Soft Computing (2020) 24:10657–10679
https://doi.org/10.1007/s00500-019-04572-5

METHODOLOGIES AND APPLICATION

Weight-and-Universum-based semi-supervised multi-view learning machine

Changming Zhu[1,2] · Duoqian Miao[2] · Rigui Zhou[1] · Lai Wei[1]

Published online: 3 December 2019

Abstract

Semi-supervised multi-view learning machine is developed to process the corresponding semi-supervised multi-view data sets which consist of labeled and unlabeled instances. But in real-world applications, for a multi-view data set, only few instances are labeled with the limitation of manpower and cost. As a result, few prior knowledge which is necessary for the designing of a learning machine is provided. Moreover, in practice, different views and features play diverse discriminant roles while traditional learning machines treat these roles equally and assign the same weight just for convenience. In order to solve these problems, we introduce Universum learning to obtain more prior knowledge and assign different weights for views and features to reflect their diverse discriminant roles. The proposed learning machine is named as weight-and-Universum-based semi-supervised multi-view learning machine (WUSM). In WUSM, we first obtain weights of views and features. Then, we construct Universum set to obtain more prior knowledge on the basis of these weights. Different from traditional construction ways, the used construction way makes full use of the information of all labeled and unlabeled instances rather than only a pair of positive and negative training instances. Finally, we design the machine with the usage of the Universum set along with original data set. Our contributions are given as follows. (1) With the usage of all (labeled, unlabeled) instances of the data set, the Universum set provides more useful prior knowledge. (2) WUSM considers the diversities of views and features. (3) WUSM advances the development of semi-supervised multi-view learning machines. Experiments on bipartite ranking, feature selection, dimensionality reduction, classification, clustering, etc. validate the advantages of WUSM and draw a conclusion that with the introduction of Universum learning, view weights, and feature weights, the performance of a semi-supervised multi-view learning machine is boosted.

Keywords Semi-supervised learning · Multi-view learning · View weights · Feature weights · Universum learning

1 Introduction

Semi-supervised multi-view data set consists of labeled instances and unlabeled instances. Each instance can be represented by multiple distinct feature sets (views). The solutions to process this kind of data sets is called semi-supervised multi-view learning machines. For such a data set, compared with the unlabeled instances, the labeled ones can provide more useful prior knowledge which is necessary for the designing of a learning machine. While in real-world applications, data sets are generated and collected very quickly. With the limitation of manpower and cost, we cannot label all instances and in practice, only few instances are labeled. As a result, many multi-view data sets only provide few prior knowledge. Moreover, different views (text, video, image, etc.) and features (text size, text color, text thickness, etc.) play diverse discriminant roles. For exam-

Communicated by V. Loia.

✉ Duoqian Miao
dqmiao@tongji.edu.cn

Changming Zhu
cmzhu@shmtu.edu.cn

Rigui Zhou
rgzhou@shmtu.edu.cn

Lai Wei
weilai@shmtu.edu.cn

1 College of Information Engineering, Shanghai Maritime University, Shanghai 201306, People's Republic of China

2 Department of Computer Science and Technology, Tongji University, Shanghai 201804, People's Republic of China

ple, text and video give people different understanding of data sets and text size and text color bring different sensory stimulations. While for the sake of convenience, traditional learning machines treat these roles equally and assign the same weight. So few prior knowledge and equivalent discriminant roles of views and features limit the improvement in the performances of learning machines.

Thus in this work, we pay more attention to the following two research questions. One is how to increase useful prior knowledge, the other is how to reflect diverse discriminant roles of views and features.

The targets of our research are increasing more useful prior knowledge, considering the differences of views and features, and finally enhancing the performances of semi-supervised multi-view learning machines.

In order to state the research questions and targets clearly, the rest parts of this section are organized as follows.

1.1 Why is the topic important

Semi-supervised multi-view data set consists of labeled (class label is known beforehand) and unlabeled (class label is unknown beforehand) instances. Each instance consists of multiple views and each view represents information of the instance in a certain area. Take a web page data set $\mathbf{X}$ for example (see Fig. 1). $\mathbf{X}$ is a semi-supervised multi-view data set and it consists of three instances. The first two have been labeled and the third one has not been labeled. Each instance x_i ($i \in \{1, 2, 3\}$) is a web page from the Internet and possesses three views, text (x_i^1), image (x_i^2), and video (x_i^3). Let x_i^v be v-th view of i-th instance and $\mathbf{X}^v = \{x_i^v\}_{i=1}^3$ represents v-th view, then $\mathbf{X} = \{\mathbf{X}^v\}_{v=1}^3$ (Xu et al. 2016).

With the coming of big-data age and development of modernization of industry, semi-supervised multi-view data sets are common in the various walks of life including multi-view clustering (Tzortzis and Likas 2012), handwritten digit recognition (Sun and Zhang 2011), human gait recognition (Deng et al. 2016), image recognition (Wu et al. 2016; Zhu et al. 2016; Wang et al. 2014). For example, in port (Port of Shanghai, Port of Rotterdam, etc.), there are many containers in and out of the port every day. The multiple camera sets in the port take a lot of pictures of containers (pictures from a camera form a view of the data set) and the staff collect and label these pictures so as to design an identification tracking system. This system aims to track the containers and promises the safety and accurate loading and unloading of these containers. While with the limitation of manpower and cost, the staff can collect many pictures but hard to label all of them. In this event, the collected data set is a semi-supervised multi-view one and the performance of the system will be degraded. As a result, the working efficiency of the port will slow down and the national trade will be disturbed further. Thus, the effective processing of semi-supervised multi-view data sets is very important in many fields including logistics industry and even national trade.

In order to process such a kind of data sets, semi-supervised multi-view learning machines (Sheikhpour et al. 2017) including multi-view semi-supervised classification via adaptive regression (MVAR, developed by Tao et al. (2017)), co-labeling (developed by Xu et al. (2016)), sparse Markov chain-based semi-supervised multi-instance multi-label method (sparse Markov, developed by Han et al. (2015)), and semi-supervised multi-view hash model (SSMVH, developed by Zhang and Zheng (2017)) are proposed and popularly used.

Although related experiments validate the effectiveness of these machines, there are two problems that should be taken into consideration. The first one is that if the manpower and cost are very limited, the number of labeled instances will be far less than that of the unlabeled ones, and then the above-mentioned machines cannot process that well. The second one is that for a multi-view data set, different views and features play diverse discriminant roles. While the mentioned machines treat them equally by assigning the same weight and neglect the differences of views and features. There is no doubt, these two problems will disturb the processing of real-world applications and degrade the performances of corresponding learning machines. Furthermore, the worse performances of learning machines will delay the development of national economy and industry. Thus, researching and solving these problems is an important topic.

1.2 What are the research questions

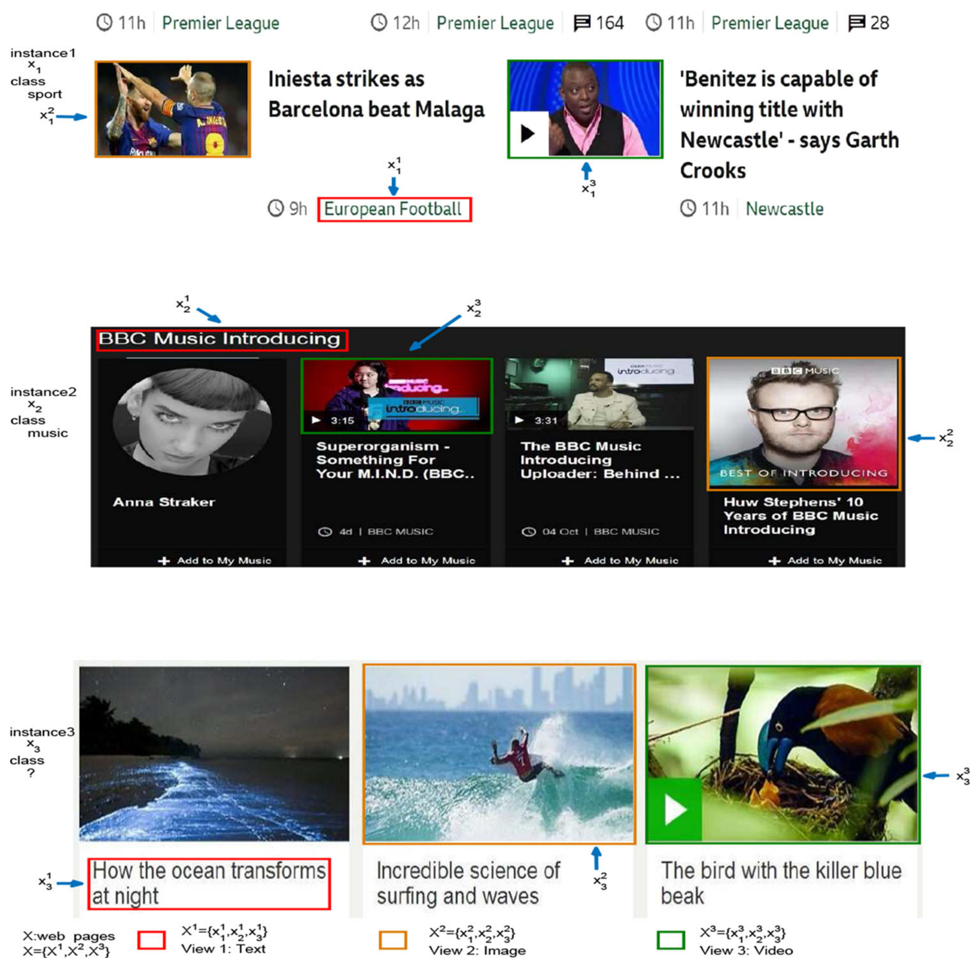
As what we said above, there are two research questions in our work. One is how to increase useful prior knowledge, the other is how to reflect diverse discriminant roles of views and features.

1.2.1 How to increase useful prior knowledge

Although above-mentioned machines achieve considerable success in the domain of machine learning (Sheikhpour et al. 2017), availability of only few labeled instances may affect classification performance (Liu et al. 2016; Dhar 2014). See Fig. 2a,[1] the learning machine fails to learn a robust hyperplane with insufficient labeled instances and enormous unlabeled instances. Indeed, as we said before, due to labeling instances is a high-cost task, thus many real-world data sets consist of insufficient labeled instances and sufficient unlabeled instances. This phenomenon limits the improvement in performance for learning machines. In order to boost the performance, Vapnik and Kotz (1982) develop an algorithm, Universum, to encode prior knowledge by given instances. In

[1] Example in this figure is also given in Liu et al. (2016).

Fig. 1 Example of a semi-supervised multi-view data set



general, Universum learning collects some instances which do not belong to any class of data, but do belong to the same domain as the problem. These collections are named Universum instances which reflect some prior knowledge. By Universum, we can obtain a robust decision hyperplane, see Fig. 2b.

Besides for Vapnik and Kotz (1982), some other references also validate that a semi-supervised learning machine with Universum has a better classification performance, particularly under conditions of insufficient labeled instances. For example, Universum support vector machine (U-SVM, developed by Weston et al.) Weston et al. (2006) is developed to demonstrate that using Universum as a penalty term of the standard support vector machine (SVM) objective function can enhance classification performance. UAdaBoost.MH developed by Liu et al. (2016) explores how Universum impacts semi-supervised learning with insufficient labeled instances for the first time. Moreover, Lap-Universum (U-Lap) and NLap-Universum (U-NLap) which are developed by Zhang et al. (2008) propose a graph-based framework to formulate the semi-supervised learning with Universum problems. With U-Lap and U-NLap, performance of graph-based algorithms can be boosted further. Document clustering with Universum (DCU, developed by Zhang et al. (2011)) validates that Universum instances are useful for semi-supervised clustering task. Regularized matrix-pattern-oriented classification machine with Universum (U-RMM, developed by Li et al. (2017)) validates that Universum is also fit to matrix-instances classification tasks. Due to the advantages of Universum, Universum learning has been gradually spread into body pose recognition (Peng et al. 2008), boosting strategy (Shen et al. 2012), dimensionality reduction technique (Chen et al. 2012), and multi-view learning (Wang et al. 2014; Liu et al. 2014).

But there is a common problem existing in Universum even though above Universum-based learning machines have been validated effectiveness. According to Zhang et al. (2011) and Weston et al. (2006), there are three widely used ways to construct Universum set. Namely, Universum set can be composed of instances from other non-target classes, randomly constructed instances, or instances constructed by randomly combining instances from different classes. For these three ways, they have some disadvantages. In terms of the first way, if instances from some other non-target

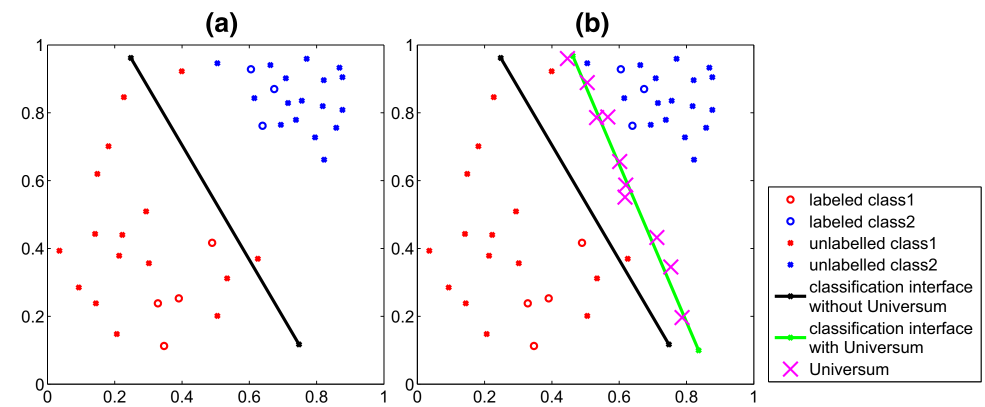


Fig. 2 **a** Example of semi-supervised multi-view learning without Universum. **b** Example of semi-supervised multi-view learning with Universum

classes are also insufficient, Universum learning won't work well. For the second way, although constructing instances in random can bring sufficient Universum instances, it still cannot promise these Universum instances provide enough prior knowledge. For the third way (also called random averaging, RA, developed by Cherkassky et al.), it has been proposed in Cherkassky and Dai (2009) for the first time and according to Cherkassky and Dai (2009), Universum instances are constructed by randomly selecting a pair of positive and negative training instances, and averaging them. Since each constructed Universum instance depends on a pair of positive and negative training instances rather than all (labeled, unlabeled) instances, so the provided useful prior knowledge is still limited even though this way constructs sufficient Universum instances.

Thus how to select a feasible way to construct sufficient and useful Universum instances so as to increase useful prior knowledge is the first research question.

1.2.2 How to reflect diverse discriminant roles of views and features

Compared with the first research problem, the second one is easy to understand.

According to what we said before, for a real-world semi-supervised multi-view data set, different views and features play diverse discriminant roles on designing a learning machine and they should be assigned different weights. For example, for a container data set, picture view is more important than text view to track and recognize the container and we should assign a larger weight for picture view. Another example, for an artwork data set, we want learn about it from text introduction (i.e., text view) and since text content is more important than text color, so we should assign a larger weight for text content. While traditional learning machines treat these views and features have an equivalent role and assign a same weight. This is not feasible and we cannot assign weights for views and features in random without the consideration of their information. Thus how to assign feasible weights to reflect diverse discriminant roles of views and features is the second research problem.

1.3 What is aim of our work

Since there are two main research problems should be solved in our work, thus the aims of work are given as follows.

First, this work aims to design a feasible way to construct sufficient and useful Universum instances and increase more useful prior knowledge. Second, this work aims to assign feasible weights for views and features so as to reflect diverse discriminant roles of them. After solve these two research problems, the performances of semi-supervised multi-view learning machines can be enhanced and finally, the development of national economy and industry can be boosted.

1.4 Why is to propose the WUSM model: motivation and originality

Motivation: in our work, we should solve two research problems. For the first one, we adopt the following idea. According to the ways to construct Universum set which we mentioned before, it is found that averaging operation can mix some prior knowledge belonging to positive and negative training instances. Inspired by this operation, we can put a question that why not to consider the mean of all instances, all labeled instances, or all unlabeled instances? In other words, if we take the mean of all (labeled, unlabeled) instances as a reference point and select some or both original instances as alternative instances, maybe averaging this mean and an alternative instance will obtain sufficient Universum instances with more prior knowledge due to this mean also includes prior knowledge. For the second one, in order to reflect diverse discriminant roles of views and features, some improvements are proposed including weighted multi-view clustering (WMVC, developed by Xu et al. (2016)). WMVC is a multi-view clustering method to extract global and local features and WMVC aims to find the optimal cluster assignment with the consideration of view weights and feature weights. In other words, with the usage of WMVC, we can assign feasible weights for views and features.

Originality: in terms of these two problems, we design some schemes for trial to increase more useful prior knowledge and assign feasible view weights and feature weights with WMVC used. In general, we first obtain view weights and feature weights of a semi-supervised multi-view data set by WMVC. Second, on the base of these weights, we design some schemes to construct Universum set with more prior knowledge and in terms of these schemes, the mean of all instances, all labeled instances, or all unlabeled instances is taken into consideration. Third, we apply the Universum set along with the original data set to a semi-supervised multi-view learning machine. For this proposed learning machine, we name it a weight-and-Universum-based semi-supervised multi-view learning machine (WUSM). To the best of our knowledge, this is the first trial for the combination of averaging all (labeled, unlabeled) instances with WMVC.

According to the framework of WUSM, we can adopt the mean of all instances, all labeled instances, or all unlabeled instances to increase more useful prior knowledge and adopt WMVC to reflect diverse discriminant roles of views and features. Thus the two research problems can be solved by WUSM and the aim of our work can be attained in theory. That's why we propose the WUSM model.

1.5 Contributions

Our WUSM has three main contributions. First, it makes full use of the information of all (labeled, unlabeled) instances rather than only a pair of positive and negative training instances to construct Universum set. This contribution indicates the Universum set possesses more useful prior knowledge. Second, it adopts WMVC to assign feasible weights of views and features and reflect their diverse discriminant roles. Third, this is the first trial to construct Universum set with the combination of averaging all (labeled, unlabeled) instances and WMVC. This trial advances the development of semi-supervised multi-view learning machines. With the usage of WUSM, in the tasks of bipartite ranking, feature selection, dimensionality reduction, classification, clustering, etc., performances of semi-supervised multi-view learning machines boost.

1.6 Organization of our work

The rest of this paper is organized as follows. Section 2 reviews semi-supervised multi-view learning machines, Universum learning, and the ways to construct Universum set. Description of WUSM is given in Sect. 3. Experiments are given in Sect. 4. Section 5 shows further discussions. Section 6 gives the conclusions and future work.

2 Related work

This section reviews some classical semi-supervised multi-view learning machines, Universum learning, and ways to construct Universum set.

2.1 Semi-supervised multi-view learning machines

Semi-supervised multi-view learning machines aim to process multi-view data sets with insufficient labeled instances and sufficient unlabeled instances (Seliya and Khoshgoftaar 2007; Nie et al. 2011; Wang 2011; Liu et al. 2014; Huang et al. 2014) and MVAR (Tao et al. 2017), co-labeling (Xu et al. 2016), and SSMVH (Zhang and Zheng 2017) are widely used machines.

MVAR adopts adaptive regression to address semi-supervised multi-view classification problems. Regressing to class labels directly makes MVAR efficient in calculation and can be applied to large-scale data sets; co-labeling is developed to solve the multi-view weakly labeled learning problem. It models the learning problem on each view as a weakly labeled learning problem and learns an optimal classifier from a set of pseudo-label vectors constructed by using the classifiers trained from other views; SSMVH is a model incorporating a portion of label information. SSMVH minimizes loss jointly on multi-view features when using relaxation on learning hashing codes, explores statistically uncorrelated multi-view features for constructing hash codes, and preserves locally compact coding.

Related experiments have validated the advantages of the above machines. But according to Schölkopf et al. (1997) and Epshteyn and DeJong (2006), prior knowledge has proven useful for classification and it is notoriously difficult to apply in practice due to the difficulty of explicitly specifying prior knowledge. Thus, the scope of applications of these methods are limited.

2.2 Universum learning

Universum learning was initially proposed by Vapnik and Kotz (1982) and it encodes prior knowledge by given instances. To the best of our knowledge, the first learning machine to explore how Universum impacts semi-supervised learning machine with insufficient labeled instances is U-SVM (Weston et al. 2006). U-SVM collects instances which do not belong to any target class as Universum instances and helps to encode prior knowledge by representing meaningful concepts in the same domain. Then, Sinz et al. (2008) analyze the influence of the Universum on U-SVM further.

In the recent ten years (specially from 2016 to present), with the success of Universum learning, many learning machines with Universum are proposed in the field of graph-based methods (Zhang et al. 2008), clustering (Zhang et al.

2011), boosting strategy (Shen et al. 2012), dimensionality reduction techniques (Chen et al. 2012) and so on. For example, UAdaBoost.MH (Liu et al. 2016), U-RMM (Li et al. 2017), Universum canonical correlation analysis (UCCA, developed by Chen et al. (2018)) and non-parallel hyperplane Universum support vector machine (U-NHSVM, developed by Zhao et al. (2019)).

UAdaBoost.MH incorporates Universum learning with AdaBoost.MH which maintains a set of weights over all training instances. Moreover, UAdaBoost.MH is a new learning machine with using boosting techniques and Universum instances; U-RMM adopts a previous work MatMHKS (Chen et al. 2007) which is developed by Chen et al. and can be used to classify the matrixized instances as the basic and incorporates it with Universum learning (Li et al. 2017); UCCA aims to find basis vectors in multiple views to ensure that correlations between projections of target data are mutually maximized but correlations between projections of Universum data and target data mutually minimized; U-NHSVM shows flexibility by exploiting the prior knowledge ensconced in Universum and provides consistency by constructing two non-parallel hyperplanes simultaneously.

Although many learning machines with Universum are proposed and developed in this decade, there are some issues should be overcome. First is that some learning machines with Universum are developed on the base of supervised learning problems and they have no ability to process semi-supervised learning problems, for example, Uboost. Second is that performances of those learning machines with Universum are influenced by the ways to construct Universum set.

2.3 Ways to construct Universum set

As what we mentioned above, different ways to construct Universum set affect the performances of learning machines. For example, U-SVM (Weston et al. 2006) collects instances which from some other non-target classes as the Universum instances and these Universum instances can encode prior knowledge of data. But U-SVM will be useless when instances of other classes are insufficient and how to choose Universum instances from non-target classes so as to achieve the most improvement is also not taken into consideration in U-SVM. Thus, finding in-between Universum (FIBU, developed by Chen and Zhang (2009)) is proposed to select informative Universum instances, i.e., in-between Universum (IBU) instances which deposit in between the two different classes. After that, in order to enhance the prior knowledge further, RA (Cherkassky and Dai 2009) gathers Universum instances using a random selection approach, in which instances are constructed by randomly selecting a pair of positive and negative training instances, and averaging them. Zhu (2016) and Li et al. (2017) improve RA and develop a creating in-between Universum (CIBU) algorithm, respectively. CIBU selects a pair of neighbor training instances rather than a pair of positive and negative training instances to construct Universum set. The CIBU developed by Zhu (abbr. CIBU-Zhu) is adapted to vector instances including the data sets from UCI machine learning repository (Asuncion and Newman 2007) while the CIBU developed by Li et al. (abbr. CIBU-Li) is adapted to matrix instances including images. Moreover, v-twin support vector machine with Universum (Uv-TSVM, developed by Xu et al.) is proposed (Xu et al. 2016) and it is different from CIBU. In Uv-TSVM, it uses two Hinge loss functions to locate Universum instances in a non-parallel insensitive loss tube rather than constructing Universum instances by randomly combining neighbor instances. Although these ways can construct sufficient Universum instances and possess more prior knowledge about data sets, they still neglect the various discriminant roles of different views and features.

3 Weight-and-Universum-based semi-supervised multi-view learning machine

We develop WUSM with the following three steps to improve the performance of a traditional semi-supervised multi-view learning machine.

3.1 Obtaining weights of views and features based on WMVC (Xu et al. 2016)

Suppose $\mathbf{X} = \{\mathbf{X}^v\}_{v=1}^V$=$\{x_i\}_{i=1}^N$ is a semi-supervised multi-view data set where V is the number of views and N is the number of training instances. The v-th view is $\mathbf{X}^v$=$\{x_i^v\}_{i=1}^N$, the i-th instance is $x_i = \{x_i^v\}_{v=1}^V$, and x_i^v represents v-th view of i-th instance. For each view $\mathbf{X}^v$, its dimension is d^v and this indicates the view consists of d^v features.

Then let ω_v denotes the weight of v-th view and τ_l^v denotes the weight of l-th feature of v-th view where $v = 1, 2, \ldots, V$ and $l = 1, 2, \ldots, d^v$ with the constraints $\omega_v \geq 0$, $\tau_l^v \geq 0$, $\sum_{v=1}^V \omega_v = 1$, $\sum_{l=1}^{d^v} \tau_l^v = 1$. Then the whole semi-supervised multi-view data set is divided into M clusters with WMVC and if x_i belongs to k-th cluster, we let $\delta_{ik} = 1$, otherwise, $\delta_{ik} = 0$. Here, for any instance x_i, $\sum_{k=1}^M \delta_{ik} = 1$.

In order to take the weights into consideration, we construct the objective function with Eq. (1) where $\varepsilon_H = \sum_{v=1}^V (\omega_v)^p \sum_{i=1}^N \sum_{k=1}^M \delta_{ik} \, || \, diag(\tau^v)(x_i^v - m_k^v) \, ||^2 + \beta \sum_{v=1}^V || \, \tau^v \, ||^2$, $\tau^v = \{\tau_1^v, \tau_2^v, \ldots, \tau_{d^v}^v\}$, and $diag(\tau^v)$ is a diagonal matrix. Moreover, $m_k^v = (\sum_{i=1}^N \delta_{ik} x_i^v)/(\sum_{i=1}^N \delta_{ik})$ represents the cluster center of k-th cluster in the v-th view. Furthermore, $\beta \sum_{v=1}^V || \, \tau^v \, ||^2$ is used to control the sparsity of the feature weight vectors τ^v, $\forall v$ so as to avoid the situation that only few features are selected in getting a very small but meaningless objective value. According to priori knowledge of data, exponential parameter p and balance parameter β are selected to help controlling the sparsity of the view weight vector $\omega = \{\omega_v\}_{v=1}^{V}$ and the feature weight vectors $\tau^v, \forall v = 1, 2, \ldots, V$, respectively.

With the solution of Eq. (1), $\omega = \{\omega_1, \omega_2, \ldots, \omega_V\}$ and feature weight vectors $\tau^v, \forall v = 1, 2, \ldots, V$ can be obtained and optimized. Reference Xu et al. (2016) shows the detailed procedure.

$$
\begin{aligned}
&\min_{\delta_{ik}, \omega_v, \tau^v} \quad \varepsilon_H \\
s.t. \quad &\sum_{k=1}^{M} \delta_{ik} = 1, \quad \forall i, \quad \delta_{ik} \in \{0, 1\} \\
&\sum_{v=1}^{V} \omega_v = 1, \quad \omega_v \geq 0 \\
&\sum_{l=1}^{d^v} \tau_l^v = 1, \quad \forall v, \quad \tau_l^v \geq 0
\end{aligned} \tag{1}
$$

Algorithm 1 Semi-supervised multi-view learning machine with Universum set for a binary class classification problem.

Input:

The labeled positive set: L_P; The labeled negative set: L_N;
The unlabeled set: UL; The Universum set: Un;
Index of iteration: i; The maximum iteration: T;
The number of labeled instances in each step: M;
The number of unlabeled instances which are added into the labeled set in each step: m;
Other parameters of the learning machine;
The predicted unlabeled instances which locate far from the center of positive or negative Universum instances are reliable instances.

Output:

Decision function of the learning machine;

1: Construct Un, train the semi-supervised multi-view learning machine on Un, L_P, and L_N and get decision function.
2: Apply decision function to classify UL and Un. UL is classified as two different classes, UL^+ and UL^- while Un is also classified as two different classes, Un^+ and Un^-.
3: Compute the mean of Un^+ as p-center while the mean of Un^- as n-center.
4: Select m unlabeled instances in UL^+ which are farthest from p-center as U_P and m unlabeled instances in UL^- which are farthest from n-center as U_N;
5: Let $L_P = \{L_P; U_P\}$ and $L_N = \{L_N; U_N\}$.
6: Then let $i = i + 1$, go to 1 until $i = T$ or these sets don't change anymore;

3.2 Constructing Universum set based on labeled and unlabeled instances

In our work, in order to obtain more prior knowledge, we refer to the meaning of averaging operation, compute the mean of all (labeled, unlabeled) instances and designing some schemes to construct Universum set with more useful prior knowledge. Table 1 shows the codes of the used ways to construct Universum set and details are given as follows.

(A) Center is computed by all instances.

$\dot{U}_{1-1}$: We first compute the mean of **all** instances as a center. Then we take the **midpoint** of an instance and the center to construct Universum set.

$\dot{U}_{1-2}$: We first compute the mean of **all** instances as a center. Then we select K instances which locate **nearest** from this center. Finally we take the **midpoint** of a selected instance and the center to construct Universum set.

$\dot{U}_{1-3}$: We first compute the mean of **all** instances as a center. Then we select K instances which locate **farthest** from this center. Finally we take the **midpoint** of a selected instance and the center to construct Universum set.

$\dot{U}_{1-4}$: We first compute the mean of **all** instances as a center. Then we select K instances which locate **nearest** from this center to construct Universum set.

$\dot{U}_{1-5}$: We first compute the mean of **all** instances as a center. Then we select K instances which locate **farthest** from this center to construct Universum set.

(B) Center is computed by unlabeled instances.

$\dot{U}_{2-1}$, $\dot{U}_{2-2}$, $\dot{U}_{2-3}$, $\dot{U}_{2-4}$, and $\dot{U}_{2-5}$ are very similar with $\dot{U}_{1-1}$, $\dot{U}_{1-2}$, $\dot{U}_{1-3}$, $\dot{U}_{1-4}$, and $\dot{U}_{1-5}$, respectively. **The difference is that the center is depended on unlabeled instances rather than all instances**. Moreover, during the processing, Universum set is changed since the unlabeled instances are changed. Take $\dot{U}_{2-1}$ as an example. We first compute the mean of unlabeled instances as a center. Second, we take the midpoint of each instance (including the labeled and unlabeled) and the center to construct Universum set. Third, we apply the Universum set, unlabeled set, and labeled set to a semi-supervised multi-view learning machine. After the training of the semi-supervised multi-view learning machine, some unlabeled instances will be labeled, and the set of unlabeled instances will be changed. Meanwhile, the Universum set will also be changed.

(C) Center is computed by labeled instances.

$\dot{U}_{3-1}$, $\dot{U}_{3-2}$, $\dot{U}_{3-3}$, $\dot{U}_{3-4}$, and $\dot{U}_{3-5}$ are very similar with $\dot{U}_{2-1}$, $\dot{U}_{2-2}$, $\dot{U}_{2-3}$, $\dot{U}_{2-4}$, and $\dot{U}_{2-5}$, respectively. **Only difference is that the mean of labeled instances is taken as a center.**

3.3 Applying Universum set to a semi-supervised multi-view learning machine

After constructing Universum set, we train a semi-supervised multi-view learning machine with labeled, unlabeled, and Universum set. Procedure is given in Fig. 3 and Algorithm 1 and an example is given in Fig. 4. According to Fig. 3, for

Table 1 The codes of used Universum set construction ways

Code	Way	Code	Way	Code	Way
$\dot{U}_{1-1}$	All-all-mid	$\dot{U}_{2-1}$	Unlabeled-all-mid	$\dot{U}_{3-1}$	Labeled-all-mid
$\dot{U}_{1-2}$	All-near-mid	$\dot{U}_{2-2}$	Unlabeled-near-mid	$\dot{U}_{3-2}$	Labeled-near-mid
$\dot{U}_{1-3}$	All-far-mid	$\dot{U}_{2-3}$	Unlabeled-far-mid	$\dot{U}_{3-3}$	Labeled-far-mid
$\dot{U}_{1-4}$	All-near-self	$\dot{U}_{2-4}$	Unlabeled-near-self	$\dot{U}_{3-4}$	Labeled-near-self
$\dot{U}_{1-5}$	All-far-self	$\dot{U}_{2-5}$	Unlabeled-far-self	$\dot{U}_{3-5}$	Labeled-far-self

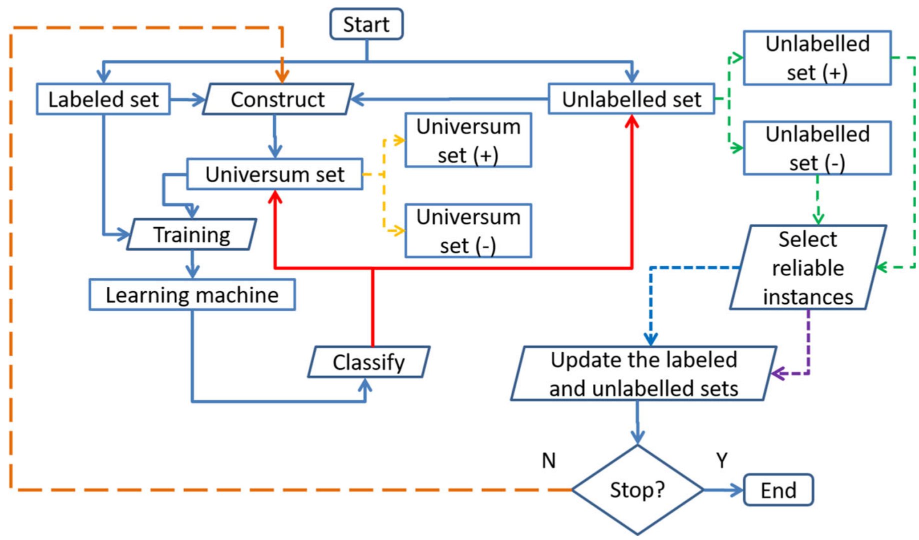


Fig. 3 Framework of how to apply Universum set and the original data set to a semi-supervised multi-view learning machine

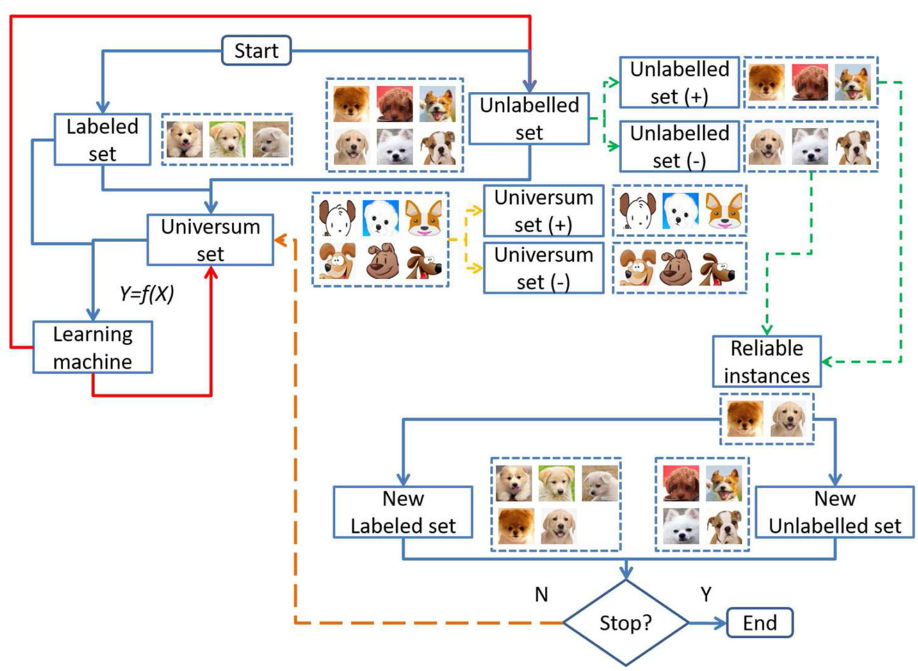


Fig. 4 Example of the framework that how to apply Universum set and the original data set to a semi-supervised multi-view learning machine

a binary class problem,[2] we first train a semi-supervised multi-view learning machine on labeled and Universum set. Second, with the usage of this learning machine, we classify the unlabeled and Universum set iteratively. The classified results can be labeled with '+' or '−' or other symbols. Third, guided by some criterions of selection, we select some reliable instances from the predicted unlabeled instances and

[2] For multiple classes, it can be divided into several binary class problems and the solution is the combination of optimal results of those binary class problems.

update the original labeled and unlabeled sets. The procedure will stop if these sets don't change anymore or the maximum iteration achieves.

In the framework, the criterions of selection represents that if the predicted unlabeled instances locate far from the classification boundary, or center computed by all labeled and unlabeled instances, or center computed by Universum set (shown in Algorithm 1), we think these predicted unlabeled instances are reliable.

3.4 How to apply weights of views and features

When we compute the center, mean, or midpoint, weights of views and features should be used. Simply speaking, in order to get the center, mean, or midpoint, the weights are treated as coefficients and weighted average method is used for computation. Moreover, in order to search the nearest or farthest instances, the distances between two points or instances should be computed. For example, if there are two instances, $x_i = \{x_i^v\}_{v=1}^V$ and $x_j = \{x_j^v\}_{v=1}^V$ where x_i^v or x_j^v represents the v-th view of i-th or j-th instance, respectively, and suppose the weight of v-th view is ω_v and weight of l-th feature of this view is τ_l^v. Then the distance between them is $d = \sum_{v=1}^V \sum_{l=1}^{d^v} \left|\left|\omega_v \tau_l^v (x_{il}^v - x_{jl}^v)\right|\right|_2^2$ where x_{il}^v (x_{jl}^v) is l-th feature of v-th view of i-th (j-th) instance.

3.5 Computational complexity analysis

Computational complexity of WUSM consists of three parts, namely, complexity of obtaining weights of views and features, complexity of constructing Universum set, and complexity of carrying out the semi-supervised multi-view learning machine with Universum set. In terms of obtaining weights of views and features, according to WMVC (Xu et al. 2016), the complexity should be $O(NMd + d + VNM)$. For the complexity of constructing Universum set, it should be $O(N)$ in generally. For carrying out the semi-supervised multi-view learning machine with Universum set, the complexity is depended on the one of the used learning machine. We suggest this complexity be $O(A)$. Then the total computational complexity of WUSM is $O(NMd + d + VNM) + O(N) + O(A)$. Compared with previous learning machines, the additional computational complexity of WUSM is $O(NMd + d + VNM) + O(N)$.

Moreover, according to the total computational complexity of WUSM, it is found the complexity is nothing to do with the number of classes. Thus, in WUSM, converting the multi-class problems into several binary problems won't bring additional computational complexity in theory.

3.6 Convergence

Convergence is an important index to measure the performance of a learning machine. According to the above contents, due to constructing Universum set need not to optimize some parameters, thus the convergence of WUSM depends on the convergence of WMVC and the one of used semi-supervised multi-view learning machine. Since the convergence of WMVC can be promised by Xu et al. (2016) and the one of used semi-supervised multi-view learning machine can be promised by related reference, thus the convergence of our proposed WUSM is also guaranteed.

3.7 Advantage of WUSM over other existing methods

Compared with the traditional semi-supervised multi-view learning machines, the proposed WUSM can process the multi-view data sets with insufficient labeled instances and enormous unlabeled instances. Moreover, it can take the diverse discriminant roles of views and features into consideration.

Compared with the traditional ways to construct Universum set, with the WUSM used, information of all instances, all labeled instances, or all unlabeled instances is taken into consideration during the construction of Universum set. Then more useful prior knowledge is provided.

4 Experiments

Experimental setting and performance comparisons are given in this section.

4.1 Experimental setting

Experimental setting includes description of data sets and compared methods, parameter setting, and experimental set up.

4.1.1 Description of data sets

WUSM are conducted on multi-view data sets Mfeat (see Table 2), Reuters (see Table 3), and Corel (see Table 4). With the limitation of this paper, we describe them in simple as follows. The details can be found in WMVC (Xu et al. 2016).

Mfeat[3] consists of features of handwritten numerals ('0'–'9') extracted from a collection of Dutch utility maps. Each handwritten numeral (class) has 200 instances and each

[3] http://archive.ics.uci.edu/ml/datasets/Multiple+Features.

Table 2 Detailed information of Mfeat. In this table, number of features represents dimension

View	No. instances	No. features	No. digits
Profile correlations (fac)	2000	216	10
Fourier coefficients of the character shapes (fou)	2000	76	10
Karhunen–Love coefficients (kar)	2000	64	10
Pixel averages in 2×3 windows (pix)	2000	240	10
Zernike moments (zer)	2000	47	10
Morphological features (mor)	2000	6	10

Table 3 Detailed information of Reuters. In this table, vocabulary size represents dimension

View	No. documents	Vocabulary size	Topic	No. documents	Percentage (%)
English (EN)	18,758	21,513	C15	18,816	16.84
French (FR)	26,648	24,839	CCAT	21,426	19.17
German (GR)	29,953	34,279	E21	13,701	12.26
Spanish (SP)	12,342	11,547	ECAT	19,198	17.18
Italian (IT)	24,039	15,506	GCAT	19,178	17.16
			M11	19,421	17.39

Table 4 Detailed information of Corel. In this table, number of features represents dimension

View	No. images	No. features	No. categories
Color histogram (Col-h)	1000	32	10
Color histogram layout (Col-hl)	1000	32	10
Color moments (Col-m)	1000	9	10
Co-occurrence texture (Coo-t)	1000	16	10

instance has 6 views (Asuncion and Newman 2007); Reuters[4] consists of machine translated documents (instances) which are written in 5 different languages (views) and these documents are also categorized into 6 different topics (classes) (Amini et al. 2009, 17); Corel[5] contains image features extracted from a Corel image collection (Asuncion and Newman 2007). In our experiments, we randomly select 1000 photo images (instances) from 10 various categories (classes) and each category has 100 photos. The 10 categories are C0-Africa, C1-Beach, C2-Building, C3-Buses, C4-Dinosaurs, C5-Elephants, C6-Flowers, C7-Horses, C8-Mountains and C9-Food and for each instance, 4 sets of features (views) are adopted.

4.1.2 Description of compared methods

In order to validate the effectiveness of WUSM, some traditional Universum construction ways are used for comparisons. They are CIBU-Zhu (Zhu 2016), CIBU-Li (Li et al. 2017), and Uv-TSVM (Xu et al. 2016).

Moreover, we adopt multiple semi-supervised multi-view learning machines for comparisons in different fields so that the effectiveness of WUSM can be validated in different applications (see the following contents and detailed description of each machine can be found in their respective references).

Bipartite ranking: semi-supervised multi-view ranking (SmVR, developed by Usunier et al. (2011)).

Feature selection: multi-view Laplacian sparse feature selection (MLSFS, developed by Shi et al. (2015)).

Dimensionality reduction: multiple-view semi-supervised dimensionality reduction (MVSSDR, developed by Hou et al. (2010)).

Classification: multi-view semi-supervised learning proposed by Zhu et al. (MvSs-Zhu (Zhu et al. 2016) , multiple-view multiple-learner (MVML, developed by Sun and Zhang (2011)), co-graph (developed by Du et al. (2013)), adaptive multi-view selection (AMVS, developed by Yang et al. (2014)), MVAR (Tao et al. 2017), co-labeling (Xu et al. 2016).

Clustering: semi-supervised unified latent factor learning (SULF, developed by Jiang et al. (2014)) and semi-supervised three-way clustering (SSTC, developed by Yu et al. Yu et al. (2017)).

[4] http://archive.ics.uci.edu/ml/datasets/Reuters+RCV1+RCV2+Multilingual\%2C+Multiview+Text+Categorization+Test+collection.

[5] http://archive.ics.uci.edu/ml/datasets/Corel+Image+Features.

Table 5 Parameter settings of WUSM

Step	Parameter	Notation	Initial value
1	The weight of v-th view	ω_v	$\frac{1}{V}$
	The weight of l-th	τ_l^v	$\frac{1}{d^v}$
	Feature of v-th view		
	Exponential parameter	p	$\{5, 10, 15, 20, 25, 30\}$
	Balance parameter	β	$(0, 1]$
	The number of clusters	M	Based on the ground-truth labels of the instances
2	Number of nearest	K	$\{1, \ldots, N_{e-\max}\}$
	Or farthest neighbors		
3	The maximum iteration	T	100
	Number of unlabeled	m	$\{0.1, 0.2, \ldots, 1\%\}$
	Instances which are added into the labeled set in each step		$\times$ Nul

Limited by the length of this paper, for each experimental subsection, if there is no special explanation, we only show the results on one data set. But the conclusions derived from the results are also fit to other two data sets.

4.1.3 Parameter setting

First, we select 10% instances of each data set for test in random and the rest 90% are used for training. For the training set, we further label 10%, 20%, 30%, 40%, 50%, 60%, 70%, 80%, and 90% instances and the rest are used as unlabeled instances, respectively, in random.

Second, due to WUSM consists of three steps (see Sects. 3.1, 3.2, 3.3), thus in Table 5 (steps 1, 2, and 3) shows the corresponding parameter settings. Moreover, in Table 5, $N_{e-\max} = N_t - N_{\max}$. N_t is the total number of training instances and $N_{\max}$ is the number of instances from largest training class.[6] Nul is the number of original unlabeled instances. Furthermore, since we construct Universum set with 15 ways and select reliable instances with 3 criterions, so in order to show experimental results clearly, we adopt symbols in Table 6 to simplify the clarifications.

Third, parameter settings of other compared Universum construction ways are given in Table 7.

Fourth, parameter settings of other compared semi-supervised multi-view learning machines can be found in their respective references.

Moreover, in order to get best parameters, we adopt grid-search approach. Simply speaking, for any one of Universum sets, we try all other parameters and get the corresponding best parameters for each Universum set. Then we conduct experiments on 45 kinds of Universum sets and choose the best one.

4.1.4 Experimental set up

All computations are performed on a node of compute cluster with 16 CPUs (Intel Core Due 3.0GHz) running RedHat Linux Enterprise 5 with 48GB main memory. The coding environment is MATLAB 2016.

4.2 Performance comparisons for different applications

4.2.1 Bipartite ranking performances comparisons

In this experiment, we first adopt WUSM to construct more Universum instances and then apply the Universum set along with the original data set into SmVR until optimal decision function of SmVR is gotten. Similar with what SmVR has done (Usunier et al. 2011) and for convenience, we show the average precision (AvP) and area under the ROC[7] curve (AUC) for SmVR on Reuters (see Table 8).

From this table, if we rank these compared Universum construction ways in terms of AUC and AvP, it is found that for each class (topic), WUSM > CIBU-Li > CIBU-Zhu > Uv-TSVM > Original. Namely, with WUSM, SmVR performs best while without any Universum processing, SmVR performs worst. The experimental results indicate that

- Averaging all (labeled, unlabeled) instances brings more prior knowledge compared with averaging a pair of training instances.
- Although CIBU-Zhu and CIBU-Li both select a pair of neighbor training instances to construct Universum set, CIBU-Li has a better performance than CIBU-Zhu. As Zhu and Gao (2015) said, compared with an algorithm based on vector instances, an algorithm based on matrix instances has two advantages, one is reducing the computational complexity and the other is improving the classification performance. Since CIBU-Li is designed on the base of matrix instances while CIBU-Zhu is developed on the base of vector instances, so CIBU-Li outperforms CIBU-Zhu in average.

[6] For example, a training data set consists of three classes, one has 100 instances, another has 120 instances, and the third has 140 instances, then $N_{e-\max} = 220$.

[7] ROC: receiver operating characteristic.

Table 6 Symbols and orders of different Universum construction ways and criterions for WUSM. Here, $\tilde{a}$ represents 'far from the classification boundary', $\tilde{b}$ represents 'far from center computed by all labeled and unlabeled instances', $\tilde{c}$ represents 'far from center computed by Universum set'

Criterion	$\tilde{a}$	$\tilde{b}$	$\tilde{c}$	Criterion	$\tilde{a}$	$\tilde{b}$	$\tilde{c}$	Criterion	$\tilde{a}$	$\tilde{b}$	$\tilde{c}$
$\dot{U}_{1-1}$	1-$\dot{U}_{11\tilde{a}}$	16-$\dot{U}_{11\tilde{b}}$	31-$\dot{U}_{11\tilde{c}}$	$\dot{U}_{2-1}$	6-$\dot{U}_{21\tilde{a}}$	21-$\dot{U}_{21\tilde{b}}$	36-$\dot{U}_{21\tilde{c}}$	$\dot{U}_{3-1}$	11-$\dot{U}_{31\tilde{a}}$	26-$\dot{U}_{31\tilde{b}}$	41-$\dot{U}_{31\tilde{c}}$
$\dot{U}_{1-2}$	2-$\dot{U}_{12\tilde{a}}$	17-$\dot{U}_{12\tilde{b}}$	32-$\dot{U}_{12\tilde{c}}$	$\dot{U}_{2-2}$	7-$\dot{U}_{22\tilde{a}}$	22-$\dot{U}_{22\tilde{b}}$	37-$\dot{U}_{22\tilde{c}}$	$\dot{U}_{3-2}$	12-$\dot{U}_{32\tilde{a}}$	27-$\dot{U}_{32\tilde{b}}$	42-$\dot{U}_{32\tilde{c}}$
$\dot{U}_{1-3}$	3-$\dot{U}_{13\tilde{a}}$	18-$\dot{U}_{13\tilde{b}}$	33-$\dot{U}_{13\tilde{c}}$	$\dot{U}_{2-3}$	8-$\dot{U}_{23\tilde{a}}$	23-$\dot{U}_{23\tilde{b}}$	38-$\dot{U}_{23\tilde{c}}$	$\dot{U}_{3-3}$	13-$\dot{U}_{33\tilde{a}}$	28-$\dot{U}_{33\tilde{b}}$	43-$\dot{U}_{33\tilde{c}}$
$\dot{U}_{1-4}$	4-$\dot{U}_{14\tilde{a}}$	19-$\dot{U}_{14\tilde{b}}$	34-$\dot{U}_{14\tilde{c}}$	$\dot{U}_{2-4}$	9-$\dot{U}_{24\tilde{a}}$	24-$\dot{U}_{24\tilde{b}}$	39-$\dot{U}_{24\tilde{c}}$	$\dot{U}_{3-4}$	14-$\dot{U}_{34\tilde{a}}$	29-$\dot{U}_{34\tilde{b}}$	44-$\dot{U}_{34\tilde{c}}$
$\dot{U}_{1-5}$	5-$\dot{U}_{15\tilde{a}}$	20-$\dot{U}_{15\tilde{b}}$	35-$\dot{U}_{15\tilde{c}}$	$\dot{U}_{2-5}$	10-$\dot{U}_{25\tilde{a}}$	25-$\dot{U}_{25\tilde{b}}$	40-$\dot{U}_{25\tilde{c}}$	$\dot{U}_{3-5}$	15-$\dot{U}_{35\tilde{a}}$	30-$\dot{U}_{35\tilde{b}}$	45-$\dot{U}_{35\tilde{c}}$

Table 7 Parameter settings of other compared Universum construction ways

Method	Parameter	Notation	Initial value
CIBU-Zhu	Number of nearest neighbors	K	3
	Control parameter	Ux_k	$\{0.1, 0.2, \ldots, 1, 2, \ldots, 10\}$
CIBU-Li	Number of nearest neighbors	K	3
	Control parameter	UA_k	$\{0.01, 0.1, 0.3, 0.5, 0.8, 1, 1.2, 1.5, 1.8, 2\}$
Uv-TSVM	Control parameter	v	$\{0.1, 0.2, \ldots, 0.9, 1\}$
	Slack parameter	ε	$\{0.1, 0.2, 0.3, 0.4, 0.5, 0.6\}$
	Constant parameter	c	$\{2^i \mid i = -4, -3, \ldots, 8\}$
	Gaussian kernel parameter	γ	$\{2^i \mid i = -4, -3, \ldots, 8\}$

Table 8 AvP and AUC for SmVR on Reuters in terms of bipartite ranking performance. Here, original represents the original Reuters without any Universum processing and the best performance is given in bold

Topic	C15		CCAT		E21		ECAT		GCAT		M11	
Reuters	AvP	AUC	AvP	AUC	AvP	AUC	AvP	AUC	AvP	AUC	AvP	AUC
Original	0.443	0.783	0.347	0.708	0.307	0.667	0.308	0.682	0.519	0.846	0.711	0.881
Uv-TSVM	0.451	0.785	0.350	0.720	0.307	0.681	0.309	0.692	0.536	0.848	0.717	0.884
CIBU-Zhu	0.452	0.799	0.360	0.724	0.308	0.696	0.309	0.700	0.542	0.856	0.736	0.884
CIBU-Li	0.454	0.802	0.360	0.726	0.313	0.713	0.314	0.711	0.543	0.864	0.740	0.889
WUSM	**0.460**	**0.817**	**0.361**	**0.738**	**0.315**	**0.721**	**0.315**	**0.729**	**0.546**	**0.876**	**0.771**	**0.925**

- Since Uv-TSVM pays more attention to locate Universum instances rather than the construction of Universum instances, thus sometimes Uv-TSVM cannot promise the prior knowledge of Universum instances be sufficient. That's why Uv-TSVM cannot perform better than other three Universum construction ways.
- Since Universum instances can provide some prior knowledge of data, thus without any Universum processing means no more prior knowledge can be provided. According to the experiments about bipartite ranking, WUSM is validated to be effective for this task. In other words, with WUSM used, SmVR can improve its performance in both AUC and AvP.

4.2.2 Feature selection performances comparisons

In this experiment, we adopt MLSFS which utilizes multi-view Laplacian regularization to boost semi-supervised sparse feature selection performance to validate the advantage of WUSM for feature selection.

First, we show the average performances of MLSFS for Mfeat from different evaluation metrics[8] including precision, recall, specificity, accuracy, and F-measure. See Fig. 5. From this figure, it is found that WUSM can outperform others in average in terms of feature selection performance. More-

[8] For these evaluation metrics, precision $= \frac{TP}{TP+FP}$, recall $= \frac{TP}{TP+FN}$, specificity $= \frac{TN}{TN+FP}$, accuracy $= \frac{TP+TN}{TP+FP+TN+FN}$, and F-measure $= \frac{2\text{recall}\times\text{precision}}{\text{recall}+\text{precision}}$. Here, TN: true negative, TP: true positive, FP: false positive, FN: false negative.

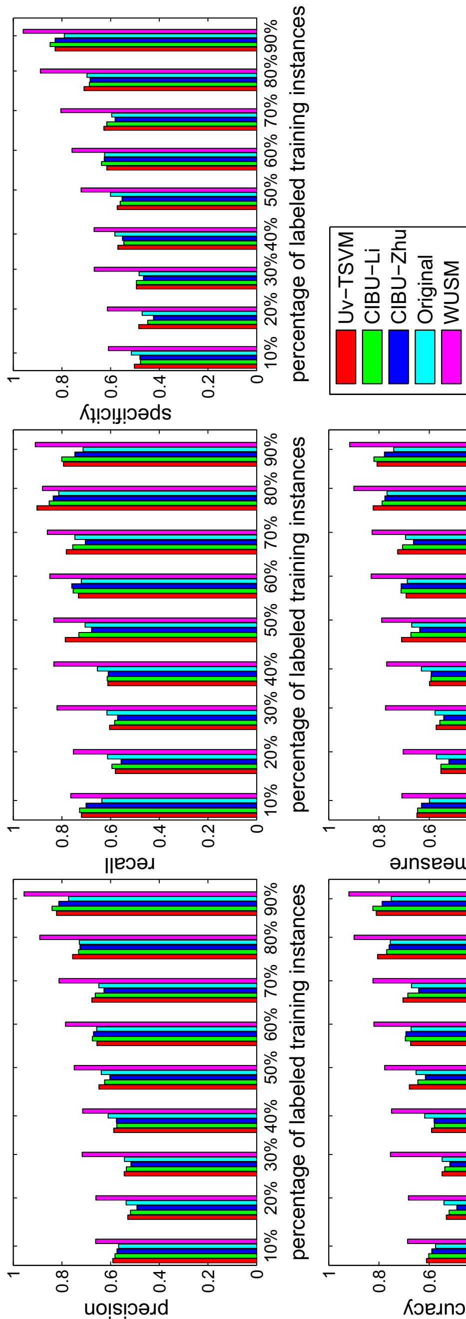


Fig. 5 Performance comparison with MLSFS and different Universum construction ways used on Mfeat

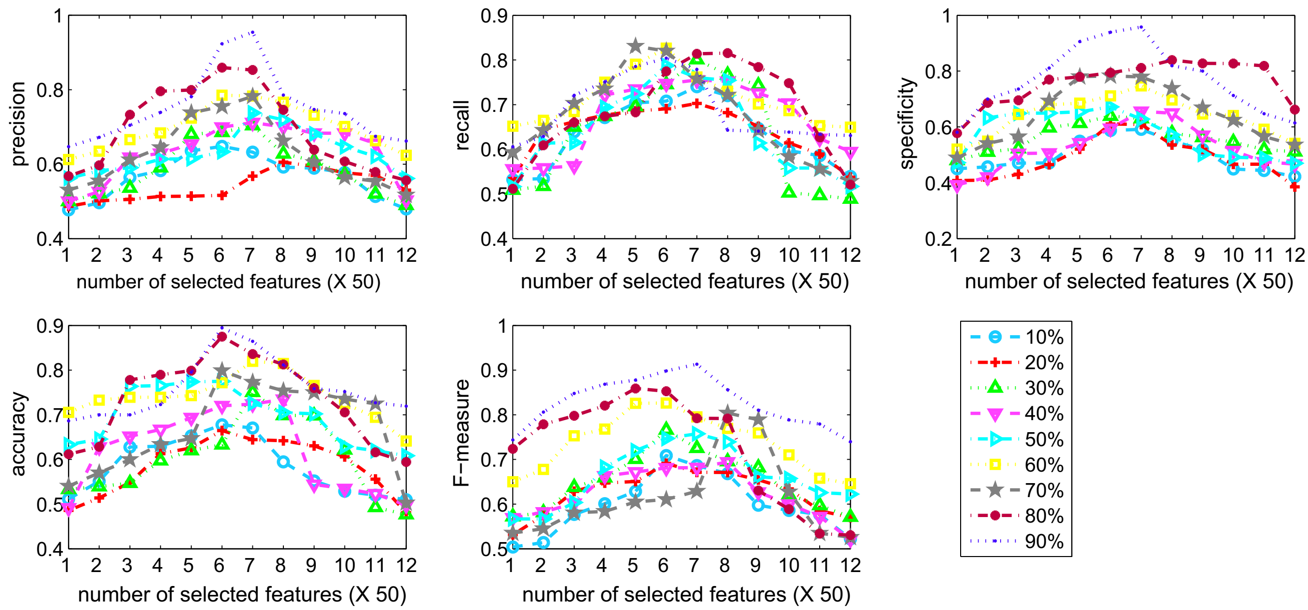


Fig. 6 Performance variation of MLSFS with different numbers of selected features and percentages of labeled training instances when Mfeat used

over, with the increasing of percentage of labeled training instances, the performance of MLSFS is also boosted, especially under the case of WUSM.

Second, we illustrate the performance of MLSFS varies when the number of selected features changes. For convenience, Universum construction way WUSM and data set Mfeat are adopted for illustration. Indeed, for other cases, the conclusions are similar. Although different views of a multi-view data set have different features, we still show the total number of selected features. In Fig. 6, we illustrate values of evaluation metrics of MLSFS varies when the number of selected features and percentage of labeled training instances change with Mfeat used. From this figure, it is found that values of evaluation metrics of MLSFS increases with the number of selected features increases, and then it decreases with the number of selected features increases after arriving the peak. This result is similar with the ones given in Shi et al. (2015) which indicates that with WUSM used, MLSFS still keeps its properties.

4.2.3 Dimensionality reduction performances comparisons

As is known to all, information derived from some or all views (features) can guide the labeling of a multi-view data set. Thus, the performance varies under different number of views and features is worthy to be discussed. A good method to change the number of views or features is dimensionality reduction. Different from feature selection which aims to select a subset of relevant features (variables, predictors) for use in model construction, dimensionality reduction aims to reduce the number of random variables under consideration via obtaining a set of principal variables. Here, we adopt MVSSDR (Hou et al. 2010) for experiments so as to validate the effectiveness of WUSM in dimensionality reduction and discuss the influence of number of used views, percentages of labeled training instances, and dimensions. For convenience, we only adopt Mfeat to show the experimental results. In terms of this experiment, Fig. 7 shows the performance comparisons of MVSSDR with Universum construction ways and numbers of views used on Mfeat. Figure 8 shows the accuracy of MVSSDR under different numbers of views and dimensions on Mfeat.[9] Figure 9 shows the accuracy of MVSSDR under different percentages of labeled training instances and dimensions on Mfeat and in this figure, *m* represents the percentage of labeled training instances.

According to these figures, we can draw the following conclusions. (1) Numbers of views has a great influence on the performance of MVSSDR and more views brings a better performance no matter which evaluation metric is adopted. (2) In terms of different evaluation metrics, it is found that the average rank of different Universum construction ways is WUSM > CIBU-Li > CIBU-Zhu > Uv-TSVM > Original. This result is very similar with the one given in 4.2.1. While compared with the previous experimental results, in the task of dimensionality reduction, WUSM brings a larger promote. Especially, for precision, recall, and F-measure, the performance values of WUSM

[9] Limited by the length of this paper, we only show the results about accuracy rather than precision, recall, specificity, and F-measure. But the results on other evaluation metrics won't change our conclusions.

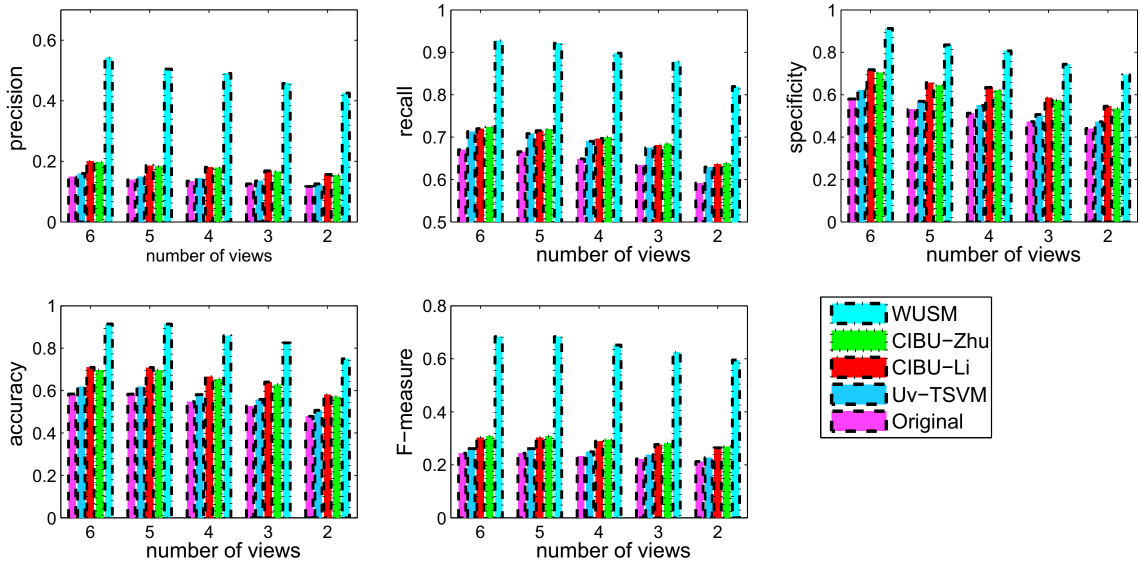


Fig. 7 Performance comparisons with different Universum construction ways and numbers of views when MVSSDR and Mfeat used

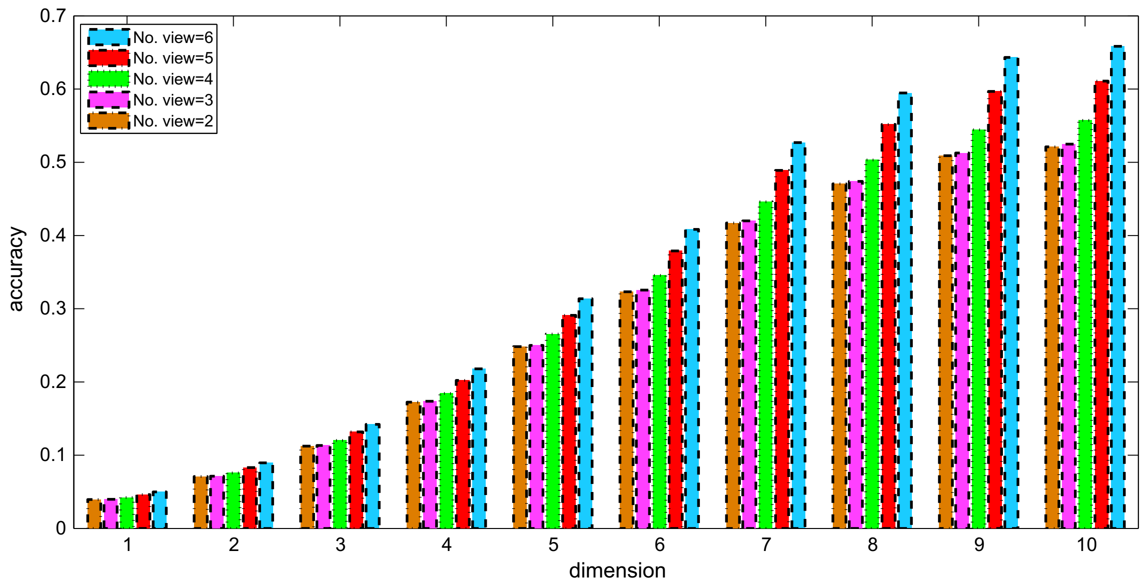


Fig. 8 The average accuracy of MVSSDR under different numbers of views and dimensions when Mfeat and WUSM used

are at least twice than others. This indicates that with WUSM used, it can label instances accurately further. (3) Under the same number of view, a higher dimension brings a higher performance. (4) Similarly, under the same number of dimension, more views used brings a better performance. But it is found that when the view number is 3, the performance is slight better than the performance when the number of view is 2. For this phenomenon, we find the reason depends on the used data set. For Mfeat which is a data set with six views, if we want to label the instances with a high accuracy, we need at least information from four views. Only two or three views used won't bring a great improvement on performance. (5) Even though under few cases, when the number of dimension is fixed, a larger percentage cannot bring a better performance, but on average, when more training instances are labeled, the performances are better. (6) All experimental results validate that a dimensionality reduction learning machine can achieve better performances by fusing different kinds of features from more views. When WUSM is adopted, the performances of the machine are boosted further.

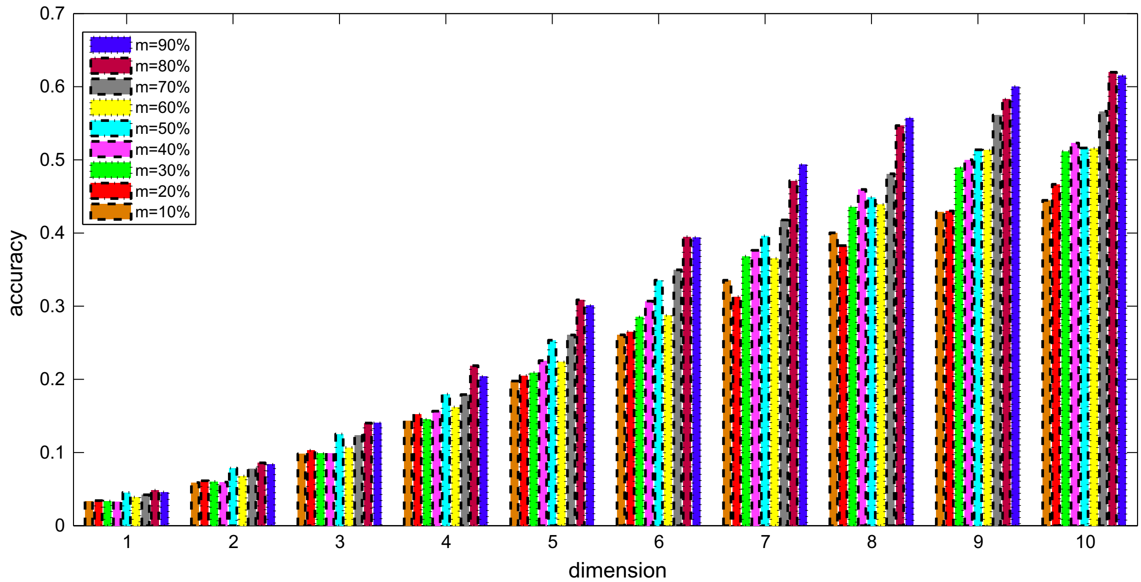


Fig. 9 The average accuracy of MVSSDR under different percentages of labeled training instances and dimensions when Mfeat and WUSM used

4.2.4 Classification performances comparisons

Here we adopt Corel and learning machines MvSs-Zhu (Zhu et al. 2016), MVML (Sun and Zhang 2011), co-graph (Du et al. 2013), AMVS (Yang et al. 2014), MVAR (Tao et al. 2017), co-labeling (Xu et al. 2016) for experiments. Then we show the accuracy comparisons with different Universum construction ways, percentages of labeled training instances, and learning machines. Figure 10 shows the corresponding accuracies comparisons. From this figure, some conclusions are given as follows. (1) On average, the rank of different Universum construction ways is WUSM > CIBU-Li > CIBU-Zhu > Uv-TSVM > Original. But on some cases, for example, with MvSs-Zhu adopted, when the percentages of labeled training instances are 60% and 70%, the performance of CIBU-Li is not worse than that of WUSM. Another example, with MVAR used, when the percentage of labeled training instances is 90%, Uv-TSVM performs better than CIBU-Li. As we know, since we select instances for training and labeling in random, so sometimes, if the selected labeled instances can provide enough useful prior knowledge, they can also boost the performances of other compared Universum construction ways. (2) When the percentages of labeled training instances are higher, the accuracy of each learning machine is better. This is very easy to understand. As we know, labeled instances can provide useful prior knowledge and guide the training and designing of a learning machine. Thus, more labeled instances can provide more useful prior knowledge and the performances of learning machines can boost.

4.2.5 Clustering performances comparisons

In order to show the effectiveness of WUSM on the task of clustering, we adopt Corel, learning machines SULF (Jiang et al. 2014) and SSTC (Yu et al. 2017) for experiments. The measure indexes include clustering accuracy (CA) and normalized mutual information (NMI) and their definitions can refer to Jiang et al. (2014). In generally speaking, a larger CA and NMI indicate a better clustering performance. Then, we use Table 9 to show the related experimental results. According to this table, we can draw the following conclusions. (1) The rank of different Universum construction ways is same as before. Namely, our proposed WUSM still outperforms others. (2) Compared with the 'Original' case, the performance under the 'WUSM' case has a 10% improvement at least. This result indicates with the usage of weights of views and features and providing more useful prior knowledge, instances in an area are more representative and the accuracy of clustering is higher. (3) For WUSM, the average standard deviations of CA and NMI are small and this means the performance of WUSM is stable. As a summary, WUSM can enhance the ability of clustering effectively and compared with other Universum construction ways, WUSM brings a higher improvement in terms of CA and NMI.

5 Further discussion

In this section, we provide some further discussions as follows.

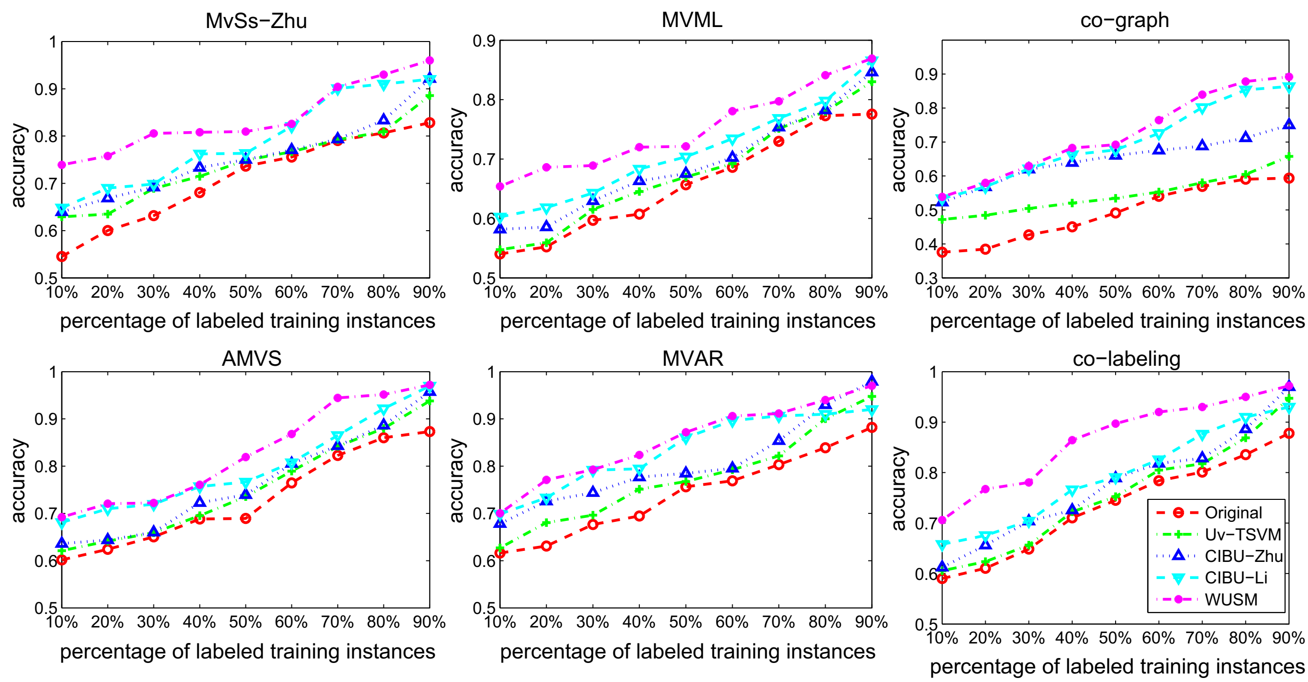


Fig. 10 Accuracy comparisons with different Universum construction ways and percentages of labeled training instances under different classification learning machines when Corel is used

Table 9 CA (average std) and NMI (average std) on Corel with SULF and SSTC and different Universum construction ways used

	WUSM	CIBU-Li	CIBU-Zhu	Uv-TSVM	Original
SULF-CA	0.712 (0.009)	0.681 (0.011)	0.664 (0.003)	0.639 (0.001)	0.622 (0.029)
SULF-NMI	0.660 (0.019)	0.627 (0.020)	0.613 (0.007)	0.608 (0.017)	0.601 (0.024)
SSTC-CA	0.735 (0.037)	0.703 (0.008)	0.692 (0.014)	0.644 (0.025)	0.633 (0.028)
SSTC-NMI	0.681 (0.008)	0.662 (0.024)	0.652 (0.009)	0.613 (0.004)	0.611 (0.028)

5.1 Significance analysis

In order to check whether the differences between WUSM and other compared methods are significant, we adopt Friedman–Nemenyi statistical test (Demsar 2006) and Wilcoxon rank sum test (Barros et al. 2018).

First, in terms of Friedman–Nemenyi statistical test, Table 10 shows the average ranks of WUSM, CIBU-Li, CIBU-Zhu, Uv-TSVM, and without Universum construction ways (i.e., original semi-supervised multi-view data set and using 'Original' to represent such a case) according to all experimental results including the given ones and not given ones (some experimental results are not given since we find the results are similar with the given ones). Rank differences between WUSM and others are also given. For the need of comparison, we regard a semi-supervised multi-view learning machine as a problem and we show the ranks of WUSM, CIBU-Li, CIBU-Zhu, Uv-TSVM, and original on different problems. For the Friedman–Nemenyi test, we treat each problem as a 'data set' and finally, we give the average ranks. Then with the results of this table and refer to Demsar (2006), we first carry out Friedman test and Friedman statistic $\chi_F^2 = \frac{12\times 11}{5(5+1)}[1.0000^2 + 2.0545^2 + 2.9477^2 + 4.0023^2 + 4.9909^2 - \frac{5(5+1)^2}{4}] = 43.2855$ and $F_F = \frac{(11-1)\chi_F^2}{11(5-1)-\chi_F^2} = 605.8153$, further, $F_{0.05}(5-1, (5-1)(11-1)) = F_{0.05}(4, 40) = 2.6060$ and $F_{0.10}(5-1, (5-1)(11-1)) = F_{0.10}(4, 40) = 2.0909$. Since $F_F > 2.6060$ and $F_F > 2.0909$, so we can reject the null hypothesis and say the differences between all compared methods on multiple 'data sets' are significant. Then we carry out Nemenyi test for pairwise comparisons. The critical value for Universum construction ways at $q_{0.05}$ is 2.728 and corresponding critical difference (CD) is $2.728\sqrt{\frac{5\cdot(5+1)}{6\cdot 11}} = 1.8392$ while the one at $q_{0.10}$ is 2.459 and corresponding CD is $2.459\sqrt{\frac{5\cdot(5+1)}{6\cdot 11}} = 1.6579$. Since the rank difference between WUSM and CIBU-Zhu (Uv-TSVM, Original) is larger than 1.8392 and 1.6579, so we can say the performance of WUSM is better than the one of CIBU-Zhu (Uv-TSVM, Original) and their difference is significant. When for the difference between WUSM and CIBU-Li, the difference is not signifi-

Table 10 Average ranks of WUSM, CIBU-Li, CIBU-Zhu, Uv-TSVM, and original on different problems

Problem	WUSM	CIBU-Li	CIBU-Zhu	Uv-TSVM	Original
SmVR	1.0000	2.0000	3.0000	4.0000	5.0000
MLSFS	1.0000	2.0500	2.9500	4.0000	5.0000
MVSSDR	1.0000	2.1000	2.9000	4.0250	4.9750
MvSs-Zhu	1.0000	2.0000	3.0000	4.0000	5.0000
MVML	1.0000	2.1500	2.8500	4.0000	5.0000
Co-graph	1.0000	2.0250	2.9750	4.0000	5.0000
AMVS	1.0000	2.0750	2.9250	4.0000	5.0000
MVAR	1.0000	2.2000	2.8000	4.0250	4.9250
Co-labeling	1.0000	2.0000	3.0250	3.9750	5.0000
SULF	1.0000	2.0000	3.0000	4.0000	5.0000
SSTC	1.0000	2.0000	3.0000	4.0000	5.0000
Average	1.0000	2.0545	2.9477	4.0023	4.9909
Rank difference	–	$2.0545 - 1 = 1.0545$	$2.9477 - 1 = 1.9477$	$4.0023 - 1 = 3.0023$	$4.9909 - 1 = 3.9909$

cant. But in generally, from all experiments, we can validate the effectiveness of WUSM from an average point.

Second, in terms of Wilcoxon rank sum test which is a nonparametric test, we first let 'learning machine'-'measure index' be a problem, for example, SmVR-AvP indicates the case in Sect. 4.2.1. Then we compute the ranksum according to the 'measure index' between WUSM and the compared Universum construction way (for example, Uv-TSVM). The significance level is 5%. Table 11 shows the related experimental results. In this table, each element in this table represents the ranksum between WUSM and the compared Universum construction way for a problem. For example, in the first row and second column, value '0.0083' indicates that with SmVR used, according to the results of AvP, the ranksum between WUSM and Uv-TSVM is '0.0083' and the ranksum rejects the null hypothesis of equal medians at the default 5% significance level. In other words, according to AvP, with SmVR used, the difference between WUSM and Uv-TSVM is significant. According to this table, we can draw a conclusion that in most cases, the ranksums between WUSM and the compared Universum construction ways are smaller than 0.05 and their differences are significant. For those cases which are not significant, we can also validate that the difference between WUSM and CIBU-Li is not significant again.

5.2 Rademacher complexity analysis

Rademacher complexity is a widely used evaluation criteria for a learning machine (Bartlett et al. 2002; Koltchinskii 2001; Koltchinskii and Panchenko 2000; Mendelson 2002) and it can be used to measure the generalization risk bound for learning machines in qualitative analysis (Schölkopf et al. 1999; Wang et al. 2012). The classical risk bound theory (Vapnik and Chervonenkis 1971) is given in the following equation.

$$P(\varphi \neq g(x)) \leq \hat{P_N}(\varphi \neq g(x)) + c\sqrt{\frac{VC(F)}{N}} \tag{2}$$

where $VC(F)$ represents the Vapnik-Chervonekis dimension of F and $\hat{P_N}$ represents the empirical risk error of the function g. Here, F is a $\{\pm 1\}$-valued function class and g is a subset of class F on the data set $\{x_i, \varphi_i\}_{i=1}^N$ where $x_i \in \mathbb{R}^d$ and its class label is $\varphi_i \in \{+1, -1\}$. Actually, the function of VC(F) is used to measure the complexity of the class function F. The Rademacher complexity is an alternative notion of VC(F) (Koltchinskii 2001) which is given in the following equation.

$$P(\varphi \neq g(x)) \leq \hat{P_N}(\varphi \neq g(x)) + \frac{R_N(F)}{2} + \sqrt{\frac{\ln(1/\delta)}{2N}} \tag{3}$$

In Eq. (3), P is a probability distribution on $\chi \times \{\pm 1\}$ and $\{x_i, \varphi_i\}_{i=1}^N$ is chosen independently according to P. F is also a $\{\pm 1\}$-valued function class while the domain of F is χ, with probability at least $1-\delta$ over $\{x_i, \varphi_i\}_{i=1}^N$. g is a function which belongs to F. The $R_N(F)$ is the generalization risk bound of F with the Rademacher complexity. The Rademacher complexity is defined with Eqs. (4) and (5) (Koltchinskii 2001; Wang et al. 2012).

$$\hat{R}_N(F) = \mathbf{E}\left[\sup_{g \in F} |\frac{2}{N}\sum_{i=1}^{N} \sigma_i g(x_i) \parallel x_1, \ldots, x_N\right] \tag{4}$$

where $\mathbf{E}$ is the operator of the expected value of a random variable. Here, $g(x_i)$ is the classification function of F, $\{\sigma_i\}_{i=1}^N$ is a set whose members are chosen from $\{1, -1\}$ arbitrarily, and x_is represent independent instances selected

Table 11 Wilcoxon rank sum test analysis with different problems. The case which the difference between WUSM and the compared Universum construction way is not significant is shown in bold

	Original	Uv-TSVM	CIBU-Zhu	CIBU-Li
SmVR-AvP	0.0339	0.0083	0.0031	**0.0542**
SmVR-AUC	0.0377	0.0368	0.0198	0.0440
MLSFS-precision	0.0005	0.0005	0.0012	0.0106
MLSFS-recall	0.0056	0.0040	0.0040	0.0142
MLSFS-specificity	0.0244	0.0142	0.0019	0.0011
MLSFS-accuracy	0.0028	0.0008	0.0028	0.0056
MLSFS-F-measure	0.0056	0.0040	0.0142	0.0244
MVSSDR-precision	<0.0001	<0.0001	<0.0001	<0.0001
MVSSDR-recall	0.0142	<0.0001	0.0008	<0.0001
MVSSDR-specificity	0.0003	<0.0001	<0.0001	<0.0001
MVSSDR-accuracy	0.0002	<0.0001	<0.0001	<0.0001
MVSSDR-F-measure	<0.0001	<0.0001	<0.0001	<0.0001
MvSs-Zhu-accuracy	0.0106	0.0315	0.0006	0.0066
MVML-accuracy	0.0171	0.0012	0.0324	**0.0535**
Co-graph-accuracy	0.0012	0.0090	0.0123	**0.0586**
AMVS-accuracy	0.0021	0.0063	0.0109	0.0181
MVAR-accuracy	0.0000	0.0002	0.0286	0.0197
Co-labeling-accuracy	0.0315	0.0042	0.0049	0.0050
SULF-CA	<0.0001	<0.0001	0.0717	0.0134
SULF-NMI	0.0021	0.0040	0.0002	0.0106
SSTC-CA	<0.0001	<0.0001	0.0003	<0.0001
SSTC-NMI	0.0385	<0.0001	0.0070	**0.0635**

from χ according to the probability distribution P. Then the Rademacher complexity of F is

$$R_N(F) = \mathbf{E}\hat{R}_N(F) \tag{5}$$

In general, a smaller Rademacher complexity indicates a better performance. Here, we use Table 12 to show the Rademacher complexities of different semi-supervised multi-view learning machines with different Universum construction ways used. According to this table, it is found that WUSM can make a semi-supervised multi-view learning machine have a smaller Rademacher complexity and this means that the learning machine will have a better performance in average.

5.3 What kind of Universum set is more effective

As we know, there are three simple ways to construct Universum sets. Different ways can construct diverse Universum sets, and what kind of Universum set is more effective is necessary to be discussed. Here, we further discuss each way and elaborate its advantages or disadvantages.

The first way to construct Universum set is selecting instances from some other classes that are known unlikely to belong to any of the target classes. The representative learning machine is U-SVM and its kernel version (Weston et al. 2006). For this way, the construction operation seems to be timesaving due to we need not to average anything, just select some instances. But how to select feasible Universum instances from non-target classes should be considered. If we select some instances with a wrong way, the performance of a learning machine will be affected and decreased. See Fig. 11, we use a kernel-based U-SVM (i.e., U-SVM adopts kernel functions to process nonlinear classification tasks) for experiments and the black solid line represents the testing decision boundary. Instances are constructed in random. Then, according to this figure, it is found that selection of Universum instances from non-target classes is a key to get a feasible testing decision boundary. Good Universum set brings a better testing decision boundary while bad Universum set affects the testing decision boundary to a certain extent. This phenomenon is also mentioned in Chen and Zhang (2009). Moreover, for this way, if the instances from non-target classes are insufficient, for example, only two or three instances in non-target classes, and then Universum set construction ways of this way will not work well. Because few instances cannot provide enough prior knowledge.

The second way is selecting instances in random to construct Universum set. The representative learning machine is FIBU (Chen and Zhang 2009) and FIBU selects informative instances, i.e., IBU instances which deposit in between the two different classes to construct Universum set. Com-

Table 12 Rademacher complexities of different semi-supervised multi-view learning machines and Universum construction ways used. The best Rademacher complexity for each semi-supervised multi-view learning machine is given in bold

	WUSM	CIBU-Li	CIBU-Zhu	Uv-TSVM	Original
SmVR	**0.1203**	0.1220	0.1421	0.1427	0.1510
MLSFS	**0.1063**	0.1153	0.1184	0.1226	0.1314
MVSSDR	**0.0461**	0.0477	0.0499	0.0512	0.0522
MvSs-Zhu	**0.0712**	0.0736	0.0745	0.0773	0.0809
MVML	**0.0831**	0.0956	0.0968	0.0983	0.1011
Co-graph	**0.0732**	0.0778	0.0782	0.0849	0.0867
AMVS	**0.1312**	0.1390	0.1457	0.1540	0.1602
MVAR	**0.0612**	0.0705	0.0791	0.0832	0.0891
Co-labeling	**0.0681**	0.0782	0.0905	0.1072	0.1085
SULF	**0.0781**	0.0834	0.0953	0.1015	0.1028
SSTC	**0.0512**	0.0555	0.0647	0.0712	0.0771

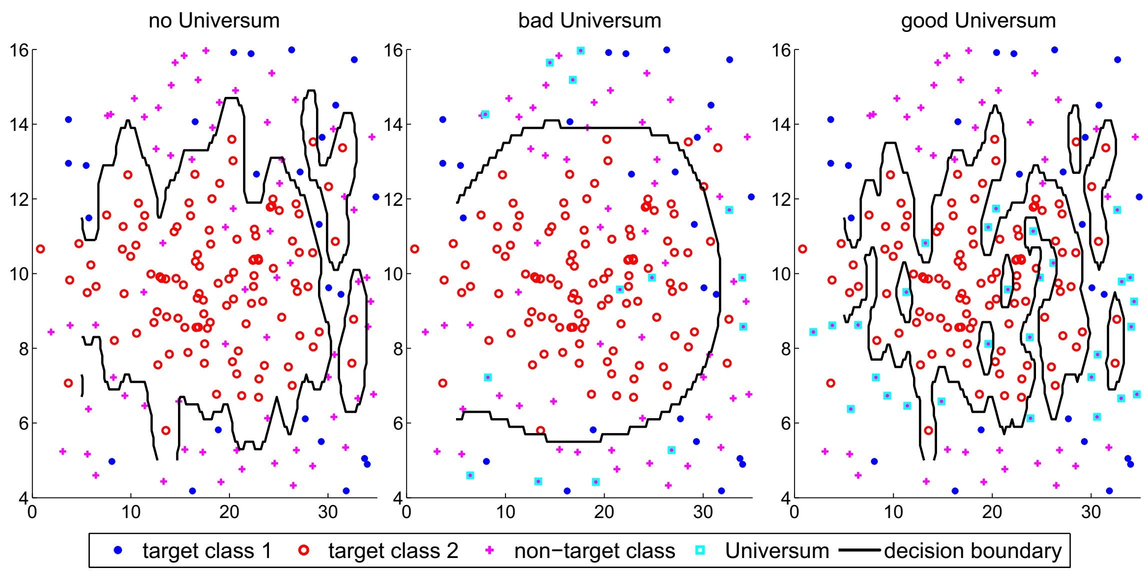


Fig. 11 Testing decision boundaries of kernel-based U-SVM with three cases on a set of randomly constructed instances. Left: no Universum instance is selected; middle: some instances from the non-target classes are selected as Universum ones while the selection is not good; (3) right: some instances from the non-target classes are selected as Universum ones and the selection is good

pared with the first way, this operation avoids the selection of bad Universum instances. Moreover, this operation makes the number of IBU be small compared to that of the whole non-target classes. But this way still exists a latent danger. If all constructed instances locate very closer to the target instances, then they will be nonsense. See Fig. 12. We construct the instances in random and adopt a variant of kernel-based U-SVM (i.e., the Universum instances are selected by FIBU) to classify these instances. According to this figure, we can see if the construction is not good, the obtained testing decision boundary will not work well.

For the third way, there are two sub-kinds. One is randomly selecting a pair of positive and negative training instances, and averaging them, for example, RA (Cherkassky and Dai 2009) and RPNA (random positive and negative average) (Liu et al. 2014). The other is selecting a pair of neighbor training instances in random and averaging them, for example, CIBU-Li (Li et al. 2017) and CIBU-Zhu (Zhu 2016). Since the averaging operation makes the Universum instance locate in the center rather than nearby of two instances, thus both of them can avoid the disadvantages of Universum set construction ways of the second way. But methods of the third way introduce the prior knowledge of a pair of instances each time, this kind of prior knowledge is not sufficient.

For our WUSM, it also adopts the averaging operation. Different from the methods of the third way, WUSM averages the center and an instance. The center is depended on all (labeled, unlabeled) instances. This means our WUSM can introduce much more prior knowledge at once.

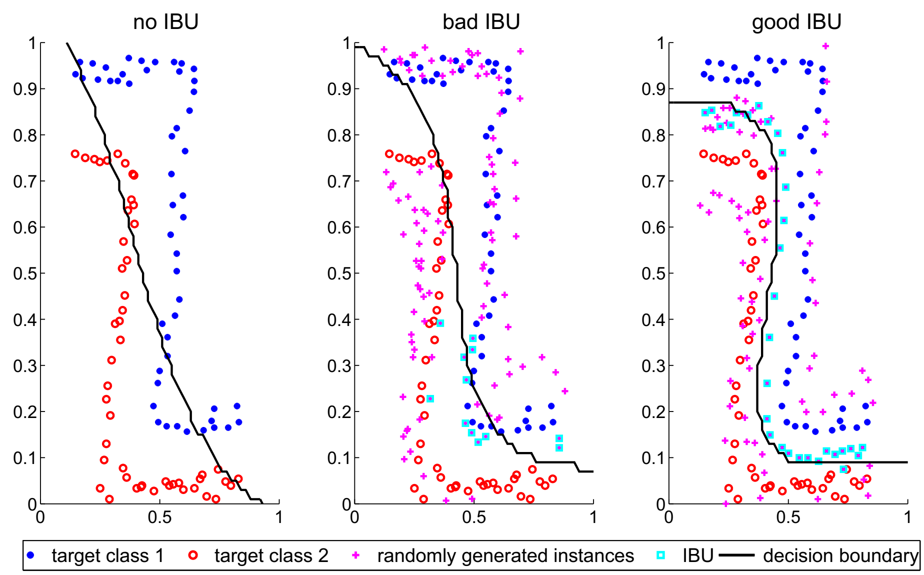


Fig. 12 Testing decision boundaries of a variant of kernel-based U-SVM (Universum instances are selected by FIBU) with three cases on a set of randomly constructed instances. Left: no IBU instance is constructed and selected; middle: some IBU instances are constructed and selected while the selection is not good; right: some IBU instances are constructed and selected and the selection is good

According to the discussion, we can draw a conclusion that our WUSM is more effective than other compared ways. Related previous experimental results can also validate it.

6 Conclusions and future work

Semi-supervised multi-view learning machines are easy to be affected by only few labeled instances which provide limited prior knowledge. Thus some scholars have developed Universum learning to construct Universum sets so as to add more prior knowledge. While the traditional construction ways neglect to consider the various discriminant roles of views and features. Thus this work considers their weights, designs some schemes to construct Universum set with the introduction of operation of averaging and notion of center so as to overcome the disadvantages of some traditional Universum construction ways.

The new developed learning machine is named weight-and-Universum-based semi-supervised multi-view learning machine (WUSM). The procedure of WUSM consists of three steps. First is obtaining weights of views and features by an effective weighted multi-view clustering approach WMVC, second is constructing Universum set by some schemes which introduce operation of averaging and notion of center, and third is applying the whole data set including constructed Universum set to a semi-supervised multi-view learning machine.

The novelty of our proposed WUSM is that (1) it considers the information of all (labeled, unlabeled) instances to construct Universum set and introduce more useful prior knowledge; (2) it assigns feasible weights for views and features and considers their diverse discriminant roles; (3) this is the first trial to construct Universum set with the combination of averaging all (labeled, unlabeled) instances and the feasible weights of views and features. Our work advances the development of semi-supervised multi-view learning machines.

Related experiments on three multi-view data sets Mfeat, Reuters, and Corel validate the usefulness of WUSM in different fields including bipartite ranking, feature selection, dimensionality reduction, classification, and clustering. Furthermore, discussions about significance, Rademacher complexity, etc. also validate the effectiveness of WUSM.

Although our WUSM performs better in some fields, we should notice its limitations. First, with the coming of big-data age, more and more data sets are generated in every minute and should be online processed with limitation of the storage. This cannot be solved by the current version of WUSM. Second, deep learning which is a good method to extract more features is not used in our work and this limits the improvement in the performance of WUSM. Thus, according to these limitations, our future directions should cover the following parts. First, we should propose an online-version WUSM so as to process the data generated all the time. Second, in our future work, we want to combine deep learning with WUSM so as to extract more useful features and enhance the performance of the semi-supervised multi-view learning machines.

Acknowledgements This work is supported by Project funded by China Postdoctoral Science Foundation under Grant Number 2019M651576, National Natural Science Foundation of China (Grant Nos. 61602296, 61673301), Natural Science Foundation of Shanghai (Grant Nos. 16ZR1414500), National Key R&D Program of China (Grant No. 213), Major Project of Ministry of Public Security (Grant No. 20170004). Furthermore, this work is also sponsored by 'Chenguang Program' supported by Shanghai Education Development Foundation and Shanghai Municipal Education Commission under Grant Number 18CG54. The authors would like to thank their supports.

Compliance with ethical standards

Conflict of interest The authors declare that they have no conflict of interest.

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

Informed consent Informed consent was obtained from all individual participants included in the study.

References

Amini MR, Usunier N, Goutte C (2009) Learning from multiple partially observed views-an application to multilingual text categorization. In: Neural information processing systems (NIPS), pp 28–36

Asuncion A, Newman D (2007) UCI machine learning repository. http://archive.ics.uci.edu/ml/. Accessed Sept 2019

Barros RSMD, Hidalgo JIG, Cabral DRDL (2018) Wilcoxon rank sum test drift detector. Neurocomputing 275:1954–1963

Bartlett P, Boucheron S, Lugosi G (2002) Model selection and error estimation. Mach Learn 48:85–113

Chen SC, Zhang CS (2009) Selecting informative Universum sample for semi-supervised learning. In: International joint conferences on artificial intelligence, pp 1016–1021

Chen SC, Wang Z, Tian YJ (2007) Matrix-pattern-oriented Ho–Kashyap classifier with regularization learning. Pattern Recognit 40(5):1533–1543

Chen XH, Chen SC, Xue H (2012) Universum linear discriminant analysis. Electron Lett 48(22):1407–1409

Chen XH, Yin HJ, Jiang F, Wang LP (2018) Multi-view dimensionality reduction based on Universum learning. Neurocomputing 275:2279–2286

Cherkassky V, Dai W (2009) Empirical study of the Universum SVM learning for high-dimensional data. In: International conference on artificial neural networks, pp 932–941

Demsar J (2006) Statistical comparisons of classifiers over multiple data sets. J Mach Learn Res 7(1):1–30

Deng MQ, Wang C, Chen QF (2016) Human gait recognition based on deterministic learning through multiple views fusion. Pattern Recognit Lett 78(C):56–63

Dhar S (2014) Analysis and extensions of Universum learning. Dissertations and Theses, Grad works, University of Minnesota, ProQuest Dissertations Publishing

Du YT, Li Q, Cai ZM, Guan XH (2013) Multi-view semi-supervised web image classification via co-graph. Neurocomputing 122:430–440

Epshteyn A, DeJong G (2006) Generative prior knowledge for discriminative classification. AI Access Foundation 27(27):25–53

Han C, Chen J, Wu QY, Mu S, Min HQ (2015) Sparse Markov chain-based semi-supervised multi-instance multi-label method for protein function prediction. J Bioinf Comput Biol 13(5):1543001

Hou CP, Zhang CS, Wu Y, Nie FP (2010) Multiple view semi-supervised dimensionality reduction. Pattern Recognit 43(3):720–730

http://multilingreuters.iit.nrc.ca/ReutersMultiLingualMultiView.htm

Huang G, Song S, Gupta JND, Wu C (2014) Semi-supervised and unsupervised extreme learning machine. IEEE Trans Cybern 44(12):2405–2417

Jiang Y, Liu J, Li ZC, Lu HQ (2014) Semi-supervised unified latent factor learning with multi-view data. Mach Vis Appl 25(7):1635–1645

Koltchinskii V (2001) Rademacher penalties and structural risk minimization. IEEE Trans Inf Theory 47(5):1902–1914

Koltchinskii V, Panchenko D (2000) Rademacher processes and bounding the risk of function learning. High Dimens Probab II:443–459

Li DD, Zhu YJ, Wang Z, Chong CY, Gao DQ (2017) Regularized matrix-pattern-oriented classification machine with Universum. Neural Process Lett 45(3):1077–1098

Liu CL, Hsaio WH, Lee CH, Gou FS (2014) Semi-supervised linear discriminant clustering. IEEE Trans Cybern 44(7):989–1000

Liu DL, Tian YJ, Bie RF, Shi Y (2014) Self-Universum support vector machine. Pers Ubiquitous Comput 18:1813–1819

Liu CL, Hsaio WH, Lee CH, Chang TH, Kuo TH (2016) Semi-supervised text classification with Universum learning. IEEE Trans Cybern 46(2):462–473

Mendelson S (2002) Rademacher averages and phase transitions in Glivenko–Cantelli classes. IEEE Trans Inf Theory 48(1):251–263

Nie F, Xu D, Li X, Xiang S (2011) Semi-supervised dimensionality reduction and classification through virtual label regression. IEEE Trans Syst Man Cybern B Cybern 41(3):675–685

Peng B, Qian G, Ma YQ (2008) View-invariant pose recognition using multilinear analysis and the Universum. Adv Vis Comput 5359:581–591

Schölkopf B, Shawe-Taylor J, Smola AJ, Gmd BS, Smola EJ, Williamson RC (1999) Generalization bounds via eigenvalues of the gram matrix. Technical report 99-035, NeuroColt

Schölkopf B, Simard P, Smola AJ, Vapnik V (1997) Prior knowledge in support vector kernels. In: Neural information processing systems (NIPS), pp 640–646

Seliya N, Khoshgoftaar TM (2007) Software quality analysis of unlabeled program modules with semisupervised clustering. IEEE Trans Syst Man Cybern A Syst Hum 37(2):201–211

Sheikhpour R, Sarram MA, Gharaghani S, Chahooki MAZ (2017) A survey on semi-supervised feature selection methods. Pattern Recognit 64:141–158

Shen C, Wang P, Shen F, Wang H (2012) UBoost: boosting with the Universum. IEEE Trans Pattern Anal Mach Intell 34(4):825–832

Shi CJ, Ruan QQ, An GY, Ge C (2015) Semi-supervised sparse feature selection based on multi-view Laplacian regularization. Image Vis Comput 41:1–10

Sinz F, Chapelle O, Agarwal A, Schölkopf B (2008) An analysis of inference with the Universum. In: Advances in neural information processing systems (NIPS2008), pp 1369–1376

Sun SL, Zhang QJ (2011) Multiple-view multiple-learner semi-supervised learning. Neural Process Lett 34:229–240

Tao H, Hou CP, Nie FP, Zhu JB, Yi DY (2017) Scalable multi-view semi-supervised classification via adaptive regression. IEEE Trans Image Process 26(9):4283–4296

Tzortzis G, Likas A (2012) Kernel-based weighted multi-view clustering. In: 2012 IEEE 12th international conference on data mining, pp 675–684

Usunier N, Amini MR, Goutte C (2011) Multiview semi-supervised learning for ranking multilingual documents. In: Gunopulos D, Hofmann T, Malerba D, Vazirgiannis M (eds) Machine learning and knowledge discovery in databases. ECML PKDD 2011. Lecture notes in computer science, vol 6913. Springer, Berlin, Heidelberg, pp 443–458

Vapnik V, Chervonenkis A (1971) On the uniform convergence of relative frequencies of events to their probabilities. Theory Probab Appl 16(2):264–280

Vapnik V, Kotz S (1982) Estimation of dependences based on empirical data. Springer, Berlin

Wang F (2011) Semisupervised metric learning by maximizing constraint margin. IEEE Trans Syst Man Cybern B Cybern 41(4):931–939

Wang Z, Xu J, Chen SC, Gao DQ (2012) Regularized multi-view machine based on response surface technique. Neurocomputing 97:201–213

Wang HY, Wang X, Zheng J, Deller JR, Peng HY, Zhu LQ, Chen WG, Li XL, Liu RJ, Bao HJ (2014) Video object matching across multiple non-overlapping camera views based on multi-feature fusion and incremental learning. Pattern Recognit 47(12):3841–3851

Wang Z, Zhu YJ, Liu WW, Chen ZH, Gao DQ (2014) Multi-view learning with Universum. Knowl Based Syst 70:376–391

Weston J, Collobert R, Sinz F, Bottou L, Vapnik V (2006) Inference with the Universum. In: The 23rd international conference on machine learning, pp 1009–1016

Wu F, Jing XY, You XG, Yue D, Hu RM, Yang JY (2016) Multi-view low-rank dictionary learning for image classification. Pattern Recognit 50:143–154

Xu XX, Li W, Xu D, Tsang IW (2016) Co-labeling for multi-view weakly labeled learning. IEEE Trans Pattern Anal Mach Intell 38(6):1113–1125

Xu YM, Wang CD, Lai JH (2016) Weighted multi-view clustering with feature selection. Pattern Recognit 53:25–35

Xu YT, Chen M, Yang ZJ, Li GH (2016) ν-twin support vector machine with Universum data for classification. Appl Intell 44(4):956–968

Yang ZK, Liu Z, Liu SY, Min L, Meng WT (2014) Adaptive multi-view selection for semi-supervised emotion recognition of posts in online student community. Neurocomputing 144:138–150

Yu H, Wang XC, Wang GY (2017) A semi-supervised three-way clustering framework for multi-view data. In: International joint conference on rough sets, pp 313–325

Zhang CH, Zheng WS (2017) Semi-supervised multi-view discrete hashing for fast image search. IEEE Trans Image Process 26(6):2604–2617

Zhang D, Wang JD, Wang F, Zhang CS (2008) Semi-supervised classification with Universum. In: SIAM international conference on data mining, vol 2(4), pp 323–333

Zhang D, Wang J, Si L (2011) Document clustering with Universum. In: International conference on research and development in information retrieval, pp 873–882

Zhao J, Xu YT, Fujita H (2019) An improved non-parallel Universum support vector machine and its safe sample screening rule. Knowl Based Syst 170:79–88

Zhu CM (2016) Improved multi-kernell classification machine with Nyström approximation technique and Universum data. Neurocomputing 175:610–634

Zhu CM, Gao DQ (2015) Multiple matrix learning machine with five aspects of pattern information. Knowl Based Syst 83:13–31

Zhu SH, Sun X, Jin DL (2016) Multi-view semi-supervised learning for image classification. Neurocomputing 208:136–142