi-label data with incomplete information which are ubiquitous in real applications, we develop a

[16] adjusted weight voting random forest

[17] matrix completion for multi-view weak label learning

[18] pseudo-label conditional generative adversarial imputation networks for incomplete data

[19] multi-view semi-supervised learning for classification on dynamic networks

[20] online semi-supervised active learning framework

[21] semi-supervised one-pass multi-view learning

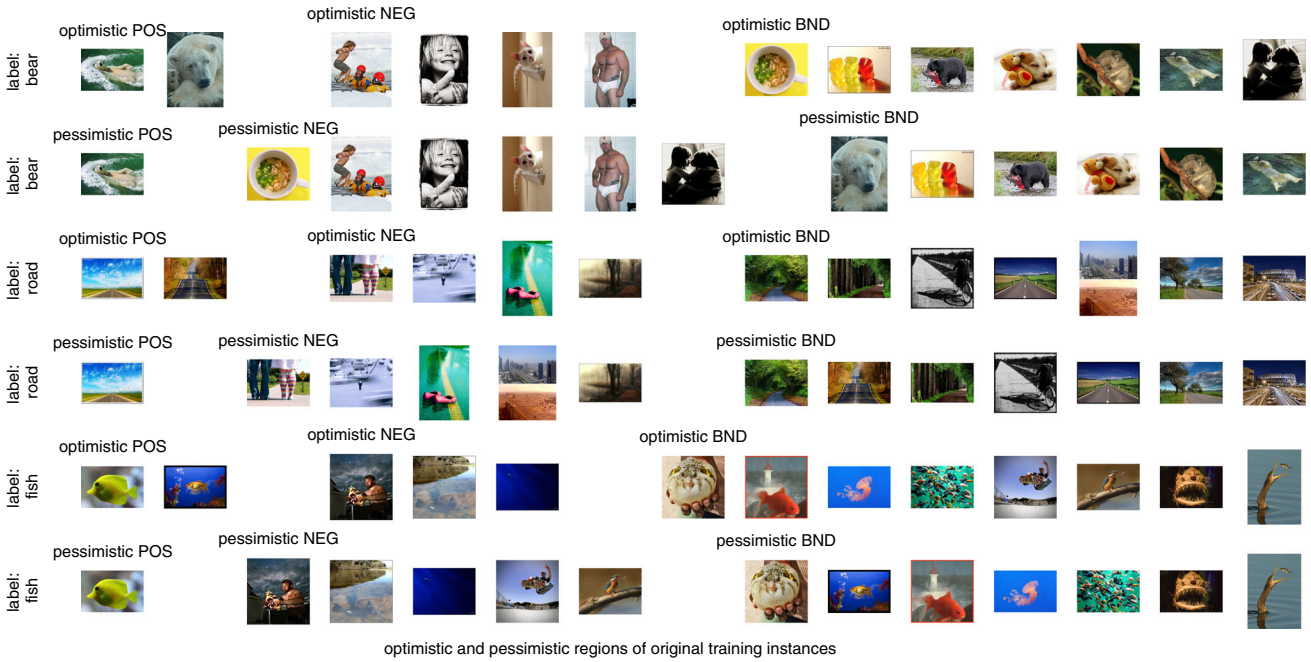


**Fig. 10** Original stored labelled training instances (three labels and thirteen instances per label selected) and the corresponding positive, boundary, and negative regions of them

multi-view multi-label-based online method with threefold correlations and dynamic updating multi-region ($M^2CR$). $M^2CR$ try to recover missing information with threefold correlations and predict labels of unlabelled instances with updating multi-region dynamically.

## 6.2 Experimental conclusions

Experimental results on 7 multi-view multi-label data sets have demonstrated that (1) $M^2CR$ brings a better average classification performance; (2) $M^2CR$ can converge within 15 iterations in general; (3) statistical results of $M^2CR$ are significant compared with the methods considering online learning or methods without the consideration about the correlations, online learning, and semi-supervised cases; (4) $M^2CR$ performs significantly better than other compared methods average; (5) under most cases, with the increase of the number of training instances, the accuracy of $M^2CR$ appears a first up and then down trend; (6) for $M^2CR$, more labelled instances bring a better accuracy; (7) $M^2CR$ is good for online learning and with new instances arriving, it can label the instances accurately with a higher probability; (8) even though we lose more information about features and labels initially, $M^2CR$ still retains a relative high accuracy; (9) compared with threefold correlations, dynamic updating multi-region has a greater effect on improving the classification accuracies.

## 6.3 Important advantages

Besides the ability to process multi-view multi-label data sets, $M^2CR$ still possesses another two important advantages. (1) Compared with the tradition solutions to incomplete form, $M^2CR$ recovers the missing information with the usage of threefold valuable correlations among information and this operation makes the recovered information be more authentic. Then, such an operation can reduce the computational complexity and bring a simpler framework; (2) compared with the tradition solutions to real-time generation and semi-supervised forms, $M^2CR$ combines dynamic updating multi-region with recovering missing information and updating correlations among information to process new arriving instances, so that it processes semi-supervised real-time generation with incomplete information better and labels an new arriving unlabelled instance with a higher accuracy.

## 6.4 Future work

Two open issues can be considered in the future work. One is what can we do when we have no enough space to store the training instances and the whole model, the other is what will happen if the correlations between features and labels are hard to calculate and $M^2CR$ may be not suitable.

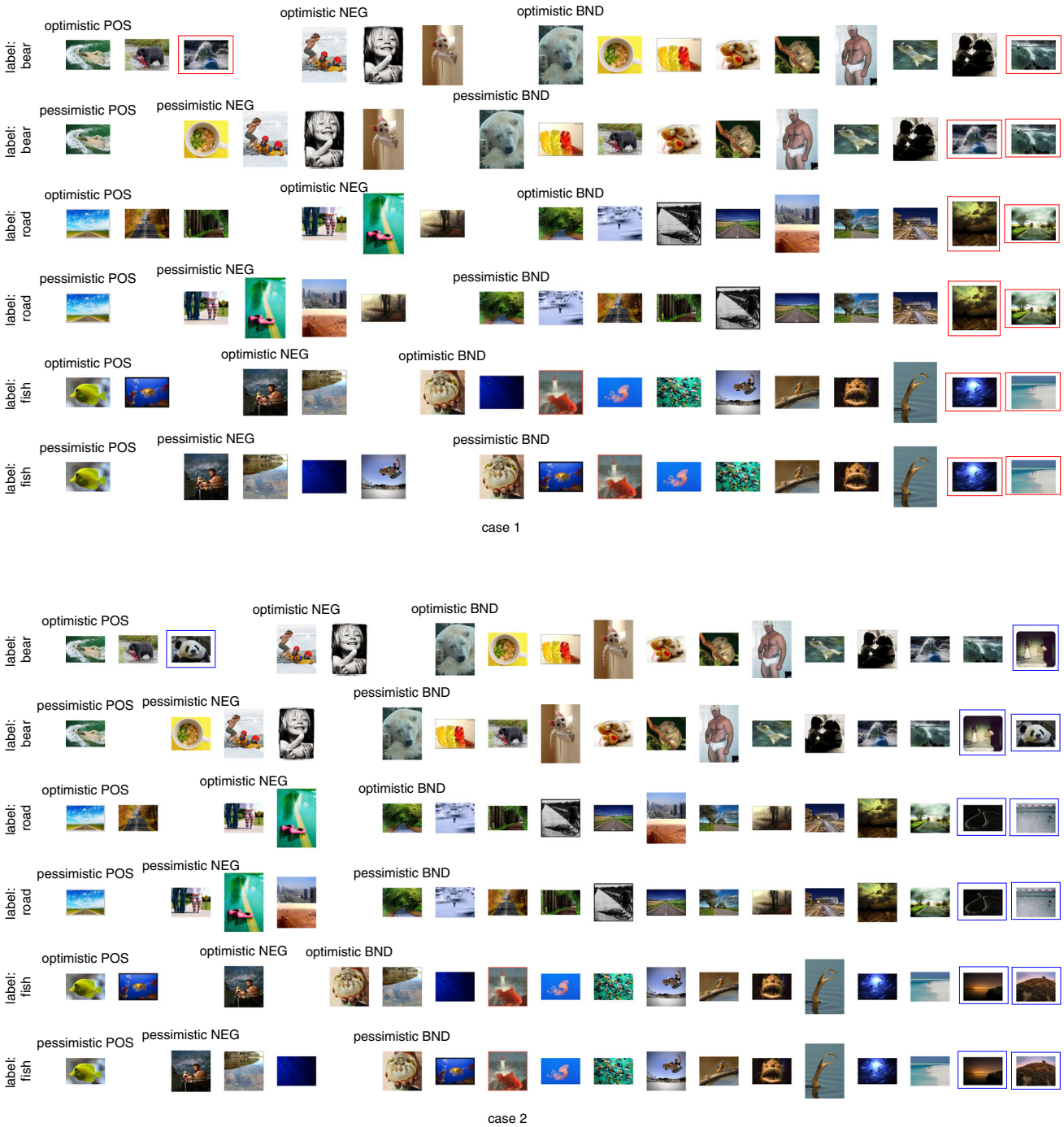


**Fig. 11** Case 1 (top): positive, boundary, and negative regions when some labelled training instances (marked by red boxes) arrive; Case 2 (bottom): positive, boundary, and negative regions when some unlabelled training instances (marked by blue boxes) arrive

**Table 2** Accuracy changes of $M^2CR$ on NUS-WIDE and without the consideration about the correlations, online learning, and semi-supervised cases. In this table, $x\%$ represents the percentage of missing features or labels

| NUS-WIDE | | features | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 0% | 10% | 20% | 30% | 40% | 50% | 60% | 70% | 80% | 90% |
| labels | 0% | 0.967 | 0.960 | 0.950 | 0.949 | 0.945 | 0.939 | 0.938 | 0.937 | 0.933 | 0.927 |
| | 10% | 0.965 | 0.959 | 0.949 | 0.942 | 0.934 | 0.932 | 0.929 | 0.926 | 0.921 | 0.911 |
| | 20% | 0.955 | 0.948 | 0.941 | 0.940 | 0.931 | 0.921 | 0.912 | 0.911 | 0.901 | 0.893 |
| | 30% | 0.954 | 0.943 | 0.936 | 0.928 | 0.918 | 0.911 | 0.909 | 0.901 | 0.899 | 0.886 |
| | 40% | 0.953 | 0.940 | 0.929 | 0.922 | 0.908 | 0.901 | 0.901 | 0.898 | 0.898 | 0.881 |
| | 50% | 0.943 | 0.936 | 0.919 | 0.911 | 0.898 | 0.893 | 0.884 | 0.877 | 0.875 | 0.875 |
| | 60% | 0.939 | 0.927 | 0.911 | 0.907 | 0.893 | 0.889 | 0.882 | 0.871 | 0.869 | 0.864 |
| | 70% | 0.935 | 0.919 | 0.901 | 0.897 | 0.886 | 0.886 | 0.879 | 0.866 | 0.857 | 0.854 |
| | 80% | 0.931 | 0.918 | 0.893 | 0.893 | 0.886 | 0.881 | 0.870 | 0.861 | 0.848 | 0.844 |
| | 90% | 0.923 | 0.918 | 0.886 | 0.877 | 0.870 | 0.865 | 0.858 | 0.852 | 0.847 | 0.843 |

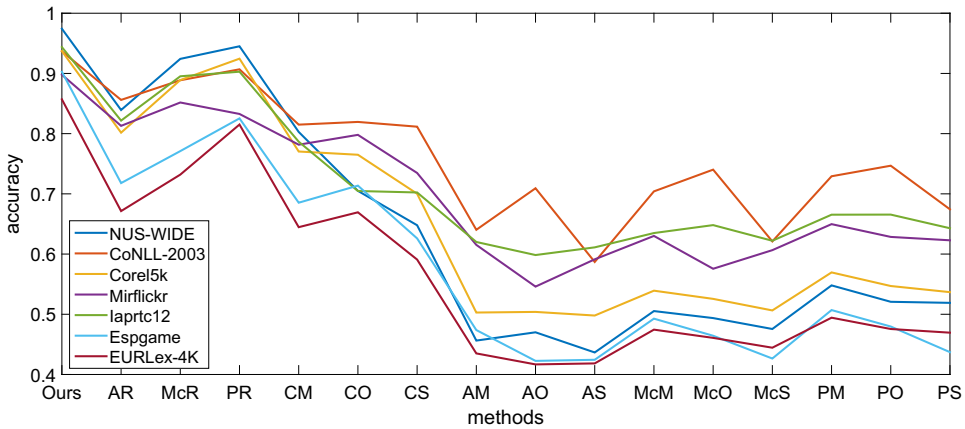


**Fig. 12** Ablation study for $M^2CR$

**Acknowledgements** This work is supported by 'Chenguang Program' supported by Shanghai Education Development Foundation and Shanghai Municipal Education Commission under Grant Number 18CG54. Furthermore, this work is also sponsored by National Natural Science Foundation of China (CN) under Grant Number 61602296. The authors would like to acknowledge their supports.

## Declarations

**Conflict of interest** We declare that we have no financial and personal relationships with other people or organizations that can inappropriately influence our work; there is no professional or other personal interest of any nature or kind in any product, service, and/or company that could be construed as influencing the position presented in, or the review of, the manuscript entitled, 'Multi-view multi-label-based Online Method with Threefold Correlations and Dynamic Updating Multi-Region'.

## References

1. Zhu Y, Kwok JT, Zhou ZH (2018) Multi-label learning with global and local label correlation. IEEE Trans Knowl Data Eng 30(6):1081–1094
2. Tan QY, Yu GX, Domeniconi C, Wang J, Zhang ZL (2018) Multi-view weak-label learning based on matrix completion. In: Proceedings of the 2018 SIAM international conference on data mining (SIAM 2018), pp 450–458
3. Zong LL, Miao FQ, Zhang XC, Liu XY, Yu H (2021) Incomplete multi-view clustering with partially mapped instances and clusters. Knowl-Based Syst 212:106615
4. Hu XC, Pedrycz W, Wu KY, Shen YH (2021) Information granule-based classifier: a development of granular imputation of missing data. Knowl-Based Syst 214:106737
5. Sun L, Wang LY, Ding WP, Qian YH, Xu JC (2020) Neighborhood multi-granulation rough sets-based attribute reduction using Lebesgue and entropy measures in incomplete neighborhood decision systems. Knowl-Based Syst 192:105373
6. Park LAF, Bezdek JC, Leckie C, Kotagiri R, Bailey J, Palaniswami M (2016) Visual assessment of clustering tendency for incomplete data. IEEE Trans Knowl Data Eng 28(12):3409–3422
7. Zhang XS, Zhuang Y, Wang W, Pedrycz W (2018) Online feature transformation learning for cross-domain object category recognition. IEEE Trans Neural Netw Learn Syst 29(7):2857–2871
8. Yan YG, Wu QY, Tan MK, Ng MK, Min HQ, Tsang IW (2018) Online heterogeneous transfer by hedge ensemble of offline and online decisions. IEEE Trans Neural Netw Learn Syst 29(7):3252–3263
9. Li GX, Shen YL, Zhao PL, Lu X, Liu J, Liu YY, Hoi SCH (2019) Detecting cyberattacks in industrial control systems using online learning algorithms. Neurocomputing 364:338–348
10. Qian BY, Wang X, Ye JP, Davidson I (2015) A reconstruction error based framework for multi-label and multi-view learning. IEEE Trans Knowl Data Eng 27(3):594–607

11. Nie FP, Tian L, Wang R, Li XL (2020) Multiview semi-supervised learning model for image classification. IEEE Trans Knowl Data Eng 32(12):2389–2400
12. Bai L, Liang JY, Cao FY (2021) Semi-supervised clustering with constraints of different types from multiple information sources. IEEE Trans Pattern Anal Mach Intell 43(9):3247–3258
13. Jia XD, Jing XY, Zhu XK, Chen SC, Du B, Cai ZY, De ZY, Yue D (2021) Semi-supervised multi-view deep discriminant representation learning. IEEE Trans Pattern Anal Mach Intell 43(7):2496–2509
14. Gao C, Zhou J, Miao DQ, Wen JJ, Yue XD (2021) Three-way decision with co-training for partially labeled data. Inf Sci 544:500–518
15. He L, Zhang H (2018) Kernel k-means sampling for Nyström approximation. IEEE Trans Image Process 27(5):2108–2120
16. Li M, Bi W, Kwok JT, Lu BL (2015) Large-scale Nyström kernel matrix approximation using randomized SVD. IEEE Trans Neural Netw Learn Syst 26(1):152–164
17. Zhong G, Pun CM (2020) Revisiting Nyström extension for hypergraph clustering. Neurocomputing 403:247–256
18. Zhu CM, Wang Z, Zhou RG, Wei L, Zhang XF, Ding Y (2019) Semi-supervised one-pass multi-view learning. Neural Comput Appl 31:8117–8134
19. Davenport MA, Romberg J (2016) An overview of low-rank matrix recovery from incomplete observations. IEEE J Sel Top Signal Process 10(4):608–622
20. Peng SJ, He GF, Liu X, Wang HZ (2015) Hierarchical block-based incomplete human mocap data recovery using adaptive nonnegative matrix factorization. Comput Gr 49:10–23
21. Zhang L, Zhao Y, Zhu ZF, Shen DG, Ji SW (2018) Multi-view missing data completion. IEEE Trans Knowl Data Eng 30(7):1296–1309
22. Niu GL, Yang YL, Sun LQ (2021) One-step multi-view subspace clustering with incomplete views. Neurocomputing 438:290–301
23. Lin WC, Tsai CF (2020) Missing value imputation: a review and analysis of the literature [2006-2017]. Artif Intell Rev 53:1487–1509
24. Garciarena U, Santana R (2017) An extensive analysis of the interaction between missing data types, imputation methods, and supervised classifiers. Expert Syst Appl 89:52–65
25. Shao J, Meng W, Sun GD (2017) Evaluation of missing value imputation methods for wireless soil datasets. Pers Ubiquit Comput 21(1):113–123
26. Pati SK, Das AK (2017) Missing value estimation for microarray data through cluster analysis. Knowl Inf Syst 52(3):709–750
27. Oh S, Kang DD, Brock GN, Tseng GC (2011) Biological impact of missing-value imputation on downstream analyses of gene expression profiles. Bioinformatics 27(1):78–86
28. Mesquite DPP, Gomes JPP, Junior AHS, Nobre JS (2017) Euclidean distance estimation in incomplete datasets. Neurocomputing 248:11–18
29. Zhang L, Bing ZH, Zhang LY (2015) A hybrid clustering algorithm based on missing attribute interval estimation for incomplete data. Pattern Anal Appl 18:377–384
30. Purwar A, Singh SK (2015) Hybrid prediction model with missing value imputation for medical data. Expert Syst Appl 42(13):5621–5631
31. Huang JL, Keung JW, Sarro F, Li YF, Yu YT, Chan WK, Sun HY (2017) Cross-validation based K nearest neighbor imputation for software quality datasets: an empirical study. J Syst Softw 132:226–252
32. Xia J, Zhang SY, Cai GL, Li L, Pan Q, Yan J, Ning GM (2017) Adjusted weight voting algorithm for random forests in handling missing values. Pattern Recogn 69:52–60
33. Yoon JS, Jordon J, Schaar MVD (2018) GAIN: missing data imputation using generative adversarial nets. In: Proceedings of the 35th international conference on machine learning (ICML 2018), vol 80, pp 5689–5698
34. Wang YF, Li D, Li X, Yang M (2021) PC-GAIN: Pseudo-label conditional generative adversarial imputation networks for incomplete data. Neural Netw 141:395–403
35. Zhang Y, Zhou BH, Cai XR, Guo WY, Ding XK, Yuan XJ (2021) Missing value imputation in multivariate time series with end-to-end generative adversarial networks. Inf Sci 551:67–82
36. Sun LJ, Ye P, Lyu GY, Feng SH, Dai GJ, Zhang H (2020) Weakly-supervised multi-label learning with noisy features and incomplete labels. Neurocomputing 413:61–71
37. Jiang L, Yu GX, Guo MZ, Wang J (2020) Feature selection with missing labels based on label compression and local feature correlation. Neurocomputing 395:95–106
38. Li LC, Liu HL, Zhou HJ, Zhang CD (2020) Missing data estimation method for time series data in structure health monitoring systems by probability principal component analysis. Adv Eng Softw 149:102901
39. Baisa NL (2021) Robust online multi-target visual tracking using a HISP filter with discriminative deep appearance learning. J Vis Commun Image Represent 77:102952
40. Li Z, Xing YY, Huang JM, Wang HB, Gao JL, Yu GX (2021) Large-scale online multi-view graph neural network and applications. Futur Gener Comput Syst 116:145–155
41. Chen C, Li YZ, Qian H, Zheng ZB, Hu YQ (2020) Multi-view semi-supervised learning for classification on dynamic networks. Knowl-Based Syst 195:105698
42. Nie XL, Fan MY, Huang XY, Yang WJ, Zhang B, Ma XS (2020) Online semisupervised active classification for multiview PolSAR data. IEEE Trans Cybern. https://doi.org/10.1109/TCYB.2020.3026741
43. Zhu CM (2016) Improved multi-kernel classification machine with Nyström approximation technique and universum data. Neurocomputing 175:610–634
44. Chua TS, 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: Proceedings of the ACM international conference on image and video retrieval, pp 48
45. He ZY, Chen C, Bu JJ, Li P, Cai D (2015) Multi-view based multi-label propagation for image annotation. Neurocomputing 168:853–860
46. Sun XL, Sun SL, Yin MZ, Yang H (2020) Hybrid neural conditional random fields for multi-view sequence labeling. Knowl-Based Syst 189:105151
47. Duygulu P, Barnard K, Freitas JFGd, Forsyth DA (2002) Object recognition as machine translation: learning a lexicon for a fixed image vocabulary. In: Proceedings of the 7th European conference on computer vision-part IV (ECCV 2002), pp 97–112
48. Huiskes MJ, Lew MS (2008) The mir flickr retrieval evaluation. In: Proceedings of the 1st ACM international conference on multimedia information retrieval (MIR 2008), pp 39–43
49. Makadia A, Pavlovic V, Kumar S (2008) A new baseline for image annotation. In: Proceedings of the 10th European conference on computer vision: part III (ECCV 2008), pp 316–329
50. Ahn LV, Dabbish L (2004) Labeling images with a computer game. In: Proceedings of the SIGCHI conference on human factors in computing systems (CHI 2004), pp 319–326
51. LozaMencía E, Fürnkranz J (2010) Efficient multilabel classification algorithms for large-scale problems in the legal domain. In: S Montemagni, W Peters, D Tiscornia (Eds.) Semantic processing of legal texts: where the language of law meets the law of language, Springer, Berlin, pp 192–215
52. He ZY, Chen C, Bu JJ, Li P, Cai D (2015) Multi-view based multi-label propagation for image annotation. Neurocomputing 168:853–860

53. Zhu PF, Hu Q, Hu QH, Zhang CQ, Feng ZZ (2018) Multi-view label embedding. Pattern Recogn 84:126–135
54. Zhang JY, Rao Y, Zhang JL, Zhao YQ (2019) Trigraph regularized collective matrix tri-factorization framework on multi-view features for multilabel image annotation. IEEE Access 7:161805–161821
55. Liu Y, Jiang CQ, Zhao HM (2018) Using contextual features and multi-view ensemble learning in product defect identification from online discussion forums. Decis Support Syst 105:1–12
56. Liu JH, Lin YY, Li YW, Weng W, Wu SX (2018) Online multi-label streaming feature selection based on neighborhood rough set. Pattern Recogn 84:273–287
57. Zhang HY, Pedrycz W, Miao DQ, Wei ZH (2014) From principal curves to granular principal curves. IEEE Trans Cybern 44(6):748–760
58. Lai ZH, Mo DM, Wong WK, Xu Y, Miao DQ, Zhang D (2018) Robust discriminant regression for feature extraction. IEEE Trans Cybern 48:2472–2484
59. Cai H, Liu B, Xiao YS, Lin LY (2020) Semi-supervised multi-view clustering based on orthonormality-constrained nonnegative matrix factorization. Inf Sci 536:171–184
60. Luo Y, Liu TL, Tao DC, Xu C (2015) Multiview matrix completion for multilabel image classification. IEEE Trans Image Process 24(8):2355–2368
61. Luo Y, Tao DC, Xu C, Xu C, Liu H, Wen YG (2013) Multiview vector-valued manifold regularization for multilabel image classification. IEEE Trans Neural Netw Learn Syst 24(5):709–722
62. Tan QY, Yu GX, Wang J, Domeniconi C, Zhang XL (2021) Individuality- and commonality-based multiview multilabel learning. IEEE Trans Cybern 51(3):1716–1727
63. Zhu XF, Li XL, Zhang SC (2016) Block-row sparse multi-view multi-label learning for image classification. IEEE Trans Cybern 46(2):450–461
64. Zhu CM, Wang Z (2017) Entropy-based matrix learning machine for imbalanced data sets. Pattern Recogn Lett 88:72–80
65. Demsar J (2006) Statistical comparisons of classifiers over multiple data sets. J Mach Learn Res 7(1):1–30